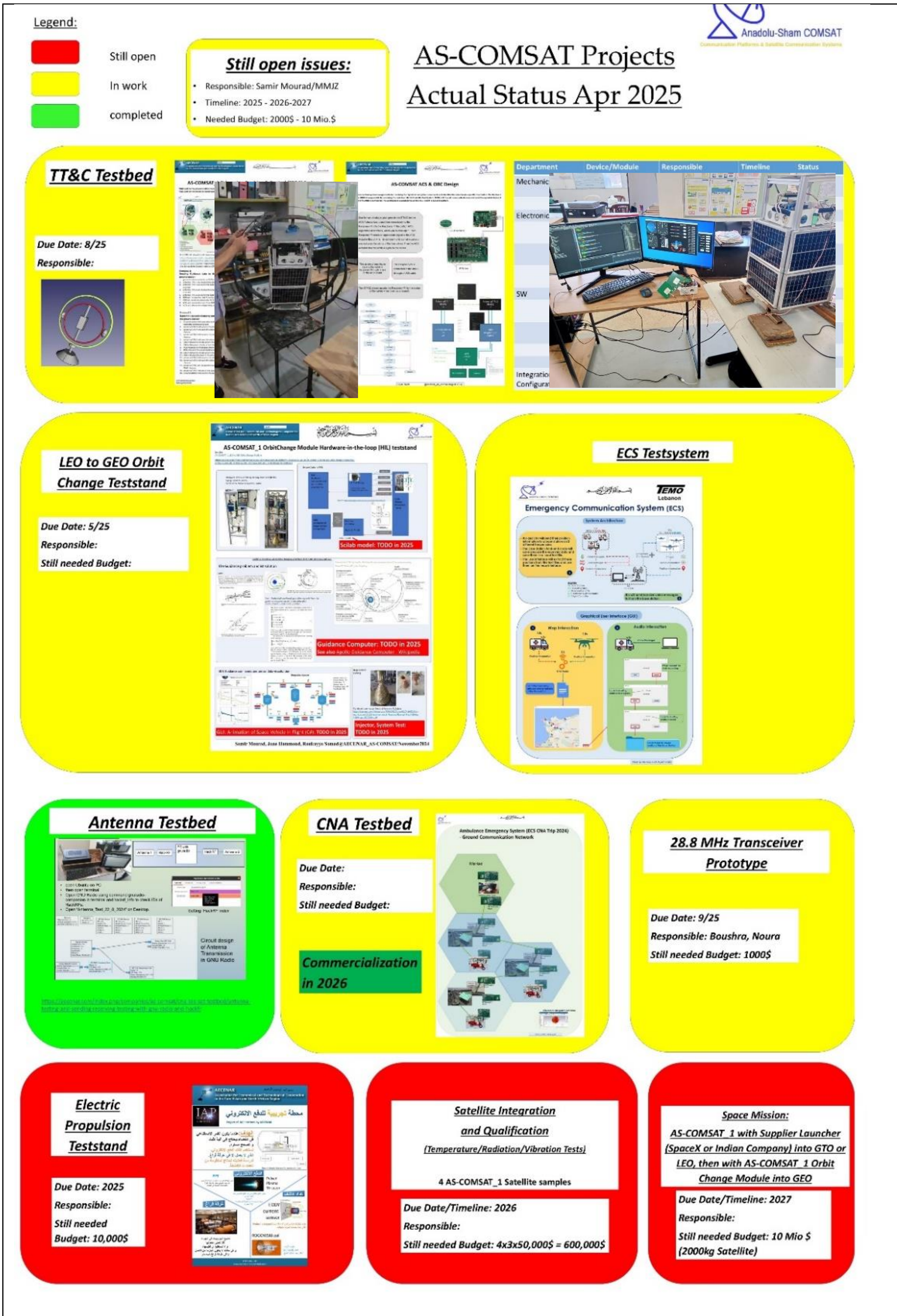
بسم الله الرحمن الرحيم

https://aecenar.com/index.php/companies/as-comsat

AS-COMSAT Final Report (2020 – Apr 2025) *Editor:* Dr. Samir Mourad



With contributions of:
Raja Mourad,
Nour Krayem,
Dr. Samir Mourad,
Siham Aisha,
Ali Asaad,
Mohamad Mourad,
Mohammad Najjarine,
Karim Meemary,
Jana Othman,
Rozan Mustafa,
Dr. Salih Bayar,
Dr. Ömer Korçak,
Dr. Ihsan Çiçek,
Ahmad Dannawi,
Jana Hammoud,
Roukaya Al Samad
Mohammad Ali,
Ahmad Achaji,
Abdullah Munla,
Layla Taleb,
Bushra Elwan,
Noura Kabbara,

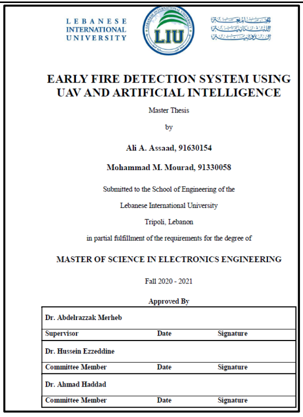
Last Update: 19.04.2025 23:33

In this report all activities concerning AS-COMSAT are documented for the period Oct 2020 – April 2025.

Content

1 List of included Reports and relevant articles from www.aecenar.com

1.1 Included reports


LEBANESE INTERNATIONAL UNIVERSITY

EARLY FIRE DETECTION SYSTEM USING UAV AND ARTIFICIAL INTELLIGENCE

Master Thesis

by

Ali A. Assaad, 91630154

Mohammad M. Mourad, 91330058

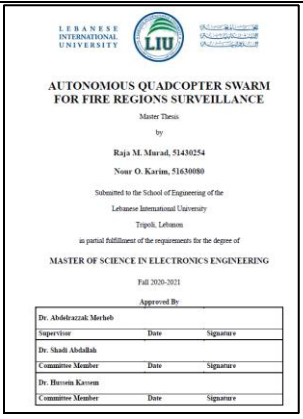
Submitted to the School of Engineering of the Lebanese International University Tripoli, Lebanon in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRONICS ENGINEERING

Fall 2020 - 2021

Approved By

Dr. Abdelrazzak Merheb		
Supervisor	Date	Signature
Dr. Hussein Ezzeddine		
Committee Member	Date	Signature
Dr. Ahmad Haddad		
Committee Member	Date	Signature


LEBANESE INTERNATIONAL UNIVERSITY

AUTONOMOUS QUADCOPTER SWARM FOR FIRE REGIONS SURVEILLANCE

Master Thesis

by

Raja M. Murad, 51430254

Nour O. Karim, 51630080

Submitted to the School of Engineering of the Lebanese International University Tripoli, Lebanon in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRONICS ENGINEERING

Fall 2020-2021

Approved By

Dr. Abdelrazzak Merheb		
Supervisor	Date	Signature
Dr. Shadi Abdallah		
Committee Member	Date	Signature
Dr. Hussein Kassem		
Committee Member	Date	Signature


nomical and Technological Cooperation nd North-African Region

IAP INSTITUT FOR ASTROPHYSICS

Steps for Launching Satellite

Author:

Siham Aisha

Last Update: 27.12.2021


AS-COMSAT-1 - Technical Development Documentation

Requirements, SystemDesign, Mech/HW/SW Design&Realization, Off-the-shelf/supplier parts, System Integration&Testing

Author:

Dr. Samir Mourad

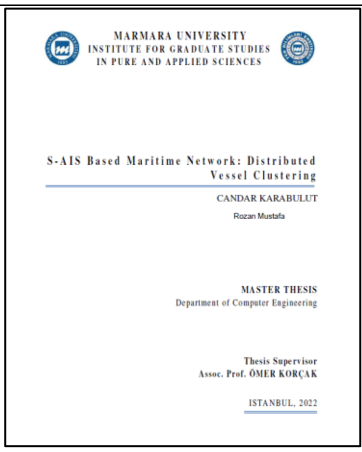
Last update: Sunday, January 23, 2022

Anadolu-Sham COMSAT

Communication Platforms & Satellite Communication Systems

Name of document:

C:\AS-COMSAT\TechnicalDocumentation\230122AS-COMSAT-1_TechnicalDevelopmentDocumentation.docx


MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES
IN PURE AND APPLIED SCIENCES

S-AIS Based Maritime Network: Distributed Vessel Clustering

CANDAR KARABULUT

Rozan Mustafa

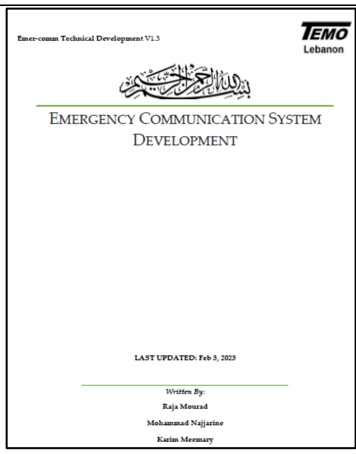
MASTER THESIS

Department of Computer Engineering

Thesis Supervisor

Assoc. Prof. ÖMER KORÇAK

ISTANBUL, 2022


Emer-comm Technical Development V1.3

TEMO Lebanon

EMERGENCY COMMUNICATION SYSTEM DEVELOPMENT

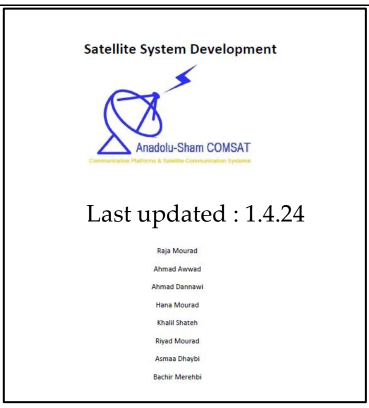
LAST UPDATED: Feb 3, 2023

Written By:

Raja Mourad

Mohammad Najjarine

Karim Meemary


Satellite System Development

Anadolu-Sham COMSAT

Last updated : 1.4.24

Raja Mourad

Ahmad Awwad

Ahmad Dannawi

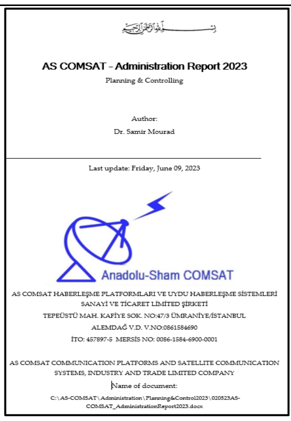
Hana Mourad

Khalil Shateh

Riyad Mourad

Asmaa Dhaybi

Bachir Merehbi


AS COMSAT - Administration Report 2023

Planning & Controlling

Author:

Dr. Samir Mourad

Last update: Friday, June 09, 2023

Anadolu-Sham COMSAT

AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ

TEPEÜSTÜ MAH. KAFİYE SOK. NO:47/3 ÜMRANİYE/İSTANBUL

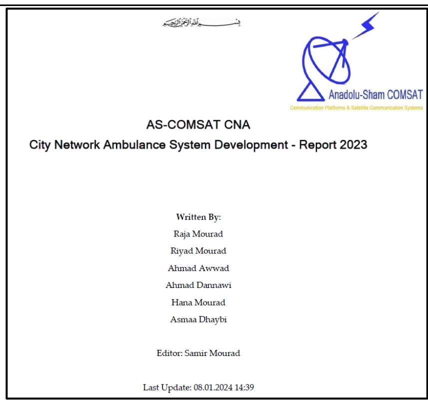
ALEMDAĞ V.D. V.NO:0861584690

İTO: 457897-5 MERSİS NO: 0086-1584-6900-0001

AS COMSAT COMMUNICATION PLATFORMS AND SATELLITE COMMUNICATION SYSTEMS, INDUSTRY AND TRADE LIMITED COMPANY

Name of document:

C:\AS-COMSAT\Administration\Planning&Control2023\020523AS-COMSAT_AdministrationReport2023.docx


Anadolu-Sham COMSAT

AS-COMSAT CNA

City Network Ambulance System Development - Report 2023

Written By:

Raja Mourad

Riyad Mourad

Ahmad Awwad

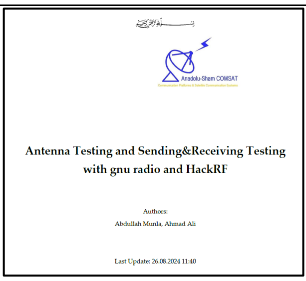
Ahmad Dannawi

Hana Mourad

Asmaa Dhaybi

Editor: Samir Mourad

Last Update: 08.01.2024 14:39

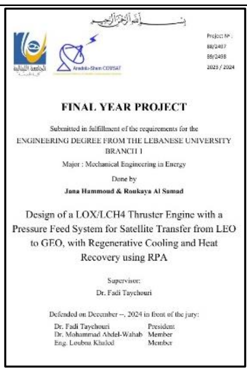

Anadolu-Sham COMSAT

Antenna Testing and Sending&Receiving Testing with gnu radio and HackRF

Authors:

Abdullah Munla, Ahmad Ali

Last Update: 26.08.2024 11:40


Anadolu-Sham COMSAT

FINAL YEAR PROJECT

Submitted in fulfillment of the requirements for the ENGINEERING DEGREE FROM THE LEBANESE UNIVERSITY BRANCH I

Major : Mechanical Engineering in Energy

Done by

Jana Hammoud & Roukaya Al Samad

Design of a LOX/LCH4 Thruster Engine with a Pressure Feed System for Satellite Transfer from LEO to GEO, with Regenerative Cooling and Heat Recovery using RPA

Supervisor:

Dr. Fadi Taychouri

Defended on December --, 2024 in front of the jury:

Dr. Fadi Taychouri President
Dr. Mohammad Abdel-Wahab Member
Eng. Loubna Khaled Member

1.2 Submenues of AS-COMSAT (https://aecenar.com/index.php/companies/as-comsat)

- AS-COMSAT SW&HW Repository
- AS-COMSAT Planning&Controlling
    - TEMO Lebanon 2016 - 2020
        - Ballon/Airship Based Communication Platforms
        - Satellite Based Communication Platforms
        - Management Software
    - AS-COMSAT Planning and Controlling 2022
    - AS-COMSAT Planning&Controlling 2023
        - AS-COMSAT Procurement 2023
        - AS-COMSAT Office&Atelier Istanbul
    - AS-COMSAT Planning & Controlling 2024
- AS-COMSAT Platforms&Devices
    - AS-COMSAT_1 (LEO Communication Satellite)
        - AS-COMSAT_1 (LEO Satellite) System Architecture
        - AS-COMSAT_1 ACS (Design&Realization&Testing)
            - AS-COMSAT_1 ACS Board STM32 SW
            - ACS Board - Ver. 0524
            - AS-COMSAT_1 ACS Sun Sensor
            - AS-COMSAT_1 ACS Teststand (Requirements&Design&Realization)
        - AS-COMSAT_1 Power Management Unit (PMU)
            - AS-COMSAT_1 PMU SW
        - AS-COMSAT_1 LEO Satellite - Structure and Integration
            - AS-COMSAT_1 Space Radiation Protection
        - AS-COMSAT_1 TT&C
            - AS-COMSAT_1 TT&C Ground Station HW
            - AS-COMSAT TT&C GUI
            - TT&C Ground Station and Satellite Transceiver Boards STM32 SW
            - Monitoring values of TT&C Ground Station Transceiver STM32 C Code
        - AS-COMSAT_1 On-Board-Computer (OBC)

- DevOps CI/CD Development Environment (HW, GUI and embedded SW)
- AS-COMSAT Testbeds CNA+LEO-Sat, Antenna
    - Testing of CNA 3-1-2024
    - Reduced Testbed (Defined 8 Jan 2024)
    - Antenna Testing and Sending&Receiving Testing with gnu radio and HackRF
- Launch Issues (SpaceX and other suppliers from India and Russia)

2 AS-COMSAT SW&HW Repository[1]

2.1 Documentation

AS-COMSAT_1 Satellite_HW_SW_Development_Report 2023 (docx)

AS-COMSAT_1 Satellite_HW_SW_Development Report 2023 (pdf)

PMU (Solar Charger) Development Documentation (docx) (last updated: 23.08.23)

CNA System Development Report 2023 (docx)

CNA System Development Report 2023 (pdf)

Antenna Testing Lab Manual (docx) (Aug 2024)

CNA with 1 Gateway, 3 nodes, and n mobile users - Report (Sep 24)

Link to google-drive AS-COMSAT (not actual)

1 https://aecenar.com/index.php/companies/as-comsat/as-comsat-sw-hw-repository

2.2 Posters

AS-COMSAT_1_OrbitChangeHIL_Poster (pptx), Nov 24

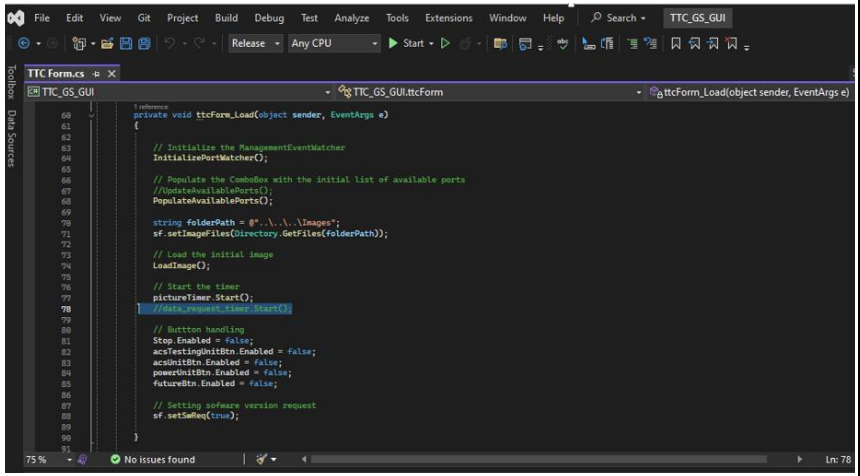
AS-COMSAT_1_TT&C_POSTER (pptx), 10.7.24

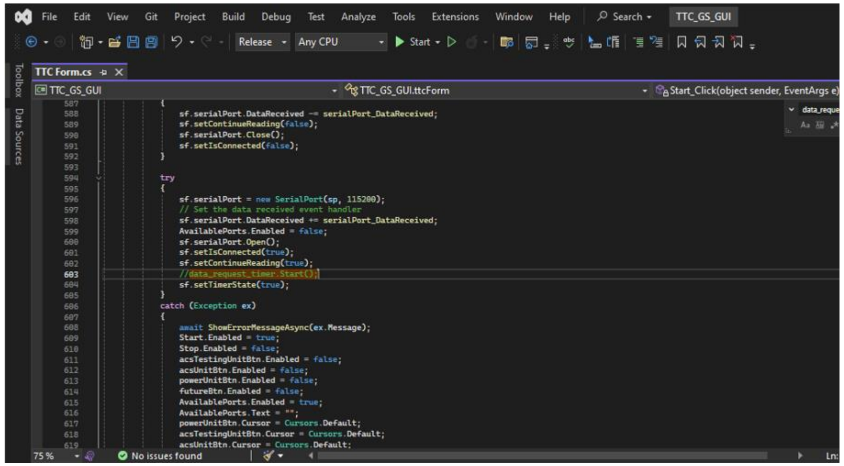
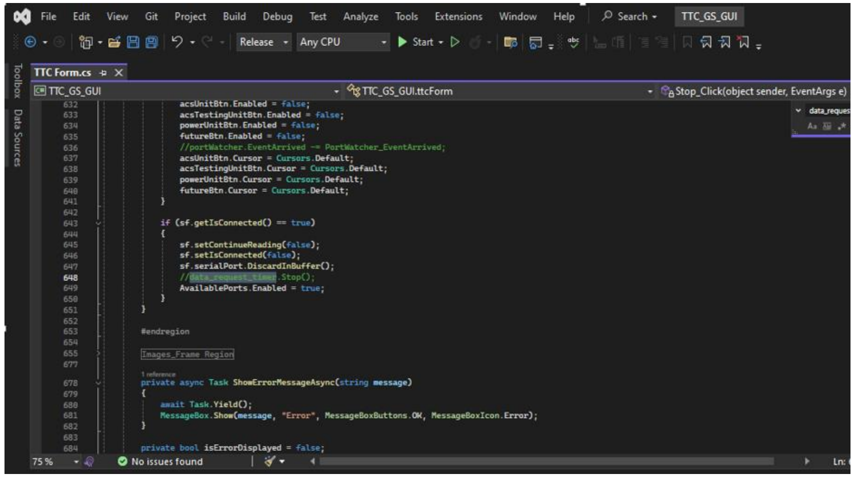
AS-COMSAT_ACS&OBC_POSTER290824_pptx

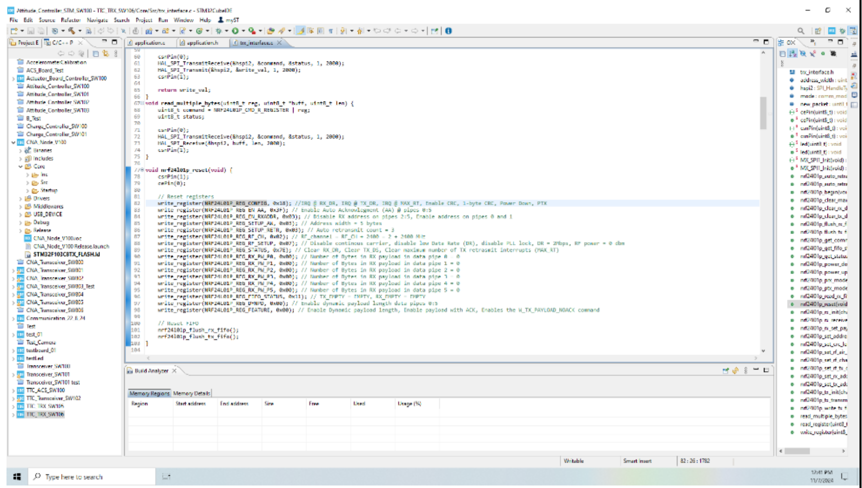
AS-COMSAT CNA 2 nodes and 2 mobile users (GUI Code and STM Code for each of the two transceiver cards) (Poster+Code)

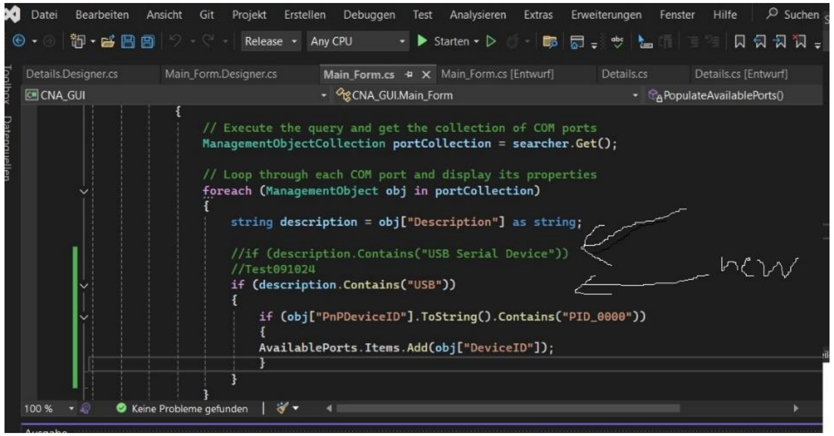
2.3 SW Code Repository

TT&C GUI C#	TT&C GUI C# Ver. 180923 TT&C GUI C# Ver. 200124 TT&C GUI C# Ver. 220124 ACS_Test_Stand_GUI integrated in TT&C_GUI Ver.300124 TTC_GS_GUI_Ver_070224_ACS_Test_Stand_GUI integrated in TT&C_GUI update valve TTC_GS_GUI_V29_100224_ACS_Test_Stand_GUI integrated in TT&C_GUI Debugging TTC_GS_GUI_V_170224 ACS_Test_Stand_GUI integrated in TT&C_GUI open valve 1 to 8 TTC_GS_GUI_V_190224 ACS_Test_Stand_GUI integrated in TT&C_GUI open valve 1 to 9		

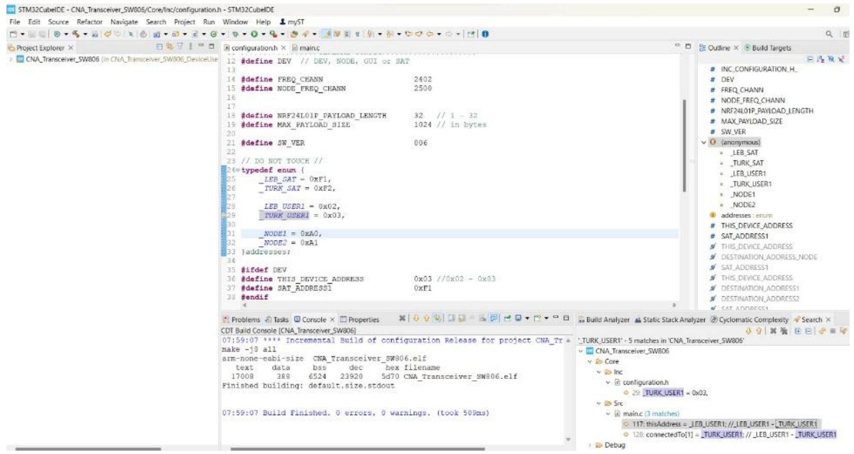
	TTC_GS_GUI_V_200224 ACS_Test_Stand_GUI integrated in TT&C_GUI full connected version		
TT&C GUI Bug fixing 3.1.25			GUI_TTC_GS Commenting out parts of the code : // data_request_timer.Start(); (Comment reading data) 

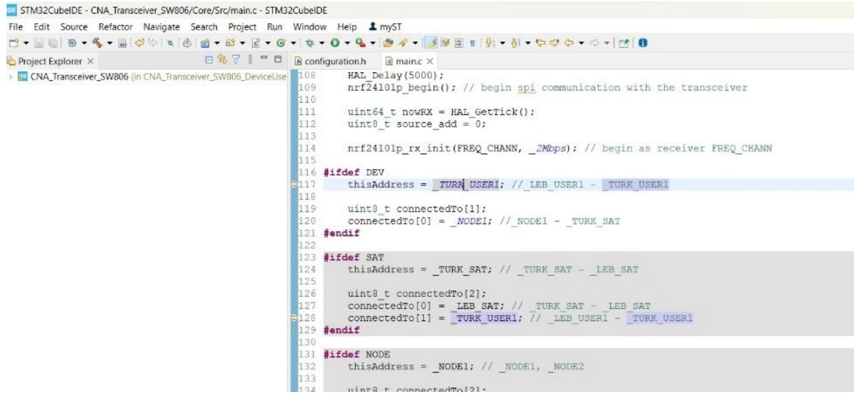
			 
TT&C STM32 (C)	TT&C STM32 (C) Ver. 071124		Bug fixed: Maximum retries are disabled. Communication is much stable. In reset registers function (line 82), instead of write_register(NRF24L01P_REG_CONFIG, 0x08), it became write_register(NRF24L01P_REG_CONFIG, 0x18).

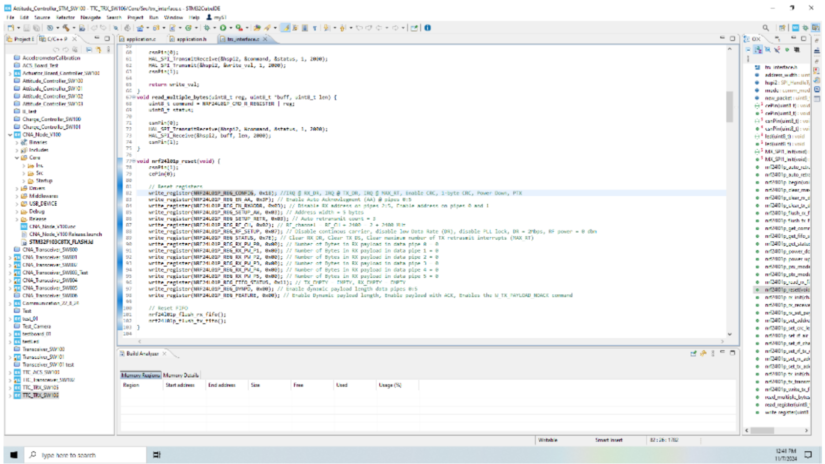
			
TT&C STM32 (C)	TT&C STM32 (C) Ver. 280823 TT&C STM32(C) -170224- Valve open correctly 1 to 8 TT&C STM32(C) -190224- Valve open correctly 1 to 9 https://aecenar.com/index.php/downloads/send/13-temo-space-communication/1707-tt-c-stm32-sw-110724 (A Acha)		
CNA GUI (C#)	CNA GUI C# Ver. 010124 (Fully Functional)		

<table>
<tr>
<td>CNA GUI (C#) Corrected</td>
<td>CNA GUI C# Ver. 091024 (Fully Functional, as before with predefined texts)</td>
<td></td>
<td>Bug: Port_not_recognized_inGermanWindowsVersion
fixed on 091024


Bug: Map and Coordinates were not correctly shown in Form.
Fix: in MapCoordinates.txt files: for German version "," instead of "."</td>
</tr>
<tr>
<td>CNA User GUI (C#)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CNA STM32 (C) baseline</td>
<td>CNA STM32 (C) Ver. 311223 (communication partner is defined in file configuration.h and(!) main.c)</td>
<td>Testreport see CNA System Development Report 2023 (pdf) at end</td>
<td>communication partner is defined in file configuration.h and(!) main.c: In the following pictures there is the configuration of Turk User to be programmed on STM:</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td></td>
<td>of the document</td>
<td>


</td>
</tr>
<tr>
<td>CNA STM32(C) baseline311223, Bugfixed 071124</td>
<td><u>CNA STM32(C) baseline311223, Bugfixed 071124</u></td>
<td></td>
<td>Bug fixed: Maximum retries are disabled. Communication is much stable. In reset registers function (line 82), instead of write_register(NRF24L01P_REG_CONFIG, 0x08), it became write_register(NRF24L01P_REG_CONFIG, 0x18).</td>
</tr>
</table>

			
CNA STM32 (C), Ver. Sep 24, COnfiguration with external HW jumpers (gateway or user or node is defined with a jumper on the board (see Users Guide, Getting	CNA STM32 (C) Ver. 100924 CNA STM32 (C) Ver. 180924 CNA_STM32 (C) Ver. 260924 (fixed users full functional)		

Started - CNA with 1 Gateway, 3 nodes, and n fixed users (aecenar.com))			
CNA Gateway STM32 (C)			
CNA Node STM32 (C)			
CNA User STM32 (C)			
OBC Rasberry (python)	OBC Raspberry (python) Ver. 061223 OBC Raspberry (python) Ver. until April 24 https://aecenar.com/index.php/downloads/send/13-temo-space-communication/1713-acs-valvecontrol-pythoncode-ver-0724 (AhmAli)		
ACS Board STM32 (C)	Attitude_Controller_SW101_241123 Attitude_Controller_SW103 210324 Attitude_Controller_SW103 230524		

ACS on OBC (python code)	ACS OBC python code Ver. 27.7.24 ACS OBC python code Sun Sensor - Valve Feedback Ver. 200824		
PV Charge Controller STM32 (C)	PMU (Solar Charger) SW (last updated: 23.08.23)		
ACS PLC Teststand - GUI	ACS PLC Teststand_180122 - GUI (C#)		

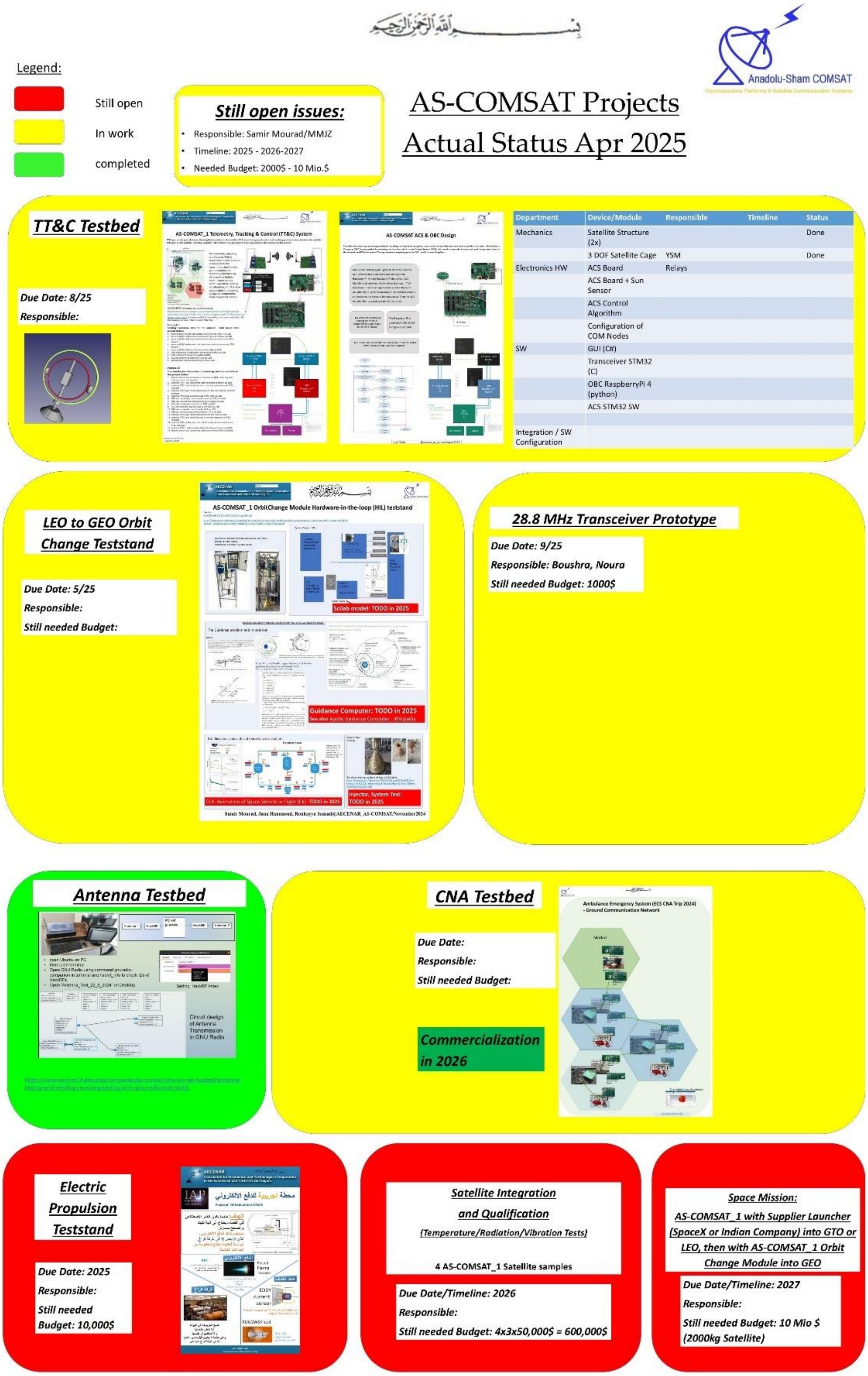
2.4 Hardware Boards

ACS Board	Gerber file for ACS Board.December.2023 Schematic of ACS Board.December.2023 Schematics of ACS Board May 2024 (without multiplexers)
Transceiver Board	Gerber file for Transiver Board.December.2023 BOM Transceiver Board, Ver. 25.10.23 Schematic of Transiver Board.December.2023
Power Charging Board	Gerber file for PC Board.December.2023 Schematic of Power Charging Board.December.2023

Amplifier Board	Gerber file for Emplifier Board.16.March.2024 Schematic of Emplifier Board.16.March.2024
Sun Sensor Board	Gerber file for Sun Sensor Board.20.March.2024 Schematic of Sun Sensor Board.20.March.2024
Patch Antenna Board	Fusion Design of Patch Antenna.21.March.2024

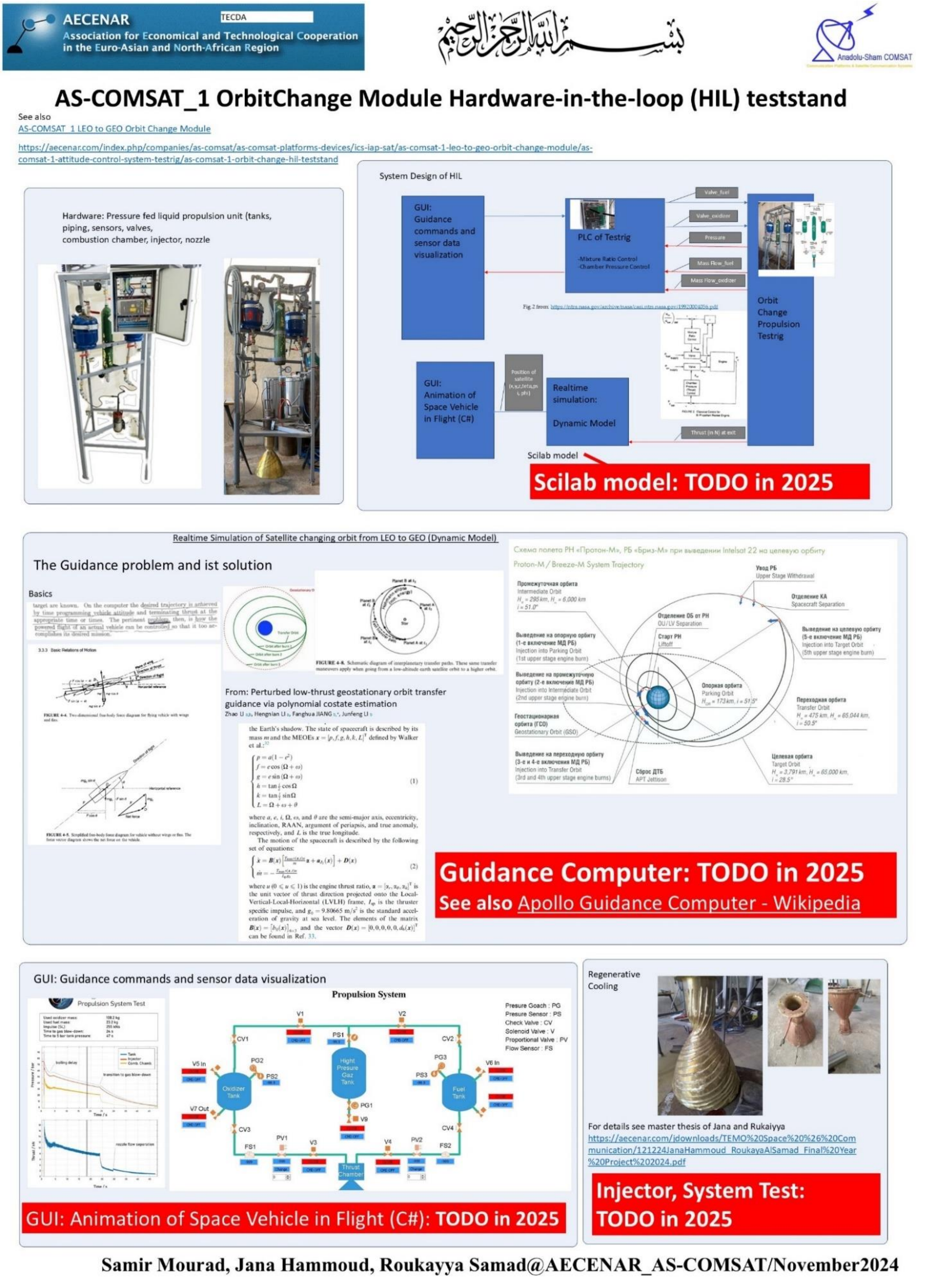
3 Summary

3.1 Actual Status of Projects in Apr 2025



3.2 Next TODOs

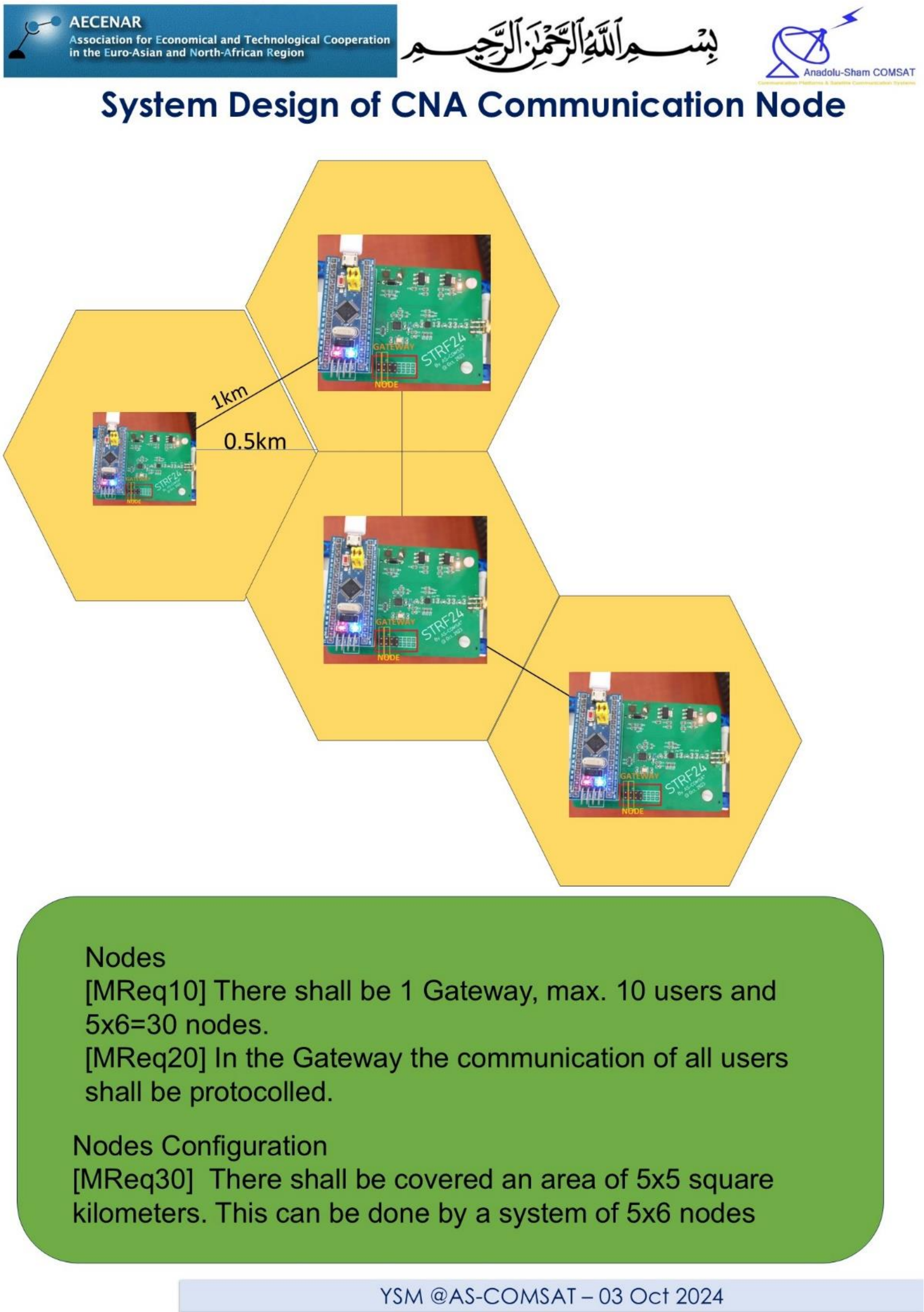
3.2.1 OrbitChange Testrig



3.2.2 CNA

3.2.2.1 Transceiver Development 28.8 MHz

3.2.2.2 CNA Area Testbed



3.2.3 TT&C Testrig

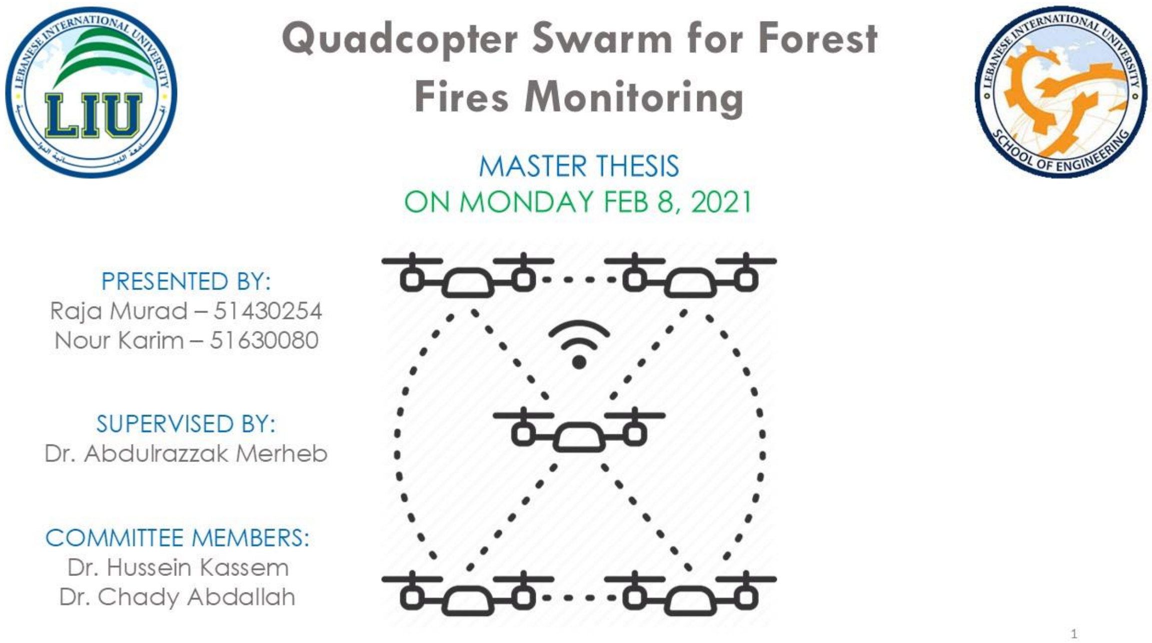
- Completing Testrig Guidance&Control
- Data Link

3.2.4 ECS

Completing Teststand Configuration

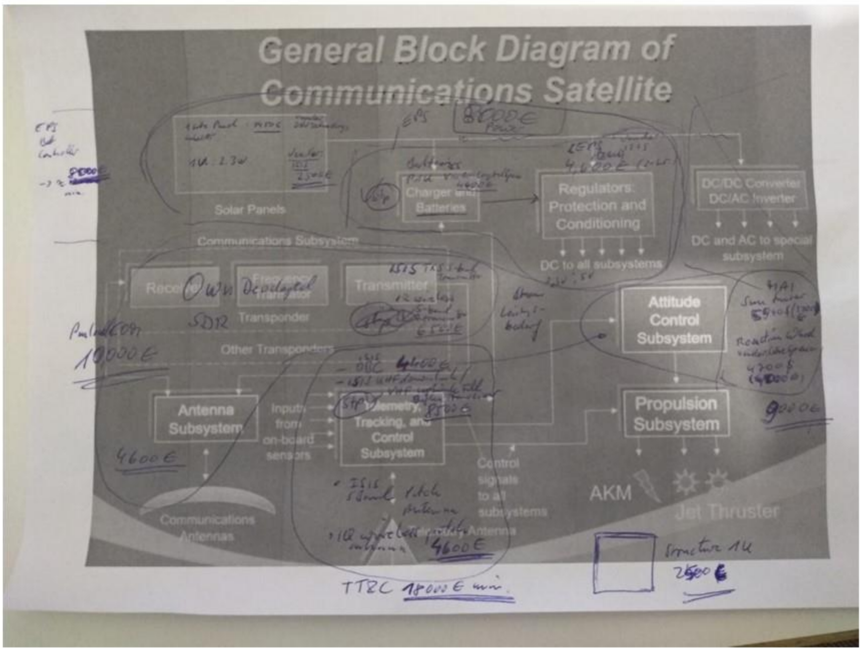
3.3 Posters/Presentations of partly realized projects

3.3.1 Early Fire Detection with Drones



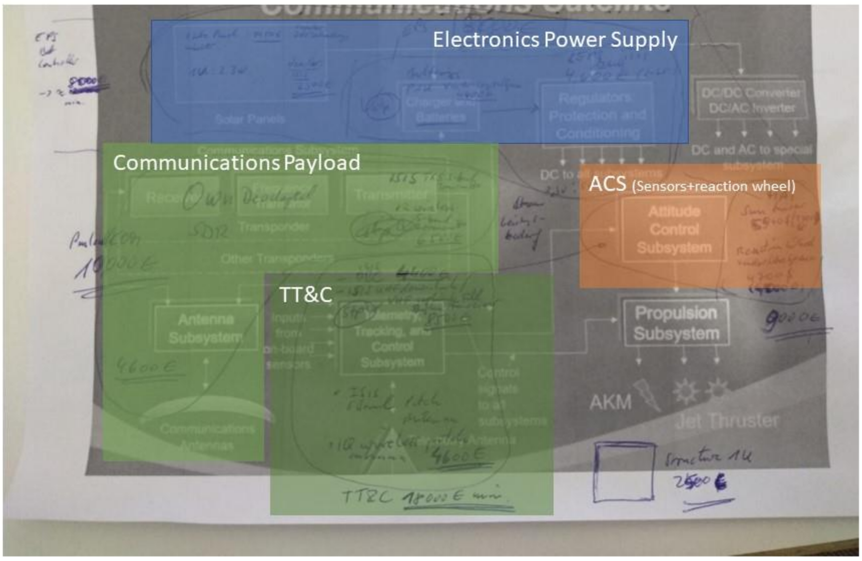
3.3.2 Nanosat with suppliers components

Minimal System for Demonstration



1/23/2022 AS-COMSAT, Jan-Feb 2021 152

Minimal System for Demonstration



1/23/2022 AS-COMSAT, Jan-Feb 2021 153

Minimal System for Demonstration

The parts for a minimal communication 1U cubesat (without camera, only communication payload) will cost inscha Allah about 48,000 EUR (See above sheet). The launch also about 40,000 EUR.

That means we need about 90,000 EUR for a demonstration satellite system.

The communication payload costs about 10,000 EUR from cubesatshop.With the master thesis proposal from Salih hoca this will be reduced to the cost of an S-band patch antenna (about 4600 EUR)

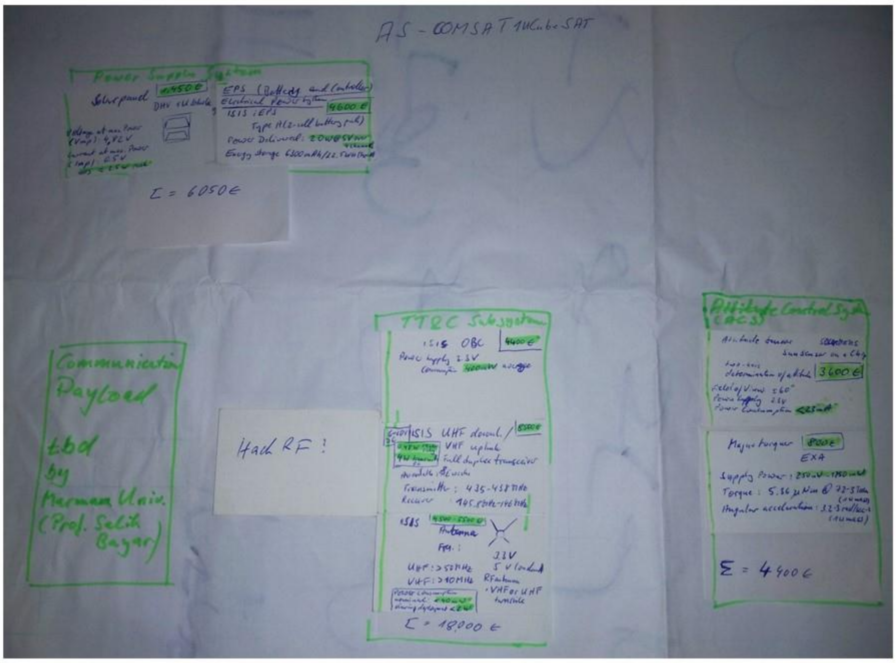
When we have this demonstration system working, then we will find inscha Allah investors for larger systems.

Next step:

Verifying power requirements for electronic parts. (Is the 1U Power Solar Panels + Batteries System enough?)

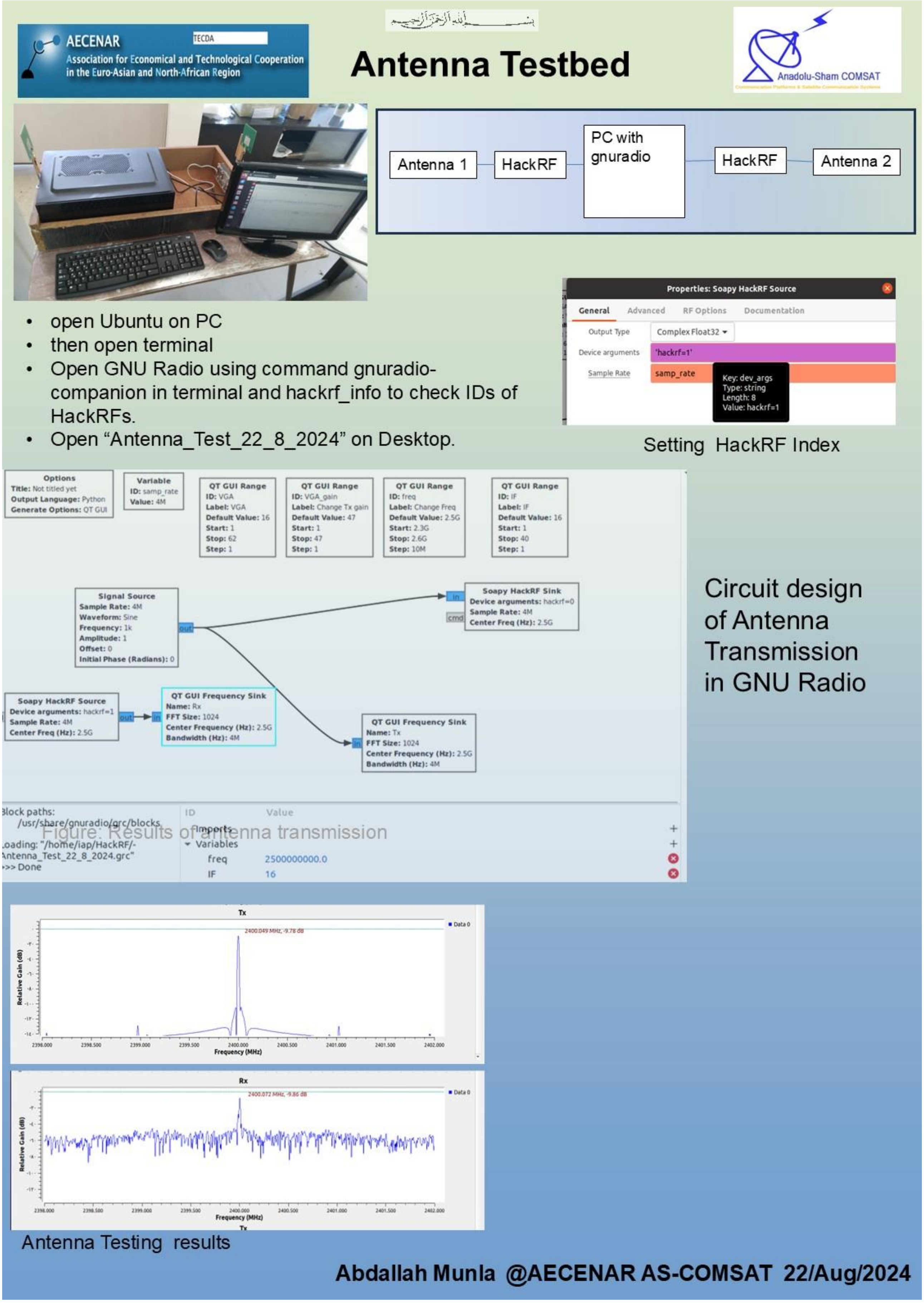
1/23/2022 AS-COMSAT, Jan-Feb 2021 154

Minimal System for Demonstration



1/23/2022 AS-COMSAT, Jan-Feb 2021 156

3.3.3 Antenna Testbed



3.3.4 TT&C Testrig

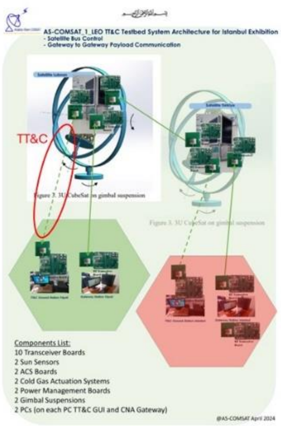
AECENAR — Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region — TECDA

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

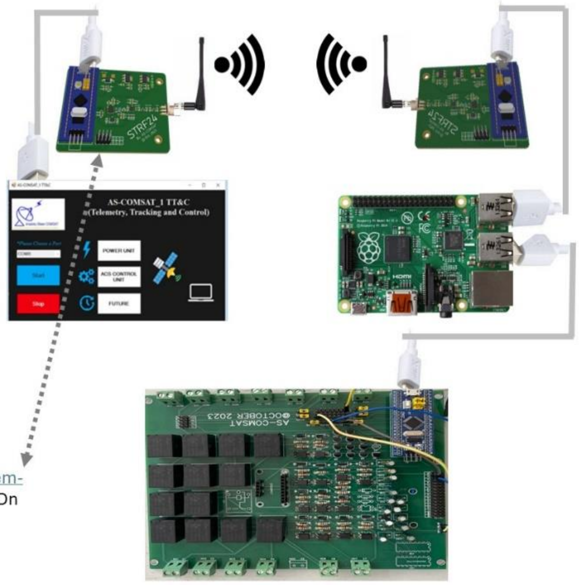
Anadolu-Sham COMSAT

AS-COMSAT_1 Telemetry, Tracking & Control (TT&C) System

TT&C part on the ground station: Sending Guidance data to the satellite ACS board from ground station and receiving position status data from the satellite
TT&C part on the satellite: Sending navigation information to the ground and receiving guidance information from the ground



The Telemetry, Tracking, and Control (TT&C) transceiver on the satellite is responsible for transmitting back to the ground station the telemetry and tracking data such as battery health, solar panels health, position updates, acceleration, etc. It is also responsible for receiving navigation commands from the ground station.



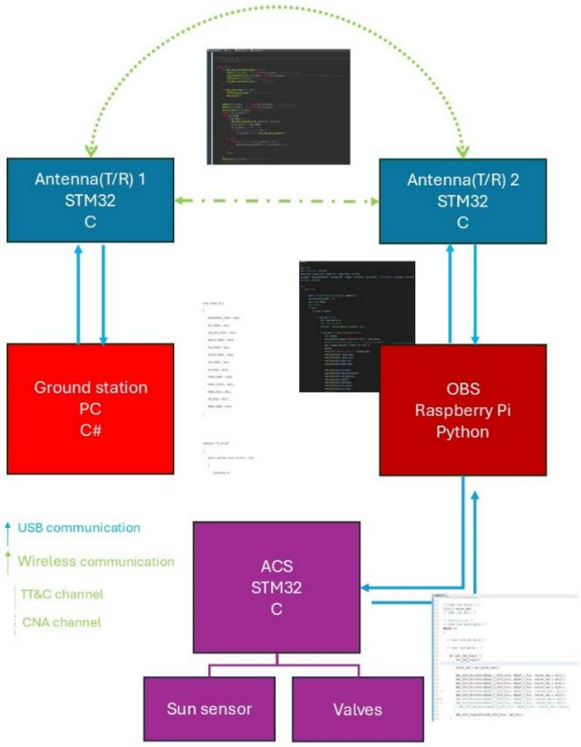
AS-COMSAT developed transceiver boards (https://aecenar.com/index.php/companies/as-comsat/as-comsat-platforms-devices/rf-2-4ghz-tranceiver-unit-prototype/rf-2-4ghz-system-design-microchip) are responsible for providing this communication. On the transceiver board there is a STM32 chip.

Protocol 1:
Sending Guidance data to the satellite ACS board from ground station

1. ground station sends data (CMD) to the antenna (T/R) 1 via USB
2. antenna (T/R) 1 will receive the data from ground station via USB
3. antenna (T/R) 1 will send this data to the second antenna via TT&C channel
4. antenna (T/R) 2 will receive this data from the first antenna via TT&C channel
5. antenna (T/R) 2 will send this data to the OBS via USB
6. OBS will receive the data from the antenna (T/R) 2 via USB
7. OBS will send the data to the ACS via USB
8. ACS will receive the data from OBS via USB
9. ACS will take actions depending on those data

Protocol 2:
Transmitting back telemetry and tracking data from satellite to the ground station

1. Ground station will send data to the antenna (T/R) 1 that requests telemetry and tracking data
2. antenna (T/R) 1 will receive the data from ground station via USB
3. antenna (T/R) 1 will send this data to the second antenna via TT&C channel
4. antenna (T/R) 2 will receive this data from the first antenna via TT&C channel
5. antenna (T/R) 2 will send this data to the OBS via USB
6. OBS will receive the data from the antenna (T/R) 2 via USB
7. OBS will request the data from the ACS system via USB
8. ACS will receive the request from OBS via USB
9. ACS will send the required data to the OBS via USB
10. OBS will receive the data from the ACS via USB
11. OBS will send the data to the antenna (T/R) 2 via USB
12. antenna (T/R) 2 will receive this data from the OBS via USB
13. antenna (T/R) 2 will send this data to the first antenna via TT&C channel
14. antenna (T/R)1 will receive this data from the second antenna via TT&C channel
15. antenna (T/R) 1 will send this data to the Ground station via USB
16. Ground station will receive the data from antenna (T/R) 1 via USB



Ahmad Ismail Achaji
10/July(07)/2024

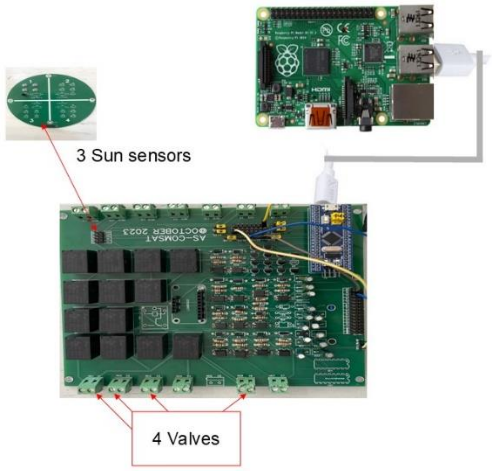
AECENAR
TECDA
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region

Anadolu-Sham COMSAT

AS-COMSAT ACS & OBC Design

The attitude control system is responsible for receiving the high-level navigation commands and stabilize the satellite at a specific orientation. The On-Board Computer (OBC) is responsible for receiving the data from the Attitude Control System (ACS) Unit then do some calculations and send the updated status of the valve to the ACS to command. The system uses a raspberry pi as the OBC, and it is coded in python.

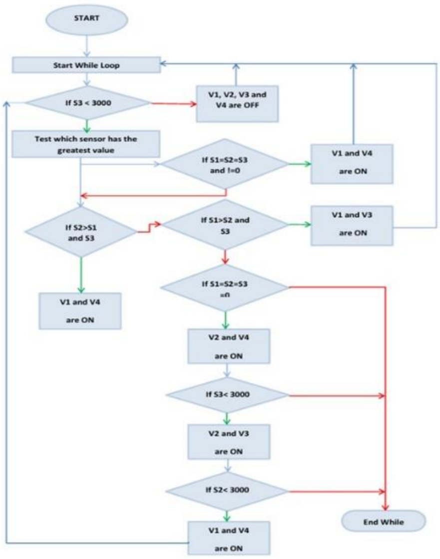
Sun Sensor, Analog signal, goes to the STM32 on the ACS Actuator board and then immediately to the Raspberry Pi. On the Raspberry Pi the python ACS Algorithms determines, which valves shall open. Then Raspberry Pi sends an appropriate signal to the ACS Actuator Board, which is obviously a list contains zeros or one indicates the status of the four valves. Then the ACS Actuator Boards sends a signal to the valves.

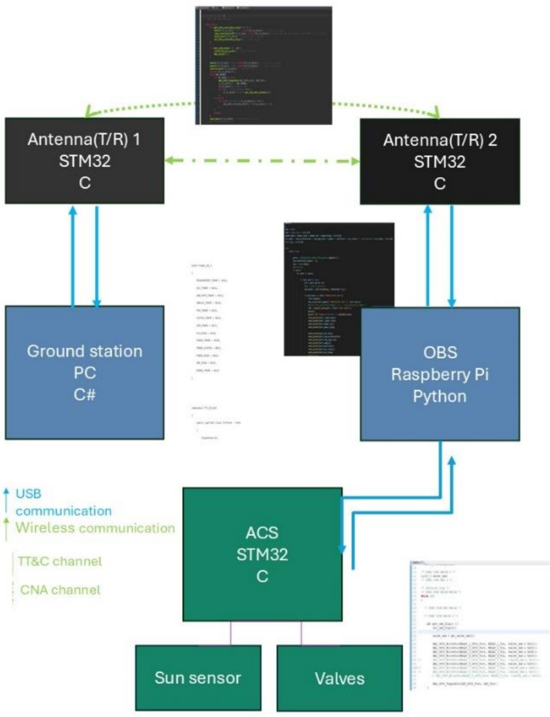


This action of opening or closing the valve is happened through relays on the ACS board.

The Raspberry Pi is connected to the stm32 through a USB cable.

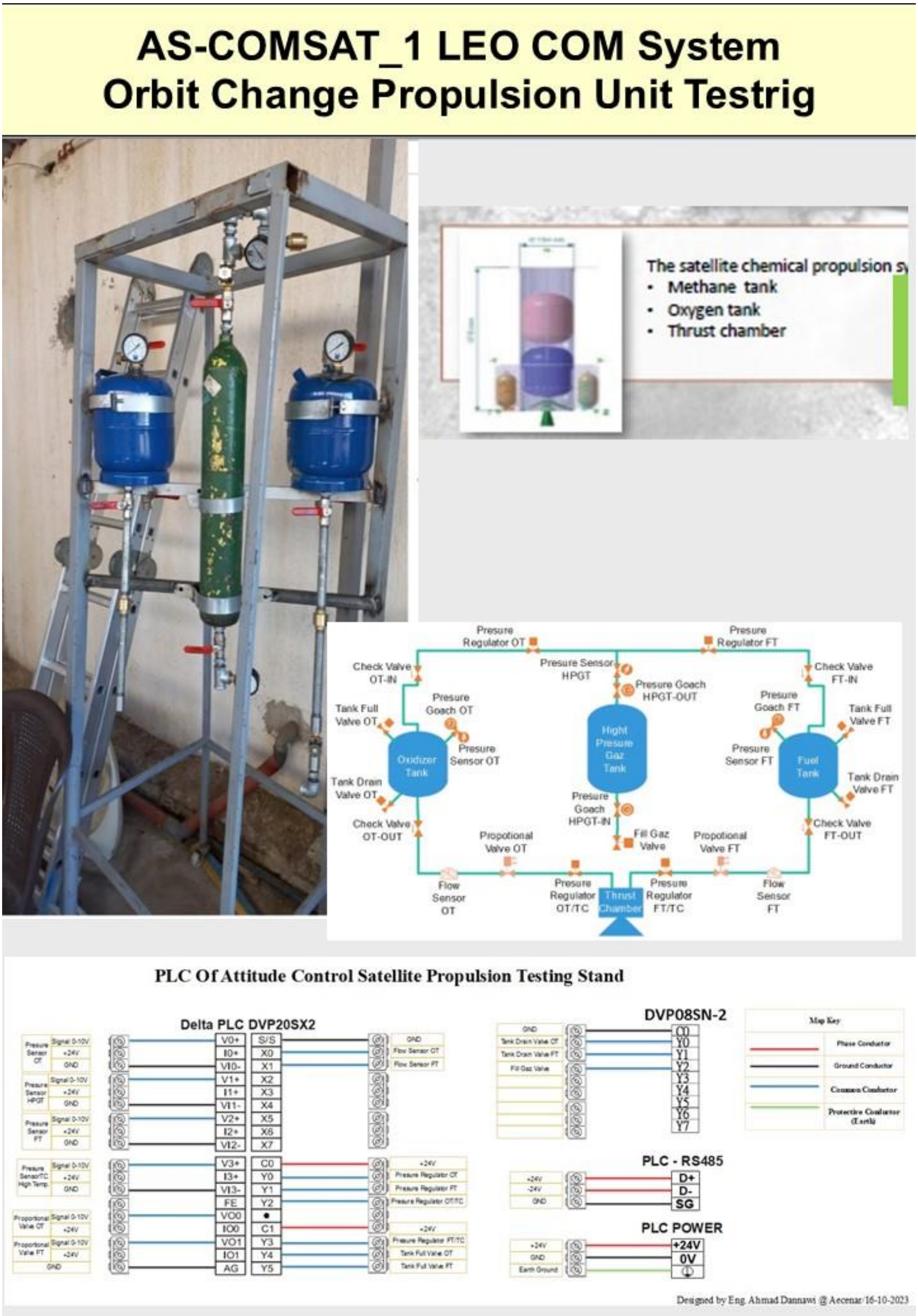
The STM32 should update the Raspberry Pi by the status of the valves when it sends a request.





Layla Taleb @aecenar_as_comsat/august 2024

3.3.5 Orbit Change Testrig



AECENAR
TECDA
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region

بسم الله الرحمن الرحيم

Anadolu-Sham COMSAT

AS-COMSAT_1 OrbitChange Module Hardware-in-the-loop (HIL) teststand

See also
AS-COMSAT_1 LEO to GEO Orbit Change Module

https://aecenar.com/index.php/companies/as-comsat/as-comsat-platforms-devices/ics-iap-sat/as-comsat-1-leo-to-geo-orbit-change-module/as-comsat-1-attitude-control-system-testrig/as-comsat-1-orbit-change-hil-teststand

Hardware: Pressure fed liquid propulsion unit (tanks, piping, sensors, valves, combustion chamber, injector, nozzle

System Design of HIL

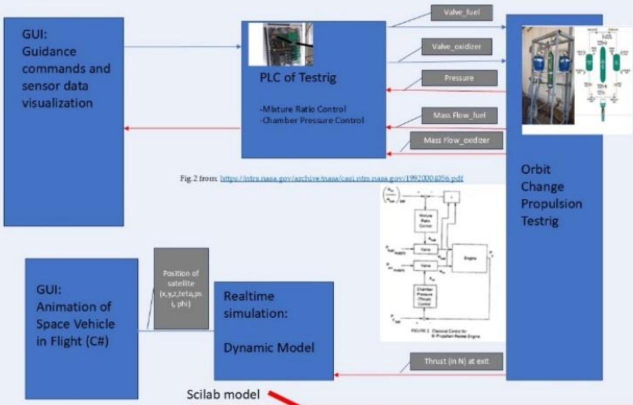


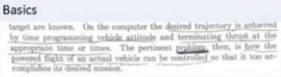
Fig.2 from: https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19920004056.pdf

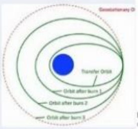
Scilab model: TODO in 2025

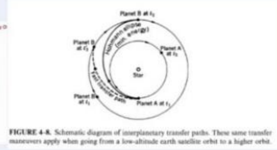
Realtime Simulation of Satellite changing orbit from LEO to GEO (Dynamic Model)

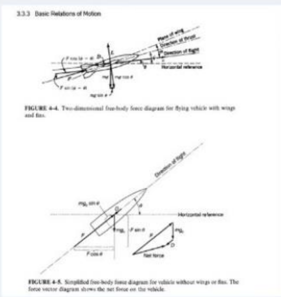
The Guidance problem and ist solution

Basics









From: Perturbed low-thrust geostationary orbit transfer guidance via polynomial costate estimation
Zhao LI a,b, Hengnian LI b, Fanghua JIANG a,*, Junfeng LI a

the Earth's shadow. The state of spacecraft is described by its mass m and the MEOEs $\boldsymbol{x} = [p, f, g, h, k, L]^T$ defined by Walker et al.:[32]

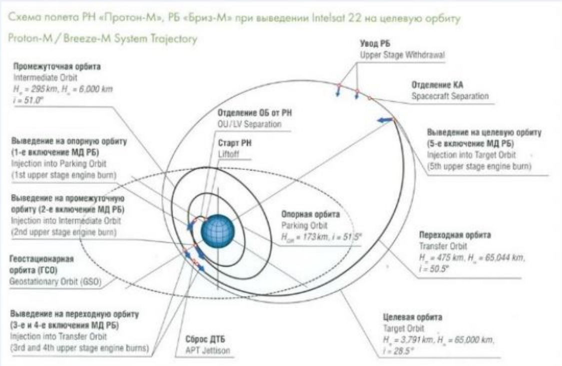
$$\begin{cases} p = a(1-e^2) \\ f = e\cos(\Omega+\omega) \\ g = e\sin(\Omega+\omega) \\ h = \tan\frac{i}{2}\cos\Omega \\ k = \tan\frac{i}{2}\sin\Omega \\ L = \Omega+\omega+\theta \end{cases} \quad (1)$$

where a, e, i, Ω, ω, and θ are the semi-major axis, eccentricity, inclination, RAAN, argument of periapsis, and true anomaly, respectively, and L is the true longitude.

The motion of the spacecraft is described by the following set of equations:

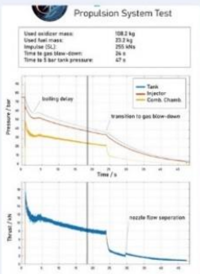
$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{B}(\boldsymbol{x})\left[\frac{T_{max}u(\boldsymbol{x},t)}{m}\boldsymbol{\alpha} + \boldsymbol{a}_{J_2}(\boldsymbol{x})\right] + \boldsymbol{D}(\boldsymbol{x}) \\ \dot{m} = -\frac{T_{max}u(\boldsymbol{x},t)}{I_{sp}g_0} \end{cases} \quad (2)$$

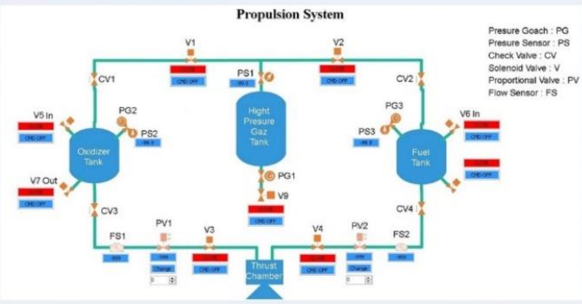
where u $(0 \leqslant u \leqslant 1)$ is the engine thrust ratio, $\boldsymbol{\alpha} = [\alpha_r, \alpha_\theta, \alpha_h]^T$ is the unit vector of thrust direction projected onto the Local-Vertical-Local-Horizontal (LVLH) frame, I_{sp} is the thruster specific impulse, and $g_0 = 9.80665\ m/s^2$ is the standard acceleration of gravity at sea level. The elements of the matrix $\boldsymbol{B}(\boldsymbol{x}) = [b_{ij}(\boldsymbol{x})]_{6\times3}$ and the vector $\boldsymbol{D}(\boldsymbol{x}) = [0,0,0,0,0,d_6(\boldsymbol{x})]^T$ can be found in Ref. 33.



Guidance Computer: TODO in 2025
See also Apollo Guidance Computer - Wikipedia

GUI: Guidance commands and sensor data visualization





GUI: Animation of Space Vehicle in Flight (C#): TODO in 2025

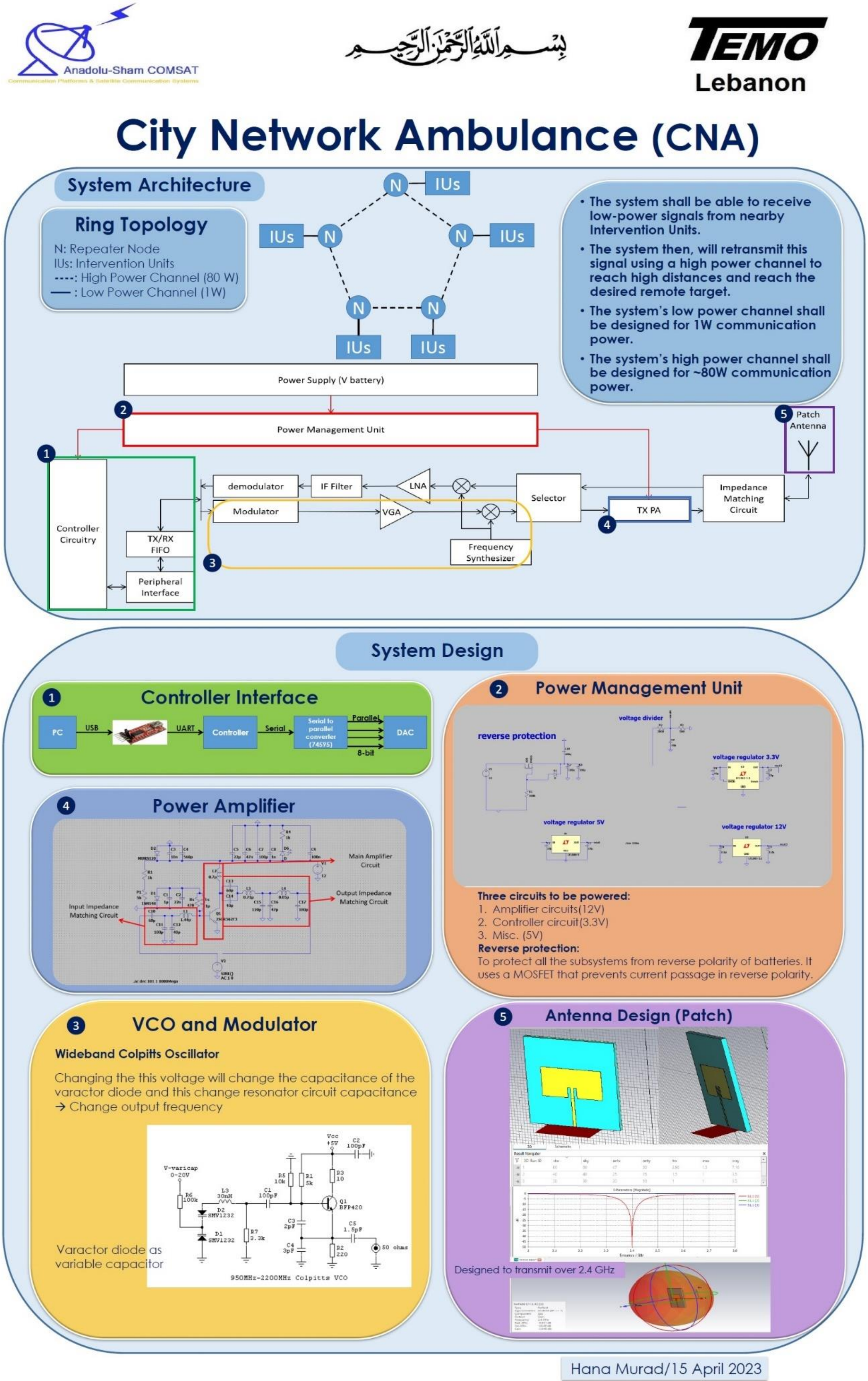
Regenerative Cooling

For details see master thesis of Jana and Rukaiyya
https://aecenar.com/idownloads/TEMO%20Space%20%26%20Communication/121224JanaHammoud_RoukayaAlSamad_Final%20Year%20Project%202024.pdf

Injector, System Test: TODO in 2025

Samir Mourad, Jana Hammoud, Roukayya Samad@AECENAR_AS-COMSAT/November2024

3.3.6 CNA



Anadolu-Sham COMSAT
بِسۡمِ ٱللَّهِ ٱلرَّحۡمَٰنِ ٱلرَّحِيمِ
TEMO
Lebanon
City Network Ambulance (CNA)
System Architecture
Ring Topology
N: Repeater Node
IUs: Intervention Units
----: High Power Channel (80 W)
—— : Low Power Channel (1W)
N
IUs
• The system shall be able to receive low-power signals from nearby Intervention Units.
• The system then, will retransmit this signal using a high power channel to reach high distances and reach the desired remote target.
• The system's low power channel shall be designed for 1W communication power.
• The system's high power channel shall be designed for ~80W communication power.
Power Supply (V battery)
Power Management Unit
Controller Circuitry
TX/RX FIFO
Peripheral Interface
demodulator
IF Filter
LNA
Modulator
VGA
Frequency Synthesizer
Selector
TX PA
Impedance Matching Circuit
Patch Antenna
System Design
Controller Interface
PC
USB
UART
Controller
Serial
Serial to parallel converter (74595)
Parallel
8-bit
DAC
Power Management Unit
voltage divider
reverse protection
voltage regulator 3.3V
voltage regulator 5V
voltage regulator 12V
Three circuits to be powered:
1. Amplifier circuits(12V)
2. Controller circuit(3.3V)
3. Misc. (5V)
Reverse protection:
To protect all the subsystems from reverse polarity of batteries. It uses a MOSFET that prevents current passage in reverse polarity.
Power Amplifier
Main Amplifier Circuit
Output Impedance Matching Circuit
Input Impedance Matching Circuit
VCO and Modulator
Wideband Colpitts Oscillator
Changing the this voltage will change the capacitance of the varactor diode and this change resonator circuit capacitance
→ Change output frequency
Varactor diode as variable capacitor
V-varicap 0-20V
950MHz-2200MHz Colpitts VCO
Antenna Design (Patch)
Designed to transmit over 2.4 GHz
Hana Murad/15 April 2023

AECENAR
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

Anadolu-Sham COMSAT

City Network Ambulance (CNA)
System Architecture & Functionality

CNA System Architecture

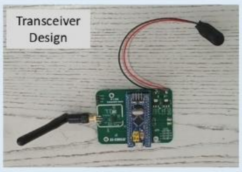
We have a "Star Topology" like architecture representing a prototype scheme for a network and its connectivity.

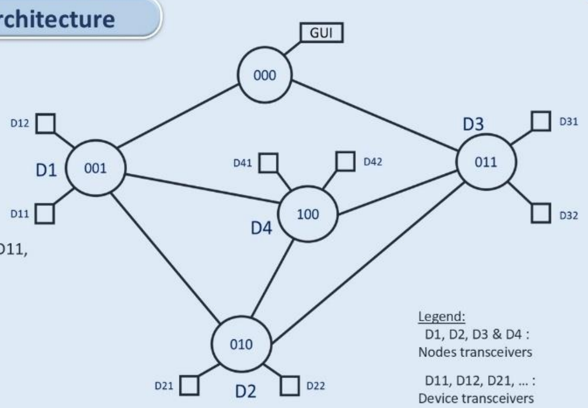
The way the network works is by using a central gateway to control the flow of the transmitting and receiving of data.

The central gateway will be a Graphical User Interface (GUI) connected on one of the transceivers.

The CNA GUI will serve as the gateway between all the Transceivers.

Each Node (D1, D2, D3, D4) will broadcast a request to all the connected transceivers (D11, D12, D21, ...) on the network and will store all the devices into an Address Table.





Frame Architecture

The Frame will be distributed into 4 parts:

* First Part contains the "Source Device Address" which consists of 4 Bytes (2 Bytes for the Node Source Address and 2 Bytes for Device Source Address).

Source Device	Destination Device	Gateway Address	Packet
4 Bytes	4 Bytes	2 Bytes	1014 Bytes

* Second Part contains the "Destination Device Address" which consists of 4 Bytes (2 Bytes for the Node Destination Address and 2 Bytes for Device Destination Address).

* Third Part contains the "Gateway Address" which will be the address of the GUI (0000) and it consist of 2 Bytes.

* Last Part contains the "Packet" that will be sent across the network and it consists of 1014 Bytes.

Functionality

There will be 3 possibilities on how the data will be sent across the network and how the architecture will handle it:

- <u>Case 1: GUI To Node:</u>

- The Frame will contain the Device Source Address which is 0000 001 that is the standard address related to the central GUI, and the Destination Address that the user specifies according to the address that he wants to communicate with.

- The Frame will also contain the Gateway Address which will always be the address of the GUI so each packet can reach the GUI and the GUI will transmit the data to the network to reach the specified address.

- The last piece of the message will contain the packet that will be sent as a message to the other user.

Case 1			
0000 001 Node Source Address / Device Source Address	0010 011 Node Destination Address / Device Destination Address	0000	Packet

- <u>Case 2: Node To GUI:</u>

- The Frame will contain the Device Source Address which is the address of the device related to the network, and the Destination Address that is the central gateway in this case which is 0000 001.

- The Frame will also contain the Gateway Address which will always be the address of the GUI so each packet can reach the GUI and the GUI will transmit the data to the network to reach the specified address.

- The last piece of the message will contain the packet that will be sent as a message to central GUI.

Case 2			
0010 001 Node Source Address / Device Source Address	0000 001 Node Destination Address / Device Destination Address	0000	Packet

- <u>Case 3: Node To Node:</u>

- The Frame will contain the Device Source Address which is the address of the device related to the network, and the Destination Address that the user specifies according to the address that he wants to communicate with.

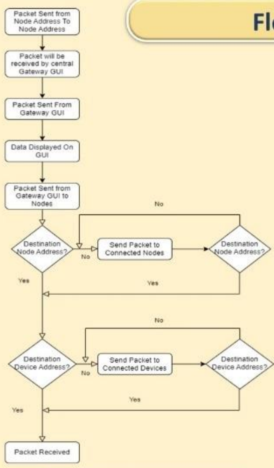
- The Frame will also contain the Gateway Address which will always be the address of the GUI so each packet can reach the GUI and the GUI will transmit the data to the network to reach the specified address.

- The last piece of the message will contain the packet that will be sent as a message to the other user.

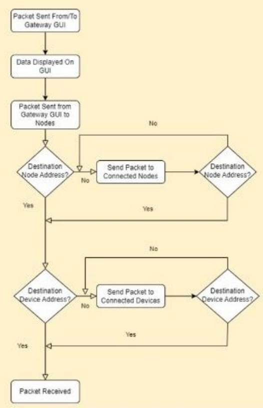
Case 3			
0010 001 Node Source Address / Device Source Address	0011 100 Node Destination Address / Device Destination Address	0000	Packet

In all cases, the Frame will pass through the central gateway (GUI) so the information will be displayed and the GUI will be responsible for transmitting the data again to the network

Flow Chart

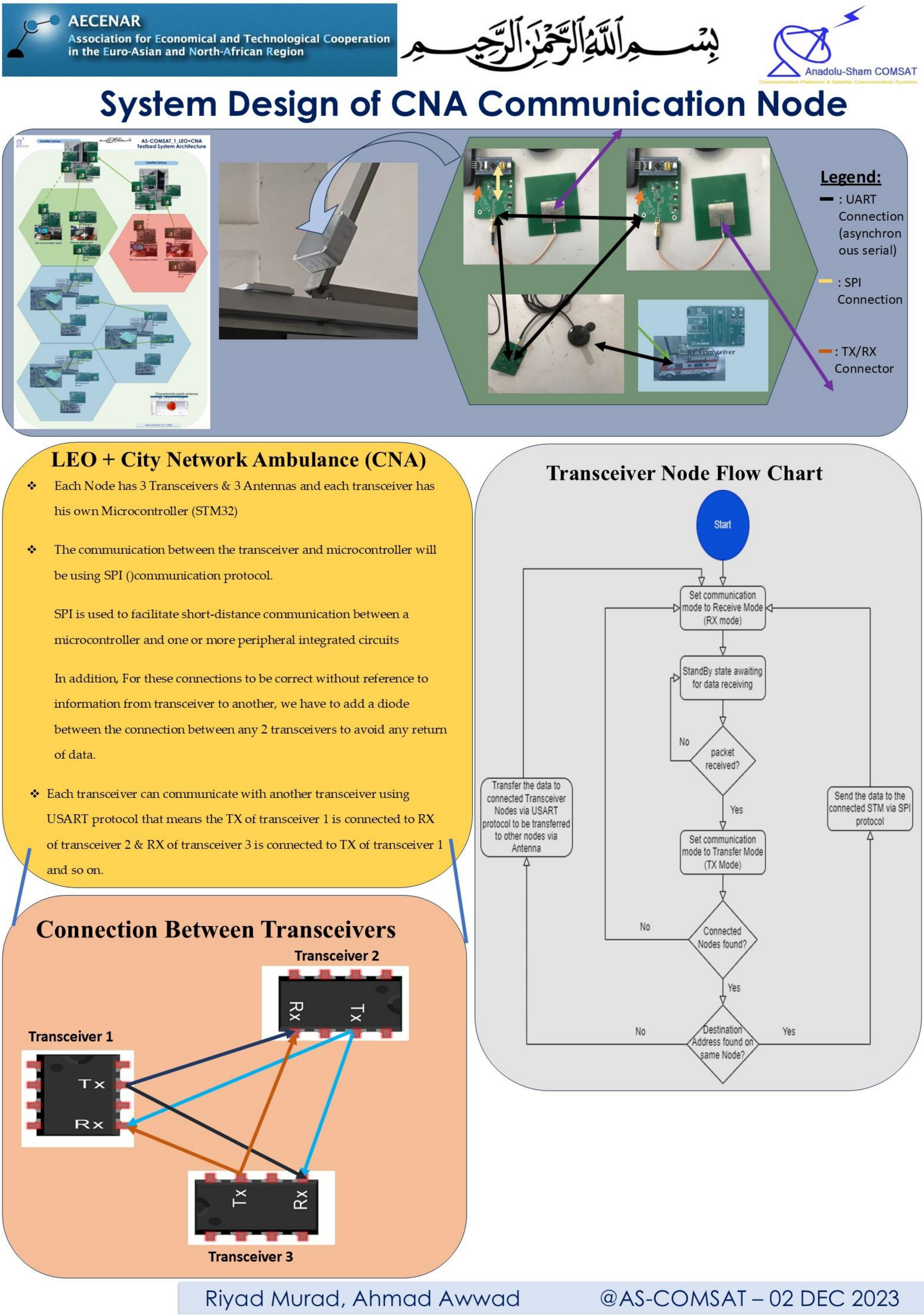


Node To Node Flow Chart

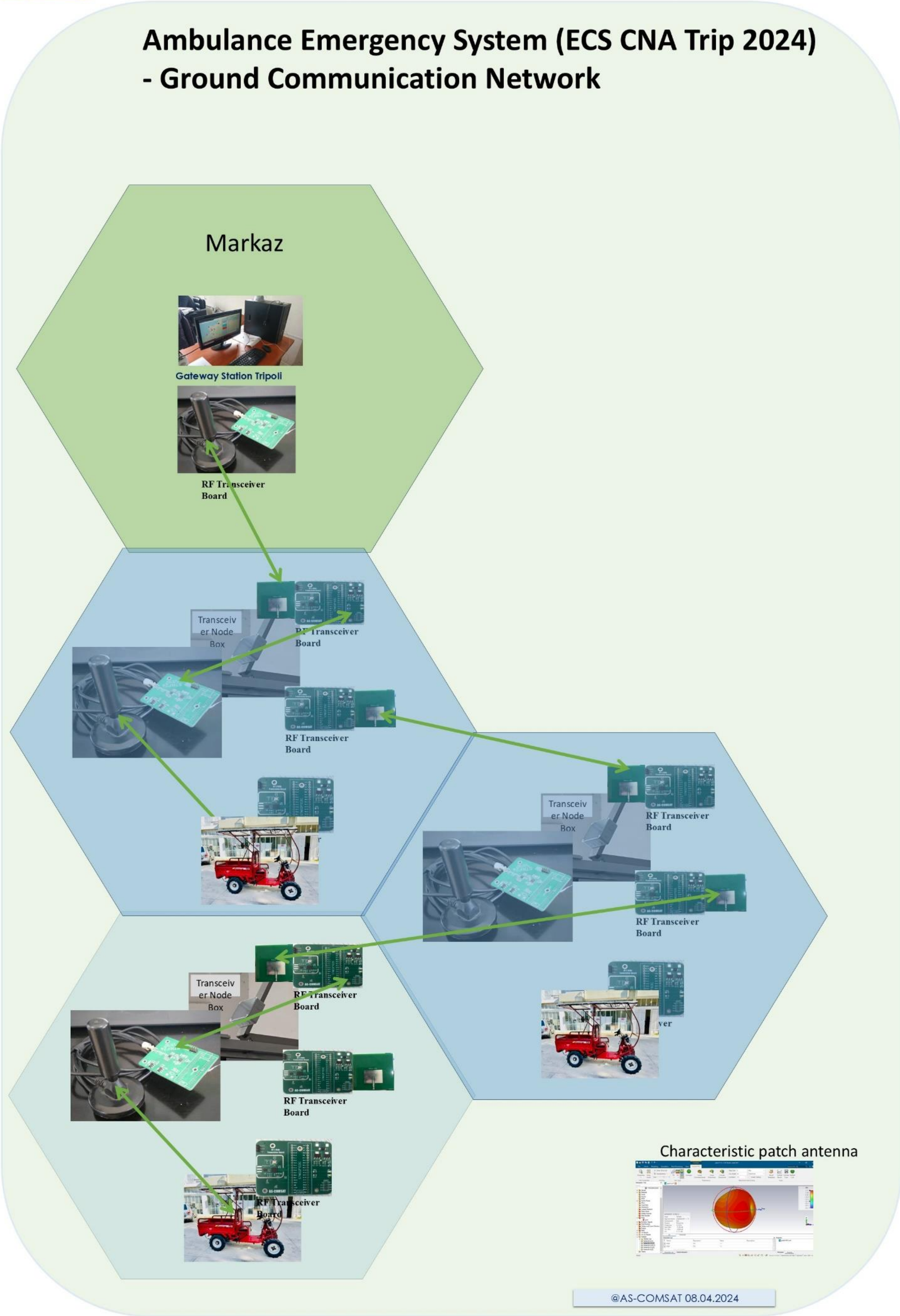


GUI to Node or Node to Node Flow Chart

Riyad Murad @AS-COMSAT – 06 OCT 2023


AECENAR
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region
بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ
Anadolu-Sham COMSAT
System Design of CNA Communication Node
AS-COMSAT_1_LEO+CNA Testbed System Architecture
Legend:
: UART Connection (asynchronous serial)
: SPI Connection
: TX/RX Connector
LEO + City Network Ambulance (CNA)
Each Node has 3 Transceivers & 3 Antennas and each transceiver has his own Microcontroller (STM32)
The communication between the transceiver and microcontroller will be using SPI ()communication protocol.
SPI is used to facilitate short-distance communication between a microcontroller and one or more peripheral integrated circuits
In addition, For these connections to be correct without reference to information from transceiver to another, we have to add a diode between the connection between any 2 transceivers to avoid any return of data.
Each transceiver can communicate with another transceiver using USART protocol that means the TX of transceiver 1 is connected to RX of transceiver 2 & RX of transceiver 3 is connected to TX of transceiver 1 and so on.
Connection Between Transceivers
Transceiver 1
Transceiver 2
Transceiver 3
Tx
Rx
Transceiver Node Flow Chart
Start
Set communication mode to Receive Mode (RX mode)
StandBy state awaiting for data receiving
packet received?
No
Yes
Set communication mode to Transfer Mode (TX Mode)
Connected Nodes found?
Destination Address found on same Node?
Transfer the data to connected Transceiver Nodes via USART protocol to be transferred to other nodes via Antenna
Send the data to the connected STM via SPI protocol
Riyad Murad, Ahmad Awwad
@AS-COMSAT – 02 DEC 2023

بسم الله الرحمن الرحيم

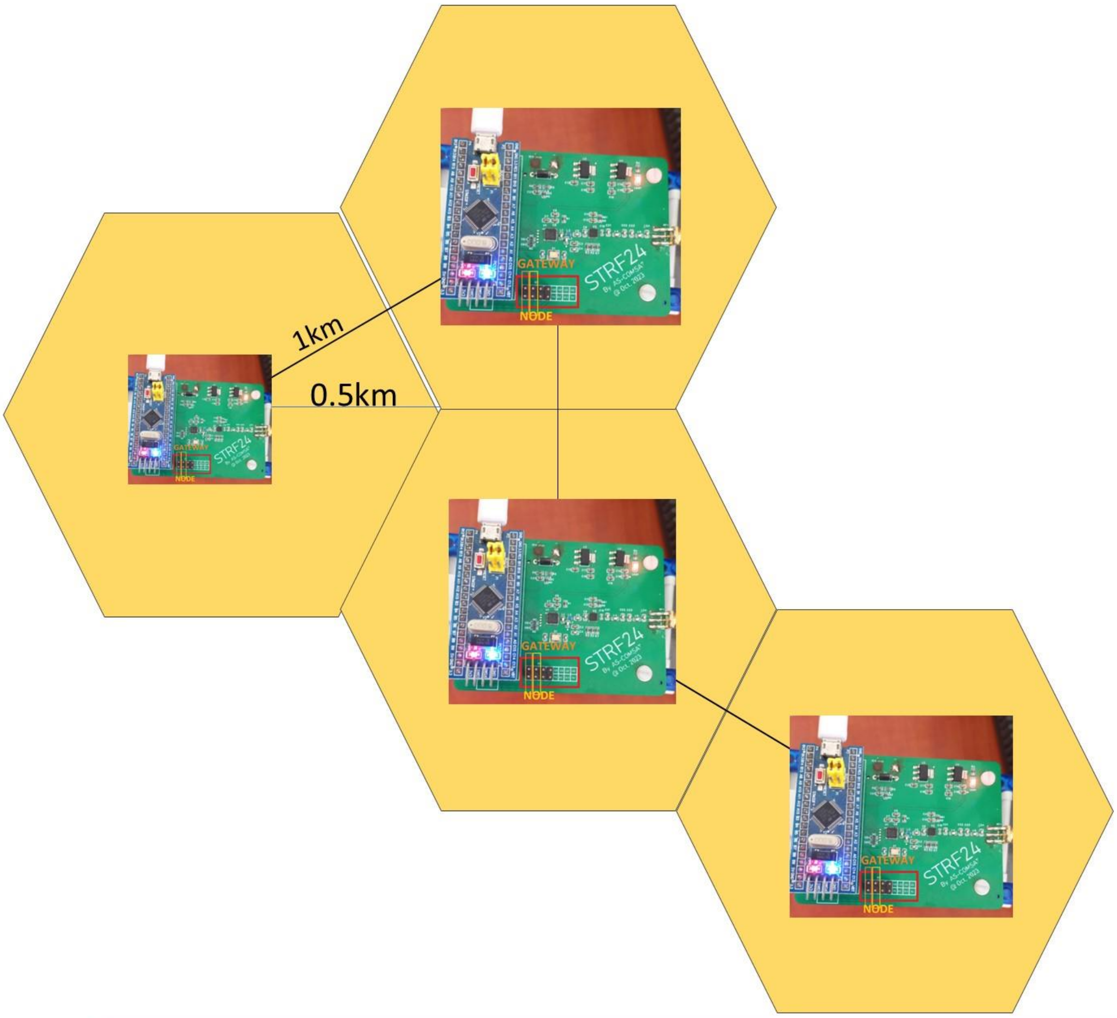


AECENAR
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

Anadolu-Sham COMSAT

System Design of CNA Communication Node

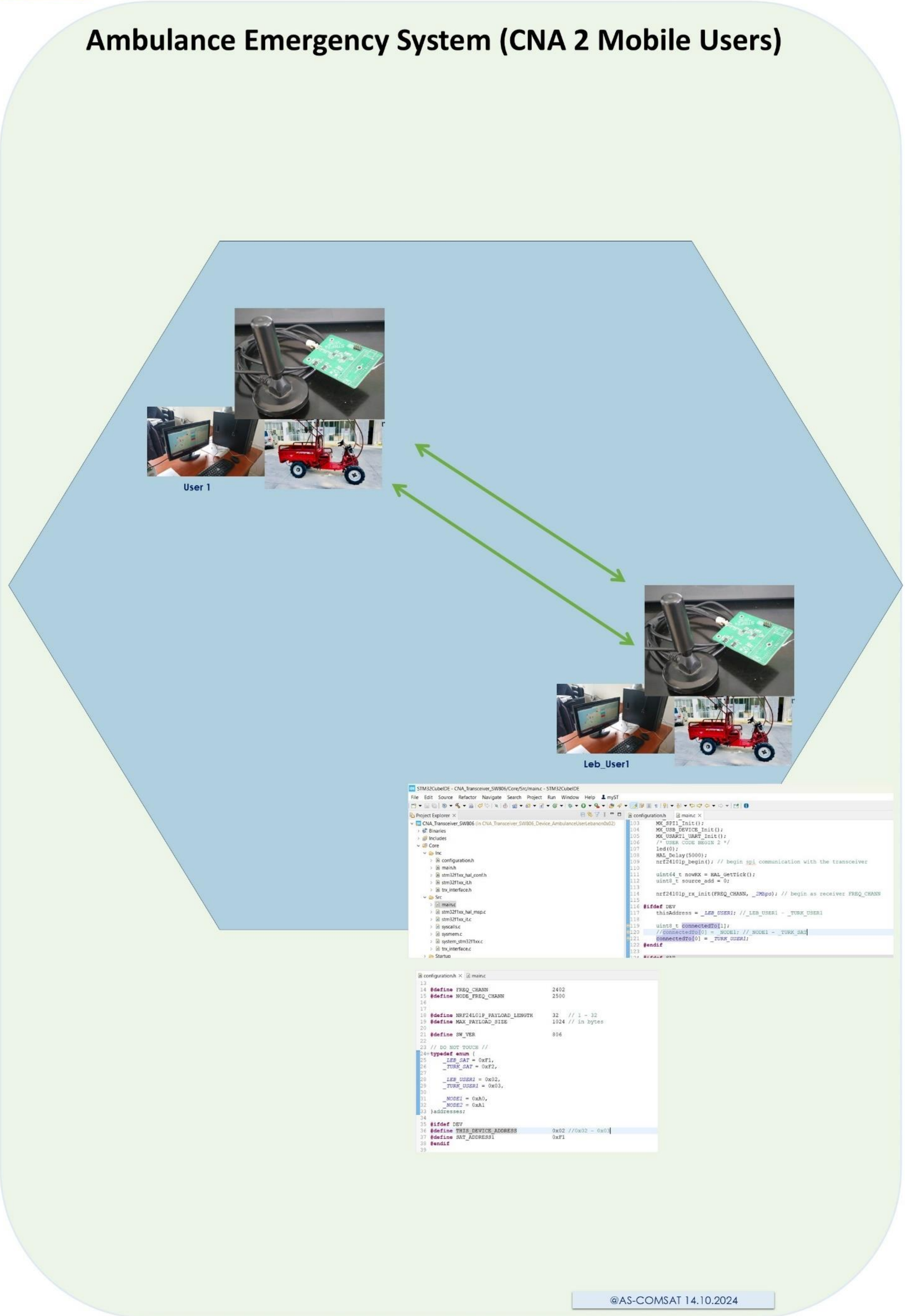


Nodes
[MReq10] There shall be 1 Gateway, max. 10 users and 5x6=30 nodes.
[MReq20] In the Gateway the communication of all users shall be protocolled.

Nodes Configuration
[MReq30] There shall be covered an area of 5x5 square kilometers. This can be done by a system of 5x6 nodes

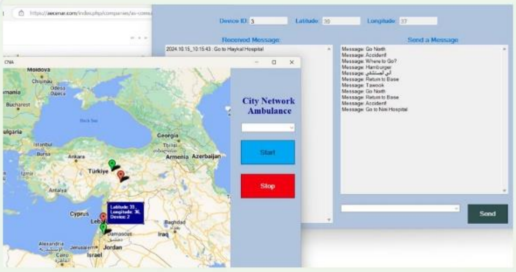
YSM @AS-COMSAT – 03 Oct 2024

بِسْمِ اللهِ الرَّحْمٰنِ الرَّحِيْمِ

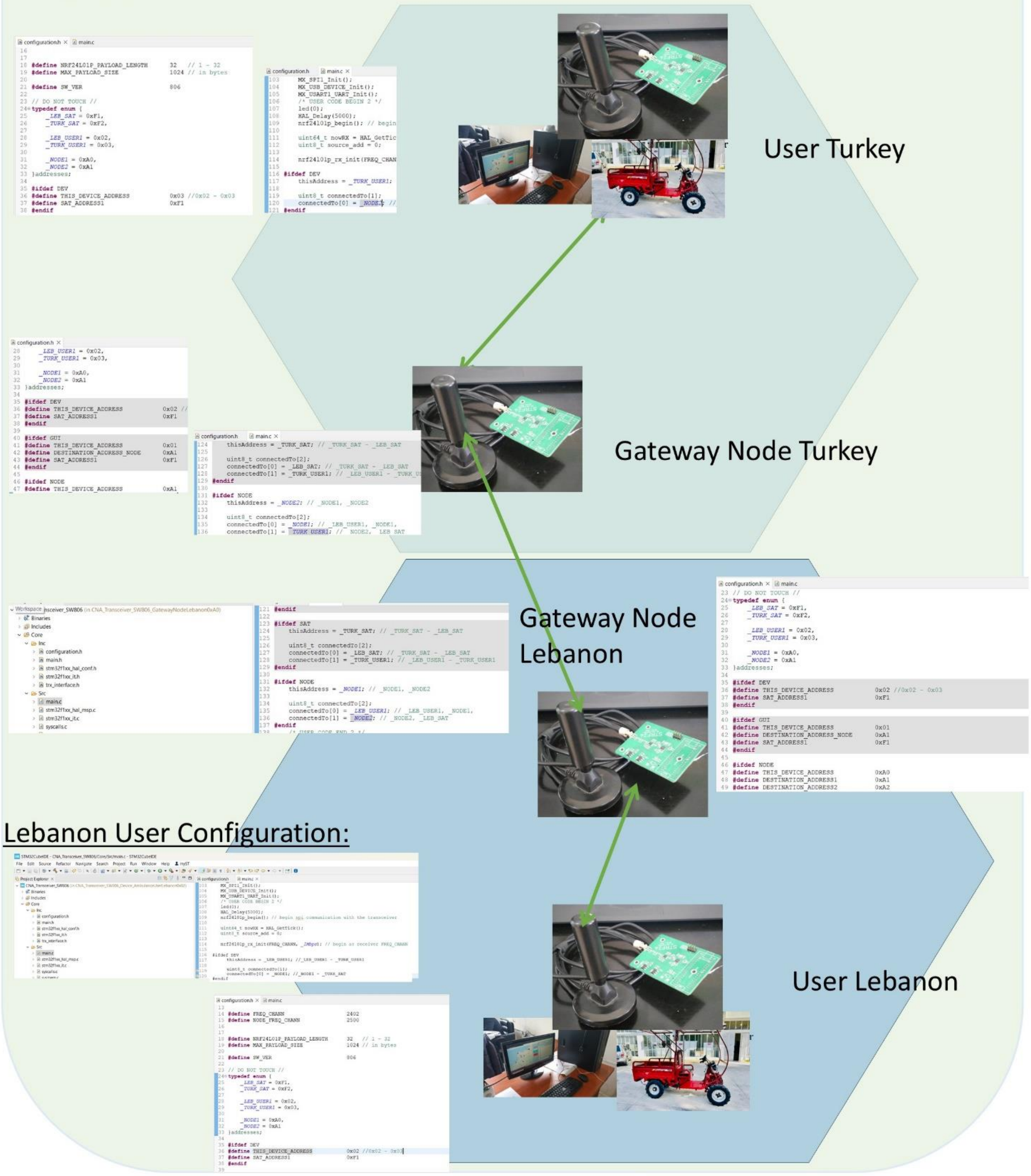


بسم الله الرحمن الرحيم

Ambulance Emergency System (CNA 2 Mobile Users, 2 Nodes)

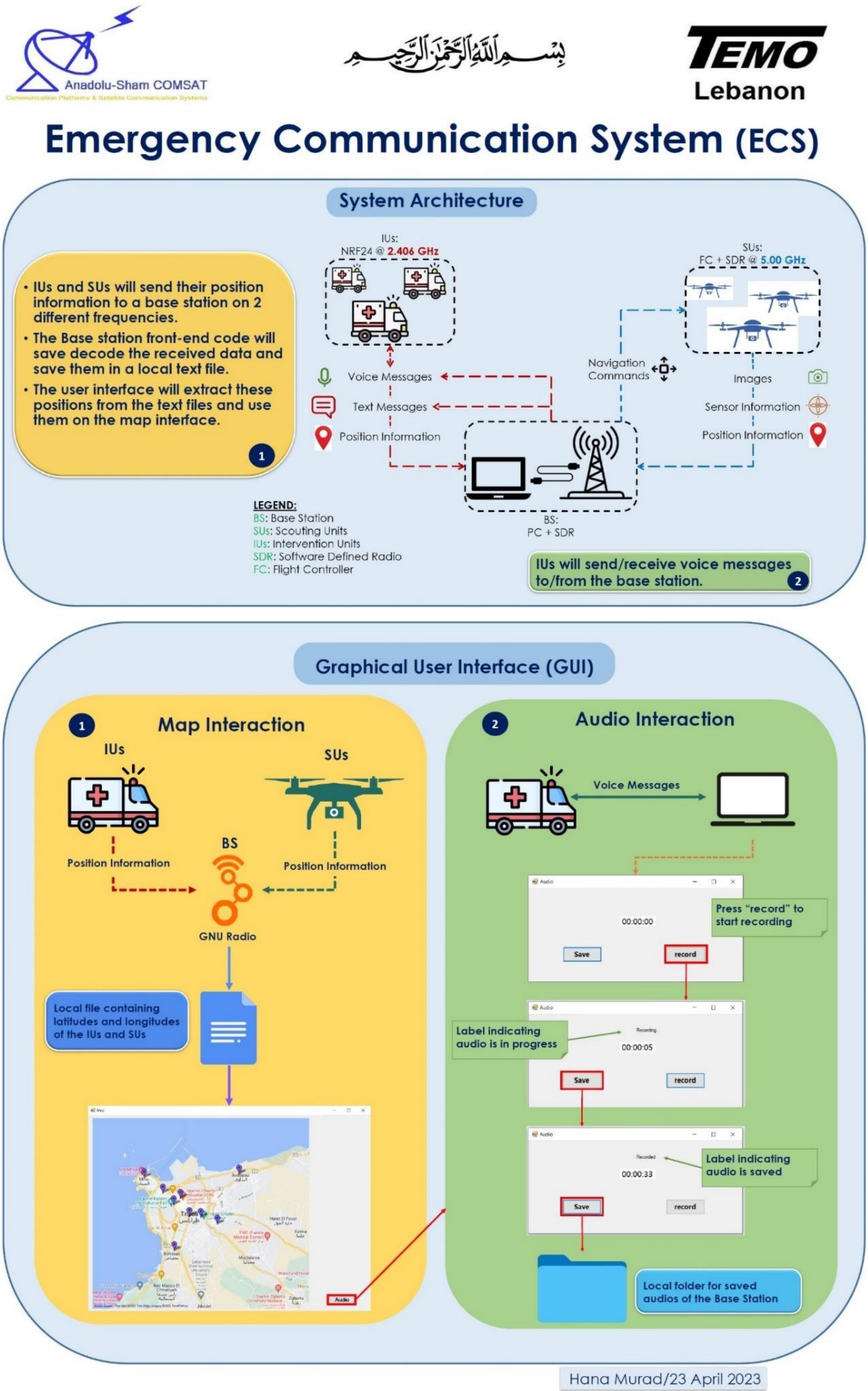


Transceiver	Connected to [0]	Connected to [1]
User Lebanon 0x02	Gateway node Lebanon	-
Gateway node Lebanon 0xA0	User Lebanon	Gateway node Turkey
Gateway node Turkey 0xA1	Gateway node Lebanon	User Turkey
User Turkey 0x03	Gateway Node Turkey	



@AS-COMSAT 15.10.2024

3.3.7 ECS


Anadolu-Sham COMSAT
بِسۡمِ ٱللَّهِ ٱلرَّحۡمَٰنِ ٱلرَّحِيمِ
TEMO
Lebanon
Emergency Communication System (ECS)
System Architecture
• IUs and SUs will send their position information to a base station on 2 different frequencies.
• The Base station front-end code will save decode the received data and save them in a local text file.
• The user interface will extract these positions from the text files and use them on the map interface.
1
IUs:
NRF24 @ 2.406 GHz
SUs:
FC + SDR @ 5.00 GHz
Voice Messages
Text Messages
Position Information
Navigation Commands
Images
Sensor Information
Position Information
BS:
PC + SDR
LEGEND:
BS: Base Station
SUs: Scouting Units
IUs: Intervention Units
SDR: Software Defined Radio
FC: Flight Controller
IUs will send/receive voice messages to/from the base station.
2
Graphical User Interface (GUI)
1
Map Interaction
IUs
SUs
BS
Position Information
Position Information
GNU Radio
Local file containing latitudes and longitudes of the IUs and SUs
2
Audio Interaction
Voice Messages
Press "record" to start recording
Save
record
Label indicating audio is in progress
Label indicating audio is saved
Local folder for saved audios of the Base Station
Hana Murad/23 April 2023

4 AS-COMSAT Administration Report 2023 with Supplement from 2024

بسم الله الرحمن الرحيم

AS COMSAT – Administration Report 2023

Planning & Controlling

Author:

Dr. Samir Mourad

Last update: Friday, June 09, 2023

Anadolu-Sham COMSAT

AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ

TEPEÜSTÜ MAH. KAFİYE SOK. NO:47/3 ÜMRANİYE/İSTANBUL

ALEMDAĞ V.D. V.NO:0861584690

İTO: 457897-5 MERSİS NO: 0086-1584-6900-0001

AS COMSAT COMMUNICATION PLATFORMS AND SATELLITE COMMUNICATION SYSTEMS, INDUSTRY AND TRADE LIMITED COMPANY

Name of document:

C:\AS-COMSAT\Administration\Planning&Control2023\020523AS-COMSAT_AdministrationReport2023.docx

AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ

TEPEÜSTÜ MAH. KAFİYE SOK. NO:47/3 ÜMRANİYE/İSTANBUL

ALEMDAĞ V.D. V.NO:0861584690

İTO: 457897-5 MERSİS NO: 0086-1584-6900-0001

AS COMSAT COMMUNICATION PLATFORMS AND SATELLITE COMMUNICATION SYSTEMS, INDUSTRY AND TRADE LIMITED COMPANY

Name of document:

C:\AS-COMSAT\Administration\Planning&Control2023\020523AS-COMSAT_AdministrationReport2023.docx

4.1 Content

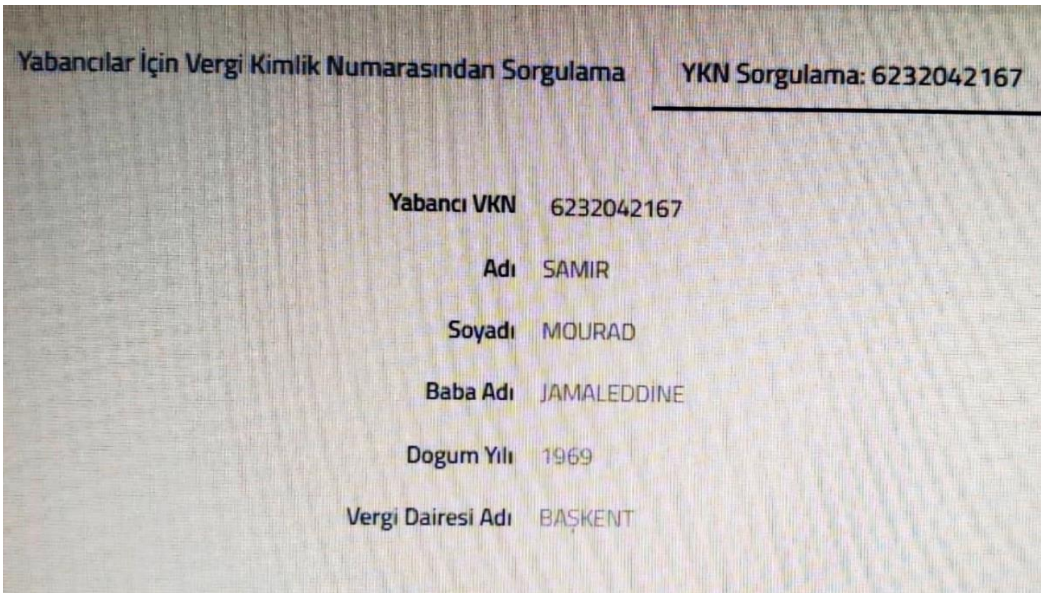
Content

4.2 Investors

4.2.1 Data of Investors

Name	Shares	Date, Remarks
Samir Mourad	80%	1.5.23
Salih Bayar	10%	1.5.23
ÖMER KORÇAK	10%	1.5.23

4.2.1.1 Turkish VKN number (T.C. No.) of Manager Samir Mourad



Samir Mourad, Yabanci VKN 6232042167

4.3 Initial Investment from company foundation until IMEX exhibition (1.5.-15.9.23)

AS COMSAT Initial Investment				last updated: 13.5.2023			
	per months						
	May	June	Jul	Aug	Sep (IMEX)		
3 Ing Lib	$ 1.500	$ 1.500	$ 1.500	$ 1.500	$ 1.500		
2 Ing TR (Koskep)	$ -						
Atelier	$ 250,00	$ 250	$ 250	$ 250	$ 250		
Foundation Company							
						Investment	Shares
Ömer hoca	$ 432,00	$ 432	$ 432	$ 432	$ 216	$ 1.944	0,101261
Salih hoca	$ 432,00	$ 432	$ 432	$ 432	$ 216	$ 1.944	0,101261
Samir hoca	$ 1.080,00	$ 1.080	$ 1.080	$ 1.080	$ 540	$ 15.310	0,797479
Material ECS	$ 300,00	$ 300					
Material CNA	$ 300,00	$ 300					
Material AS-COMSAT_1 Demo System							
Exhibition Fee, …					$ 500		
						Total	
	$ 4.294,00	$ 4.294	$ 3.694	$ 3.694	$ 3.222	$ 19.198	
IMEX: 14.-17.9.23							

4.4 Official company papers

4.4.1 Name of Company

AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ

TEPEÜSTÜ MAH. KAFİYE SOK. NO:47/3 ÜMRANİYE/İSTANBUL

ALEMDAĞ V.D. V.NO:0861584690

İTO: 457897-5 MERSİS NO: 0086-1584-6900-0001

4.4.2 Renting contract for company space (Kafiye sokak)

Buero in Kafiye Sokak No. 47, Tepeüstü, Umraniyye, Istanbul (Asian side)

250 EUR all in rent for the whole flat (rented from Samir).

One part of it is the company space.

KİRA SÖZLEŞMESİ ÖZEL HÜKÜMLER

1- Kiracı, kiralananı kiraya verenin yazılı iznini almadan başkalarına kullandıramaz, kiraya veremez ve
devredemez.

2- Kira paraları her ayın 25'inde peşin olarak ödenecektir.
3- Kira sözleşmesinin yenilenmesi halinde yenilenen her dönemin kira parası, kendinden önce gelen dönemin kira parasına, İstanbul Ticaret Odasınca belirlenecek olan ÜFE-TÜFE endeksinin yıllık ortalamaları toplamının ½ 'si oranında zam yapılarak hesaplanacaktır.
4- Kira artırımları için kiraya verenin kiracıya ihtarname keşide etmesine ve ihbarda bulunmasına gerek olmayacak ve kiracı kira parasını yukarıda 3. maddede belirtilen koşullara göre kendiliğinden hesaplayıp ona göre ödemede bulunacaktır.
5- Kiracı, sözleşmeyi yenilemek istemediği takdirde bu hususu kira süresinin bitmesinden en az 60 gün önce kiraya verene yazılı olarak bildirecektir.
6- Bu sözleşme ile ilgili olarak kiraya veren tarafından kiracıya gönderilecek her türlü bildirim kiralananın bulunduğu adrese tebliğ edilecek ve yasal açıdan geçerli olacaktır.
7- Bu sözleşmede hüküm bulunmayan hususlarda Türk Borçlar Yasasının taşınmaz kiraları ile ilgili hükümleri uygulanacaktır.
8- Bu sözleşme ile ilgili uyuşmazlıklar İstanbul Anadolu Mahkemelerince çözüme kavuşturulacaktır.
9- Kiracı, kiralananı kontratta yazılı kullanım amacı dışında kullanamaz. Kullanma amacını genişletici fiil ve davranışlara giremez. Aksi takdirde bu husus akde aykırılık teşkil edecek ve tahliye sebebi olacaktır.
10- Kiracı, kira sözleşmesi yapılıp kiralanana taşınmadan önce kiralananın elektrik su ve doğalgaz aboneliklerini ilgili kurumlarda kendi adına tesis ettirecek ve kiralanan bu işlemlerin tamamlanmasından sonra kendisine teslim edilecektir.
11- Kiracıdan 1000.-TL tutarında depozito alınmış olup bu depozito hiçbir şekilde kira parasına mahsup edilmeyecek ve tahliye sırasında kiracının kiralanana normal kullanımdan kaynaklanan yıpranma dışında bir zarar vermediğinin tespiti halinde kiraya veren tarafından kiracıya aynen geri verilecektir.
12- Kiralanan kiracıya boya ve badanası yapılmış bir şekilde ve eksiksiz olarak teslim edilmiş olup kiracı da tahliye sırasında kiralananı kiraya verene aynı şekilde teslim edecektir.
13- Bu sözleşme taraflar arasında düzenlenip birlikte okunmuş ve içeriğinin tarafların bu konudaki iradelerini yansıttığı görülünce müştereken imzalanıp bir örneği kiralayana, bir örneği de kiracıya verilmiştir.

27.04.2023

KİRACI	KİRAYA VEREN
AS-COMSAT Haberleşme Sistemleri, Yazılım, Donanım ve Danışmanlık Hizmetleri Limited Şirketi	

4.4.3 KİRA KONTRATOSU

Dairesi	: ÜMRANİYE
Mahallesi	: TEPEÜSTÜ
Sokağı	: KAFİYE
Numarası	: 47 D:3
Kiralanan şeyin cinsi	: KONUT
Kiraya verenin adı, soyadı : ve ikametgahı	 MERDİVENKÖY MH. ÇÖMLEKÇİ ÇUKURU SK. NO:47A KOSOVA APT. A BLOK, D:14 MERDİVENKÖY- KADIKÖY/ İSTANBUL
Kiracının adı, soyadı ve ikametgahı	: AS-COMSAT Haberleşme Sistemleri, Yazılım, Donanım ve Danışmanlık Hizmetleri Limited Şirketi TEPEÜSTÜ MH. KAFİYE SOK. NO:47, D:3 ÜMRANİYE/ İSTANUL
Bir senelik kira karşılığı Bir aylık kira karşılığı	: 12.000 TL (Onikibin Türk Lirası) : 1000.-TL (Dokuzuyüzyetmişbeş Türk Lirası)
Kira karşılığının ne şekilde: ödeneceği	Her ayın 2'sinde peşin olarak ödenecektir.

Kira müddeti	: 1 yıl
Kiranın başlangıcı	: 2.Mayıs.2023
Kiralanan şeyin şimdiki durumu	: Tam ve mükemmel
Kiralanan şeyin ne için kullanılacağı	: İşyeri olarak kullanılacaktır.
Kiralanan şey ile beraber teslim olunan demirbaş eşyanın beyanı Aspiratör ve kombi.	

1-) 24.12.1980 gün ve 2361 sayılı kanunla değiştirilen 193 sayılı Gelir Vergisi Kanununun 70.94.ve 106. maddeleri gereğince;

Tüccar, serbest meslek erbabı ve çiftçiler, ticari, mesleki ve zirai işleri ile ilgili olarak yaptıkları kira ödemelerinden %25 oranında gelir vergisi tevkifatı yaparak vergi dairesine yatıracaklardır.

2-) 488 sayılı Damga Vergisi Kanununu değiştiren 21.11.1980 gün ve 2344 sayılı ve 30.12.1980 gün ve 2367 sayılı kanunlar gereğince;

Malsahibi ve kiracı için %01 kefil için %05 olmak üzere mukavele müddetine göre bütün kira bedelinin binde altısı nispetinde damga pulu yapıştıracaktır. Teslimat hakkındaki şerhlerde her imza için 501 liradan 1000 liraya kadar(1000 dahil) 10 lira, 1001 liradan 5000 liraya kadar (5000) dahil 15 lira, 5001 liradan 10.000 liraya kadar (10.000 dahil) 20 lira, 10.001 liradan 100.000 liraya kadar(100.000 dahil) 30 lira 100.001 liradan 500.000 liraya kadar olan (500.000 dahil) 50 lira, 500.000 liradan yukarı olanlar 100 liralık damga pulu yapıştırırlar.Pullar tarih ile beraber imza veya mühür konulmak suretiyle iptal olunacaktır.

İki pul için bir iptal muamelesi kabul edilir.

4.4.4 RENTAL CONTRACT (translated by google translator)

Department : UMRANİYE

Neighborhood : TEPEUSTU

Street : Kafiye

Number : 47 D:3

Type of rented property : HOUSING

Name, surname of the lessor: Şennur BAYAR (T.C NO: 56554410246)

and its residence is MERDİVENKÖY MH. ÇÖMLEKÇİ ÇUKURU SK. NO:47A
KOSOVO APT. A BLOK, D:14 MERDİVENKÖY-
KADIKOY/ ISTANBUL

Tenant's name, surname: AS-COMSAT Communication Systems, Software, Hardware and Consulting Services Limited Company

and his residence is TEPEÜSTÜ MH. KAFIYE SOK.
NO:47, D:3 UMRANİYE/ ISTANUL

One-year rental equivalent: 12.000 TL (Twelve Thousand Turkish Liras)

One month's rent equivalent: 1000.-TL (Nine hundred and seventy-five Turkish Liras)

How the rent is paid: It will be paid in advance on the 2nd of each month.

to be paid

Rental period : 1 year

Beginning of the lease : 2.May.2023

The present of the rented thing: Complete and perfect

status

What the rented thing is for: It will be used as a workplace.

to be used

Declaration of the fixed goods delivered together with the rented thing

Aspirator and combi.

1-) In accordance with Articles 70.94 and 106 of the Income Tax Law No. 193 amended by Law No. 2361 of 24.12.1980;

Traders, self-employed and farmers will deposit 25% of their income tax withholding from their rent payments related to their commercial, professional and agricultural works to the tax office.

2-) In accordance with the laws numbered 21.11.1980 and numbered 2344 and day 30.12.1980 and numbered 2367 amending the Stamp Duty Law No. 488;

It will affix a stamp at the rate of six per thousand of the entire rental price according to the contract period, 01% for the owner and the tenant, and 05% for the surety. In the annotations about the delivery, for each signature from 501 liras to 1000 liras (including 1000), 10 liras, from 1001 liras to 5000 liras (5000 included) 15 liras, from 5001 liras to 10,000 liras (including 10,000) 20 liras, from 10,001 to 100,000 liras (including 100,000)) 30 liras from 100,001 liras to 500,000 liras (including

500,000) 50 liras, and those over 500,000 liras affix a 100 lira stamp stamp. Stamps will be canceled by placing a signature or seal together with the date.

A cancellation treatment is accepted for two stamps.

4.4.5 Visit from Government

[11:37, 3.5.2023] Salih Bayar Istanbul: "A polling officer will come from the tax office. An SMS will be sent to Mr. Mourad's phone 1 day before he arrives.

You can say: "We do not have an insured employee, we pay a net rent of 1.000 TL".

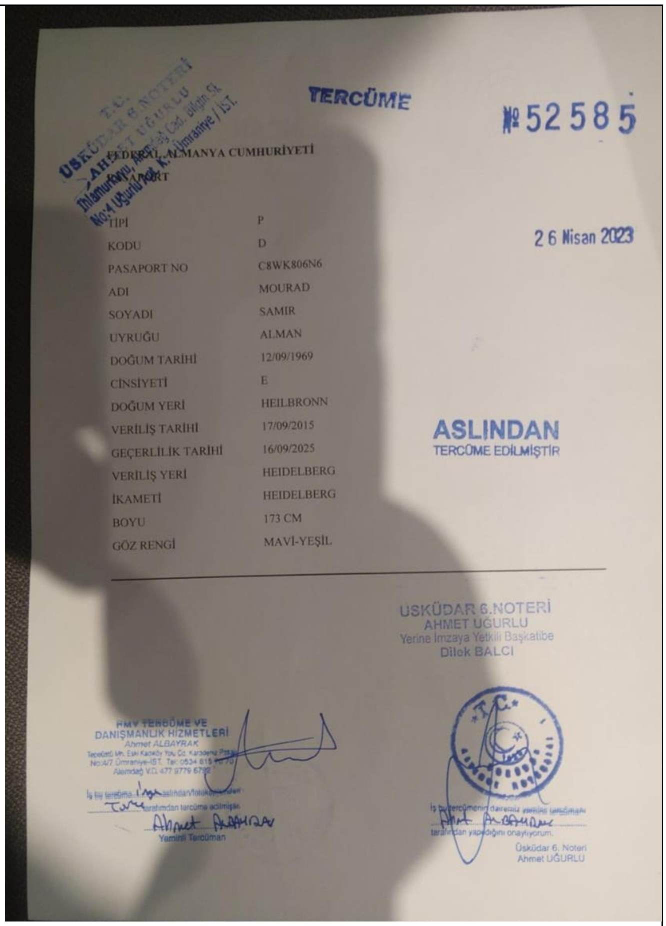
They do not enter the house, they ask at the door and take a photo, and then no one comes."

[11:37, 3.5.2023] Salih Bayar Istanbul: 😊

[11:38, 3.5.2023] Salih Bayar Istanbul: I think I will get an SMS. I will be there on that day inshaAllah.

4.4.6 Foundation of AS COMSAT (AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VETİCARET LİMİTED ŞİRKETİ)

4.4.6.1 Wikala from Samir Mourad to Salih Bayar at noter for founding the company (26.4.23)



TERCÜME

№52585

26 Nisan 2023

FEDERAL ALMANYA CUMHURİYETİ

TİPİ	P
KODU	D
PASAPORT NO	C8WK806N6
ADI	MOURAD
SOYADI	SAMIR
UYRUĞU	ALMAN
DOĞUM TARİHİ	12/09/1969
CİNSİYETİ	E
DOĞUM YERİ	HEILBRONN
VERİLİŞ TARİHİ	17/09/2015
GEÇERLİLİK TARİHİ	16/09/2025
VERİLİŞ YERİ	HEIDELBERG
İKAMETİ	HEIDELBERG
BOYU	173 CM
GÖZ RENGİ	MAVİ-YEŞİL

ASLINDAN TERCÜME EDİLMİŞTİR

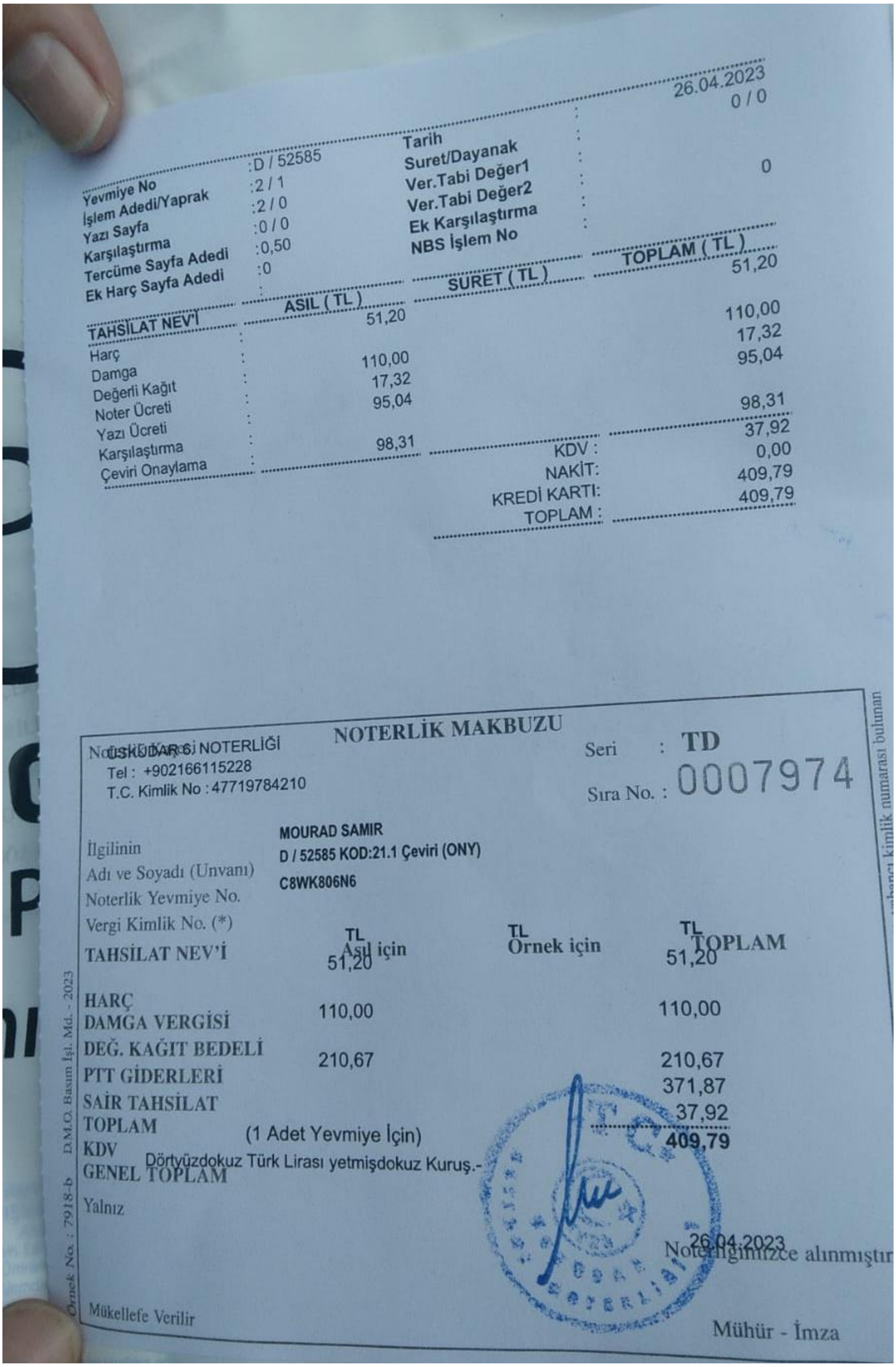
ÜSKÜDAR 6.NOTERİ AHMET UĞURLU Yerine İmzaya Yetkili Başkatibe Dilek BALCI

DANIŞMANLIK HİZMETLERİ Ahmet ALBAYRAK

Yeminli Tercüman

Üsküdar 6. Noteri Ahmet UĞURLU

Translation of passport of Samir Mourad



Yevmiye No	:D / 52585	Tarih	:	26.04.2023
İşlem Adedi/Yaprak	:2 / 1	Suret/Dayanak	:	0 / 0
Yazı Sayfa	:2 / 0	Ver.Tabi Değer1	:	0
Karşılaştırma	:0 / 0	Ver.Tabi Değer2	:	
Tercüme Sayfa Adedi	:0,50	Ek Karşılaştırma	:	
Ek Harç Sayfa Adedi	:0	NBS İşlem No		

TAHSİLAT NEV'İ	ASIL (TL)	SURET (TL)	TOPLAM (TL)
Harç :	51,20		51,20
Damga :	110,00		110,00
Değerli Kağıt :	17,32		17,32
Noter Ücreti :	95,04		95,04
Yazı Ücreti :			
Karşılaştırma :			
Çeviri Onaylama :	98,31		98,31
		KDV :	37,92
		NAKİT:	0,00
		KREDİ KARTI:	409,79
		TOPLAM :	409,79

NOTERLİK MAKBUZU

Noterliğin Adı: ÜSKÜDAR 6. NOTERLİĞİ
Tel : +902166115228
T.C. Kimlik No : 47719784210

Seri : TD
Sıra No. : 0007974

İlgilinin
Adı ve Soyadı (Unvanı): MOURAD SAMIR
Noterlik Yevmiye No.: D / 52585 KOD:21.1 Çeviri (ONY)
Vergi Kimlik No. (*): C8WK806N6

TAHSİLAT NEV'İ	Asıl için TL	Örnek için TL	TOPLAM TL
HARÇ	51,20		51,20
DAMGA VERGİSİ	110,00		110,00
DEĞ. KAĞIT BEDELİ	210,67		210,67
PTT GİDERLERİ			371,87
SAİR TAHSİLAT			37,92
TOPLAM			409,79
KDV			
GENEL TOPLAM			

(1 Adet Yevmiye İçin)

Yalnız Dörtyüzdokuz Türk Lirası yetmişdokuz Kuruş.-

26.04.2023 Noterliğimizce alınmıştır

Mükellefe Verilir

Mühür - İmza

Örnek No.: 7918-b D.M.O. Basım İşl. Md. - 2023

(*) yabancı kimlik numarası bulunan

Türkiye Cumhuriyeti

Tarih: 26/04/2023
Yev.No: (A)

VEKALETNAME

SÜRELİDİR: Bu vekaletname **26/05/2023 (Yirmialtı Mayıs İkibinyirmiüç) Cuma günü mesai bitimine kadar** geçerlidir.

Adıma her tür ticari şirketler kurmaya, şirket ana sözleşmesi ve eklerini dilediği şartlarla tanzim ve imzaya, şirket ana sözleşmesinde yapılması talep edilecek olan düzeltme ve tadilatları yapmaya, düzeltme beyannameleri ve tadil sözleşmelerini imzaya, mahkemelere başvurarak izinler almaya, Bakanlıklar, belediyeler, ticaret sicil müdürlükleri ve sair tüm kurum ve kuruluşlarda yapılacak işlemleri yapmaya, gerekli tescilleri yaptırmaya, ticaret sicili müdürlüğünde ticaret sicili gazetesinde ilan ettirmeye, gerekli harç ve vergileri yatırmaya, şirketin ilgili vergi dairesine müracaat ederek mükellefiyet kaydı tesis ettirmeye, vergi numarasını almaya, ilgili belediye ve kurumlara müracaatla yerleşim yeri belgesi almaya, ticaret sicil tasdiknamesi ile faaliyet belgesini ilgili ticaret sicili müdürlüğünden almaya, ticaret odası ve sanayi odasında yapılması gerekli işlemleri yapmaya, başvurularda bulunmaya, kurulacak şirketlerle ilgili olarak bilumum bankalarda teminat hesapları açtırmaya, teminat ve blokajlar yaptırmaya, iadelerini talep etmeye ve geri almaya, AHZU KABZA, her türlü evrak ve belgeyi imzaya , bu hususta yapılması gereken her türlü iş ve işlemleri yapmaya, takip etmeye ve imzası ile neticelendirmeye, **26/05/2023 (Yirmialtı Mayıs İkibinyirmiüç) Cuma günü mesai bitimine kadar** yetkili olmak üzere baba adı **İSMAİL** doğum tarihi **28/08/1980** olan **20647473968** T.C.Kimlik Numaralı **SALİH BAYAR** tarafımdan vekil tayin edildi.

VEKİL EDEN
SAMIR MOURAD C8WK806N6
Tepeüstü Mah. Kafiye Sk. N:47 D :3 Ümraniye / İSTANBUL
Samir Mourad 5077664818

TERCÜMAN BEYANI: **(İNGİLİZCE)** dilini konuşan **SAMIR MOURAD** isimli kişinin gerçek isteğini**ÜSKÜDAR 6. NOTERİ AHMET UĞURLU Vekili İmzaya Yetkili Başkatip DİLEK BALCI** isimli kişiye aktardım. Yazılan metni ilgiliye tercüme ettim. İlgili gerçek isteklerinin aynen yukarıda yazılmış olduğunu söyledi. Ben de ilgilinin bu metni kabul ederek yanımda imzaladığını beyan ederim.

YEMİNLİ TERCÜMAN
AHMET ALBAYRAK 47797796792
Ayazağa Mah. 170. Sk. No: 38 / 14 Sarıyer / İSTANBUL

Bu Onaylama işlem altındaki imzanın gösterdiği **ÜSKÜDAR 6** Noterliği'nin **26/04/2023** tarih ve **52585** numaralı işlem ile çevirisi yapılmış olan **HEIDELBERG** verilme, geçerlilik tarihi **16/09/2025** , **PASAPORT ÇEVİRİSİ** kimliğine göre **ALMAN** uyruklu, **12/09/1969** doğum tarihli, **İNGİLİZCE** dilini bilen, **okuryazar olduğunu , Türkçe bilmediğini** , halen yukarıdaki adreste bulunduğunu pasaport numarasının **C8WK806N6** olduğunu bildiren **SAMIR MOURAD** isimli kişiye ait olduğunu , ilgili **SAMIR MOURAD** Türkçe bilmeyip **(İNGİLİZCE)** dilini bildiğinden kendisi ile anlaşılabileceğini bildiğimiz dairemiz yeminli tercümanı **47797796792** T.C. kimlik numaralı **AHMET ALBAYRAK** tarafından bu metnin okunup anlatılması ve kabul edilmesi üzerine noterlikte huzurumda alındığını, onaylarım. **(Yirmialtı Nisan İkibinyirmiüç) Çarşamba günü 26/04/2023**

ÜSKÜDAR 6. NOTERİ
AHMET UĞURLU
Vekili
İmzaya Yetkili Başkatip
DİLEK BALCI

i ve Değerli Kağıt bedeli makbuz karşılığı tahsil edilmiştir.
d: 1.12.1
41406 - **3660280262**

A-1 / 1 - 1

Republic of Türkiye

Date: 26/04/2023
Reg. No: (A)

POWER OF ATTORNEY

TIME: This power of attorney expires on Friday, 26/05/2023 (Twenty Six May Two Thousand and Twenty Three). until valid.

To establish all kinds of commercial companies on my behalf, to issue and sign the articles of association and its annexes, to make corrections and amendments to be requested to be made in the company's articles of association, to sign correction declarations and amendment agreements, to obtain permissions by applying to the courts, to ministries, municipalities, trade registry directorates and To carry out the transactions to be made in all other institutions and organizations, to make the necessary registrations, to have them announced in the trade registry gazette at the trade registry directorate, to deposit the necessary fees and taxes, to apply to the relevant tax office of the company to establish a taxpayer registration, to obtain the tax number, to apply to the relevant municipalities and institutions, to obtain the trade registry certification and activity certificate from the relevant trade registry directorate, to carry out the necessary procedures in the chamber of commerce and industry, to make applications, to open collateral accounts in all banks in relation to the companies to be established, to have guarantees and blockages, to request their returns and to take them back. AHZU KABZA is authorized to sign all kinds of documents and documents, to carry out all kinds of work and transactions that need to be done in this regard, to follow up and conclude with his signature, until the end of the working day on Friday, 26/05/2023 (Twenty-six May, two thousand and twenty), father's name ISMAIL was born. SALIH BAYAR with the Turkish Republic ID Number 20647473968 with a date of 28/08/1980 was appointed as my proxy.

ATTORNEY
SAMIR MOURAD C8WK806N6
Tepeüstü Mah. Rhyme Sk. N:47 D:3 Umraniye/ISTANBUL
Samir Moured

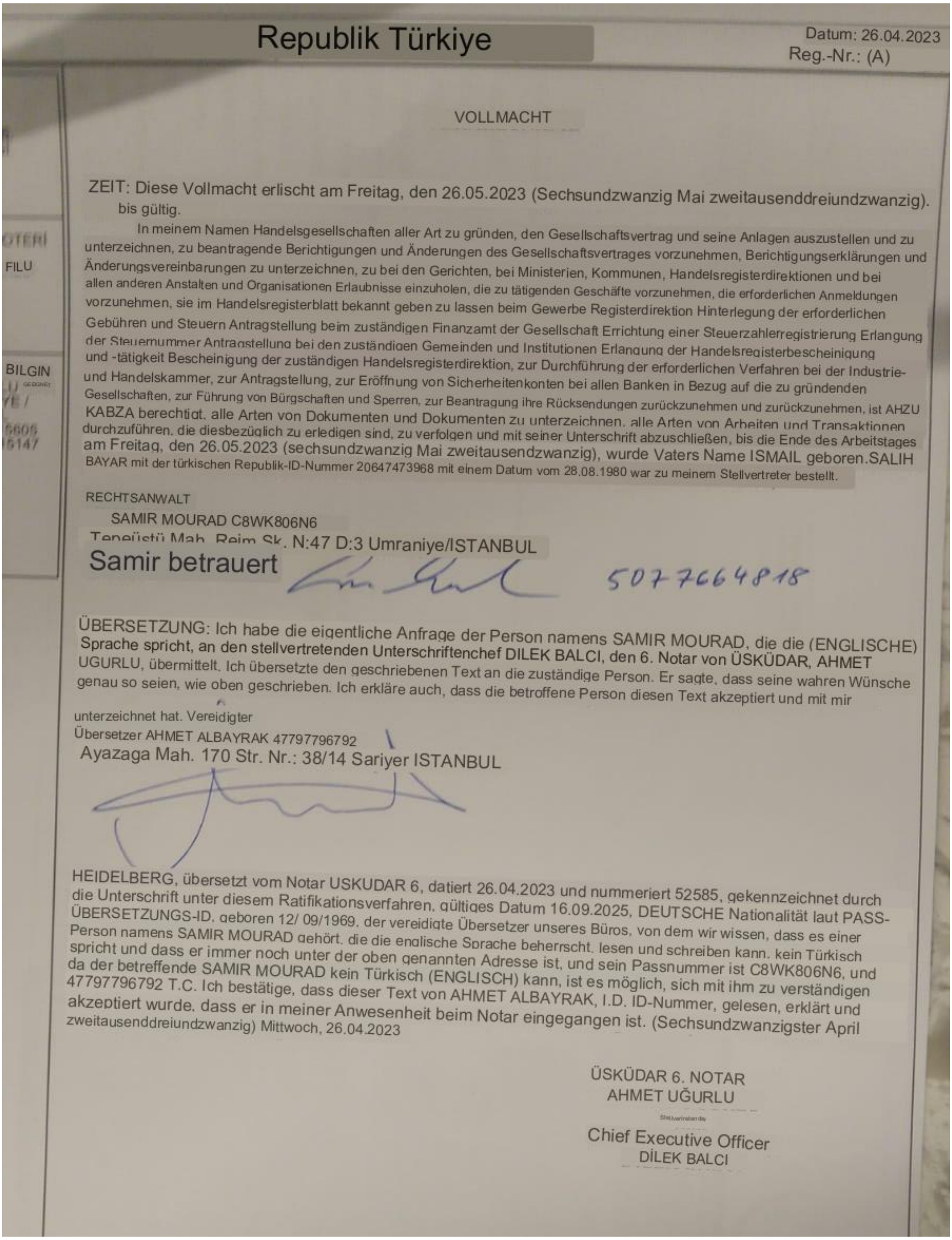
5077664818

TRANSLATION STATEMENT: I conveyed the real request of the person named SAMIR MOURAD, who speaks the (ENGLISH) language, to the Deputy Signature Chief DILEK BALCI, the 6th NOTary Public of ÜSKÜDAR, AHMET UGURLU. I translated the written text to the relevant person. He said that his true wishes are exactly as written above. I also declare that the person concerned has accepted this text and signed it with me. Sworn Translator
AHMET ALBAYRAK 47797796792
Ayazaga Mah. 170 St. No: 38/14 Sariyer ISTANBUL

HEIDELBERG, which was translated by the USKUDAR 6 Notary Public, dated 26/04/2023 and numbered 52585, indicated by the signature under this ratification process, valid date 16/09/2025, GERMAN nationality according to PASSPORT TRANSLATION ID, born 12/09/1969 , the sworn translator of our office, which we know that it belongs to a person named SAMIR MOURAD, who knows the English language, is literate, does not speak Turkish, and that he is still at the address above, and his passport number is C8WK806N6, and since the relevant SAMIR MOURAD does not know Turkish (ENGLISH), it is possible to be understood with him 47797796792 T.C. I certify that this text was read, explained and accepted by AHMET ALBAYRAK, I.D. ID number, that it was received in my presence at the notary public. (Twenty-Six April Two Thousand and Twenty-Three) Wednesday, 26/04/2023

ÜSKÜDAR 6th NOTARY
AHMET UĞURLU
Acting
Chief Executive Officer
DİLEK BALCI

Translated with google tranlater (there are mistakes)



Republik Türkiye

Datum: 26.04.2023
Reg.-Nr.: (A)

VOLLMACHT

ZEIT: Diese Vollmacht erlischt am Freitag, den 26.05.2023 (Sechsundzwanzig Mai zweitausenddreiundzwanzig). bis gültig.

In meinem Namen Handelsgesellschaften aller Art zu gründen, den Gesellschaftsvertrag und seine Anlagen auszustellen und zu unterzeichnen, zu beantragende Berichtigungen und Änderungen des Gesellschaftsvertrages vorzunehmen, Berichtigungserklärungen und Änderungsvereinbarungen zu unterzeichnen, zu bei den Gerichten, bei Ministerien, Kommunen, Handelsregisterdirektionen und bei allen anderen Anstalten und Organisationen Erlaubnisse einzuholen, die zu tätigenden Geschäfte vorzunehmen, die erforderlichen Anmeldungen vorzunehmen, sie im Handelsregisterblatt bekannt geben zu lassen beim Gewerbe Registerdirektion Hinterlegung der erforderlichen Gebühren und Steuern Antragstellung beim zuständigen Finanzamt der Gesellschaft Errichtung einer Steuerzahlerregistrierung Erlangung der Steuernummer Antragstellung bei den zuständigen Gemeinden und Institutionen Erlangung der Handelsregisterbescheinigung und -tätigkeit Bescheinigung der zuständigen Handelsregisterdirektion, zur Durchführung der erforderlichen Verfahren bei der Industrie- und Handelskammer, zur Antragstellung, zur Eröffnung von Sicherheitenkonten bei allen Banken in Bezug auf die zu gründenden Gesellschaften, zur Führung von Bürgschaften und Sperren, zur Beantragung ihre Rücksendungen zurückzunehmen und zurückzunehmen, ist AHZU KABZA berechtigt, alle Arten von Dokumenten und Dokumenten zu unterzeichnen, alle Arten von Arbeiten und Transaktionen durchzuführen, die diesbezüglich zu erledigen sind, zu verfolgen und mit seiner Unterschrift abzuschließen, bis die Ende des Arbeitstages am Freitag, den 26.05.2023 (sechsundzwanzig Mai zweitausendzwanzig), wurde Vaters Name ISMAIL geboren.SALIH BAYAR mit der türkischen Republik-ID-Nummer 20647473968 mit einem Datum vom 28.08.1980 war zu meinem Stellvertreter bestellt.

RECHTSANWALT
SAMIR MOURAD C8WK806N6
Tepeüstü Mah. Reim Sk. N:47 D:3 Umraniye/ISTANBUL
Samir betrauert

5077664818

ÜBERSETZUNG: Ich habe die eigentliche Anfrage der Person namens SAMIR MOURAD, die die (ENGLISCHE) Sprache spricht, an den stellvertretenden Unterschriftenchef DILEK BALCI, den 6. Notar von ÜSKÜDAR, AHMET UGURLU, übermittelt. Ich übersetzte den geschriebenen Text an die zuständige Person. Er sagte, dass seine wahren Wünsche genau so seien, wie oben geschrieben. Ich erkläre auch, dass die betroffene Person diesen Text akzeptiert und mit mir unterzeichnet hat. Vereidigter Übersetzer AHMET ALBAYRAK 47797796792
Ayazaga Mah. 170 Str. Nr.: 38/14 Sariyer ISTANBUL

HEIDELBERG, übersetzt vom Notar USKUDAR 6, datiert 26.04.2023 und nummeriert 52585, gekennzeichnet durch die Unterschrift unter diesem Ratifikationsverfahren, gültiges Datum 16.09.2025, DEUTSCHE Nationalität laut PASS-ÜBERSETZUNGS-ID, geboren 12/ 09/1969, der vereidigte Übersetzer unseres Büros, von dem wir wissen, dass es einer Person namens SAMIR MOURAD gehört, die die englische Sprache beherrscht, lesen und schreiben kann, kein Türkisch spricht und dass er immer noch unter der oben genannten Adresse ist, und sein Passnummer ist C8WK806N6, und da der betreffende SAMIR MOURAD kein Türkisch (ENGLISCH) kann, ist es möglich, sich mit ihm zu verständigen 47797796792 T.C. Ich bestätige, dass dieser Text von AHMET ALBAYRAK, I.D. ID-Nummer, gelesen, erklärt und akzeptiert wurde, dass er in meiner Anwesenheit beim Notar eingegangen ist. (Sechsundzwanzigster April zweitausenddreiundzwanzig) Mittwoch, 26.04.2023

ÜSKÜDAR 6. NOTAR
AHMET UĞURLU

Chief Executive Officer
DİLEK BALCI

Translated with google tranlater (there are mistakes)

4.4.7 Meeting with financial advisors Mesut Yildiz at Fatih Derya at 27.4.23

[11:06, 3.5.2023] Salih Bayar Istanbul:

GENEL MUHASEBE VEKALETİ İÇİN İSİMLER

FATİH DERYA : 4264 955 4728

MESUT YILDIZ : 5410 018 1572

SERHAT TENGİZ : 2603 940 2400

FURKAN ÇALIK : 2625 510 1248

SERKAN SÖNMEZ : 1560 545 6428

TUTKU SARPKAYA : 3629 268 7452

4.4.8 Meeting Date with Industry Ministry

Randevu Bilgileri

Unvan
AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ
Firma Türü
ORTAK SAYISI BİRDEN FAZLA LİMİTED ŞİRKET
Randevu Alınan TSM
PERPA BÖLGE TEMSİLCİLİĞİ
Randevu Tarihi
02.05.2023
Randevu Saati
14:20
Randevu Durumu
Randevu rezervasyonunuz işlem sırasına alınmıştır.
Talep Türü
Kuruluş
Talep No
6322274

Randevu rezervasyonunuz işleme alınmıştır. Ticaret sicili müdürlüğü tarafından uygun bulunması halinde tarafınıza mail olarak iletilecektir ve ilgili ekranlardan da takip edebilirsiniz.

Randevuya zamanında gelinmemesi halinde, yeni randevu ancak ileri bir tarihe alınabilir.

Machine Translated by Google

Termininformationen

Titel
AS COMSAT KOMMUNIKATIONSPLATTFORMEN UND SATELLITENKOMMUNIKATIONSSYSTEME INDUSTRIE UND HANDELSGESELLSCHAFT BEGRENZT
Firmentyp
ANZAHL DER PARTNER MEHR ALS EINE LIMITED COMPANIES
TSM nach Vereinbarung
PERPA REGIONALVERTRETER
Termin
02.05.2023
Termin
14:20
Terminstatus
Ihre Terminreservierung wurde in die Bearbeitungswarteschlange gestellt.
Anfragetyp
Einrichtung
Anfrage Nr.
6322274

ÿ Ihre Terminreservierung wurde bearbeitet. Wenn es von der Handelsregisterdirektion genehmigt wird, wird es Ihnen per E-Mail zugesandt und Sie können es auf den entsprechenden Bildschirmen verfolgen.

ÿ Wird der Termin nicht fristgerecht wahrgenommen, kann der neue Termin nur auf einen späteren Zeitpunkt verschoben werden.

4.4.9 Circular of Signatures (IMZA SIRKÜLERI) - Needed for all official issues (bank account, import, export, …)

Türkiye Cumhuriyeti

Tarih: 10/05/2023
Yev.No: (A)

T.C.
ÜSKÜDAR 6.
NOTERLİĞİ

ÜSKÜDAR 6. NOTERİ
AHMET UĞURLU

ALEMDAĞ CD. BİLGİN SOK. N:4 UĞURLU APT. K:1 ÜMRANİYE / İSTANBUL
Tel:+902166115605
Fax:+902166115147

İMZA SİRKÜLERİ №54999

ÜNVANI	: **AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ**
ADRES	: Tepeüstü Mah. Kafiye Sk. - No: 47 / 3 Ümraniye / İSTANBUL
YETKİLİ	: SAMIR MOURAD
YETKİNİN KULLANIM ŞEKLİ	: MÜNFERİDEN/AKSİ KARAR ALININCAYA KADAR
TEMSİL ŞEKLİ / SÜRESİ	: MÜNFERİDEN
TİCARET SİCİL ADI - NO	: İSTANBUL - 457897-5
VERGİ DAİRESİ - NO	: 034223 - ALEMDAĞ VERGİ DAİRESİ MÜD. - 0861584690

Yukarıda adresi yazılı AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ ünvanlı şirketin Türkiye Ticaret Sicili Gazetesinin 02.05.2023 tarih, 10822 sayılı nüshasının 983. ve 984. sayfalarında neşir ve ilan olunan İstanbul Ticaret Sicili memurluğundan 02.05.2023 tarihinde tescili olan "şirketin idaresi" ile ilgili 7.maddesi ve "temsil" ile ilgili 8.maddesine göre; Şirketin işleri ve işlemleri genel kurul tarafından seçilecek bir veya birkaç müdür tarafından yürütülür. Aksi Karar Alınıncaya Kadar SAMIR MOURAD Müdür olarak seçilmiştir. Yetki Şekli: Münferiden Temsile Yetkilidir. Şirketi müdürler temsil ederler. Şirketi temsil edecek imzalar genel kurul tarafından tespit tescil ve ilan olunur. Müdürler şirkete hizmet akdi ile bağlı olanları sınırlı yetkiye sahip ticari vekil veya diğer tacir yardımcıları olarak atayabilir. Bu şekilde atanacak olanların görev ve yetkileri hazırlanacak iç yönergede açıkça belirlenir. Bu durumda iç yönergenin tescil ve ilanı zorunludur. İç yönerge ile ticari vekil ve diğer tacir yardımcıları atanamaz. Yetkilendirilen ticari vekil veya tacir yardımcıları da ticaret siciline tescil ve ilan edilir. Bu kişilerin, Şirket ve üçüncü kişilere verilecek her türlü zarardan dolayı müdürler müteselsilen sorumludur. denildiğinden, şirket ünvanı altında kullanacağım aşağıda örnekleri bulunan imzamın onaylanmasını talep ederim.
AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ ADINA TEMSİLEN (...) SAMIR MOURAD

İMZA İMZA İMZA

TERCÜMAN BEYANI: (**İngilizce**) dilini konuşan **SAMIR MOURAD** isimli kişinin gerçek isteğini **ÜSKÜDAR 6. NOTERİ AHMET UĞURLU** isimli kişiye aktardım. Yazılan metni ilgiliye tercüme ettim. İlgili gerçek isteklerinin aynen yukarıda yazılmış olduğunu söyledi. Ben de ilgilinin bu metni kabul ederek yanında imzaladığını beyan ederim.

YEMİNLİ TERCÜMAN
AHMET ALBAYRAK 47797796792
Ayazağa Mah. 170. Sk. No: 38 / 14 Sarıyer / İSTANBUL

Bu Onaylama işlem altındaki imzanın **0861584690** vergi numaralı **AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ**

KDV, Harç, Damga Vergisi ve Değerli Kağıt bedeli makbuz karşılığı tahsil edilmiştir.
DB72 A / 5 Yazı : 3 / 0 Kod: 10.1.4
NBS NO: 202305100341406 - **0639644592**

A-3 / 1 - 2

Türkiye Cumhuriyeti

Tarih: 10/05/2023
Yev.No: (A)
№ 54999

T.C.
ÜSKÜDAR 6.
NOTERLİĞİ

ÜSKÜDAR 6. NOTERİ
AHMET UĞURLU

ALEMDAĞ CD. BİLGİN SOK. N:4 UĞURLU APT. K:1 ÜMRANİYE / İSTANBUL
Tel:+902166115605
Fax:+902166115147

SANAYİ VE TİCARET LİMİTED ŞİRKETİ adına **YETKİLİSİ** olarak hareket eden, gösterdiği **ÜSKÜDAR 6** Noterliği'nin **26/04/2023** tarih ve **52585** numaralı işlem ile çevirisi yapılmış olan **heidelberg** verilme, geçerlilik tarihi **16/09/2025** , **ÇEVİRİ PASAPORT** kimliğine göre **ALMAN** uyruklu, **12/09/1969** doğum tarihli, **İNGİLİZCE** dilini bilen, **okuryazar olduğunu , Türkçe bilmediğini** , halen yukarıdaki adreste bulunduğunu vergi numarasının **6232042167** olduğunu bildiren **SAMIR MOURAD** isimli kişiye ait olduğunu , ilgili **SAMIR MOURAD** Türkçe bilmeyip (...) dilini bildiğinden kendisi ile anlaşılabileceğini bildiğimiz dairemiz yeminli tercümanı **47797796792** T.C. kimlik numaralı **AHMET ALBAYRAK** tarafından bu metnin okunup anlatılması ve kabul edilmesi üzerine noterlikte huzurumda alındığını, onaylarım. **On Mayıs İkibinyirmiüç, Çarşamba günü 10/05/2023**

DAYANAK:Türkiye Ticaret Sicili Gazetesinin 02.05.2023 tarih, 10822 sayılı nüshasının 983. ve 984. sayfalarında neşir ve ilan olunan İstanbul Ticaret Sicili memurluğundan 02.05.2023 tarihinde tescili olan "şirketin idaresi" ile ilgili 7.maddesi ve "temsil" ile ilgili 8.maddesine göre;Şirketin temsil ve ilzamı sirküler metninde yazılı olduğu şekildedir.

ÜSKÜDAR 6. NOTERİ
AHMET UĞURLU

KDV, Harç, Damga Vergisi ve Değerli Kağıt bedeli makbuz karşılığı tahsil edilmiştir.
DB72 A / S Yazı : 3 / 0 Kod: 10.1.4
NBS NO: 202305100341406 - **0639644592**

A-3 / 2 - 2

4.4.10 General Wikala from Manager Samir Mourad to Salih Bayar for all transactions concerning the company

Türkiye Cumhuriyeti

Tarih: 10/05/2023
Yev.No: (A)

T.C.
ÜSKÜDAR 6.
NOTERLİĞİ

ÜSKÜDAR 6. NOTERİ
AHMET UĞURLU

ALEMDAĞ CD. BİLGİN SOK. N:4 UĞURLU APT. K:1 ÜMRANİYE / İSTANBUL
Tel:+902166115605
Fax:+902166115147

SOĞUK DAMGA VARDIR

№55021

VEKALETNAME

Türkiye Cumhuriyeti hudutları dahilinde bulunan Defterdarlıklar, Mal müdürlükleri, Vergi Daireleri, Sosyal Güvenlik Kurumu,(S.S.K, Bağ-Kur) Müdürlükleri ve Şubeleri, Çalışma ve İş Kurumu il Müdürlükleri ve Hizmet Merkezleri, Ticaret ve Sanayi Odaları, Ticaret Sicili Müdürlükleri, Esnaf ve Sanatkarlar Sicil Müdürlükleri, Özel İdare Müdürlükleri, Esnaf ve Sanatkarlar Odaları, Belediye Başkanlıkları ve ilgili birimleri ve bilumum resmi makam ve mercilerde beni tam yetkili olarak temsile, işlemleri takibe ve sonuçlandırmaya, her türlü izin ve ruhsatları almaya, kayıt ve tescillerini yaptırmaya, yapılacak olan tesciller ile ilgili terkin ve tadil işlemlerini yapmaya, yürütmeye, evrakları akt ve imzalamaya, gerektiğinde kayıtları sildirmeye, ilgili mercilere teftişler vermeye, ticari defterlerimi ve belgelerimi sunmaya, tarhiyat öncesi veya tarhiyat sonrası her türlü vergi ve cezalardan dolayı vergi itiraz, temyiz, uzlaşma ve takdir komisyonlarında, vergi mahkemelerinde beni temsile, hak ve menfaatlerimi savunmaya, uzlaşma talebinde bulunmaya, uzlaşmaya, tutanakları imzaya, ilgili vergi dairelerine başvurarak fiş, fatura, irsaliye, gider pusulası, vesair belge tasdik ve basım izinlerini almaya, matbaalarda defter, belge (fiş, fatura, gider pusulası, irsaliye ve sair) bastırmaya, gerektiğinde sayılan bu defter ve belgeleri Noterliklerde tasdik ettirmeye, gerektiğinde basılı veya onaylı belgeleri ilgili vergi dairesine müracaatla iade etmeye, iptal ettirmeye, iptal ve iade tutanaklarını tanzim ve imzaya, teslim almaya, kayıt ve suretler çıkarttırmaya, her türlü yazılı ve sözlü beyan ve izahatlarda bulunmaya, tahakkuk etmiş veya edecek olan vergi iadelerini ve fazla yatırılanları ödenecek olan her türlü vergilerden mahsup ettirmeye, tebliğ ve tebellüğe, Sosyal Güvenlik Kurumunca tahakkuk ettirilmiş veya ettirilecek olan prim ve cezalarına itiraza, esnaf ve sanatkarlar odasına kayıt ve tescil işlemlerini yaptırmaya, Hazine ve Maliye Bakanlığı, Defterdarlıklar, Vergi Daireleri ve Mal Müdürlüklerine verilmesi gereken beyanname ve bildirimleri elektronik ortamdan ve internetten yararlanarak verilmesi için gerekli şifre kullanıcı kodlarını almaya, Vergi Dairelerinde yapılacak her türlü ödemelerin ve borçların izlenmesi için, elektronik ortam ve internet kullanım şifrelerimi almaya, bu şifreleri kullanmaya, elektronik ortamda beyannamelerin ve tahakkukların alınması ve ödenmesi için gerekli başvuruları yapmaya, düzeltme talebinde bulunmaya, belirttiğim konularla ilgili olarak yapılması gereken her türlü yasal işlemleri tüm resmi makam ve merciler önünde yapmaya ve imzalamaya, aylık prim ve hizmet belgesinin Sosyal Güvenlik Kurumu'na internet ortamında verilebilmesi için anılan kuruma veya il müdürlüklerine başvuruda bulunmaya, e-bildirge ve e-beyanname sözleşmesini imzalamaya, kullanıcı kodu ve kullanıcı şifresi zarfını kurumdan imza karşılığında almaya, iş yerinde çalışan sigortalılara ilişkin aylık prim ve hizmet belgesinin internet ortamında kuruma gönderilmesi ve bu konudaki diğer işlemlerin yerine getirilmesi hususunda beni temsil ve ilzama, yürürlükte bulunan V.U.K. Genel Tebliğleri gereğince, Elektronik Tebligata ilişkin şifre talebinde bulunmaya, interaktif vergi dairesi şifre talebinde bulunmaya, bu talepler ile ilgili dilekçe ve formları düzenlemeye ve imzalamaya, dilekçe veya formları ilgili Vergi Dairesi veya Mal Müdürlüğüne teslim etmeye, Elektronik tebligat ve interaktif vergi dairesi şifrelerini teslim almaya, resmi kurum ve kuruluşlarda gerekli iş ve işlemleri yapmaya, ilgili vergi dairelerinde mal müdürlüklerinde uzlaşma talebinde bulunmaya, uzlaşmaya girmeye, yazar kasa veya yazar kasalı P.O.S. cihazı izin ve ruhsatlarını almaya, onaylattırmaya, tekel izin ve ruhsatları almaya ve onaylattırmaya, ilgili vergi daireleri, mal müdürlükleri ve S.G.K.'na müracaatla işyeri açılış ve işyeri kapanış işlemlerimi yapmaya, işyeri açılış ve kapanış tutanaklarını imzaya, bu hususlarda yapılması gereken her türlü iş ve işlemleri yapmaya, takip etmeye ve imzası ile neticelendirmeye, **HESAP AÇMA, PARA ÇEKME :** T.C. hudutları dahilinde halen mevcut bulunan ve bundan böyle mevcut olacak olan bilumum resmi, yarı resmi, hususi, milli ve yabancı bankalar ile özel finans kurumları nezdinde adıma TL ve/veya döviz, vadeli vadesiz her türlü YENİ HESAPLAR AÇMAYA, hesap açılışı için gerekli olan hesap açılış sözleşmelerini ve diğer belgeleri imzalamaya, bu yeni açılacak hesaplar ile bu güne kadar açılmışmevcut hesaplar ve bu günden sonra açılacak diğer her nevi hesaplarımdan dilediği miktarda gerek Türk parası gerekse döviz olarak para çekmeye, hesaplara para yatırmaya, ATM(bankamatik) kartı talep etmeye, verilecek kart ve şifrelerini teslim almaya, yeni hesaplar açmaya, açılmış ve açılacak hesapları dilerse kapatmaya, vadeli hesapların vadelerini bozmaya, yenilemeye, ahzu kabza, makbuz ve ibralar vermeye, adıma bankalara gelmiş ve gelecek olan gerek Türk parası ve gerekse döviz cinsindeki havaleleri almaya, her türlü hesaplarım arasında virman yapmaya, her türlü hesaplarımdan havaleler yapmaya, başka banka hesaplarına EFT yapmaya, ibralar vermeye, hesap özeti, ekstre ve ayrıntılı hesap dökümleri almaya, gerektiğinde internet bankacılığı işlemlerimi takip etmeye, elektronik ortamda bankacılık işlemleri yapmaya, internet bankacılığı ile ilgili şifreler talep etmeye, şifreleri elden teslim almaya, dilerse değiştirmeye, dilerse yenisini talep etmeye, her türlü evrak ve belgeyi tanzim ve imzalamaya, tüm bu hususlarla ilgili olarak yapılması gereken iş ve işlemleri yapmaya, takip etmeye ve imza ile neticelendirmeye

İŞYERİ İDARESİ :Tepeüstü Mah. Kafiye Sk. - No: 47 / 3 Ümraniye / İSTANBUL adresindeki şirketimizin işyerindeki faaliyet konumla ilgili olarak tüm işlerimi yürütmeye, her türlü sözleşmeler tanzim ve imzaya , adıma ticari mal almaya, satmaya, bedellerini tahsile, fatura, fiş ve makbuzları tanzim ve imzalamaya, işyerimle ilgili her türlü evrak, belge, izin, ruhsatları ilgili makamlardan temin etmeye, defter,belge ve fatura onaylattırmaya,bastırmaya ve teslim almaya, ticari defterlerimi, işyerimle ilgili belgeleri ibraza, tutulacak tutanakları imzalamaya, itirazlarda bulunmaya, yazışmalar yapmaya, tebliğ ve tebellüğe, işyerimde işçi çalıştırmaya, mesul müdür ve işçilerle sözleşmeler yapmaya, feshetmeye, ihtarname, ihbarname çekmeye, işveren vekili sıfatı ile S.G.K. Sosyal Sigorta bordrolarını tanzim ve imzalamaya, Ticaret ve Sanayi Odaları, Ticaret Sicili Müdürlüğü, Esnaf ve Sanatkarlar Odası ile esnaf derneklerine kayıt ve tescil işlemlerimi yaptırmaya, işyerim ve faaliyet konumla ilgili olarak Maliye, Vergi daireleri, SGK, Özel idare, Belediyeler, Emniyet Müdürlüğü, Sağlık Müdürlüğü, Hükümet Tabipliği, Mülki ve Askeri makamlar ve tüm resmi dairelerde özel ve tüzel kişiler nezdinde yapmam gereken bilumum işlemleri, başvuruları benim adıma yapmaya, bu işlerimle ilgili olarak istenecek taahhütname ve muvafakatnameleri tanzim ve imzalamaya, bu hususlarda yapılması gereken her türlü iş ve işlemleri yapmaya, takip etmeye ve imzası ile neticelendirmeye, yetkili olmak üzere baba adı **İSMAİL** doğum tarihi **28/08/1980** olan **20647473968** T.C.Kimlik Numaralı **SALİH BAYAR** tarafımdan vekil tayin edildi.

KDV, Harç, Damga Vergisi ve Değerli Kağıt bedeli makbuz karşılığı tahsil edilmiştir.
EÇ24 A / S Yazı : 7 / 0 Kod: 1.15.6
NBS NO: 202305100341406 - **0156506756**

A-2 / 1 - 2

Tarih: 10/05/2023
Yev.No: (A)

Türkiye Cumhuriyeti

№55021

ÜSKÜDAR 6. NOTERİ
AHMET UĞURLU

ALEMDAĞ CD. BİLGİN SOK. N:4 UĞURLU APT. K:1 ÜMRANİYE / İSTANBUL
Tel:+902166115605
Fax:+902166115147

VEKİL EDEN
AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ 0861584690
Tepeüstü Mah. Kafiye Sk. - No: 47 / 3 Ümraniye / İSTANBUL
YETKİLİSİ:
SAMIR MOURAD 6232042167

TERCÜMAN BEYANI: (**İNGİLİZCE**) dilini konuşan **SAMIR MOURAD** isimli kişinin gerçek isteğini**ÜSKÜDAR 6. NOTERİ AHMET UĞURLU** isimli kişiye aktardım. Yazılan metni ilgiliye tercüme ettim. İlgili gerçek isteklerinin aynen yukarıda yazılmış olduğunu söyledi. Ben de ilgilinin bu metni kabul ederek yanımda imzaladığını beyan ederim.

YEMİNLİ TERCÜMAN
AHMET ALBAYRAK 47797796792
Ayazağa Mah. 170. Sk. No: 38 / 14 Sarıyer / İSTANBUL

Bu Onaylama işlem altındaki imzanın **0861584690** vergi numaralı **AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ** adına **YETKİLİSİ** olarak hareket eden, gösterdiği **ÜSKÜDAR 6** Noterliği'nin **26/04/2023** tarih ve **52585** numaralı işlem ile çevirisi yapılmış olan **HEIDELBERG** verilme, geçerlilik tarihi **16/09/2025** . **PASAPORT ÇEVİRİSİ** kimliğine göre **ALMAN** uyruklu, **12/09/1969** doğum tarihli, **İNGİLİZCE** dilini bilen, **okuryazar olduğunu , Türkçe bilmediğini** , halen yukarıdaki adreste bulunduğunu vergi numarasının **6232042167** olduğunu bildiren **SAMIR MOURAD** isimli kişiye ait olduğunu , ilgili **SAMIR MOURAD** Türkçe bilmeyip (**İNGİLİZCE**) dilini bildiğinden kendisi ile anlaşılabileceğini bildiğimiz dairemiz yeminli tercümanı **47797796792** T.C. kimlik numaralı **AHMET ALBAYRAK** tarafından bu metnin okunup anlatılması ve kabul edilmesi üzerine noterlikte huzurumda alındığını, onaylarım. **On Mayıs İkibinyirmiüç, Çarşamba günü 10/05/2023**

DAYANAK: **ÜSKÜDAR 6.** Noterliği'nden **10/05/2023** tarih ve **54999** yevmiye no ile tasdikli imza sirkülerinin incelenmesinden **AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ** adına **aksi karar alınıncaya kadar Münferiden** temsile **SAMIR MOURAD** isimli kişinin yetkili olduğu görüldü.

ÜSKÜDAR 6. NOTERİ
AHMET UĞURLU

KDV, Harç, Damga Vergisi ve Değerli Kağıt bedeli makbuz karşılığı tahsil edilmiştir.
EC24 A / S Yazı : 7 / 0 Kod: 1.15.6
NBS NO: 202305100341406 - 0156506756

A-2 / 2 - 2

4.4.11 Turkish Trade Registry Newspaper

(Baştarafı 982.Sayfada)

1.Şirket amacını gerçekleştirmek için her türlü gayrimenkul alabilir, satabilir, işletebilir, kiralar, kiraya verebilir, bunlar üzerinde ayni ve şahsi hakları tesis edebilir, ipotek alabilir, ipotek verebilir ve ipotekleri fek edebilir. Gayrimenkulleri üzerinde irtifak, intifa, sükna, gayrimenkul mükellefiyeti, bedelli veya bedelsiz yola terk, kamuya terk, ifraz, tevhit yapabilir, kat irtifakı ve kat mülkiyeti tesis edebilir. Tapu dairelerinde gayrimenkullerle ilgili her türlü cins tashihi, ifraz, tevhid, taksim, parselasyon ile ilgili her nevi muamele ve tasarrufları gerçekleştirebilir.
2.Lüzumlu görülecek menkul ve gayrimenkul malları satın alabilir, satabilir, kiralayabilir, finansal kiralama yapabilir, yahut kiraya verebilir, yurt içinde ve yurt dışından, teminatlı yahut teminatsız, her türlü borç alabilir. Şirketin sahip olduğu her türlü menkul ve gayri menkulleri ile her türlü sabit tesis ve demirbaşları üzerinde gerek namına gerekse üçüncü şahıs ve tüzel kişiler namına teminat, kefalet, işletme rehni ve ipotek tesis edebilir, fek edebilir. Alacaklarına karşılık, ipotek veya rehin alabilir, borçlarının teminatı olarak veya üçüncü şahıslar lehine, ipotek yahut rehin verebilir, bu ipoteklerin terkinini talep edebilir, her türlü kefalet ve teminat verebilir.
3.Şirket konusu işlerin oluşabilmesi için kanuni mevzuatın gerekli kıldığı tüm teşebbüs ve merasimi ifa ederek gerekli izin, imtiyaz, ruhsatname, patent, ihtira beratları, lisans, know-how marka, fesihname, belge, alameti farika gibi sınai, mülkiyet haklarını ve emsali hakları iktisap etmek veya tamamen üçüncü şahıslara devredebilir, her türlü hukuki tasarruf işlemlerde bulunmak. Ayrıca, yukarıdaki hakları kiraya verebilir veya kiralayabilir.
4.Şirket gaye ve konusunun gerçekleştirilmesi için arsa, arazi, bina ve gayrimenkul hakları satın alabilir, inşa edebilir, iktisap edebilir, satabilir. Iktisap edilen gayri menkulleri devir ve ferağ, bunları kısmen veya tamamen üçüncü şahıslara kullandırabilir veya kiralayabilir.
5.Şirket gayelerinin oluşmasını teminen kısa, orta ve uzun vadeli istikrazlar akdedebilir, aval ve kefalet kredileri temin edebilir. Şirketin kuracağı tesisleri için gerekli iç ve dış kredileri dahili ve harici finansman kurumlarıyla yerli ve yabancı kuruluş ve işletmelerden temin edebilir.
6.Şirket konusuyla ilgili mamullerin depolanması, nakli, satısı için gerekli her türlü makine, tesisat, nakil vasıtaları, cihaz alet ve edevat ve sair menkul malları ve hakları satın alabilir veya satabilir, ithal ve ihraç edebilir, kullanabilir, kiralayabilir ve kiraya verebilir.
7.Şirket gayelerini gerçekleştirmek için tüm sınai mali ve hukuk işlem ve tasarruflarda bulunabilir, resmi ve özel bilumum ihalelere iştirak edebilir, acentelik, mümessillik, komisyonculuk, bayilik, distribütörlük alabilir ve verebilir.
8.Şirket konusuyla ilgili olan şirket kurabilir, ortaklık yapabilir, mevcut ve ileride kurulacak işletme ve şirketlere iştirak edebilir. Şirket konularıyla iştigal eden firmaları ve ticari işletmeleri devren iktisap edebilir, aracılık yapmamak kaydıyla kurulmuş ve kurulacak şirketlerin, hisse senetlerini ve paylarını satın alabilir veya satabilir, devredebilir ve bunlar üzerinde her türlü tasarrufta bulunabilir.
9.Şirket gayesine faydalı olmak şartıyla ve aracılık yapmamak kaydıyla şirket konusuna bağlı olmaksızın özel hukuk, kamu hukuku tüzel kişiler tarafından ihraç olunan ve olunacak hisse senetleri, tahviller, intifa senetleri ve kuponları gibi kıymetleri satın alabilir, devralabilir, teminat gösterebilir, kabul edebilir.
10.Konusu ile ilgili hammadde, yan madde, yan mamul ve mamul maddeleri satın alabilir, ithal, ihraç ve sair suretle tedarik edebilir, bunları kısmen veya tamamen işletebilir.
11.Yukarıda sözü edilen konularla ilgili yerli ve yabancı şirket işletme ve müesseseler ile işbirliği yapabilir veya bunlarla ortaklık yapabilir.
12.Yukarıda sözü edilen konularla ilgili olarak yerli ve yabancı şirket işletme ve müesseselerden distribütörlük, bayilik veya mümessillik alabilir, verebilir.
13.Şirket yukarıdaki tüm konularıyla ilgili yurt içinde ve yurt dışında resmi ve özel kurumların açtığı ihalelere katılabilir.
14.Şirket yukarıdaki tüm konularıyla ilgili olarak kanuni mevzuatlar yerine getirilerek ihracat ve ithalat işlemleri yapabilir.
15.Konusu ile ilgili fason isçilik yapabilir, yaptırabilir.
16.Konusu ile ilgili olmak üzere soğuk hava depoları kurabilir, işletebilir, kiralayabilir, kiraya verebilir.
17.Şirket maksat ve mevzuu ile ilgili olarak her türlü fuar, sergi ve panayırlara iştirak edebilir.

4. ŞİRKETİN MERKEZİ
Şirketin merkezi **TRABZON**ili **ORTAHİSAR**ilçesi'dir. Adresi **ÇARŞI MAH. EMİN ALEMDAR SK NO: 1 İÇ KAPI NO: 5 ORTAHİSAR / TRABZON**'dir.Adres değişikliğinde yeni adres, ticaret siciline tescil ve Türkiye Ticaret Sicili Gazetesi'nde ilan ettirilir. Tescil ve ilan edilmiş adrese yapılan tebligat şirkete yapılmış sayılır. Tescil ve ilan edilmiş adresinden ayrılmış olmasına rağmen, yeni adresini süresi içinde tescil ettirmemiş şirket için bu durum fesih sebebi sayılır

5. SÜRE
Şirketin süresi, kuruluşundan itibaren **sınırsız**'dır. Bu süre esas sözleşmesini değiştirmek suretiyle uzatılıp kısaltılabilir.

6. SERMAYE
Şirketin sermayesi, beheri 500,00 Türk Lirası değerinde 1000 paya ayrılmış toplam 500000,00 Türk Lirası değerindedir.Bu payların tamamı nama yazılıdır.
-Beheri 500,00 Türk Lirası değerinde 1000 adet paya karşılık gelen 500000,00 Türk Lirası SERHAT UNAT tarafından nakdi, olarak taahhüt edilmiştir.
Nakden taahhüt edilen payların itibari değerlerinin 125000,00 TL'si şirketin tescilinden önce ödenmiştir. Nakden taahhüt edilen payların geri kalan kısmı ise yönetim kurulunun alacağı kararlara göre şirketin tescilini izleyen 24 ay içinde ödenecektir.

7. YÖNETİM KURULU ve SÜRESİ
Şirketin işleri ve idaresi Genel Kurul tarafından Türk Ticaret Kanunu Hükümleri çerçevesinde seçilecek en az 1 üyeden oluşan bir yönetim kurulu tarafından yürütülür. Yönetim Kurulu süresi 3 yıldır. İlk Yönetim Kurulu üyesi olarak 3 yıllığına aşağıdakiler seçilmiştir.
Türkiye Uyruklu 351******68 Kimlik No'lu, DİYARBAKIR / YENİŞEHİR adresinde ikamet eden, SERHAT UNAT Yönetim Kurulu Üyesi olarak seçilmiştir.

8. ŞİRKETİN İDARESİ VE TEMSİLİ
Şirketin Yönetimi ve dışarıya karşı temsili Yönetim Kurulu'na aittir. Şirket tarafından verilecek bütün belgelerin ve yapılacak sözleşmelerin geçerli olabilmesi için, bunların şirket unvanı altına konmuş ve şirketi temsile yetkili kişi veya kişilerin imzasını taşıması gereklidir. Yönetim kurulu, temsil yetkisini bir veya daha fazla murahhas üyeye veya müdür olarak üçüncü kişilere devredebilir. En az bir yönetim kurulu üyesinin temsil yetkisini haiz olması şarttır.
Yönetim kurulu, düzenleyeceği bir iç yönergeye göre, yönetimi, kısmen veya tamamen bir veya birkaç yönetim kurulu üyesine veya üçüncü kişiye devretmeye yetkilidir. Bu iç yönerge şirketin yönetimini düzenler; bunun için gerekli olan görevleri tanımlar, yerlerini gösterir, özellikle kimin kime bağlı ve bilgi sunmakla yükümlü olduğunu belirler. Yönetim kurulu, istem üzerine pay sahiplerini ve korunmaya değer menfaatlerini ikna edici bir biçimde ortaya koyan alacaklıları, bu iç yönerge hakkında, yazılı olarak bilgilendirir.
Yönetim kurulu, temsile yetkili olmayan yönetim kurulu üyelerini veya şirkete hizmet akdi ile bağlı olanları sınırlı yetkiye sahip ticari vekil veya diğer tacir yardımcıları olarak atayabilir. Bu şekilde atanacak olanların görev ve yetkileri, hazırlanacak iç yönergede açıkça belirlenir. Bu durumda iç yönergenin tescil ve ilanı zorunludur. İç yönerge ile ticari vekil ve diğer tacir yardımcıları atanamaz. Yetkilendirilen ticari vekil veya diğer tacir yardımcıları da ticaret siciline tescil ve ilan edilir. Bu kişilerin, şirkete ve üçüncü kişilere verecekleri her tür zarardan dolayı yönetim kurulu müteselsilen sorumludur.

9. TEMSİLE YETKİLİ OLANLAR, TEMSİL ŞEKLİ VE GÖREV DAĞILIMI
Temsile Yetkili Olanlar ve Temsil Şekli:
İlk 3 Yıl için Türkiye Uyruklu 351******68 Kimlik No'lu, DİYARBAKIR / YENİŞEHİR adresinde ikamet eden, SERHAT UNAT (Yönetim Kurulu Başkanı) Temsile Yetkili olarak seçilmiştir.Yetki Şekli: Münferiden Temsile Yetkilidir.
Görev Dağılımı:
İlk 1 Yıl için SERHAT UNAT Yönetim Kurulu Başkanı olarak seçilmiştir.

10. GENEL KURUL
Genel Kurullar, olağan ve olağanüstü toplanırlar. Olağan toplantı her faaliyet dönemi sonundan itibaren üç ay içinde yapılır; olağanüstü toplantı ise, Şirket işlerinin gerektirdiği hallerde ve zamanlarda toplanır. Genel kurul toplantılarında, her pay sahibinin oy hakkı, sahip olduğu payların itibari değerleri toplamının, şirket sermayesinin itibari değerinin toplamına oranlanmasıyla hesaplanır. Pay sahibi genel kurul toplantılarına kendisi katılabileceği gibi pay sahibi olan veya olmayan bir temsilci de yollayabilir. Şirket genel kurul toplantılarında, Türk Ticaret Kanununun 409.maddesinde yazılı hususlar müzakere edilerek gerekli kararlar alınır. Genel kurul toplantıları ve bu toplantılardaki karar nisabı, Türk Ticaret Kanunu hükümlerine tabidir. Genel kurul, şirketin merkez adresinde veya yönetim merkezinin bulunduğu şehrin elverişli bir yerinde toplanır.

11. İLAN
Genel kurulun toplantıya çağrılmasına ilişkin ilanlar da dahil olmak üzere Şirkete ait ilanlar Türkiye Ticaret Sicili Gazetesinde yapılır.
Genel kurulun toplantıya çağrılmasına ilişkin ilanların Türk Ticaret Kanunu'nun 414 üncü maddesi hükmü gereğince ilan ve toplantı günleri hariç olmak üzere en az iki hafta önce yapılması zorunludur.

12. HESAP DÖNEMİ
Şirketin hesap yılı, Ocak ayının 1. gününden başlar ve Aralık ayının 31. günü sona erer. Fakat birinci hesap yılı, Şirketin kesin olarak kurulduğu tarihten itibaren başlar ve o senenin aralık ayının otuz birinci günü sona erer.

13. KARIN TESPİTİ VE DAĞITIMI
Şirketin net dönem kârı, faaliyet dönemi sonunda tespit edilen gelirlerden, yapılmış her çeşit giderlerin çıkarılmasından sonra kalan miktardır. a)Net dönem kârının %5'i, ödenmiş sermayenin %20'sine ulaşıncaya kadar genel kanuni yedek akçeye ayrılır. b)Kalan miktarın %5'i pay sahiplerine kâr payı olarak dağıtılır. Genel kurul, net dönem kârından (a) ve (b) bentlerindeki tutarlar düşüldükten sonra kalan tutarın dağıtılmayarak yedek akçeye ayrılmasına, kısmen veya tamamen dağıtılmasına karar vermeye yetkilidir. Genel kurul, ilgili mevzuat çerçevesinde pay sahiplerine kâr payı avansı dağıtılmasına karar verebilir. Kâr payı avansı tutarının hesaplanmasında ve dağıtımında ilgili mevzuat hükümlerine uyulur.

14. YEDEK AKÇE
Yedek akçelerin ayrılması hususunda Türk Ticaret Kanununun 519 ila 523. maddeleri hükümleri uygulanır.

15. KANUNİ HÜKÜMLER
Bu esas sözleşmede bulunmayan hususlar hakkında Türk Ticaret Kanunu hükümleri uygulanır.

KURUCULAR

Sıra No	Kurucu	Uyruk	İmza
1	SERHAT UNAT	TÜRKİYE	....

(18234413)

T.C. İSTANBUL TİCARET SİCİLİ MÜDÜRLÜĞÜ'NDEN

İlan Sıra No: 94834
Mersis No: 0086158469000001
Ticaret Sicil/Dosya No: 457897-5

Ticaret Unvanı:
AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ

Adres : Tepeüstü Mah. Kafiye Sk. - No: 47 İç Kapı No: 3 Ümraniye / İstanbul
Yukarıda bilgileri verilen şirket ile ilgili olarak aşağıda belirtilen hususlar müdürlüğümüze ibraz edilen belgelere istinaden ve Türk Ticaret Kanunu'na uygun olarak 2.5.2023 tarihinde tescil edildiği ilan olunur.

Tescil Edilen Hususlar: Kuruluş

(Devamı 984.Sayfada)

SAYFA : 984 TÜRKİYE TİCARET SİCİLİ GAZETESİ 2 MAYIS 2023 SAYI: 10822

(Baştarafı 983.Sayfada)

1. Kuruluş
Aşağıdaki adları, soyadları, unvanları, yerleşim yerleri ve uyrukları yazılı kurucular arasında bir Limited Şirket kurulmuş bulunmaktadır.

Sıra No	Kurucu	Adres	Uyruk	Kimlik No
1	ÖMER KORÇAK	İSTANBUL / ZEYTİNBURNU	TÜRKİYE	104******78
2	SALİH BAYAR	İSTANBUL / KADIKÖY	TÜRKİYE	206******68
3	SAMIR MOURAD	ALMANYA	ALMANYA	C*******6

2. ŞİRKETİN UNVANI
Şirketin unvanı **AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ**dir.

3. AMAÇ VE KONU
1-) Uzay, havacılık, haberleşme platformları ve savunma teknolojileriyle ilgili her türlü bilgisayar, elektro teknik, elektronik, elektromekanik ve mekatronik ve bunlara ait yedek parça ve yazılımlarının imalatı, ithalatı, ihracatı, tamiri, bakımı, onarımı, toptan ve perakende ticaretini yapmak,
2-) Uzay, havacılık, haberleşme platformları ve savunma teknolojileriyle ilgili her türlü araştırma ve geliştirme faaliyetinde bulunabilir ve teknoloji geliştirme bölgeleri ve serbest bölgelerde girişimci, yönetici ve danışman şirket olarak yer alabilir,
3-) Uzay, havacılık, haberleşme platformları ve savunma teknolojileriyle ilgili bilişim, yazılım ve uygulama geliştirilmesi ve bunlarla ilgili arge, danışmanlık servis hizmetleri sağlayabilir,
4-) Her türlü bilgisayar sistemleri, yazılım ve donanım destek ve danışmanlık hizmetlerini sağlamak,
5-) Veri hazırlama, onarma, düzenleme işlemleri veya hizmetlerini yapmak, satmak, yada bunları kiraya vermek, internet üzerinde yayınlamak,
6-) Ticari amaçlı bilgi işlem merkezleri kurmak, bu merkezlerin yurtiçi ve yurtdışı data şebekeleri ile irtibatını sağlamak,
7-) Her marka bilgisayar ve alt mikro işlemcileri ile bilgi kaydedici, saklayıcı, yazıcı, çizici, iletici, gösterici, tahvil edici, yorumlayıcı cihaz aksesuar donanım ünite parça ve yedek parçaları üretimi, toptan ve perakende alımı satımı, dağıtımı, ithalat ve ihracatını yapmak, montajı, servisi,
8-) Bilgisayar programlarının ve bilgisayar kontrolü sistemlerin üretimi, toptan ve perakende alımı satımı, dağıtımı, ithalat ve ihracatı, montajı, bakımı için gerekli ham mamul yarı mamul sistemlerinin ünitelerin parçaların ve yedeklerinin ve bunlara uygun her türlü bilgisayarlar ile kontrol edilebilecek ve gelecekte uygulanabilecek teknolojide imal edilmiş mamul ve yarı mamul parça sistem ve üniteler olarak üretimi, toptan ve perakende alımı satımı, dağıtımı, ithalat ve ihracatını yapmak,
9-) Bilgisayar sistemlerinin kurulması ve bunun için her türlü elektrik elektronik donanımın toptan ve perakende alımı satımı, dağıtımı, ithalat ve ihracatı, montajını yapmak,
10-) Kamu, özel ve tüzel kişilere bilgisayar yazılım ve donanım seçiminde fizibilite raporları hazırlamak, mühendislik, müşavirlik, destek ve servis hizmeti almak, vermek,
11-) Her marka bilgisayar, bilgisayar donanım ve sarf malzemelerinin ithalatı, ihracatı ve dahili ticaretini, tüm bilgisayar ve yan donanımları birimleri mikrofilm faksim ile bunlar gibi tüm bilişim sistemleri, endüstriyel büro otomasyon, tüm yazılım ve donanım faaliyetleri servis büro hizmetleri vermek. internet üzerinden her türlü bilgi, ses ve görüntü hizmeti vermek, internet sitesi hazırlamak, hazırlatmak ve internet servis sağlayıcılığı yapmak,
12-) Her türlü güvenlik sistemleri ve yedek parça ile aksesuarlarının üretimi, alımı satımı, dağıtımı, pazarlaması, ithalatını ve ihracatını yapmak ve yaptırmak ve bu sistemlere ait bilgisayar yazılımı, programları hazırlamak, hazır yazılım programlarını ve işletim sistemlerinin alımı, satımı, dağıtımı, pazarlamasını, ithalatını ve ihracatını yapmak ve yaptırmak,
13-) Telefon ve her türlü iletişim araç gereçlerinin ve yedek parçalarının ticaret, ithalat ve ihracatını yapmak,
14-) Her türlü elektrik, elektronik, mekanik ve elektromekanik malzemelerini, bunların aksesuarlarının üretimi, alımı, satımı, dağıtımı, pazarlamasını, montaj ve servis hizmetlerini, ithalatını ve ihracatını yapmak ve yaptırmak,
15-) Elektrik malzemeleri, elektrik kabloları, ışıklı reklam panoları, bina ve tesislerin dış cephe dekoratif aydınlatma üniteleri imalat ve montajı, elektrikli güç kaynakları, trafolar, regülatörler, ses ve musiki aletleri ve teferruatları, elektrikli aletler elektrikli tesisat ile ilgili otomatik kontrol cihazları sayaç, kablo, kablo kanalı, her türlü elektrikli tesisat kalıpları aydınlatma mamulleri, ampul flüoreans lambaları imalatı ithalatı, ihracatı pazarlaması ve ticaretini yapmak ve yaptırmak,
16-) Güvenlik ve korumaya ilişkin tüm elektronik, optik, görsel-işitsel mekanik cihaz, kameralar ve monitörler üretimi, alımı, satımı, dağıtımı, pazarlamasını, ithalatını ve ihracatını yapmak ve yaptırmak,
17-) Her türlü hırdavat malzemelerinin üretimi, alımı, satımı, dağıtımı, pazarlamasını, ithalatını ve ihracatını yapmak ve yaptırmak,
18-) Her türlü temizlik maddeleri ve malzemelerinin, sıvı, toz ve katı deterjan çeşitleri konsantre deterjan, sıvı sabun, sabun, yumuşatıcı ve benzeri, petro-kimya ürünlerin, evsel ve sınai gazların imalatı, toptan ve perakende alımı satımı, dağıtımı, ithalat ve ihracatını yapmak,
19-) Her türlü gıda maddelerinin imalatı, toptan ve perakende alımı, satımı, ithalatı ve ihracatını yapmak,
20-) Her türlü sanayi ve iş makineleri, cihazları ile bunlara ait yedek parçaları, teçhizat ve ekipmanlarının üretimi, toptan ve perakende alımı satımı, dağıtımı, ithalat ve ihracatını yapmak,
21-) İnternet ortamında elektronik ticaret (e-ticaret) veya elektronik tahsilat ve (e-tahsilat) yapılabilmesi için web sayfaları, sanal mağazalar veya online ödeme siteleri açar bunlar için gerekli yazılımları üretmek ve geliştirmek. İnternet ortamında online bilgi alışverişi online ticaret veya online tahsilat yapılabilmesi için yazılım üretmek ve geliştirmek, internet ortamında elektronik ticaret (e-ticaret) ve elektronik tahsilat (e-tahsilat) yapılabilmesi için ar-ge çalışması yapmak, donanım hizmetleri, yer sağlayıcı hizmetleri, internet hizmetleri danışmanlık hizmetleri vermek,
22-) Her türlü motorlu taşıtları ve bunlara ait parça ve aksesuarlarının alım ve satımını, ithalat ve ihracatını yapmak,
23-) Her türlü dayanıklı tüketim malları, buzdolabı, çamaşır makinesi, bulaşık makinesi, televizyon ve benzeri ürünlerin ve sarf malzemelerinin alımını, satımını ve pazarlamasını, ithalatını, ihracatını, toptan ve perakende olarak yapmak, bakım, onarım ve servis hizmetlerini yapmak, yedek parçalarının ticaretini yapmak,
24-) Her türlü giyim ürünlerinin imalatı, toptan ve perakende alımı, satımı, ithalatı ve ihracatını yapmak,
Şirket amacını gerçekleştirmek için her türlü gayrimenkul alabilir satabilir kiralar, kiraya verir bunlar üzerinde ayni ve şahsi her türlü hakları tesis edebilir, ipotek alabilir, ipotek verebilir ve ipotekleri fek edebilir. Şirket gayrimenkulleri üzerinde irtifak, intifa, sükna, gayrimenkul mükellefiyeti, kat irtifakı, kat mülkiyeti tesis edebilir. Her türlü gayrimenkullerle ilgili olarak tapu daireleri nezdinde cins tashihi, ifraz, tevhit, taksim, parselasyon ile ilgili her nevi muamele ve tasarrufları gerçekleştirebilir.

4. ŞİRKETİN MERKEZİ
Şirketin merkezi **İSTANBUL**ili **ÜMRANİYE**ilçesi'dir. Adresi **TEPEÜSTÜ MAH. KAFİYE SK - NO: 47 İÇ KAPI NO: 3 ÜMRANİYE / İSTANBUL**'dir.Adres değişikliğinde yeni adres, ticaret siciline tescil ve Türkiye Ticaret Sicili Gazetesi'nde ilan ettirilir. Tescil ve ilan edilmiş adrese yapılan tebligat şirkete yapılmış sayılır. Tescil ve ilan edilmiş adresinden ayrılmış olmasına rağmen, yeni adresini süresi içinde tescil ettirmemiş şirket için bu durum fesih sebebi sayılır.

5. SÜRE
Şirketin süresi, kuruluşundan itibaren **sınırsız**'dır. Bu süre şirket sözleşmesini değiştirmek suretiyle uzatılıp kısaltılabilir.

6. SERMAYE VE PAY SENETLERİNİN NEV'İ
Şirketin sermayesi, beheri 25,00 Türk Lirası değerinde 4000 paya ayrılmış toplam 100000,00 Türk Lirası değerindedir.
-Beheri 25,00 Türk Lirası değerinde 400 adet paya karşılık gelen 10000,00 Türk Lirası ÖMER KORÇAK tarafından nakdi,
-Beheri 25,00 Türk Lirası değerinde 400 adet paya karşılık gelen 10000,00 Türk Lirası SALİH BAYAR tarafından nakdi,
-Beheri 25,00 Türk Lirası değerinde 3200 adet paya karşılık gelen 80000,00 Türk Lirası SAMIR MOURAD tarafından nakdi,
olarak taahhüt edilmiştir.
Nakden taahhüt edilen payların itibari değerleri, şirketin tescilini izleyen 24 ay içinde ödenecektir.

7. ŞİRKETİN İDARESİ
Şirketin işleri ve işlemleri genel kurul tarafından seçilecek bir veya birkaç müdür tarafından yürütülür.
Aksi Karar Alınıncaya Kadar Almanya Uyruklu C*******6 Kimlik No'lu, Almanya adresinde ikamet eden, SAMIR MOURAD Müdür olarak seçilmiştir.Yetki Şekli: Münferiden Temsile Yetkilidir.

8. TEMSİL
Şirketi müdürler temsil ederler. Şirketi temsil edecek imzalar genel kurul tarafından tespit, tescil ve ilan olunur.
Müdürler, şirkete hizmet akdi ile bağlı olanları sınırlı yetkiye sahip ticari vekil veya diğer tacir yardımcıları olarak atayabilir. Bu şekilde atanacak olanların görev ve yetkileri, hazırlanacak iç yönergede açıkça belirlenir. Bu durumda iç yönergenin tescil ve ilanı zorunludur. İç yönerge ile ticari vekil ve diğer tacir yardımcıları atanamaz. Yetkilendirilen ticari vekil veya diğer tacir yardımcıları da ticaret siciline tescil ve ilan edilir. Bu kişilerin, şirkete ve üçüncü kişilere verecekleri her tür zarardan dolayı müdürler müteselsilen sorumludur.

9. GENEL KURUL
Genel Kurullar, olağan ve olağanüstü toplanırlar. Olağan genel kurul, her yıl hesap döneminin sona ermesinden itibaren 3 ay içinde; olağanüstü genel kurullar ise, Şirket işlerinin gerektirdiği hallerde ve zamanlarda toplanır. Genel kurul toplantılarında, her ortağın oy hakkı, esas sermaye paylarının itibari değerine göre hesaplanır. Genel kurul toplantıları ve bu toplantılardaki karar nisabı, Türk Ticaret Kanunu hükümlerine tabidir.
Genel kurul, şirketin merkez adresinde veya yönetim merkezinin bulunduğu şehrin elverişli bir yerinde toplanır.

10. İLAN
Genel kurulun toplantıya çağrılmasına ilişkin ilanlar da dahil olmak üzere Şirkete ait ilanlar Türkiye Ticaret Sicili Gazetesinde yapılır. Genel kurul toplantılarına ilişkin ilanların toplantı gününden en az on gün önce yapılması zorunludur.

11. HESAP DÖNEMİ
Şirketin hesap yılı, Ocak ayının 1. gününden başlar ve Aralık ayının 31. günü sona erer. Fakat birinci hesap yılı, Şirketin kesin olarak kurulduğu tarihten itibaren başlar ve o senenin aralık ayının otuz birinci günü sona erer.

12. KARIN TESPİTİ VE DAĞITIMI
Şirketin net dönem karı yapılmış her çeşit masrafların çıkarılmasından sonra kalan miktardır. Net dönem kârından her yıl %5 genel kanuni yedek akçe ayrılır, kalan miktar, genel kurul kararı ile pay sahiplerine kar payı olarak dağıtılır. Kar payı, esas sermaye payının itibari değerine, yerine getirilen ek ödeme yükümlülüğünün tutarı eklenmek suretiyle oluşacak toplam miktara oranla hesaplanır.

13. YEDEK AKÇE
Yedek akçelerin ayrılması hususunda Türk Ticaret Kanununun 519 ila 523. maddeleri hükümleri uygulanır.

14. KANUNİ HÜKÜMLER
Bu şirket sözleşmesinde bulunmayan hususlar hakkında Türk Ticaret Kanunu hükümleri uygulanır.
KURUCULAR

Sıra No	Kurucu	Uyruk	İmza
1	ÖMER KORÇAK	TÜRKİYE	....
2	SALİH BAYAR	TÜRKİYE	....
3	SAMIR MOURAD	ALMANYA	....

(18234400)

2 MAY 2023 ISSUE: 10822 — TURKISH TRADE REGISTRY NEWSPAPER — PAGE: 983

(Bagara 982. Page)

1. The company can buy, stabilize, operate, rent, lease all kinds of real estate in order to realize debt, there are deep in kind and chassis rights can be established, mortgages can be given, mortgages can be paid and mortgages can be paid alone. It can make usufruct, usufruct, money, property liability, abandonment to the road with or without payment, rock abandonment, performance, union, and establish condominium servitude and condominium ownership on its real estates. In the title deed offices, all kinds of corrections, execution, unification, taken, parcellation related to the real estates and all kinds of transactions and

2.1.arms oeslocek mav buv.
sell, lease, final lease, or borrow money, collateral or collateral, at home and abroad, depending on whether movable and immovable property. Behind every movable and immovable property owned by the company, as well as fixed facilities and fixtures on every island, both private and legal persons may establish and release collateral.
surety, business choice and mortgage on behalf of the company. Against the receivables, he can take a mortgage or a loan, give a mortgage or pledge as a guarantee of the borrower or in favor of the first debtors, and demand the cancellation of these mortgages.

3. In order for the company to be coma, the company performs public legislation, pertinent kold tim enterprise and ceremonies, such as resignation, natume, potcnt, in beration, license.
know-how marks, termination,
document, tool feka, property rights and eli han To assign or transfer to the trump, to engage in all kinds of ilms. In addition, it can rent or identify its rights. Transfer and sale of the acquired real estates, they can be used or leased by third parties wholly or wholly.
Domestic and foreign credits required for the company's tampering facilities are obtained from domestic and foreign corporations and enterprises through domestic and foreign financial institutions.

6. The company may purchase all kinds of machinery, installations, transportation vehicles, chur tools and devices and other movable properties and rights necessary for the storage, transportation, right of the goods related to the subject.
or sell, import and export, use, rent and lease

7. In order to realize the objectives of the company, the company can make financial and legal transactions and savings, participate in official and private tenders, agency, marnet, brokerage, dealership, distributor can buy and give. Saket may establish companies related to the subject, make partnerships, participate in existing and future businesses and companies. The company can transfer the companies and commercial enterprises that are easily engaged in, not to make intermediaries, and it has been established and will be established.
Companies can buy or sell stocks and shares, can buy and sell securities such as stocks, bills, usufruct

9. Provided that it is beneficial for the purpose of the company and not intermediary, shares and coupons that will be issued and will be issued by individuals based on civil and public law, depending on each and every size of stock, It can take over, show

10. Can buy raw materials, by-products, by-products and finished products related to the subject, collateral, accept, supply with import, export and other securities, find it or completely operate it.
i may cooperate or partner with them. 12. Domestic and foreign companies operating and You can get distributor, dealership or agency from abroad, give or give. 13. The company has official and private information about the cola on the side of the company, both at home and abroad.

14. The company can carry out export and import transactions by fulfilling certain issues regarding all the cola on the side.

15. He/she can do and do work related to the subject.

16. To be related to the subject, ere cold storage can be established, operated, rented.

17. Participation in all kinds of four, exhibitions and shares related to the purpose and subject odche

4. HEADQUARTERS OF THE COMPANY

The head office of the company is TRABZON ORTAHISAR. Its address is in ÇARŞI MAH. EMIN ALEMDAR SK NO: 1 IC DOOR NO: 5 ORTAHISAR/ TRABZON. In case of a change of address, the new badge is registered with the trade registry and registered in the Turkish Trade Registry Gazette. Notification to the registered and announced address is sent to the company. For the company that did not register its new address in due time, although I did not leave its registered and announced address, this situation will be terminated due to termination

5. SÜRE

The stress of the company sınarvarder from the beginning. Changing the main stringing of each string

6. CAPITAL

The capital of the company is a total of 1000 shares, each worth 500.00 Turkish Liras, in kind. It is worth 500000.00 Tark Liras All of this share has been registered 500000,00 Tiak Lira, which came to 1000 shares worth 500,00 Turkish Lira each, was fixed in cash from SERHAT UNAT. The remainder of the shares committed in cash will be paid within 24 months following the registration of the company according to the profits received by the management.

7. BOARD OF DIRECTORS AND TERM

The business and administration of the company is carried out by a board of directors with at least 1 person to be elected by the General Assembly within the framework of Türk Ticaret Profit. The stress of the Board of Directors is 3 years. As a member of the First Board of Directors, the following for 3 years

SERHAT UNAT, residing at Turkey Uynklis 351*****8 ID No, DİYARBAKIR / YENİŞEHİR, has been elected as a Member of the Board of Directors.

8. ADMINISTRATION AND REPRESENTATION OF THE COMPANY

The Management of Şuket and the representation of Dışya Kery belong to the Management Karma. In order for the documents to be sinnedy and the agreements to be canceled to be valid, these must be submitted to the company

The signature of the person or persons authorized to contact the samall and the company is tapmast pad. The board of directors may delegate its authority to one or more members of the board of directors or to third parties as a manager. Provided that at least one member of the board of directors has the authority to represent.

The board of directors is authorized to delegate the management to one or more members of the board of directors or to a hungry person, in accordance with an internal directive to be drawn up. This internal directive determines who manages the company, defines the necessary gaevs, places gostars, and especially determines who is responsible for giving feedback and information to whom. The management law informs the shareholders of the claim and the creditors who convincingly demonstrate their interests worth protecting, about this internal directive, basedly. The Board of Directors may appoint members of the board of directors who are not authorized to establish a foundation, or as commercial attorneys or other merchant assistants who have the authority to pay tribute to the company. In this way, only the action and authority of those who are are clearly determined by the internal directive to be intensified. In this case, the registration and application of the internal directive is zonal. Commercial attorneys and other merchant assistants are appointed by the internal directive. The board of directors is consolingly responsible for any time delays that these persons may give to the company and to the persons.

9. AUTHORIZED REPRESENTATIVES, REPRESENTATION METHOD AND DISTRIBUTION OF DUTY

For Persons Authorized to Represent and
Representation Sek 3 Yi, SERHAT UNAT (Chairman of the Board), residing at the address of DİYARBAKIR / YENİŞEHİR, with Turkey Nationality ID 3518, as Authorized Representation deleted it. Type of Authority: Individually Authorized.

Görev Dağılımı:
Ik I YSERHAT UNAT has been elected as the Chairman of the Board of Directors.

10. GENERAL ASSEMBLY

General Assemblies, olşan and olağanüstü totals. Ordinary meetings are held within the sedan strong month of each activity period, while the ordinary meetings are toplone when the business of the Company is necessary and at times, the general shareholders' toplastlants, collect the nominal values of the shares in which each shareholder has the right to vote, by proportioning the ibur value of the company's capital to the total. The condis may remain, as well as send a representative who is a shareholder or not. In the company's board of directors meetings, partial decisions are taken by making false claims in article 409 of the Turkish Commercial Code. General assembly meetings and the quorum of these meetings are subject to Tük Turnet Ka formations. The general assembly convenes at the headquarters of the authority or at a convenient place in the city where the management marker is located.

Announcements of the Company, including the announcements regarding the invitation to the meeting from the general meeting.

12. ACCOUNTING PERIOD

Saket's account starts from the 1st day of January and ends on the 31st day of December.

13. DETECTION AND DISTRIBUTION OF PROFIT

The company's record-breaking, operating-period sounds are in bold amounts after deducting any multiple expenses incurred from the revenues identified. 565 of the remaining amount is not in the same way as the general public reserve fund until the period is deducted to 20% of the ordinary capital. The anima is authorized to decide whether to disperse or not to disperse completely. The general public may decide to distribute dividends to the shareholders within the framework of the relevant legislation. To the relevant legislation experts in the calculation and distribution of the profit share advances.

14. SPARE ACCESSORIES

Yok skderin man han in Turkish Commercial Code, articles 519 to 523. hori g

15. LEGAL PROVISIONS

Turkish Commercial Code provisions about the husker not included in the baus streyme

FOUNDERS

[illegible]	[illegible]	[illegible]	[illegible]
	[illegible]	[illegible]	

(18234413)

T.R. FROM ISTANBUL TRADE REGISTRY DIRECTORATE

Ian Greenhouse No: 9434
Mersis No: 0086158400000000
Trade Registry File No: 457897-5

Trade Name: AS
COMSAT COMMUNICATION PLATFORMS AND SATELLITE COMMUNICATION SYSTEMS SANAYI VE TICARET LIMITED SIRKETI

Address: Tepeüstü Mah. Rhyme Sk. No: 47 Inner Cap No: 3 Ümraniye/İstanbul The matters stated in the name of the above-given company, based on the documents submitted to the mile and in accordance with the Turkish Commercial Code, were established on 2.5.2023.

Registered Matters Kun

(Continued on 984 Pages)

[11:30, 3.5.2023] Salih Bayar Istanbul: "Mr. Mourad has to go to the notary with this newspaper within 15 days and take out the signature circular and the general accountant's power of attorney and send the scanned photo of it to us."

(Bagara 982. Seite)

1. Das Unternehmen kann alle Arten von Immobilien kaufen, stabilisieren, betreiben, mieten, verleasen, um Schulden zu realisieren, dass sind tief in der Sache und es können Grundrechte begründet, Hypotheken vergeben, Hypotheken bezahlt

mit oder ohne Zahlung, Steinaufgabe, Leistung, Vereinigung machen und Eigentumswohnungen und

Vereinigungen, Teilen, Parzellierungen im Zusammenhang mit den Immobilien und allen Arten von Transaktionen und

2. Lanne geloost kann im In- und Ausland, je nachdem ob es sich um bewegliche und unbewegliche Sachen handelt, Geld, Sicherheiten oder Sicherheiten kaufen, verkaufen, leasen, endverpachten oder leihen. Hinter jedem beweglichen und unbeweglichen Eigentum des Unternehmens sowie festen Einrichtungen und Einrichtungen auf jeder Insel können sowohl private als auch juristische Personen Sicherheiten, Bürgschaften, Geschäftsentscheidungen

und Hypotheken im Namen des Unternehmens begründen und freigeben. Er kann auf die Forderungen eine Hypothek oder ein Darlehen aufnehmen, eine Hypothek oder ein Pfandrecht als Bürgschaft des Darlehensnehmers oder zugunsten der Erstschuldner erteilen und die Aufhebung dieser Hypotheken verlangen.

3. Damit das Unternehmen im Koma ist, führt das Unternehmen öffentliche Rechtsvorschriften, einschlägige kold tim Unternehmen und Zeremonien durch, wie z rechte und ell han an den trump abzutreten oder zu übertragen, sich an filmen aller art zu beteiligen, darüber hinaus kann er seine rechte vermieten oder identifizieren. Übertragung und Verkauf der erworbenen Immobilien, sie können ganz oder vollständig von Dritten genutzt oder verpachtet werden. In- und ausländische Kredite, die für die Manipulationseinrichtungen des Unternehmens erforderlich sind, werden von in- und ausländischen Körperschaften und Unternehmen über in- und ausländische Finanzinstitute erhalten.

6. Die Gesellschaft kann alle Arten von Maschinen, Anlagen, Transportfahrzeugen, Werkzeugen und Werkzeugen und anderen

oder verkaufen, importieren und exportieren, nutzen, vermieten und verleasen.

7. Zur Verwirklichung des Gesellschaftszwecks kann die Gesellschaft Geld- und Rechtsgeschäfte tätigen und Einsparungen tätigen, an öffentlichen und privaten Ausschreibungen teilnehmen, vermitteln, mamat, Makler, Händler, Händler können kaufen und geben. Selbst kann mit dem

Thema verbundene Unternehmen gründen, Partnerschaften eingehen, sich an bestehenden und zukünftigen Geschäften und Unternehmen beteiligen. Das Unternehmen kann die Unternehmen und Handelsunternehmen übertragen, die leicht zu engagieren sind, keine Zwischenhändler zu machen, und es wurde gegründet und wird gegründet.

Unternehmen können Aktien und Aktien kaufen oder verkaufen, Wertpapiere wie Aktien, Wechsel, Nießbrauchsanteile

9. Sofern es dem Gesellschaftszweck dient und nicht vermittelt, und Coupons kaufen und verkaufen, die von natürlichen Personen auf der Grundlage des Zivil- und öffentlichen Rechts ausgegeben werden und ausgegeben werden, je nach Aktiengröße. Sie kann übernehmen, Sicherheiten

10. Rohstoffe, Nebenprodukte, Nebenprodukte und Fertigprodukte zum Thema kaufen können, stellen, entgegennehmen, mit Import-, Export- und anderen Sicherheiten versorgen, finden oder komplett betreiben.

kooperieren oder mit ihnen zusammenarbeiten.

12. In- und ausländische Unternehmen, die tätig sind und

hat offizielle und private Informationen über die Cola auf der Seite des Unternehmens, sowohl im In- als auch im Ausland

14. Das Unternehmen kann Export- und Importgeschäfte durchführen, indem es bestimmte Punkte in Bezug auf alle Colas auf der Seite erfüllt.

15. Er/sie kann fachbezogene Arbeiten ausführen und ausführen.

16. Um auf das Thema einzugehen, wo Kühllager errichtet, betrieben, gemietet,

17. Teilnahme an allen vier, Ausstellungen und Beteiligungen im Zusammenhang mit Zweck und Thema od.chr

4. HAUPTSITZ DES UNTERNEHMENS

Der Hauptsitz des Unternehmens ist TRABZON ORTAHISARpes. Seine Adresse ist in ÇARSI MAH EMIN ALEMDAR SK NO: 1 IC DOOR NO: 5 ORTAHISAR/ TRABZON Im Falle einer Adressänderung wird der neue Ausweis beim Handelsregister und im Türkischen Handelsregisterblatt eingetragen. Die Benachrichtigung an die registrierte und bekannt gegebene Adresse wird an das Unternehmen gesendet. Für das Unternehmen, das seine neue Anschrift nicht fristgerecht angemeldet hat, obwohl ich seine eingetragene und bekannt gegebene Anschrift nicht hinterlassen habe, wird diese Situation durch Kündigung beendet.

5 WUND

Der Stress des Unternehmens anarvander von Anfang an. Ändern der Hauptberatung jeder Seite

6. HAUPTSTADT

Es ist 500000,00 Tark Lira wert. Alle diese Anteile wurden registriert. 500000,00 Tiak Lira, was 1000 Aktien im Wert von je 500,00 Türkischen Lira entsprach, wurde in bar von SERHAT UNAT fixiert. Der Rest der in bar zugesagten Aktien wird innerhalb von 24 Monaten nach Eintragung der Gesellschaft entsprechend den von der Geschäftsführung erzielten Gewinnen ausbezahlt.

7. VERWALTUNGSRAT UND ZEITRAUM

Die Geschäfte und die Verwaltung der Gesellschaft werden von einem Vorstand mit mindestens 1 von der Generalversammlung zu wählenden Person im Rahmen von Türk Ticaret Profit geführt. Die Amtszeit des Vorstandes beträgt 3 Jahre. Als Mitglied des Ersten Vorstandes im Anschluss für 3 Jahre

SERHAT UNAT, wohnhaft in der Türkei Uynkls 351*****8 ID Ndo, DİYARBAKIR / YENİŞEHIR, wurde zum Mitglied des Board of Directors gewählt.

8. VERWALTUNG UND VERTRETUNG DER GESELLSCHAFT

Das Management von Suket und die Vertretung von Dayya Kary gehören zum Management Kanna

Die Unterschrift der Person oder Personen, die befugt sind, das Samalt und das Unternehmen zu kontaktieren, ist tepmast pad. Der Verwaltungsrat kann seine Befugnisse an ein oder mehrere Verwaltungsratsmitglieder oder an Dritte als Geschäftsführer delegieren. Sofern mindestens ein

Der Vorstand ist ermächtigt, die Geschäftsführung nach Maßgabe einer zu erarbeitenden internen Weisung einem oder mehreren Vorstandsmitgliedern oder einer hungernden Person zu übertragen. Diese interne Richtlinie bestimmt, wer das Unternehmen führt, definiert die notwendigen Geevs, setzt Gösbars und bestimmt insbesondere, wer für die Rückmeldung und Information an wen verantwortlich ist. Das Verwaltungsgesetz informiert die Anteilseigner der Forderung und die Gläubiger, die ihre schutzwürdigen Interessen überzeugend darlegen, über diese interne Weisung vorweggenommen. Der Vorstand kann Vorstandsmitglieder, die nicht zur Errichtung einer Stiftung befugt sind, oder zu Handelsanwälten oder sonstigen kaufmännischen Hilfskräften ernennen, die befugt sind, der Gesellschaft Tribut zu zahlen. Auf diese Weise werden nur das Handeln und die Autorität derjenigen, die es sind, durch die zu intensivierende interne Direktive eindeutig bestimmt. In diesem Fall erfolgt die Registrierung und Anwendung der internen Richtlinie zonal. Handelsanwälte und andere Kaufmannsgehilfen werden durch die Hausordnung bestellt. Für etwaige Zeitverzögerungen, die diese Personen der Gesellschaft und den Personen zufügen, trägt der Vorstand die gütliche Verantwortung.

9. AUTORISIERTE VERTRETER, VERTRETUNGSMETHODE UND AUFGABENVERTEILUNG

Für vertretungsberechtigte und vertretungsberechtigte

Personen Sek 3 Yi., SERHAT UNAT (Vorsitzender des Vorstands), wohnhaft an der Adresse DİYARBAKIR / YENİŞEHİR, mit türkischer Nationalität ID 3518 als bevollmächtigte Vertretung löschte es. Art der Behörde: Einzelautorisiert.

Görev Dağılımı

Ik I YSERHAT UNAT wurde zum Vorsitzenden des Board of Directors gewählt.

10. GENERALVERSAMMLUNG

Generalversammlungen, Oljan- und Clajestet-Summen. Ordentliche Versammlungen werden innerhalb des sechsstarken Monats jeder Aktivitätsperiode abgehalten, während die ordentlichen Versammlungen toplone stattfinden, wenn die Geschäfte der Gesellschaft notwendig sind, und zuweilen sammeln die Hauptaktionäre die Nennwerte der Anteile, an denen jeder beteiligt ist. Der Aktionär hat das Stimmrecht, indem er den Nur-Wert des Gesellschaftskapitals an die Gesamtsumme anpasst. Die Bedingungen können bestehen bleiben, sowie einen Vertreter entsenden, der Aktionär ist oder nicht. In den Vorstandssitzungen des Unternehmens werden Teilentscheidungen getroffen, indem falsche Behauptungen gemäß Artikel 409 des türkischen Handelsgesetzbuchs aufgestellt werden. Generalversammlungssitzungen und das Quorum dieser Sitzungen unterliegen den Tak Tumet Ka-Formationen. Die Generalversammlung tritt am Sitz der Behörde oder an

11. ANKÜNDIGUNG

Bekanntmachungen der Gesellschaft, einschließlich der Bekanntmachungen über die Einladung zur Versammlung durch die Hauptversammlung.

12. RECHNUNGSZEITRAUM

Das Konto von Saket beginnt am 1. Januar und endet am 31. Dezember.

13. ERMITTLUNG UND VERTEILUNG DER GEWINNE

Die rekordverdächtigen Betriebsgebräuche des Unternehmens sind in fetten Zahlen dargestellt, nachdem alle entstandenen Mehrfachausgaben von den identifizierten Einnahmen abgezogen wurden. 565 des verbleibenden Betrags stellen bis zum Zeitpunkt des Abzugs auf 20 % des ordentlichen Kapitals nicht in gleicher Weise den allgemeinen öffentlichen Reservefonds dar. Die Anima ist befugt, zu entscheiden, ob sie vollständig aufgelöst wird oder nicht. Die breite Öffentlichkeit kann beschließen, im Rahmen der einschlägigen Gesetzgebung Dividenden an die Aktionäre auszuschütten. An die zuständigen Rechtsexperten bei der Berechnung und Verteilung der Gewinnanteilsvorschüsse.

14. ERSATZZUBEHÖR

Eigelbskelett man han im türkischen Handelsgesetzbuch, Artikel 519 bis 523. hon g

15. RECHTLICHE BESTIMMUNGEN

Bestimmungen des türkischen Handelsgesetzbuchs über den Schüler, die nicht in der Beus Streyne enthalten sind.

GRÜNDER

[illegible]	[illegible]	[illegible]	[illegible]
	SERHAT UNAT	[illegible]	

(18234413)

T.R. VON DER DIREKTION DES HANDELSREGISTERS ISTANBUL

Mersis-Nr.: 00861584 0000000

Handelsregisternummer: 457897-5

Handelsname

AS COMSAT COMMUNICATION PLATFORMS AND SATELLITE COMMUNICATION SYSTEMS SANAYİ VE TİCARET LİMİTED ŞİRKETİ

Adresse: Tepebaşı Mah. Reim Sk. Nr.: 47 Inner Cap Nr.: 3 Ümraniye/Istanbul Die im Namen der oben genannten Gesellschaft genannten Angelegenheiten wurden auf Grundlage der bei der Meile eingereichten Unterlagen und in Übereinstimmung mit dem türkischen Handelsgesetzbuch am 2.5.2023 gegründet.

Registrierte Angelegenheiten: Kun

(Fortsetzung auf 984 Seite)

4.4.12 İ S T A N B U L CHAMBER OF COMMERCE OPERATING CERTIFICATE (COPY OF REGISTRY REGISTRATION)

İ S T A N B U L

T İ C A R E T   O D A S I

FAALİYET BELGESİ

(SİCİL KAYIT SURETİ)

TARİH: 03/05/2023

SİCİL NO : 457897-5 **MERSİS:** 0086-1584-6900-0001

FİRMA : AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ

ADRES : TEPEÜSTÜ MAH. KAFİYE SK. - NO: 47 İÇ KAPI NO: 3 ÜMRANİYE

MEŞGALE:

1-) Uzay, havacılık, haberleşme platformları ve savunma teknolojileriyle ilgili her türlü bilgisayar, elektro teknik, elektronik, elektromekanik ve mekatronik ve bunlara ait yedek parça ve yazılımlarının imalatı, ithalatı, ihracatı, tamiri, bakımı, onarımı, toptan ve perakende ticaretini yapmak, 2-) Uzay, havacılık, haberleşme platformları ve savunma teknolojileriyle ilgili her türlü araştırma ve geliştirme faaliyetinde bulunabilir ve teknoloji geliştirme bölgeleri ve serbest bölgelerde girişimci, yönetici ve danışman şirket olarak yer alabilir, 3-) Uzay, havacılık, haberleşme platformları ve savunma teknolojileriyle ilgili bilişim, yazılım ve uygulama geliştirilmesi ve bunlarla ilgili arge, danışmanlık servis hizmetleri sağlayabilir, 4-) Her türlü bilgisayar sistemleri, yazılım ve donanım destek ve danışmanlık hizmetlerini sağlamak, 5-) Veri hazırlama, onarma, düzenleme işlemleri veya hizmetlerini yapmak, satmak, yada bunları kiraya vermek, internet üzerinde yayınlamak, 6-) Ticari amaçlı bilgi işlem merkezleri kurmak, bu merkezlerin yurtiçi ve yurtdışı data şebekeleri ile irtibatını sağlamak, 7-) Her marka bilgisayar ve alt mikro işlemcileri ile bilgi kaydedici, saklayıcı, yazıcı, çizici, iletici, gösterici, tahvil edici, yorumlayıcı cihaz aksesuar donanım ünite parça ve yedek parçaları üretimi, toptan ve perakende alımı satımı, dağıtımı, ithalat ve ihracatını yapmak, montajı, servisi, 8-) Bilgisayar programlarının ve bilgisayar kontrolü sistemlerin üretimi, toptan ve perakende alımı satımı, dağıtımı, ithalat ve ihracatı, montajı, bakımı için gerekli ham mamul yarı mamul sistemlerinin ünitelerin parçaların ve yedeklerinin ve bunlara uygun her türlü bilgisayarlar ile kontrol edilebilecek ve gelecekte uygulanabilecek teknolojide imal edilmiş mamul ve yarı mamul parça sistem ve üniteler olarak üretimi, toptan ve perakende alımı satımı, dağıtımı, ithalat ve ihracatını yapmak, 9-) Bilgisayar sistemlerinin kurulması ve bunun için her türlü elektrik elektronik donanımın toptan ve perakende alımı satımı, dağıtımı, ithalat ve ihracatı, montajını yapmak, 10-) Kamu, özel ve tüzel kişilere bilgisayar yazılım ve donanım seçiminde fizibilite raporları hazırlamak, mühendislik, müşavirlik, destek ve servis hizmeti almak,

https://eportal.ito.org.tr/belge/ZA7HdFwrsHir7QHmkK-PqA2

vermek, 11-) Her marka bilgisayar, bilgisayar donanım ve sarf malzemelerinin ithalatı, ihracatı ve dahili ticaretini, tüm bilgisayar ve yan donanımları birimleri mikrofilm faksim ile bunlar gibi tüm bilişim sistemleri ,endüstriyel büro otomasyon, tüm yazılım ve donanım faaliyetleri servis büro hizmetleri vermek. internet üzerinden her türlü bilgi, ses ve görüntü hizmeti vermek, internet sitesi hazırlamak, hazırlatmak ve internet servis sağlayıcılığı yapmak, 12-) Her türlü güvenlik sistemleri ve yedek parça ile aksesuarlarının üretimi, alımı satımı, dağıtımı, pazarlamasını, ithalatını ve ihracatını yapmak ve yaptırmak ve bu sistemlere ait bilgisayar yazılımı, programları hazırlamak, hazır yazılım programlarını ve işletim sistemlerinin alımı, satımı, dağıtımı, pazarlamasını, ithalatını ve ihracatını yapmak ve yaptırmak, 13-) Telefon ve her türlü iletişim araç gereçlerinin ve yedek parçalarının ticaret, ithalat ve ihracatını yapmak, 14-) Her türlü elektrik, elektronik, mekanik ve elektromekanik malzemelerini, bunların aksesuarlarının üretimi, alımı, satımı, dağıtımı, pazarlamasını, montaj ve servis hizmetlerini, ithalatını ve ihracatını yapmak ve yaptırmak, 15-) Elektrik malzemeleri, elektrik kabloları, ışıklı reklam panoları, bina ve tesislerin dış cephe dekoratif aydınlatma üniteleri imalat ve montajı, elektrikli güç kaynakları, trafolar, regülatörler, ses ve musiki aletleri ve teferruatları, elektrikli aletler elektrikli tesisat ile ilgili otomatik kontrol cihazları sayaç, kablo, kablo kanalı, her türlü elektrikli tesisat kalıpları aydınlatma mamulleri, ampul flüoreans lambaları imalatı ithalatı, ihracatı pazarlaması ve ticaretini yapmak ve yaptırmak, 16-) Güvenlik ve korumaya ilişkin tüm elektronik, optik, görsel-işitsel mekanik cihaz, kameralar ve monitörler üretimi, alımı, satımı, dağıtımı, pazarlamasını, ithalatını ve ihracatını yapmak ve yaptırmak, 17-) Her türlü hırdavat malzemelerinin üretimi, alımı, satımı, dağıtımı, pazarlamasını, ithalatını ve ihracatını yapmak ve yaptırmak, 18-) Her türlü temizlik maddeleri ve malzemelerinin, sıvı, toz ve katı deterjan çeşitleri konsantre deterjan, sıvı sabun, sabun, yumuşatıcı ve benzeri, petro-kimya ürünlerin, evsel ve sınai gazların imalatı, toptan ve perakende alımı satımı, dağıtımı, ithalat ve ihracatını yapmak, 19-) Her türlü gıda maddelerinin imalatı, toptan ve perakende alımı, satımı, ithalatı ve ihracatını yapmak, 20-) Her türlü sanayi ve iş makineleri, cihazları ile bunlara ait yedek parçaları, teçhizat ve ekipmanlarının üretimi, toptan ve perakende alımı satımı, dağıtımı, ithalat ve ihracatını yapmak, 21-) İnternet ortamında elektronik ticaret (e-ticaret) veya elektronik tahsilat ve (e-tahsilat) yapılabilmesi için web sayfaları, sanal mağazalar veya online ödeme siteleri açar bunlar için gerekli yazılımları üretmek ve geliştirmek. İnternet ortamında online bilgi alışverişi online ticaret ...ve anasözleşmesinde yazılı olan diğer işler.

DERECESİ	:	Y
SERMAYE	:	*********100.000,00 TL' dir.
İŞE BAŞLAMA TARİHİ	:	2/5/2023
MESLEK	:	54 MOTORLU ARAÇLAR TAMİR, BAKIM VE İMALATI

https://eportal.ito.org.tr/belge/ZA7HdFwrsHir7QHmkK-PqA2

GR.

NACE KODU	**:**
30.30.06	Uzay aracı, uzay aracı fırlatma araçları ve mekanizmaları ile uydular, uzay roketleri, yörünge istasyonları ve uzay mekiklerinin imalatı NACE Rev.02

BU BELGE ODA KAYDI FAAL OLAN YUKARIDA ÜNVANI YAZILI ÜYENİN TALEBİ ÜZERİNE VERİLMİŞTİR. ÜZERİNDE TAHRİFAT YAPILAN BELGELER GEÇERSİZDİR.

Bu belge 60 (altmış) gün süreyle geçerlidir.

5070 Sayılı Elektronik İmza Kanununa göre Güvenli Elektronik İmza ile imzalanmıştır. Elektronik olarak imzalanan belgeye asağıdaki URL adresinden ulaşılabilmektedir.

MEHMET SADIK TEMEL

e-imza
Genel Sekreter y.
ÜYELİK HİZMETLERİ MÜDÜRLÜĞÜ

https://eportal.ito.org.tr/belge/ZA7HdFwrsHir7QHmkK-PqA2

4.4.13 Tax Sheet (VERGI LEVHASI)

VERGİ LEVHASI

Gelir İdaresi Başkanlığı

MÜKELLEFİN

ADI SOYADI		VERGİ DAİRESİ	ALEMDAĞ
TİCARET ÜNVANI	AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED	VERGİ KİMLİK NO	0 8 6 1 5 8 4 6 9 0
İŞ YERİ ADRESİ	TEPEÜSTÜ MAH. KAFİYE SK. - NO: 47 İÇ KAPI NO: 3 ÜMRANİYE/ İSTANBUL	TC KİMLİK NO	
VERGİ TÜRÜ	KURUMLAR VERGİSİ	İŞE BAŞLAMA TARİHİ	02.05.2023
ANA FAALİYET KODU VE ADI	303006-UZAY ARACI, UZAY ARACI FIRLATMA ARAÇLARI VE MEKANİZMALARI İLE UYDULAR, UZAY ROKETLERİ, YÖRÜNGE İSTASYONLARI VE UZAY MEKİKLERİNİN İMALATI		

TAKVİM	BEYAN OLUNAN MATRAH	TAHAKKUK EDEN VERGİ	ONAY KODU
2022	Yeni işe başlama	02.05.2023	5 R D J 3 T K J T V R

https://intvd.gib.gov.tr adresinden güncelliğini ve doğruluğunu sorgulayabilirsiniz.

4.4.14 Foundation Costs

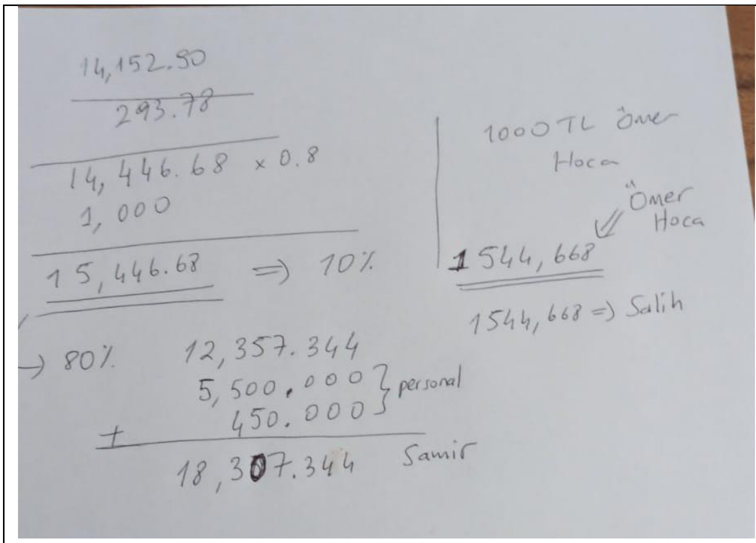
[11:49, 3.5.2023] Salih Bayar Istanbul: So far, I make the following payments:

[11:50, 3.5.2023] Salih Bayar Istanbul: 954.74TL for notary

[11:50, 3.5.2023] Salih Bayar Istanbul: 409.79TL for notary

[11:51, 3.5.2023] Salih Bayar Istanbul: 7,000TL (Seven thousand TL) money transfer to Mesut Yıldız (Financial Advisor) for foundation

[11:52, 3.5.2023] Salih Bayar Istanbul: 8,364.53TL totally



5,500 TL for private account of Samir Mourad at Kuveyt Turk bank.

(To be able to open a company banking account at Islamic Bank Kuveyt Türk, the Manager Samir Mourad had to open fist a private account)

12.5.23 Total Foundation Costs: 15,446 TL (<u>**718 EUR**</u>)

Cost paid according to shares:

Samir 80%	12,557 TL
Salih 10%	1,545 TL
Ömer hoca 10%	1,545 TL

80% 12357 TL

10%

4.5 Banking account at Kuveyt Türk Bank

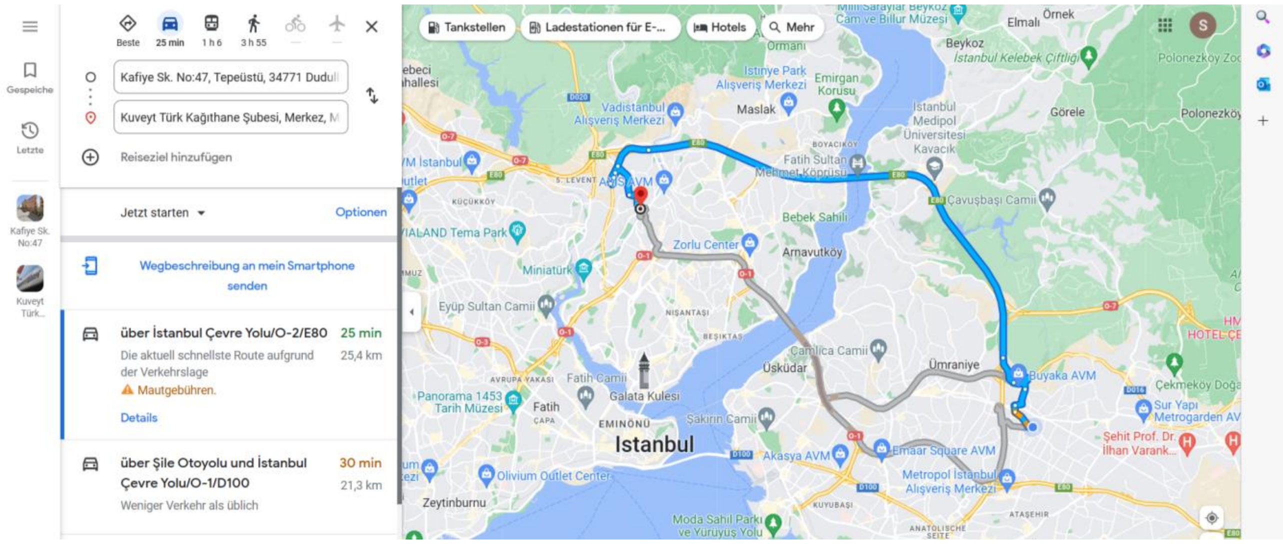
"You can open an account in the name of the company from the bank together with the signature circular, trade registry newspaper, activity certificate, tax plate, passport translation and stamp."

Opened at Kuveyt Türk Kağıthane Şubesi

[20:45, 11.5.2023] Salih Bayar Istanbul: kağıthane ilçesinde Gültepe şubesi

[20:46, 11.5.2023] Salih Bayar Istanbul: Gültepe branch in Kağıthane town 😀

[20:47, 11.5.2023] Salih Bayar Istanbul: Ortabayır, Talatpaşa Cd. No: 70, 34410 Kâğıthane/İstanbul



4.5.1 Private Account of Manager Samir Mourad at Kuveyt Turk Bank Istanbul

Kuveyt Türk Bank

Owner: SAMIR MOURAD

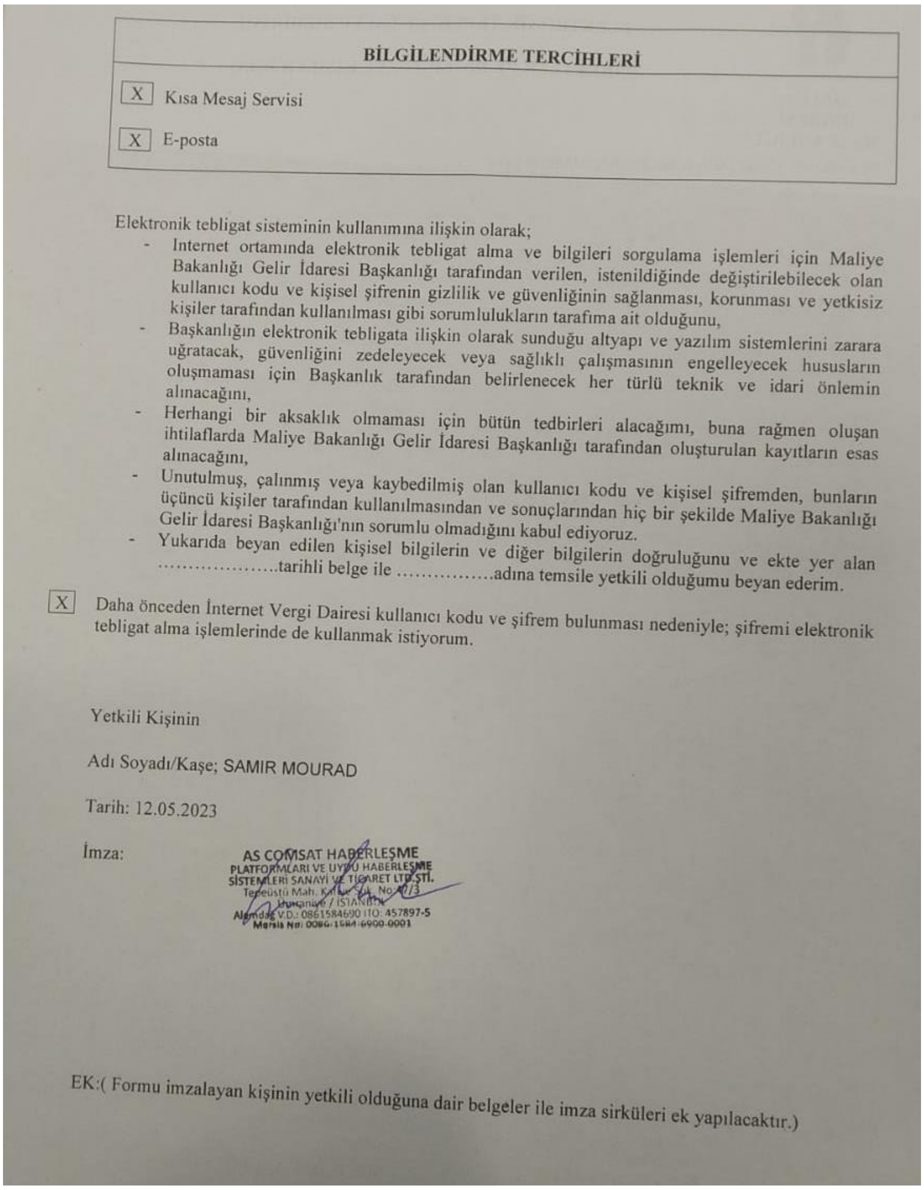
IBAN TR220020500009855530800102 (EUR Account)

Private account

[14:18, 12.5.2023] Samir Mourad: I had to open a private Account before are able to open as comsat account

4.5.2 AS COMSAT Account at Kuveyt Turk Bank

User name online banking: samır (Not samir)



BİLGİLENDİRME TERCİHLERİ

[X] Kısa Mesaj Servisi

[X] E-posta

Elektronik tebligat sisteminin kullanımına ilişkin olarak;

- Internet ortamında elektronik tebligat alma ve bilgileri sorgulama işlemleri için Maliye Bakanlığı Gelir İdaresi Başkanlığı tarafından verilen, istenildiğinde değiştirilebilecek olan kullanıcı kodu ve kişisel şifrenin gizlilik ve güvenliğinin sağlanması, korunması ve yetkisiz kişiler tarafından kullanılması gibi sorumlulukların tarafıma ait olduğunu,
- Başkanlığın elektronik tebligata ilişkin olarak sunduğu altyapı ve yazılım sistemlerini zarara uğratacak, güvenliğini zedeleyecek veya sağlıklı çalışmasının engelleyecek hususların oluşmaması için Başkanlık tarafından belirlenecek her türlü teknik ve idari önlemin alınacağını,
- Herhangi bir aksaklık olmaması için bütün tedbirleri alacağımı, buna rağmen oluşan ihtilaflarda Maliye Bakanlığı Gelir İdaresi Başkanlığı tarafından oluşturulan kayıtların esas alınacağını,
- Unutulmuş, çalınmış veya kaybedilmiş olan kullanıcı kodu ve kişisel şifremden, bunların üçüncü kişiler tarafından kullanılmasından ve sonuçlarından hiç bir şekilde Maliye Bakanlığı Gelir İdaresi Başkanlığı'nın sorumlu olmadığını kabul ediyoruz.
- Yukarıda beyan edilen kişisel bilgilerin ve diğer bilgilerin doğruluğunu ve ekte yer alan ....tarihli belge ile ....adına temsile yetkili olduğumu beyan ederim.

[X] Daha önceden İnternet Vergi Dairesi kullanıcı kodu ve şifrem bulunması nedeniyle; şifremi elektronik tebligat alma işlemlerinde de kullanmak istiyorum.

Yetkili Kişinin

Adı Soyadı/Kaşe; SAMIR MOURAD

Tarih: 12.05.2023

İmza:

AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LTD.ŞTİ.

EK:(Formu imzalayan kişinin yetkili olduğuna dair belgeler ile imza sirküleri ek yapılacaktır.)

EK: 1

GELİR
İDARESİ
BAŞKANLIĞI

ELEKTRONİK TEBLİGAT TALEP BİLDİRİMİ

(ŞİRKETLER VE DİĞER TÜZEL KİŞİLER İÇİN)

ALEMDAĞ Vergi Dairesi Başkanlığı/Müdürlüğü'ne

Kurum Kanuni Temsilcisinin/Vekilinin			
Adı, Soyadı	SAMIR MOURAD	Adı, Soyadı	
Vergi Kimlik Numarası/ T.C. Kimlik Numarası	6232042167	Vergi Kimlik Numarası/ T.C. Kimlik Numarası	
Cep Telefonu 1	0536 267 87 90	Cep Telefonu 1	
Cep Telefonu 2		Cep Telefonu 2	
e-mail Adresi	smourad69@googlemail.com	e-mail Adresi	
İmza		İmza	

Kurum Kanuni Temsilcisinin/Vekilinin			
Adı, Soyadı		Adı, Soyadı	
Vergi Kimlik Numarası/ T.C. Kimlik Numarası		Vergi Kimlik Numarası/ T.C. Kimlik Numarası	
Cep Telefonu 1		Cep Telefonu 1	
Cep Telefonu 2		Cep Telefonu 2	
e-mail Adresi		e-mail Adresi	
İmza		İmza	

KURUM BİLGİLERİ			
Vergi Kimlik Numarası	0861584690		
Unvanı	AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ		
Kuruluş Yeri	ÜMRANİYE/İST.	Kuruluş Tarihi	02.05.2023
Ticaret Sicil Numarası	457897-5	Tescil Tarihi	02.05.2023
Telefon Numarası	0536 267 87 90	Mersis Numarası	0086158469000001
E-posta Adresi	fatih_21009@hotmail.com	Faks Numarası	
Adres	TEPEÜSTÜ MAH. KAFİYE SK. - NO: 47 İÇ KAPI NO: 3 ÜMRANİYE / İSTANBUL		
Diğer İletişim Bilgileri			

INTERNET HİZMETLERİ KULLANIM BAŞVURU FORMU

(TÜZEL KİŞİLER)

GELİR İDARESİ BAŞKANLIĞI

TARİH: 15.05.2023

ALEMDAĞ .... Vergi Dairesi Müdürlüğü'ne / Başkanlığı'na,

Aşağıda kimlik bilgileri bulunan firmamızın/kurumumuzun Internet Üzerinden Vergi Bilgileri Sorgulama işlemlerini yapabilmesi,bildirim ve beyanname gönderebilmesi için gereğini arz ederiz.

Kurum Yetkilisinin
Adı, Soyadı : SAMIR MOURAD
Vergi Kimlik Numarası : 6232042167
İmza :

KURUM BİLGİLERİ			
Vergi Kimlik Numarası	0861584690		
Unvanı	AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARE		
Kuruluş Yeri	ÜMRANİYE/İST.	Kuruluş Tarihi	02.05.2023
Ticaret Sicil Numarası	457897-5	Tescil Tarihi	02.05.2023
Telefon Numarası	0536 267 87 90	Faks Numarası	
E-posta Adresi	fatih_21009@hotmail.com		
Adres	TEPEÜSTÜ MAH. KAFİYE SK. - NO: 47 İÇ KAPI NO: 3 ÜMRANİYE / İSTANBUL		

Internet ortamında verilen bilgileri sorgulama işlemleri için Maliye Bakanlığı Gelir İdaresi Başkanlığı tarafından verilen ve tarafımızca değiştirilebilecek olan kullanıcı kodu ve kişisel şifrenin korunması ve her türlü riskinin tarafımıza ait olduğunu, Maliye Bakanlığı Gelir İdaresi Başkanlığı'nın herhangi bir sorumluluğunun bulunmadığını, Maliye Bakanlığı Gelir İdaresi Başkanlığı'nın unutulmuş, çalınmış veya kaybedilmiş olan kullanıcı kodu ve kişisel şifremden, bunların üçüncü kişiler tarafından kullanılmasından ve sonuçlarından hiç bir şekilde sorumlu olmadığını kabul ediyoruz.

Bu kullanıcı kodu ve kişisel şifre kullanılarak düzenlenen ve tarafımızca onaylanarak manyetik ortamda gönderilen bildirim ve beyannamelerin doğruluğunu, bunları Maliye Bakanlığı Gelir İdaresi Başkanlığı bilgisayar sistemleri tarih ve saati esas alarak göndermeyi kabul ve taahhüt ediyoruz.

Kurum Yetkilisi

İmza

SAMIR MOURAD

AS COMSAT HABERLEŞME
PLATFORMLARI VE UYDU HABERLEŞME
SİSTEMLERİ SANAYİ VE TİCARET LTD. ŞTİ.
Tepeüstü Mah. Kafiye Sok. No:47/3
Ümraniye / İSTANBUL
Alemdağ V.D.: 0861584690 İTO: 457897-5
Mersis No: 0086-1584-6900-0001

KİŞİSEL VERİLERİN KORUNMASI HAKKINDA BİLGİLENDİRME

6698 sayılı Kişisel Verilerin Korunması Kanunu ("KVKK") uyarınca, genel merkezi Büyükdere Cad. No: 129/1 Esentepe / Şişli / İstanbul olan, 060000268140074 Mersis no'lu **KUVEYT TÜRK KATILIM BANKASI A.Ş.** tarafından veri sorumlusu sıfatı ile, kişisel bilgileriniz/verileriniz tarafımızca KVKK'da açıklandığı çerçevede; kaydedilecek, saklanacak, güncellenecek, mevzuatın izin verdiği durumlarda üçüncü kişilere açıklanabilecek, sınıflandırılabilecek ve yine KVKK'da sayılan şekillerde işlenebilecektir. Konuya ilişkin ayrıntılar ve haklarınız bilgilendirme amaçlı olarak aşağıda özetlenmiştir.

1) Adınız soyadınız, açık adresleriniz, telefon numaralarınız, görsel ve işitsel kayıtlarınız, gelir seviyeniz, eğitim seviyeniz, risk bilgileriniz, işleminize tablet üzerinden ilerlemek istemeniz halinde biyometrik verileriniz, Bankamız nezdindeki hesaplarınıza ilişkin tüm bilgiler de dahil olmak üzere şahsınıza ait her türlü bilgi "kişisel veri" kapsamında olduğundan iş bu kişisel verileriniz bankacılık faaliyetlerimizin Bankamız prosedürleri ve/veya ilgili mevzuata uygun olarak yürütülmesinin temini; sözleşmelerinizin ve talimatlarınızın gereğinin yerine getirilmesi; tarafınıza sunduğumuz hizmet için gerekli değerlendirmelerin yapılması; sizlere ait verilerin doğru ve güncel tutulmasının sağlanması; Bankamız ürün ve hizmetlerini tanıtmak, pazarlamak ya da tarafınıza kutlama ya da temennilerini iletmek gibi amaçlarla her türlü ticari elektronik iletiler gönderebilmek ve Bankamızın sağladığı hizmetlerden kolaylıkla yararlanabilmenizi temin edebilmek, mevzuatın öngördüğü bilgi saklanması, raporlanması ve diğer yükümlülüklere uymak, iş faaliyetlerimizin ve banka içi sistem ve uygulama yönetimi operasyonlarının planlanması, icrası ve sürdürülmesi; güvenlik sebebiyle ve kanundan kaynaklanan yükümlülüklerimiz kapsamında bankamıza ait ATM ve şubelerde kamera görüntülerinin kaydedilmesi; şikayet, itiraz, talep, öneri, memnuniyet gibi bildirimlerin sizlere daha iyi hizmet verebilmemiz için bildirim yönetim sistemimizde kayıt altında tutulması; Bankamızın ticari ve iş stratejilerinin belirlenmesi ve uygulanması, finans, iletişim, pazar araştırması, kredibilite değerlemelerinin yapılması ve satın alma operasyonlarımızın yürütülmesi; bankacılık mevzuatı, sigortacılık mevzuatı, suç gelirlerinin aklanmasının önlenmesine ilişkin mevzuat, sermaye piyasası mevzuatı, vergi mevzuatı, Sosyal Güvenlik Kurumu mevzuatı ve tüketici hukuku başta olmak üzere yurt içi ve uluslararası mevzuata uyumun sağlanması ve bu kanunlar kapsamında risk izleme ve bilgilendirme yükümlülüklerine uyulması; Bankacılık Düzenleme ve Denetleme Kurumu, Türkiye Katılım Bankaları Birliği, Kredi Kayıt Bürosu, Bankalararası Kart Merkezi, Türkiye Cumhuriyet Merkez Bankası, Sermaye Piyasası Kurulu, Mali Suçları Araştırma Kurulu, T.C. Maliye

INFORMATION ON PERSONAL DATA PROTECTION

Pursuant to the Law Number 6698 on the Protection of Personal Data ("PDPL"), your personal information/data shall be recorded, stored, updated within the framework described in PDP, disclosed to third parties in conditions allowed by the legislation, categorized, and also processed in ways listed in PDPL by **KUVEYT TÜRK KATILIM BANKASI A.Ş.**, at registered address at Büyükdere Cad. No: 129/1 Esentepe / Şişli / Istanbul with MERSIS No. 0600002681400074, in the capacity of the data controller. Details on the subject and your rights are summarized below for information.

1) As all kinds of information of yours including your name, family name, mailing address, phone numbers, visual and audio records, income level, education level, risk information, biometric data in case you want to proceed your transaction on tablet and all information on your accounts at our Bank are within the scope of "personal data", your personal data shall be processed **within the framework of legal reasons stated below** by the Data Processors authorized by our Bank in the capacity of the Data Controller for the purposes of ensuring execution of our banking activities in compliance with our Bank's procedures and/or relevant legislation; fulfilling requirements of your agreements and orders; making evaluations necessary for services we offer to you; ensuring accurate and updated keeping of your data; being able to send all kinds of electronic mails to introduce and market products and services of our Bank or send you celebration or good wishes and enable you to benefit easily from the services provided by our Bank, to comply with storing, reporting information and other liabilities stipulated by legislation, planning, performing and maintaining our business activities and in-bank system and implementation management operations; recording camera images of our Bank's ATM and branches for security and within the scope of our liabilities arising from laws; recording notifications as complaint, objection, request, suggestion, satisfaction, etc. in our notification management system to offer you better services; determining and implementing commercial and business strategies of our Bank, performing finance, communication, market survey, credibility assessments and conducting purchasing operations; enabling compliance with national and international legislation primarily banking legislation, insurance legislation, legislation regarding the prevention of laundering of crime revenues, capital market legislation, tax legislation, Social Security Institution legislation and consumer law and following risk monitoring and information liabilities within the scope of these laws; fulfilling information sharing, reporting, information liabilities stipulated by Banking Regulation and Supervision Agency, the Participation Banks Association of Turkey, Credit Bureau, Interbank Card Center, the Central Bank of the Republic of Turkey, Capital Markets Board, Financial Crimes Investigation Board, T.R. Ministry of Finance

KUVEYT TÜRK KATILIM BANKASI A.Ş. | Büyükdere Street No: 129/1 Esentepe / Şişli / Istanbul.
Mersis No: 0600002681400074 | kuveytturk@hs03.kep.tr | www.kuveytturk.com.tr | Tel: 444 0 123
Activity: Any Activity Allowed Under Banking Law 5411.
MSTS.0184.06

Sayfa-Page 4 / 5

DELİL SÖZLEŞMESİ / EVIDENTIAL CONTRACT

Bankanız ile tablet ve mobil ortamda ve/veya benzer şekillerde akdedeceğimiz Sözleşme ile bilgilendirme formları ve bütün ekleri içeriği iş ve işlemler nedeniyle, Bankanızla aramızda çıkabilecek olası her türlü anlaşmazlıkta, Bankanızın özellikle bu ortamlarda tuttuğu kayıtlarının, defterlerinin, muhasebe, bilgisayar, mikrofilm, mikrofiş, ses, görüntü ve diğer kayıtlarının ve belgelerinin esas alınacağını, bunların Hukuk Muhakemeleri Kanunu m.193 anlamında kesin delil olacağını, bunların içeriğini ve doğruluğunu kabul ettiğimizi, ayrıca kişiliğimizi temsile yetkili kişilerin tablet ve mobil ortam da dahil olmak üzere, bankanızın kağıt ortamı dışındaki ortamlarında kişiliğimizi temsilen sözleşme akdetme ve her türlü işlemde bulunmalarını kabul ettiğimizi, dönülemez biçimde kabul, beyan ve taahhüt ederiz. / We irrevocably accept, declare and undertake that in all conflicts which may arise from all annexes, contents, works and transactions of Agreements we shall conclude with your Bank in tablet and mobile environment and/or in such means and of information forms, we accept that the records, books, accounting, computer, microfilm, microfiche, voice, image and other records and documents your Bank specifically keeps in these environments shall be taken as the basis and they shall be material evidence in terms of Art. 193 of the Code of Civil Procedure and that we accept their content and accuracy and also we accept that the persons authorized to represent our personality to conclude agreement and perform all transactions in the capacity of our representative in all environments of your Bank including tablet and mobile environment but excluding paper medium of your Bank.

Ad Soyad / Unvan Name Surname/Title	AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ		
Gerçek Kişi Doğum Tarihi ve Yeri/Real Person Date and Place of Birth	- (stamp: AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LTD.ŞTİ. … No:47/3 Ümraniye / İSTANBUL Alemdağ V.D. [illegible] İTO 457897-5 Mersis No: [illegible])		
Gerçek Kişi TCKN Real Person TRID No	-	Tüzel Kişi Vergi No Legal Entity Tax No.	
Cep Telefonu / Mobile Phone	-	E-Posta E-mail Address	salihbayar@gmail.com
Adres/Address	TEPEÜSTÜ MAH. Mah. KAFİYE SK. - No: 47 Daire: 3 ÜMRANİYE/İstanbul		
Ticari Elektronik İleti / Commercial Electronic Message	Gerek bu sözleşme ile gerekse sözleşmenin imzalanmasından sonra Bankaya bildirdiğimiz faks, telefon, e-posta vs. gibi bütün iletişim araçlarımıza Bankaca her türlü ticari elektronik iletinin gönderilmesine ve bu iletilerin gönderilebilmesi amacıyla sınırlı olmak üzere, gerekmesi halinde kişisel verilerimizin Kişisel Verilerin Korunması Kanunu kapsamında işlenmesine 6563 sayılı Kanun ve ilgili mevzuata uygun olarak; We accept all kinds of commercial electronic messages to be sent by the Bank to all our fax, phone, e-mail, etc. communication tools we have notified to the Bank both during and after the execution of the agreement and processing our personal data when necessary within the scope of the Law on Protection of Personal Data provided that it shall be limited with the purpose of sending these messages and in compliance with the Law number 6563 and the relevant legislation; ☐ *ONAY VERİYORUZ / I/WE GRANT CONSENT* / ☐ *ONAY VERMİYORUZ (REDDEDİYORUZ) I/WE DO NOT GRANT CONSENT (I/WE REFUSE)*		

Vergi Mukimi Olunan Ülkeler / Countries in which being a taxpayer	Bu Ülke Tarafından Verilen Vergi Kimlik Numarası / Tax No Assigned by this Country
Türkiye	Tü 0861584630
Germany	C8WK806N6

KUVEYT TÜRK KATILIM BANKASI A.Ş. | Büyükdere Street No: 129/1 Esentepe / Şişli / Istanbul.
Mersis No: 0600002681400074 | kuveytturk@hs03.kep.tr | www.kuveytturk.com.tr | Tel: 444 0 123
Activity: Any Activity Allowed Under Banking Law 5411.
MSTS.0184.06

Sayfa-Page 5 / 5

***Kişinin Türkiye dışında başka bir ülkede yerleşimi yoksa** (müşterinin ibraz ettiği bilgi/belgelerden teyit edilir); tablo boş bırakılmaz. Ülke kısmına Türkiye, TIN kısmına TCKN yazılır. Daha sonra tablonun bu kısmına** **"Türkiye dışında bir ülkede vergi mükellefiyetim olmadığını beyan ederim."** beyanını müşterinin yazması istenir.
*The table will not be left blank **if the person does not have residence in a country other than Turkey** (this will be confirmed with the information / documents submitted by the customer). Turkey will be entered in the Country field, and TR ID no will be entered in TIN field. Then, it will be requested from the customer to write the statement **"I hereby declare that I have no tax obligation in a country other than Turkey."** in this part of the table**.

**……

- *İşbu metnin ön yüzünde bulunan Kişisel Verilerle İlgili Bilgilendirme Metni'ni (Aydınlatma) okudum ve bilgilendirildim. / I have read and I am informed about the Informing Text Regarding Personal Data (Clarification).*
- *Delil Sözleşmesi'ni okudum, kabul ediyorum. / I have read the Evidential Contract and I agree.*
- *5549 sayılı Kanun ve ilgili Yönetmelik uyarınca beyan: Sözleşme kapsamında gerçekleştireceğimiz iş ve işlemlerde, tamamen kendi hesabımıza hareket edileceğini, her ne surette olursa olsun başkası hesabına hareket edilmeyeceğini; aksine davranışların sorumluluğumuzu mucip olduğunun tarafımıza hatırlatıldığını kabul ve taahhüt ederiz. / Statement as per Law no. 5549 and related Regulations: We accept and guarantee that we were informed any business and transactions that will be carried out as per the Contract will be on our behalf, and under no circumstance no action will be taken on any other third party's behalf; otherwise it will require our responsibility.*

İMZA / SIGNATURE*** ((Tüzel kişilerde unvan/kaşe ile)/(For Legal Entities with title/stamp)	İMZA TEKRARI / REPEAT OF SIGNATURE
TARİH / DATE :12.05.2023	
AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LTD.ŞTİ. Tepeüstü Mah. Kafiye Sok. No:47/3 Ümraniye / İSTANBUL Alemdağ V.D.: 0861584690 İTO: 457897-5 Mersis No: 0086-1584-6900-0001	AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LTD.ŞTİ. Tepeüstü Mah. Kafiye Sok. No:47/3 Ümraniye / İSTANBUL Alemdağ V.D.: 0861584690 İTO: 457897-5 Mersis No: 0086-1584-6900-0001

Aşağıdaki alan; müşterimizin bankacılık işlemlerinin güvenliği için tablet üzerinden alınacak biyometrik imza örnekleri nedeniyle, ***işlemlere tablet üzerinden devam edilmek istenmesi halinde*** *imzalanmalıdır./ Due to the biometric signature samples that will be received through the tablet for the safety of our customer's banking transactions, this field shall be signed* ***if the transaction is to be proceeded through the tablet.***

ÖZEL NİTELİKLİ KİŞİSEL VERİLERLE İLGİLİ ONAY (TABLETE İMZA) / VERIFICATION REGARDING SENSITIVE PERSONAL DATA (SIGNATURE ON TABLET) : *Kuveyt Türk Katılım Bankası A.Ş.'nin tarafıma daha iyi hizmet verilebilmesi için bankanın tarafıma sunacağı ekran, tablet gibi cihazlara banka personeli huzurunda atacağım biyometrik dijital imzamın banka sistemine işlenmesine, bu imzanın banka nezdinde gerçekleştireceğim işlemlerdeki imzaların kontrolü amacıyla kullanılmasına 6698 Sayılı Kanun ve ilgili mevzuata uygun olarak açıkça rızam vardır. / As per Law No. 6698 and the related regulations, I explicitly agree that my biometric digital signature that I sign in the presence of the bank personnel, on the devices such as screens and tablets, provided to me by Kuveyt Türk Katılım Bankası A.Ş. in order to provide a better service, and my signature to be used to check the signatures for the transactions made through the bank.*

ONAY VEREN /CERTIFIER :

İMZA / SIGNATURE : AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LTD.ŞTİ. Tepeüstü Mah. Kafiye Sok. No:47/3 Ümraniye / İSTANBUL Alemdağ V.D.: 0861584690 İTO: 457897-5 Mersis No: 0086-1584-6900-0001

KUVEYT TÜRK KATILIM BANKASI A.Ş. | Büyükdere Street No: 129/1 Esentepe / Şişli / Istanbul.
Mersis No: 0600002681400074 | kuveytturk@hs03.kep.tr | www.kuveytturk.com.tr | Tel: 444 0 123
Activity: Any Activity Allowed Under Banking Law 5411.
MSTS.0184.06

4.6 AS COMSAT & Partners Facilities

4.6.1 Cooperation Partners

4.6.2 Office in Heidelberg/Germany (TEMO / hi enterprises)

Procurement

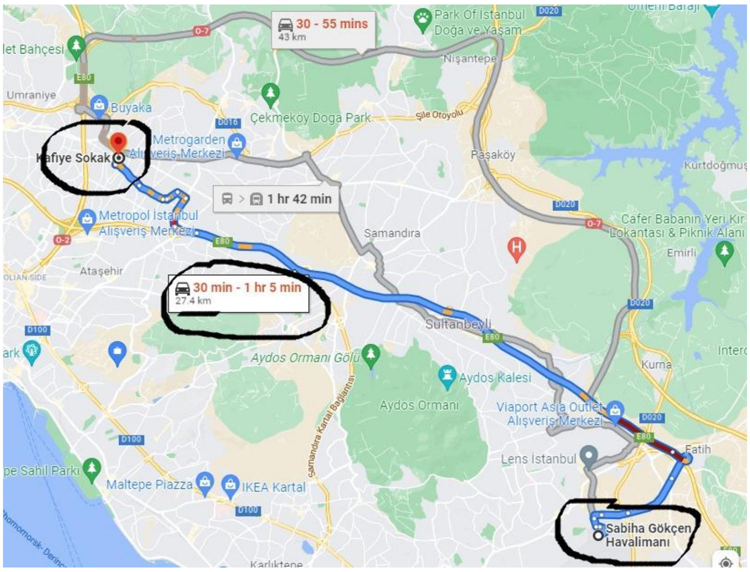
4.6.3 Administration Flat in Istanbul (AS COMSAT)

[19:46, 1/25/2022] Samir Mourad: 3000TL oder 200 Euro für Miete

Erdgas 400TL (~25 Euro), Strom 250TL (~10 Euro), Kalt Wasser 100TL (~6 Euro), Internet 155TL (~11 Euro), Insgesamt ungefaehr 250 Euro

[19:46, 1/25/2022] Samir Mourad: https://goo.gl/maps/2tNKYdmc2fPnNifY9

2 rooms (1 bureau, 1 private room) + kitchen + bath room



	
1 Feb 22	2 Feb 22
	
19 May 22	19 May 22
	
10 May 23	10 May 23

4.6.4 KİRA KONTRATOSU

Dairesi	: ÜMRANİYE
Mahallesi	: TEPEÜSTÜ
Sokağı	: KAFİYE
Numarası	: 47 D:3
Kiralanan şeyin cinsi	: HOME OFİS
Kiraya verenin adı, soyadı ve ikametgahı	: Şennur BAYAR (T.C NO: 56554410246) MERDİVENKÖY MH. ÇÖMLEKÇİ ÇUKURU SK. NO:47A KOSOVA APT. A BLOK, D:14 MERDİVENKÖY-KADIKÖY/ İSTANBUL
Kiracının adı, soyadı ve ikametgahı	: AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ TEPEÜSTÜ MAH. KAFİYE SOK. NO:47/3 ÜMRANİYE/İSTANBUL ALEMDAĞ V.D. VERGİ NO: 0861584690
Bir senelik kira karşılığı	: 12.000 TL (Onikibin Türk Lirası)
Bir aylık kira karşılığı	: 1.000.-TL (Dokuzuyüzyetmişbeş Türk Lirası)
Kira karşılığının ne şekilde ödeneceği	: Her ayın 2'sinde peşin olarak ödenecektir.
Kira müddeti	: 1 yıl
Kiranın başlangıcı	: 2.Mayıs.2023
Kiralanan şeyin şimdiki durumu	: Tam ve mükemmel
Kiralanan şeyin ne için kullanılacağı	: İşyeri olarak kullanılacaktır.
Kiralanan şey ile beraber teslim olunan demirbaş eşyanın beyanı	Aspiratör ve kombi.

1-) 24.12.1980 gün ve 2361 sayılı kanunla değiştirilen 193 sayılı Gelir Vergisi Kanununun 70.94.ve 106. maddeleri gereğince;

Tüccar, serbest meslek erbabı ve çiftçiler, ticari, mesleki ve zirai işleri ile ilgili olarak yaptıkları kira ödemelerinden %25 oranında gelir vergisi tevkifatı yaparak vergi dairesine yatıracaklardır.

2-) 488 sayılı Damga Vergisi Kanununu değiştiren 21.11.1980 gün ve 2344 sayılı ve 30.12.1980 gün ve 2367 sayılı kanunlar gereğince;

Malsahibi ve kiracı için %01 kefil için %05 olmak üzere mukavele müddetine göre bütün kira bedelinin binde altısı nispetinde damga pulu yapıştıracaktır. Teslimat hakkındaki şerhlerde her imza için 501 liradan 1000 liraya kadar(1000 dahil) 10 lira, 1001 liradan 5000 liraya kadar (5000) dahil 15 lira, 5001 liradan 10.000 liraya kadar (10.000 dahil) 20 lira, 10.001 liradan 100.000 liraya kadar(100.000 dahil) 30 lira 100.001 liradan 500.000 liraya kadar olan (500.000 dahil) 50 lira, 500.000 liradan yukarı olanlar 100 liralık damga pulu yapıştırırlar.Pullar tarih ile beraber imza veya mühür konulmak suretiyle iptal olunacaktır.

İki pul için bir iptal muamelesi kabul edilir.

4.6.5 Office in Inilek Sok/Umraniyye

4.6.5.1 Renting Contract

KİRA SÖZLEŞMESİ

KİRALANAN TAŞINMAZ
MAHALLE : Tepeüstü Mahalesi
CADDE = ALEMDAĞ CAD.
SOKAK: Yenilik Sokak NO= 47 /A. ÜMRANİYE/İSTANBUL
KİRALANAN ŞEYİN CİNSİ :TİCARETHANE (DÜKKAN)
KİRAYA VERENİN ADI - SOYADI : ALİ ACAR İKÂMETGAH VE T.C. KİMLİK NO. : 47002798276 TEPEÜSTÜ MAH.YENİLİK SOK. İKİZLER APT. NO 45.ÜMRANİYE/İSTANBUL BANKA HESAP =HALK BANKASI BANKALAR CAD.ŞUBESİ. IBAN=TR30 0001 2009 1360 0009 2159 24
KİRALAYANIN FİRMA ADI -SOYADI =AS COMSAT HABERLEŞME VE UYDU HABERLEŞME SİSTEMLERİ SAN VE TİC.LTD. (VEKELETEN) SALİH BAYAR TC.20647473968
BİR YILLIK KİRA KARŞILIĞI:108.000 (YÜZSEKİZBİN) TL +KDV.
BİR AYLIK KİRA KARŞILIĞI:9.000 (DOKUZ BİN)+KDV
KİRANIN NE ŞEKİLDE ÖDENECEĞİ : HER AY PEŞİN
KİRA SÜRESİ : 1 YIL (BİRSENE)
KİRANIN BAŞLANGIÇ TARİHİ : 20.05.2023 - 20.05.2024
KİRALANAN ŞEYİN ŞİMDİKİ DURUMU : BOŞ
KİRALANAN ŞEYİN NE İÇİN KULLANILACAĞI =TİCARETHANE (YAZHANE)
KİRALANAN ŞEY İLE BERABER TESLİM OLUNAN DEMİRBAŞ EŞYANIN BEYANI,OTOMATİK ELEKTRİKLİ KEPENKLER,ALİMUNYUM DOĞRAMALI KAPI VE PENCERLER YER KAROLU,LED SPOT,WC , ,LEVBO KLOZET TAŞI, MUSLUK BOYALI BADALI SIFIR MÜKEMEL DÜKKAN KİRA SÖZLEŞMESİ ÖZEL HÜKÜMLERİ

1-Kiracı kiraladığı şeyi kendi malı gibi kullanmaya ve bozulmasına evsaf meziyetlerini şöhret ve itibarını kaybetmesine meydan vermemeye mecburdur.

2- Kiralananın İŞYERİNİN yıllık zam artışı devletin açıkladığı TEFE+TÜFE ½ ORANINDA yapılacaktır.

3-Kiralanan yerin su, elektrik, , masrafları, apartmanın, , temizlenmesi gibi nedenlerle doğacak apartman aidatları vb giderler kiracıya aittir.

4-Kiralanan malik tarafından satılığa çıkarılacak olursa kiracı müşteri adaylarının kiralananı gezip görmelerine müsaade edecektir.

5- Kiracı kiraladığı şeyi ne halde buldu ise kiraya verene o halde teslim etmeye mecburdur.

Ancak kiralananda, giderilmesi kiraya verenin sorumluluğu kapsamında bulunan herhangi bir arıza veya hasar meydana gelirse, kiracı durumu kiraya verene yazılı olarak ihbar edecektir. İhbar yapmadan, kiracının kendiliğinden yapacağı harcamalardan kiraya veren sorumlu olmayacaktır. Kiracı kiraya verenin muvafakatı çerçevesinde anlaşarak, kiralananda bir takım faydalı tadilat, tamirat ve dekorasyon yapabilir. Tahliye sırasında kiraya verenin seçimlik hakkı yapılan anlaşma ile sınırlı olacaktır.

6- Elektrik, su ve belediye abonelikleri, sözleşmenin imzalanmasından itibaren bir hafta içinde işlemlerin takibi harcamaları yapma sorumluluğu kendilerine ait olmak üzere, kiracının kendi adına yaptıracaktır.

7- Kiracı, kiralanan gayrimenku'lun kira bedeli üzerinden 193 Sayılı Gelir Vergisi Kanunun 94. maddesi gereğince %20 oranında gelir vergisi tevkifatı yapacaklardır.YADA KDV.OLARAK ÖDİYECEKTİR. İŞ YERİ İÇİN

8-Kiralanan şeyin vergisi ve tamiri kiraya verene kullanılması için lazım gelen temizleme ıslah masrafları kiracıya aittir. Bu hususta âdete bakılır.İŞ YERİ İÇİN

9- Kiracı bu kira kontratı ek 1 ve 2 belirtilen hususi şartlar baki kalmak şartı ile, dönem sonunda kiralananı tahliye etmek isterlerse bu isteklerini dönem sonundan en az bir ay önce, kiraya verene yazılı olarak bildirecektir.

10-Kiracı kira sözleşmesi imzalanmasıyla iki aylık kira bedeli olan 18.000 TL depozito olarak kiraya veren hesabına yatıracaktır. Bina içerisinde evcil hayvan bakmak beslemek yasktır..binanın ön cehpesine 'de .ÖN BAHÇE KISMINA ÇÖP ATILMASI VE ÇİMENLERE BASILMASI YASAKTIR.

11-Bu sözleşmede yazılı bulunmayan hükümlere ihtiyaç duyulduğunda 6570 sayılı kira kanunu, Medeni kanun, Borçlar kanunu,634 Sayılı kat mülkiyeti kanunu ve diğer yürürlükteki alakalı kanun ve Yargıtay kararları uygulanır

12-İhtilaf Halinde İstanbul Anadolu Mahkemeleri ve İstanbul Anadolu İcra Müdürlükleri yetkilidir.

Tarafların özgür rızaları tahtında tanzim ve imza olunan ve 12 maddeden ibaret " Özel hükümleri" içeren işbu sözleşme; iki suret olarak düzenlenmiş ve taraflara birer sureti verilmiştir.

Düzenleme Tarihi 18.05.2023 iki sayfada ibarettir.

KİRACI	**KİRAYA VEREN MALSAHİBİ**
AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SAN VE TC. LTD ŞTİ.	**ALİ ACAR**
V. SALİH BAYAR	

BENAR AYDINLATMA SANAYİ ve TİCARET ALİ ACAR
Şair Ziyapaşa Caddesi Kapak Çiçeği Sk. Taşkın Han No: 2/4 Kat: 1 Karaköy 34420 İSTANBUL
Tel: (0212) 251 43 11 - 251 43 17 Fax: 251 43 05
Beyoğlu V.D.: 410 027 9827

MIETVERTRAG

VERMIETETES EIGENTUM

NACHBARSCHAFT: Tepeüstü-Viertel

CADDE = ALEMDAĞ CAD.

STRASSE: Yenilik Sokak NR= 47 /A. UMRANİYE/İSTANBUL

Art der vermieteten Dinge: Gewerbehalle (Laden)

VOR- und NACHNAME DES VERMIETERS: ALI ACAR

WOHNSITZ UND T.C. ID-NR.: 47002798276

TEPEÜSTÜ MAH. YENİLİK SOK. IKILER APT. NEIN 45.UMRANİYE/İSTANBUL

BANKKONTO = HALK BANK BANKALAR CAD.ZWEIG.

IBAN=TR30 0001 2009 1360 0009 2159 24

FIRMENNAME DES VERMIETERS – NACHNAME =AS COMSAT HABERLEŞME UND SATELLITE COMMUNICATIONS SYSTEME INDUSTRIE UND HANDEL LTD.

(VEKELETEN) SALİH BAYAR TR.20647473968

EINJÄHRIGE MIETE: 108.000 (Hundertachttausend) TL + MwSt.

MIETE FÜR EINEN MONAT: 9.000 (NEUNTAUSEND) + MwSt

WIE DIE MIETE BEZAHLT WIRD: JEDEN MONAT IM VORAUS

MIETDAUER: 1 JAHR (EIN JAHR)

STARTDATUM DES LECKS: 20.05.2023 – 20.05.2024

AKTUELLER STATUS DES VERMIETETEN IMMOBILIEN: LEER

WOFÜR WIRD DAS VERMIETETE GEGENSTAND VERWENDET = GEWERBEHALLE (GELÄNDE)

ERKLÄRUNG DER MIT DEM MIETGEGENSTAND GELIEFERTEN EINRICHTUNGEN, AUTOMATISCHE ELEKTRISCHE ROLLÄDEN, TÜREN UND FENSTER AUS ALUMINIUM, BODENFLIESEN, LED-SPOT, WC, LEVBO-WC-STEIN, WASSERHAHN
BLUKAN BADALI

BESONDERE BESTIMMUNGEN DES MIETVERTRAGS

1-Der Mieter sollte die gemietete Sache als sein Eigentum nutzen und sie, seine Tugenden, seinen Ruhm und sein Ansehen beeinträchtigen. Er darf nicht zulassen, dass er verliert.

2- Die jährliche Erhöhung des gemieteten ARBEITSPLATZES erfolgt in Höhe von ½ WPI + CPI, die vom Staat bekannt gegeben wird.

3-Kosten wie Wasser, Strom, Mietkosten, Wohnungsgebühren usw., die aus Gründen wie der Reinigung der Wohnung entstehen, gehen zu Lasten des Mieters.

4- Wenn das Mietobjekt zum Verkauf angeboten wird, gestattet der Mieter potenziellen Kunden, das Mietobjekt zu besichtigen.

5- Der Mieter ist verpflichtet, die von ihm gemietete Sache dem Vermieter in dem Zustand zu übergeben, in dem er sie vorgefunden hat.

Sollte es jedoch zu Störungen oder Schäden an der Mietsache kommen, die im Verantwortungsbereich des Vermieters liegen, hat der Mieter dies dem Vermieter schriftlich mitzuteilen. Der Vermieter haftet nicht für die dem Mieter ohne Vorankündigung entstandenen Kosten. Der Mieter kann mit Zustimmung des Vermieters einige nützliche Änderungen, Reparaturen und Dekorationen an der Mietsache vornehmen. Das Wahlrecht des Vermieters während der Räumung beschränkt sich auf die getroffene Vereinbarung.

6- Strom-, Wasser- und Gemeindeabonnements werden im Namen des Mieters innerhalb einer Woche nach Vertragsunterzeichnung erstellt, unter der Verantwortung, die Transaktionen zu verfolgen und die Ausgaben zu tätigen.

7- Der Mieter muss gemäß Artikel 94 des Einkommensteuergesetzes Nr. 193 eine Einkommensteuer in Höhe von 20 % auf den Mietpreis der gemieteten Immobilie einbehalten. FÜR DEN ARBEITSPLATZ

8-Die Steuer- und Reparaturkosten der Mietsache, die für die Nutzung des Vermieters erforderlich sind, gehen zu Lasten des Mieters. Dabei werden die Gepflogenheiten berücksichtigt. FÜR DEN ARBEITSPLATZ

9- Am Ende des Zeitraums, sofern die in Anhang 1 und 2 dieses Mietvertrags genannten Sonderbedingungen eingehalten werden. Möchte er die Mietsache räumen, muss er seinen Antrag spätestens einen Monat vor Ablauf der Frist schriftlich beim Vermieter einreichen. werde melden.

10-Der Mieter hinterlegt bei Unterzeichnung des Mietvertrags den zweimonatigen Mietbetrag von 18.000 TL als Kaution auf das Konto des Vermieters. Es ist verboten, sich innerhalb des Gebäudes um Haustiere zu kümmern. Es ist verboten, Müll auf die Vorderseite des Gebäudes zu werfen und auf den Rasen zu treten.

11-Wenn Bestimmungen erforderlich sind, die nicht in diesem Vertrag enthalten sind, gelten das Mietgesetz Nr. 6570, das Bürgerliche Gesetzbuch, das Obligationenrecht, das Eigentumswohnungsgesetz Nr. 634 sowie andere anwendbare Gesetze und Entscheidungen des Obersten Gerichtshofs.

12- Im Streitfall sind die anatolischen Gerichte und die anatolischen Exekutivdirektionen von Istanbul zuständig.

Dieser Vertrag, der mit freier Zustimmung der Parteien erstellt und unterzeichnet wird und „Sonderbestimmungen" bestehend aus 12 Artikeln enthält; Es wurde in zwei Exemplaren erstellt und den Parteien eine Kopie ausgehändigt.

Ausgabedatum 18.05.2023 besteht aus zwei Seiten.

MIETER

ALS COMSAT-KOMMUNIKATIONSPLATTFORMEN

UND UYDU HABERLEŞME SAN VE TC. GMBH.

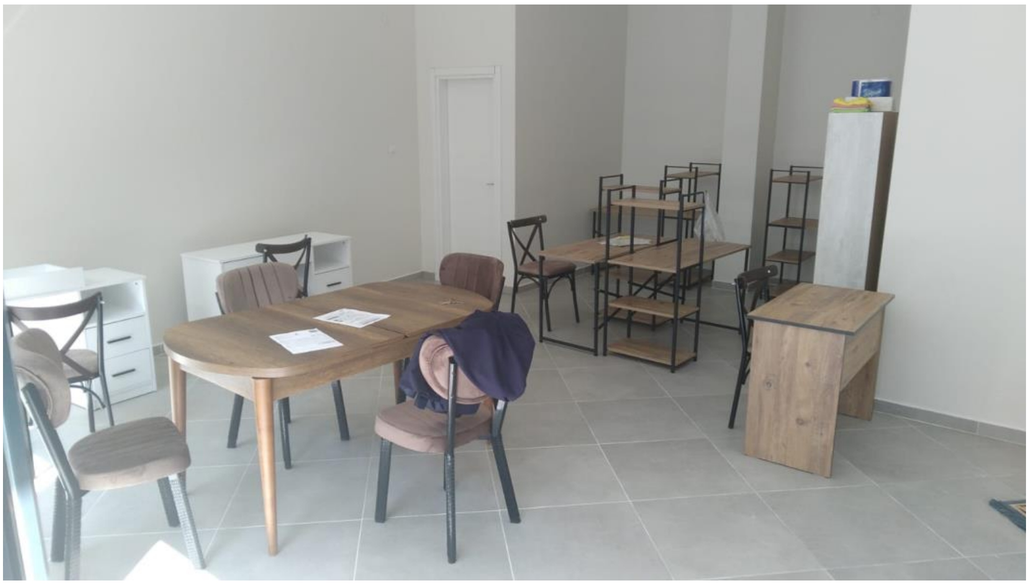
V. SALİH BAYAR

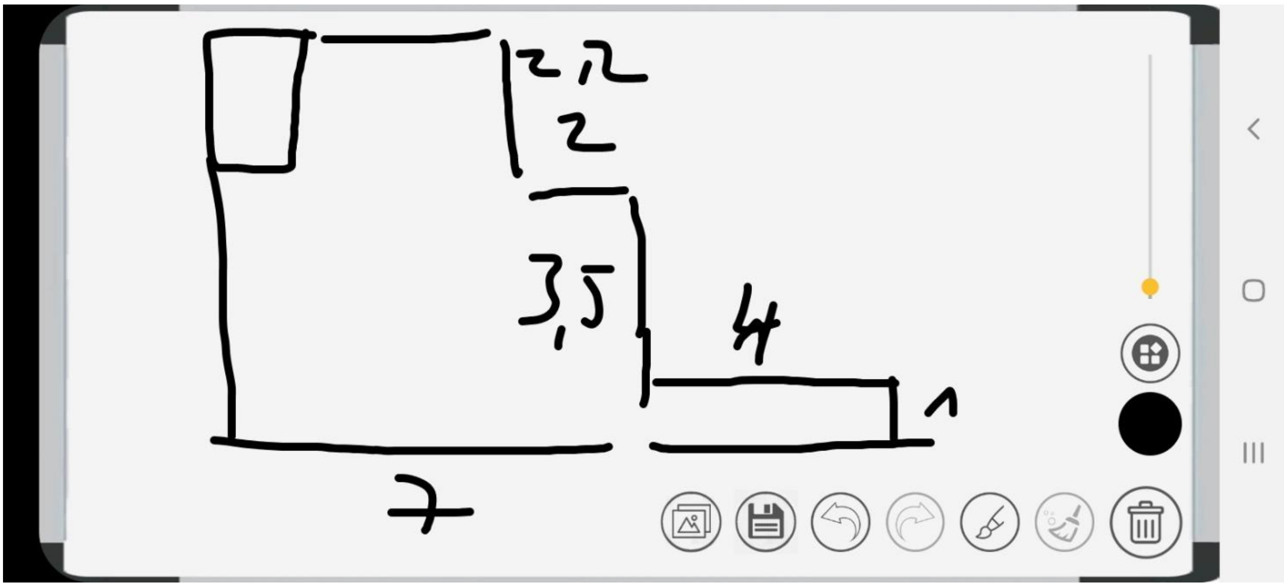
VERMIETER-EIGENTÜMER

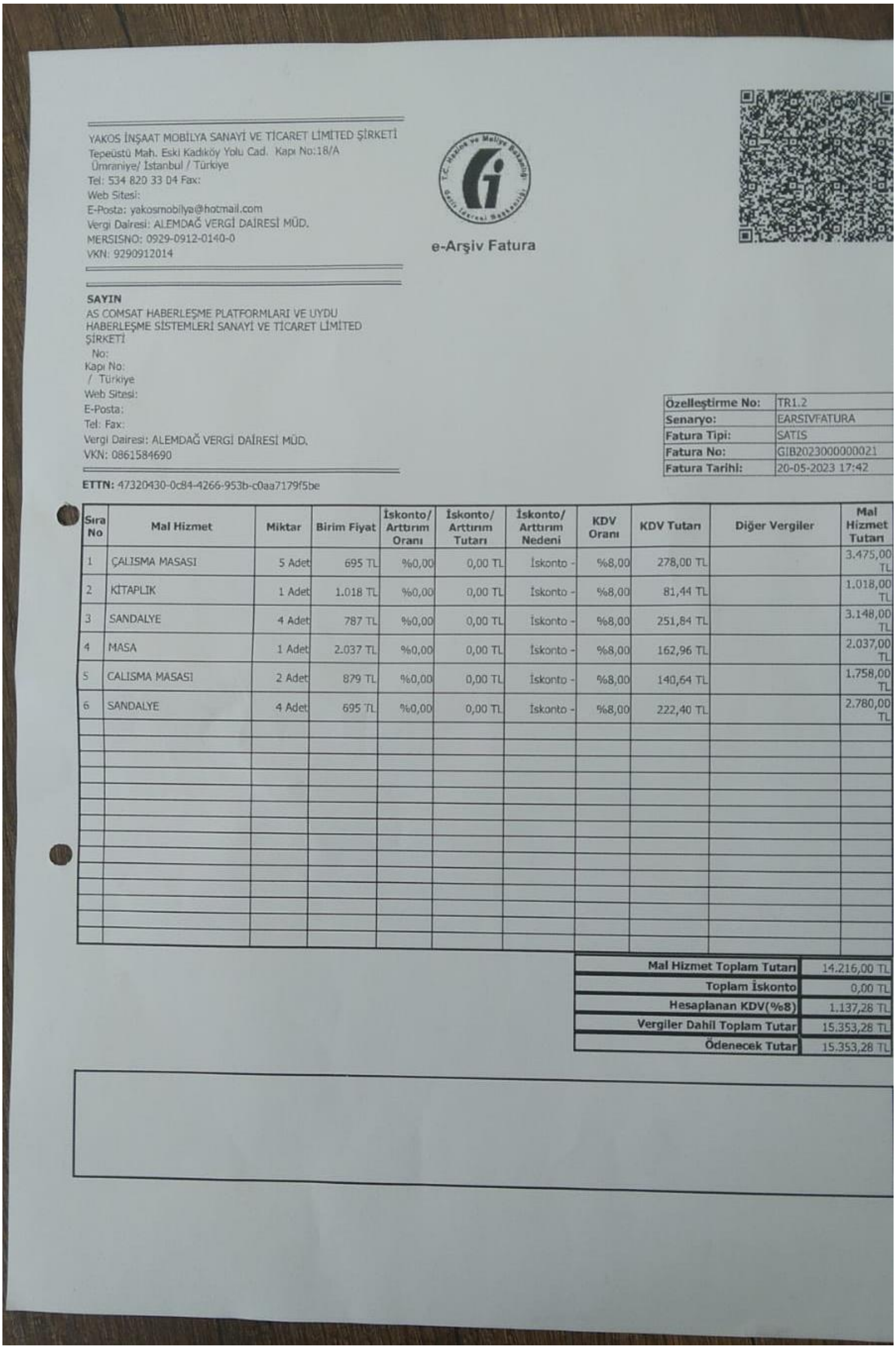
ALI ACAR

ba BENAR
BELEUCHTUNG
Industrie und Handel
ALI ACAR
Sair Ziyapa Street Cover Flower Taşkın Han
Nr.: 2 Etage 1 Karak y 34420 1STANBUL
Tel: (0212) 25 43 11-21 43 17 F 251 43/08
Beyoglu D40 027 9827

4.6.5.2 Office Inventory




2,2
2
3,5
4
1
7



YAKOS İNŞAAT MOBİLYA SANAYİ VE TİCARET LİMİTED ŞİRKETİ
Tepeüstü Mah. Eski Kadıköy Yolu Cad. Kapı No:18/A
Ümraniye/ İstanbul / Türkiye
Tel: 534 820 33 04 Fax:
Web Sitesi:
E-Posta: yakosmobilya@hotmail.com
Vergi Dairesi: ALEMDAĞ VERGİ DAİRESİ MÜD.
MERSISNO: 0929-0912-0140-0
VKN: 9290912014

e-Arşiv Fatura

SAYIN
AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ
No:
Kapı No:
/ Türkiye
Web Sitesi:
E-Posta:
Tel: Fax:
Vergi Dairesi: ALEMDAĞ VERGİ DAİRESİ MÜD.
VKN: 0861584690

Özelleştirme No:	TR1.2
Senaryo:	EARSIVFATURA
Fatura Tipi:	SATIS
Fatura No:	GIB2023000000021
Fatura Tarihi:	20-05-2023 17:42

ETTN: 47320430-0c84-4266-953b-c0aa7179f5be

Sıra No	Mal Hizmet	Miktar	Birim Fiyat	İskonto/ Arttırım Oranı	İskonto/ Arttırım Tutarı	İskonto/ Arttırım Nedeni	KDV Oranı	KDV Tutarı	Diğer Vergiler	Mal Hizmet Tutarı
1	ÇALISMA MASASI	5 Adet	695 TL	%0,00	0,00 TL	İskonto -	%8,00	278,00 TL		3.475,00 TL
2	KİTAPLIK	1 Adet	1.018 TL	%0,00	0,00 TL	İskonto -	%8,00	81,44 TL		1.018,00 TL
3	SANDALYE	4 Adet	787 TL	%0,00	0,00 TL	İskonto -	%8,00	251,84 TL		3.148,00 TL
4	MASA	1 Adet	2.037 TL	%0,00	0,00 TL	İskonto -	%8,00	162,96 TL		2.037,00 TL
5	CALISMA MASASI	2 Adet	879 TL	%0,00	0,00 TL	İskonto -	%8,00	140,64 TL		1.758,00 TL
6	SANDALYE	4 Adet	695 TL	%0,00	0,00 TL	İskonto -	%8,00	222,40 TL		2.780,00 TL

Mal Hizmet Toplam Tutarı	14.216,00 TL
Toplam İskonto	0,00 TL
Hesaplanan KDV(%8)	1.137,28 TL
Vergiler Dahil Toplam Tutar	15.353,28 TL
Ödenecek Tutar	15.353,28 TL

4.7 AECENAR Laboratories

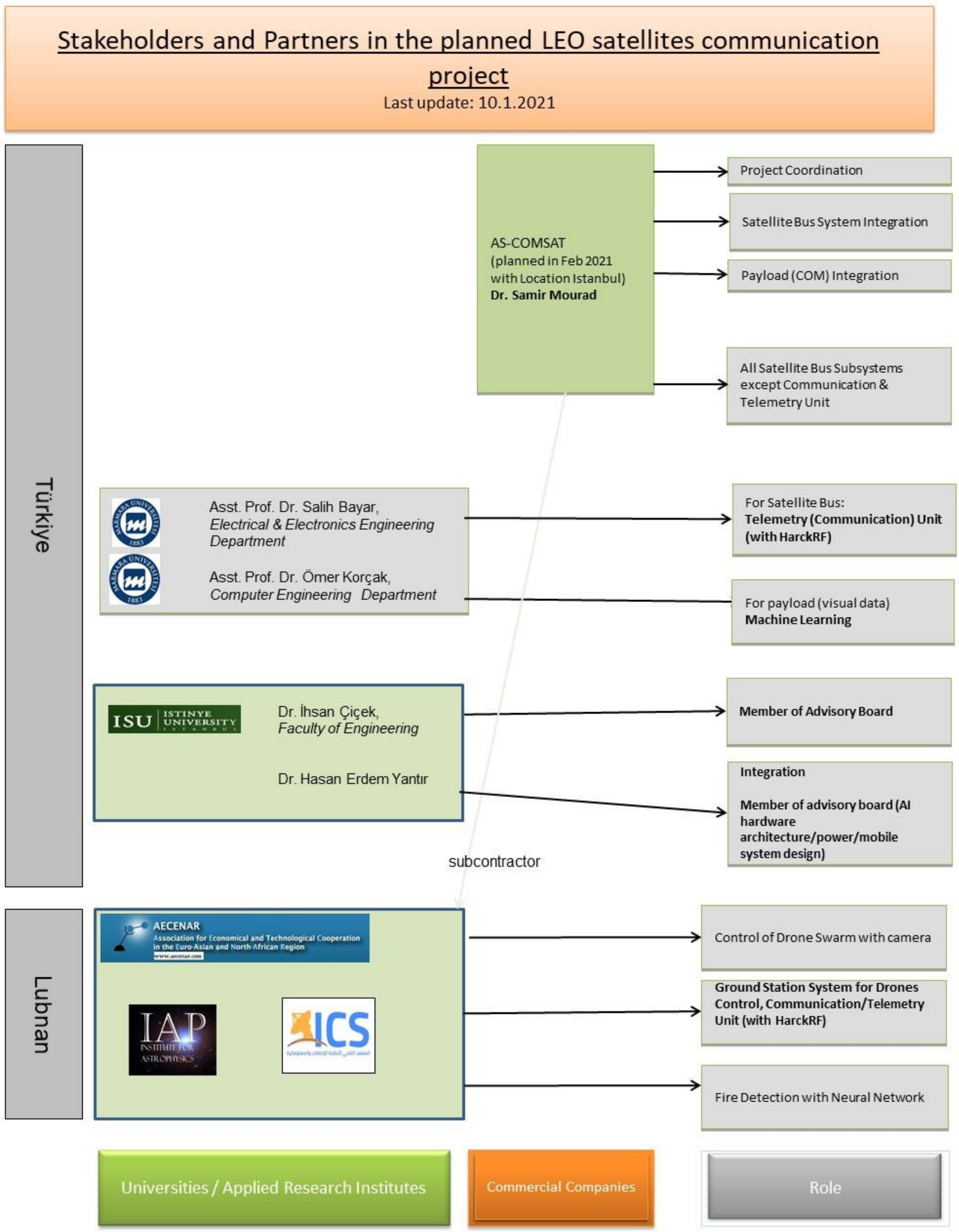
Ras Maska/Tripoli, www.aecenar.com

Electronic Lab, Vacuum Chamber (to be completed)

4.8 History of AS COMSAT

4.8.1 Planned Stakeholders in Oct 2021

بسم الله الرحمن الرحيم



4.8.2 Company vision

2023 Demonstration System for Exhibition

2023 Establishing company space and Testing facility at Istanbul

14.-17.9.2023 IMEX exhibition in Istanbul

2024-2025 First Operational System

2026-2030 International Supplier

4.8.3 Specification Phase Aug 2020 - May 2021

AnalysisPhase_Specification_SupplierParts_pptx

4.8.4 Planning Aug 2020 - Jul 2023 (3 years)

Aug 2020 – Sep 2021	Team Foundation through weekly zoom meeting
2021	Development 2U Sat (COM(Transponder), Scientific (X-Ray Sensor) + 1U Sat (COM) + Ground Station
2022	
2023	

4.8.5 Initial Investment Planning for 1U Satellite (last update: Dec 2020)

Bismillah Last update: 19 Dec 2020

AS-COMSAT cubeSAT System (1 Satellite 10cmx10cmx10cm)

Material (from Suppliers) & Engineering Costs

Satellite Development Cost and Launch Cost

Working Package	Material Cost	Man Month	Qualification	Salary/MM	Personnel Cost per item	Total item cost		Material from suppliers
Camera	$50,000	6	Eng	$2,000	$12,000	$62,000		cube sat solar
COM&Telemetry	$50,000	10	Eng	$2,000	$20,000	$70,000		deployable solar
Gyroscopes	$20,000	5	Eng	$2,000	$10,000	$30,000		ISIS cubesat scolare
Accelerometers	$20,000	5	Eng	$2,000	$10,000	$30,000		crystal space P14 "Vasik"
Reaction Wheel	$10,000	3	Eng	$2,000	$6,000	$16,000		EXA BA0 high energy density battery array
			Eng	$2,000	$0	$0		EXA Titan-1 350 whr high energy density...
Power System Solar panels, battery system	$15,000	4	Eng	$2,000	$8,000	$23,000		
			Eng	$2,000	$0	$0		
Board Control Computer	$10,000	8	Eng	$2,000	$16,000	$26,000		
Antenna system	$10,000	3	Eng	$2,000	$6,000	$16,000		
Integration	$10,000	3	Eng	$2,000	$6,000	$16,000		
Test	$10,000	2	Eng	$2,000	$4,000	$14,000		
Launch	$44,000	1	Eng	$2,000	$2,000	$46,000		
Ground Station	$20,000	6	Eng	$2,000	$12,000	$32,000		
	$269,000							
				Total Cost		**$381,000**		

Operational Cost per year

Working Package	Material Cost	Man Month	Qualification	Salary/MM	Personnel Cost per item	Total item cost
Maintenance	$40,000	12	Eng	$2,000	$24,000	$64,000
Ground Station	$20,000	36	Eng	$2,000	$72,000	$92,000
				Total Cost		**$156,000**

-> 381 k$ Investment

4.8.6 Investment Planning update for 2U + 1U Satellite + Ground Station System (last update: May 2021)

Goals of investment:

- Demonstration System (2U+1U+Ground Station)
- Team Building:
- Payloads: AIS Ship Communication System, Astronomical X-Ray Sensor

Engineering (ManPower Aug 2021-Jul 2023) 2 yrs x 5 persons x 12 k$	120 k$
Administration TEMO Aug 2020 – Jul 2021 (Analysis Phase)	25 k$
Administration TEMO (2 yrs Aug 2021 – Jul 2023) 2 x 25k$	50 k$
Material Cost Satellites (1U + 2U) + Ground Station	75 k$
Launching Cost (1U + 2U)	150 k$
Total	**$420,000**

4.8.7 Budget for 2022 (bureau in Istanbul from 2/2022)

rent Istanbul	Feb-Dec 2022	2.750 €
Salih Alfmeier		600 €
Reisekosten		3.600 €
Satellite material		40.000 €
Rozan Mustafa	Sep-Dec 2022	4.000 €
	Sum	**50.950 €**

4.8.8 Milestones 2022

Tue, 1.2.2022: Taking bureau in Istanbul

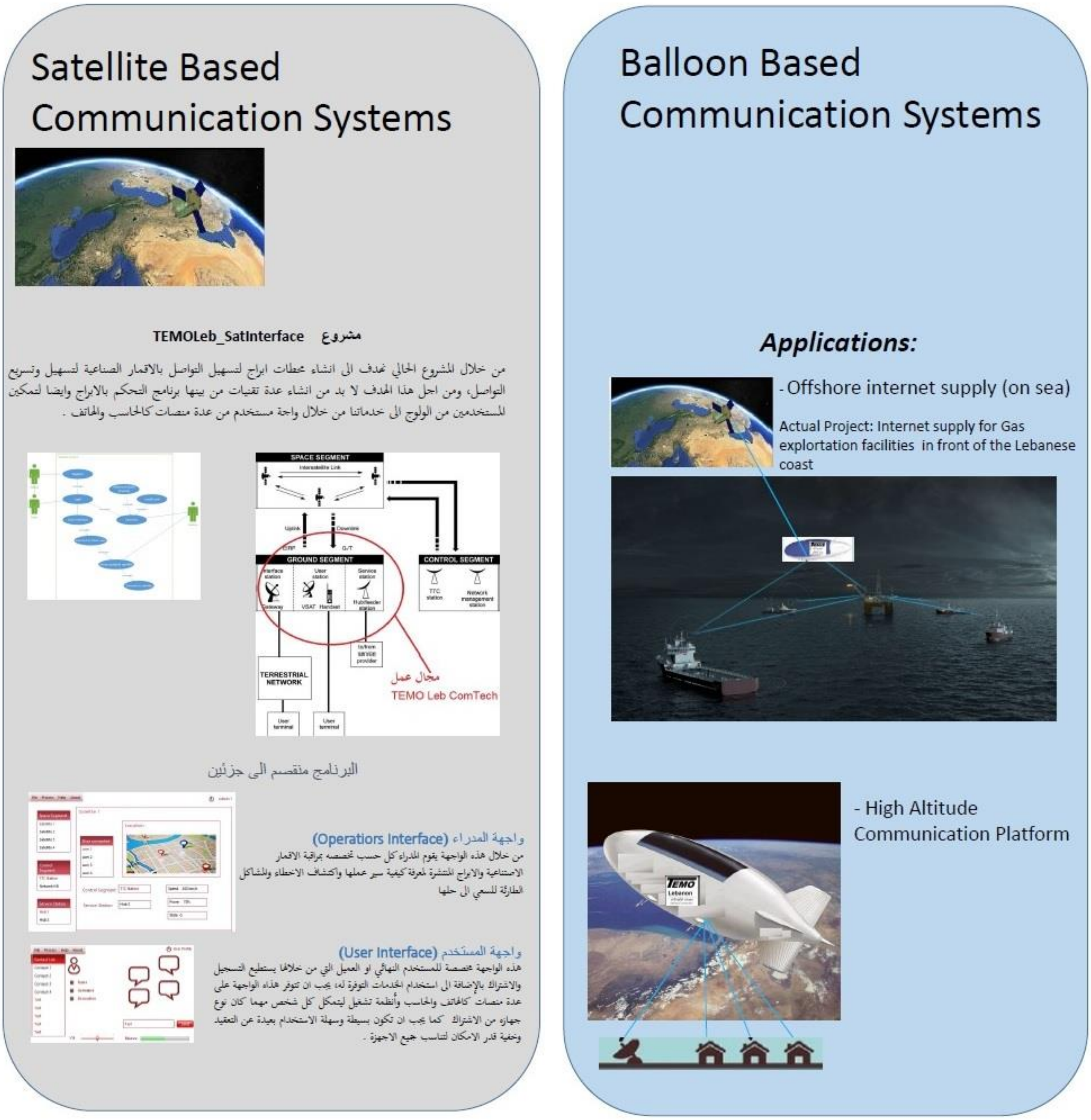
In AECENAR: COM unit (with HackRF)

In Istanbul: Integration of satellite (with local students)

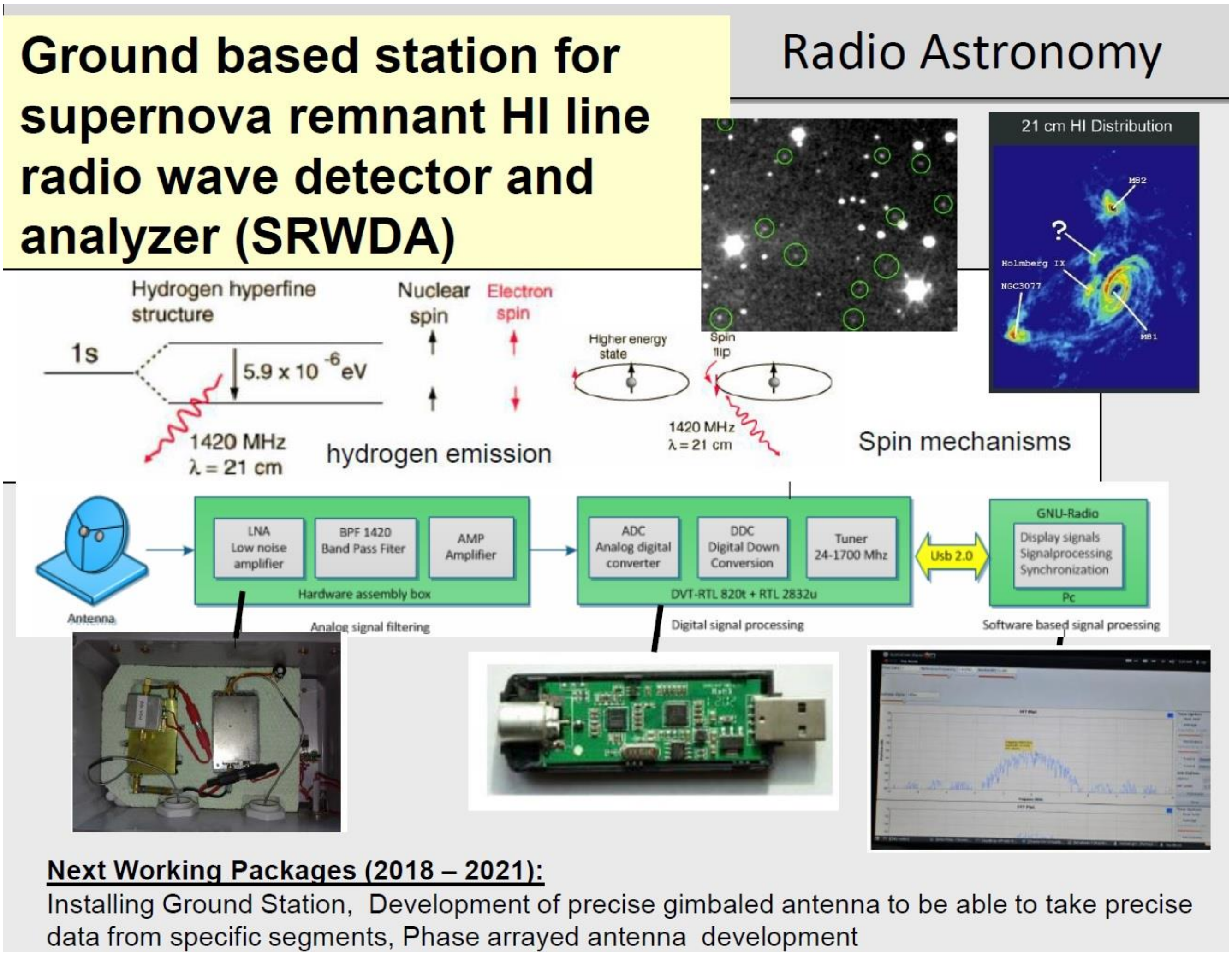
In Istanbul: electronics lab, integration and testing lab, marketing / meeting space for customers

4.9 Predevelopment Studies 2012- 2022

4.9.1 Satellite-airship COM system



4.10 Payload Radioastronomy Sensor

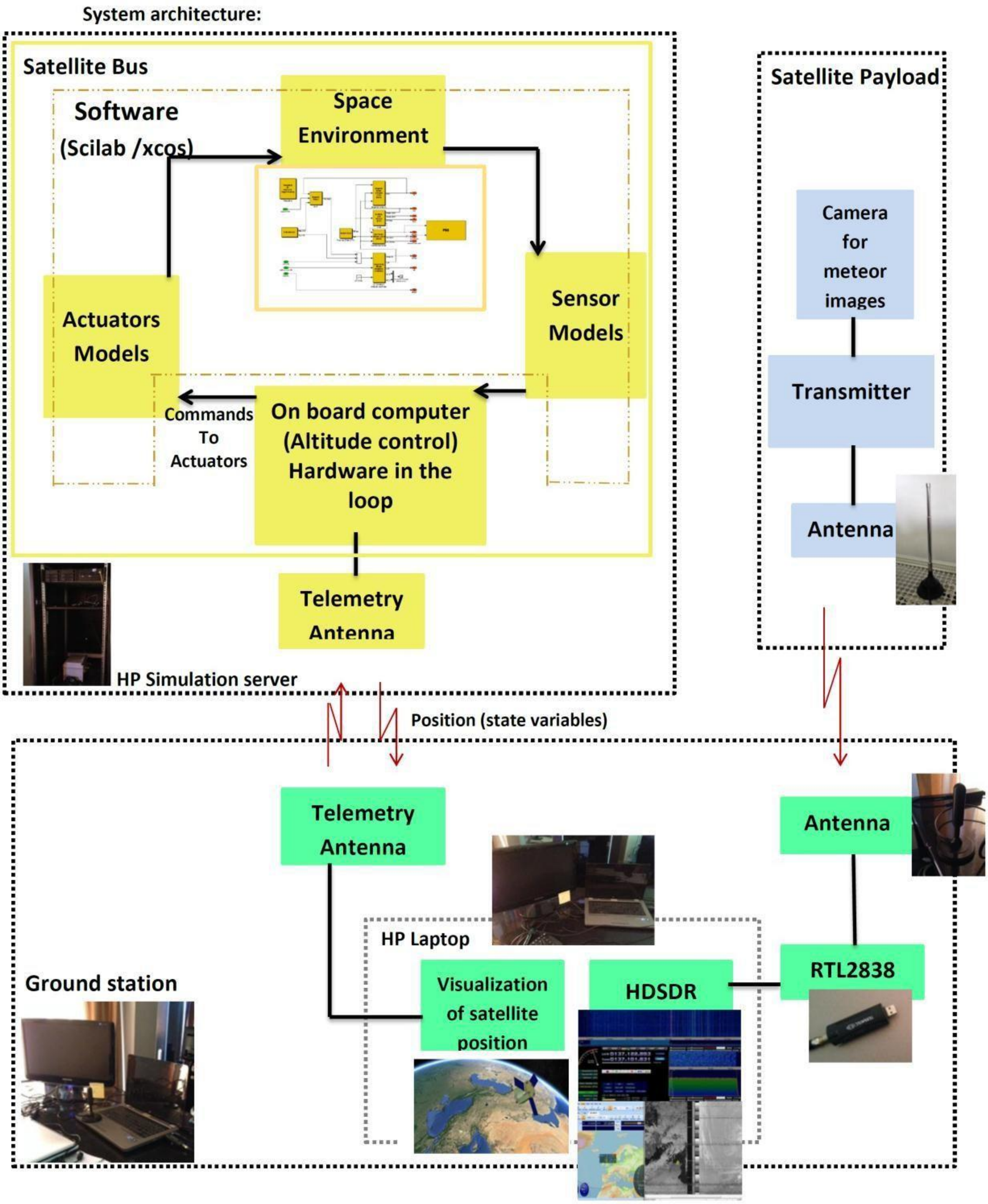


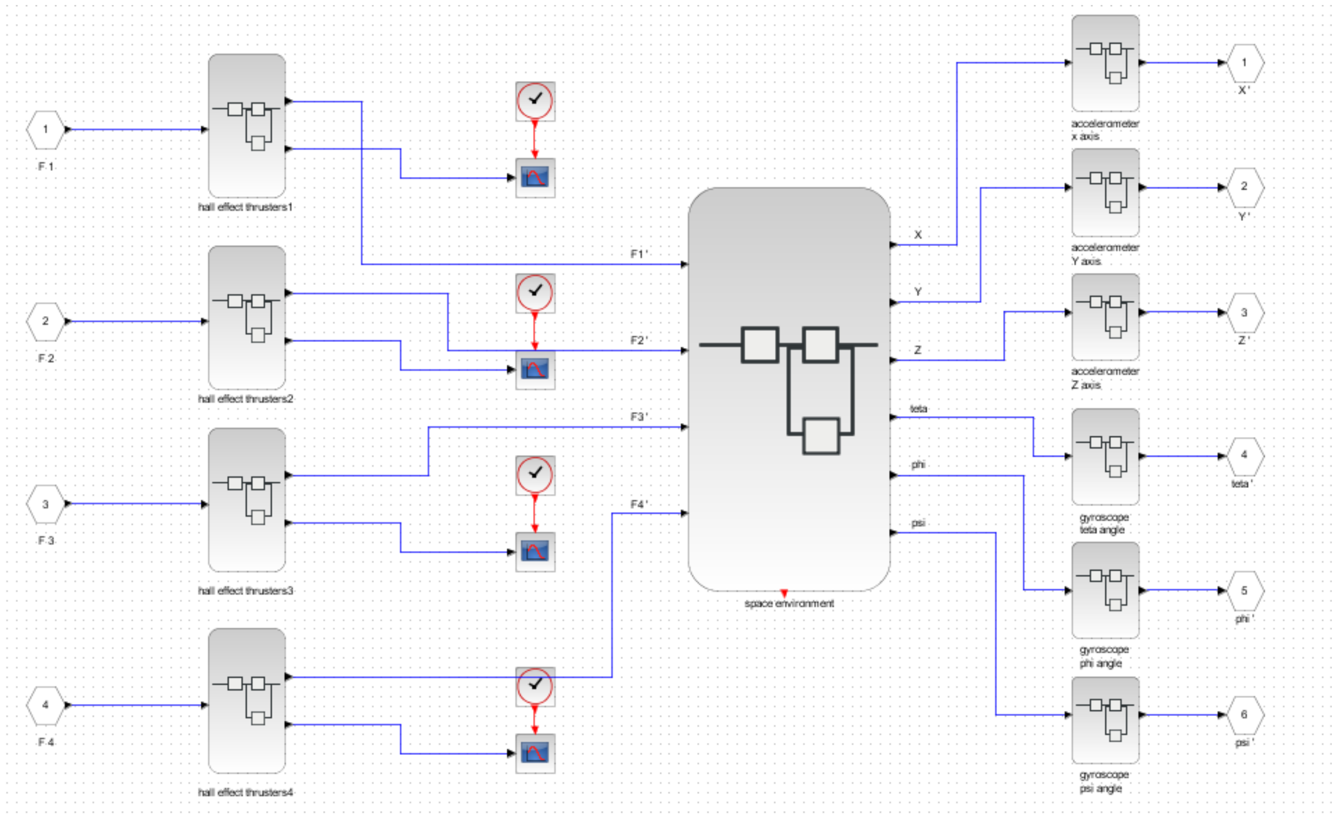
See AECENAR IAP-SAT Final Report (2013-2020),
http://aecenar.com/index.php/downloads/send/10-iap/664-iap-sat-final-report-2012-2020-pdf

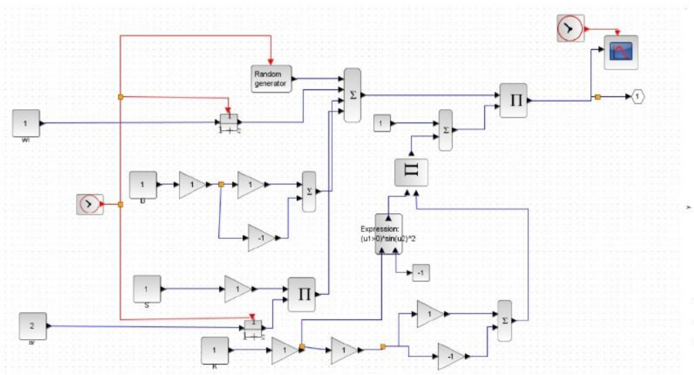
4.10.1 LEO Satellite System

Tasks	Needed Human Resource	External Cost	Year
TEMOLeb-Mintad Gas filling	2 Technicians, 4 MM (man months)	5000$	2019
TEMOLeb-Mintad& TEMOLeb-SAT solar panels and system	-	2000$	2019
TEMOLeb-Mintad Control System development & Validation (simulink/scilab)	Eng., 5 MM	400$	2019
TEMOLeb-Mintad Actuators, Wings integration	2 Technicians, 2 MM	1000$	2019
TEMOLeb-Mintad Sensors integration	Eng., 1 MM	300$	2019
TEMOLeb-SAT Actuator (chem.)	1 Technician, 1 MM	1500$	2019
TEMOLeb-SAT Platform Integration	1 Technician, 1/2 MM	500$	2019
TEMOLeb-Mintad Telemetry Unit	2 Eng., 6 MM	3000$	2020
TEMOLeb-SAT Telemetry Unit	2 Eng., 6 MM	3000$	2020
TEMOLeb-Mintad COM Unit (Payload)			2020
TEMOLeb-SAT COM Unit (Payload)			2020

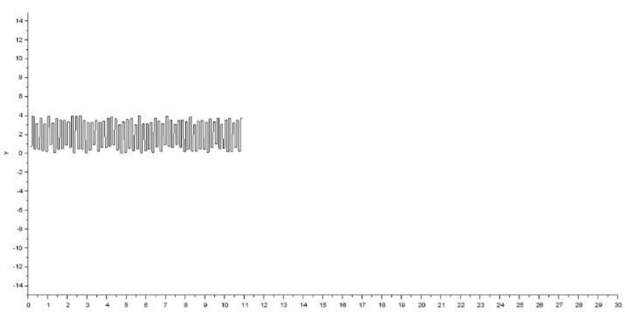
4.10.2 Hardware-in-the-Loop test rig for IAP-SAT (Overview)







After simulation:



In this figure, input 1 is the angular speed wi, and imput 2 is the acceleration. These inputs is passed by the scale factor, the noise, transfer function and the misalignement of the axis to measure the final angular speed as output of gyroscope.

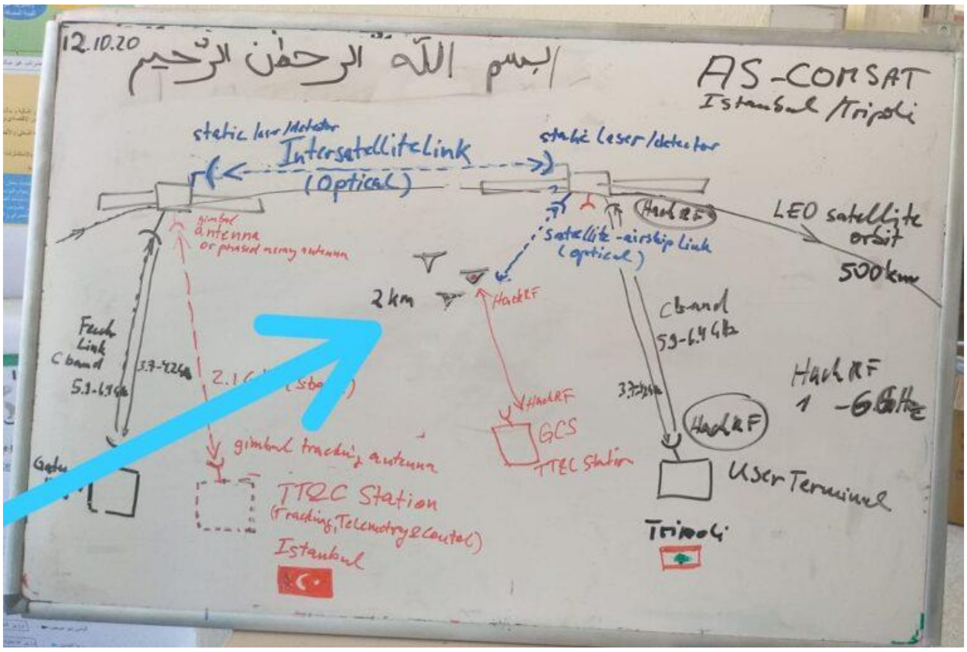
Overview is including Simulation model of actuators, space model and sensors of IAP-SAT (for 1. xcos model, 2. Graphs of simulation results please refer to documentation)

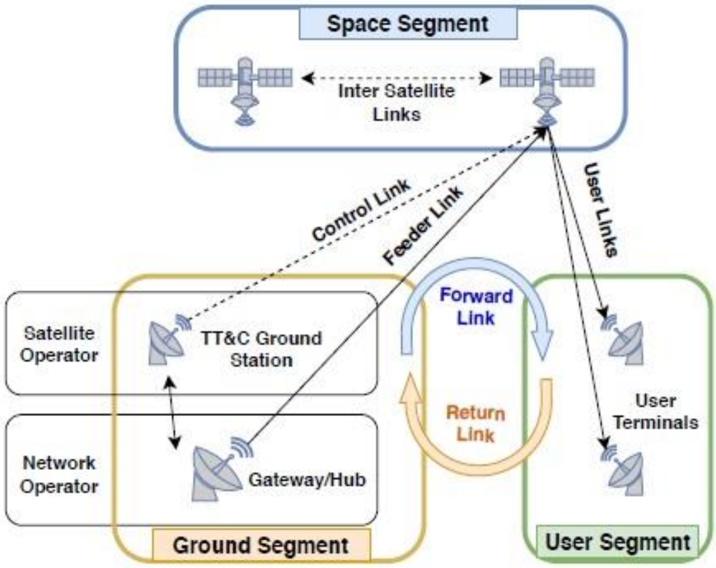
- Interface between Simulation Server and Board Computer of IAP-SAT (for 1. xcos model, 2. Graphs of simulation results): please refer to documentation)

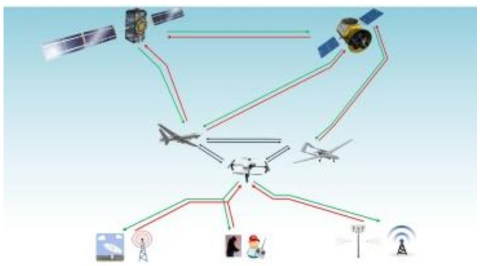
- Meteorological Images supply by HDSDR

4.10.3 Fire Detection System 2020/2021

- This System Specification Document describes a distributed system for fire detection in rural and agriculture areas. On one satellite there will be inscha Allah also a X-Ray Astronomy sensor, which data is sent via a transmitter online to earth.
- System Configuration







4.10.4 Payload X Ray Sensor

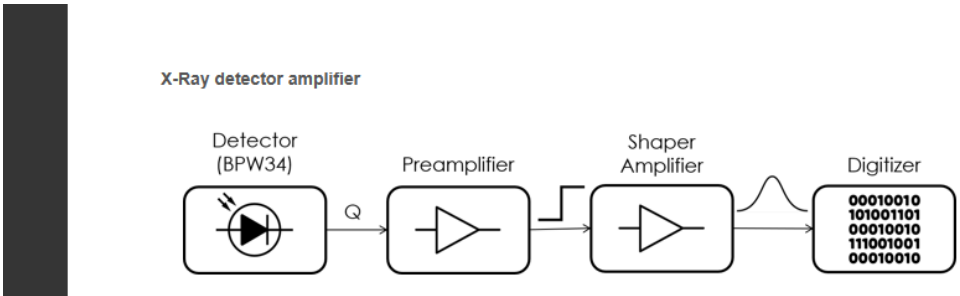
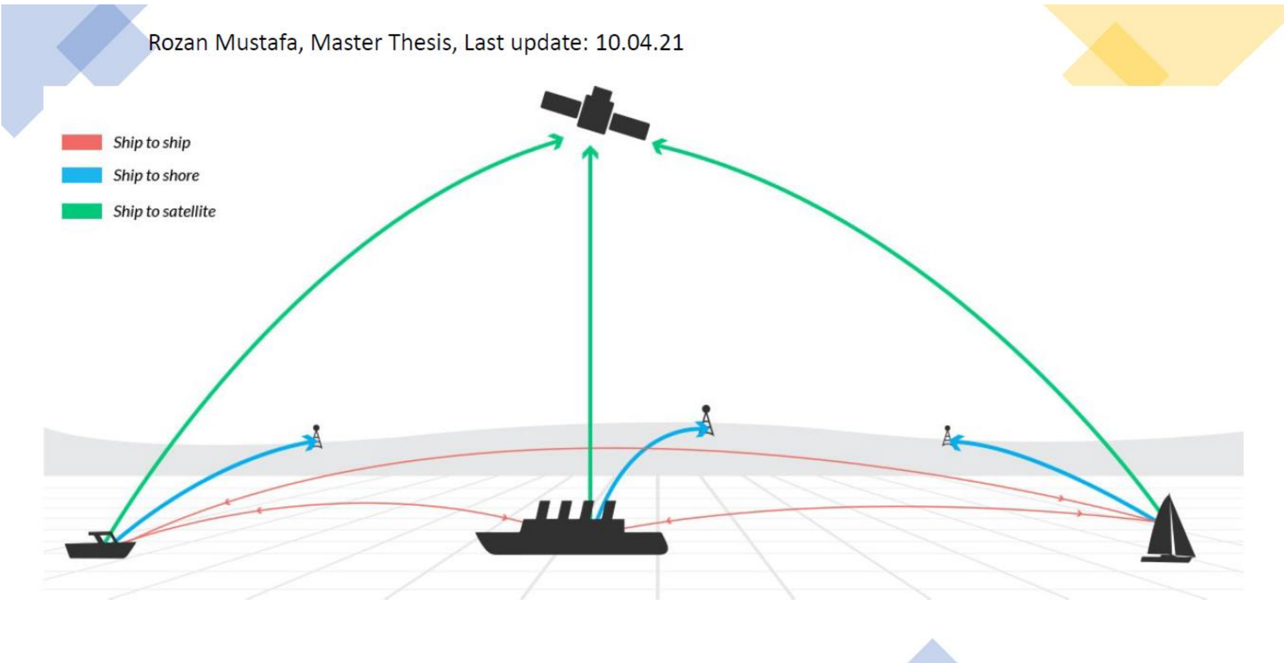


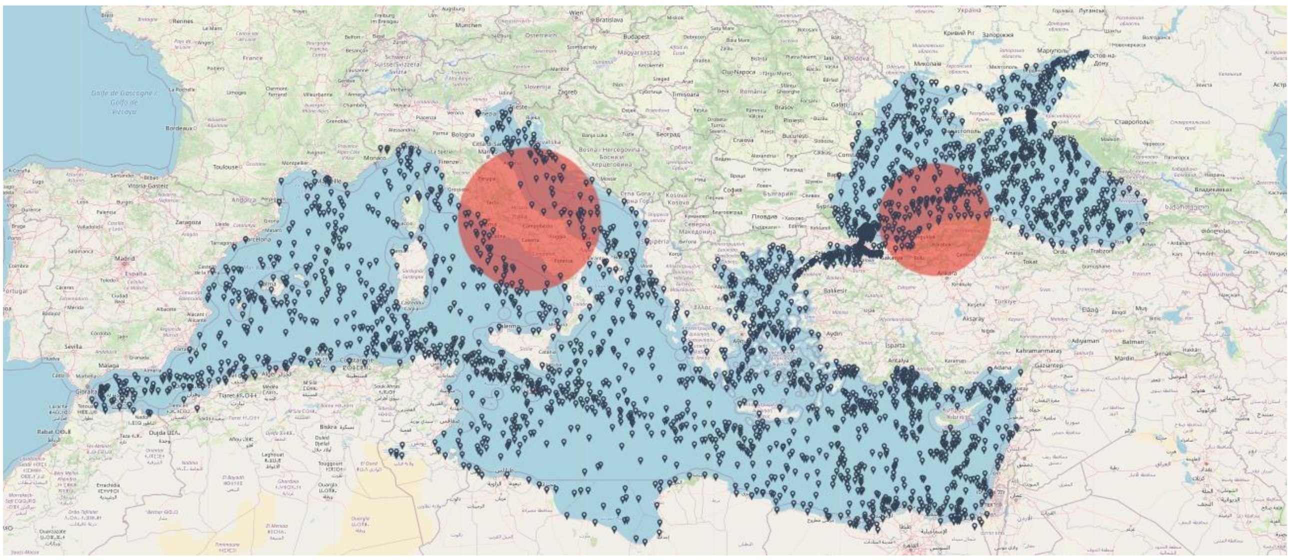
Figure 10. Silicon Detector Amplification Stages

4.10.5 Space Radiation detection

PowerPoint Sunusu (aecenar.com)

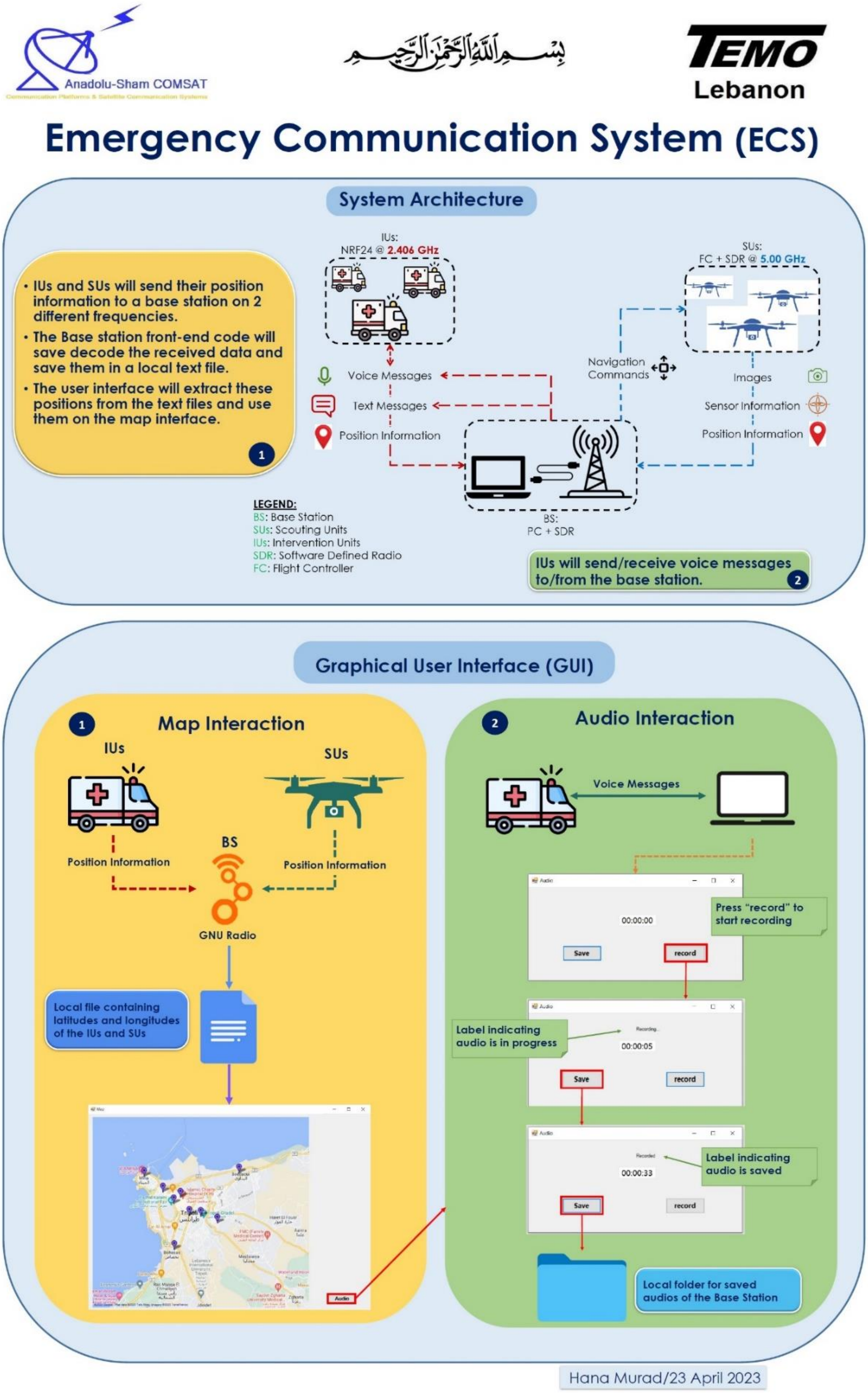
4.10.6 AIS



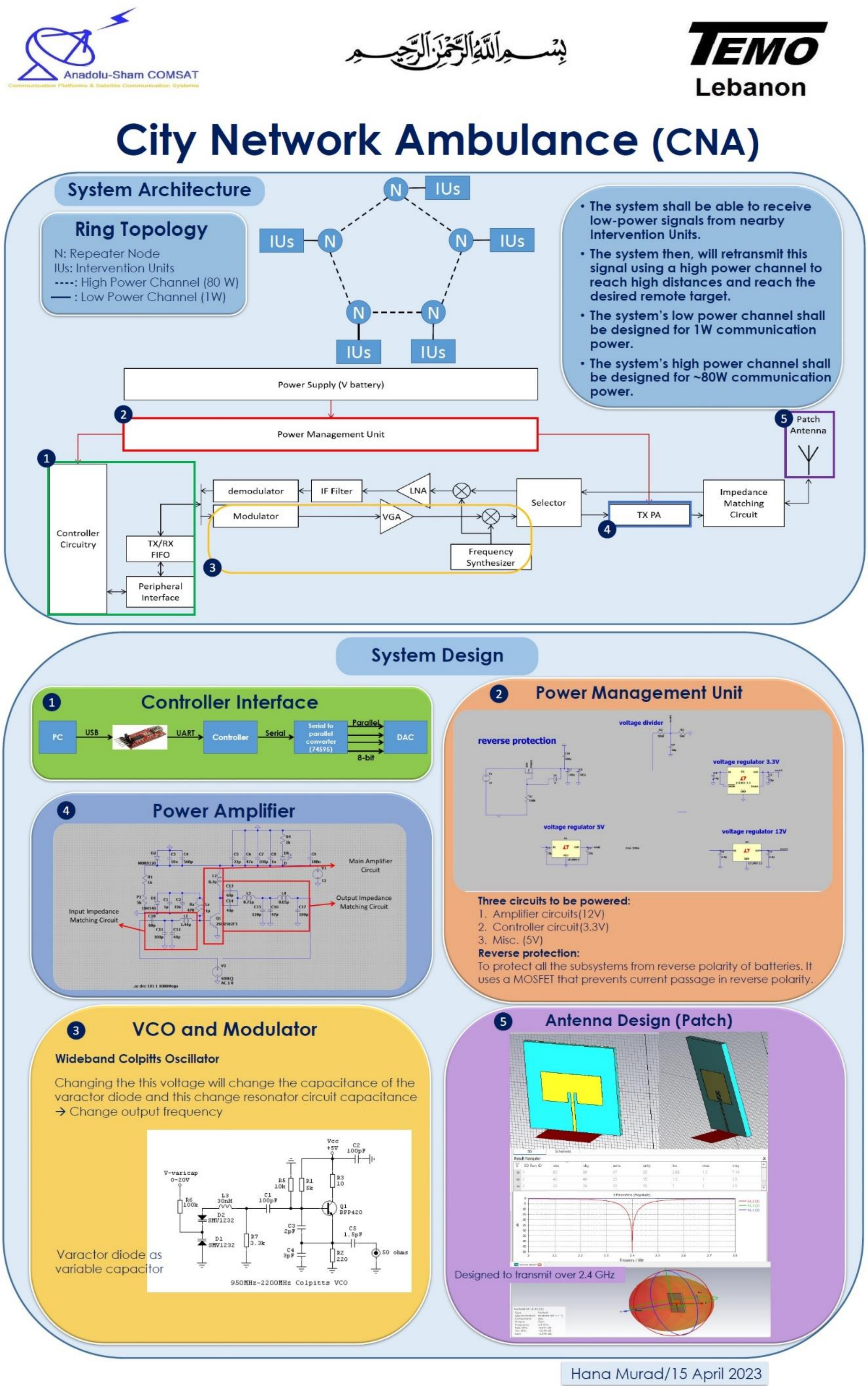


4.11 Business Plan/Marketing: Plans for different Projects (to be presented at IMEX, Istanbul 14.-17.9.23)

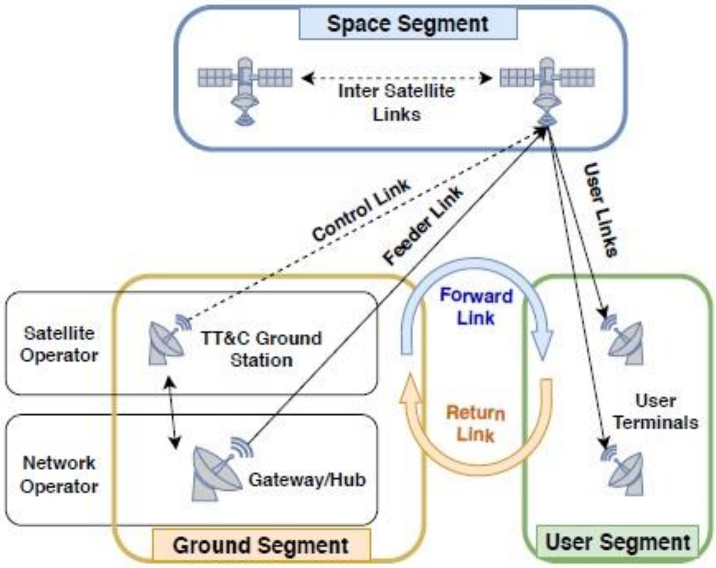
4.11.1 ECS



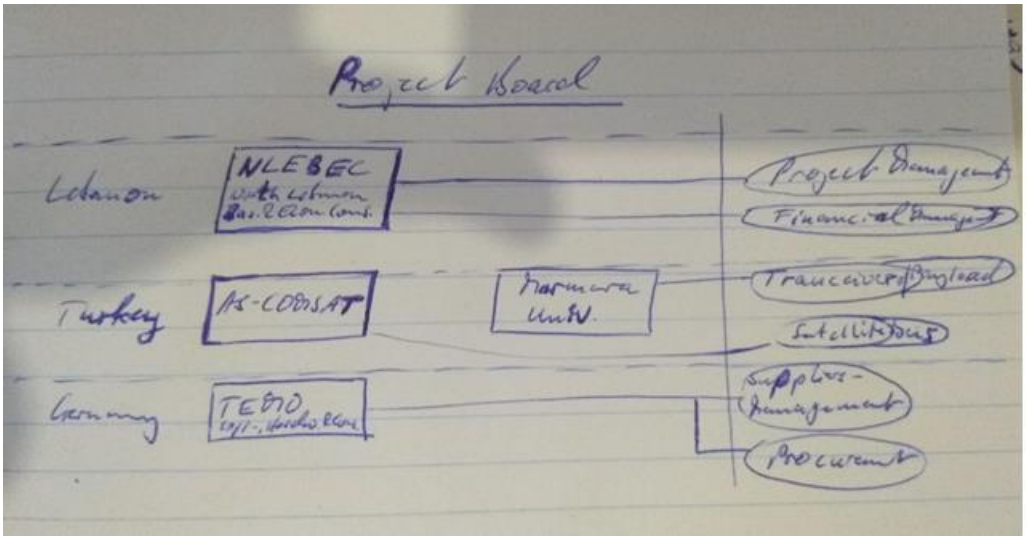
4.11.2 CNA


Anadolu-Sham COMSAT
Communication Platforms & Satellite Communication Systems
بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ
TEMO
Lebanon
City Network Ambulance (CNA)
System Architecture
Ring Topology
N: Repeater Node
IUs: Intervention Units
----: High Power Channel (80 W)
——: Low Power Channel (1W)
N
IUs
• The system shall be able to receive low-power signals from nearby Intervention Units.
• The system then, will retransmit this signal using a high power channel to reach high distances and reach the desired remote target.
• The system's low power channel shall be designed for 1W communication power.
• The system's high power channel shall be designed for ~80W communication power.
Power Supply (V battery)
Power Management Unit
Patch Antenna
Controller Circuitry
TX/RX FIFO
Peripheral Interface
demodulator
IF Filter
LNA
Modulator
VGA
Frequency Synthesizer
Selector
TX PA
Impedance Matching Circuit
System Design
Controller Interface
PC
USB
UART
Controller
Serial
Serial to parallel converter (74595)
Parallel
8-bit
DAC
Power Management Unit
reverse protection
voltage divider
voltage regulator 3.3V
voltage regulator 5V
voltage regulator 12V
Three circuits to be powered:
1. Amplifier circuits(12V)
2. Controller circuit(3.3V)
3. Misc. (5V)
Reverse protection:
To protect all the subsystems from reverse polarity of batteries. It uses a MOSFET that prevents current passage in reverse polarity.
Power Amplifier
Main Amplifier Circuit
Output Impedance Matching Circuit
Input Impedance Matching Circuit
VCO and Modulator
Wideband Colpitts Oscillator
Changing the this voltage will change the capacitance of the varactor diode and this change resonator circuit capacitance
→ Change output frequency
V-varicap 0-20V
Vcc +5V
BFP420
50 ohms
950MHz-2200MHz Colpitts VCO
Varactor diode as variable capacitor
Antenna Design (Patch)
Designed to transmit over 2.4 GHz
Hana Murad/15 April 2023

4.11.3 AS COMSAT 1 LEO Communication System



4.12 AS COMSAT Staff

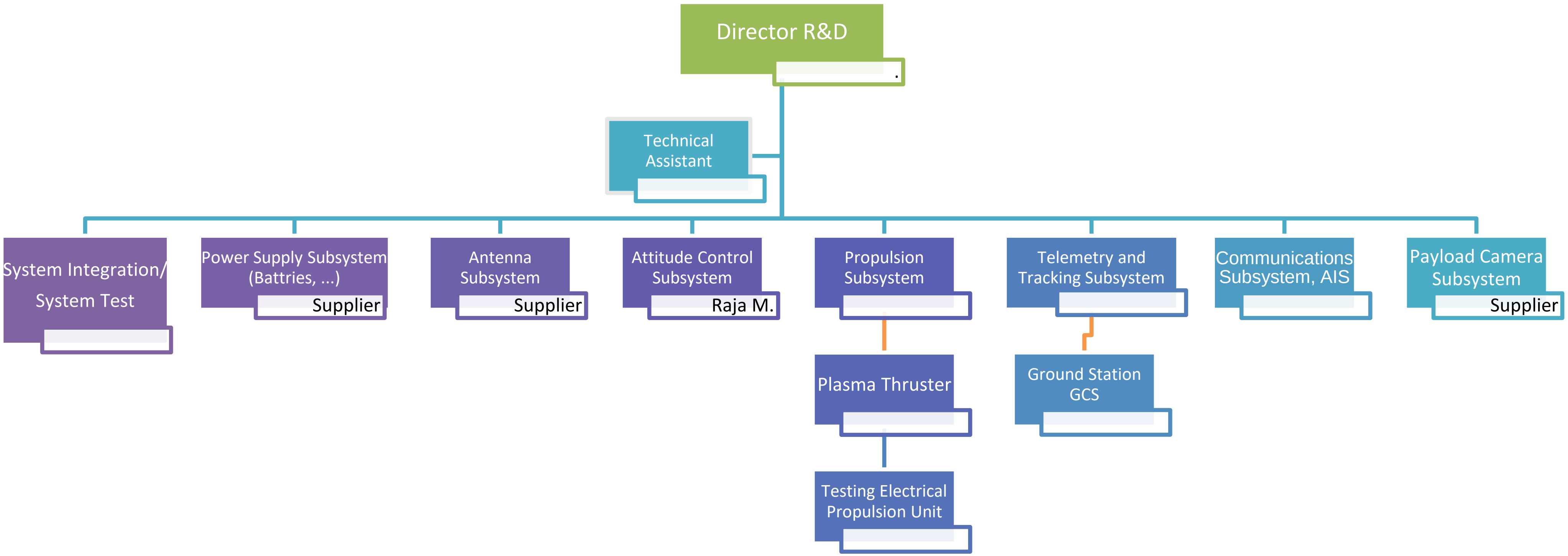


Launching supplier: to be defined

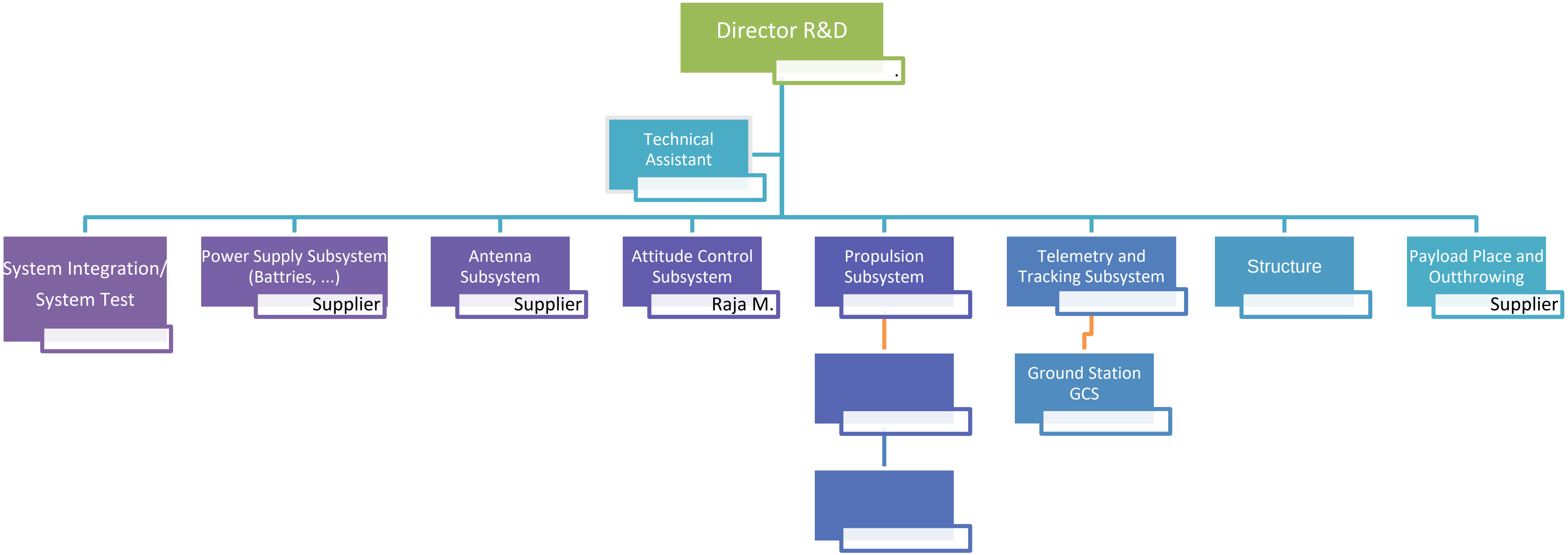
4.12.1 Salary list for beginners at AS-COMSAT

	Salary for fulltime (160h/month)	
Worker		
Specific worker (Welder)		
Student without Bachelor/Licence		
Bachelor Holder		
Master Holder	500 $ (in Lebanon)	

4.12.2 Organizational chart for AS COMSAT 1 satellite development



4.12.3 Organizational chart for AS COMSAT 1 launcher development



4.13 AS COMSAT 1 (Satellite System)

4.13.1 Summary of System Parts

4.13.1.1 Actuator

MAGNETOMETERS - Magnetometers sense magnetic field strengths and direction. The measurements are compared to the Earth's magnetic field map (which is dependent on the spacecraft position) to determine the attitude. Moreover, it can only be used at low altitude orbits, where the magnetic field is strong enough.

4.13.1.2 How communication is done

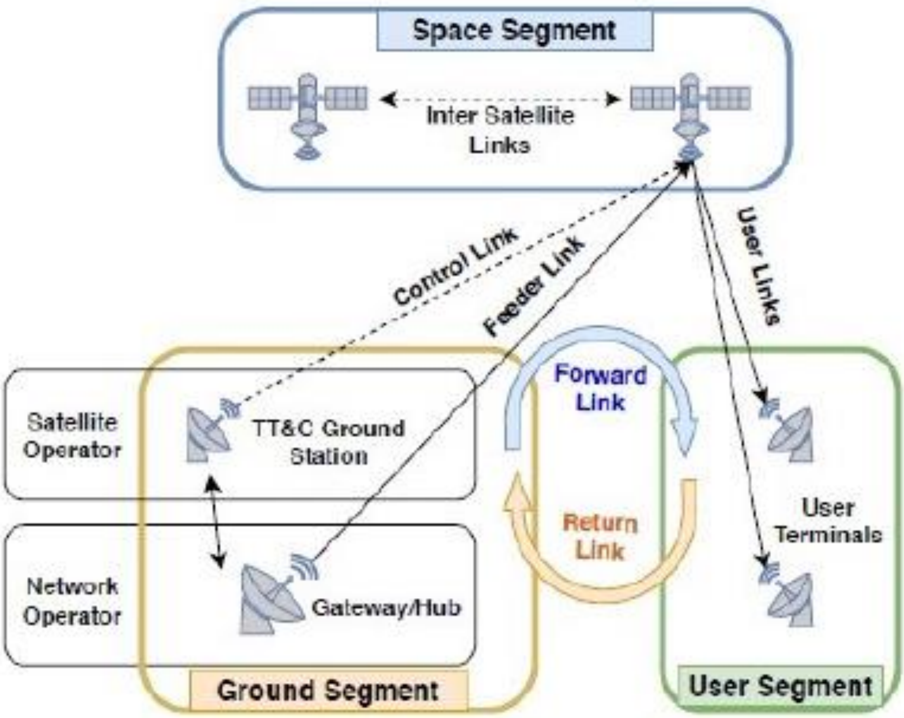


Figure 1: Communication and TT&C

4.13.2 Physical System Overview Example: the IRVINE-03 Education satellite

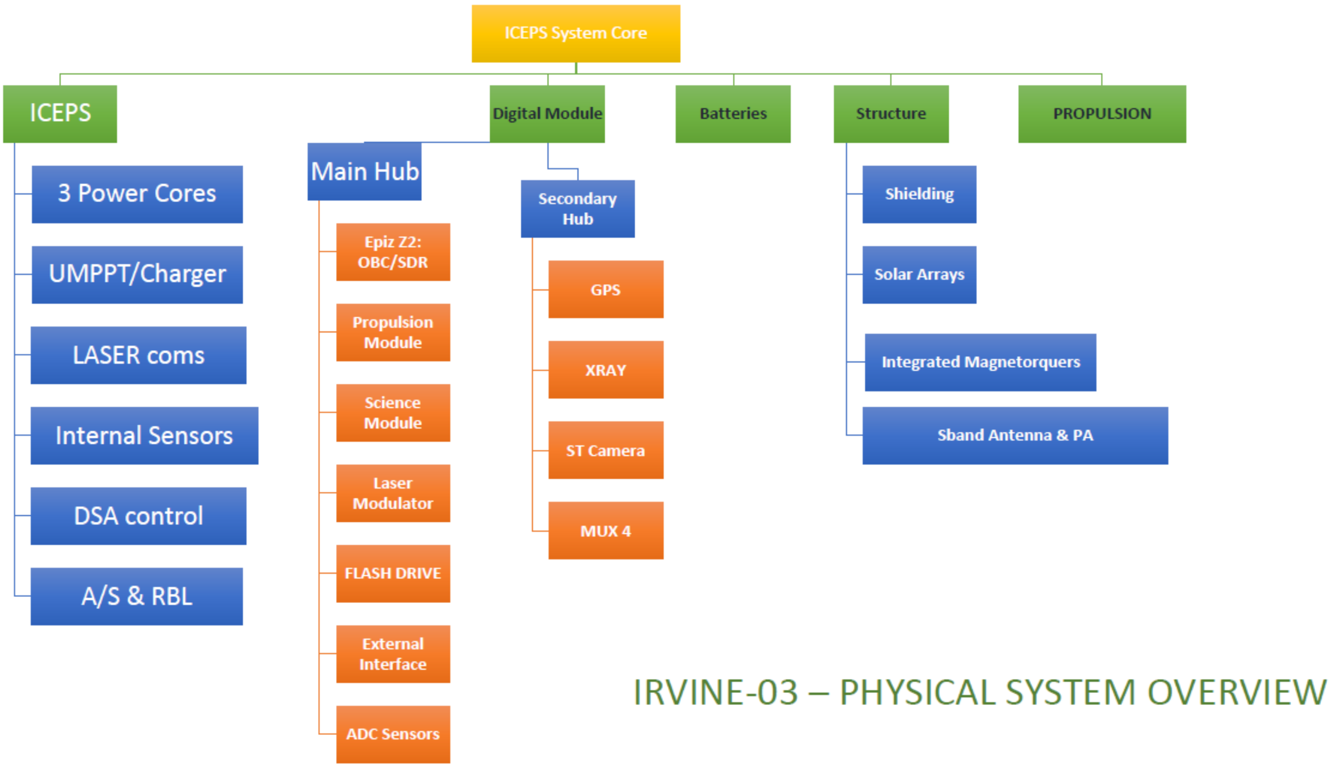


Figure 2: physical overview

- PA: (power amp) amplifies when transmitting.
- LNA: (low noise amp) amplifies when receiving.
- both sit between circuitry and antenna.
- for duplexed signal, passive duplexer shifts between the two on Rx/Tx

4.13.3 On-Board Computer (Raspberry Pi)

On this computer the NASA core flight system shall be implemented. System design: Jana Othman (Internship AECENAR July-Aug 2021).

4.13.3.1 OBC System

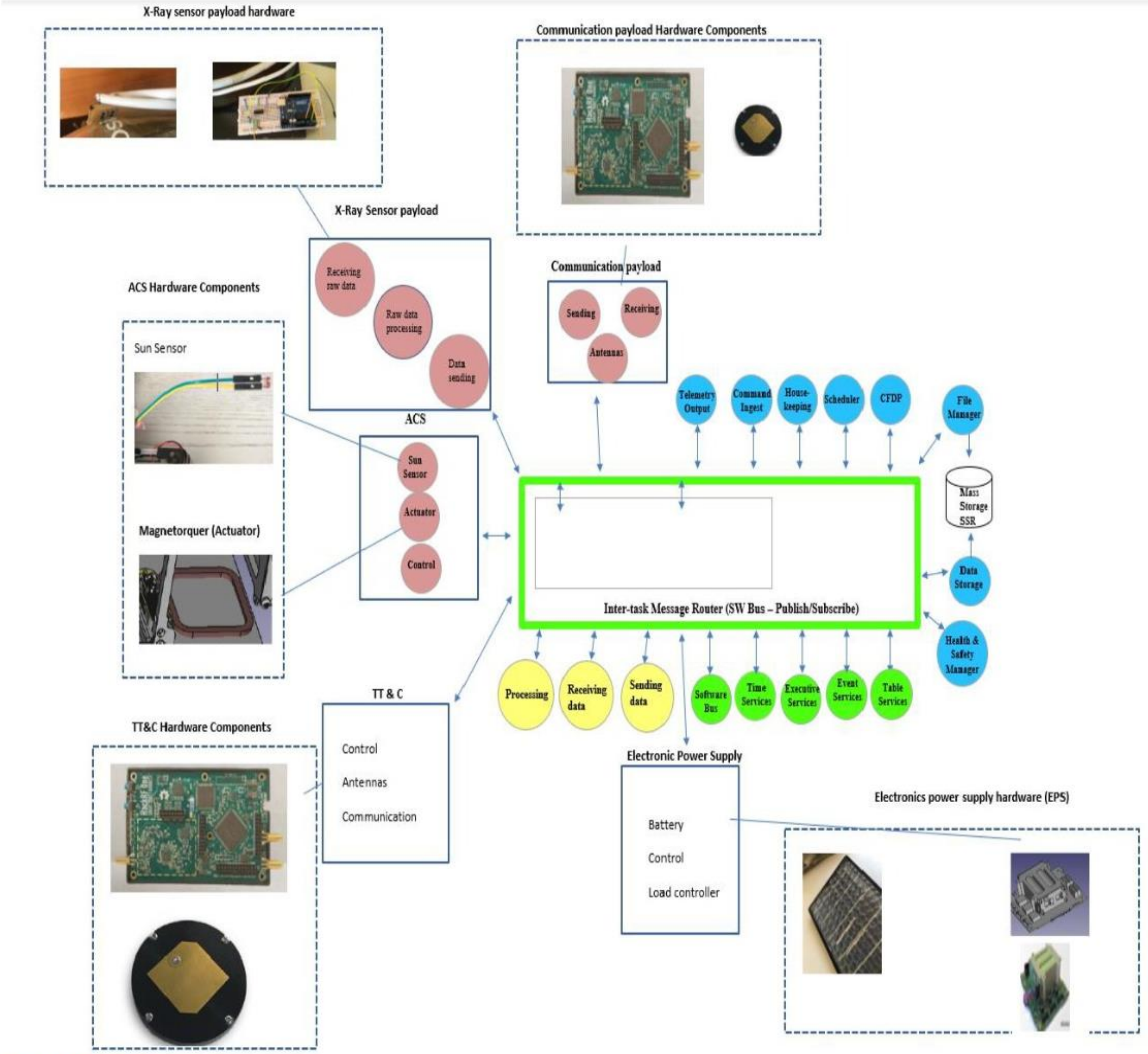


Figure 9: System Design

Legend:

: Components of a block

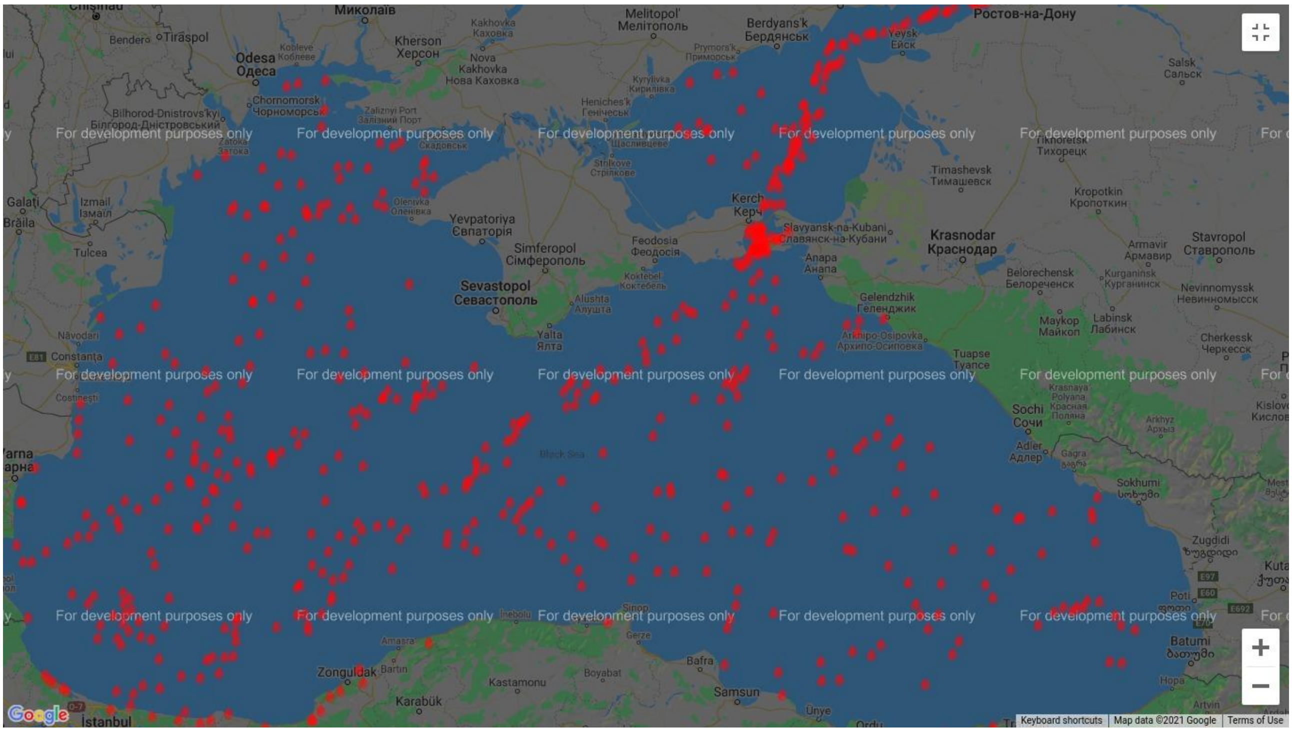
: CFS Configurable Applications

: cFE Core Services

: X-Ray Specific Apps

As payload also the AIS system shall be implemented (Rozan Mustafa as master thesis at Marmara Univ.)

4.13.3.2 AIS system on the OBC



Tasks:

- Satellite Footprint investigation to know how many visible satellites we need for AIS for black sea

Result: Requirements for satellite system:

- Orbit height: 500-600 km (over 600 km we get problems with AIS signals)
- Sun-synchronous orbit (restriction from launcher)
- Inclination: ?

Result from Rozan Oct 21:

>10 satellites needed

Height over earth: about 650 km

Period: 97.72 min.

Orbit ellipse axis a = 7027.748 km

Inclination: 100°-127°

(sun-synchronous orbit)

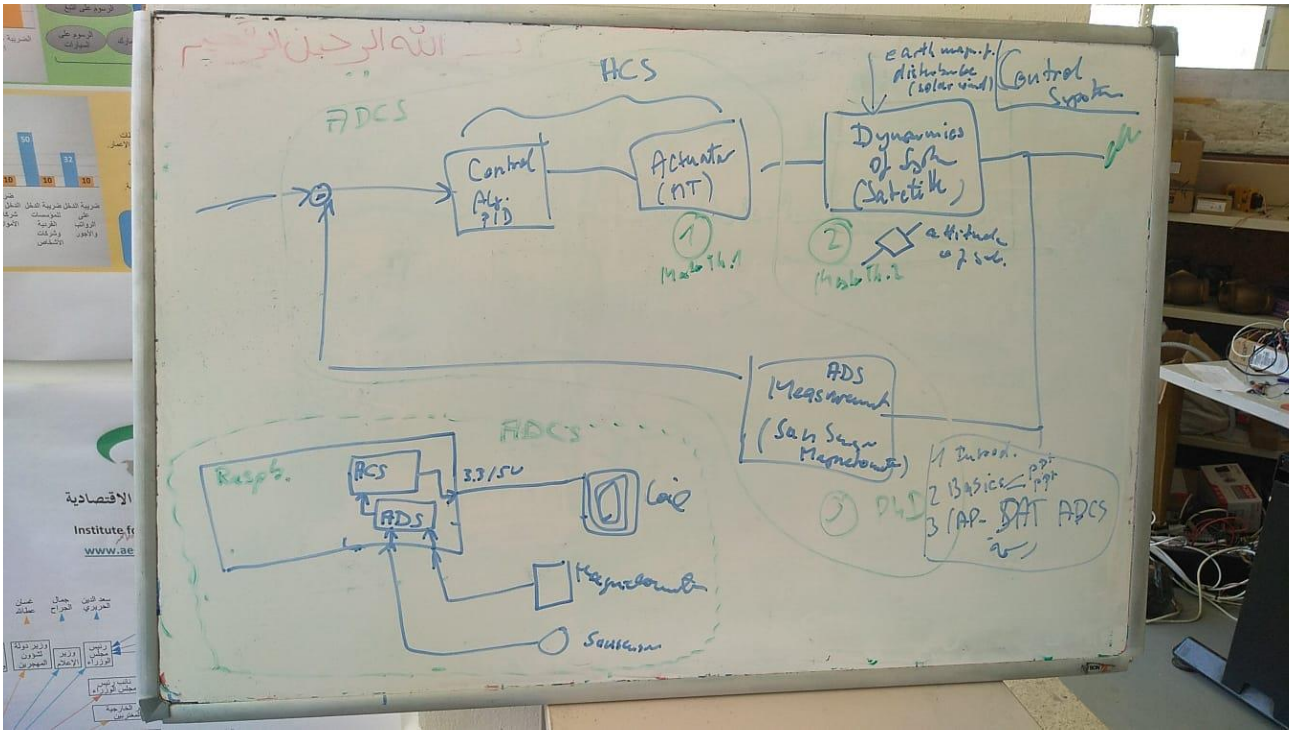
4.13.4 Attitude Control System (ACS) (Responsible: Raja)

Sensors: Sun sensor, IMU (see FCS of TEMOLeb-Mintad 2018 in TEMOLeb-Mintad Final Report[2])

Actuators: Magnetorquer

[2] See [2]

4.13.4.1 System Concept for Attitude Determination and Control System (ADS, ACS)



4.13.5 X-Ray Detector (Sensor)

4.13.5.1 Silicon Detector Amplification Stages

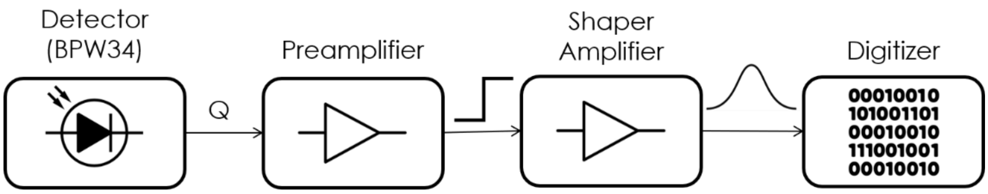
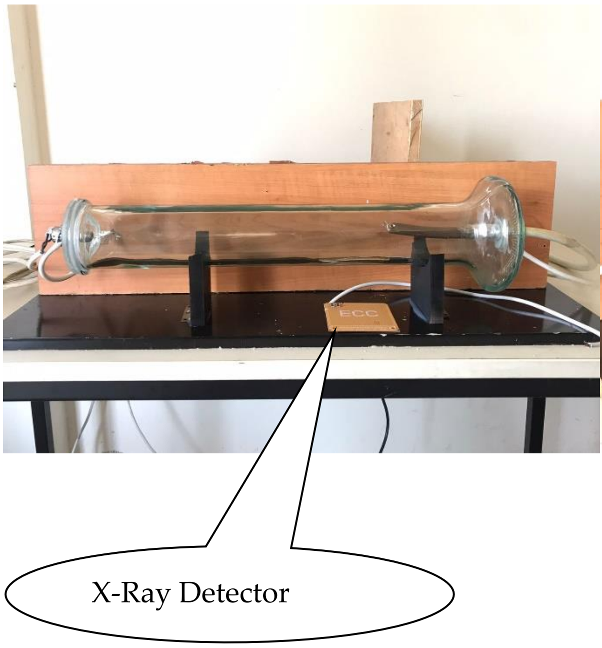
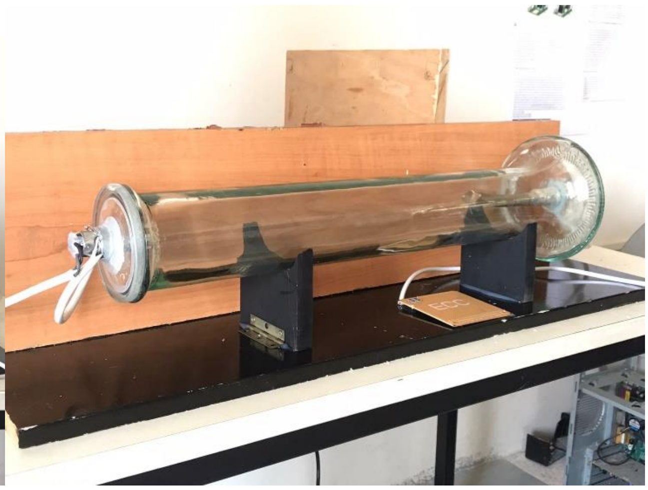


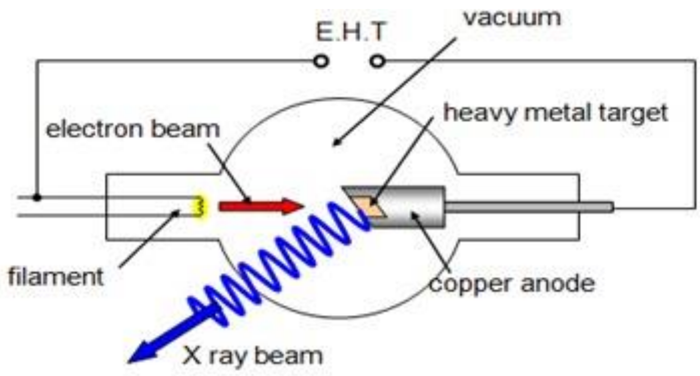
Figure 3. Silicon Detector Amplification Stages

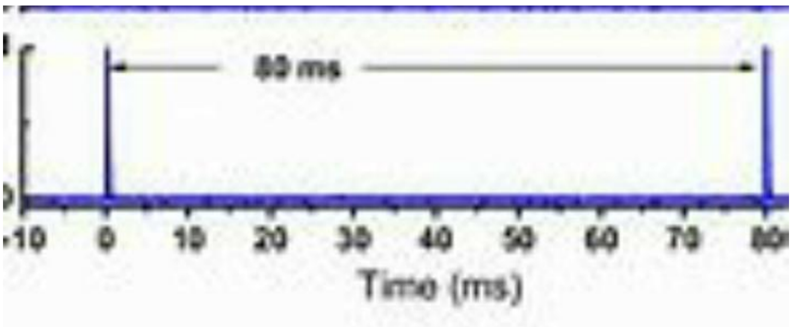
4.13.5.2 Testing X-Ray Sensor with e-beam on cupper anode



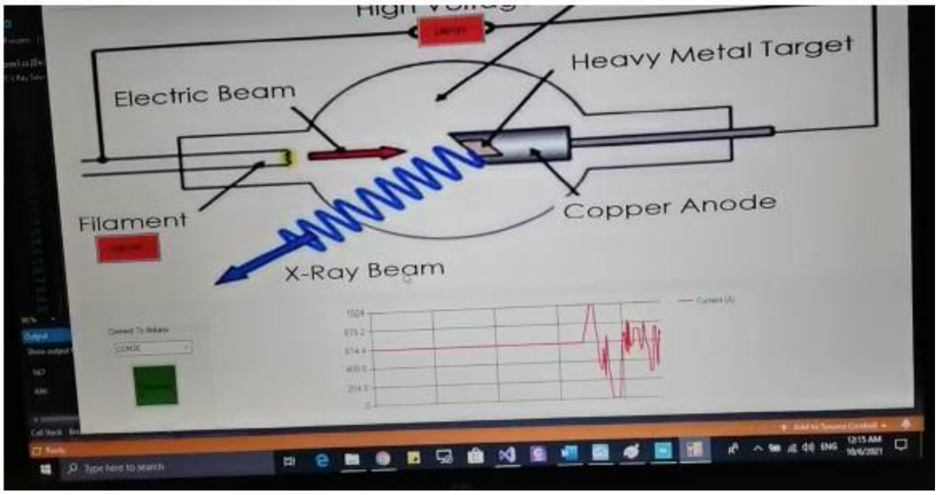


4.13.5.3 Specification of GUI:





4.13.5.4 Implementation of GUI:



4.13.5.5 Realization of Field Control Unit:

Arduino board with RF module (wireless connection to GUI computer)

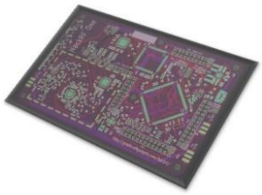
4.14 Telemetry, Tracking & Control (TT&C) on board

		
On-Board Computer (RaspberryPi)	HackRF SDR Card	Antenna

AECENAR
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region

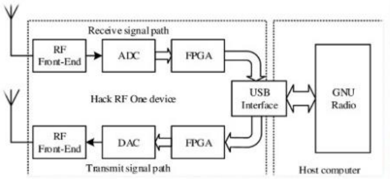
HackRF One

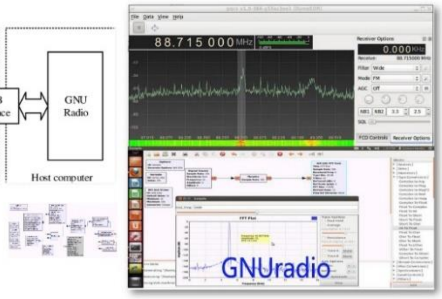
Software Defined Radio
1MHz to 6GHz

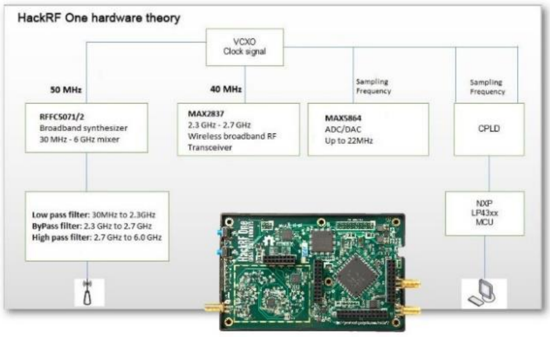


HackRF One from **Great Scott Gadgets** is a Software Defined Radio peripheral capable of transmission or reception of radio signals from 1 MHz to 6 GHz. Designed to enable test and development of modern and next generation radio technologies, **HackRF One** is an open source hardware platform that can be used as a USB peripheral or programmed for stand-alone operation.

Specifications:
- 1 MHz to 6 GHz operating frequency
- Half-duplex transceiver
- Up to 20 million samples per second
- 8-bit quadrature samples (8-bit I and 8-bit Q)
- Compatible with GNU Radio, SDR#, and more
- Software-configurable RX and TX gain and baseband filter
- Software-controlled antenna port power (50 mA at 3.3 V)
- SMA female antenna connector
- SMA female clock input and output for synchronization
- Convenient buttons for programming
- Internal pin headers for expansion
- Hi-Speed USB 2.0
- USB-powered
- Open source hardware







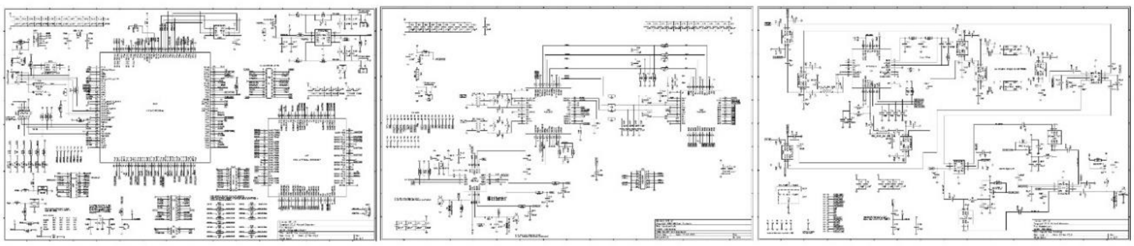
Buttons:
The **RESET** button resets the microcontroller. This is a reboot that should result in a USB re-enumeration.
The **DFU** button invokes a USB DFU bootloader located in the microcontroller's ROM. This bootloader makes it possible to unbrick a HackRF One with damaged firmware because the ROM cannot be overwritten.

To invoke DFU mode: Press and hold the DFU button. While holding the DFU button, reset the HackRF One either by pressing and releasing the RESET button or by powering on the HackRF One. Release the DFU button.
The DFU button only invokes the bootloader during reset. This means that it can be used for other functions by custom firmware.

- The RF switch determines whether to amplify via a 14db amplifier
- The signal is filtered by high pass or loss pass filter
- Signal RFFC5072 chip mixing to 2.6GHz fixed medium frequency
- The firmware supports variable intermediate frequency options: range 2.15 GHz to 2.75 GHz
- Signal into the MAX2837 chip mixing to the baseband, output differential IQ signal (MAX2837 chip can limit the bandwidth of the signal)
- The MAX5864 chip digitizes the baseband signal and sends it to CPLD
- The LPC4320/4330 processor sends the sampled data to computer via USB
- RFFC5072 and MAX2837 are protected in a shield to prevent interference from the outside world or other chips on the board, and to prevent static electricity form penetrating some chips

Parameters:
- Frequency band: 1MHz-6Ghz
- Data bandwidth: 20MHz
- Sampling accuracy (ADC/DAC) : 8BIT
- Sampling speed (ADC/DAC): 20Mbps
- Maximum transmitting power: 10dbm
- 64QAM transmitting EVM: 1.5%
- Complex sampling bandwidth: 20Mhz

Hardware:
- Mixer RFFC5072: 80MHz-4200MHz
- Wireless bandwidth RF transceiver MAX2837: 2.3Ghz-2.7Ghz
- Processor LPC4330: Main frequency 204MHz
- Amplifier MGA-81563: 0.1-6Ghz, 3V, 14dbm



Notes:
- Due to the HackRF clock is passive OC, factory board accuracy usually in 20PPM, 1MHZ frequency offset at about 20HZ, 1GHZ is 20KHZ changes, usually frequency will affect the change with the environment, usually this effect is not high on the requirements of the occasion to have any effect, such as what is the impact of wireless receiving and not after all, HACKRF has 20MHZ bandwidth, but in wireless transmission on different meanings, such as the GPS system, the frequency of 1575.42MHZ if PPM is 20 then the frequency offset 1575.42MHZ+ (1575.42*20=0.0315084MHZ) =1575.42 + 3, so it can not satisfy the applications of GPS, but the PPM 2 accuracy of 0.003MHZ is obviously meet the requirements. TCXO clock module PPM0.1 uses the high accuracy TCXO design, the DIP packing accuracy in 0.1-0.5ppm.
- Some connectors that appear to be SMA are actually RP-SMA. If you connect an RP-SMA antenna to HackRF One, it will seem to connect snugly but won't function at all because neither the male nor female side has a center pin. RP-SMA connectors are most common on 2.4 GHz antennas and are popular on Wi-Fi equipment.
- Board may Need to flash the firmware from before use the following link: https://github.com/sharebrained/portapack-hackrf/releases

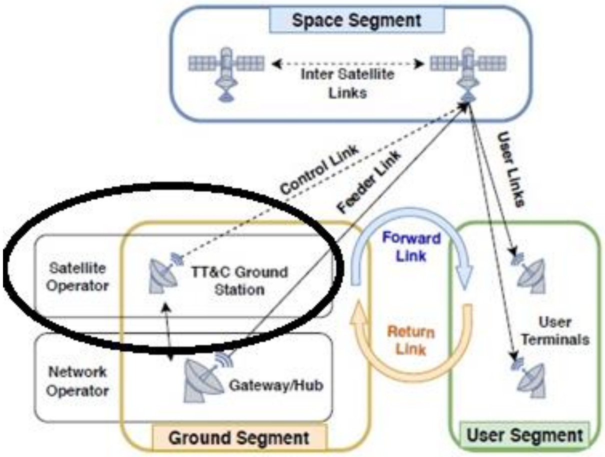
References:
https://greatscottgadgets.com/hackrf/one/, https://greatscottgadgets.com/sdr/, https://github.com/sharebrained/portapack-hackrf, https://github.com/mossmann/hackrf , https://github.com/mossmann/hackrf/wiki/HackRF-One,

ICS
المركز التقني للإتصالات والمعلوماتية

AECENAR - ICS
www.aecenar.com/institue/ics

4.14.1 Telemetry, Tracking & Control (TT&C) Ground Station

4.14.1.1 Requirements (A DESCRIPTION OF A STANDARD SMALL SATELLITE GROUNDSTATION FOR USE BY WMO MEMBERS [4])



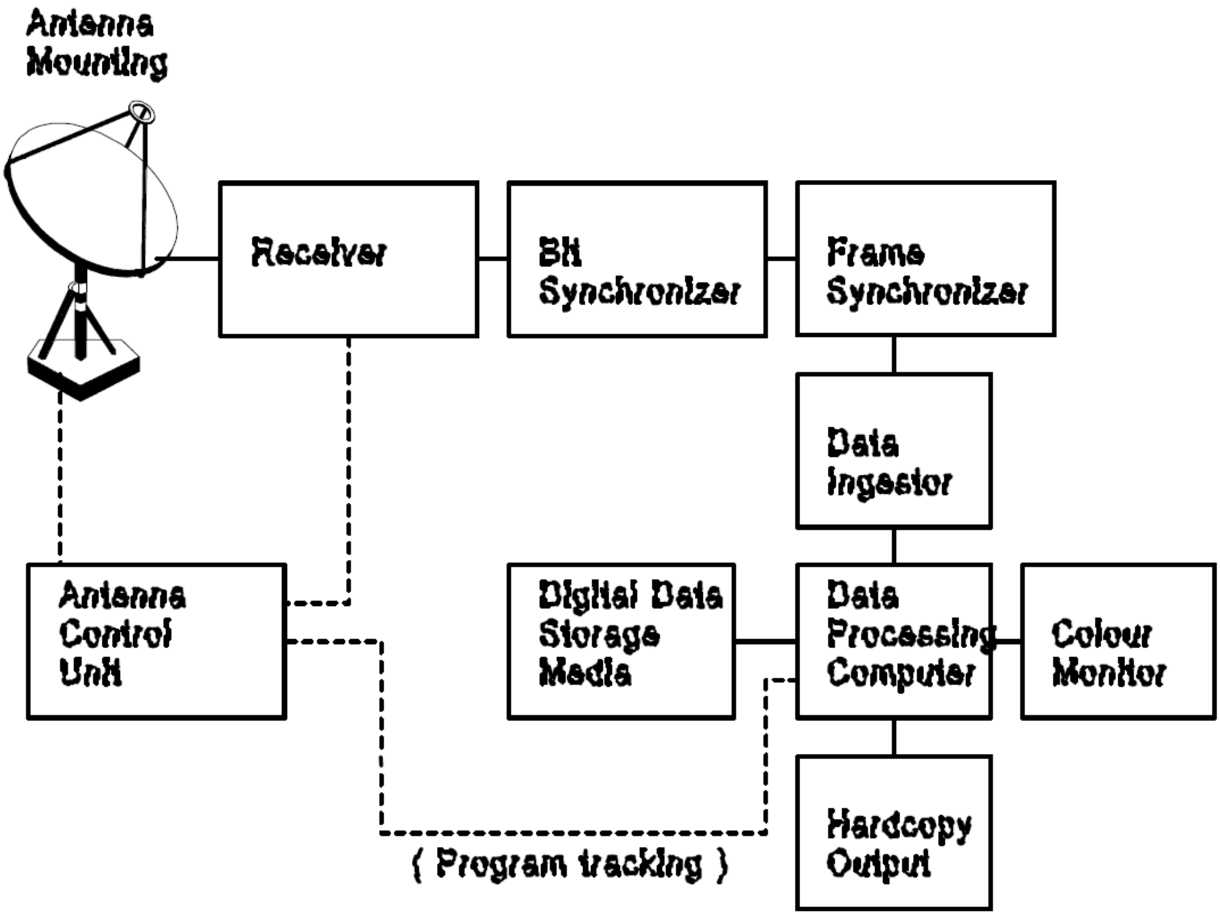


Fig.1 Block Diagram of a small satellite ground station

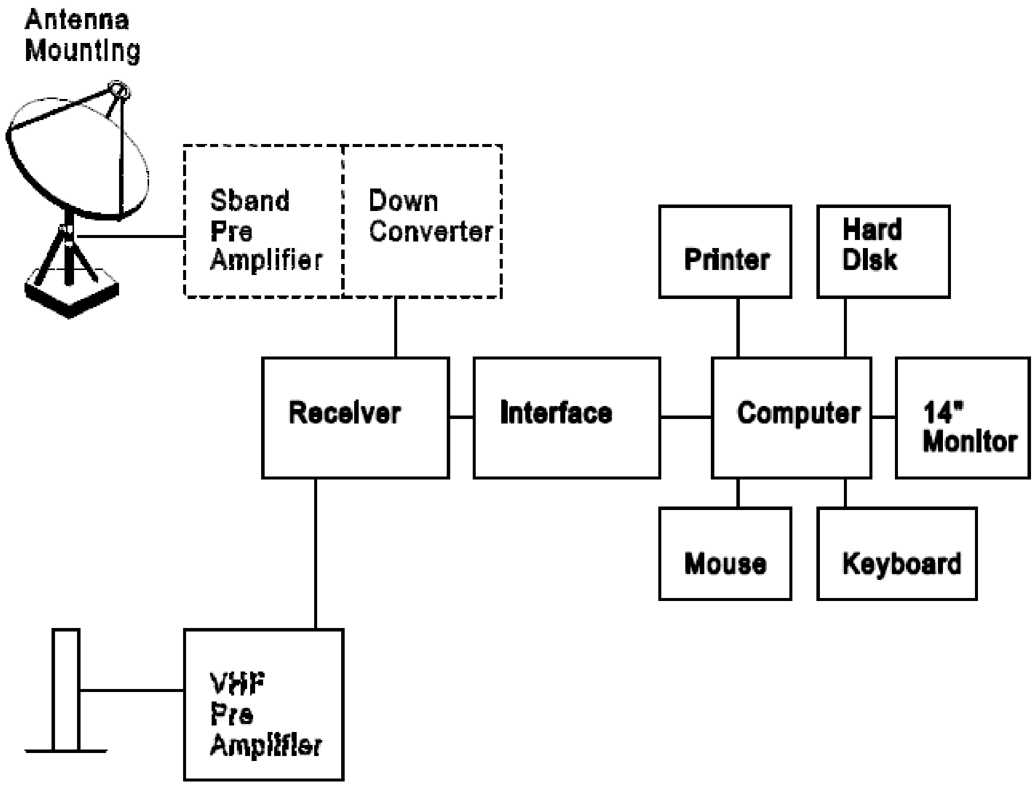


Fig.2 APT/WEFAX Receiving and Processing Station

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

07.10.21

Task (Bachelor Thesis/Student Project)

Development of a Ground Station for a system of 4 LEO satellites (AS-COMSAT_1)

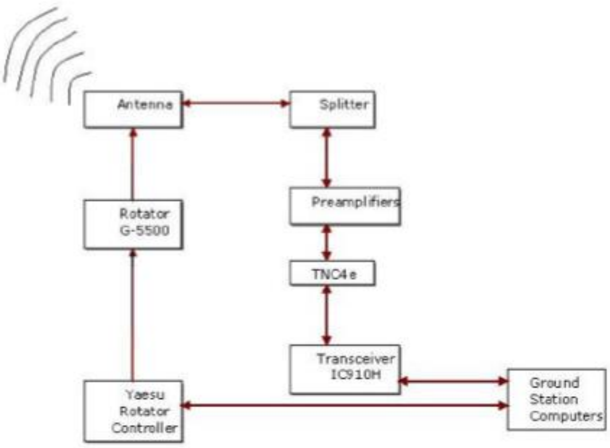
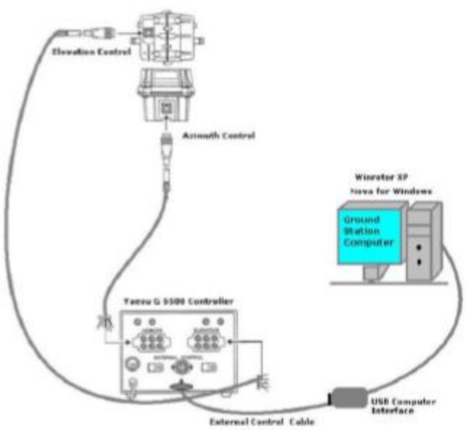


Figure 1.4: Block Diagram of the Ground Station.



Rotor-Controller-Computer Interfacing.

Graphical User Interface

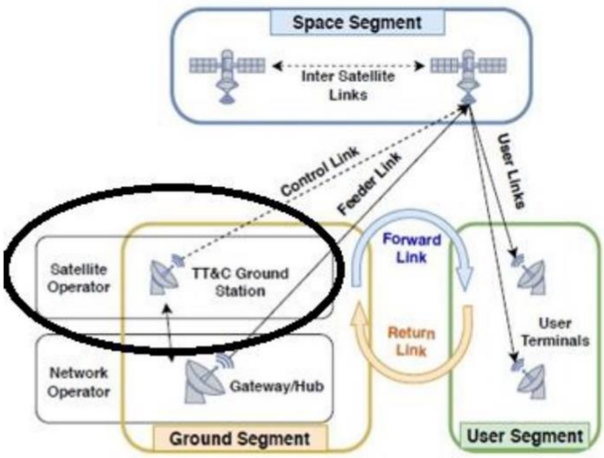
4.14.2 Hardware in the loop (HIL) Test System

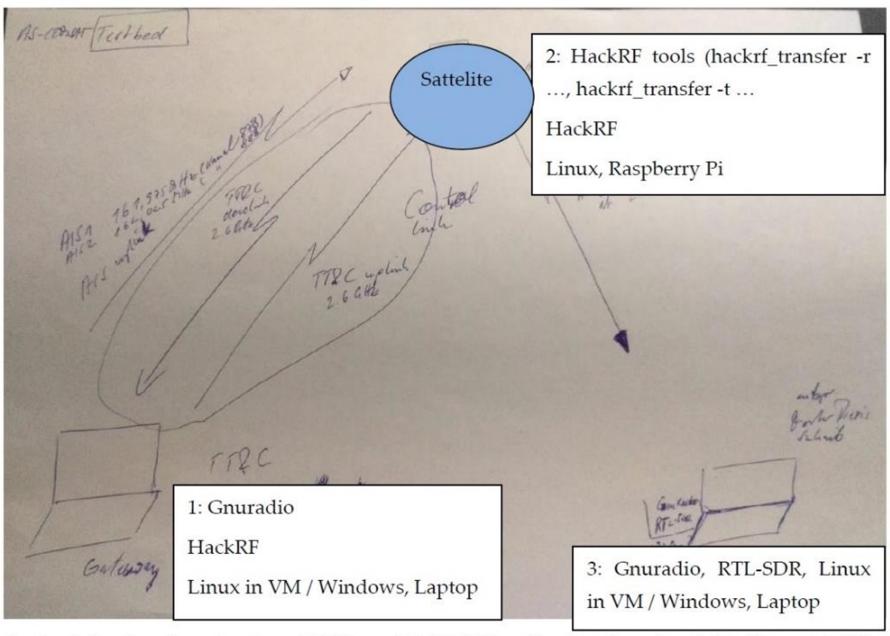
Space Segment:

- Raspberry (Linux, Gnuradio, NASA coreFlightSystem)
- ACS (magnetorquer, IMU from IAP-SAT))
- HackRF
- Antenna
- Structure

Ground Segment:

- Laptop (Windows or Linux, GnuRadio/PothosSDR)
- Antenna
- HackRF





Payload: Sending from 1 to 2 an AIS file on 161.975 MHz – Sending from 2 to 3 this file on 2.6 GHz

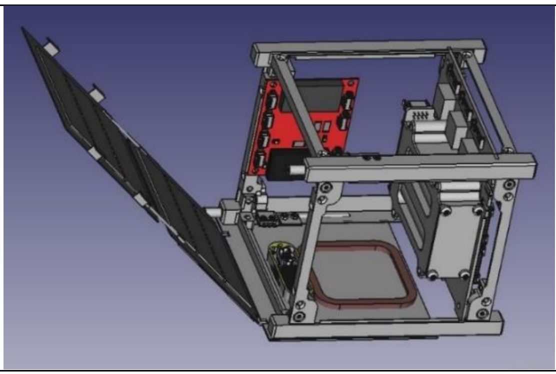
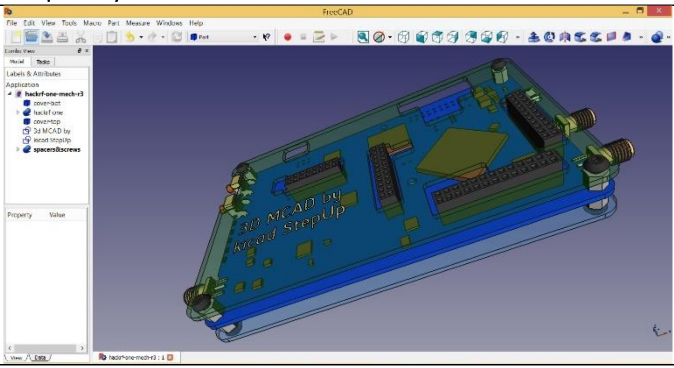
Telemetry, Tracking & Control (TT&C): Sending from 1 to 2 a control command file on 2.6 GHz, sending from 2 to 1 a file with sensor information on 2.6 GHz

4.14.3 Project Documents & Databases for AS-COMSAT-1 (1 1U Satellite) (Last update: 8.4.21)

4.14.3.1 Development Documents

Development Phase	Name of Document	Purpose/Content
Analysis	Fire Detection System Description Presentation	about 130 pages Project Description, Supplier Parts
	Initial Cost Estimation	381,000 $ for satellite and ground station (including launch)
Systemdesign	System Design Document	
Mechanical Design		
Hardware Design	HackRF	
SW Specification	Software Specification Document (SDS)	on OBC and HackRF
SW Design	Software Design Document (SDD)	

4.14.3.2 Mechanical CAD Models

AS-COMSAT- 1 Integration	
Payload 1 (X-Raysensor)	9,5 cm x 2,5 cm x 7 cm
PV Cells and Controller	
Load Controller&Batteries	
Magnetorquer	
Sun Sensor (Photocell)	
On-Board-Computer (OBD)	Raspberry Pi 3
SDR (HackRF) (TT&C and Payload COM)	

4.15 Launch issues

From

AECENAR
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region
www.aecenar.com

IAP
INSTITUTE FOR ASTROPHYSICS

Steps for Launching Satellite

Author:
Siham Aisha

Last Update: 27.12.2021

4.15.1 Offer from Russian Company (Launch with Soyuz) from May 2021

www.gklaunch.ru, info@gklaunch.ru

GK LAUNCH SERVICES

CONFIDENTIAL
#00548 06.05.2021

2 U, 3 kg: 110,000$

Dear Samir! Thank you for your launch quote request. We have prepared a ROM price proposal based on your satellite’s characteristics and selected a suitable mission.

Mission:

> Cluster launch
> Launch period: 2 quarter of 2022
> Orbit: SSO, LTAN 11:00
> Baikonur Cosmodrome
> Primary payload: contracted
> Secondary payload: available

Launch price:
for 3kg satellite is 110000$
The price includes:

1. Program management and program documents;

2. Support of Customer personnel visits;
3. Administrative support to the Customer personnel at GK facilities in accordance with the terms and conditions to be defined in the contract;
4. Program reviews and meetings as may be necessary;
5. Interface Control Document with the results of analyses and reports as may be necessary;
6. Hardware (adapter and separation system, umbilical connector, harness for testing), personnel and equipment for fit-check to be performed at the NPOL facility (SC dummy for testing to be provided by Customer);
7. Hardware (adapter and separation system, umbilical connector, harness), personnel and equipment for integration of the flight SC with the launcher;
8. Hardware, ground support equipment (Space Head Module (SHM), Fregat upper stage, launch vehicle) and personnel for processing of the SHM with SC and execution of launch;
9. Customs clearance of SC/GSE on entry into Russia, customs clearance of GSE on exit from Russia;

10. Tran�ortation of satellite and GSE from the entry/exit port to payload preparation facility and back of GSE, including their customs clearance;
11. Performance of launch campaign and provision of:
Work place in AITB;
Administrative and storage premises;
Power supply;
Provision of logistics to Customer's personnel whilst at the launch site (payment for the services to be made by Customer);
Communications services (international telephone calls to be paid for in accordance with the terms and conditions to be defined in the contra�).
Launch of deployer with CubeSat(s) into the required orbit;
12. Provision to Customer of LV telemetry data confirming the SC separation and initial orbit parameters;
13. Post-launch services;
14. Photographic and video documentation;
15. Linguistic support;
16. Procurement of third party liability insurance for the damage due to the launch activities and support in obtaining the satellite insurance;
17. Launch observation.
You can add insurance of the satellite ground and space related risks and insurance of
the launch service. To do so, call us or write a response letter.
This is a ROM launch price proposal to be finalized after we receive all the documents. Some mission parameters may change.

Standard Milestone payment plan:

	Milestone	Payment, USD (%)	Preliminary Date of milestone completion and submission of invoice
1	Manifesting of Payload on a Launch Mission	15%	Manifesting of Payload
2	IRD review	15%	Completion of IRD review
3	ICD approval	25%	Approval of ICD
4	Ground Tests	20%	Completion of ground tests
5	Start of Launch Campaign	15%	Upon Payload arrival at Launch Site
6	Successful launch	10%	L+2 weeks

The proposal is valid for 20 work days.

GK Team
Contact us at:
+7 (495) 150-44-71
sales@gklaunch.ru

4.15.2 Companies launch satellite

4.15.2.1 GK laucnh (Russian)[3]

GK Launch Services is a company established by Glavkosmos (a subsidiary of ROSCOSMOS State Space Corporation) and International Space Company Kosmotras. GK Launch Services is an operator of Soyuz-2 commercial launches from the Russian spaceports.

The key targets this joint venture aims at include advancing of commercial launch services, promotion of Russian launch vehicles on the world market, and strengthening the positions of Russia as the most competitive launch service provider.

A solid competence of the two partners facilitates achieving these ambitious goals. Glavkosmos has been participating in global international space projects over 30 years and has already had experience in provision of launch services with Soyuz-2 rocket. Kosmotras has lofted over 100 payloads within 22 commercial launches.

4.15.2.2 Swarm company (U.S)[4]

Swarm Technologies, Inc. is a private company building a low Earth orbit satellite constellation for communications with Internet Of Things (IOT) devices using a Store and forward design. An early investor was Craft Ventures. On July 16, 2021, Swarm entered into an agreement to become a direct wholly-owned subsidiary of SpaceX.[2]

They have an Federal Communications Commission (FCC) licence for low bandwidth communications satellites in low Earth orbit.[3]

In 2018 Swarm became the first ever company found to have deployed satellites without regulatory approval after an FCC investigation into the startup's launch on an Indian PSLV rocket of its first four picosatellites in January that year.[4]

By December 2020, Swarm had launched 9 test satellites and 36 of a planned 150 low Earth orbit satellites to provide communication with IOT devices.[5]

In February 2021 Swarm announced that its commercial services were now live using 72 commercial satellites providing its global low cost data to customers.[6]

The Swarm Tile is its dedicated satellite two-way data modem designed to be low energy and embedded on the PCB of third party products. Other products include a data plan and development kit.[7]

4.15.2.3 Exolaunch company (Germany)[5]

The protocol of launching is:

[3] http://gklaunch.ru/en/

[4] https://en.wikipedia.org/wiki/Swarm_Technologies

[5] https://exolaunch.com/

1) LAUNCH PLANNING

Every mission is unique. We listen to you and offer solutions that will enable the successful launch of your satellite. With precision, knowledge and expertise, we make the complex work of your specific campaign a simple and affordable experience. No stress.

2) MISSION MANAGEMENT

Next, we produce your event. We provide technical management of the satellite adaptation to a launch vehicle, interface control document development, mission analysis delivery, and launch schedule coordination.

3) SEPARATION SYSTEMS

EXOpod deployers for cubesats and CarboNIX, the shock-free separation systems for microsatellites, are designed and produced by Exolaunch to support your mission needs. Later, we adapt the deployment systems to the launch vehicle to safely deliver your satellite into its target orbit.

4) DEPLOYMENT SEQUENCER

EXObox is essential for smallsat cluster launches. It is a unique, highly reliable and modular deployment sequencer to manage the deployment of up to 50 satellites with just one EXObox unit. It will precisely and safely separate small satellites into their target orbits.

5) LICENSING

In this phase, we handle the complex legal and regulatory support documents that are required for launch. Your desk is now clean!

6) ENVIRONMENTAL TESTING

Our fully-fledged environmental testing services are tailored to the requirements of any launch vehicle, ensuring your satellite has made the grade. All of the tests are performed in Berlin, and yEnvironmental tests profiles and types:

a. Vibration and shock testing

We provide a full range of mechanical testing to cover the qualification, proto-flight or acceptance requirements of all launch vehicles.

b. Thermal and vacuum testing

Thermal cycling and thermal vacuum testing is available to meet your mission requirements. Whether testing survivability limits or simply performing a vacuum bake-out, we can perform the tests that fit your needs.

c. Qualifation and acceptance test

Qualification of your design based on the composed mechanical loads in order to meet the requirements of most common launch vehicles. Tailored acceptance testing profiles of your flight models to the launch vehicle of your choice, ensuring the function of your spacecraft while increasing confidence in its reliability.

d. Test profiles design

We offer support to develop individual specifications for mission tailored test profiles with optimized loads and durations. Inclusive our acceptance guarantee of the test profiles by the launch authority.

e. Adapters & additional hardware

We offer TestPods and other test fixtures with the test interfaces identical to the launch vehicle interfaces, ensuring the validity of all test results. ou are offered our cleanroom for satellite checkouts.

7) SHIPMENT

Our expertise in global logistics and experience with customs clearance allows us to process worldwide shipping of payloads and equipment in a safe, convenient and timely fashion. We will ensure your satellite reaches the launch site safely and without hassle.

8) INTEGRATION SERVICES

When your satellite meets the launch vehicle. We will seamlessly integrate your satellite with the support hardware onto its launch vehicle.

9) LAUNCH

The launch vehicle has left Earth. Shortly afterwards, we receive a positive signal that your satellite is in orbit and is ready to start changing the world.

4.15.2.4 Gunter's space page[6]

The four **SpaceBEE**, formerly known as **BEE**s (**Basic Electronic Elements**), picosatellites, built to the 0.25U CubeSat form factor are to demonstrate two-way satellite communications and data relay for Swarm Technologies Inc.

The mission is to test the world's smallest two-way communications satellites to serve as a cost-effective low-data rate Internet of Things (IoT) network connectivity solution for remote and mobile sensors. The initial experimental space deployment is comprised of four satellites, each with a 1/4U form factor employing radar signature enhancement technology, which enables them to be passively tracked, and using VHF band frequencies for communications. There will also be an experimental deployment of ground stations for communications with the space units.

The mission is to demonstrate the capabilities of these picosatellites for serving low data rate communication relays for remote sensors and data collectors. Experimental operations is scheduled to begin upon launch for a period of at least 6 months and up to 2 years

The tiny satellites have very small radar cross section, which might complicate the tracking. Therefore they featured a GPS device in each satellite that would broadcast its position on request. Also the four smallest faces of the satellites are covered with an experimental passive radar reflector

[6] https://space.skyrocket.de/doc_sdat/spacebee.htm

developed by the U.S. Navy's Space and Naval Warfare Systems Command, which according to the FCC application would increase the satellites radar profile by a factor of 10.

The FCC dismissed Swarm's application. Nevertheless, the satellites have been launched, apparently without a valid licence, in January 2018 on an Indian PSLV-XL rocket under the name **SpaceBEE**. The ownership of the SpaceBEEs remained obscure, until in an IEEE Spectrum article the identity of the SpaceBEEs with Swarm's BEE satellites was revealed.

A follow-on mission, SpaceBEE 5 to 8, with larger 1U CubeSats was also not granted a licence after this. Later the licesnce was granted. SpaceBEE 1 to 4 were also granted an operation licence.

The operational 2nd generation SpaceBEE satellites reverted back to the 0.25U form factor.

4.15.2.5 Antrix corporation (India)[7]

I contacted this company and the emails are below:

RE: FW: question for procedure to launch a satellite

November 25, 2021 7:44 am 28 KB
From:

Ganesh Mohan <ganesh_mohan@antrix.co.in>
To:

siham.aisha@temo-group.com

Hi Siham,

Could you please elaborate on what is it that you're looking for in administrative and legal procedures?

I presume all the permits, authorisations and notices of non-opposition including frequency filing / other regulatory mandates would be already done by you, during the course of the project.

We will support in any administrative matters during the import and re-export of the payload and the auxiliaries, including transportation, testing facilities, accommodation of personnel etc during the launch base. Once the satellite is separated in a low earth orbit, the control gets passed over to you. All the other aspects like Indemnity, Insurances etc would be covered in the launch contract and we can discuss over it during the course of execution of the contract.

Thanks

Ganesh

[7] https://www.antrix.co.in/

From: Siham [mailto:siham.aisha@temo-group.com]
Sent: 22 November 2021 15:53
To: Ganesh Mohan
Subject: Re: FW: question for procedure to launch a satellite

Hello,
Thank you for your response.

We need know what is the administrative and legal procedure in details, I mean if we want to keep some legal files we wish to inform us.

I prefer an email contact to be clearly.
our location is in Lebanon - Tripoli - Ras masqa, name of my organization is North Alternative Power departement TemoGroup

Regards

On November 22, 2021 at 11:57:24 am +02:00, Ganesh Mohan <ganesh_mohan@antrix.co.in> wrote:

Hi Siham,

Thanks for your interest in Antrix.

The procedure is that

1. there will be a launch services agreement that we will have to execute, wherein the said satellite will be accommodated as a ride share with one of the upcoming PSLV / SSLV missions.

2. There will be an Interface control document, where all the details of the testing, Dynamic studies, sequencing, power supply etc would be addressed and cleared.

3. The launch would happen from Sriharikotta, where the necessary testing / operations / safety procedures would be happening.

With respect to the pricing, it will vary. May I know where are you based at and the company that you’re working for? We can probably discuss it over a call.

Thanks

Ganesh Mohan

Manager, Antrix Corporation

From: Siham [mailto:siham.aisha@temo-group.com]
Sent: 16 November 2021 15:39
To: sonali@antrix.co.in
Subject: question for procedure to launch a satellite

Hello,

I want to launch a satellite, Could I have a quote of price and what is the procedure to launch it?

1- the time frame it's about in the middle of 2022

2- launch to the same inclination, I mean in the same orbit and the inclination is 100 degre - 127 degre

3- our spacecraft don't have propulsion

4- size of our spacecraft is : 10X10X20 cm 2U for 2 satellites

5- mass is: 3Kg

6- the purpose: small pilot system for communication satellite.

7- altitude: 650 km

8- elliptic semimajor axis a = 7027.748 km

9- orbit heigh: 500- 600 km

Regards,

Siham

Procedure to launch a satellite in India[8,9]

Norms, Guidelines and Procedures for Satellite Communications Announced

The Government has approved a policy that envisages allocation of INSAT system capacity for non-governmental users, registration of Indian satellite systems by private Indian companies and limited use of foreign satellites in special circumstances. The Department of Space (DOS) will be the administrative ministry in all matters related to satellite systems in India.

As per the policy, the Indian National Satellite System (INSAT) capacity will be made available to non-government (private) Indian Service Providers on a commercial basis subject to availability after meeting the government needs. The DOS will allocate INSAT capacity for private users. DOS may also build capacity in INSAT system for private users on request on commercial basis.

Private Indian companies with a foreign equity less than 74 percent are now allowed to establish Indian Satellite Systems. These companies can submit their applications for registering their satellite systems to the Committee for Authorising the establishment and operation of Indian Satellite Systems (CAISS). The office of CAISS is set up at the SatCom Programs Office at ISRO Headquarters, Antariksh Bhavan, New BEL Road, Bangalore- 560 094. The authorisation to operate the Satellite System and the Orbit spectrum notification/registration will be done by CAISS. However, operating licenses for services to be provided by the Indian Satellite Systems will be issued only by the concerned administrative departments like Department of Telecommunication for telecom services and Ministry of Information and Broadcasting for TV/Radio broadcasting.

Foreign satellites will also have allowed to be used in special circumstances for satellite communication services in India. The service licensing departments may allow the use of foreign satellites only in consultation with the Department of Space. If suitable capacity/capability is available in INSAT or Indian Satellite Systems, operations with foreign satellites will not be permitted. For the use of foreign satellites for Internet Service Provider (ISP) gateways, the existing procedures established by Telecom Commission will apply.

India's Space Policy[10]

Remote sensing

---

8 https://www.isro.gov.in/update/08-may-2000/norms-guidelines-and-procedures-satellite-communications-announced

9 https://www.isro.gov.in/contact-us-0

10 https://www.isro.gov.in/indias-space-policy-0

Recognizing that Remote Sensing data provides much essential and critical information - which is an input for developmental activities at different levels, and is also of benefit to society.

Noting that a large number of users - both within and outside government, use Remote Sensing data from Indian and foreign remote sensing satellites for various developmental applications.

Taking into consideration the recent availability of very high-resolution images, from foreign and commercial remote sensing satellites, and noting the need for proper and better management of the data acquisition/ distribution from these satellites in India.

Recognizing that national interest is paramount, and that security consideration of the country needs to be given utmost importance.

The Government of India adopts the Remote Sensing Data Policy (RSDP) - 2011 containing modalities for managing and/ or permitting the acquisition / dissemination of remote sensing data in support of developmental activities. Department of Space (DOS) of the Government of India shall be the nodal agency for all actions under this policy, unless otherwise stated.

1. For operating a remote sensing satellite from India, license and/ or permission of the Government, through the nodal agency, shall be necessary.
   - As a national commitment and as a "public good", Government assures a continuous and improved observing/ imaging capability from its own Indian Remote Sensing Satellites (IRS) programme.
   - The Government, through the nodal agency, shall be the sole and exclusive owner of all data collected/ received from IRS. All users will be provided with only a license to use the said data, and add value to the satellite data.
   - Government reserves the right to impose control over imaging tasks and distribution of data from IRS or any other Indian remote sensing satellite, when it is of the opinion that national security and/ or international obligations and/ or foreign policies of the Government so require.
   -
2. For acquisition/ distribution of remote sensing data within India, license/ permission from the Government of India, through the nodal agency, shall be necessary.
   - Government reserves the right to select and permit agencies to acquire/ distribute satellite remote sensing data in India. DOS shall be competent to decide on the procedure for granting license/ permission for dissemination of such data, and for the levy of necessary fees.
   - To cater to the developmental needs of the country, the National Remote Sensing Centre (NRSC) of the Indian Space Research Organisation (ISRO)/ DOS is vested with the authority to acquire and disseminate all satellite remote sensing data in India, both from Indian and foreign satellites.
     - NRSC shall enter into appropriate arrangements with DOS for acquiring/ distributing data from IRS within the visibility circle of NRSC's receiving station(s).

  - NRSC and/ or Antrix Corporation Ltd., shall be competent to enter into agreements with foreign satellite operator(s) for acquisition/distribution of foreign satellite data in India. However, NRSC will distribute the data as per terms agreed to with Antrix Corporation Ltd.
- NRSC shall maintain a systematic National Remote Sensing Data Archive, and a log of all acquisitions/ sales of data for all satellites.

3. For acquisition and distribution of IRS data for use in countries other than India, the Government of India, through the nodal agency, shall grant license to such bodies/ agencies of those countries as are interested in the acquisition/ distribution of IRS data, as per specific procedures.
   - The Antrix Corporation Ltd. (of DOS) is vested with the authority for receiving the applications for grant of license for acquisition/ distribution of IRS data outside of India; to consider and decide on the granting of license within the policy considerations of the Government, and to enter into licensing agreements with the prospective users on behalf of the Government. Antrix Corporation Ltd. shall also be competent to levy such fees for granting licenses as may be considered appropriate by it. It shall also be responsible, where necessary, for rendering any further help/ guidance needed by the license.
   - The Government reserves right to impose restrictions over imaging tasks and distribution of IRS data in any country when it is of the opinion that national security and/ or international obligations and/ or foreign policies of the Government so require.
4. The Government prescribes the following guidelines to be adopted for dissemination of satellite remote sensing data in India:
   - All data of resolutions up to 1 m shall be distributed on a non-discriminatory basis and on "as requested basis"
   - With a view to protect national security interests, all data of better than 1 m resolution shall be screened and cleared by the appropriate agency prior to distribution; and the following procedure shall be followed:
     - Government users namely, Ministries/ Departments/ Public Sector/ Autonomous Bodies/ Government R&D institutions/ Government Educational/ Academic Institutions, can obtain the data without any further clearance.
     - Private sector agencies, recommended at least by one Government agency, for supporting development activities, can obtain the data without any further clearance.
     - Private sector agencies, recommended at least by one Government agency, for supporting development activities, can obtain the data without any further clearance.
     - Specific requests for data of sensitive areas, by any user, can be serviced only after obtaining clearance from the HRC.
     - Specific sale/ non-disclosure agreements to be concluded between NRSC and other users for data of better than 1 m resolution.
5. This Policy (RSDP-2011) comes into effect immediately, and may be reviewed from time-to-time-by Government.

4.15.2.6 Spaceflight (US)

I contacted spaceflight:

RE: [External] - Sales - Website Submission

November 17, 2021 6:31 am 52 KB

From:

Keiko Nasu <knasu@spaceflight.com>
To:

siham.aisha@temo-group.com

Hello,

Sorry, for some reason your e-mail has been in the spam box and it took me a while to find your response. I deeply apologize for that.
We are just to close the manifest for June 2022 SpaceX rideshare. We need to provide a good portion of deliverables to SpaceX by December 1, but if you will be able to do that, we might be able to launch your satellite with that mission.
The issue is how you would like to separate the 2 satellites. Our Sherpa-OTV could provide in-plane phasing but it's not going to be very economical for 2U satellite (and I heard that our Sherpa capacity is already filled).
We might be able to separate a little bit by delay the second satellite deployment (30min or so at most), but it will not give you a lot of separation.
Please confirm SSO is the orbit you would like to go, and will check other possible launch options.
Let me know if you prefer to have a brief call.
Warm regards,

Keiko Nasu
Business Development, Spaceflight Inc.
KNasu@spaceflight.com
Mobile:+1-206-384-0678

From: Siham <siham.aisha@temo-group.com>
Sent: Monday, November 15, 2021 1:36 AM
To: Keiko Nasu <knasu@spaceflight.com>
Subject: RE: [External] - Sales - Website Submission

CAUTION: This email originated from outside of the organization. Do not click links or open attachments unless you recognize the sender and know the content is safe.

hello,
I'm waiting for response.
another question: what is the procedure to launch a satellite ?

Regards,
Siham

On October 2, 2021 at 9:20:37 am +03:00, Siham <siham.aisha@temo-group.com> wrote:
Hello Keiko,

1- the time frame it's about in the middle of 2022
2- launch to the same inclination, I mean in the same orbit
3- our spacecraft don't have propulsion
4- size of our spacecraft is : 10X10X20 cm 2U for 2 satellites

5- mass is: 3Kg
6- the purpose: small pilot system for communication satellite.

Regards,
Siham

On September 29, 2021 at 1:34:32 am +03:00, Keiko Nasu <knasu@spaceflight.com> wrote:

Hello Siham,

Thank you so much for reaching out to us.

In order to figure out the launch options and pricing, could you provide below?

- What is the time frame you are looking to launch?
- What orbit do you want to launch your spacecraft to?

-> Do you mean to launch to Mid-inc? 45 degrees inclination, Or you would like to launch to the same inclination but do the plane phasing?

- Will your spacecraft have propulsion?
- What is the size of your spacecraft?

->2U x 2 satellites

- What is the mass of your spacecraft?
- What is the purpose of your spacecraft?

Thank you!

Keiko Nasu

Business Development, Spaceflight Inc.

KNasu@spaceflight.com

Mobile:+1-206-384-0678

4.16 Launcher System Company Management like the Spanish Company PLD

Predevelopment see [IAP-Transporter Report 1] (2016), [IAP-Transporter Report 2] (2017), [IAP-Transporter Report 3] (2018). Most basics are described in [IAP-Transporter Report 2].

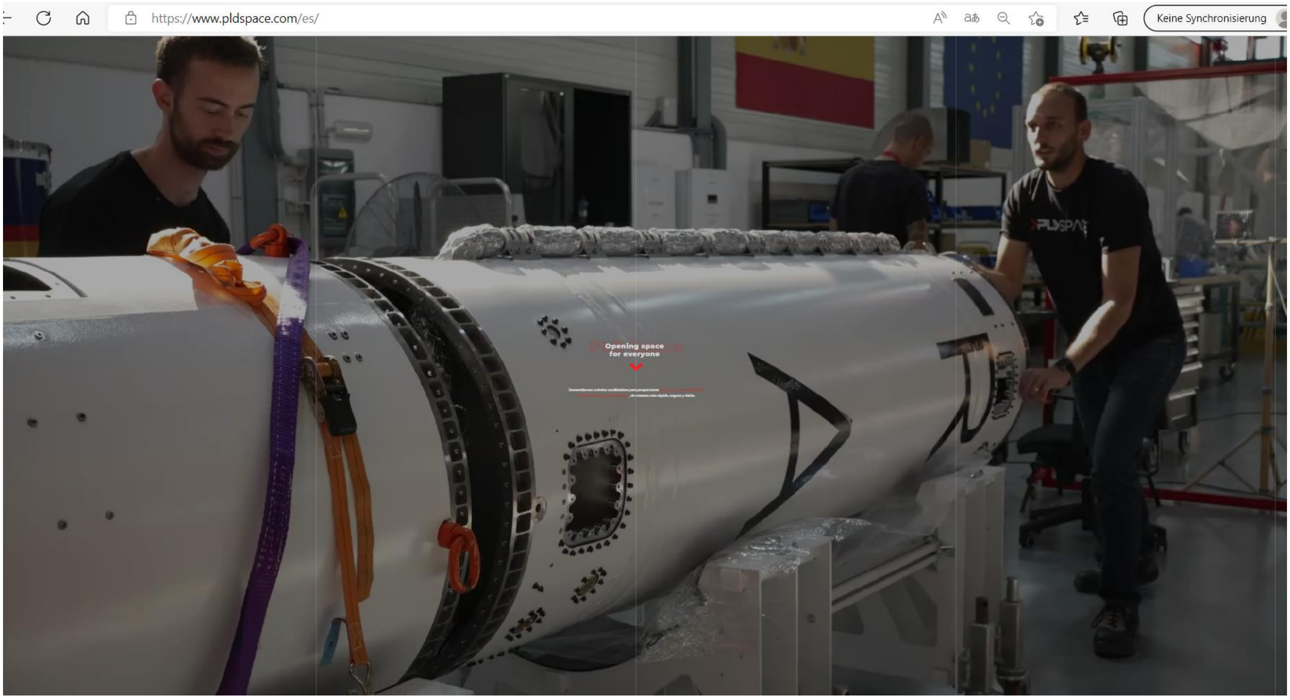
Tasks:

- System Requirements
- System Design
- Structure Mechanical Design (Materials, Shape)
- Control System
-

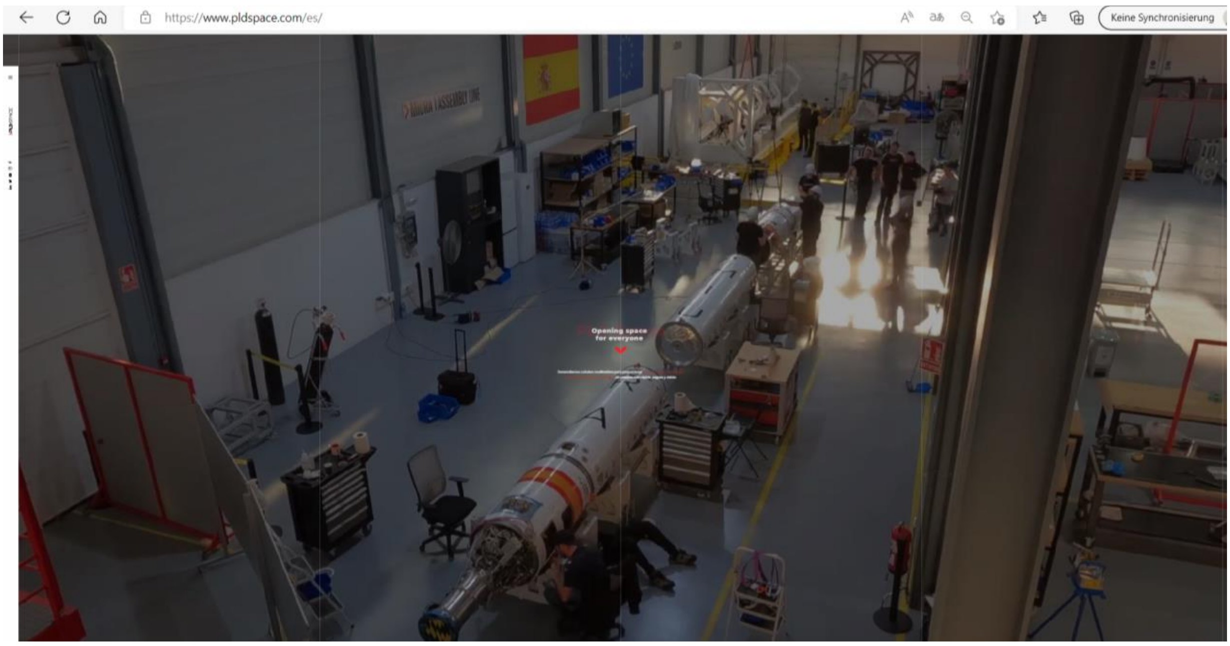
It shall be a copy of a small existing launcher like PLD

4.16.1 Construction, Manufactoring, Assembling &Test Facility

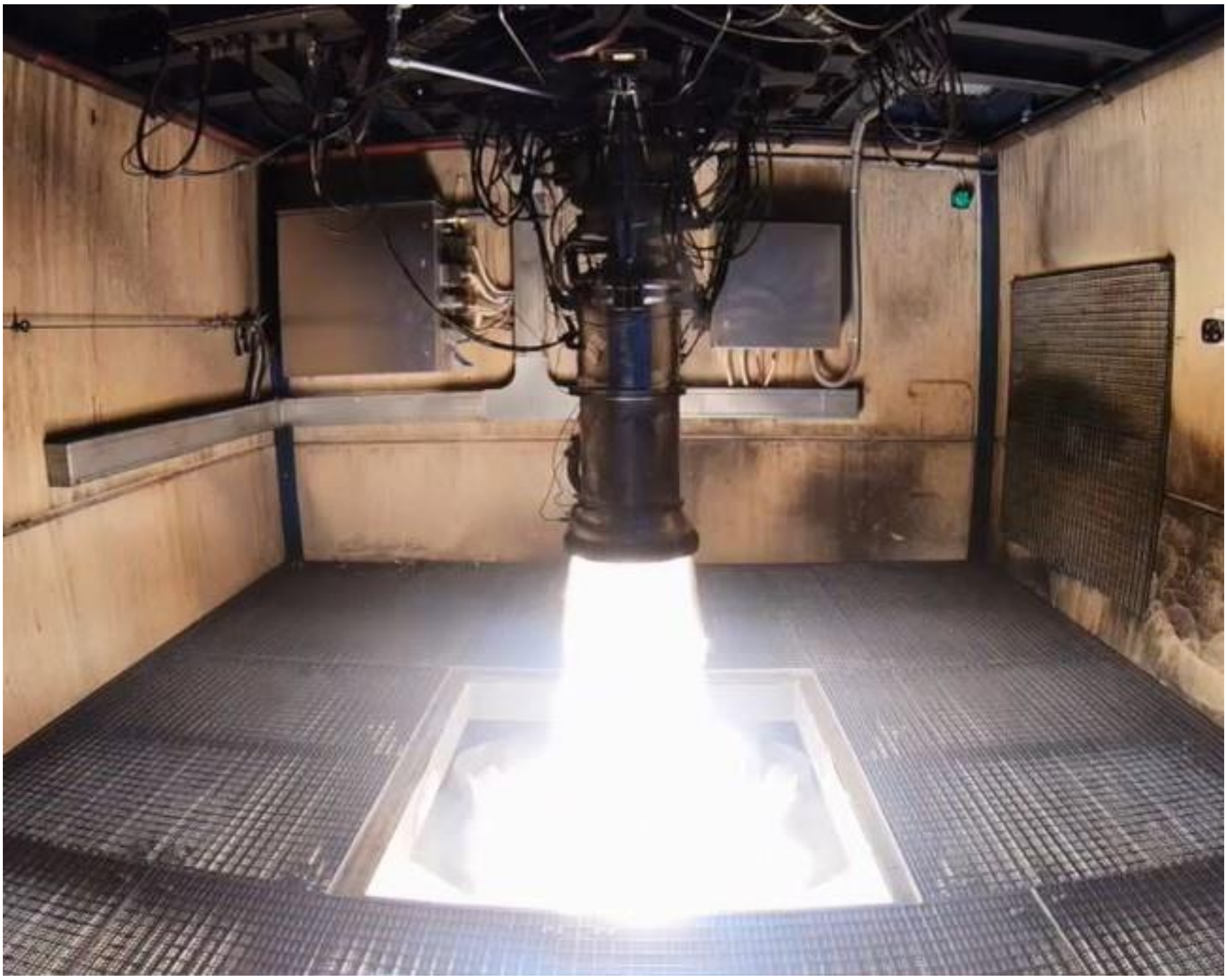
4.16.1.1 Assembly /Integration



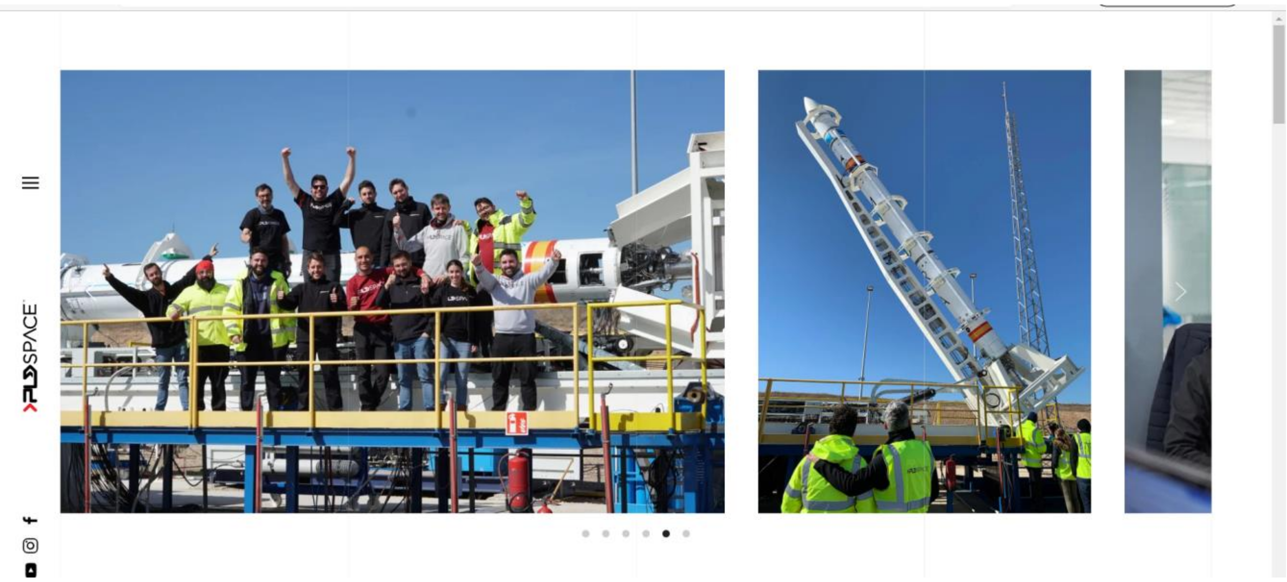

https://www.pldspace.com/en/career
Keine Synchronisierung
PLDSPACE


https://www.pldspace.com/es/
Keine Synchronisierung
Opening space
for everyone

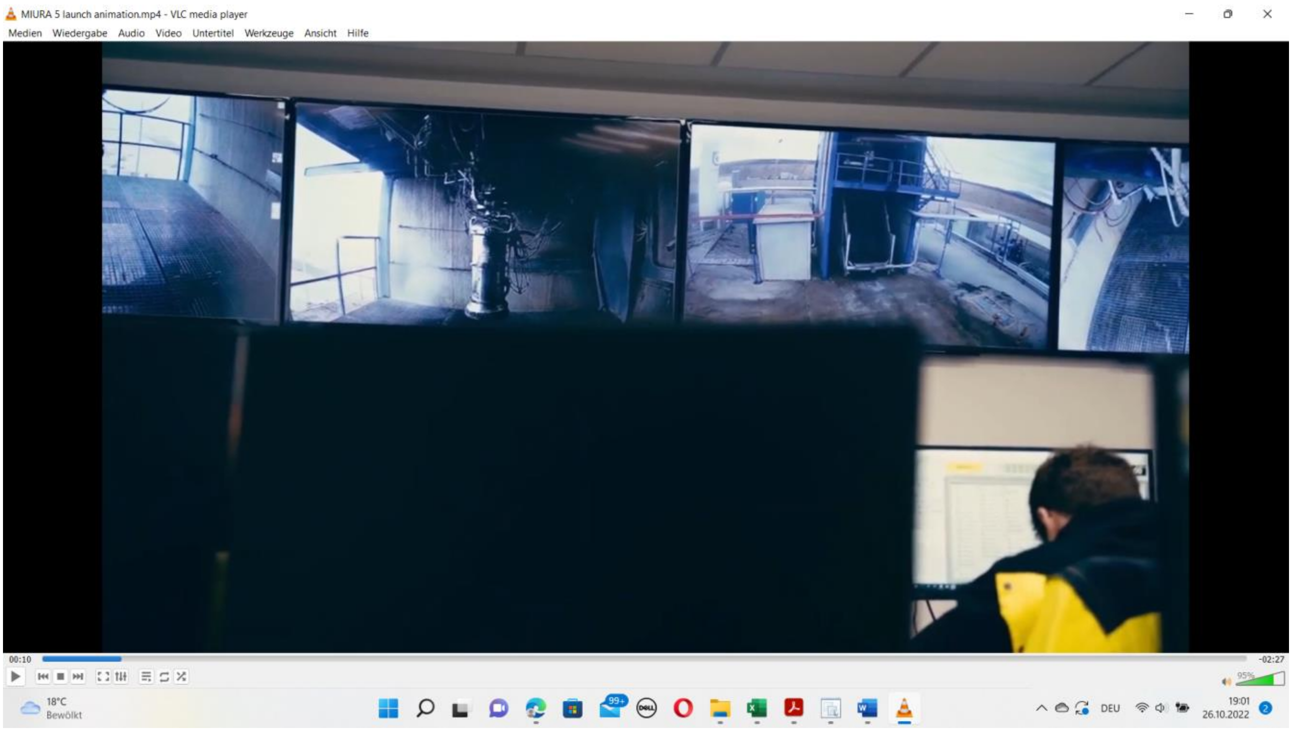
4.16.1.2 Mechanical Analysis



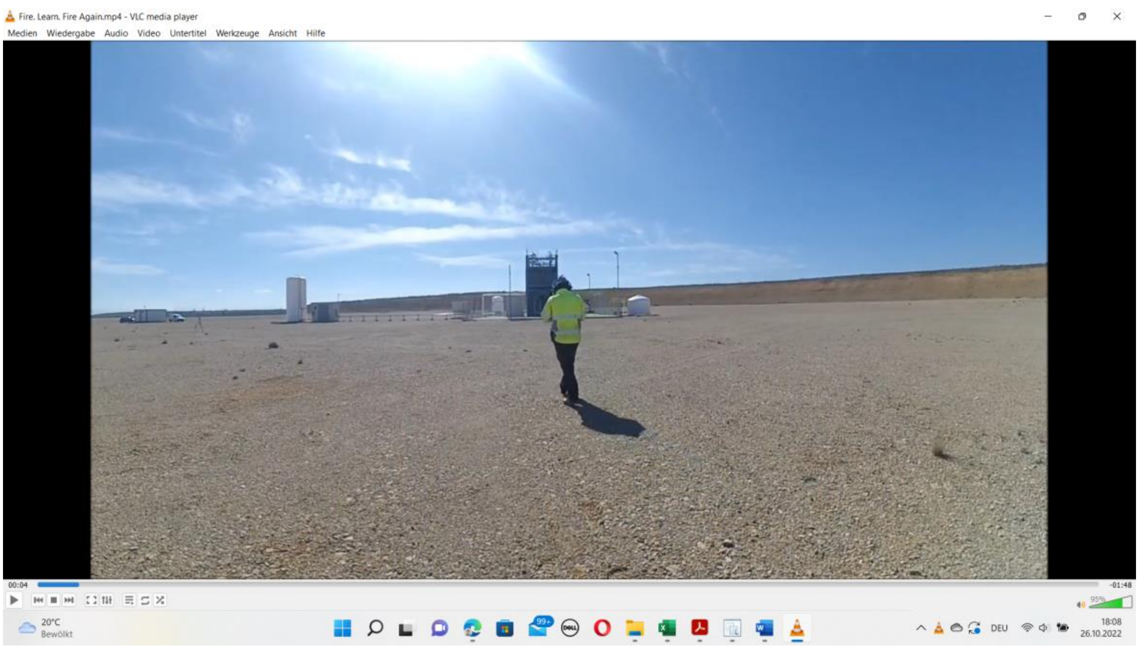
4.16.1.3 (test stands & test bed)



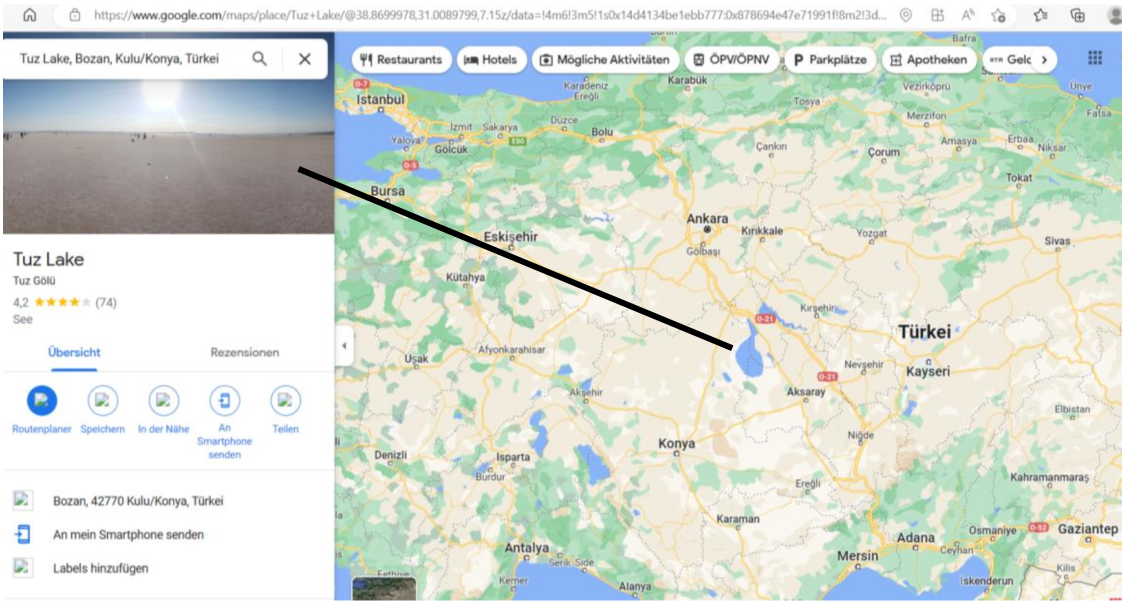
Control Room

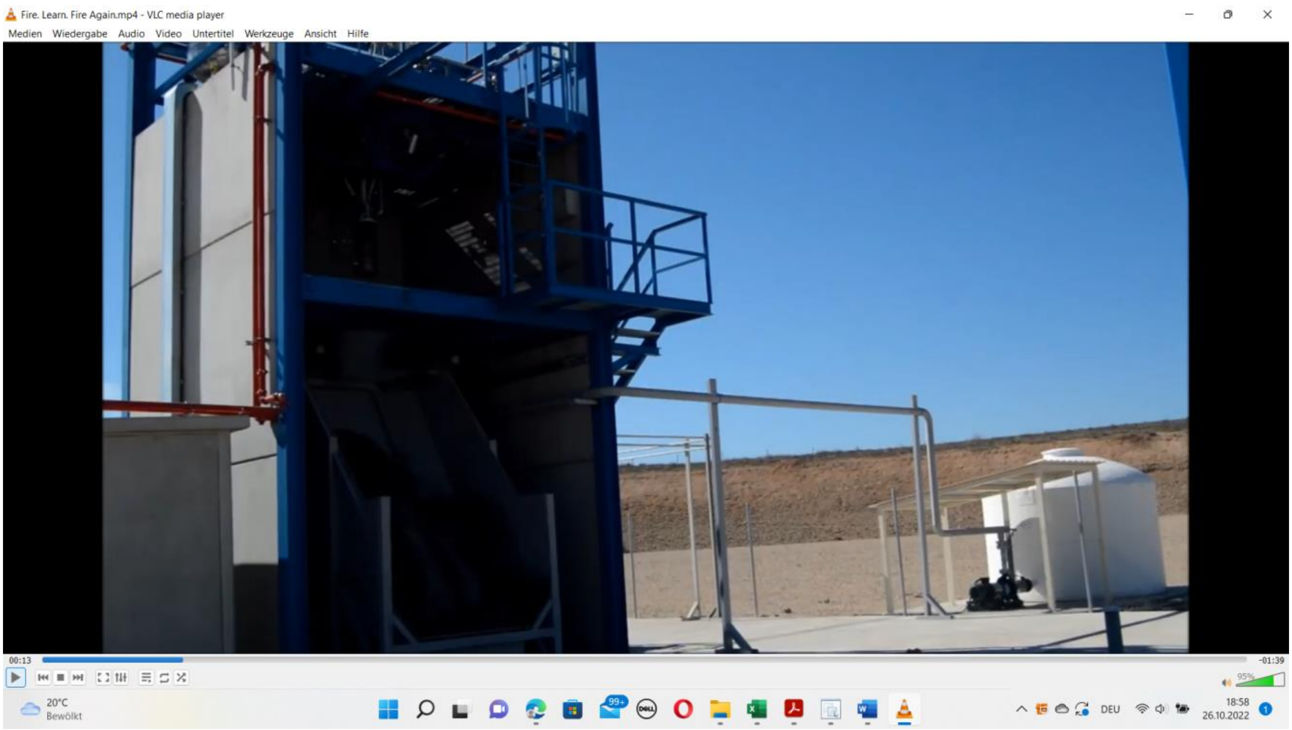


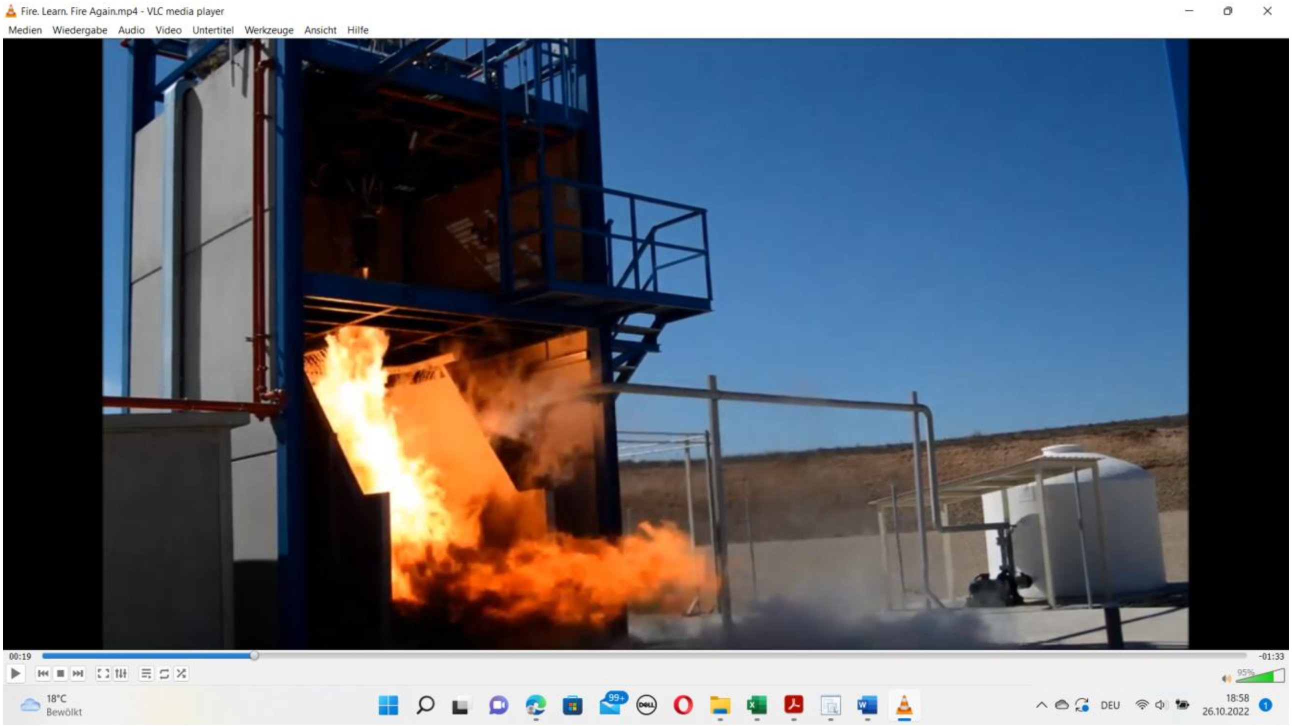
Full Teststand

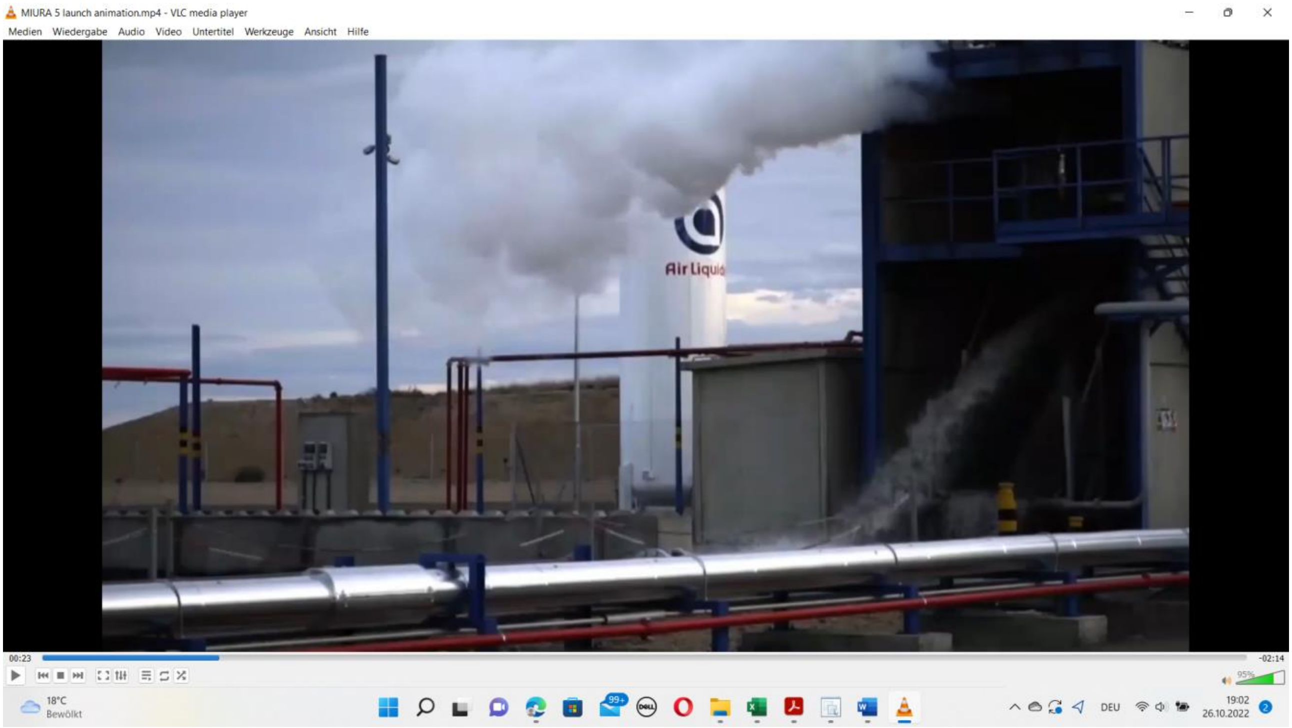


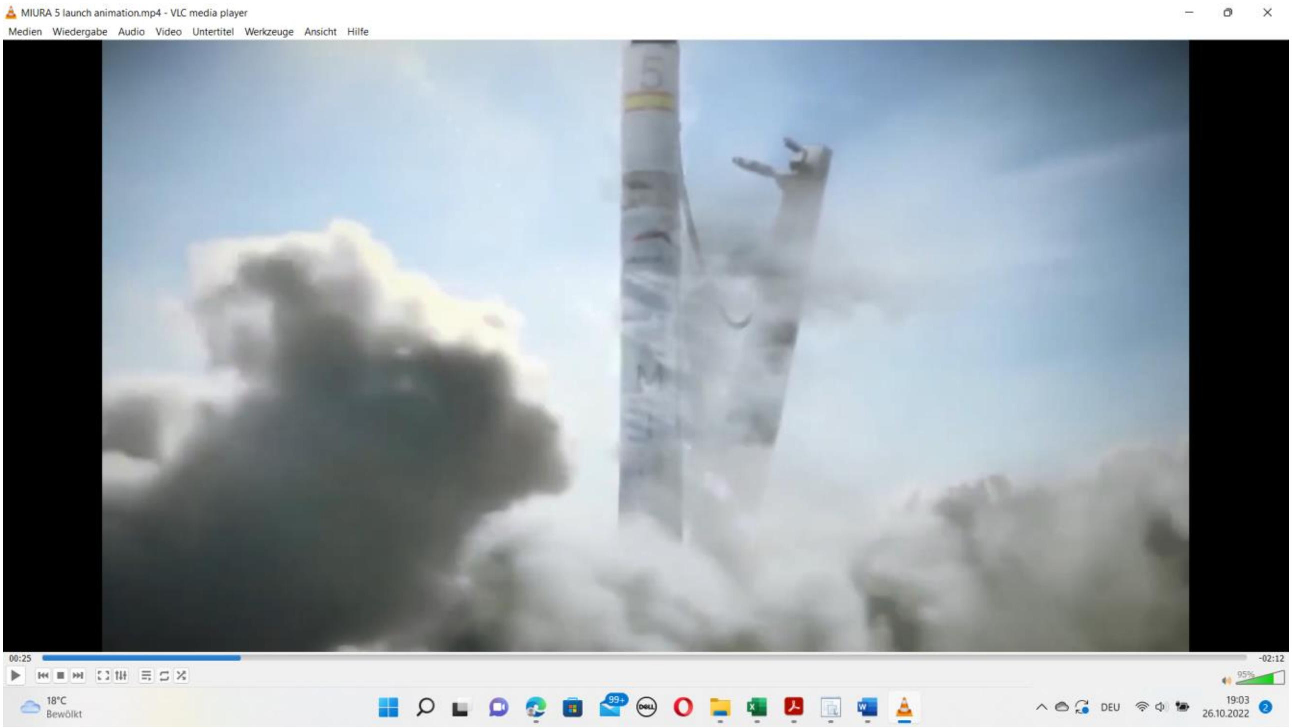
Planned to be at Tuz lake

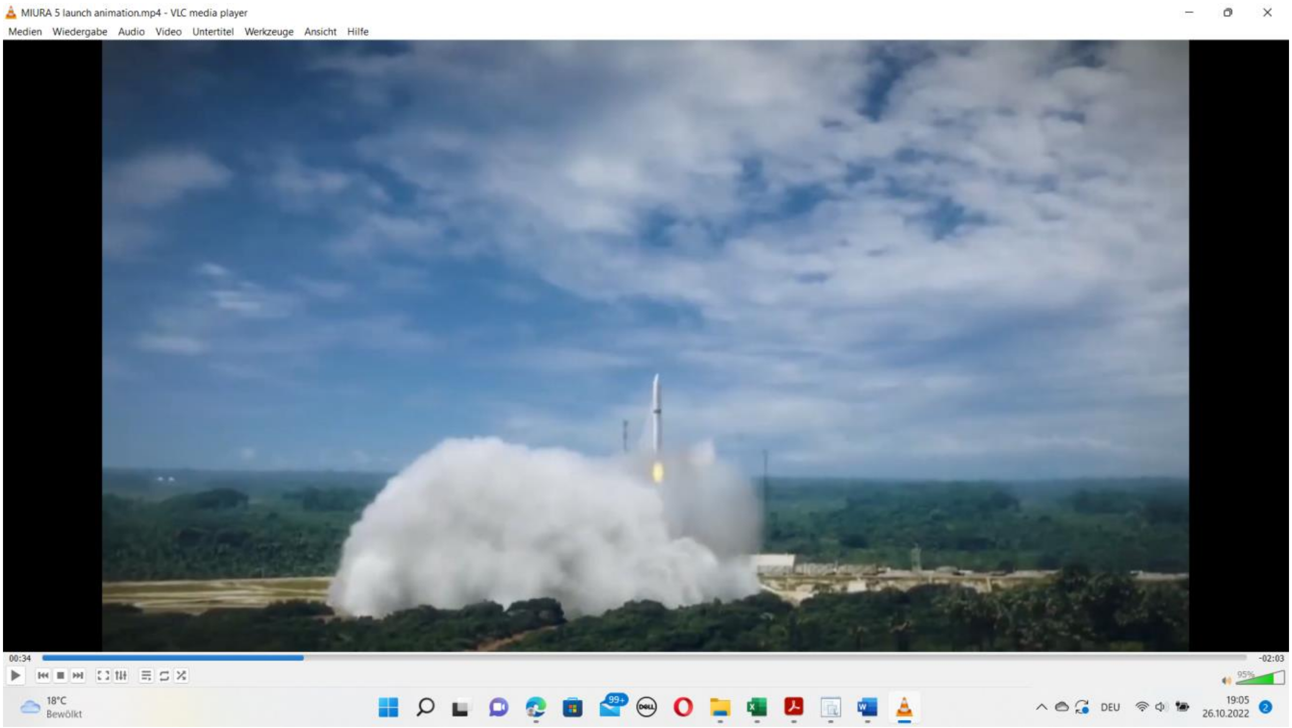




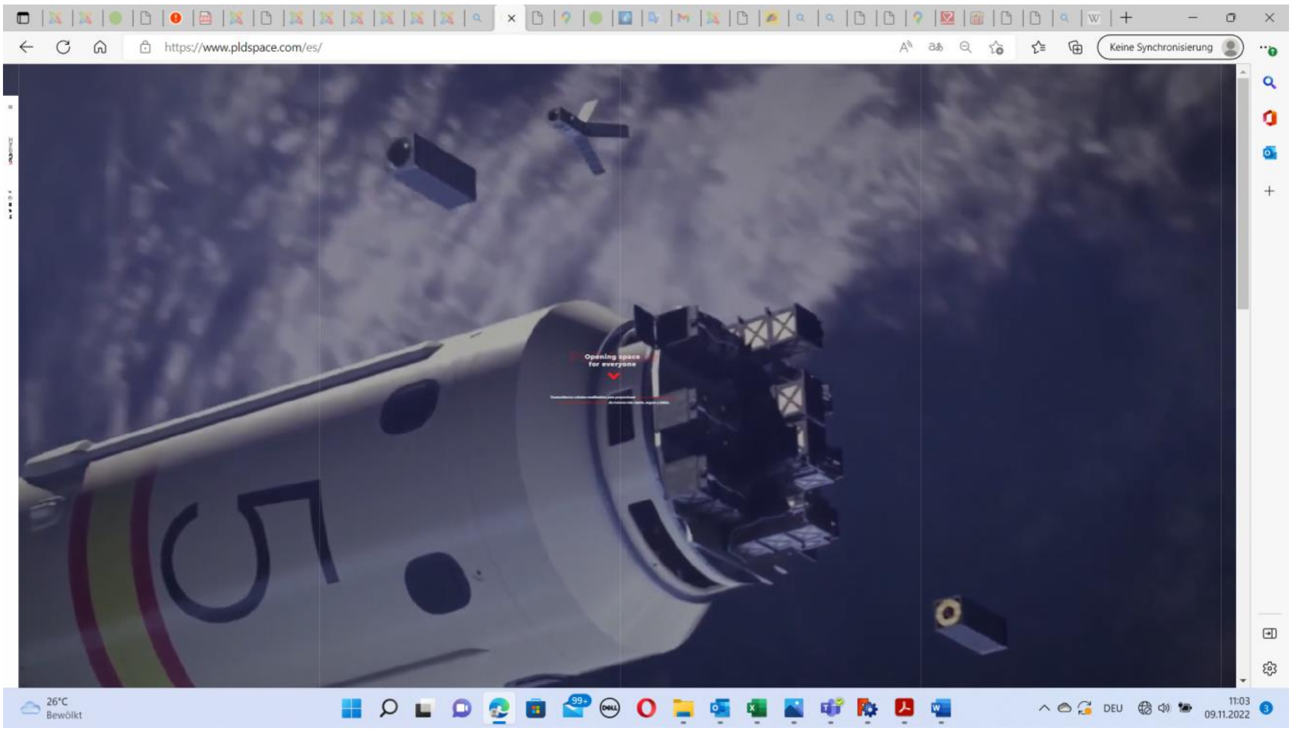

Fire. Learn. Fire Again.mp4 - VLC media player
Medien Wiedergabe Audio Video Untertitel Werkzeuge Ansicht Hilfe
00:19
-01:33
18°C
Bewölkt
DEU
18:58
26.10.2022


MIURA 5 launch animation.mp4 - VLC media player
Medien Wiedergabe Audio Video Untertitel Werkzeuge Ansicht Hilfe
00:23
-02:14
18°C
Bewölkt
DEU
19:02
26.10.2022

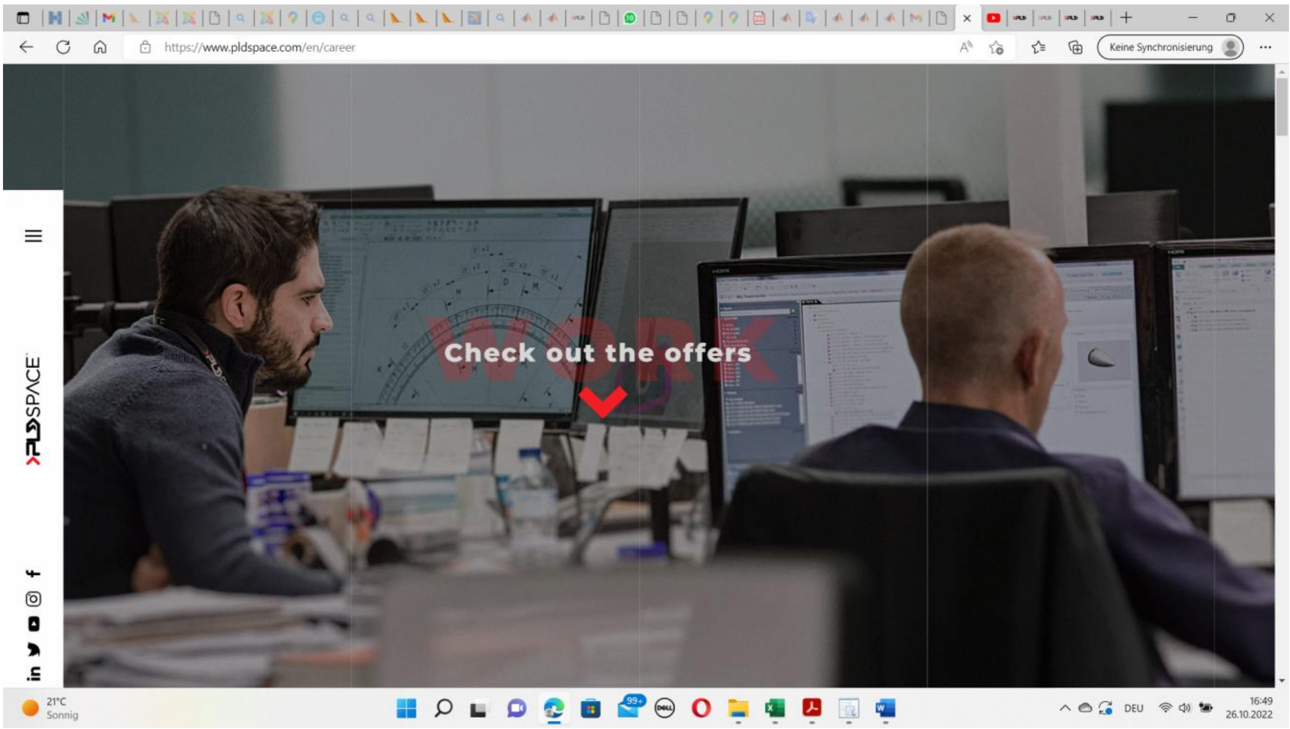

MIURA 5 launch animation.mp4 - VLC media player
Medien Wiedergabe Audio Video Untertitel Werkzeuge Ansicht Hilfe
00:25
-02:12
18°C
Bewölkt
DEU
19:03
26.10.2022


MIURA 5 launch animation.mp4 - VLC media player
Medien Wiedergabe Audio Video Untertitel Werkzeuge Ansicht Hilfe
00:34
-02:03
18°C
Bewölkt
DEU
19:05
26.10.2022

4.16.1.4 Payload



4.16.2 Required Staff[11]



4.16.2.1 MECHANISM DESIGN ENGINEER

- ENGINEERING
- Elche
- **+ view deal**

[11] From Spanish company PLD Space: www.pldspace.com/en/career

Description

Become part of our Structures Team, you will participate in Mechanism Design development for our PLD Space launch vehicles.

Responsibilities

- Mechanism design and its mechanical components, from the concept definition to the detailed design, performing also the standard components selection, such as fasteners, bearings, etc.
- Functional analysis, kinematic of mechanisms and clash analysis, as well as material selection in accordance with operating conditions, environment, etc.
- Research of locking and releasing systems, rotating mechanisms, etc. that will operate in vacuum and under extreme thermal ranges.
- Selection and integration of actuators, electronic components and pneumatic systems according to the specifications, interacting with all competent departments.
- Generation of engineering documentation: 3D models, drawings, bill of materials, weight reports, MCI, technical reports, engineering change notes for the complete definition of detailed parts, assemblies and installation manuals.
- Interaction with other engineering departments to perform the assigned tasks to obtain the required information (Stress, M&P, Testing, Propulsion, Avionics, etc.) also providing input to Manufacturing and Supply Chain departments under request.
- Meet the product specifications, subsystem requirements and interface definition during all product design stages, including: prototyping, testing, qualification and mass production.
- Participate in periodic design review meetings dealing with Supply Chain and Manufacturing for the achievement of requirements, specifications, feasibility, delivery time and cost.
- Continuous improvement research, performance and weight optimisation of the design solutions.
- Identify and recommend opportunities to reduce costs and improve efficiency.
- Complete the assigned tasks “on time, on quality, on cost”.

Required knowledge

- Aerospace or Mechanical Engineering Bachelor’s degree or demonstrable relevant experiencie in lieu thereof.
- Minimum 5 years of relevant technical experience in aerospace industry or equivalent as a Mechanisms Design Engineer.
- Knowledge in selection and application of design criteria and rules, materials, strengths, and other mechanical properties, according to the loading conditions, working environment, etc.
- Knowledge in stack-up analysis and kinematic of mechanisms.
- High degree of CAD competence and PDM tools.

Required competences

- +3Y of parametric modeling design (Siemens NX) and PDM tools (Teamcenter).
- Knowledge in dimensioning and tolerancing of aerospace parts (ASME Y14.5, ISO 1101 standards).

- Experience in mechanism design applications and standard components selection (bearings, actuators, transmissions, etc.).
- Be familiar with different manufacturing technologies (such as machining, forming, welding, composite hand lamination, etc.).

Desired Competencies

- Mechanisms design of rockets or satellites.
- Knowledge of applicable standards, such as Space Mechanisms ECSS Engineering Standards (ECSS-S-ST-33).
- Experience in design of components that operate under extreme conditions (cryogenics, high temperature, vibration, etc.).
- Knowledge of the standards related to corrosion protection and common surface treatments.
- Be familiar with aerospace grade component design, common lightweight metallic and non-metallic alloys and materials.
- Self-motivated and able to work under pressure to meet deadlines, dealing with situations that are constrained by time and managing different tasks at once.
- Able to work autonomously with little supervision.

4.16.2.2 EMBEDDED SOFTWARE ENGINEER

- ENGINEERING
- Elche
- **+ view deal**

Description

As part of a small, passionate, and accomplished team, you will be responsible for the development of the Embedded and flight software for MIURA 1 and MIURA 5 programs.

Responsibilities

- Drive the planning, project definition, technical decisions and trade-offs.
- Provide engineering expertise for design, analysis, production and testing of Launch Vehicles flight avionics.
- To define, clarify, communicate, check, and accept the Avionics and GNC requirements for MIURA 1 and MIURA 5.
- Define and perform electrical verification testing and data analysis.
- Track the execution and release avionics projects to production and testing.
- To design, manufacture and test in-house electronics & software for ground support.
- To monitor the design, manufacture, assembly and test flight qualified electronics.
- Provide support to the Project Assistant with schedule and economical resources of the avionics department.

Required knowledge

- Bachelor's degree in Aerospace engineering, Computer Science or Telecommunications engineering.
- Open-minded engineer with 3 -5 years' experience in developing software for embedded systems in C / C++ for Linux / Windows for critical systems.

- Experience with real time operating systems (RTOS) and/or VxWorks, RTEMS, etc.
- Proven experience in Kernel Development for (Embedded) Linux.
- Strong knowledge in assembly language and architecture for ARM.
- Proven experience in SoC development containing logic units, memory slots, and I/O peripherals.
- Proven experience in ADC, DAC, Timers, PWM, DMA, watchdogs, interrupt processing, and registers.
- Experience in Linux drivers: CAN/I2C/SPI/UART/USB/RS422/RS485 and protocols CSP/FTP.
- Experience with avionics inspection and integration in aerial systems.

Required competences

- English.
- Stress Tolerance.
- Problem solving.
- Multitasking.
- Wordload capacity.
- Autonomy.

Desired Competencies

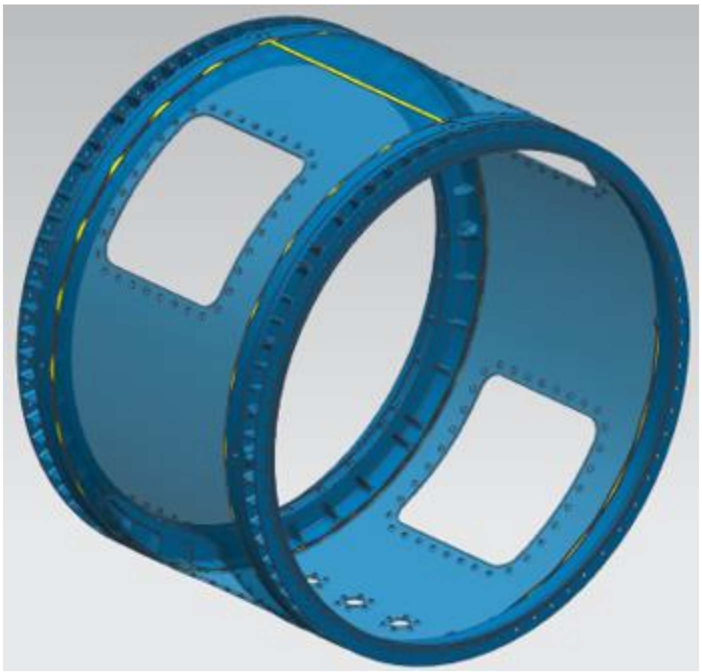
- +2 year experience with Matlab-Simulink for trajectory study and control optimization.
- Knowledge of programming languages such as Python, C, C++.
- +2 year experience with computer aided design (CAD).
- Experience with telemetry systems.
- Experience in CCSDS standards.

4.16.2.3 STRUCTURES ANALYSIS ENGINEER P85

- ENGINEERING
- Elche
- **+ view deal**

Description

- As part of a small, passionate, and accomplished team, you will be responsible for ensuring the structural integrity aspects of the rocket structure.



Responsibilities

- Performing analysis to demonstrate structural integrity and strength aspects associated with Aerospace primary and secondary structures, equipment systems, components and integration according to the technical requirements within the scope of the Project.
- Pre- and post-processing of components and assembly models. Meshing, advance load and boundary conditions definitions, material definitions.
- Perform Linear Static and Dynamic Analysis especially focused on metallic and composite materials behaviour and failure modes.
- Participate in technical reviews and configuration decisions as an expert in the structural strength discipline.
- Participate in the release process of detail, sub-assembly, general assembly and installation drawings, preparing the necessary reports, memos and formal compliance documents (written in English) in order to demonstrate and check that the design satisfies all the stress requirements.
- Liaise with other functions when required to ensure stress requirements are fully integrated.

Required competences

- Bachelor's or Master's degree in Aeronautical Engineering.
- Experience with hand calculations. - In-depth knowledge of engineering principles and design techniques relation to composite materials, classical laminate theory and finite element modelling techniques.
- Good understanding of spacecraft manufacturing methods and processes.
- In-depth knowledge of Linear and Non-Linear Static analysis for metallic components, buckling, bolted joint, fatigue…
- Proficient in using Finite Element Analysis principles and associated tools.
- Deep knowledge on implicit FEM analysis, Nastran solver is mandatory.

- Knowledge in static and thermal analysis. - Self-motivated and able to work under pressure to meet deadlines, dealing with situations that are constrained by time and managing different tasks at once.
- Proactive and good team worker.
- Fluent Spanish and English.

Desired Competencies

- Experience with HyperWorks/HyperMesh and OptiStruct is desirable.
- Good Knowledge on MatLab and VBA Excel is desirable.

4.16.2.4 Experience

- Minimum of 4 years of advance structural analysis experience in static and dynamic load cases, metallic and composite materials.

4.16.2.5 MECHANICAL ANALYSIS ENGINEER

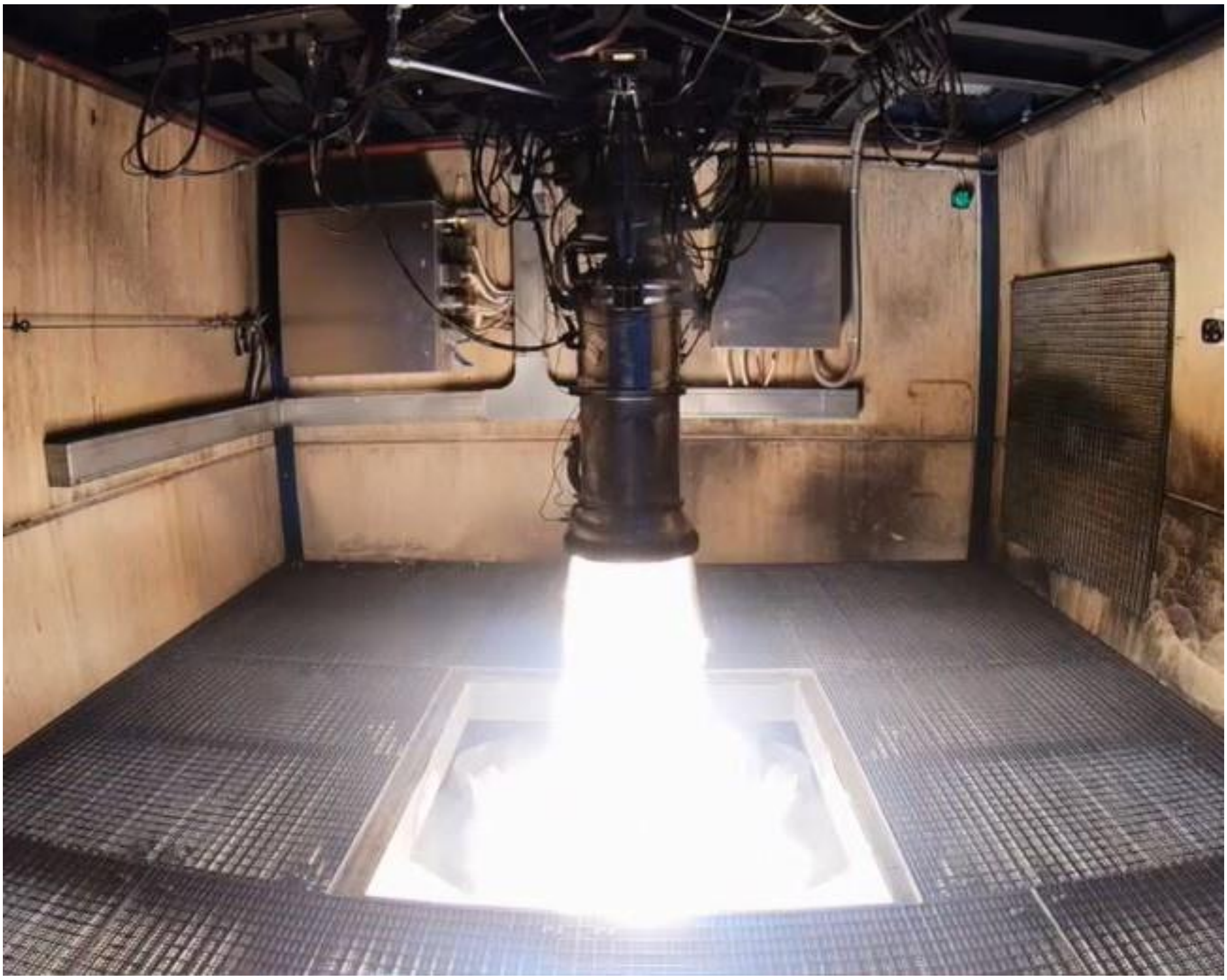
- ENGINEERING
- Elche
- **+ view deal**

Description

As part of a small, passionate, and accomplished team, you will be responsible for designing launch vehicle Propulsion elements and configurations in concert with Structures subsystem. Specific focus on Propellant lines design.

Responsibilities

- Participate in subsystem analysis and architecture of pressure-fed liquid rocket engine, including component sizing, loads determination, and detailed analysis.
- Perform thermal / mechanical load analysis for propellant lines (including cryogenics and fuel lines as well as high pressure gas lines).
- Create and manage test and development plans for engines and propulsion components.
- Perform analysis and interpretation from Propulsion test and flight data.
- Generation of Engineering documentation.
- Maintain and improve program standard approaches to analysis.
- Liaise with Design Engineers of the same department and with others Engineering departments (Structures design and stress, M&P, Quality, etc.) and provide information to the Manufacturing area under request.
- Continuous research to optimize analysis solutions.
- Identify and recommend opportunities to reduce costs and improve efficiency.
- Complete the assigned tasks “on time, on quality, on cost".



Required competences

- Aerospace or Mechanical Engineering Bachelor’s degree or demonstrable relevant experience in lieu thereof.
- Minimum 3 years of relevant technical experience.
- Demonstrated experience in analysis metallic structures for propellant lines.

- Knowledge in selection and application of analysis criteria according to the loading conditions, working environment, etc.
- Fluent written and oral communication both in Spanish and English

Desired Competencies

- Previous experience in Aerospace Industry as Propulsion analysis engineer or Stress Analysis Engineer (or an equivalent combination of education and experience).
- Familiar with the main manufacturing technologies and capabilities
- Knowledge of the most common corrosion prevention rules and surface protection.
- Knowledge of the most common cryogenic (LOX) cleaning procedures.
- Knowledge of Cryogenic (LOX) manipulation.
- Familiar with Aerospace standard parts and typical light materials, metallic and non-metallic.
- Preferred software: Ansys and Teamcenter.
- An occasional need to travel with the role will be required.
- Self-motivated and able to work under pressure to meet deadlines, dealing with situations that are constrained by time and managing different tasks at once.
- Able to work autonomously with little supervision.

4.16.2.6 MECHANICAL DESIGN ENGINEER

- ENGINEERING
- Elche
- **+ view deal**

Description

- As part of a small, passionate, and accomplished team, you will be responsible for designing launch vehicle Propulsion elements and configurations in concert with Structures subsystem. Specific focus on Propellant lines design.

Responsibilities

- Participate in subsystem design and architecture of pressure-fed liquid rocket engine, including component sizing, loads determination, and detailed design.
- Design of primary metallic structures (different size range of machined components, sheet metal parts) for propellant lines (cryogenics, fuel and gas) fitting.
- Design of secondary structures as brackets or supports for rigid pipes, valves, mechanism, propulsion components electrical harness, etc.
- Generation of Engineering documentation such as 3D models, Drawings, BOMs, and Eng. Change Notes for the complete definition of Detailed Parts, Assemblies, and Installation.
- Maintain and improve program standard approaches to design, safety factors, material allowables, drafting standards, etc.
- Liaise with others Engineering departments (Structures design and stress, M&P, Quality, etc.) and provide information to the Manufacturing area under request.
- Responsible to accomplish with specifications and requirements during the development of the assigned tasks or product design along the entire lifecycle and across all phases from Prototype, Testing and Series production.
- Participate in the periodic design reviews dealing with SCH/Manufacturing for the achievement of requirements, specifications, feasibility, lead time and prices.
- Continuous research to optimize design solutions.
- Identify and recommend opportunities to reduce costs and improve efficiency.
- Complete the assigned tasks “on time, on quality, on cost".

Required competences

- Aerospace or Mechanical Engineering Bachelor's degree or demonstrable relevant experience in lieu thereof.
- Minimum 3 years of relevant technical experience.
- Demonstrated experience in design metallic structures for propellant lines.
- Knowledge in selection and application of design criteria and rules, materials, strengths, and other mechanical properties, according to the loading conditions, working environment, etc.
- High degree of CAD competence and PDM tools.
- Fluent written and oral communication both in Spanish and English

Desired Competencies

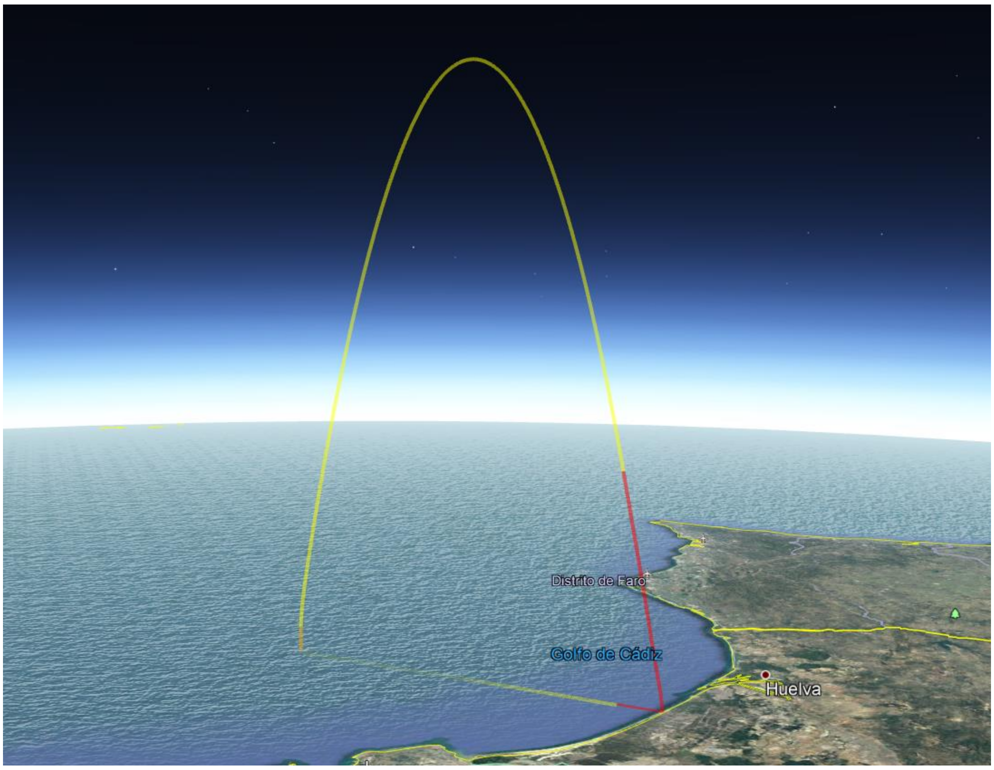
- Previous experience in Aerospace Industry as Propulsion design & analysis engineer or Structural Design Engineer (or an equivalent combination of education and experience).
- Familiar with Aerospace standards
- Familiar with the main manufacturing technologies and capabilities
- Knowledge of the most common corrosion prevention rules and surface protection.
- Knowledge of the most common cryogenic (LOX) cleaning procedures.
- Knowledge of Cryogenic (LOX) manipulation.
- Proven in the development of products from the original concept design according to Aerospace normative and rules, through to the delivery of detailed technical design packaging into final production.
- Design focused to assembly line concerns, tooling, and jigs.
- Familiar with Aerospace standard parts and typical light materials, metallic and non-metallic.
- Preferred software: Siemens NX and Teamcenter.
- An occasional need to travel with the role will be required.
- Self-motivated and able to work under pressure to meet deadlines, dealing with situations that are constrained by time and managing different tasks at once.
- Able to work autonomously with little supervision.

4.16.2.7 SYSTEM ENGINEER

- ENGINEERING
- Elche
- **+ view deal**

Description

- Systems Engineer for MIURA 1 launch vehicles and associated tests and future launch operations.



Responsibilities

- Develop and manage system and subsystem requirements, including verification techniques and interface control.
- Work proactively and collaboratively with the engineering team and customer to identify, track, mitigate, and resolve technical/programmatic issues that may arise during program execution.
- Serve as engineering focal for mission requirements and exchange of technical information between subsystems on launch vehicles and payloads.
- Assist with the development of design, integration, and test processes, procedures, and configurations for launch vehicles in concert with other engineering disciplines.
- Provide support for design/analysis results level to technical team and program management.
- To be part of launch vehicle Simulation analysis and other performance analysis across vehicle functions.
- To assist with the development of test methods and plans according to defined requirements.
- To be part of CONOPS upgrade and implementation.
- To elaborate design documents.

Required competences

- Master's degree in Aerospace engineering, Electrical Engineering, Mechanical Engineering, Systems Engineering or equivalent.
- +3 years' experience with Systems Engineering SW tools
- +3years experience in high reliability Systems Engineering design and processes
- +2 years' experience with space systems, satellite, launch vehicles, etc.
- Experience managing technical budgets, requirements, V&V, AIT and plan definition
- Expertise in Computer Aided Design (CAD) using industry tools such as Siemens NX.

- Ability to work with multidisciplinary teams in technical and programmatic areas
- Experience with space Launch Vehicle (suborbital or orbital)
- Familiarity with hardware test program requirements captured in industry standards including ESA ECSS Standards, SMC-S-016, NASA-STD-7001, NASA-STD-7002 and GSFC-STD-7000.
- Team player able to work with different subsystems

Desired Competencies

- Experience with mission definition and requirements
- Experience in environmental testing, thermal vacuum, vibration, electromagnetic compatibility, as well as test methods and validation.
- Expertise in Matlab/Phyton.
- Understanding of launch vehicle key subsystems and their associated performance parameters.
- Experience in managing subcontractors.
- Fast adaptability to rapid changes environments.

Experience

- Experience working with government, NASA, ESA or commercial launch contracts.

4.16.2.8 ELECTRONICS ENGINEER (Senior Avionics Engineer)

- ENGINEERING
- Elche
- **+ view deal**

Description

- Responsible for the avionics design, integration, and tests for PLD launch vehicle programs.

Responsibilities

- Stablish requirements for PLD launch systems avionics.
- Design the electronic components and subsystems for MIURA 5 launch vehicle.
- Integration of avionics subsystem in the vehicles.
- Stablish production prototype and testing activities for recurrent PLD Space programs.
- Electrical verification of the full avionics' subsystems.
- Flight operations and procedures.
- Ground Support for software and hardware.
- Ground integration with vehicles.
- Pre-flight / post-flight data review and correlation with simulations and tests.
- Software verification with software-in-the-loop and hardware-in-the-loop.

Required knowledge

- Bachelor’s degree in Electrical engineering, Aerospace Engineering, Telecommunications Engineering.

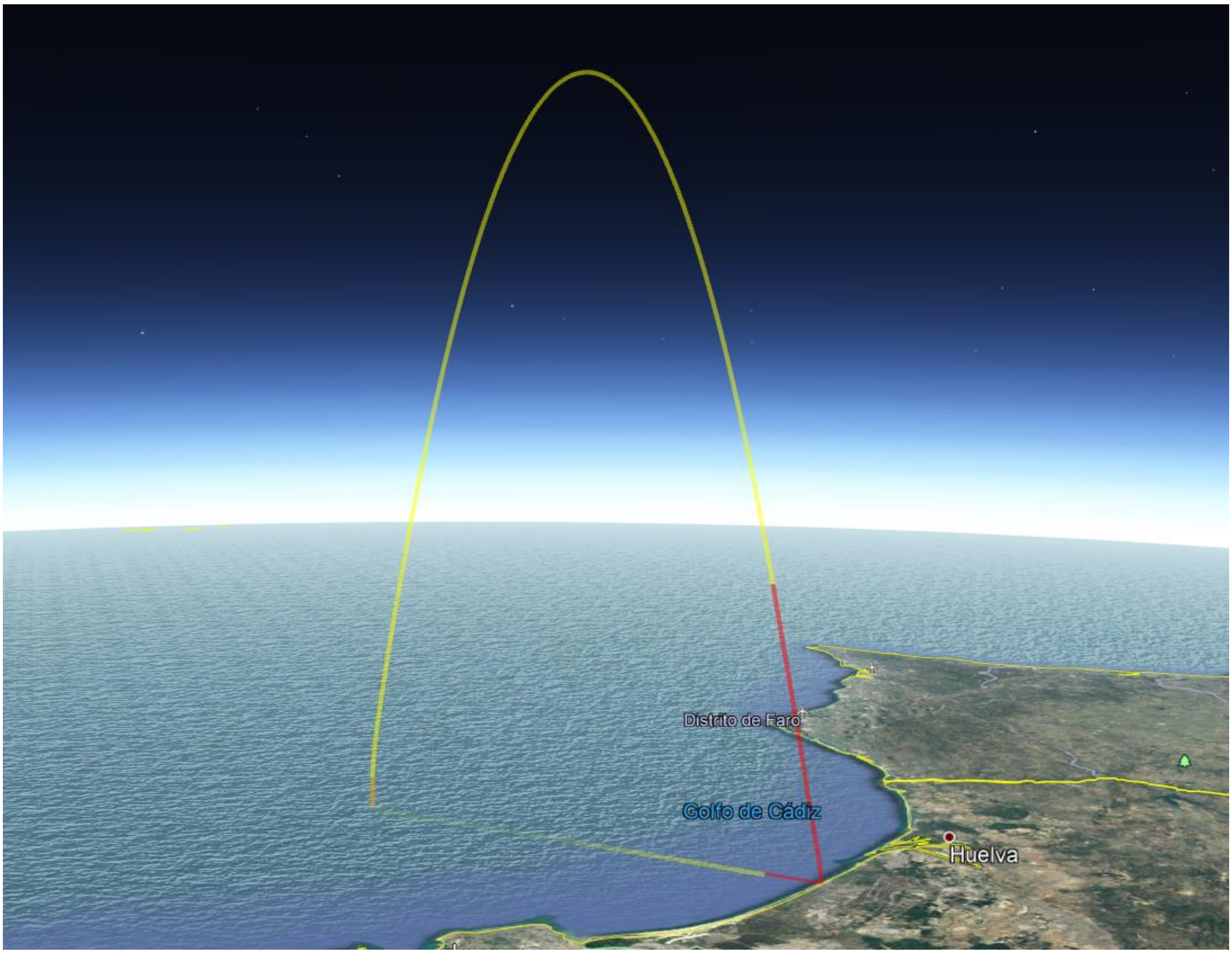
- 3-5years' experience in Electronics design, manufacturing and testing.
- 3-5 years' experience in Matlab/Simulink.
- 3-5 years' experience working with sensors, signal conditioning and data acquisition electronics.
- 3-5 years' experience in power distribution units and power systems.
- Experience in COTS, automotive, industrial and military electronic components.

Required competences

- English.
- Stress Tolerance.
- Problem solving.
- Multitasking.
- Wordload capacity.
- Autonomy.

Desired Competencies

- Experience working with government, NASA, ESA or commercial launch contracts.
- Experience in High reliability systems and industry standards.
- Experience in international regulations for launch vehicles.
- Experience in spacecraft integration and testing.
- Experience in telemetry systems.
- Experience in programming in Python, C, C++.
- Experience in FTS design and dispersion simulation.
- Experience in flight structural hardware design, testing and integration in composite materials and metallic materials.



4.16.2.9 STRUCTURAL TEST ENGINEER

- ENGINEERING
- Elche
- **+ view deal**

Description

PLD Space is seeking a new member for our team to serve as a Test Engineer, within the Structures Department. This position will be focused on assisting the qualification, and acceptance testing of structural components for its launch programs.

Responsibilities

- Definition of tests, considering available inputs, requested outputs and measurements, as well as other boundary conditions.
- Test campaign planning.
- Test suppliers management: contact, evaluation, selection, agreements.
- Test execution/follow-up.
- Assessment of externally executed tests.
- Assistance in post-processing of test data in collaborations with Stress team.
- Report writing and internal dissemination of test outcomes.

- Assessment of sensorization requirements and load conditions for which data is to be obtained in order to meet test objectives in a safe manner.
- Collaborate in the design of test benches and tooling.
- Create work methodologies or improve the existing ones in order to benefit current and future projects.
- Research state of the art in testing methodologies and sensors and propose creative approaches to complex problems.
- Occasionally travel to other PLD Space or supplier locations.

Required competences

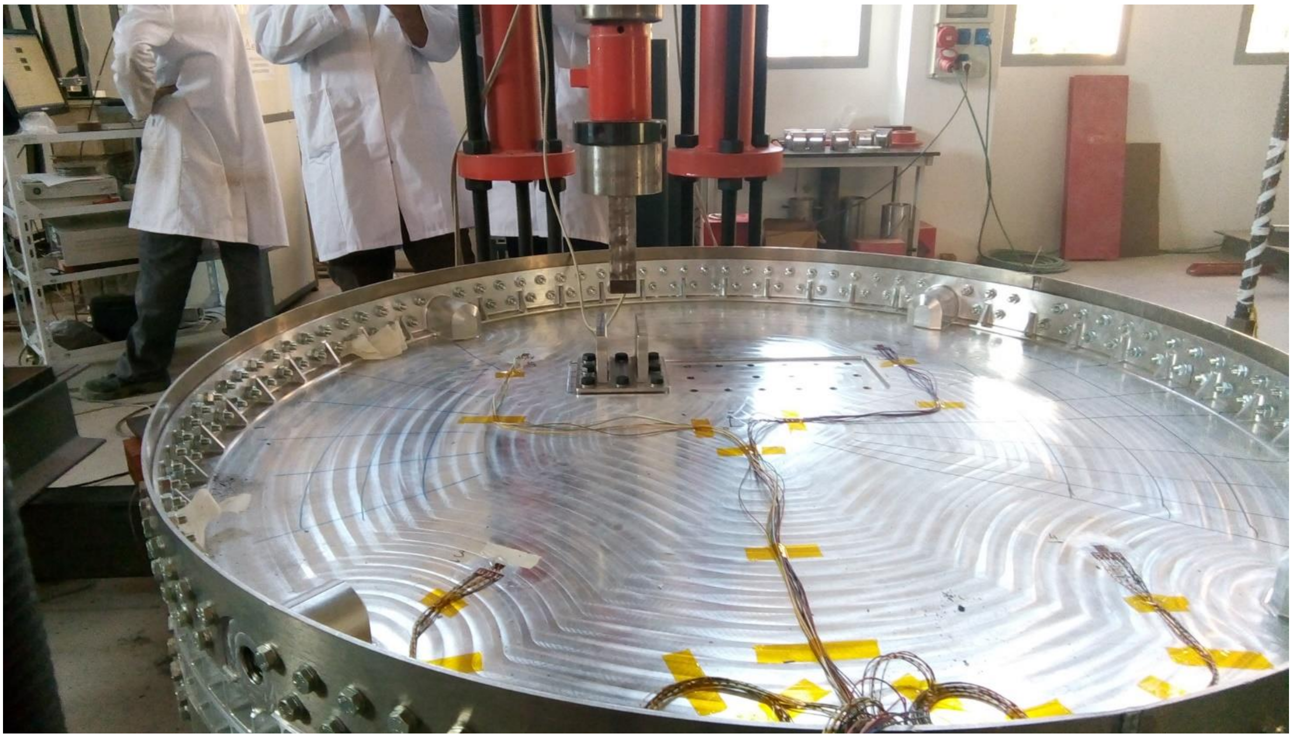
- Aerospace or Mechanical Engineer.
- Fluency in English is mandatory.
- +5 years' experience in similar position, testbench/hardware and lab work.
- Used to work with standards (ECSS, ISO, EN, ASTM, MIL...).
- Used to work under pressure without neglecting safety and quality.
- Strong analytical and problem-solving skills, along with excellent written and oral communication skills.

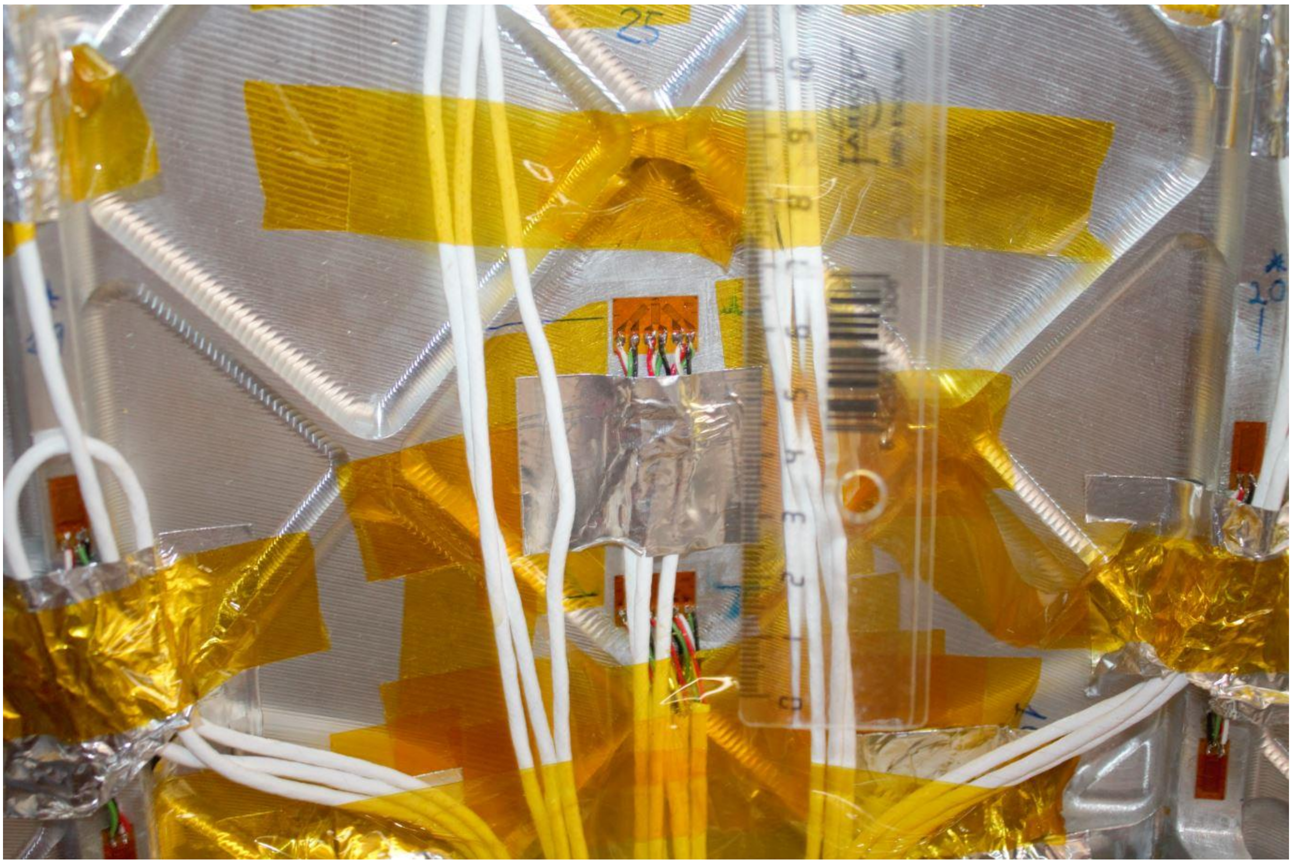
Desired Competencies

- Knowledge on specific space sector, space vehicles and launchers, metallic pressure vessels, COPV and payload structures is high desirable.
- Knowledge of pressure vessel codes (BPVC, EN 13345, PD5500, AD 2000...).
- Skills on electronics, optical instrumentation and extensiometric measurement will be welcomed.
- Vibration and dynamic analysis test experience.
- Experience in analysis or design positions is desirable.
- Experience with Catia or NX Unigraphics is desirable.
- Basic knowledge of finite element analysis shall be welcomed.
- Programming experience (Matlab/Octave, Python...) is desirable.

Experience

- Experience developing and conducting test plans for experimental structural test.
- Knowledge on developing reports to analyse test results from test execution.
- Experience working with testbenches.





4.16.2.10 PROPULSION TEST ENGINEER

- ENGINEERING
- Elche
- **+ view deal**

Description

PLD Space is seeking a new member for our team to serve as Propulsion test engineer for MIURA 1 launch vehicle. The primary role of this position involves testing hardware from development, qualification, and acceptance for flight at both subsystem and component level. The Candidate will carry out rocket engine testing activities and propulsive stage testing activities at PLD Space propulsion facilities in Teruel (Spain). This role is for a hands-on person. The main workplace will be Elche with frequent travels to Teruel and Huelva.

Responsibilities

- Deep understanding of liquid rocket engine propulsion subsystems and components.
- Deep understanding of test procedures, industry standards, and best-practices.
- Knowledge of thermodynamic principles as applied to cryogenics and high-pressure systems, to ensure test facilities can provide physical parameters defined for each engine system or component.
- Knowledge of safety principles and best practices while working with high pressure fluids and cryogenics.
- Perform fundamental fluid flow, heat transfer and structural analysis to contribute to the design and development of new propulsion test facilities
- Write test plans, test procedures, and maintain configuration management system.
- Perform data analysis and prepare data reviews and performance reports to evaluate propulsion test facility performance.
- Work with technicians in the field to prepare test articles prior to testing.
- Coordinate with technicians to operate cryogenic facilities, fuel facilities and high-pressure lines for engine testing.
- Provide support to System Engineering department to define, coordinate and control the functional and physical interfaces between the propulsion system and the launch vehicle.
- Perform other duties as assigned.

Required competences

- BSc in Aerospace Engineering or Mechanical Engineering.
- Must be able to stand for extended periods.
- Must be able to maneuver in tight spaces.
- Must be able to handle items weighting more than 15kg.

Desired Competencies

- Experience working with high pressure systems, desired experience with Fuel and/or cryogenic components and propellants.
- Knowledge of manufacturing processes, test & integration procedures of liquid aerospace propulsion systems or pressurized system.

Experience

- 5+ years of Rocket Propulsion subsystem testing activities or aerospace-related propulsion testing activities.

4.16.2.11 GNC ENGINEER

- ENGINEERING
- Elche
- **+ view deal**

Description

Trajectory design and analysis of launch vehicles and related launch operations in the department of Systems and GNC.

Responsibilities

- Perform trajectory design, optimization, and dispersion for suborbital and orbital launch vehicles.
- Development of 6-DOF launch vehicle and spacecraft dynamics simulators.
- Perform aerodynamic characterization of aerospace vehicles.
- Contribute to flight safety analysis, safety corridors, IIPs.
- Interaction with other subsystems and systems engineers for trajectory optimization.
- Contribution to launch licensing according to international regulations.
- Provide inputs for flight hardware (structural and propulsion components).
- Pre-flight / post-flight data processing and correlation with simulations and tests.
- Software verification via testing with flight-software-in-the-loop and hardware-in-the-loop.

Required competences

- Master's degree in Aerospace engineering, Space exploration or flight mechanics, or GNC.
- 5-7 years of experience with Matlab/Simulink.
- 5-7 years of experience with flight simulation programs (rocket trajectories) and orbital mechanics.
- Experience with space Launch Vehicle trajectory simulations (suborbital or orbital) and with the use of ASTOS, STK or similar software.
- Relevant experience in rocket trajectory analysis and simulation software development.
- Knowledge of Computer Fluid Dynamics (CFD) software.
- Knowledge of ballistic missile and orbital trajectory simulations.
- Knowledge of aerodynamic reentry simulations and analysis.
- Knowledge of security analysis and international regulations.
- English.
- Stress Tolerance.
- Problem solving.
- Multitasking.
- Wordload capacity.
- Autonomy.

Desired Competencies

- Experience with spacecraft orbital trajectory design and simulations.
- Experience with ballistic missile trajectory design.
- Experience with re-entry aerothermodynamics analysis and simulations.
- Experience with Safety analysis and international regulations for launch vehicles.

- Experience with FTS design and dispersion simulation.
- Experience with control dynamics and optimization strategies for missile and orbital spacecrafts.

4.16.2.12 PROPULSION ENGINEER

- ENGINEERING
- Elche
- **+ view deal**

Description

senior-rocket-propulsion-engineer-for-miura-1

Responsibilities

- To Participate in the TEPREL-B Liquid Rocket Engine Flight Qualification, including hardware integration, testing and post-hot test data processing.
- To be part of the engineering decision process of MIURA 1 propulsion subsystem final development, testing and integration.
- To provide inputs for TEPREL-B liquid rocket engine upgrades and to improve its performance.
- To contribute and participate in the engineering decision process of MIURA 1 propellant feed system (including piping, valves, sensing, actuators, etc.)
- To contribute in material characterization, material selection and to generate specification for updates and further developments.
- To develop test plans for liquid propulsion parts and assemblies.
- To generate manufacturing and assembly specifications.
- To contribute in the upgrade PLD Space Propulsion Test Bench software and hardware.
- To operate Company´s test bench during hot test campaigns.
- To provide launch and mission operations support.

Required competences

- Master´s degree in Aerospace Engineering.
- Relevant experience in liquid propulsion (+7 years of liquid propulsion experience, design and testing) .
- +5 years experience in Finite Element Analysis using industry tools such as ANSYS.
- Excellent use of Siemens NX/Teamcenter, EcosimPro, Matlab and Simulink tools.
- Relevant experience in manufacturing processes (metal machining): Copper Alloys, Aluminium Alloys and Stainless-steel Alloys, as well as Electrical Discharge Machining (EDM).
- Other relevant experience in manufacturing processes, such as Aluminium and Stainless-steel TIG Welding.
- Good communications skills, including C1 English Level (both written and oral).
- Ability and willingness to thrive in a fast-paced work environment.
- Highly motivated and committed to improvement.
- Ability to represent a workgroup.
- Ability to work as a part of a team.

Desired Competencies

- Use of Labview is desired.
- Experience with space launch vehicle design.

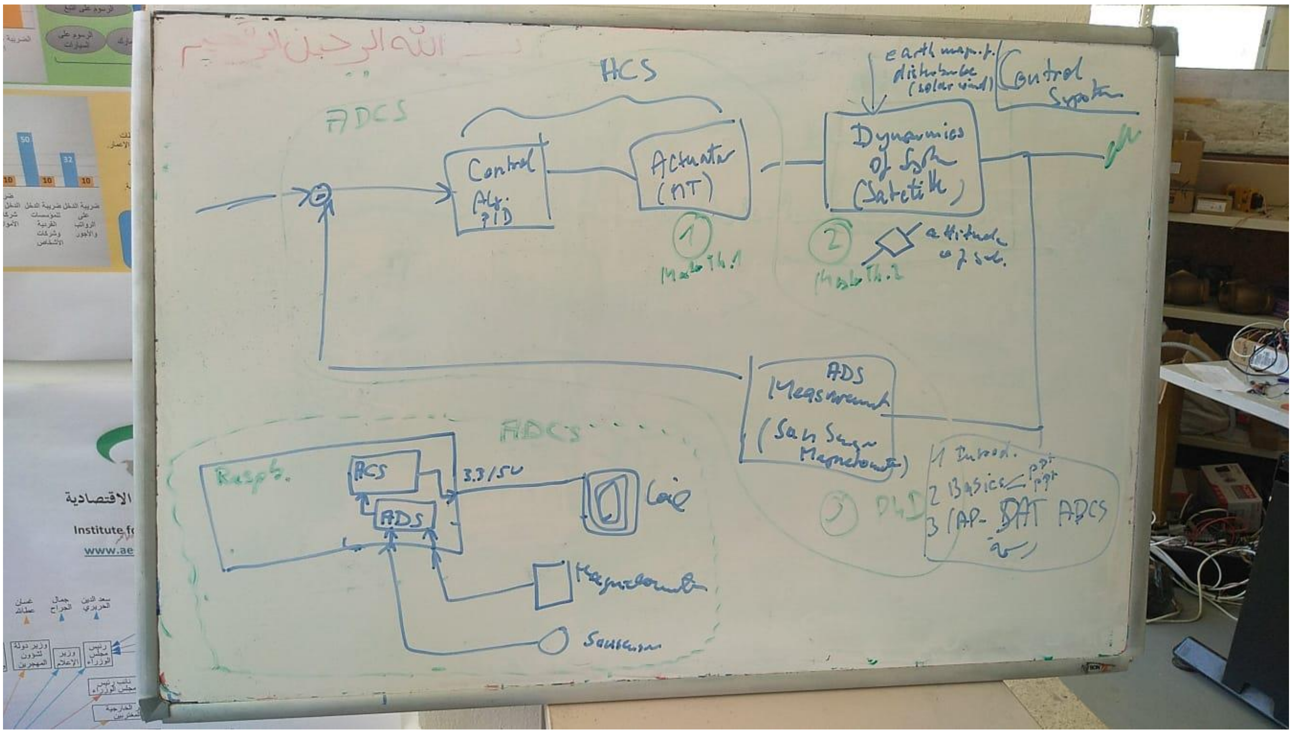
- Spanish (both written and oral).
- Experience working with Spanish Government or National Agencies (NASA, CNES or DLR).
- Experience working with Institutional Contracts with European Space Agency (ESA).

Experience

- +7 years of liquid propulsion experience, design and testing.

4.17 AS-COMSAT-1 Launcher System (Mechanics, Electronics & SW)

4.17.1 OBC System



4.17.2 AIS system on the OBC

See Master Thesis Rozan

4.17.3 System Concept for Trajectory Control System

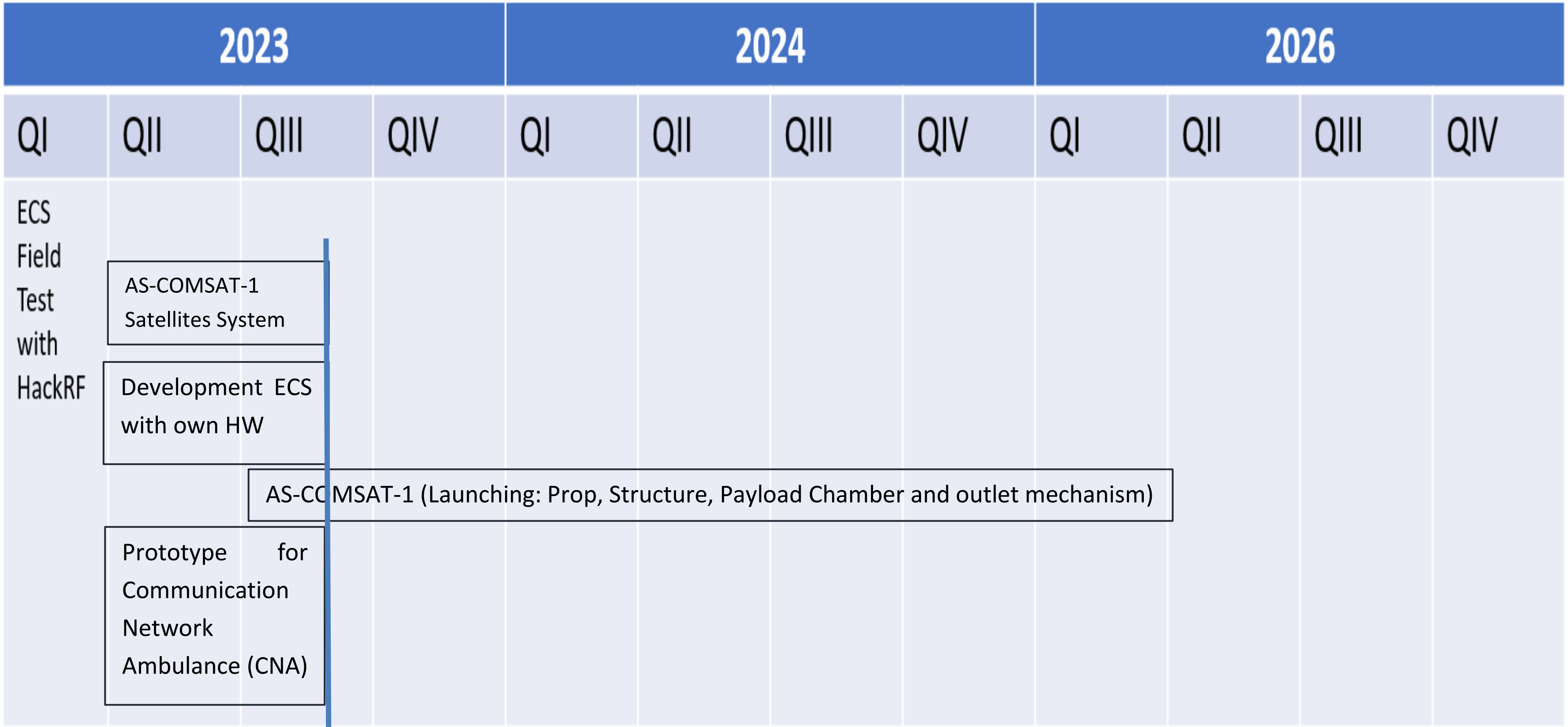
4.17.4 Telemetry, Tracking & Control (TT&C) Ground Station

See AS-COMSAT-1 Technical Development Documentation

4.18 Planning&Controlling 2023

Remark: The actual status of working packages for ECS, CAN and AS-COMSAT_1 satellite system is on AS-COMSAT Planning&Controlling 2023 (aecenar.com). It will be updated in this document at the end of each month insha Allah.

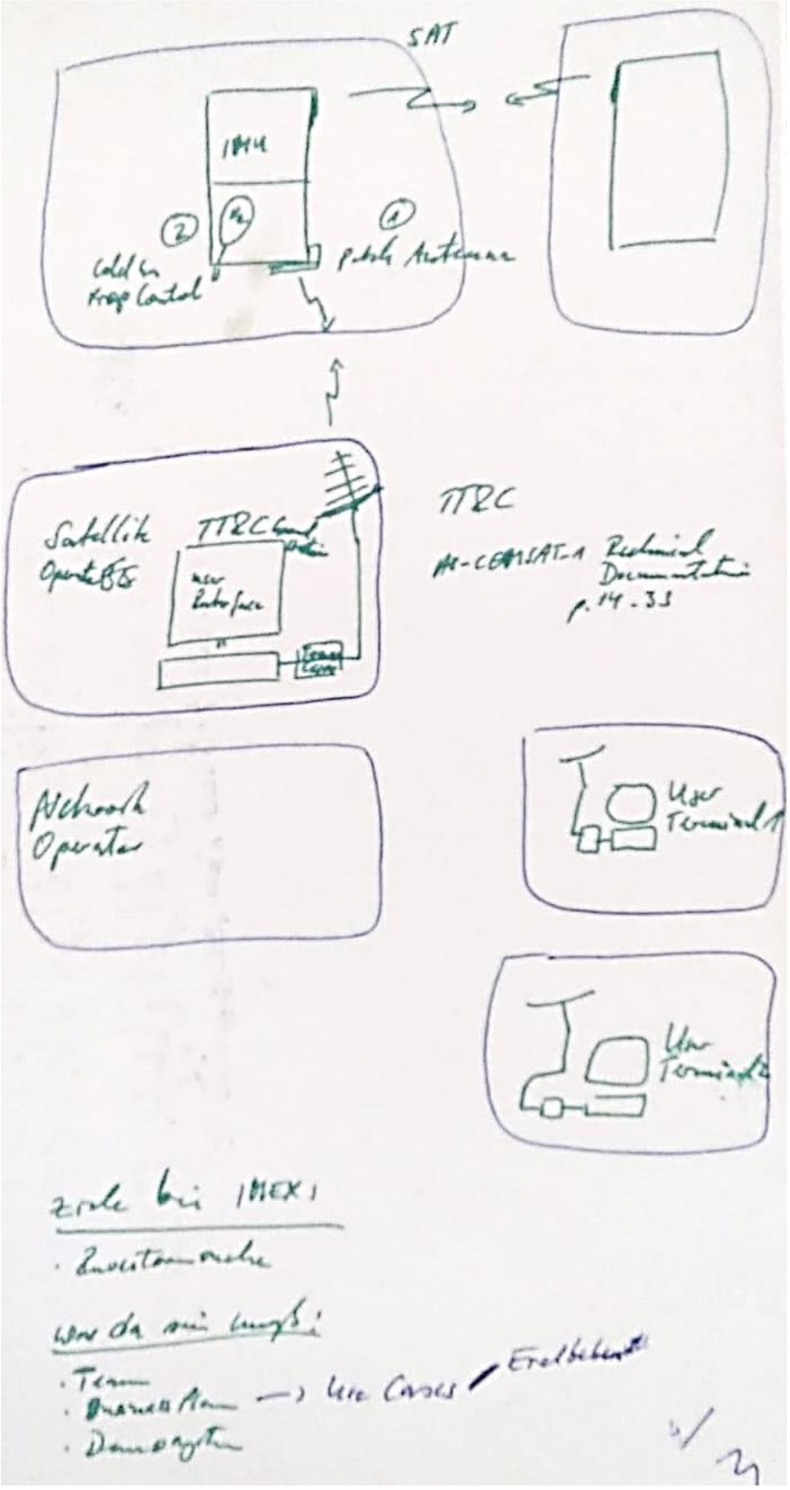
4.18.1 Milestones 2023 (Updated: 6.5.23)



4.18.2 Overall Working Packages for AS-COMSAT_1 System (Satellite System+Launching)

	Working Package	Responsible	Name of Technical Documentation	Department/ Stakeholder	Status
1	CoreFlightSystem, on-board computer	Abd	**System Design**	**ICS**	
2	Attitude control system	Raja/Hana	ACDS Technical Report	**ICS**	
3	AIS	Candar	AIS Clustering	**Marmara University, Istanbul, Faculty of Computer Science**	**done**
4	Telemetry and payload COM system, intersatellite communication	Ahmad Awad	**COM/OBD**		**open**
5		Yahya	X-Ray Sensor	**IAP**	
6	Ground station	Hana	Ground Station	**AS-COMSAT**	**open**
7	Launching issues	Siham	**Orbit and Altitude Specification, Legal issues, pre-launch activities**		**Done**
8			**Vibration damper**		
9			**Thermal Isolation**		

4.18.3 Working Packages for Satellites System AS-COMSAT-1 (Satellite system) until IMEX Istanbul 14-17 Sep 2023 (Updated: 7.5.23)



- 2 x 12 U Satellites (two identical Satellites)
- Satellite Operator (TT&C) Station) (see [AS-COMSAT-1 Technical Documentation, 2021], pp. 14-33)
- Network Operator Station
- User Terminal 1
- User Terminal 2

4.18.4 Satellite Prototypes (2 identical satellites)

	Working Package	**Responsible**	**Name of Technical Documentation**	**HW Procurement/Price**	**Statu**
1	On-board Main computer (OBC) (RaspberryPi)	Raja	System Design	100$	
2	Attitude control system	Raja	ACDS Technical Report	IMU from Mintad Project 100$	
4	Telemetry COM system, intersatellite communication - Like CNA communication system - Patch antenna	Ahmad Awad	COM/OBD	Board Development Patch Antenna 200$	In progr
	Payload COM system	Ahmad Awad		300$	
	Power System (Solar Cells, Controller, Lithium Batteries)	Abdullah Q.		1500$	
	Cold Gas Propulsion Unit (4 Outlets)	Raja		1000$	
	12 U chassis	Uthman		500$	
8			Vibration damper		
9		Uthman	Thermal/Radiation Isolation	100$	
		AS-COMSAT Staff Istanbul	Satellite Integration	300$	
			Total Material Price	**4000$**	

> **2 Satellites: 8000$ Material Price**

4.18.4.1 Status of Hardware for Satellite (Example Table)

Hardware	Status
Onboard Computer	Received
Solar Cells	Received
Power System	Ordered – 10 Week Lead Time
Chassis	Ordered – Unknown Lead Time
ITC Designed Solar Panel PCBs	Designed – Out for Quote
Radio	Ordered – 6 Month Lead Time
Clean Room	Procured and Setup for Ribbon Cutting
Deployable Antenna	Ordered – Unknown Lead Time
Camera	Received

4.18.5 Satellite Operator (TT&C Station)

	Working Package	**Responsible**	**Name of Technical Documentation**	**Department/ Stakeholder**	**Status**
6	Ground station	Hana	Ground Station	**AS-COMSAT**	**open**

4.18.6 Network Operator Station

4.18.7 Task TT&C Ground Station

Task (Bachelor Thesis/Student Project)

Development of a Ground Station for a system of 4 LEO satellites (AS-COMSAT_1)

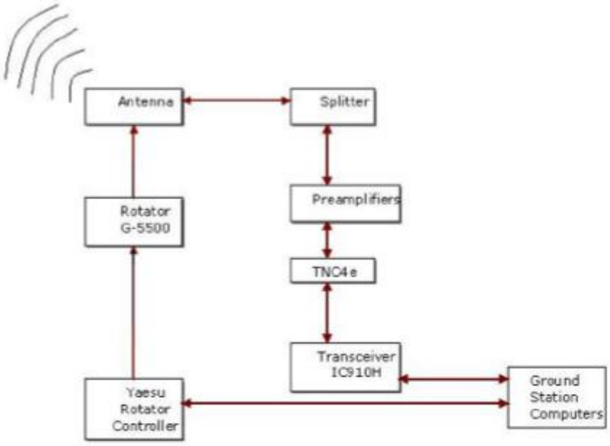
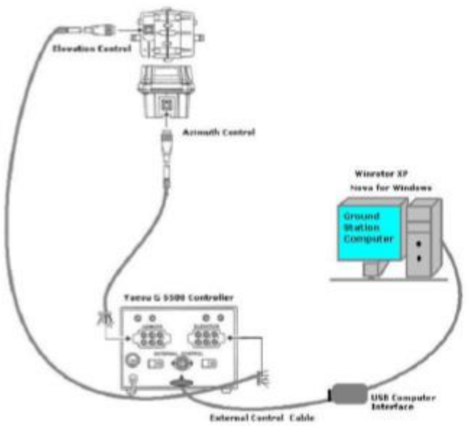


Figure 1.4: Block Diagram of the Ground Station.



Rotor-Controller-Computer Interfacing.

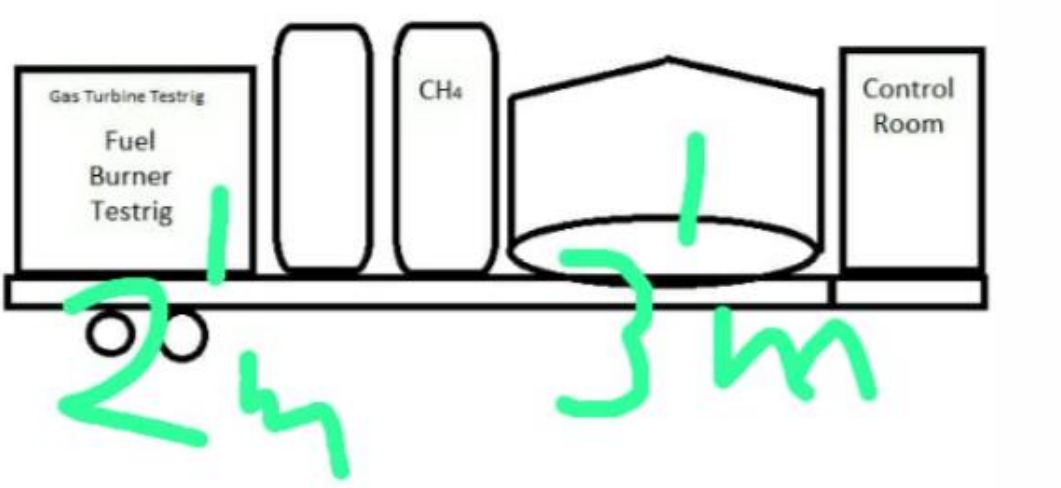
Graphical User Interface

4.19 Working Packages for ECS

4.20 Working Packages for CNA

4.21 Working Packages for Transporter

4.21.1 Mobile Testrig



-------2.3m-----|---1m-----|---------3m--------|-2.2m--|

-> 8.5m

4.21.1.1 Mobile Platform – Option1

12000$

4.21.1.2 Mobile Platform – Option 2 (8m + 2m, 6000USD) – Ali Dib

How many tons unknown

4.21.1.3 Mobile Platform – Option 3 (12mx2,6m, 58t, 8000$) – Abdullah Mohammad Mourad

4.21.1.4 Mobile Platform – Option 4 (8,5mx2,6m, 50 tons, 6200$)


55tons

4.1 Timeline and Human Ressources Management (Gantt Diagram)

To be done

4.2 References

[1] Jana Othman, AECENAR Internship report, 3.Sep 2021

[2] http://aecenar.com/index.php/downloads/send/16-ics/666-temoleb-mintad-final-report-1999-2020

[3] *Development of a GS Package suited for Spacecraft Operation Control and Optimization methods for Satellite flyby over the Ground Station*, Raj Gaurav Mishra, Master Thesis at University of Wuerzburg, Germany, 2007

[4] A DESCRIPTION OF A STANDARD SMALL SATELLITE GROUNDSTATION FOR USE BY WMO MEMBERS, TECHNICAL DOCUMENT WMO/TD No. 660, 1995

[IAP-SAT Final Report (2012-2020)]

[IAP-Transporter Report 1] (2016)

[IAP-Transporter Report 2] (2017)

[IAP-Transporter Report 3] (2018)

[AS-COMSAT-1 Technical Documentation, 2021]

4.3 Company Liquidation (in Dec 2024)

LİMİTED ŞİRKET TASFİYE SONU KARAR ÖRNEĞİ

TASFİYE HALİNDE AS COMSAT HABERLEŞME PLATFORMLARI VE UYDU HABERLEŞME SİSTEMLERİ SANAYİ VE TİCARET LİMİTED ŞİRKETİ

Karar No : 2024/02

Karar Tarihi : 30.10.2024

Toplantıya Katılanlar: SAMIR MOURAD, SALİH BAYAR, ÖMER KORÇAK

Şirketimiz 24.06.2024 tarihinde tasfiyeye girmiş olup; tasfiye kararı 04.07.2024 tarihinde tescil edilmiştir. Alacaklılara çağrı ilanları 05.07.2024, 16.07.2024 ve 24.07.2024 tarihli Türkiye Ticaret Sicili Gazetelerinde yayımlanmıştır.

Tasfiyenin sonuçlandırılmasına, tasfiye bilançosunun kabulü ile tasfiye memurunun ibra edilmesine karar verildi.

Ortak	Ortak	Ortak
SAMIR MOURAD	SALİH BAYAR	ÖMER KORÇAK
6232042167	20647473968	10462212378

TAHAKKUK FİŞİ

T.C
HAZİNE VE MALİYE BAKANLIĞI
İSTANBUL İLİ VD. BAŞKANLIĞI
034223 ALEMDAĞ VERGİ DAİRESİ MÜDÜRLÜĞÜ

2024120501G790000034

MAKİNA NO
SIRA NO

VERGİ KİMLİK NUMARASI 0861584690
SOYADI (UNVANI) TASF.HAL.AS COMSAT HABERLEŞME
ADI

Ana Vergi Kodu 0010
KURUMLAR VERGİSİ

ADRES
TEPEÜSTÜ MAH. KAFİYE SK. -
Kapı No:47 Daire No:3 Tel:5362678790
ÜMRANİYE İSTANBUL

Kabul Tarihi	Vergilendirme Dönemi	Düzenleme Tarihi
05/12/2024	07/2024-11/2024	05/12/2024

TÜRÜ	MATRAH	ORAN	TAHAKKUK EDEN	MAHSUP EDİLEN	ÖDENECEK OLAN	VADESİ
0010 KURV	0,00		0,00	0,00	0,00	25/12/2024
1047 DVER	0,00		532,00	0,00	532,00	25/12/2024
1048 5035	0,00		624,10	0,00	624,10	25/12/2024
				TOPLAM	1.156,10	

İşlem Türü 0010
Thk Türü 9014
YALNIZ BİN YÜZELLİALTI TL ON Kr .dir

5 Early fire detection system using uav and artificial intelligence (Master Thesis, June 2021)

LEBANESE INTERNATIONAL UNIVERSITY

EARLY FIRE DETECTION SYSTEM USING UAV AND ARTIFICIAL INTELLIGENCE

Master Thesis

by

Ali A. Assaad, 91630154

Mohammad M. Mourad, 91330058

Submitted to the School of Engineering of the

Lebanese International University

Tripoli, Lebanon

in partial fulfillment of the requirements for the degree of

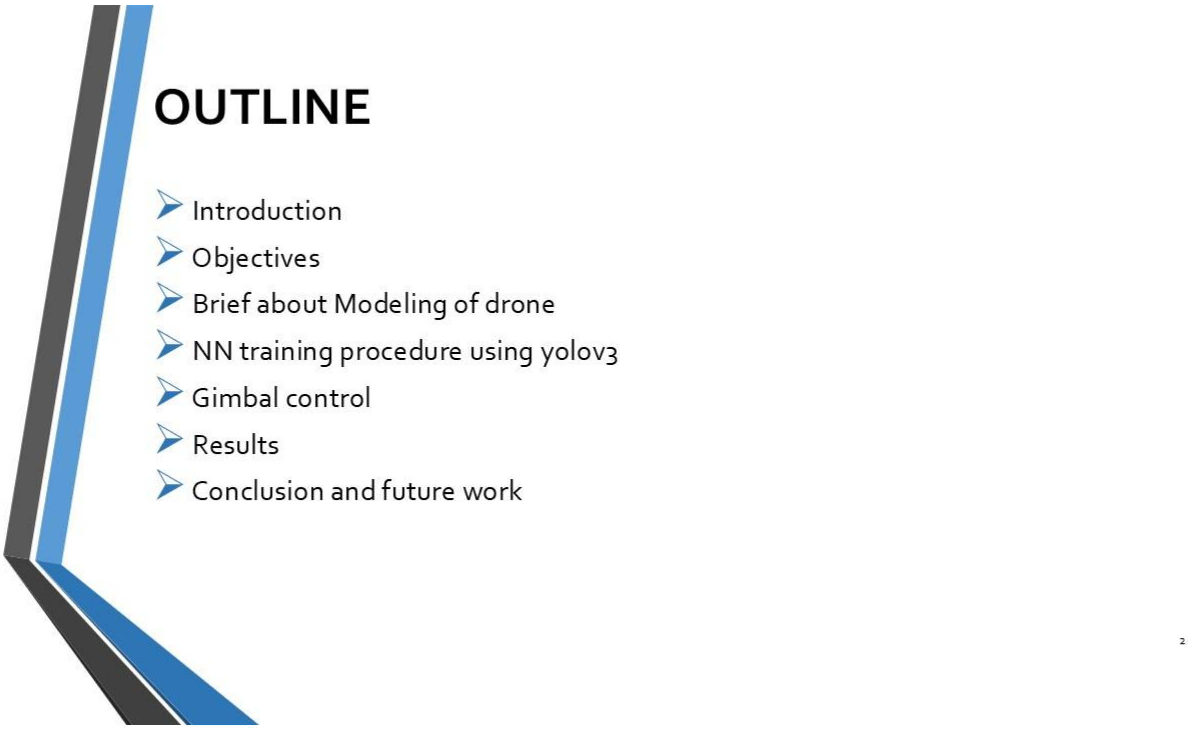
MASTER OF SCIENCE IN ELECTRONICS ENGINEERING

Fall 2020 - 2021

Approved By

Dr. Abdelrazzak Merheb		
Supervisor	**Date**	**Signature**
Dr. Hussein Ezzeddine		
Committee Member	**Date**	**Signature**
Dr. Ahmad Haddad		
Committee Member	**Date**	**Signature**

5.1 Presentation Master Thesis



INTRODUCTION

- The advancement of industry in the last century increased the environmental pollution and climate change.
- The fire detection system using drones has gained a huge turnout especially in the past two decades.
- A Convolutional Neural Network (CNN) is a class of deep neural networks for learning frame works.
- CNNs are used for image recognition and classification

3

OBJECTIVES

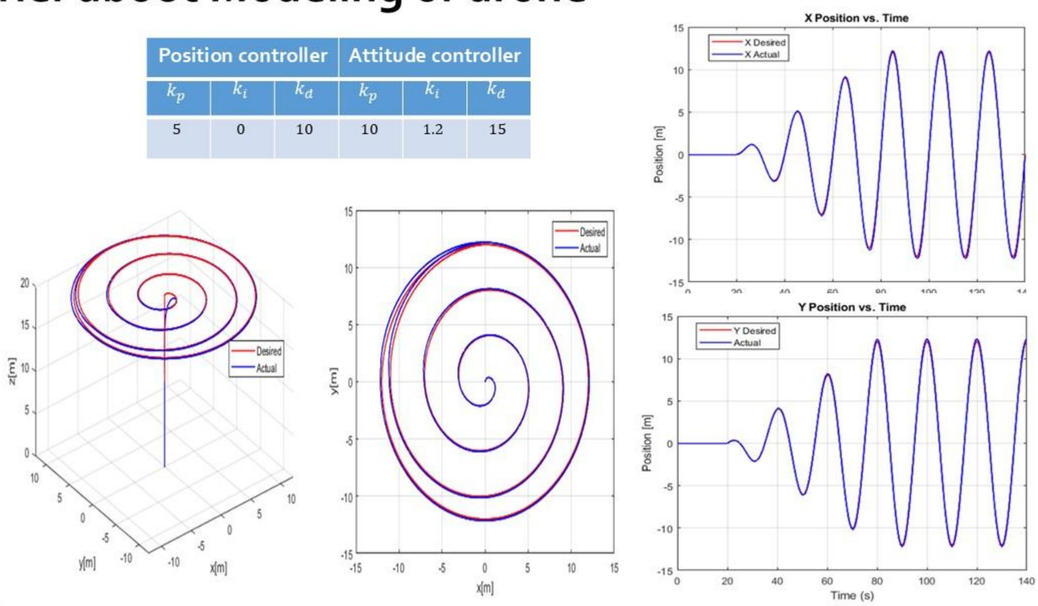
To reduce false alarms and determine the locations of wildfires

- Design and implement an early fire detection system, with the aid of quadcopter based on Machine Learning (ML)/Neural Networks (NN).
- Applying a NN training using YoloV3 to detect smoke and fire.

4

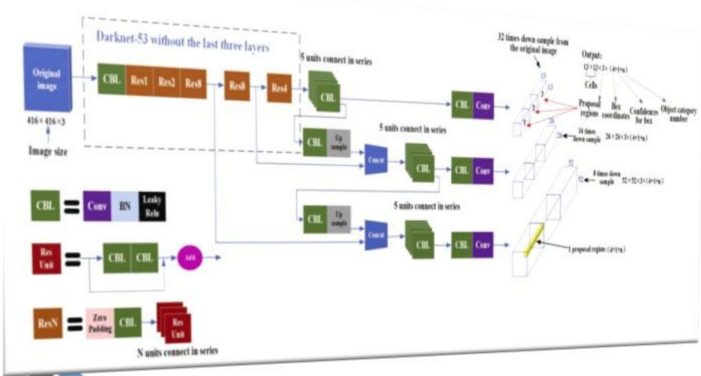
Brief about Modeling of drone

Position controller			Attitude controller		
k_p	k_i	k_d	k_p	k_i	k_d
5	0	10	10	1.2	15



NN training procedure using yolov3

YoloV3

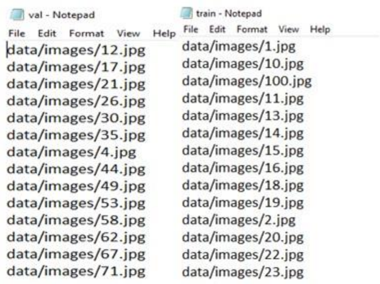


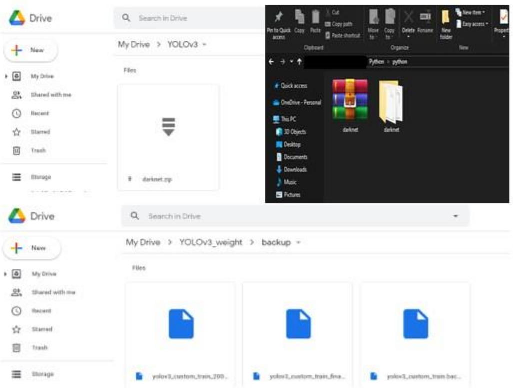
- YoloV3 is an object detection network used to detect classify images .
- YoloV3 network can be trained to detect multiple objects in on frame.
- YoloV3 network accepts input images with dimensions of 416 ×416.
- YoloV3 Architecture

6

DARKNET

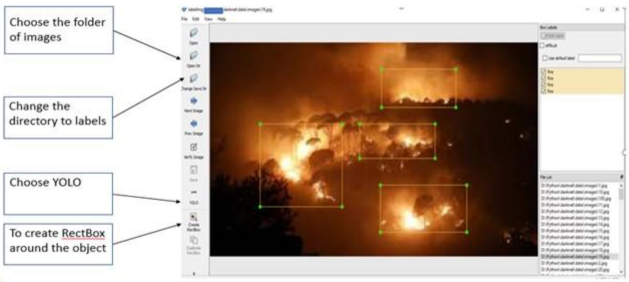
- Darknet is an open source neural network.
- Edit the Makefile in the Darknet folder.
- Applying the filter respect to number of classes.(filters = (classes + 5) *3).
- From directory darknet\examples folder, open file "detector.c" . At line 138, modify this line: if (i%1000==0 || (i < 1000 && i%100 == 0)).
- Split our dataset into a training set and validation set.
- The file has been darknet.zip format and upload it in google drive

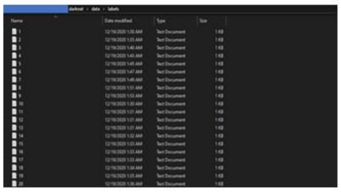




LABELING

- Training the CNN-based algorithm requires a huge amount of data. Therefore 1200 fire and smoke images were collected.
- Using Labeling image annotation tool, each image in the dataset is annotated with a bounding box around fire and smoke
- When finishing, each image will be associated with a label file which defined by the following syntax: <object-class> <x_center> <y_center> <width> <height>

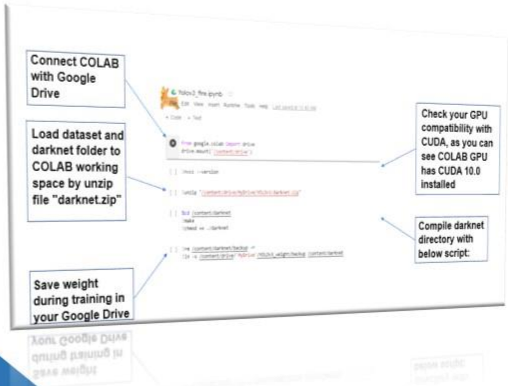




```
0 0.478182 0.418033 0.469091 0.617486
1 0.829091 0.306011 0.341818 0.590164
1 0.241818 0.278689 0.410909 0.546448
```

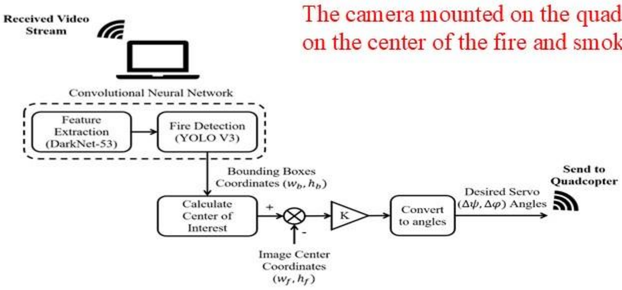
GOOGLE COLAB

- Free online cloud-based Jupyter notebook environment get free GPU.
- This notebook illustrates the step-by-step training procedure and explanation of each code cell.



GIMBAL CONTROL

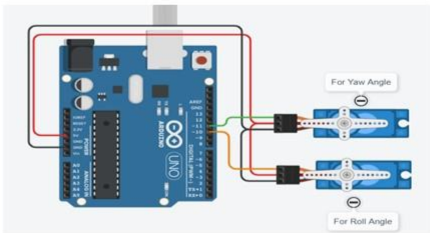
The camera mounted on the quadcopter will always concentrate on the center of the fire and smoke.



Gimbal Control Strategy

$$w_c = \frac{1}{n}\sum_{i=1}^{n} w_i$$

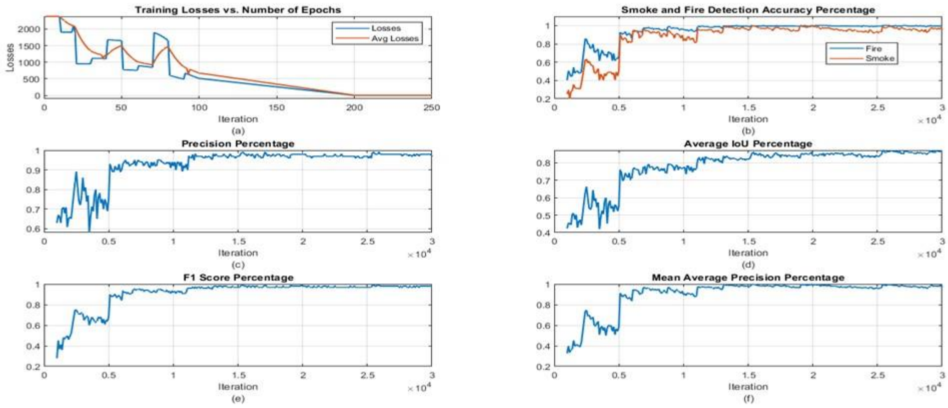
$$h_c = \frac{1}{n}\sum_{i=1}^{n} h_i$$



circuit diagram

10

RESULTS

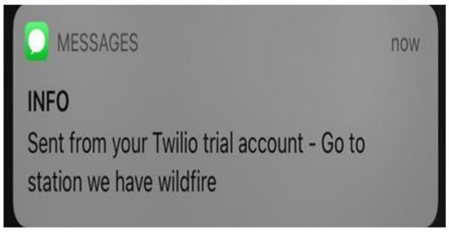


Training Losses (a), Detection Accuracy (b), Training Precision Percentage (c), Average Intersection Over Union (d), F1 Score (e), and Mean Training Precision(f)

11

```
def Alarm_SMS():


    account = "ACd59961ad43187531f9ad1d9135606e70"
    token = "233fea6edc77cf9c05646aa58b29a618"
    client = Client(account, token)

    message = client.messages.create(to="+96176145559", from_="+12565302446",body=" Go to station we have wildfire  ")
```

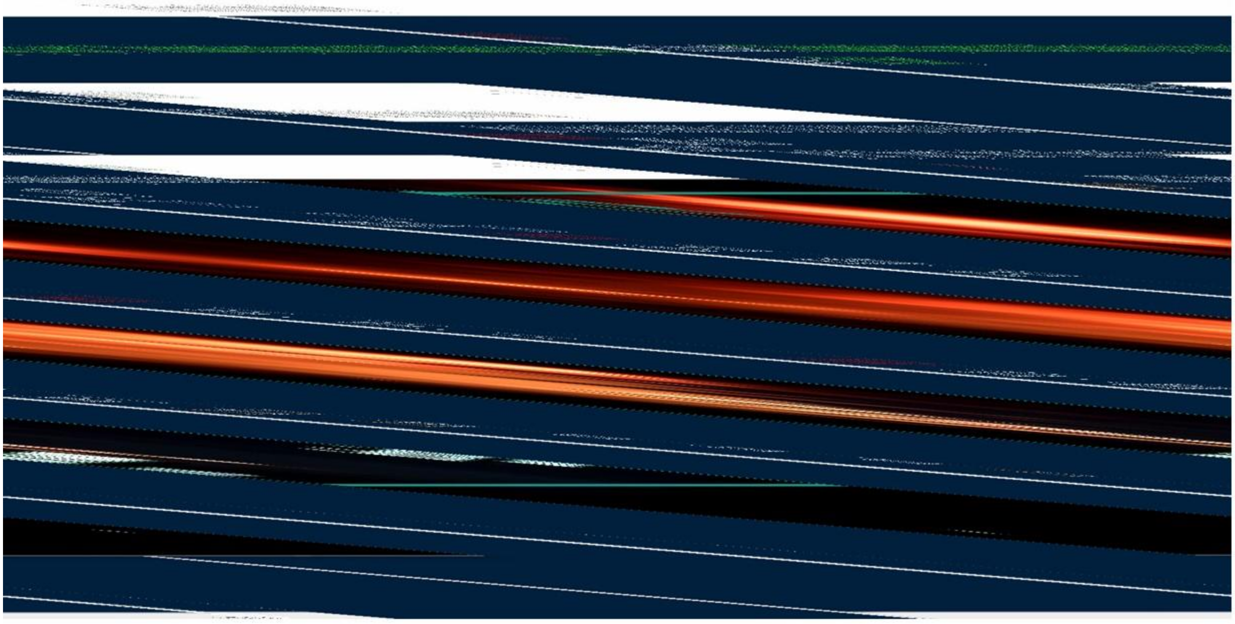
SMS Function



SMS Message on device

12

Fire and smoke detected from video



CONCLUSION

- ✓ Running the fire and smoke detection system on a GPU is 7 times faster than running it on a CPU. A significant fps drop resulted from running the system on CPU.
- ✓ Training YOLO v3 models is much flexible compared to other types of models such as RCNN, faster RCNN …
- ✓ Training the model on 5000 iterations gave significantly unsatisfactory detection accuracy of 55% for smoke and 65% for fires. The model needed to be trained at a minimum number of 11,000 iterations to exceed the 90% accuracy line.
- ✓ The propagation speed of the fires can be estimated for extra monitoring information that can help locals to better take decisions.
- ✓ The system can be trained to detect humans in video frames and send special kind of alert to the stakeholders.

15

THANK YOU FOR LISTEN

16

LEBANESE INTERNATIONAL UNIVERSITY

الجامعة اللبنانية الدولية

5.2 Dedication

This work cannot be done without the support of a lot of people and in particular our friends and families and all our teachers who they taught us. Of course we must dedicate this project to all the people who helped us and especially our parents who drive us to this success, and the thanks go also to the Lebanese International University with its entire staff and faculty members especially our advisor Drs. Abdelrazzak Merheb, Samir Mourad, and all who helped us to complete this project.

5.3 Acknowledgements

We did a lot of work to complete this project in the best way but of course this project cannot be accomplished without the help of many people.

First we must send our deep gratitude to the man who helped us to achieve what we have archived so far; to Dr. Abdelrazzak Merheb who helped us in every step in this project and who gave us the moral and practical support. We cannot forget the efforts of our doctors Especially Dr. Samir Mourad for his valuable time.

Finally, the thanks go for the people who spent his life for our happiness and success our parents and cousin who always lifted up our spirit to do the best work and to achieve this project and of course without those people it was impossible to be in this place today.

5.4 Abstract

This thesis presents the design and implementation of a forest fire and smoke detection system based on machine learning. The system will receive live video stream from a flying quadcopter over a forest region and detects the fire and smoke in the received frame. The system uses Artificial Neural Networks (ANNs) based on Darknet-53 and YOLO V3 pretrained network with 1200 train images and 200 test images for model validation. The base station is composed of a laptop PC which will receive a live video-stream from a quadcopter flying over a certain region of forests and then run the detection algorithm on CPU. To test the model accuracy, virtual fires and smokes are simulated in a recorded drone live footage. The system effectively detected fire and smoke in a video-stream with accuracies of 98% and 96% respectively.

5.5 List of Figures

5.6 List of Tables

5.7 List of Symbols and Abbreviations

ANN: Artificial Neural Networks

CPU: Central Processing Unit

EFFIS: European Forest Fire Information System

ESC: Electronic Speed Controller

FFDI: Forest Fire Detection Index

FPS: frames per second

GPS: Global Positioning System

GPU: Graphical Processing Unit

Li-ion: Lithium ion

LiPo: Lithium Polymer

ML: Machine Learning

NN: Neural Networks

PID: Proportional, Integral and Derivative

RCNN: Region-based Convolutional Neural Network

UAV: Unmanned Aerial Vehicle

5.8 Introduction

5.8.1 Background

The advancement of industry in the last century increased the environmental pollution and climate change in many regions of the world. This led to rising global temperatures which is one of the reasons for the outbreak of fires in large areas of forests [1] [2]. In 2013, the US had lost 104,131 Hectares of forest due to fires [3]. According to the European Forest Fire Information System (EFFIS) report, the Middle East and North Africa have lost at least 176,116 Hectares of Forest in 2014 [4].

In 2019, a series of fires broke out in the forests of Lebanon and nearby countries with nearly 100 fires on the Lebanese territories, according to the Lebanese Civil Defense [5]. In 2020, and based on the data and estimates of the municipalities in the villages, the area burned in this incident has reached 12 million square meters [6]. Unfortunately, this happens every year in Lebanon. Meteorological experts stated that the fires were caused by high temperatures, which reached 38 degrees Celsius (nearly 10 degrees above the average), and dry winds that contributed to forest fires [7].

Problems caused by the lack of technological aspects in the field of fire detection are obviously disastrous. There is still no way to know when are the early flames that caused the fire ignited. Because there are no methods used in Lebanon that keep on checking and identifying the forests' status, it is always too late to prevent forest fires that happen every year. These fires contribute to a great physical, climate, and economic losses. Thus, it is a must to develop monitoring systems that detect fires, especially early flames, and show their status (level, direction, speed). This will help firefighters in accessing the stage safely and treating fire quickly and efficiently which will definitely save more lives and green spaces. Unfortunately, traditional fire detecting sensors, such as ionization smoke sensors and flame detectors frequently lack efficiency when stationed in nature, and often have false alarms. This raises an urgent need for a better and more accurate technology.

The fire detection system using drones has gained a huge turnout especially in the past two decades. This system has been gradually developed throughout the years, starting from remote-controlled drones with smoke sensors, to installing cameras on autonomous drones that take pictures of the fire and transmit live video stream of the fire and how it is moving. This technology helped NASA to detect effectively the California wildfire in 2008 and helped preventing it from spreading in an uncontrolled manner [8]. On one hand, using drones to detect fires has positive aspects. They help firefighters to specify the state of the fire like its direction and extension. Besides, their convenient size helps them penetrate through areas unreachable by pilots. Drones also cost less than helicopters and decrease human losses by being their substitute. On the other hand, there are drawbacks. Being always available and hovering above forests are constrained by their low flying time and some technical risks.

Firstly, continuous use of quadcopters results in less efficient motors, this is called motor aging. Motors will lose their power as they "grow up" which may cause undesirable behavior of the drone and maybe crashing. Moreover, quadcopters' communication channels maybe interfered with other signals. This risk causes loss of information between sender and receiver and may end up with a disastrous situation. In addition, and one of the most common risks, sudden power death due to battery deficiency may occur and crashing will defiantly occur. These risks should be taken into consideration and never be ignored in the design stage of a quadcopter.

5.8.2 Problem Statement

Lebanon lacks advanced fire detection technologies that allow the early detection of fires and provide their state. Due to this deficiency in information, firefighters are exposed to great risks that threaten their lives yearly. Unfortunately, there is no safe way for firefighters to approach the fire when they do not have enough information about its size and its propagation. A large number of people and animals get either highly injured or killed as well. In addition, fires highly increase air pollution and destroy millions of hectares of forest land. Eventually, fires contribute in a great deal of economic and financial losses. Take the wildfire that broke in 2019 in the Lebanese forests for instance. This fire caused dangerous injuries for more than 88 civilians [9] and 5 firefighters, and burnt at least 4 houses in the surrounding area [10]. 3700 acres of green land became ashes in 48 hours, which means the loss of thousands of olive trees and other fruity trees that their owners depended on for their income [9]. Usually, the detection of wildfires is late due to either false alarms or the complete absence of alarms in some forests. This partially eliminates the possibility to access the terrain which allows the fire to nurture by the various fuel sources in the forest. Thus, the need for an efficient way to detect early fires is extremely necessary to avoid further losses.

5.8.3 Thesis Objective

As stated earlier, and due to the tremendous need of an early forest fire detection system in Lebanon, this thesis is dedicated to design and implement an early fire detection system, with the aid of quadcopter, based on Machine Learning (ML)/Neural Networks (NN).

The system is responsible to detect fires in video frames received from quadcopters flying over a region of forest. For this reason, a "Yolov3" neural network model will be used to detect fire in a frame. This model has been chosen for its high training accuracy and the ability to work with low-cost computers. The model will be trained using 1000 of positive images (images with fires). The labeling of images will be done using *"labelImg"* software. Model training will be handled over *"Google Collabs"* that offers a free GPU. In addition, a simple proportional controller will be used to calculate the necessary actions for a 2-D gimbal in order to keep the camera focused on fire region.

5.8.4 Literature Survey

There are a lot of fire detection algorithms and techniques used by scientists and engineers to help in early fire detection and monitoring processes using UAVs. For example, Chi Yuan et al. [11] did effectively extract and track fire pixels in an infrared video sequence received from a UAV. They used brightness and motion clues along with image processing histogram segmentation to extract hot object regions. They also used optical flow sensors to calculate motion vectors of these hot candidate regions. Another work done by Casbeer et al. [12], where they explored the feasibility of a short term, low altitude UAV team to cooperatively track and monitor forest fires' propagation. They simulated a full 6-DOF dynamic model of the UAVs and some numerical models for forest fire propagation. They did not test it in real fire situations. In addition, Zhou et al. [13] gathered video streams from a UAV and applied orthorectification method of these received images to monitor forest fires. They pointed out some specific problems which should be treated in case of forest fire detection and presented some primary solutions to these problems. However, there were no results presented. Moreover, Mubarak Mahmoud et al. [14] collected 6 videos available online and used image processing algorithms to detect fires. They first applied background subtraction to capture movements within the region detected. Then they converted the moving regions from RGB color space into YCbCr which helped them to apply 5 different fire detection rules in order to separate fire pixels. And finally, they used temporal variation method to distinguish between fire and fire-color objects. They achieved 96.63% accuracy detection rate. Another method proposed by Henry Cruz et al. [15] developed a method, which can be used on UAVs, called Forest Fire Detection Index (FFDI) based on using a new color index. This index is based on vegetation methods to detect flames and smoke. They tested this method upon a database imagery with acquiring very good results of about 96.82% precision accuracy over 960 x 540 pixels samples along with 0.0447 seconds processing speed.

5.8.5 Thesis Outline

The thesis is divided as follows; chapter 2 covers the mathematical modeling and control of a quadcopter drone. Chapter 3 lists all components to be used to fully-implement this project and state their specifications. Chapter 4 clearly defines the procedure followed to train a Neural Network (NN) in order to detect fire in camera frames and the design of the simple control system to drive a camera gimbal to keep tracking fire region. Chapter 5 shows the results and chapter 6 concludes the project.

5.9 Mathematical Model

5.9.1 Introduction

This chapter aims to introduce briefly the basic physics of a quadcopter. Modeling the non-linear dynamics and kinematics of a quadcopter in 6 DOF will be presented. The quadcopter model will be simulated using MATLAB/Simulink to follow a spiral trajectory. Position and attitude angles are controlled via Proportional, Derivative, and Integral (PID) controllers.

The quadcopter produces rotational torques in order for it to navigate in space. These rotation angles can be described as Euler angles in 3D. The rotations about x-axis, y-axis and z-axis are represented as rolling (φ), pitching (θ), and yawing (ψ) respectively as shown in Figure 5.1.

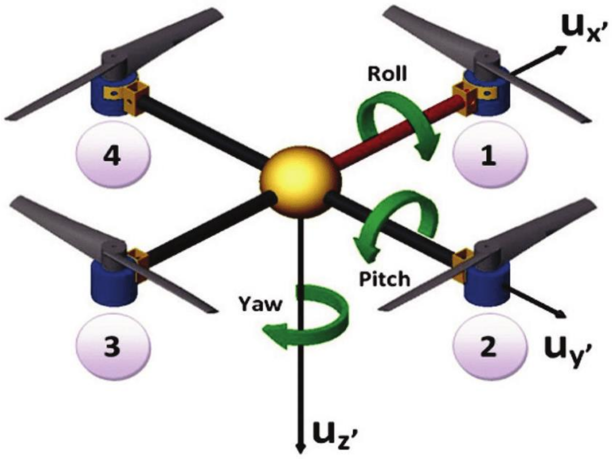


Figure 5.1. Quadcopter rotation angles

Figure 5.2 shows the overall control system structure. The quadcopter is equipped with a high-level (position) controller responsible for tracking a desired trajectory in (x_d, y_d, z_d), generated by the trajectory generator, and a low-level (attitude controller) controller responsible for tracking the rotational setpoints $(\varphi_d, \theta_d, \psi_d)$. The quadcopter then receives 4 control inputs (u_1, u_2, u_3, u_4) which control the throttle force, rolling, pitching, and yawing torques respectively.

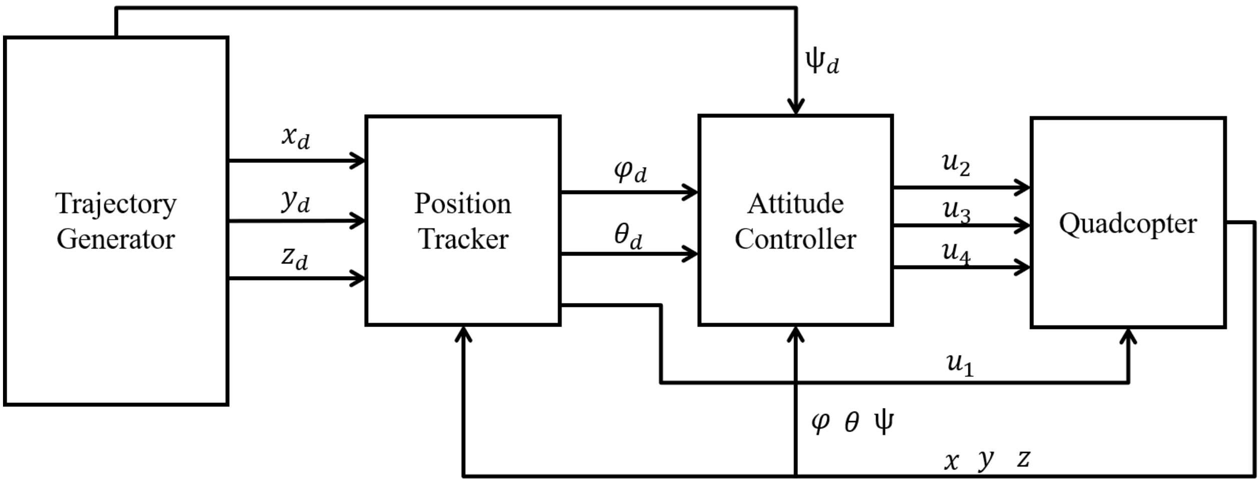


Figure 5.2. Quadcopter high- and low-level controllers

5.9.2 Kinematics and Dynamics

First, it is important to define the reference frames of the quadcopter. There are 2 coordinate frames concerned with quadcopter's movements; body fixed frame $B(x_b, y_b, z_b)$ and the inertial fixed frame $I(x_i, y_i, z_i)$ as shown in Figure 5.3. Quadcopter's aerodynamical forces and actuator forces are all applied with respect to the body fixed frame. However, and in order for the quadcopter to complete its missions in inertial frame (i.e., in real world navigation over the surface of earth), it is important to transform force vectors from inertial to body frame and vice versa. To transform translations from body (x_b, y_b, z_b) to inertial frame (x_i, y_i, z_i), the following rotation matrix is used:

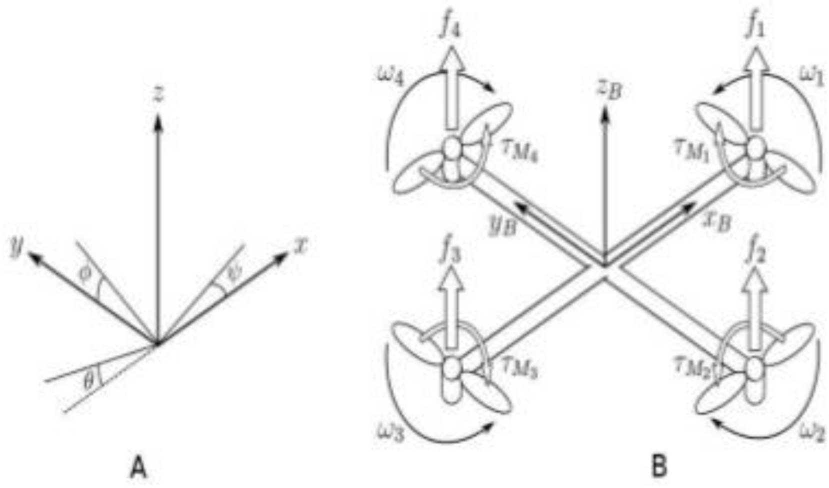


Figure 5.3. Inertial and body coordinate frames

$$\boldsymbol{R}_{body}^{inertial} = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\varphi - s\psi c\varphi & c\psi s\theta c\varphi + s\psi s\varphi \\ s\psi c\theta & s\psi s\theta s\varphi - c\psi c\varphi & s\psi s\theta c\varphi - c\psi s\varphi \\ -s\theta & c\theta s\varphi & c\theta c\varphi \end{bmatrix} \quad (5.1)$$

Where, *c* and *s* stand for *cos* and *sin* functions respectively. Similarly, the relation between Euler rates in inertial frame $(\dot{\varphi}, \dot{\theta}, \dot{\psi})$ and body angular rates (p, q, r) can be determined by equation (5.2).

.

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \boldsymbol{T}\dot{\boldsymbol{\Phi}} = \begin{bmatrix} 1 & 0 & -s\varphi \\ 0 & c\varphi & s\varphi c\theta \\ 0 & -s\varphi & c\varphi c\theta \end{bmatrix} \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (5.2)$$

As a result, the complete kinematic equations, relating body frame translations and rotations with inertial ones can be obtained by:

$$\begin{bmatrix} x_i \\ y_i \\ z_i \\ \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}_{body}^{inertial} & 0_{3x3} \\ 0_{3x3} & \boldsymbol{T}^{-1} \end{bmatrix} \begin{bmatrix} x_b \\ y_b \\ z_b \\ p \\ q \\ r \end{bmatrix} \quad (5.3)$$

On the other hand, dynamics can be divided into 2 subsystems, translational and rotational dynamics, where translational ones are fully interconnected with rotational ones while rotational dynamics are fully decoupled. From Figure 5.3, and applying Newton's second law, the total forces acting on the quadcopter are the upward forces generated from the propellers and the down force from gravity. Forces generated from motors can be written as:

$$f_i = k\Omega_i^2 \quad (5.4)$$

Where k is a thrust constant affected by the dynamics of the propeller and Ω_i is the speed of motor i. Therefore, the translational dynamics, represented in inertial frame, can be written as:

$$\ddot{x} = \frac{1}{m}[(\cos(\varphi)\sin(\theta)\cos(\psi) + \sin(\varphi)\sin(\psi))u_1 - k_{fx}\dot{x}]$$

$$\ddot{y} = \frac{1}{m}[(\cos(\varphi)\sin(\theta)\sin(\psi) - \sin(\varphi)\cos(\psi))u_1 - k_{fy}\dot{y}] \tag{5.5}$$

$$\ddot{z} = \frac{1}{m}[(\cos(\varphi)\cos(\theta))u_1 - k_{fz}\dot{z}] - g$$

Where, m is the total mass of the quadcopter, g is the gravity force, u_1 is throttling control input and (k_{fx}, k_{fy}, k_{fz}) are aerodynamical constants in x, y and z axes respectively and are affected several aerodynamical factors. Similarly, rotational dynamics using Newton's second law can be written as:

$$\ddot{\varphi} = \frac{1}{I_{xx}}[\dot{\theta}\dot{\psi}(I_{yy} - I_{zz}) - J_{tp}\bar{\Omega}\dot{\theta} + u_2]$$

$$\ddot{\theta} = \frac{1}{I_{yy}}[\dot{\varphi}\dot{\psi}(I_{zz} - I_{xx}) + J_{tp}\bar{\Omega}\dot{\varphi} + u_3] \tag{5.6}$$

$$\ddot{\Psi} = \frac{1}{I_{zz}}[\dot{\theta}\dot{\varphi}(I_{xx} - I_{yy}) + u_4]$$

With, (I_{xx}, I_{yy}, I_{zz}) are the moment of inertia about *(x,y,z)* axes respectively. J_{tp} is the total rotational moment around propeller axis and $\bar{\Omega} = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4$ is the total gyroscopic torque acting on the quadcopter. If l represents the arm length of the quadcopter, then the control inputs (u_1, u_2, u_3, u_4) can be given, in function of propellers' speed, as:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} k & k & k & k \\ 0 & -lk & 0 & lk \\ -lk & 0 & lk & 0 \\ -d & d & -d & d \end{bmatrix} \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} \tag{5.7}$$

5.9.3 Controllers

The goal of position and attitude controllers is to calculate the best control inputs (u_1, u_2, u_3, u_4) in order for the quadcopter to complete its mission efficiently, where u_1 controls the upward force (altitude), and (u_2, u_3, u_4) control rolling, pitching, and yawing respectively. For this purpose, PID controller is used to track position and attitude. Figure 5.4 shows the maneuvers in which the quadcopter can achieve by appropriately varying the rotors' speed. u_1 is the control input controlling ascending and descending and can be achieved by varying rotors' speed equally over all propellers (Figure 5.4 (e and f)). Rolling torque, or u_2, can be achieved by achieving different thrust levels between propellers 2 and 4 (Figure 5.4 (c and d)). Pitching torque, or u_3, is produced by also varying propellers 1 and 3 speeds (Figure 5.4 (a and b)). And finally, yawing torque, or u_4, is produced by increasing the speeds of 2 rotors on the same arm and reducing the remaining 2 (Figure 5.4 (g and h)).

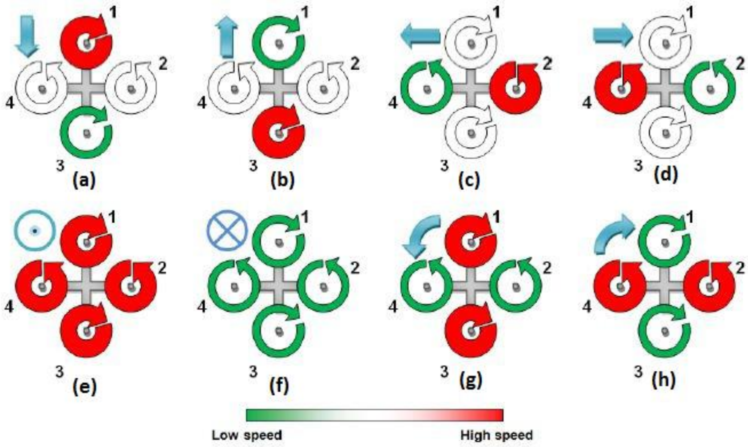


Figure 5.4. Quadcopter maneuvers

PID controller is a simple but efficient controller which is used to control a desired state of a system. It deals with the state error until it converges to 0. PID controller has the shape shown in Figure 5.5.

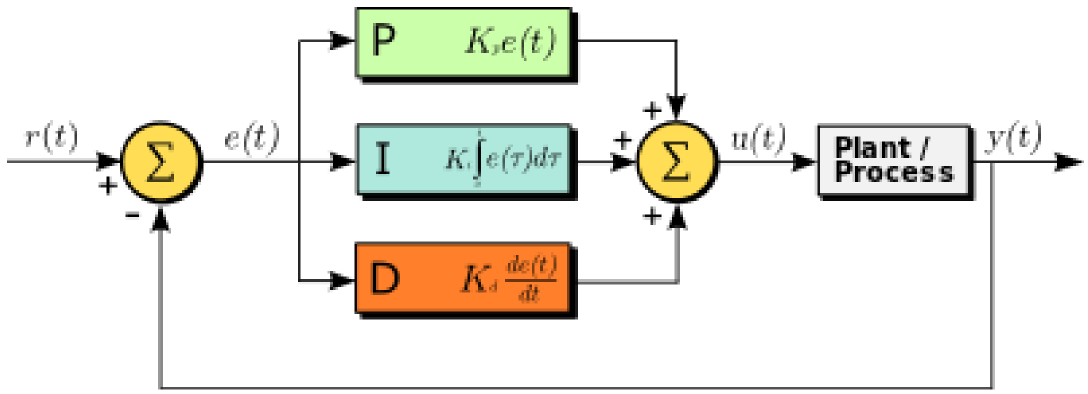


Figure 5.5. PID controller block diagram

The general equation to control a single state *x* using PID can be written as:

$$PID_{out} = k_p e_x + k_i \int e_x dt + k_d \dot{e}_x \quad (5.8)$$

Where, k_p, k_i, k_d are tunable positive gains. e_x is said to be the controlled state error signal of state x and is defined by the difference between the desired state setpoint and the actual value ($e_x = x_{desired} - x_{actual}$). The controlled state x can be any variable such as, position, velocity, acceleration, temperature etc. By tuning the 3 gains, one can achieve the desired behavior of the system which is chosen according to designer's choice.

5.9.3.1 Attitude Controller

The attitude controller is required to maintain or track desired attitude inputs $(\varphi_d, \theta_d, \psi_d)$ and outputs the required roll, pitch, and yaw torques. The closed-loop attitude control diagram can be seen in Figure 5.6. The desired torques can be calculated as:

$$e_\varphi = \varphi_d - \varphi_{actual}$$
$$e_\theta = \theta_d - \theta_{actual} \qquad (5.9)$$
$$e_\psi = \psi_d - \psi_{actual}$$

$$u_2 = k_p e_\varphi + k_i \int e_\varphi \, dt + k_d \dot{e}_\varphi$$
$$u_3 = k_p e_\theta + k_i \int e_\theta \, dt + k_d \dot{e}_\theta \qquad (5.10)$$
$$u_4 = k_p e_\psi + k_i \int e_\psi \, dt + k_d \dot{e}_\psi$$

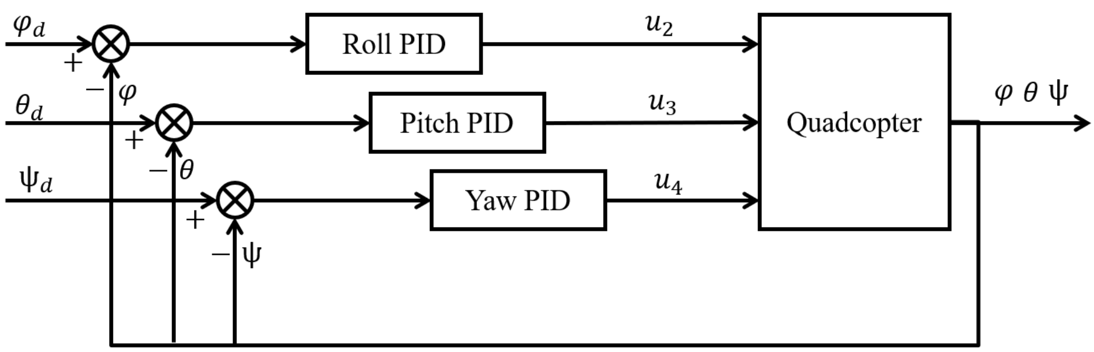


Figure 5.6. Attitude PID controller

Tuning PID gains was done by trial and error until achieving acceptable rise time and overshoot characteristics. The tuned attitude controller gains were found to be $[k_p, k_i, k_d] = [10, 1.2, 15]$. The closed-loop response to a step input for these three controllers is shown in Figure 5.7 and Figure 5.8. As seen in Figure 5.7, the responses of phi and theta systems have approximately a rise time of 0.1 seconds, maximum percent overshoot of 12% for roll system and 5% for pitch system, and 0 steady state error after approximately 3 seconds.

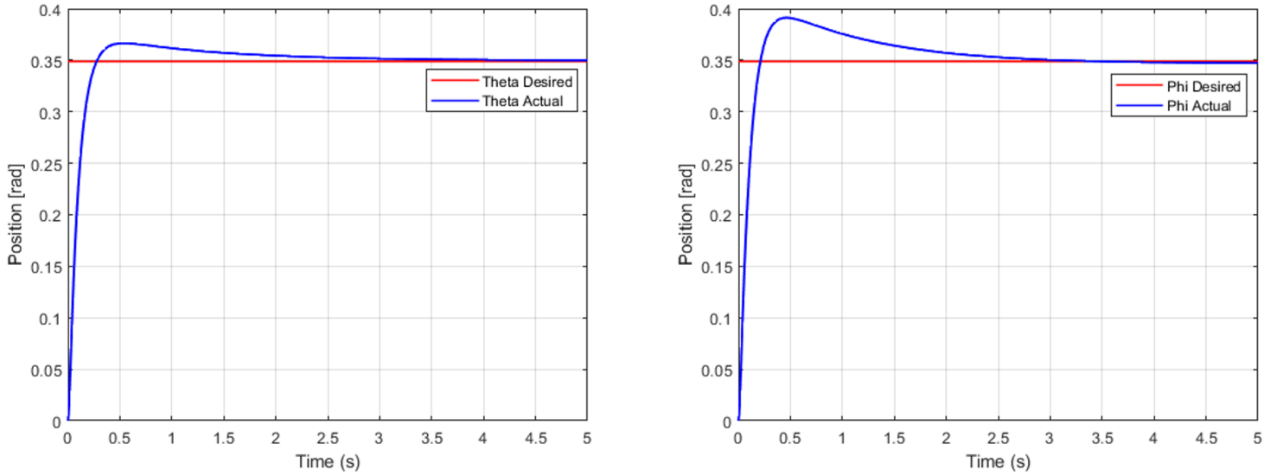


Figure 5.7. Phi (left) and theta (right) response to step input of 20 degrees

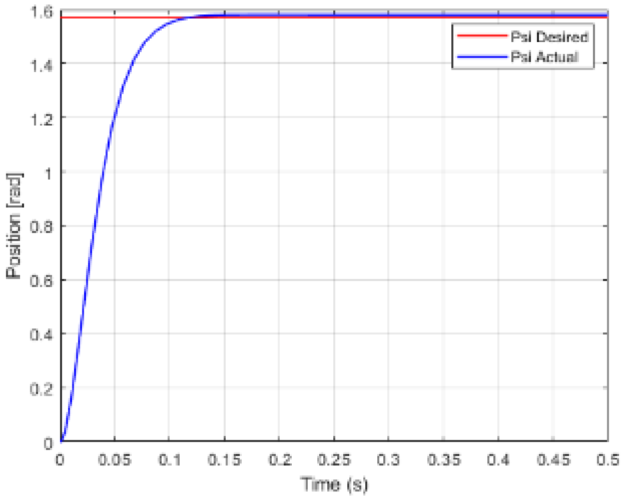


Figure 5.8. Psi response to step input of 90 degrees

5.9.3.2 Position Controller

The position controller forms the high-level controller for the quadcopter system. It takes the desired trajectory from a trajectory generator (x_d, y_d, z_d) and is responsible to output the desired the attitude (φ_d, θ_d) and throttle to achieve translation in space. Figure 5.9 illustrates the PID-based closed-loop position controller. The "Inverse Kinematics" block is responsible to translate the 2D position PID output into the desired attitude values.

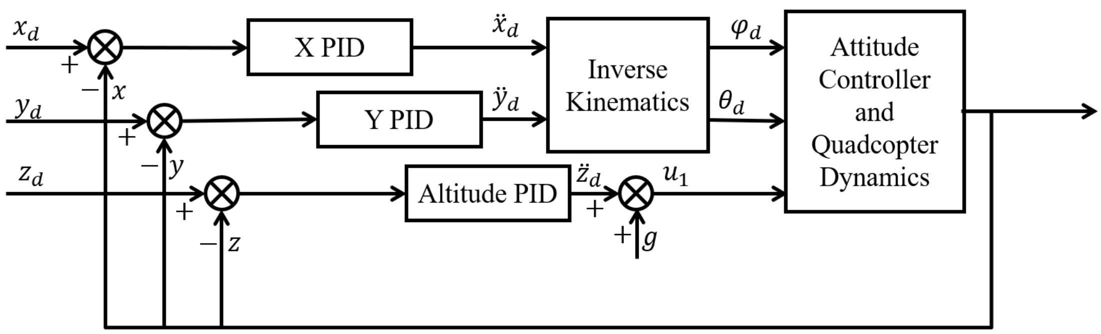


Figure 5.9. Position controller block diagram

From the translational dynamics in equation (5.5), and considering small angles approximation ($\cos(\varphi) = \cos(\theta) = 1 \; and \; \sin(\varphi) \approx \varphi \; and \; \sin(\theta) \approx \theta$), The desired attitude and throttle input can be calculated as:

$$\begin{bmatrix} e_x \\ e_y \\ e_z \end{bmatrix} = \begin{bmatrix} x_d - x_{actual} \\ y_d - y_{actual} \\ z_d - z_{actual} \end{bmatrix} \quad (5.11)$$

$$\begin{bmatrix} \ddot{x}_d \\ \ddot{y}_d \\ \ddot{z}_d \end{bmatrix} = k_p \begin{bmatrix} e_x \\ e_y \\ e_z \end{bmatrix} + k_i \begin{bmatrix} \int e_x \, dt \\ \int e_y \, dt \\ \int e_z \, dt \end{bmatrix} + k_d \begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad (5.12)$$

$$\begin{bmatrix} \varphi_d \\ \theta_d \end{bmatrix} = \frac{1}{g} \begin{bmatrix} \sin(\psi_d) & -\cos(\psi_d) \\ \cos(\psi_d) & \sin(\psi_d) \end{bmatrix} \begin{bmatrix} \ddot{x}_d \\ \ddot{y}_d \end{bmatrix} \quad (5.13)$$

Finally, the simulation was done with$(x, y, z, \varphi, \theta, \psi) = (0,0,0,0,0,0)$ as the quadcopter's initial position. The desired trajectory, which generates the spiral trajectory, is given by equation (5.14). It is worth to mention that the spiral path was chosen in order to make tracking smoother. The simulation time was set to 140 seconds. PID tuned gains are given in Table 5.1. Figure 5.10 illustrates the quadcopter's 3D and 2D position while tracking its desired circular trajectory and Figure 5.11 shows the x and y desired trajectory with respect to time. The desired trajectory was calculated in a way to provide smooth setpoints suitable for UAV navigation.

Table 5.1. Tuned PID parameters

Position Controller			**Attitude Controller**		
k_p	k_i	k_d	k_p	k_i	k_d
5	0	10	10	1.2	15

$$\begin{bmatrix} x_d \\ y_d \\ z_d \end{bmatrix} = \begin{cases} \begin{bmatrix} 0 \\ 0 \\ 10 \end{bmatrix} & , for \; t \leq 20 \\ \begin{bmatrix} \dfrac{t-20}{5\sin(0.1\pi t)} \\ \dfrac{t-20}{5\cos(0.1\pi t)} \\ 20 \end{bmatrix} & , for \; 20 < \; t \leq 80 \\ \begin{bmatrix} 12\sin(0.1\pi t) \\ 12\cos(0.1\pi t) \\ 20 \end{bmatrix} & , for \; t > 80 \end{cases} \quad (5.14)$$

The physical parameters used in this simulation are listed in Table 5.2. These parameters are calculated from the physical properties of the quadcopter used and all parts with their specs are listed in chapter 3.

Table 5.2. Quadcopter simulation parameters

Parameter	Description	Value	Unit
m	Mass of the quadcopter	2	kg
I_{xx}	Moment of inertia about x-axis	0.0641	$kg.m^2$
I_{yy}	Moment of inertia about y-axis	0.0641	$kg.m^2$
I_{zz}	Moment of inertia about z-axis	0.1148	$kg.m^2$
g	Gravity force	9.81	$m.s^{2^{-1}}$
b	Thrust constant	1.02×10^{-6}	$N.m.s^2$
d	Drag constant	1.3×10^{-7}	$N.m^{-1}$
l	Quadcopter arm length	0.275	m
J_{tp}	Total rotational moment of inertia around the propeller axis	104×10^{-6}	$kg.m^2$
k_{fx}	x-axis drag constant	0.00215	$N.m^{-1}$
k_{fy}	y-axis drag constant	0.00215	$N.m^{-1}$
k_{fz}	z-axis drag constant	0.00215	$N.m^{-1}$

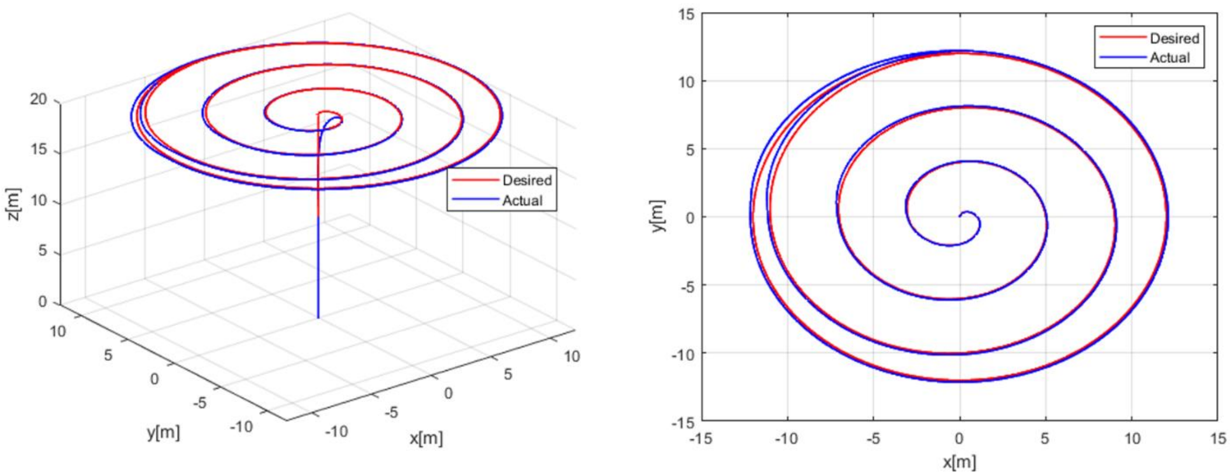


Figure 5.10. Quadcopter tracking spiral trajectory

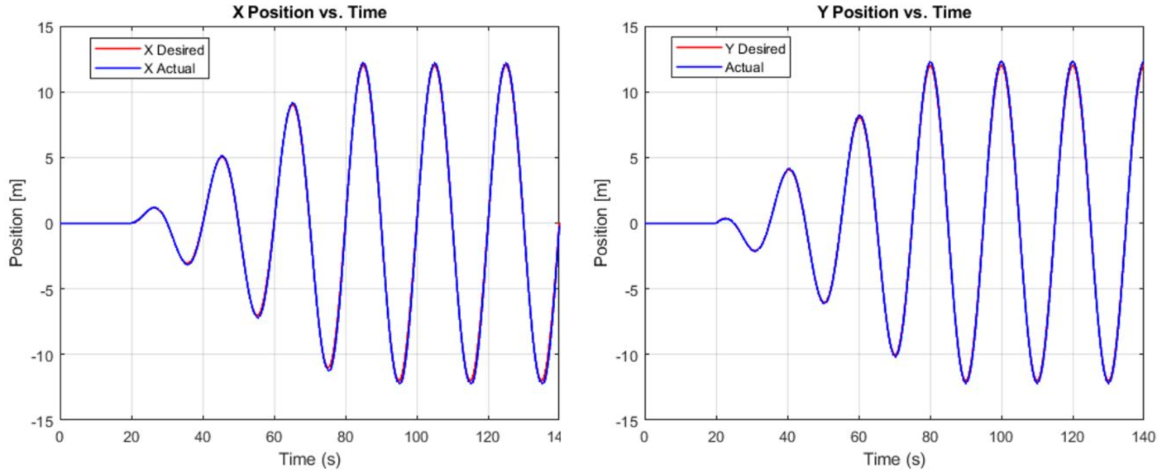


Figure 5.11. X (left) and Y (right) position tracking with respect to time

The above figures show the great tracking results from a simple PID controller. However, this controller is applicable under the small angles' approximation ($-25^o \le \varphi \le 25^o and - 25^o \le \theta \le 25^o$). Also, generating smooth setpoints also plays a major role in efficient, and less-deviated from the desired tracking.

5.9.4 Conclusion

This chapter presented the kinematic and dynamic mathematical model of a quadcopter UAV. Also, it proposed the design of PID position and attitude controllers in order for the quadcopter to track a predefined trajectory. Position controller was able to track a circular trajectory with significantly very small position error which can be accepted.

5.10 Project Specifications

5.10.1 Introduction

In this chapter, hardware and software requirements will be demonstrated. Figure 5.12 illustrates the main parts used in this project. The system components can be divided into 4 main parts: quadcopter, vision, communication and central PC.

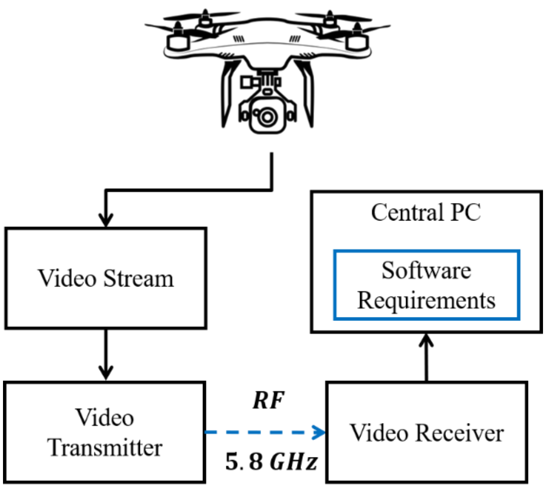


Figure 5.12. Project main parts

5.10.2 Quadcopter

For this project, the required quadcopter should be characterized by its medium size and light weight. The required components along with their quantities to build a quadcopter are listed in Table 5.3.

Table 5.3. Quadcopter parts

Component	Quantity
Quadcopter frame	1
Brushless DC motor	4
Propeller	4
Electronic Speed Controller	4
Flight controller with Inertial Measurement Unit and GPS	1
Power Distribution Board	1
Battery	1
Gimbal	1

5.10.2.1 Quadcopter Frame and Power Distribution Board

The quadcopter frame is used to mount all components together. The primary criteria of selecting the frame are the total payload to be carried and its material. In this thesis, light payload is required since it is only required to carry a small camera and a gimbal. The chosen frame, shown in Figure 5.13 (left), is made of ultra-strong, light weight glass fiber. Its different arms' colors help in better determining the heading of the quadcopter while flying.

In addition, the "*DJI450*" frame also includes a Power Distribution Board (PDB) in order to solder electronic components. The PDB uses a high-strength Printed Circuit Board (PCB) material which makes battery and ESCs wiring very safe and easy. Table 5.4 summarizes the important specs of the selected frame.

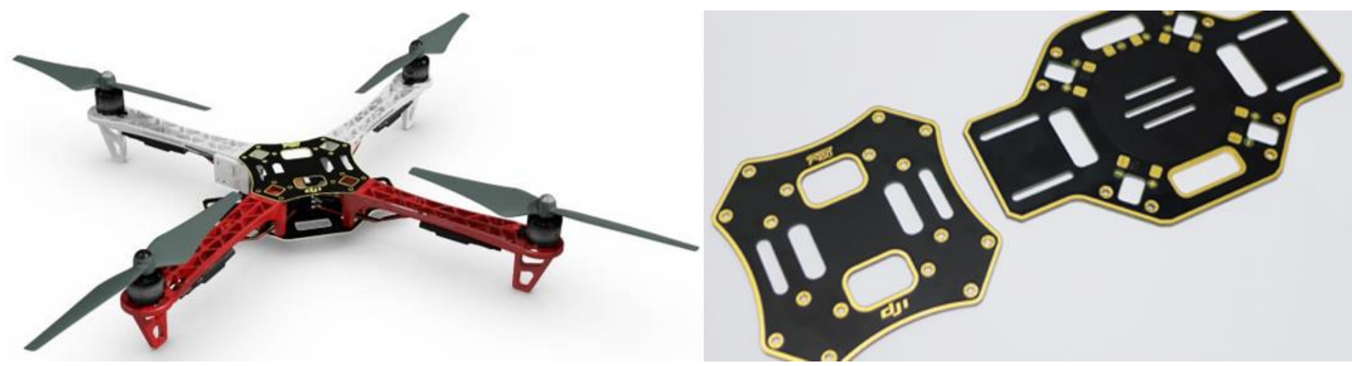


Figure 5.13. DJI450 quadcopter frame (left), Power Distribution Board PDB (right)

Table 5.4. DJI450 frame specs

Parameter	Value
Diagonal Wheelbase	$450\ mm$
Weight	$282\ g$
Material	$PA66\ +\ 30GF\ glass\ fiber$
Recommended battery	$3s \sim 4s$
Recommended motors	$900 \sim 1200\ KV$
Recommended ESC rating	$30A$

5.10.2.2 Brushless DC Motors

Brushless DC motors, or BLDC, is a type of DC motors that is widely used in quadcopters and many other fields. It utilizes a permanent magnet as its external rotor with 3 phase output voltage, unlike brushed DC motors. It is characterized by its very high efficiency (up to 85%) compared with brushed ones. They are selected among their KV ratio, where KV ratio is its $\frac{rpm}{V}$. Higher KV ratio increases the motor's speed but decreases the torque of the motor and decreases the payload lifting capabilities accordingly. For this thesis, and as stated earlier, the required payload is low but traveling speed must be $\leq 5\frac{m}{s}$, so, motors with *1200 KV* ratio are selected as shown in Figure 5.14. Table 5.5 summarizes the most important parameters of the selected motors.

Figure 5.14. PROPDRIVE V2 2826 1200 KV

Table 5.5. PROPDRIVE motor specs

Parameter	Value
Max current	$15A$
KV	1200
Power	$215\ W\ @\ 12\ V$
Weight	$59\ g$

5.10.2.3 Propellers

Propellers are used to transform the rotational movement generated from the BLDC motors into thrust force. The propellers are selected upon their physical shape, length and pitch angle which directly affect the thrust generated. Propellers usually are marked by number that represents its length and pitch angle. For example, a "*1045*" propeller means it has 10 inches length and 4.5 inches pitch angle. The selected propellers for this project are made of plastic and are shown in Figure 5.15.

Figure 5.15. 1045 propellers

5.10.2.4 Electronic Speed Controller

Electronic Speed Controller, or ESC, is an electronic device used to transform Pulse Width Modulation (PWM) signal coming from the flight controller into the appropriate 3 phase voltage to supply the BLDC motors. It is based on MOSFET transistors that switch on and off several thousands of times a second according to their frequency feature. Generally, ESCs are chosen upon their operating voltage, maximum supplied current (must be twice as the required motor's current) and frequency range. The chosen ESC is shown in Figure 5.16 and has the specs listed in Table 5.6.

Figure 5.16. AfroESC 30A

Table 5.6. ESC specs

Parameter	Value
Supply current	Continuous 30A
Voltage range	2 ~ 4s LiPo
Type	Linear Battery Elimination Circuit (LBEC)
Input frequency	Up to 1 KHz
Weight	26.5 g

5.10.2.5 Flight Controller

The flight controller is the brain of the quadcopter. It receives PWM input from the user (usually from remote control) and interpret multiple sensor inputs to help maintaining a stable flight and outputs the desired PWM signals for the ESCs to control the speed of motors. They are characterized by their control algorithms used, power consumption, user assistant software, decisions in case of failures, etc. There are plenty of flight controllers available in the market: Multiwii, KK multicopter, Pixhawk, naze32, Ardupilot, Naza collection, etc. Each is characterized by its own settings. One of the best flight controllers is the "*Naza-M v2*" made by DJI [16] and is shown in Figure 5.17. It supports 9 different quadcopter configurations as it includes a GPS/compass module for better positioning. It has a take-off assistant algorithm which helps to ensure a drift-free take-off even if the quadcopter is started from un-leveled ground. In addition, it has multiple features that help increase the flight safety such as: low voltage protection, enhance fail-safe, return to home, and Intelligent Orientation Control (IOC). Table 5.7 lists all of Naza's flight controller specs.

Figure 5.17. Naza-M v2 flight controller

Table 5.7. Naza-M v2 specs

Parameter	Value
Supported ESC output	400 Hz refresh frequency
Input voltage	2s ~ 6s LiPo recommended
Power consumption	1.5 W @ 5V max 0.6 W @ 5V normal
Hovering accuracy (GPS mode)	Vertical: ±0.8m Horizontal: ±2.5m
Max angular speed	200^o/s
Ascent/descent speed	Ascent:6 m/s descent:4.5 m/s

5.10.2.6 Battery

The battery is the power source used to power up all components onboard. There are mainly 2 types of batteries used to power up components on quadcopters, Lithium ion (Li-ion) and Lithium Polymer (LiPo). Both are known for their high durability and robust performance. However, Li-ion batteries suffer from capacity decrease over charge cycles or even when it is not used while LiPo batteries have more stable performance but it is more expensive. In this project, LiPo battery is chosen and is selected upon the desired flight time and can be calculated using the following formula:

$$flight\ time\ (in\ minutes) = \frac{Capacity\ (Ah) \times 0.8}{Total\ load\ (A)} \times 60 \qquad (5.15)$$

Where the battery capacity is expressed in Ampere hours (Ah), 0.8 is the efficiency of the LiPo battery (80%) and the total load consumption is expressed in Amperes (A) which can be obtained by adding all current consumption of each component and is approximated to be 60A. This total load cannot be checked during flight, however it is assumed to be constant. For this project, a 5000 mAh LiPo battery was chosen and is shown in Figure 5.18 with its specs listed in Table 5.8.

Figure 5.18. HRB 5000 mAh LiPo battery

Table 5.8. HRB LiPo battery specs

Parameter	Value
Output Voltage	3s (1s = 3.7/4.2V: discharged/charged)
Capacity	5000 mAh
Continuous discharge	50C
Wire gauge	10 AWG
Weight	376 g

5.10.2.7 Gimbal

The camera gimbal is a device used to control the movement smoothly without producing vibrations. It is mounted on the quadcopter to provide smooth video output from the camera (vibration-free) and is driven by 3 brushless motors to stabilize the camera's position in 3 directions (roll, pitch and yaw). The selected gimbal is shown in Figure 5.19 and has the specs listed in Table 5.9.

Figure 5.19. Gidy camera gimbal

Table 5.9. Gidy camera gimbal specs

Parameter	Value
Pitch	-90º to+30º
Roll	0º or 90º (horizontally and vertically)
Stabilization	3 axes (pitch, roll, yaw)
Weight	158.757 g

5.10.3 Vision

The appropriate selection of vision system, or camera is very crucial for NN application. There are several camera factors that will affect the performance of fire detection algorithm which are frame rate, camera resolution, field of view, and ISO range. Fps drop depends on the PC used, filters applied, and number of objects detected in a single frame. The camera selected for this project is built-in with the gimbal device shown in Figure 5.19 and its specs are listed in Table 5.10.

Table 5.10. Camera specs

Parameter	Value
Sensor	1/2.3’’ (CMOS), 12.35 MP
ISO range	Video: 100-3200 Photo: 100-1600
Image size	4K – 4000×3000
Video recording modes	**C4K**: 4096×2160 24p **4K**: 3840×2160 24/25/30p 2.7K: 2720x1530 24/25/30p **FHD**: 1920×1080 24/25/30/48/50/60/96p **HD**: 1280×720 24/25/30/48/50/60/120p

5.10.4 Communication

The communication system includes the video transmitting and receiving units. This unit is responsible to transmit live video streams from quadcopter and receive them on the central PC in order for these frames to be processed. These devices are chosen according to their Radio Frequency (RF) bandwidth, power consumption, sending rate, and sending range. For this project, it is desired to have a video-stream transmission within 1.5 to 2 km. Thus, the selected communication system is shown in Figure 5.20 and its specs are listed in Table 5.11.

Figure 5.20. AKK video transmission system

Table 5.11. Communication system specs

Parameter	Value
Sending range	2000 m and ≥3000 m in open areas
Number of channels	40 covering bands A, b, E, F, r
Operating voltage	7-16V
Power consumption	0.22/0.65A: non-transmitting/transmitting @12V
Video format	NTSC/PAL
Weight	85 g

5.10.5 Central PC

This is the most important component in this thesis. The central PC is responsible for all fire detection process. It will receive raw images from the quadcopter flying over a forest region, then it will process these incoming frames and detect fires using Artificial Neural Networks (ANN). Fire detection was done on a central computer since ANN are power-greedy and will noticeably reduce the flight time if calculated onboard. Choosing the right PC for this application is a bit expensive.

5.10.5.1 Hardware Requirements

The chosen PC is based on CPU not GPU and all fire detection scripts will be run by, which causes significant frames per second (fps) drop when detecting fires in frame. GPU has very small fps drop when running fire detection scripts on, since it can handle more graphical information than CPU but it is very expensive. However, CPU-based PC was selected upon its availability, cheapness and is shown in Figure 5.21 with its specs listed in Table 5.12.

Figure 5.21. HP laptop 15-da1xx

Table 5.12. HP laptop specs

Parameter	Value
Processor	Intel® core™ i5-8265U
Ram	8 GB
System type	64-bit Operating System (OS)
Windows	10

As mentioned before, the training will be held over "*Google Collabs*", which is a python development environment that runs in any browser, because it offers a GPU rather than CPU which is way better to train a network over. Training a NN over a CPU would take approximately 4-5 times more time than training it over a GPU, according to Buber et al. [17]. "*Google Collabs*" uses the hardware specs listed in Table 5.13.

Table 5.13. Google Collabs hardware specs

Parameter	Value
GPU	Nividia k80/T4
GPU memory	12GB/16GB
GPU memory clock	0.82GHz/1.59GHz
Performance	4.1 TFlops/8.1 TFlops
Number of CPU cores	2
Available RAM	12GB (upgradeable to 26.75GB)
Disk space	358GB

5.10.5.2 Software Requirements

This section will show the necessary software applications to be installed on the central laptop in order to setup and run all required files to detect fires in video frames. There are mainly 2 required software applications that must be installed (all installation procedure is shown in Appendix section).

Python

Python is a high-level and general-purpose programming language which is getting more popular over time. It is widely used in Machine Learning (ML) applications because it offers very easy and ready-to-use libraries. It includes a very famous computer vision library (OpenCV) that provides a very large image processing functions. For this project, the required libraries to run the fire detection code are: *"OpenCV"* and *"Numpy"*.

Figure 5.22. Python logo

LabelImg

LabelImg is an interactive image annotation tool written in python. It will be used to label fire regions in the training data set (1000 images). LabelImg is one of many image annotations tools but it is selected for its simplicity and *"YOLOv3"* network friendly. The procedure of using this software application and setting up training set will be explained in chapter 4.

Figure 5.23. LabelImg logo

5.10.6 Conclusion

In this chapter, all hardware and software requirements to build an early fire detection system using NN were shown. Python scripts for detecting fires will be handled by a CPU-based PC but training NN will be held by a GPU-based hardware offered from *"Google Collabs"*. In the next chapter, NN training procedure, using *"YOLOv3"* model to detect fires in video frames will be demonstrated.

5.11 Design

5.11.1 Introduction

In this chapter, the procedure to train a neural network using YoloV3 models to detect forest fires will be demonstrated. A brief introduction for Neural Networks (NN) will be presented and the necessary parameters that have to be tuned during training will be highlighted.

5.11.2 Convolutional Neural Network

A Convolutional Neural Network (CNN), which is also known as Multi-Layer Perception (MLP), is a class of deep neural networks for learning frame works. The first known CNN was introduced by

LeCun in 1990 and is called LeNet [18]. CNNs, unlike feedforward Networks, are used for image recognition and classification. Image classification based on CNNs can optimally and automatically learn to extract image features effectively. Figure 5.24 illustrates the flow of CNN-based fire detection algorithms. The detection CNN has region proposals, feature extraction, and image classification functions. The first step consists of the CNN accepting an image as an input and outputs region-based proposals by 2 main layers; convolution and pooling. Then, it is the turn of region-based fire detection CNN to decide whether there is fire or not in proposal regions through convolutional, pooling, fully-connected layers.

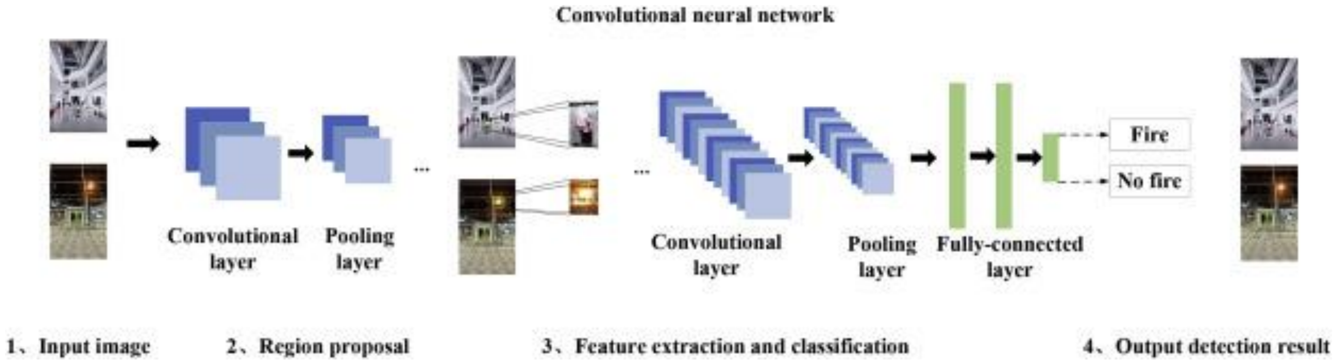


Figure 5.24. Convolutional Neural Network architecture

The convolutional layer is the most essential part in any CNN. Ordinary Neural Networks (NN) use connection weights, biases and weighted sums while convolutional layer is equipped with image transform filters also known as convolutional kernel in order to generate feature map of the original image. So, a convolutional layer is nothing but a set of convolutional kernels. The kernel slides over the whole image and computes a new pixel by some sort of weighted sum of the pixel which is floating over, in order to generate a full feature map. Equation (5.16) shows the main calculation formula of the convolutional layer.

$$y = \sum_{j=0}^{J-1} \sum_{i=0}^{I-1} w_{ij} \cdot x_{m+1,n+j} \cdot + b, \qquad (0 \leq m \leq M, 0 \leq n \leq N) \tag{5.16}$$

Where x is the input image of size $W \times H$, w is defined as a convolutional kernel of size $J \times I$, b is a bias value and y is the output of the feature maps. Practically speaking, w and b values are determined optimally through training process. Pooling layer samples the feature map acquired from the convolutional layer attempting to significantly reduce the overfitting, the number of parameters, and the computation in a CNN. Lastly, the fully-connected layer produces the final classification vector. It is connected to every single neuron in the layer before it decides the existence of possible matching between combination of features found and class labels.

5.11.3 Yolo V3

You Only Look Once Version 3 (YOLO V3) is an object detection network which is used after a feature extraction network to detect classify images with fires and smoke created by Joseph Redmon and Ali Farhadi in 2018 [19]. The feature extraction layer uses Darknet-53 network. YOLO V3 refers the idea of residual network to improve the accuracy of object detection. In addition, this network performs perfectly on detection speed for it uses a one -stage strategy. Figure 5.25 shows the detailed architecture of YOLO V3 network. The feature extraction network, or Darknet-53, generates a small-scale feature map which 32 times smaller than the original sampled images. Typically, YOLO V3 network accepts input images with dimensions of 416×416 so the size of the feature map extracted from the Darknet-53 becomes roughly 13×13. The goal of this small feature map is to detect large objects. Then, YOLO V3 network generates a large-scale feature map by enlarging the small-scale feature map got from the feature extraction network (Darknet-53) and concatenating with an earlier layer-feature map. The large-scale feature map includes information of previous layers and other complex features from deeper layers which are used to detect small objects. Practically, there are 3 scales of feature maps; 8, 12 and 32 time smaller from the original image.

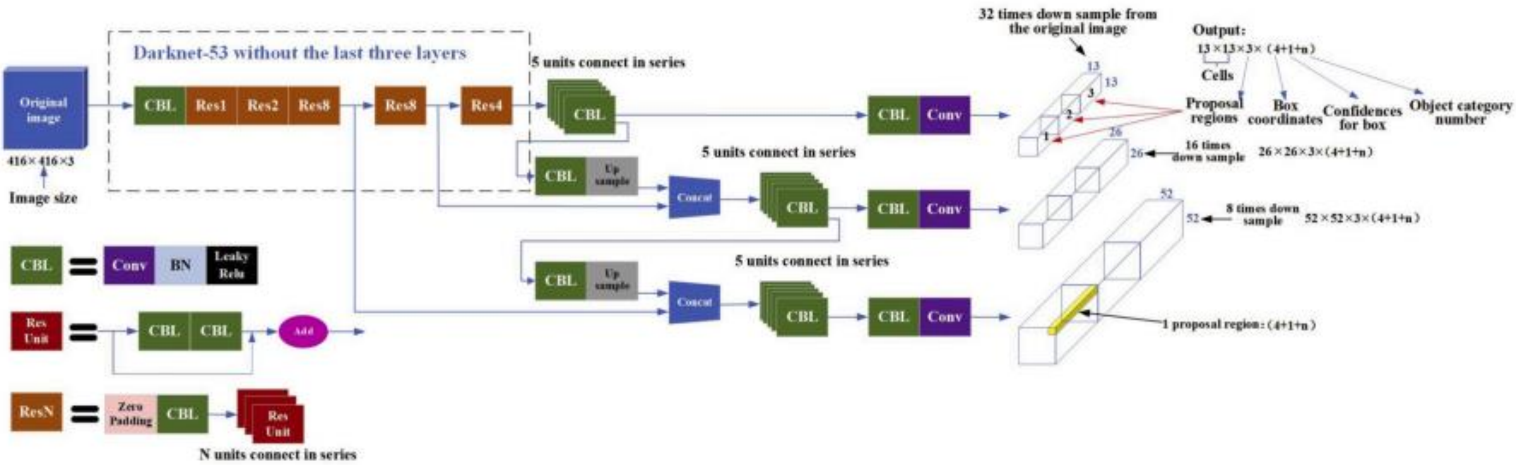


Figure 5.25. Yolo v3 Architecture

In the above figure (Figure 5.25), *N* in *ResN* clarifies that there are *N* number of *Res* units connected in series. Whereas, Concat refers to the concatenation operation which expands the dimension of the feature maps. It noteworthy that concatenation process is different than an ordinary addition operation, the normal addition does not change the dimension of the feature maps. YOLO V3 uses a sigmoid activation function to predict and detect multilabel classifications per one bounding box. The sigmoid function has the form as in equation (5.17) and Figure 5.26.

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{5.17}$$

The main advantage of sigmoid function is that it is a smooth version of an ordinary step function. In other words, it has a derivative everywhere. This is very important in neural networks since the fully-connected layer in a CNN computes the gradients through backpropagation to update the weights.

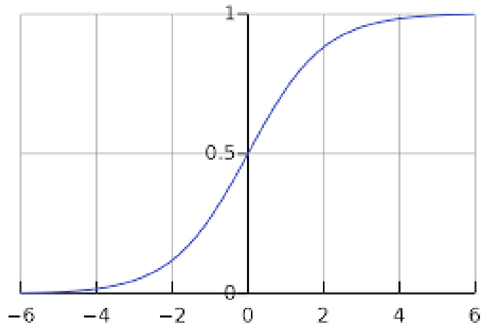


Figure 5.26. Sigmoid Activation Function

So, a YOLO V3 network can be trained to detect multiple objects in on frame. For this thesis, it was trained to detect fires and smoke in a single frame image.

5.11.4 Training and Testing

Training the CNN-based algorithm requires a huge amount of data. Therefore, in this thesis, *1200* fire and smoke images were collected from different internet sources for training and *200* for testing. Using LabelImg image annotation tool, each image in the dataset is annotated with a bounding box around fire and smoke. Figure 5.27 shows the training images setup procedure. Around *4500* fires were annotated in these *1000* images and *3265* smokes.

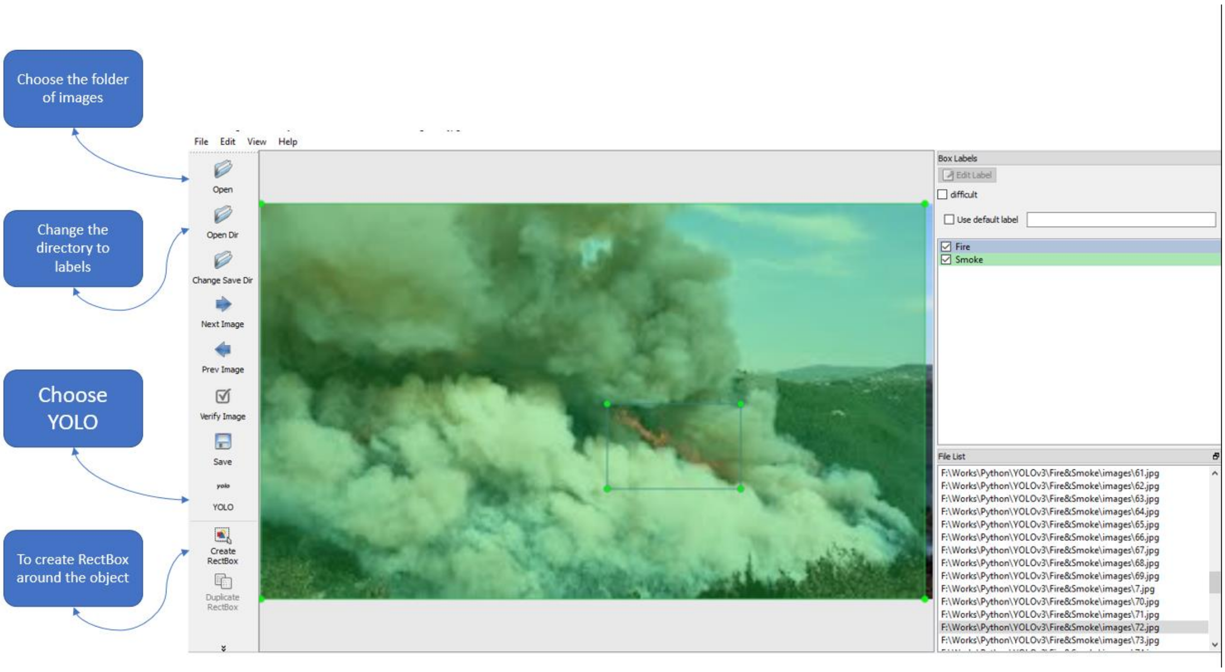


Figure 5.27. LabelImg Annotation Tool

5.11.5 Gimbal Control

This part will deal with designing an appropriate control method to drive the gimbal, holding the camera, towards the fire and smoke region. In other words, the camera mounted on the quadcopter will always concentrate on the center of the fire and smoke. The gimbal is driven by 2 servo motors; to control the yaw and the roll angles. Fortunately, a built-in feedback control system is already built inside each servo motor, thus it is enough to feed through the desired rotation angle to the servos. The control methodology is illustrated in Figure 5.28, gimbal design is shown in Figure 5.29, and the circuit diagram in Figure 5.30. In short, the central PC will detect the presence of fire in video frames received from the quadcopter and create bounding boxes around all fires and smokes in the frame. The center of each bounding box has 2D coordinates (*x or w: width, y or h: height)* in the image frame expressed in Pixels (px).

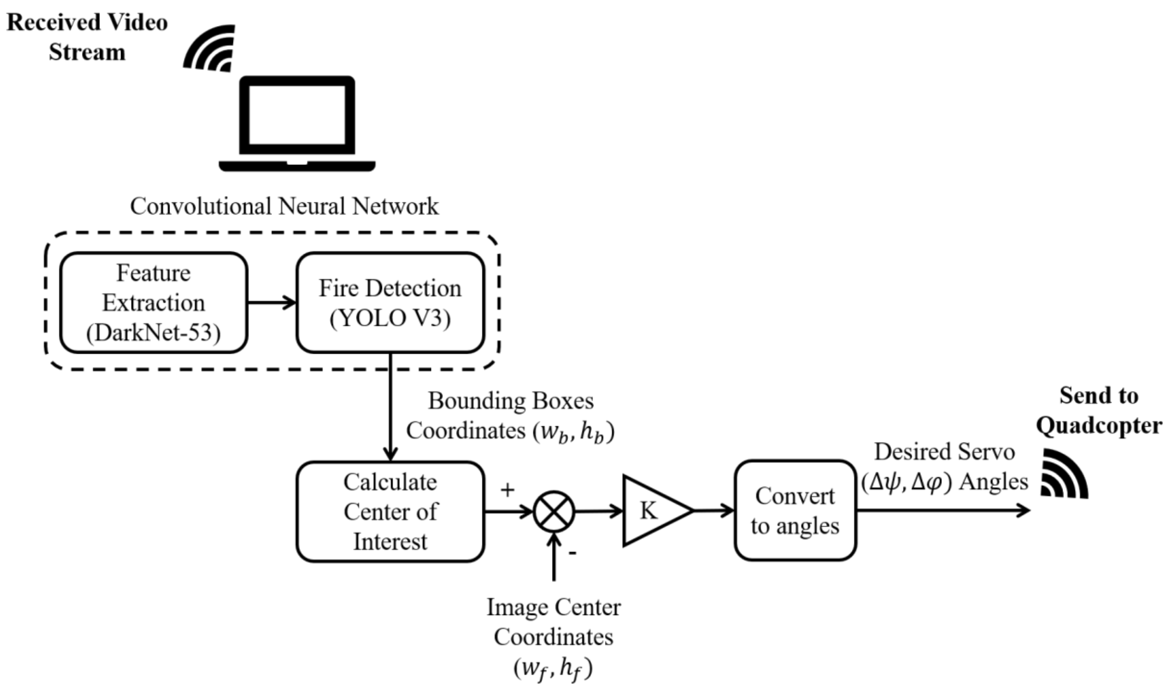


Figure 5.28. Gimbal Control Strategy

If there are multiple bounding boxes in the frame, it is required to calculate a central position between all these bounding boxes. Thus, center of interest 2D coordinates are calculated by equation (5.18):

$$w_c = \frac{1}{n}\sum_{i=1}^{n} w_i$$
$$h_c = \frac{1}{n}\sum_{i=1}^{n} h_i \qquad (5.18)$$

Where, w_i and h_i are respectively the width and height of bounding box center *i* and *n* is the number of bounding boxes in the frame. In a control system manner, and as mentioned before, it is required to keep the camera focusing on the center of interest so this variable is the setpoint of the system which will continuously be compare with the center coordinates of the frame *(w_f and h_f)*. The error signal undergoes a simple P-controller (gain *K*) to scale it and finally a mapping function is used to convert Pixels (px) into the desired servo angle deviations to be sent back to the gimbal controller (Arduino Uno) mounted on the quadcopter. Servo deviations calculation, restrained between 0^o and 180^o in yawing and 0^o and 65^o in rolling, are shown in equation (5.19).

$$\Delta\psi = map(K \times (w_c - w_f), 0, 180)$$
$$\Delta\varphi = map(K \times (h_c - h_f), 0, 65) \qquad (5.19)$$

And on the Arduino side, the required yawing and rolling is calculated through equation (5.20):

$$\psi = \psi_{current} + \Delta\psi$$
$$\varphi = \varphi_{current} + \Delta\varphi \qquad (5.20)$$

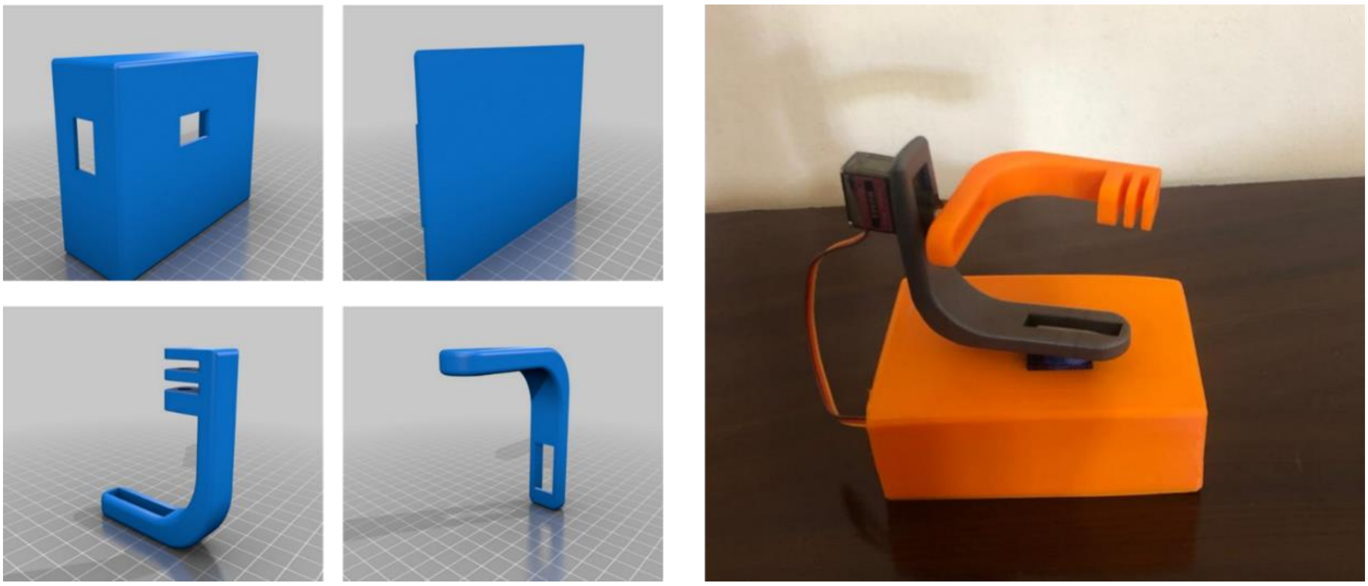


Figure 5.29. Gimbal Designed 3D and Physical Models

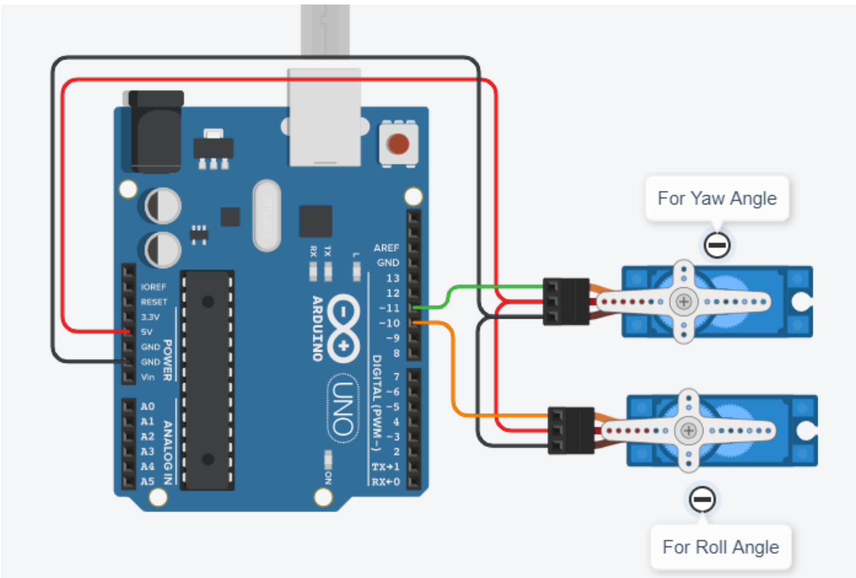


Figure 5.30. Servo Motor Control Circuit Connections

5.11.6 Conclusion

In this chapter, training a neural network model of a fire detection system via a feature extraction deep network (Darknet-53) and image recognition (YOLO V3) models was explained. Also, a P-control-driven gimbal controller was designed.

5.12 Non-Technical Aspects

5.12.1 Introduction

In this chapter, all hardware components will be presented with their cost in local/global markets in addition to the approved method of management in this work. The last part will cover the ethical, social, and environmental effect on society, as well as the sustainability will be covered.

5.12.2 Economical/Financial

The thesis is composed of a quadcopter with gimbal and a camera, as mentioned before. Table 5.14 lists the chosen items for each quadcopter and their cost. According to Local/Global markets and Table 5.15 shows the engineering staff cost.

Table 5.14. Components cost

ITEM	Price/ Piece	Quantity
Turnigy Heavy Aerial Lift frame	$40.84	1
A2212/13T motor	$8.86	4
Propeller	$2.99	2
AFRO ESC 30A	$13.50	4
Power Distribution Board	$5.91	1
MULTIWII Flight controller	$22.95	1
HRB 5000mAh 11.1v 50C	$69.99	1
Ublox Neo 6m GPS	$15.00	1
Ultrasonic Sensor HC-SR04	$0.66	1
Arduino Mega 2560	$40.30	1
Camera with Gimbal	$609.00	1
Tools	$10.00	1
Total ($)	**$910.07**	

Table 5.15. Engineering cost

Task	MM	Qualification	Salary/MM	Total Salary
Assembling quadcopter	0.25	Eng.	$1000	$250
Control system (Hardware and Software development)	0.5	Eng.	$1000	$500
Testing quadcopter	0.5	Eng.	$1000	$500
Programming (Training)	5	Eng.	$500	$2500
Testing	1	Eng.	$500	$500
Project Management	1	Eng.	$1000	$1000

Total Man Power Costs	$5000

5.12.3 Project Management

The thesis is divided into 3 main phases: Gathering Data, Simulation, and Software. Thesis started by gathering data and literature surveys. This phase started in July 27 until 16 Oct, in other words, it took 60 days. Then, the simulation phase started by modeling and designing the High/Low level control of quadcopters. The Simulation took 20 days, from Aug 24 to Sep 18. At the end, the Software phase covers programming, Training, and testing processes. The period of this phase extended from Sep 18 to Jun 2, in a total of 184 days. All phases and tasks are presented in Figure 5.31.

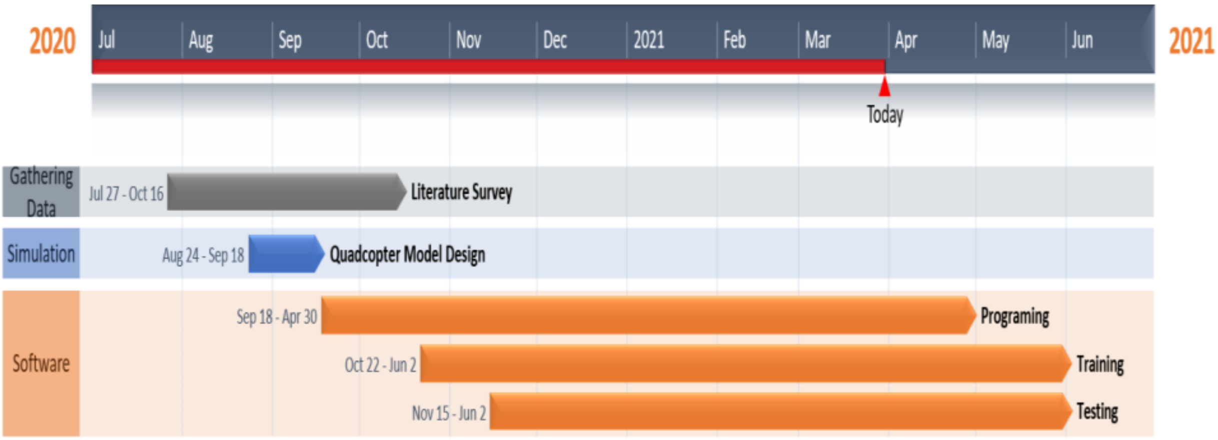


Figure 5.31. Thesis Gantt Chart

5.12.4 Ethical and Social

Ethics by definition are the moral values that guide the performance of an action or administer a person's behavior.

5.12.4.1 Quadcopter

All forms of drones face distinct ethical problems that fluctuate relying on the drone's category (Military, Civilian and commercial, and Recreational) [20]. The quadcopter in this thesis is classified as civil and commercial for fire foreign surveillance, aims at detecting fire earlier. The use of drone for this aim raises ethical problems related to the collection of citizen's data located inside or nearby the forest. This issue leads to an invasion of privacy. This adds to the noise of the operating drone problem. Also, having too many quadcopters used for surveillance purpose in a limited space might be challenging.

5.12.4.2 Camera

Taking photographs or live stream via camera mounted on a drone for a diverse purpose, may not have an ethical issue related to them. However, if a person accidentally takes a picture of another person, it can be an invasion of privacy. Hence, photography can only be evaluated based upon the action, intention, and the consequence of the actions intended by the photographer [21].

5.12.4.3 Neural Networks

Due to the hidden and complex implementation of algorithm elements or the intermediate layer of statistically trained 'neurons', there is a topic that proposes a cognitive dilemma and brings related ethical issues for deep learning (neural networks) [22]. Otherwise, computer scientists and software engineer themselves are increasingly worried about the lack of transparency of AI and deep learning [23]. This opacity of algorithmic black boxes poses a direct and long-term challenge to lawmakers and policymakers. At the end, in Thilo Hagendorf's (2020) review of twenty-two recommendations on the ethical guidelines of AI by governments and NGO's, transparency (in general, AI systems) is second only to privacy, fairness, and accountability. In other word, it is the fourth important theme of 22 themes [24].

5.12.5 Environmental and Sustainability

Quadcopter is an ecological electric machine that is often used as an alternative to polluting one. For instance, according to two studies published in 2018 [25] [26], the environmental improvement can be accomplished by using drones for delivery instead of ground transportation methods (e.g., diesel, natural gas, and even electric vehicles). Specially, when drone batteries have access to clean power source for charging. Otherwise, drones can be used in wildlife conservation. For example, it can be used for disaster relief, tracking animals, especially dangerous animals, without endangering anyone, and monitoring large area (especially agriculture area).

5.12.6 Standards

5.12.6.1 Quadcopter Standards

No drone laws have been established in accordance with Lebanese laws and regulations, but this does not mean that you can fly anywhere.in fact legislators may often refuse to use drones. To avoid problems or raise drones, it is recommended to contact the Lebanese Directorate General of Civil Aviation (DGCA). In the other hand, the U.S.'s Federal Aviation Administration's has some rules for drone usage. Table 5.16 below lists some of them:

Table 5.16. FAA's Model Aircraft Rules

Nb.	Rules
1	Do not fly from a moving vehicle, unless in a sparsely populated area
2	Avoid flying within five miles of an airport
3	Keep the drone within visual line-of-sight
4	Yield right of way to manned aircraft
5	Fly during daylight or civil twilight
6	Do not fly directly over people
7	Fly at or under 100 mph
8	Fly at or below 400 feet

Drones in particular are governed by various rules and laws around the world. According to this, quadcopters are divided into classes, categories and labels. One of the best-known regulations are the European standards 2019/947 and 2019/945 of the European Union Aviation Agency (EASA) [27] [28]. Accordingly, the quadcopter used in this work is classified as "Open-A3". In addition, several UAV standards have been developed (e.g., ISO 21384-1, ISO 21384-2, ISO 21384-3, ISO 21384-4), and others that were previously incomplete (e.g., ISO/IEC AWI 22460-2, ISO/IEC AWI 4005-1, etc.) [29].

5.12.6.2 Neural Networks Standards

In general, there are many standards that define and classify Artificial intelligence, machine learning, deep learning, and Neural network. Some of them are approved and other still under study. Table 5.17 lists some of IEEE Standards related to Neural Networks (completed and incomplete) [30].

Table 5.17. IEEE Standards Related to Neural Networks

ID	Title
IEEE 3333.1.1-2015	Standard for Operator Interfaces of Artificial Intelligence
IEEE 3333.1.2-2017	Standard for the Perceptual Quality Assessment of Three-Dimensional (3D) and Ultra-High-Definition (UHD) Contents
P3333.1.3	Standard for the Deep Learning-Based Assessment of Visual Experience Based on Human Factors
P3333.1.1	Standard for Quality of Experience (QoE) and Visual-Comfort Assessments of Three-Dimensional (3D) Contents Based on Psychophysical Studies
P2941.1	Standard for Operator Interfaces of Artificial Intelligence

5.12.7 Conclusion

As mentioned before, all components used in the thesis were presented with their cost in Local/Global markets in addition to the approved method of management in this work. Finally, the ethical, social, and environmental effects on society, and the sustainability were presented in this chapter.

5.13 Results

5.13.1 Introduction

This chapter will present the NN training and testing results. Training accuracy, training losses and training precision percentages will be plotted. In addition, the servo motor-controlled angles will be plotted vs. fires and smoke frames in a video stream.

5.13.2 Neural Networks

Training the Neural Network model to detect fires and smoke in a live video stream was done by iterating 30000 times. Figure 5.32 shows the essential training parameters. Training losses and average training losses should be as low as possible and if the average training losses becomes less than 0.0607, the training can be stopped and best results are acquired. The fire and smoke detection accuracy graphs shows that the detection accuracy after approximately 1000 iterations becomes 20% and after 30000 iterations, it settles at approximately 98% for fire and 95% for smoke. The precision percentage is the ratio of true positive images (images containing fire and smoke) to the total number of positive predictions. Intersection over Union (IoU) percentage is the accuracy of the detection algorithm given a dataset. The F1-score is the model's accuracy over a give dataset. It is a combination between the precision percentage and recall percentage of the model. And finally, the mean average precision percentage from its name, is the average of the training precision percentage.

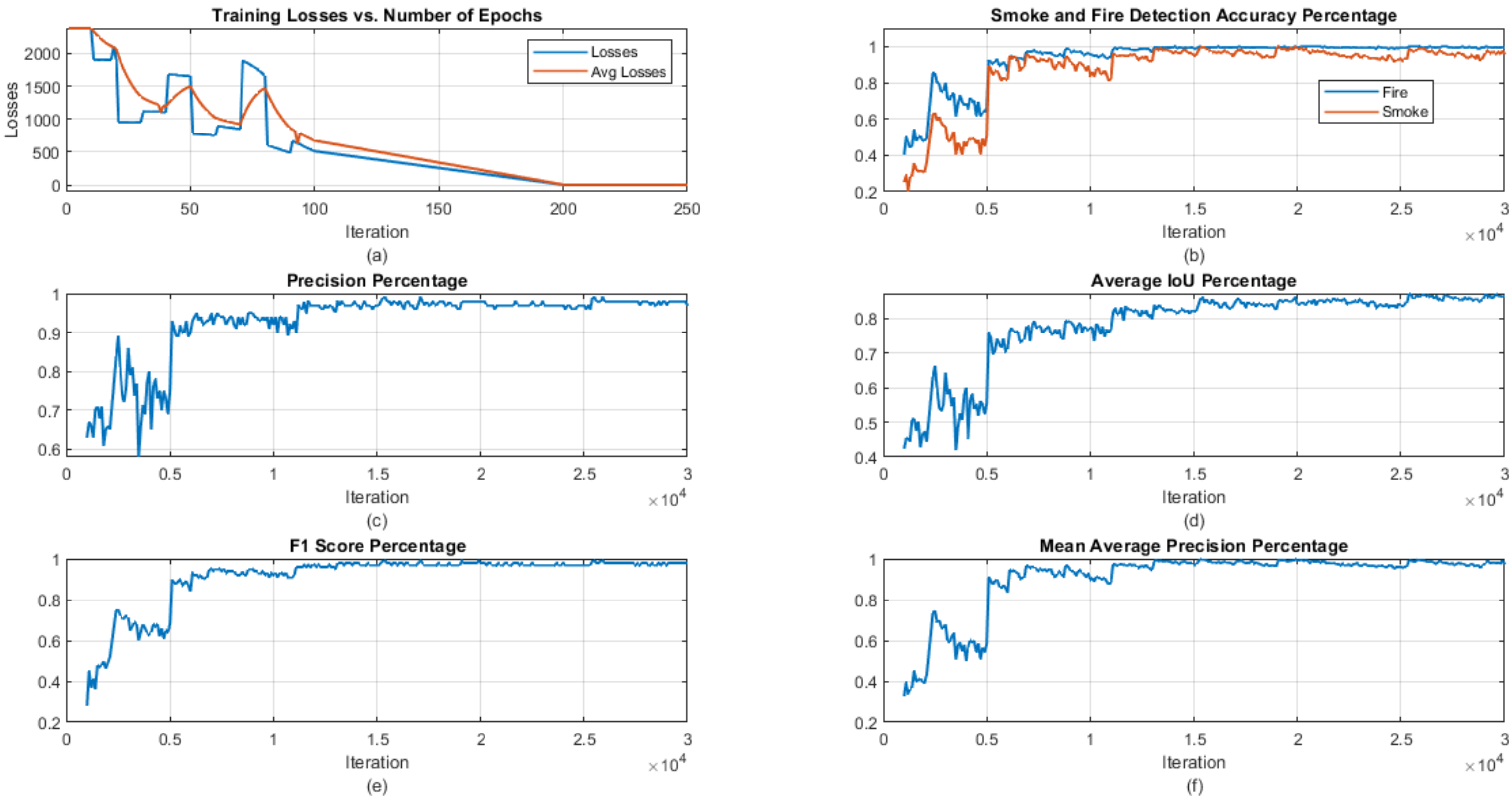


Figure 5.32. Training Losses (a), Detection Accuracy (b), Training Precision Percentage (c), Average Intersection Over Union (d), F1 Score (e), and Mean Training Precision (f)

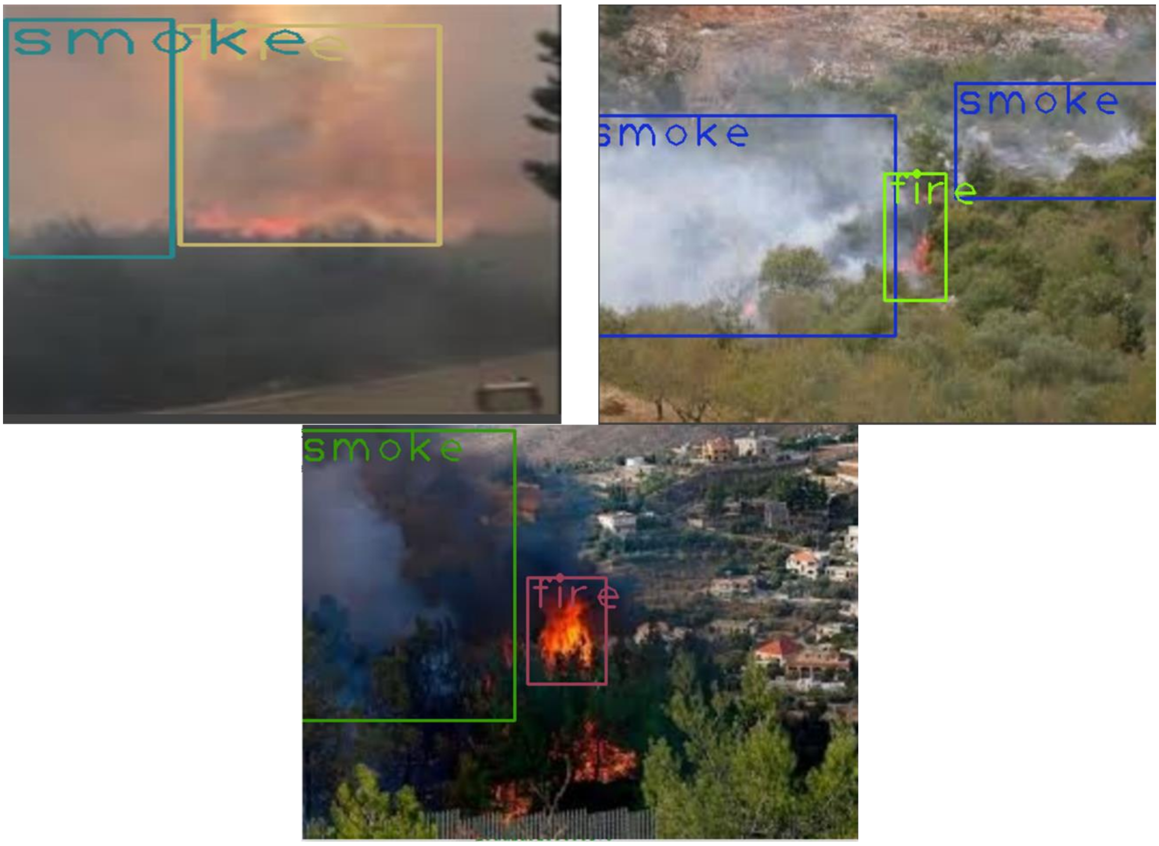


Figure 5.33. Detected Fires and Smokes in a Video Frame

5.13.3 Gimbal

The gimbal is receiving its commands from python which calculates the desired servo deviation angles based on the fires and smoke center coordinates. The gimbal's center position is at (roll = 30°, yaw = 30°). Table 5.18 below shows the desired rolling and yawing gimbal angles calculated from the central x and y coordinates and Figure 5.34 shows the gimbal desired angles received from python.

Table 5.18. Gimbal Roll and Yaw Data Associated to the frame coordinates

X	Y	Roll	Yaw
0	0	30	30
42	47	32	44
50	50	32	44
47	52	30	49
42	32	42	63
23	50	42	111
54	46	45	103
61	59	38	83
53	70	24	77
49	37	33	79
39	45	36	99
48	54	33	102
50	62	24	102
50	50	24	102

47	52	22	107
61	59	15	87
51	74	70	13
33	43	75	43
70	35	85	7
60	47	87	-11
61	50	87	-30
48	40	94	-26
38	48	95	-4
32	47	97	28
53	70	80	22
42	43	85	36
38	44	89	57
50	39	96	57
62	49	96	35
45	53	94	44

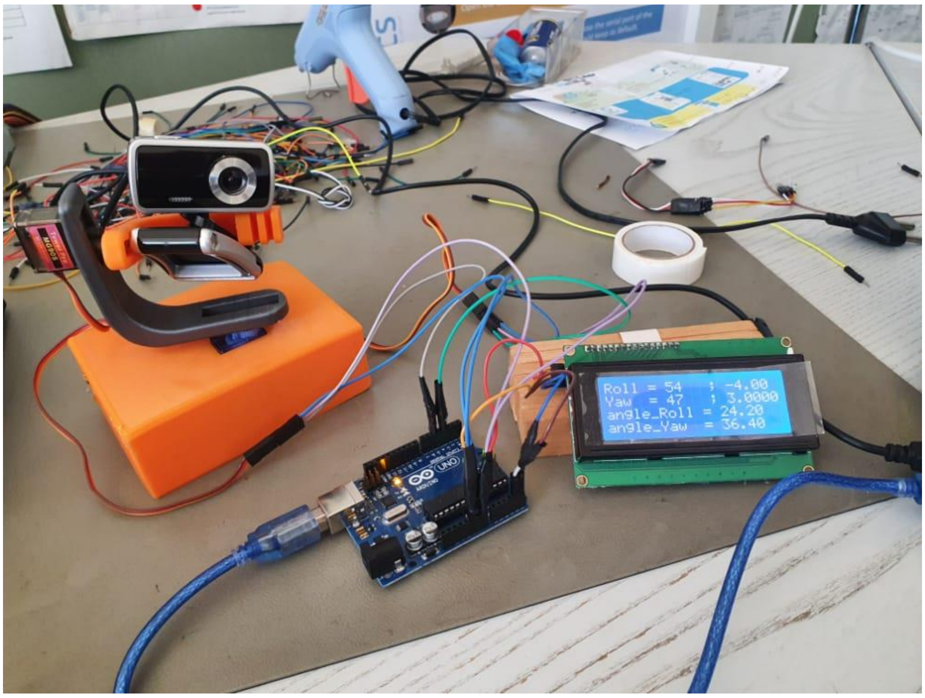


Figure 5.34. Gimbal Input Controlled from PC

5.13.4 Conclusion

This chapter presented the results of the trained neural network model and the gimbal control method.

5.14 Conclusion

5.14.1 General Conclusion

The thesis presented the implementation of a forest fire and smoke detection system based on Darknet-53 and YOLO v3 object detection networks. A camera was mounted on a quadcopter, which was not shown intact, that will transmit back a live video stream of the target forest in order for the detection system to calculate the desired gimbal deviations to maintain the video stream back at the middle of the fires and smoke. Training the neural network model was held on Google Colab's GPU while the detection was performed on a CPU-based processor. The system was able to detect fires and smoke in a video stream successfully. Through analyzing and validating the output results, it can be concluded that:

- Running the fire and smoke detection system on a GPU is 7 times faster than running it on a CPU. A significant fps drop resulted from running the system on CPU.
- Training YOLO v3 models is much flexible compared to other types of models such as RCNN, faster RCNN etc.
- Training the model on 5000 iterations gave significantly unsatisfactory detection accuracy of 55% for smoke and 65% for fires. The model needed to be trained at a minimum number of 11,000 iterations to exceed the 90% accuracy line.

5.14.2 Future Work

The system can be further developed in many ways. The propagation speed of the fires can be estimated for extra monitoring information that can help locals to better take decisions. In addition, the system can be trained to detect humans in video frames and send special kind of alert to the stakeholders.

5.15 References

[1] Gualdi S, Navarra A, "Climate scenarios in the Mediterranean region," *Forest,* vol. 2, p. 19–30, 2005.

[2] Alcamo J, Moreno JM, Nováky B, et al, "Impacts, Adaptation and Vulnerability. Contribution of Working Group II to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change," in *Climate Change* , Cambridge, Cambridge University, 2007, p. 541–580.

[3] Lydersen J.M, North M.P, Collins B.M, "Severity of an uncharacteristically large wildfire, the Rim Fire, in forestes with relatively restorted frequent rigimes," *For. Ecol. Manag.*, pp. 326-334, 2014.

[4] "European Forest Fire Information System," 28 May 2015. [Online]. Available: EFFIS: http://forest.jrc.ec.europa.eu/effis/.

[5] "Lebanon forest fires," 13 October 2019. [Online]. Available: Wikipedia: https://en.wikipedia.org/wiki/2019_Lebanon_forest_fires#cite_note-4.

[6] "information international sal.," 16 October 2020. [Online]. Available: lebeconomyfiles: https://www.lebeconomyfiles.com/48941/.

[7] "Lebanon burns and ignites the sites of communication," Al Arabiya, 15 October 2019. [Online]. Available: https://www.alarabiya.net/.

[8] "Drones in wildfire management," 3 November 2020. [Online]. Available: wikipedia: https://en.wikipedia.org/wiki/Drones_in_wildfire_management.

[9] H. Brown, "Wildfires are raging in Lebanon. Experts say they saw this coming," 17 October 2019. [Online]. Available: http://www.vox.com.

[1 Wikipedia, "2019 Lebanon forest fires," 3 June 2020. [Online]. Available: https://en.wikipedia.org.

[1 Chi Yuan, Zhixiang Liu, Youmin Zhang, "Fire Detection Using Infrared Images for UAV-based Forest Fire Surveillance," *International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 567 - 572, 2017.

[1 David W. Casbeer, Derek B. Kingston, Randal W. Beard & Timothy W. McLain, "Cooperative forest fire surveillance using a team of small unmanned air vehicles," *International Journal of Systems Science,* vol. 37, no. 6, pp. 351-360, 2006.

[1 Gouqing Zhou, Chaokui Li and Penggen Cheng, "Unmanned aerial vehicle (UAV) real-time video registration for forest fire monitoring,," *Proceedings. 2005 IEEE International Geoscience and Remote Sensing Symposium,* vol. 5, pp. 1803-1806, 2005.

[1 Mubarak A.I. Mahmoud and Honge Ren, "Forest Fire Detection Using a Rule-Based Image Processing Algorithm and Temporal Variation," *Hindawi Mathematical Problems in Engineering,* pp. 1-8, 2018.

[1 Henry Cruz ,Martina Eckert, Juan Meneses and José-Fernán Martínez, "Efficient Forest Fire Detection Index for Application in Unmanned Aerial Systems (UASs)," *Research Center on Software Technologies and Multimedia Systems for Sustainability (CITSEM),* pp. 1-16, 2016.

[1 "DJI," [Online]. Available: https://www.dji.com.

[1 E. BUBER and B. DIRI, "Performance Analysis and CPU vs GPU Comparison for Deep Learning," in *International Conference on Control Engineering & Information Technology (CEIT)*, Istanbul, 2018.

[1 Y. LeCun et al, "Handwritten digit recognition with a back-propagationnetwork," *inProc. Adv. Neural Inf. Process. Syst.*, p. 396–404, 1990.

[1 Synced, "Medium," Medium, 27 March 2018. [Online]. Available: https://medium.com/syncedreview/the-yolov3-object-detection-network-is-fast-fcceae0ab650.

[2 D. M. Marshall, Introduction to Unmanned Aircraft Systems, Second ed., CRC Press, Taylor and Francis Group, 2012, pp. 29-49.

[2 "Photography ethicscenter," 25 April 2018. [Online]. Available: https://www.photoethics.org/content/2018/5/31/photography-ethics-and-why-they-matter.

[2 B. Long, "Blog of the APA," 13 August 2020. [Online]. Available: https://blog.apaonline.org/2020/08/13/the-ethics-of-deep-learning-ai-and-the-epistemic-opacity-dilemma/.

[2 P. Voosen, "ScienceMag," 6 Jul 2017. [Online]. Available: https://www.sciencemag.org/news/2017/07/how-ai-detectives-are-cracking-open-black-box-deep-learning.

[2 T. Hagendorff, "The Ethics of AI Ethics: An Evaluation of Guidelines," pp. 99-120, 28 july 2020.

[2 Joshuah K.Stolaroff, Constantine Samaras, Emma R. O'Neill, Alia Lubers, Alexandra S. Mitchell, and Daniel Ceperley, "Energy use and life cycle greenhouse gas emissions of drones for commercial package delivery," 13 Februrary 2018.

[2 Jiyoon Park, Solhee Kim, Kyo Suh, "A Comparative Analysis of the Environmental Benefits of Drone-Based Delivery Services in Urban and Rural Areas," p. 7, 20 March 2018.

[2 E. U. A. Agency, "EASA," 13 jan 2021. [Online]. Available: https://www.easa.europa.eu/document-library/easy-access-rules/easy-access-rules-unmanned-aircraft-systems-regulation-eu.

[2 "DRONERULES," july 2020. [Online]. Available: https://dronerules.eu/en/recreational/eu_regulations_updates.

[2 "ISO," [Online]. Available: https://www.iso.org/search.html?q=drone&hPP=10&idx=all_en&p=0&hFR%5Bcategory%5D%5B0%5D=standard.

[3 I. SA, "IEEE STANDARDS ASSOCIATION," [Online]. Available: https://standards.ieee.org/search-results.html?q=neural+network&facetValue=4294967245.

6 Autonomous Quadcopter Swarm for fire regions surveillance (Master Thesis, June 2021)

LEBANESE INTERNATIONAL UNIVERSITY

الجامعة اللبنانية الدولية

AUTONOMOUS QUADCOPTER SWARM FOR FIRE REGIONS SURVEILLANCE

Master Thesis

by

Raja M. Murad, 51430254

Nour O. Karim, 51630080

Submitted to the School of Engineering of the

Lebanese International University

Tripoli, Lebanon

in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRONICS ENGINEERING

Fall 2020-2021

Approved By

Dr. Abdelrazzak Merheb		
Supervisor	**Date**	**Signature**
Dr. Shadi Abdallah		
Committee Member	**Date**	**Signature**
Dr. Hussein Kassem		
Committee Member	**Date**	**Signature**

6.1 Presentation of Master Thesis

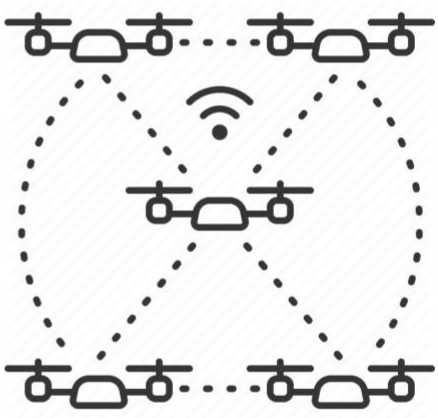
Quadcopter Swarm for Forest Fires Monitoring

MASTER THESIS
ON MONDAY FEB 8, 2021

PRESENTED BY:
Raja Murad – 51430254
Nour Karim – 51630080

SUPERVISED BY:
Dr. Abdulrazzak Merheb

COMMITTEE MEMBERS:
Dr. Hussein Kassem
Dr. Chady Abdallah



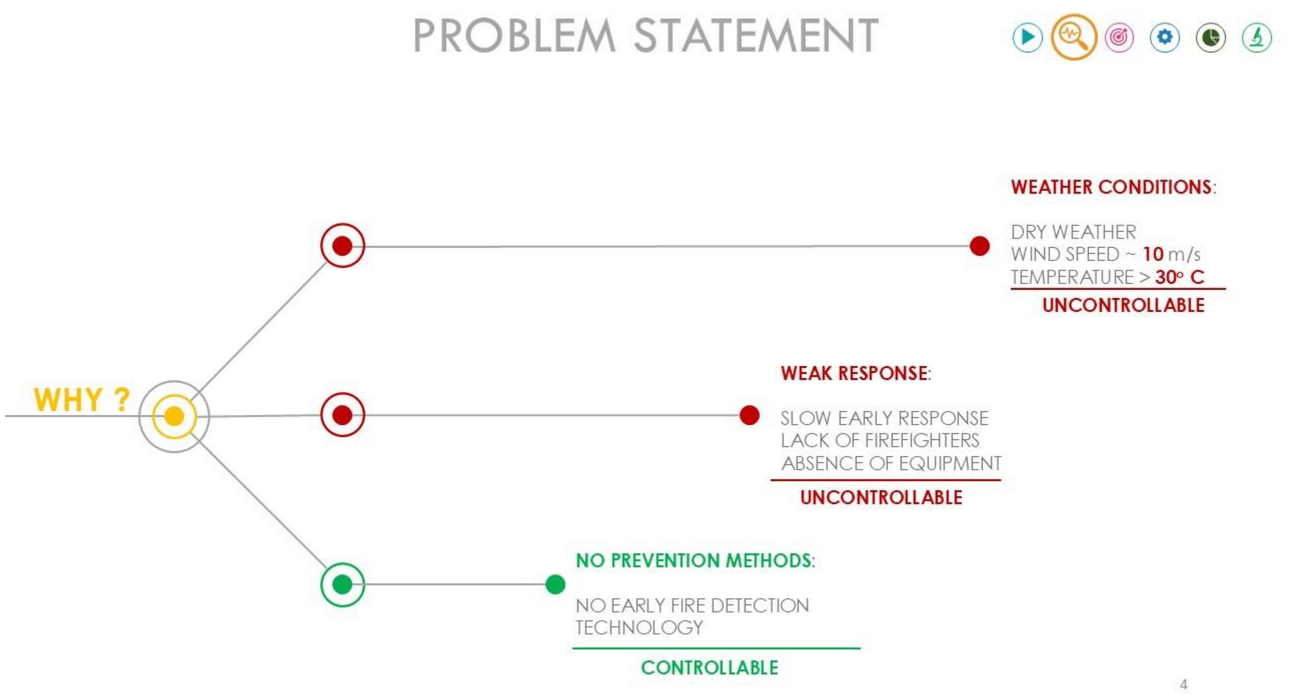
1

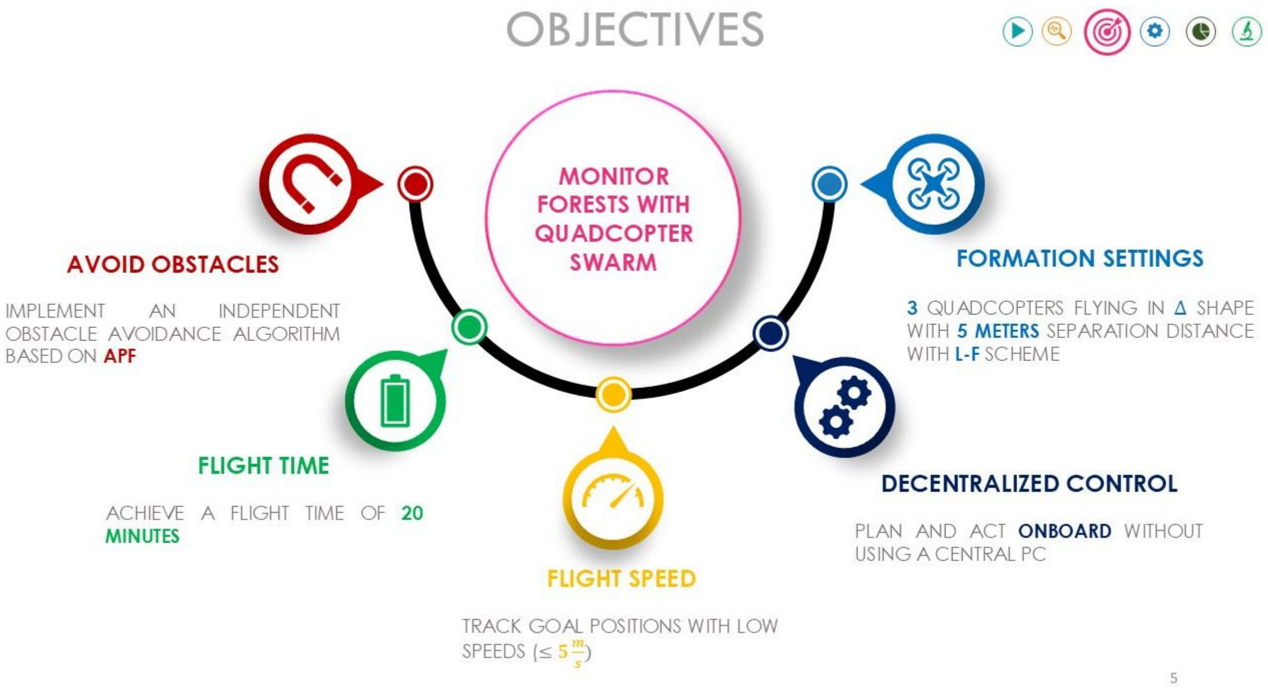
OUTLINE

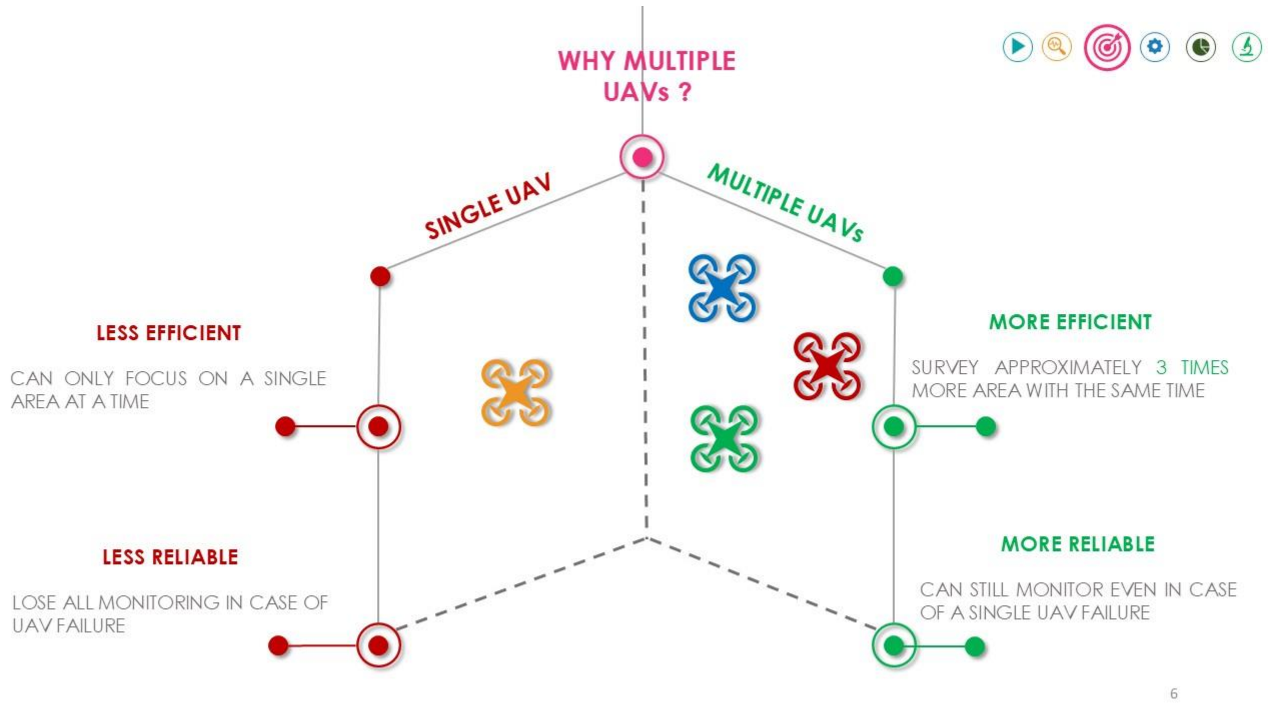
- Background
- Problem Statement
- Objectives
- Design Procedure
- Results
- Conclusion & Future Work

2


BACKGROUND
LEBANON FIRES IN THE LAST DECADE
2010
320 FIRES IGNITED
4661 HECTARES BURNED
2014
156 FIRES IGNITED
1852 HECTARES BURNED
2016
260 FIRES IGNITED
1871 HECTARES BURNED
2019
120 FIRES IGNITED
200 HECTARES BURNED
2020
--- FIRES IGNITED
--- HECTARES BURNED
3

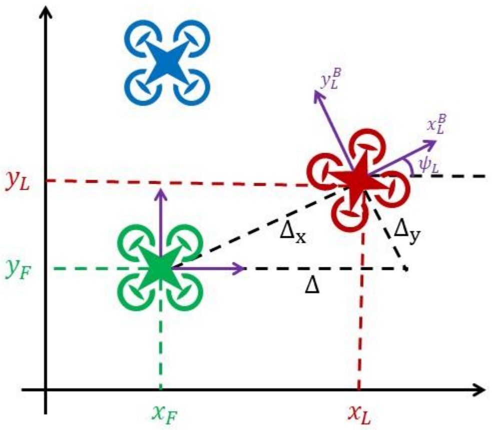

PROBLEM STATEMENT
WHY ?
WEATHER CONDITIONS:
DRY WEATHER
WIND SPEED ~ 10 m/s
TEMPERATURE > 30° C
UNCONTROLLABLE
WEAK RESPONSE:
SLOW EARLY RESPONSE
LACK OF FIREFIGHTERS
ABSENCE OF EQUIPMENT
UNCONTROLLABLE
NO PREVENTION METHODS:
NO EARLY FIRE DETECTION
TECHNOLOGY
CONTROLLABLE
4


OBJECTIVES
MONITOR FORESTS WITH QUADCOPTER SWARM
AVOID OBSTACLES
IMPLEMENT AN INDEPENDENT OBSTACLE AVOIDANCE ALGORITHM BASED ON APF
FORMATION SETTINGS
3 QUADCOPTERS FLYING IN Δ SHAPE WITH 5 METERS SEPARATION DISTANCE WITH L-F SCHEME
FLIGHT TIME
ACHIEVE A FLIGHT TIME OF 20 MINUTES
DECENTRALIZED CONTROL
PLAN AND ACT ONBOARD WITHOUT USING A CENTRAL PC
FLIGHT SPEED
TRACK GOAL POSITIONS WITH LOW SPEEDS (≤ 5 m/s)
5


WHY MULTIPLE UAVs ?
SINGLE UAV
MULTIPLE UAVs
LESS EFFICIENT
CAN ONLY FOCUS ON A SINGLE AREA AT A TIME
LESS RELIABLE
LOSE ALL MONITORING IN CASE OF UAV FAILURE
MORE EFFICIENT
SURVEY APPROXIMATELY 3 TIMES MORE AREA WITH THE SAME TIME
MORE RELIABLE
CAN STILL MONITOR EVEN IN CASE OF A SINGLE UAV FAILURE
6

DESIGN

L-F CONCEPT



DISTANCE IS MEASURED RELATIVE TO THE LEADER'S FRAME

Δ_x AND Δ_y ARE THE DESIRED SEPERATION DISTANCE

$$\begin{bmatrix}\Delta_x\\ \Delta_y\end{bmatrix} = \begin{bmatrix}-\cos(\psi_L) & -\sin(\psi_L)\\ \sin(\psi_L) & -\cos(\psi_L)\end{bmatrix}\begin{bmatrix}x_L - x_F\\ y_L - y_F\end{bmatrix}$$

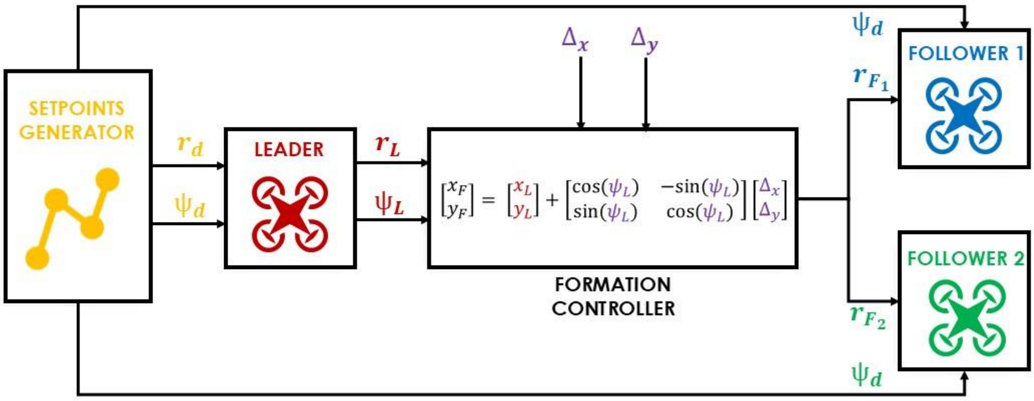
INVERSE

$$\begin{bmatrix}x_F\\ y_F\end{bmatrix} = \begin{bmatrix}x_L\\ y_L\end{bmatrix} + \begin{bmatrix}\cos(\psi_L) & -\sin(\psi_L)\\ \sin(\psi_L) & \cos(\psi_L)\end{bmatrix}\begin{bmatrix}\Delta_x\\ \Delta_y\end{bmatrix}$$

7

DESIGN

FORMATION METHODS



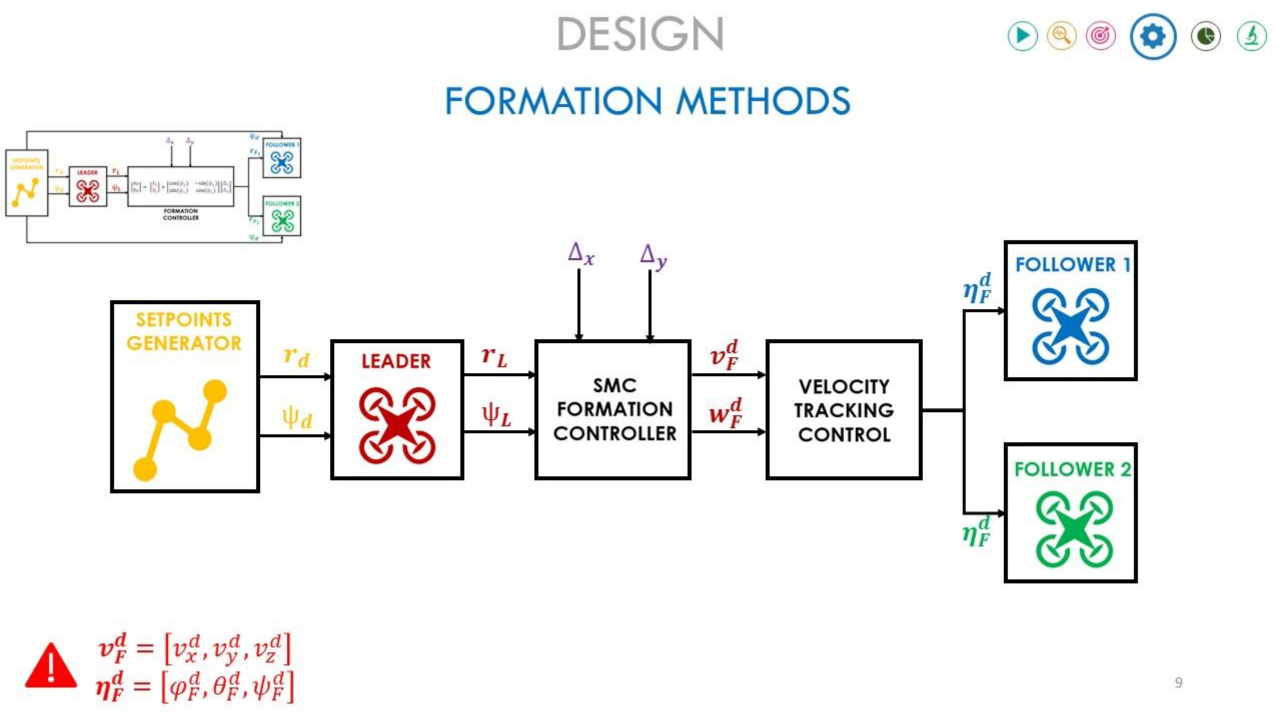
EACH QUADCOPTER WILL HAVE ITS OWN LOCAL PLANNER

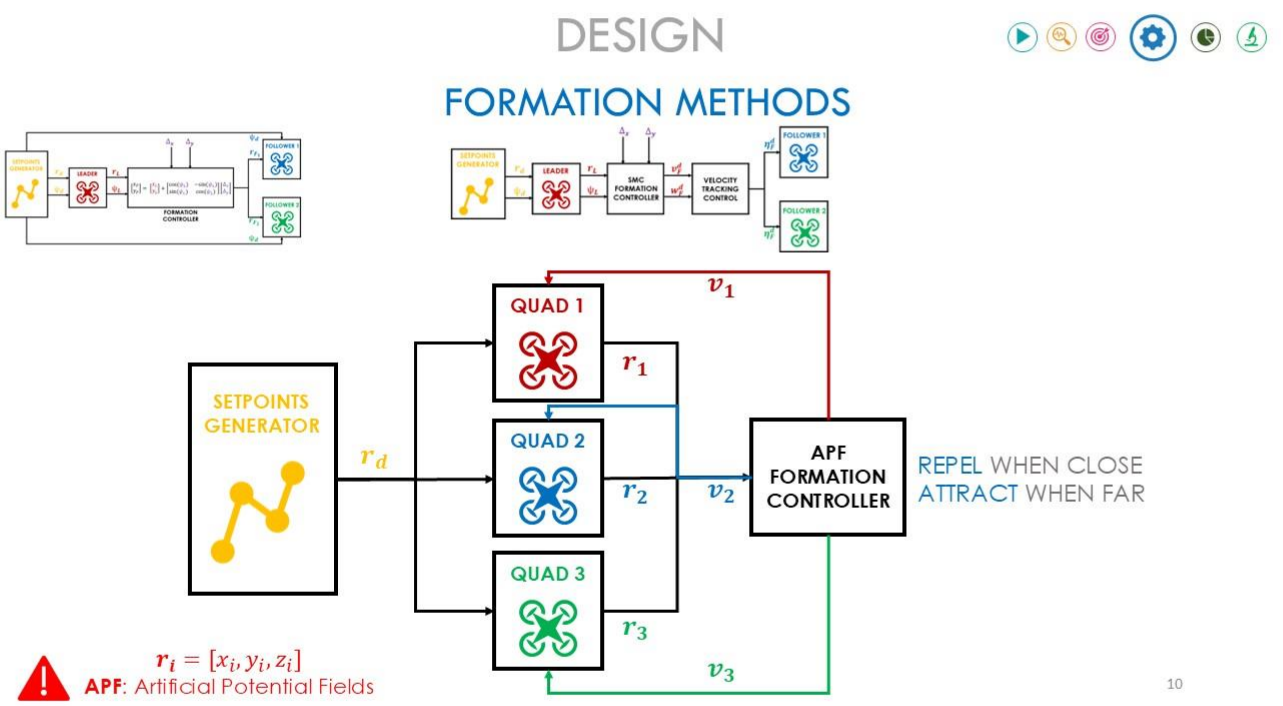
DESIRED POSITION WILL BE FED FOR EACH FOLLOWER!

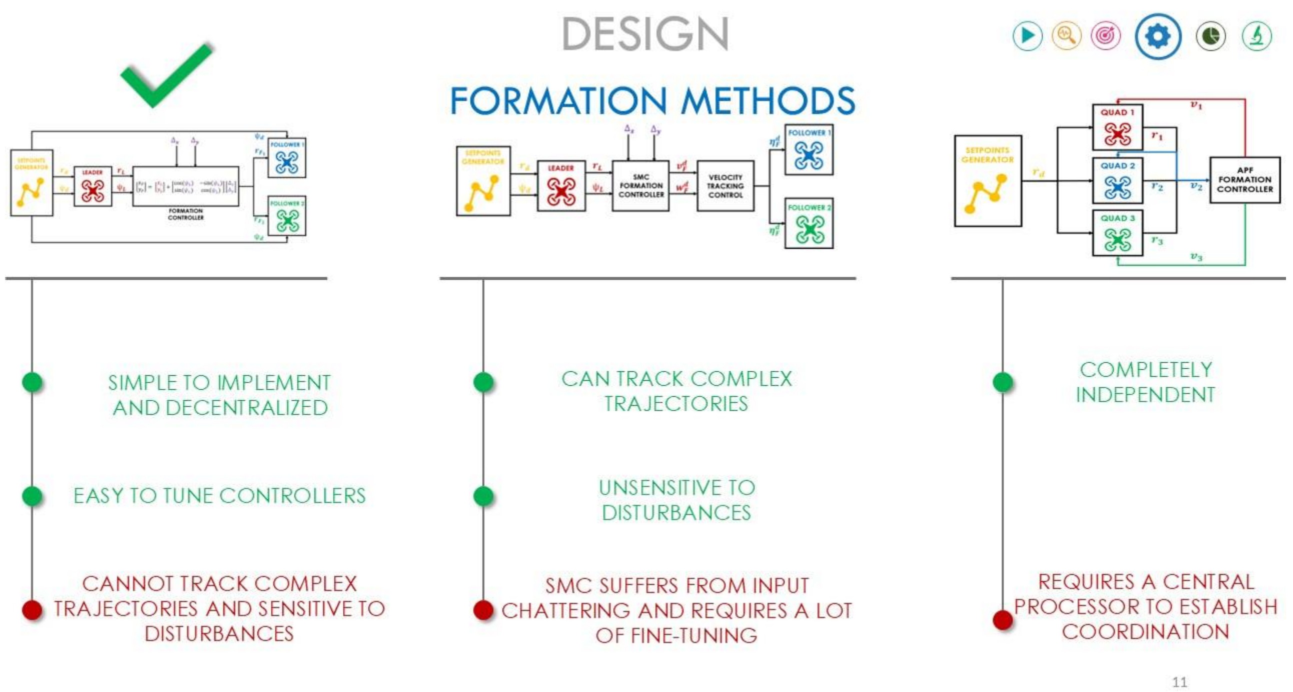
$r_i = [x_i, y_i, z_i]$

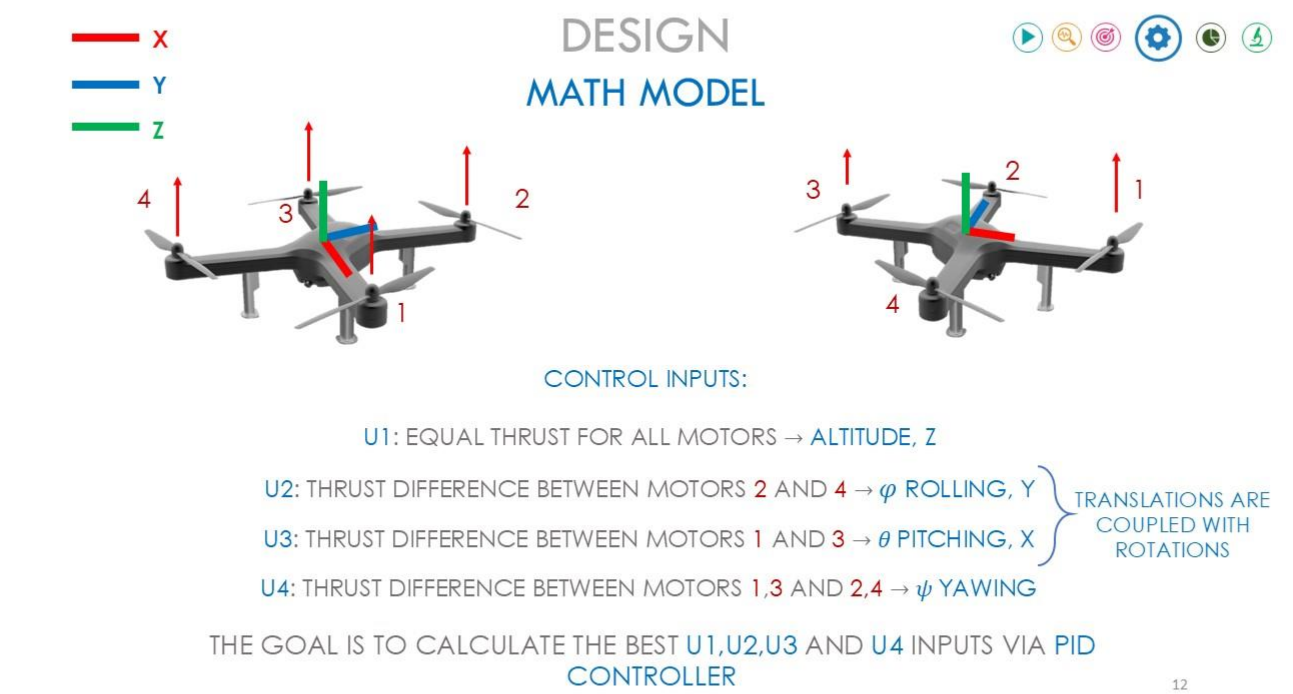
$z_L = z_F$

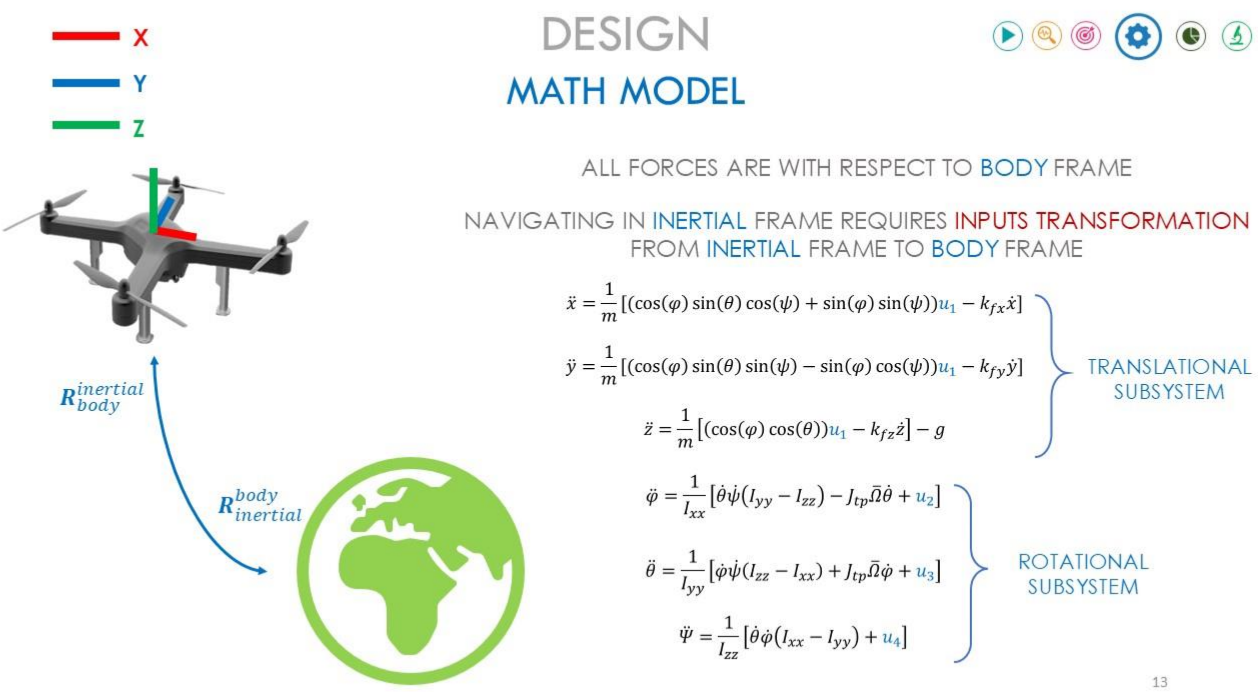
8

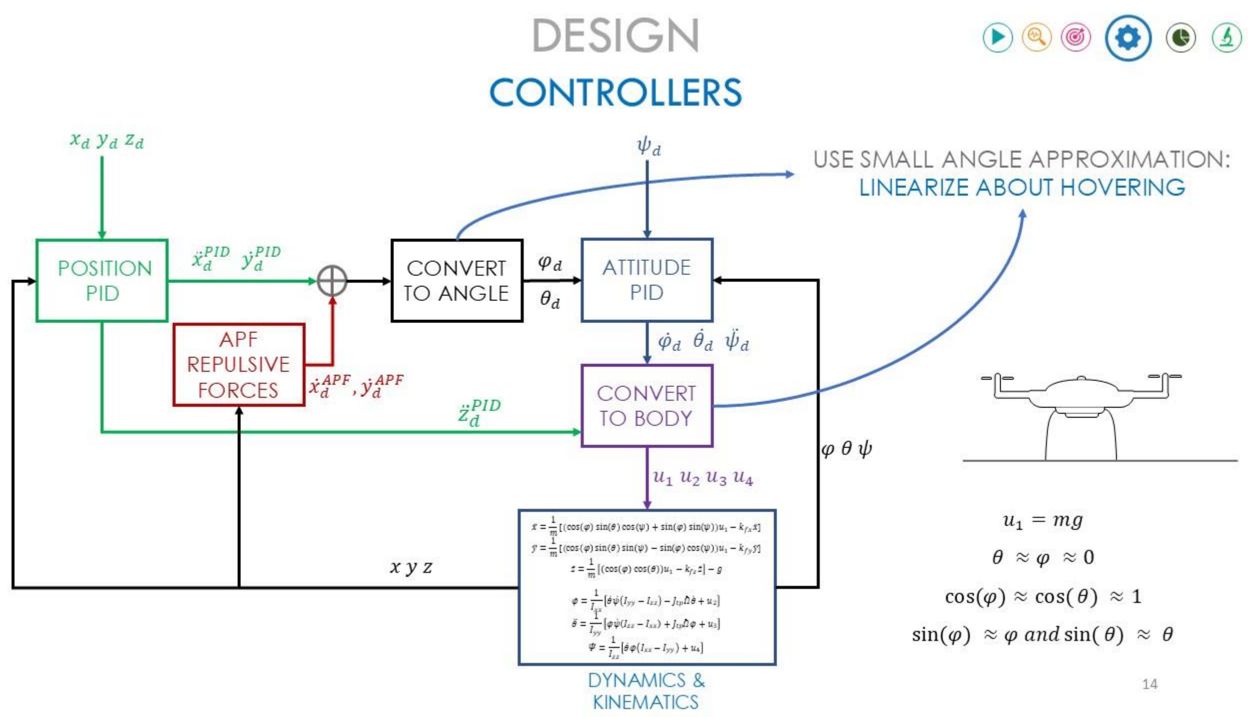

DESIGN
FORMATION METHODS
SETPOINTS GENERATOR
r_d
ψ_d
LEADER
r_L
ψ_L
Δ_x
Δ_y
SMC FORMATION CONTROLLER
v_F^d
w_F^d
VELOCITY TRACKING CONTROL
η_F^d
FOLLOWER 1
FOLLOWER 2
$v_F^d = [v_x^d, v_y^d, v_z^d]$
$\eta_F^d = [\varphi_F^d, \theta_F^d, \psi_F^d]$
9


DESIGN
FORMATION METHODS
SETPOINTS GENERATOR
r_d
QUAD 1
QUAD 2
QUAD 3
r_1
r_2
r_3
v_1
v_2
v_3
APF FORMATION CONTROLLER
REPEL WHEN CLOSE
ATTRACT WHEN FAR
$r_i = [x_i, y_i, z_i]$
APF: Artificial Potential Fields
10


DESIGN
FORMATION METHODS
SETPOINTS GENERATOR
LEADER
FORMATION CONTROLLER
FOLLOWER 1
FOLLOWER 2
SMC FORMATION CONTROLLER
VELOCITY TRACKING CONTROL
QUAD 1
QUAD 2
QUAD 3
APF FORMATION CONTROLLER
SIMPLE TO IMPLEMENT AND DECENTRALIZED
EASY TO TUNE CONTROLLERS
CANNOT TRACK COMPLEX TRAJECTORIES AND SENSITIVE TO DISTURBANCES
CAN TRACK COMPLEX TRAJECTORIES
UNSENSITIVE TO DISTURBANCES
SMC SUFFERS FROM INPUT CHATTERING AND REQUIRES A LOT OF FINE-TUNING
COMPLETELY INDEPENDENT
REQUIRES A CENTRAL PROCESSOR TO ESTABLISH COORDINATION
11


DESIGN
MATH MODEL
X
Y
Z
4
3
2
1
3
2
1
4
CONTROL INPUTS:
U1: EQUAL THRUST FOR ALL MOTORS → ALTITUDE, Z
U2: THRUST DIFFERENCE BETWEEN MOTORS 2 AND 4 → φ ROLLING, Y
U3: THRUST DIFFERENCE BETWEEN MOTORS 1 AND 3 → θ PITCHING, X
TRANSLATIONS ARE COUPLED WITH ROTATIONS
U4: THRUST DIFFERENCE BETWEEN MOTORS 1,3 AND 2,4 → ψ YAWING
THE GOAL IS TO CALCULATE THE BEST U1,U2,U3 AND U4 INPUTS VIA PID CONTROLLER
12


DESIGN
MATH MODEL
X
Y
Z
$R_{body}^{inertial}$
$R_{inertial}^{body}$
ALL FORCES ARE WITH RESPECT TO BODY FRAME
NAVIGATING IN INERTIAL FRAME REQUIRES INPUTS TRANSFORMATION FROM INERTIAL FRAME TO BODY FRAME
$\ddot{x} = \frac{1}{m}[(\cos(\varphi)\sin(\theta)\cos(\psi) + \sin(\varphi)\sin(\psi))u_1 - k_{fx}\dot{x}]$
$\ddot{y} = \frac{1}{m}[(\cos(\varphi)\sin(\theta)\sin(\psi) - \sin(\varphi)\cos(\psi))u_1 - k_{fy}\dot{y}]$
$\ddot{z} = \frac{1}{m}[(\cos(\varphi)\cos(\theta))u_1 - k_{fz}\dot{z}] - g$
TRANSLATIONAL SUBSYSTEM
$\ddot{\varphi} = \frac{1}{I_{xx}}[\dot{\theta}\dot{\psi}(I_{yy} - I_{zz}) - J_{tp}\bar{\Omega}\dot{\theta} + u_2]$
$\ddot{\theta} = \frac{1}{I_{yy}}[\dot{\varphi}\dot{\psi}(I_{zz} - I_{xx}) + J_{tp}\bar{\Omega}\dot{\varphi} + u_3]$
$\ddot{\Psi} = \frac{1}{I_{zz}}[\dot{\theta}\dot{\varphi}(I_{xx} - I_{yy}) + u_4]$
ROTATIONAL SUBSYSTEM
13


DESIGN
CONTROLLERS
$x_d\ y_d\ z_d$
ψ_d
USE SMALL ANGLE APPROXIMATION:
LINEARIZE ABOUT HOVERING
POSITION PID
$\ddot{x}_d^{PID}\ \ddot{y}_d^{PID}$
APF REPULSIVE FORCES
$\ddot{x}_d^{APF}, \ddot{y}_d^{APF}$
CONVERT TO ANGLE
φ_d
θ_d
ATTITUDE PID
$\dot{\varphi}_d\ \dot{\theta}_d\ \ddot{\psi}_d$
$\ddot{z}_d^{PID}$
CONVERT TO BODY
$u_1\ u_2\ u_3\ u_4$
$\varphi\ \theta\ \psi$
$x\ y\ z$
DYNAMICS & KINEMATICS
$u_1 = mg$
$\theta \approx \varphi \approx 0$
$\cos(\varphi) \approx \cos(\theta) \approx 1$
$\sin(\varphi) \approx \varphi$ and $\sin(\theta) \approx \theta$
14

ATTRACTIVE ACCELERATION
REPULSIVE ACCELERATION
RESULTANT ACCELERATION

DESIGN
CONTROLLERS

APF REPULSIVE FORCES

REPEL THE QUADCOPTER FROM OBSTACLES

CHARACTERIZED BY INTENSE STRENGTH NEAR OBSTACLES

HAVE NO EFFECT WHEN FAR FROM OBSTACLES

$$F_{rep_i}(q_0) = \begin{cases} k_{rep}\left(\frac{1}{d_{obst_i}(q_0)} - \frac{1}{d_0}\right)\frac{1}{d^2_{obst_i}(q_0)}\hat{e}_i \text{ , if } d_{obst_i}(q_0) \le d_0 \\ 0 \text{ , if } d_{obst_i}(q_0) > d_0 \end{cases}$$

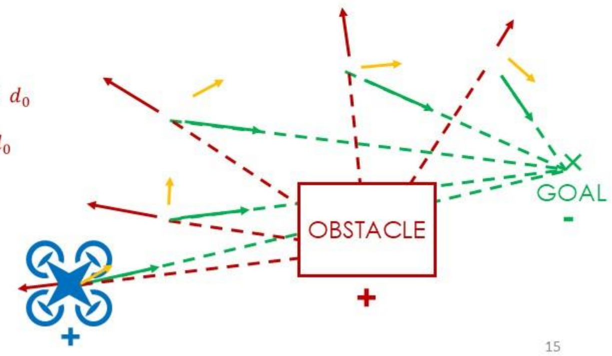
$$\ddot{x}_d = \ddot{x}_d^{PID} + \ddot{x}_d^{Rep}$$

$$\ddot{y}_d = \ddot{y}_d^{PID} + \ddot{y}_d^{Rep}$$

POSITION PID

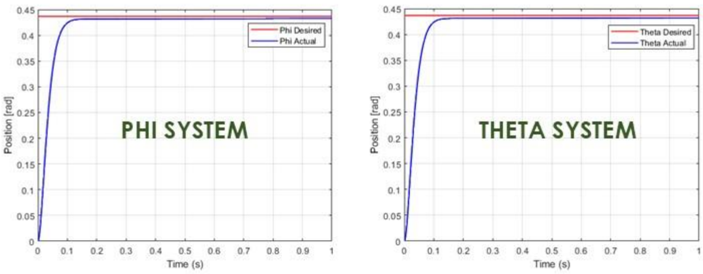
ATTRACT THE QUADCOPTER TO GOAL POSITION

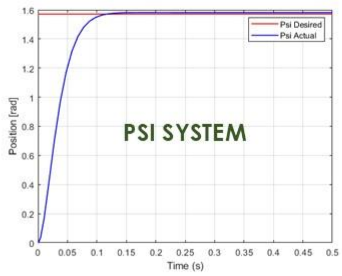
$$u(t) = K_{p_x}e(t) + K_{i_x}\int e(t)dt + K_{d_x}\frac{de(t)}{dt}$$



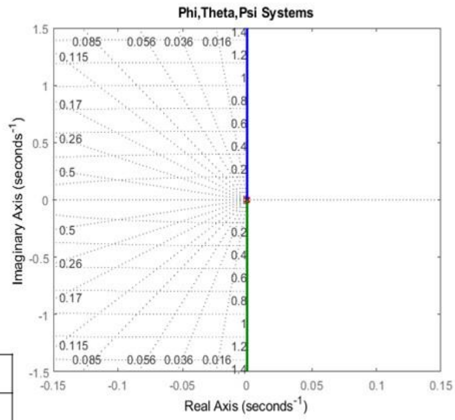
15

RESULTS
ATTITUDE CONTROLLER





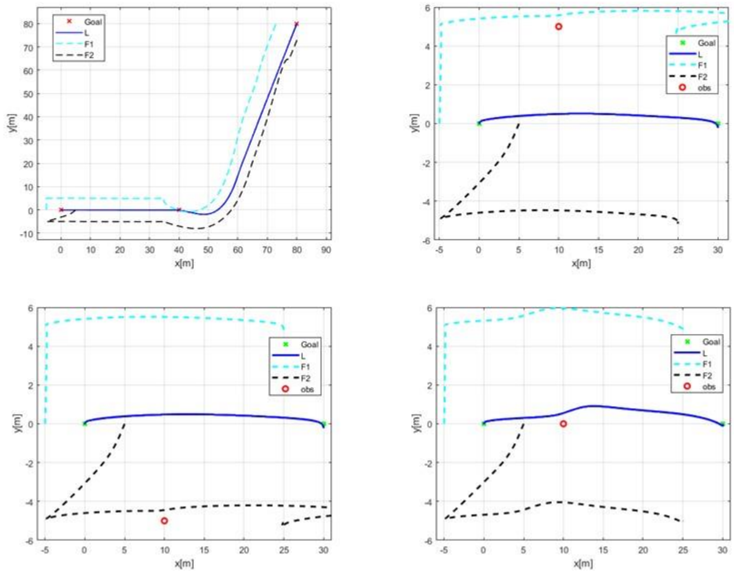
	Attitude		
	k_p	k_i	k_d
	5.3	0.8	28
%overshoot	0%		
Rise Time	0.05 *seconds*		



16

RESULTS

POSITION CONTROLLER



NECESSITY OF USING A COMPLEMENTARY FILTER

$$\ddot{x}_d = c\,\ddot{x}_d^{\,PID} + (1-c)\,\ddot{x}_d^{\,Rep}$$
$$\ddot{y}_d = c\,\ddot{y}_d^{\,PID} + (1-c)\,\ddot{y}_d^{\,Rep}$$

POSITION PID TUNED GAINS

	Leader			Followers		
	k_p	k_i	k_d	k_p	k_i	k_d
	5	0	20	15	0	20
k_{rep}	REPULSIVE FORCES SCALING FACTOR: 20					
d_0	REPULSIVE FORCES INFLUENCE DISTANCE: $2\ m$					

17

CONCLUSION & FUTURE WORK

LEADER-FOLLOWER SCHEME GAVE SATISFACTORY RESULTS BUT IT LACKS INDEPENDENCE

FOR PART B:

WILL TEST THIS FORMATION CONTROL METHOD ON REAL HARDWARE

WILL USE KALMAN FILTER TO BETTER ESTIMATE POSITION WITH THE PRECENSE OF SENSOR NOISES

18

7 Emergency Communication System (ECS) Development, V1.3 (7.2.23)

بسم الله الرحمن الرحيم

Emergency Communication System Development

LAST UPDATED: Feb 3, 2023

Written By:

Raja Mourad

Mohammad Najjarine

Karim Meemary

7.1 Document Versioning

Document Version	**Description**	**Edited By**	**Release Date**
V1.0	Initial Release	Raja Mourad	3/1/2023
V1.1	Major changes: • Added the IU transceiver development. • Updated the "How to Use" section.	Raja Mourad	13/1/2023
V1.2	Major changes: • Added a channel coding section. • Added the SU development section.	Raja Mourad	21/1/2023
V1.3	Major changes: • Added test cases section. • Added test cases results section.	Mohamad Najjarine	2/2/2023

TABLE OF CONTENTS

List of Figures

List of Tables

7.2 Source Code Version Control

IU Version	Features
V1.0	Initial release: Gnuradio script (Base Station) to receive packets from NRF24 with 2 Mbps data air rate and configurable communication channel.
V1.01	Text messages received from NRF24 through gnuradio are now decoded and the payload is extracted.
V1.02	Base station can send text data back to NRF24 modules over a configurable address and communication channel.

SU Version	Features
V1.0	Initial release: Gnuradio flowgraph to send and receive packets using GFSK modulation and channel coding.

7.3 Introduction

This article demonstrates the development process of an emergency communication system (EmerComm). This system is intended to be used where secured and independent communication channels are requested. It communicates via a Control Station (CS) with multiple Intervention Units (IU) and multiple Scouting Units (SU).

The Control Station (CS) is responsible for:

Sending position and altitude commands to scouting units.
Sending text and voice messages to intervention units.
Receiving position, images, and video livestream from SUs.
Receiving position, text, and voice information from IUs.

7.4 Requirements

The system software requirements are listed in the requirements tracking sheet. In short, the following requirements shall be met.

Table 2-1 EmerComm Software Requirements

Req. ID	Description	Field	Status
Req_001	The EmerComm system consists of 3 parts: Control station "CS", Scouting units "SU", and intervention units "IU".	General	

Req_002	SU shall send its location (Lat, Long, Alt) periodically to the CS.	SU	
Req_003	SU shall be able to send images or video to the CS on request.	SU	
Req_004	SU shall change its location by a CS command.	SU	
Req_005	SU shall use an SDR for its communication with the CS.	SU	
Req_006	A voice communication channel shall be established to make a streaming voice communication channel between the IUs and IU-CS (broadcasting).	General	
Req_007	IU shall be able to send/receive text message to/from the CS.	IU	
Req_008	IU shall send its location (Lat, Long, Alt) periodically to the CS.	IU	
Req_009	IU may use any Mid-Range communication module to make the communication with the CS.	IU	
Req_010	All communication packages shall be encrypted.	General	
Req_011	The AES standard shall be used for the encryption.	General	
Req_012	IU may have its own interaction hardware "Input/Output" or it may connect to a mobile phone via Bluetooth to do it.	CS	
Req_013	The CS application shall be developed to be run on windows computer, with an SDR unit connected for the communication.	CS	
Req_014	The CS application main dashboard consists of: radio module, units Listing module, mapping module, chatting module.	CS	
Req_015	The radio module used to listen and calling the IUs.	CS	
Req_016	The units Listing module used to show the on-range units with their info (Name, details, status).	CS	
Req_017	The map module shall be used to locate units (SU, IU) on map using pins.	CS	
Req_018	Distinct pins shall be used for SU and IU.	CS	
Req_019	SU pin shall be movable to send the unit a command to change its location.	CS	

Req_020	The map module may also use the images provided by the SUs instead of the map to locate the units.	CS	
Req_021	The chat module shall be used to send/receive text message between the CS and IU.	CS	
Req_022	The CS shall have the ability to send text message to a specific IU or to all (broadcasting).	CS	

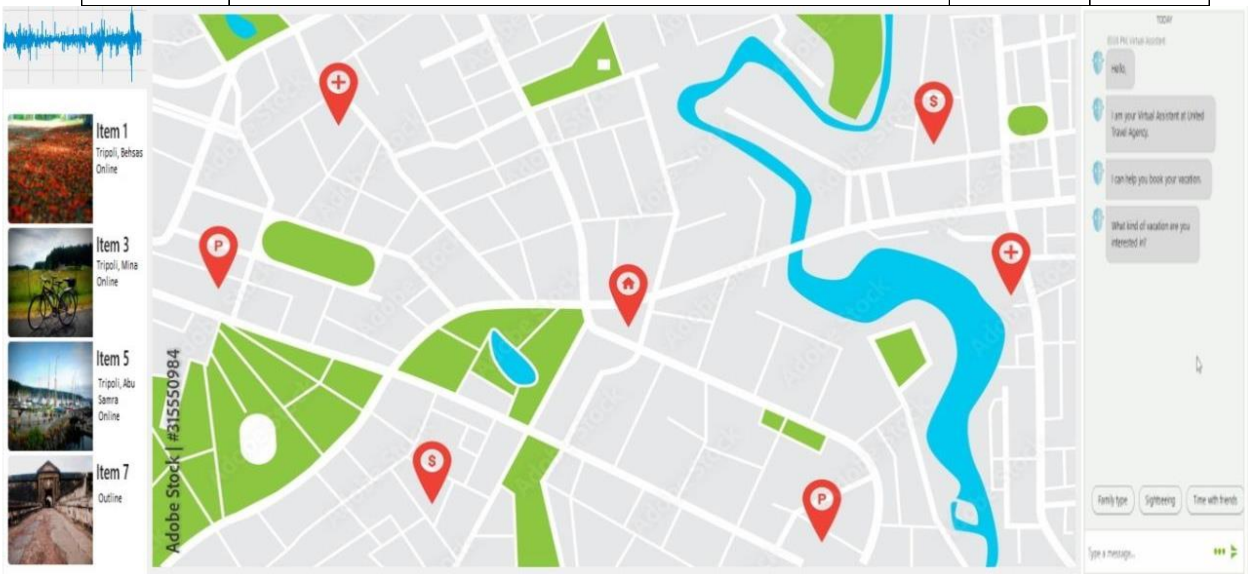


Figure 2-1 System Planned GUI

The planned Graphical User Interface shall consist of a text interface between the intervention units and the base station. In addition, it should view, in real time, the current position of each IU and each SU. Numbers of connected devices should clearly be displayed with the name of each device.

7.5 System Overview

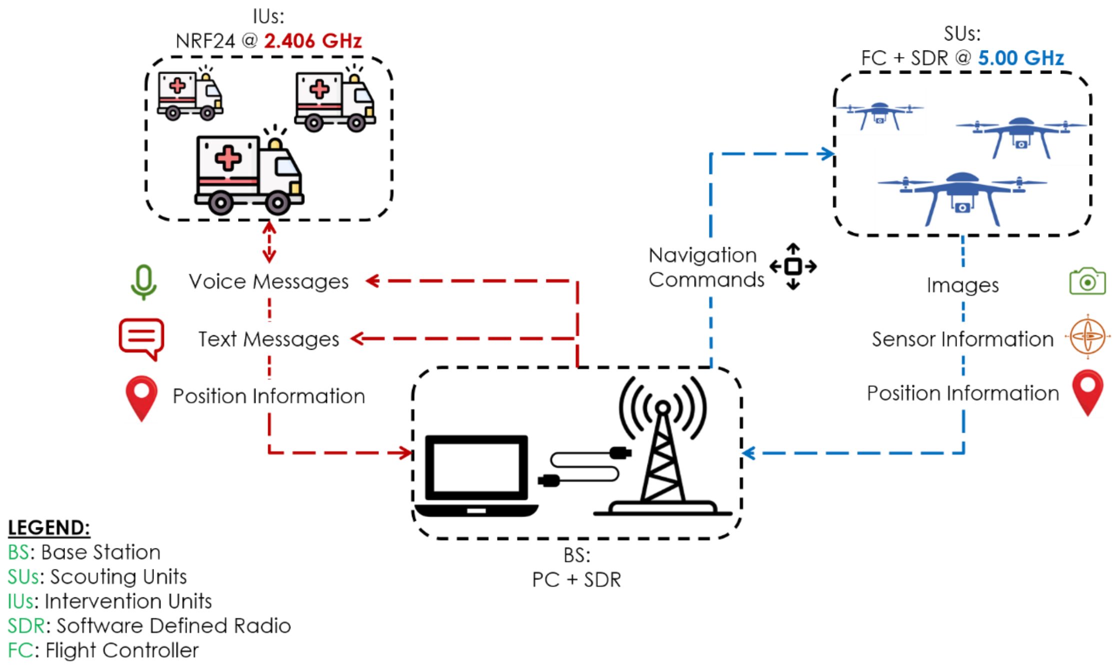


Figure 3-1 System Architecture

The system, as mentioned before, is composed of 3 units: base station, intervention units, and scouting units. The communication channels between the BS and the IUs are held somewhere between 2.4 and 2.5 GHz where this band is very efficient for digital communication protocols.

7.5.1 System Setup

The system is assembled on a lab-scaled level as shown in Figure 3-2. Each unit is described with a separate section below in the document.

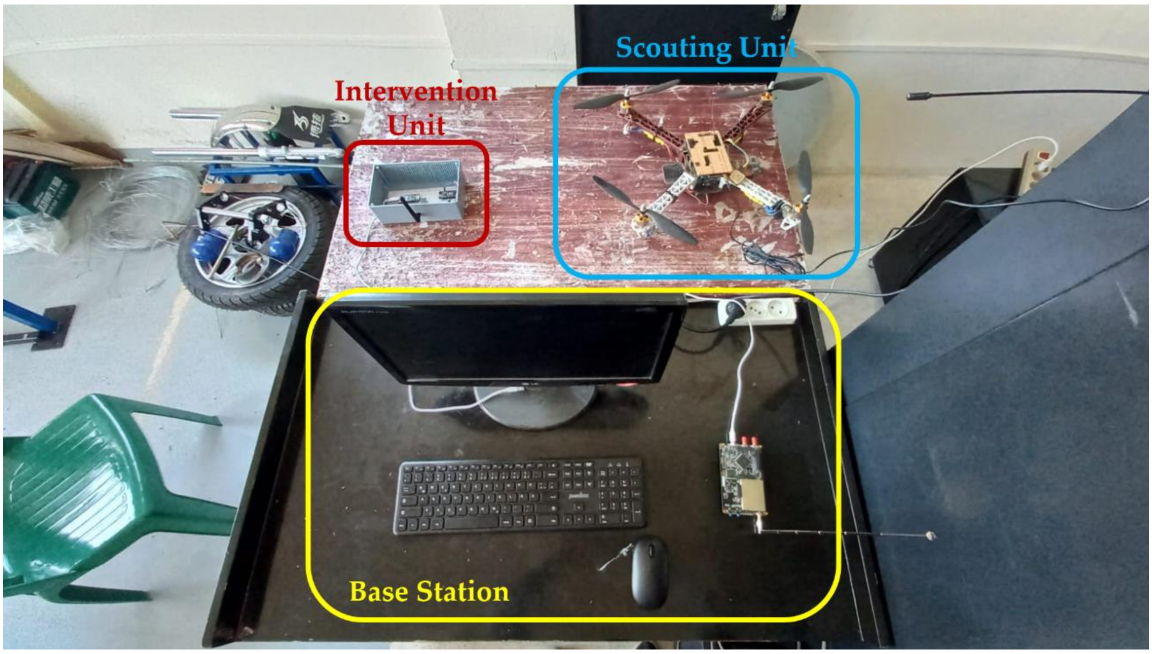


Figure 3-2 System Setup

7.6 Understanding Wireless Range Calculations

One of the key calculations in any wireless design is range, the maximum distance between transmitter and receiver for normal operation.

7.6.1 Power and dbm Calculations

RF power is most commonly expressed and measured in decibels with a milliwatt reference, or dBm. A decibel is a logarithmic unit that is a ratio of the power of the system to some reference. A decibel value of 0 is equivalent to a ratio of 1. Decibel-milliwatt is the output power in decibels referenced to 1 mW.

Since dBm is based on a logarithmic scale, it is an absolute power measurement. For every increase of 3 dBm there is roughly twice the output power, and every increase of 10 dBm represents a tenfold increase in power. 10 dBm (10 mW) is 10 times more powerful than 0 dBm (1 mW), and 20 dBm (100 mW) is 10 times more powerful than 10 dBm. You can convert between mW and dBm using the following formulas:

$$P(dBm) = 10.\left(P(mW)\right)$$

(4-1)

$$P(mW) = 1\,mW.10^{\left(\frac{P(dBm)}{10}\right)}$$

7.6.2 Path Loss

Path loss is the reduction in power density that occurs as a radio wave propagates over a distance. The primary factor in path loss is the decrease in signal strength over distance of the radio waves themselves. Radio waves follow an inverse square law for power density: the power density is proportional to the inverse square of the distance. Every time you double the distance, you receive only one-fourth the power. This means that every 6-dBm increase in output power doubles the possible distance that is achievable.

Besides transmitter power, another factor affecting range is receiver sensitivity. It is usually expressed in –dBm. Since both output power and receiver sensitivity are stated in dBm, you can use simple addition and subtraction to calculate the maximum path loss that a system can incur:

$$Maximum\ path\ loss\ =\ transmit\ power - receiver\ sensitivity\ +\ gains - losses \qquad (4\text{-}2)$$

Gains include any gains resulting from directional transmit and/or receive antennas. Antenna gains are usually expressed in dBi referenced to an isotropic antenna. Losses include any filter or cable attenuation or known environmental conditions. This relationship can also be stated as a link budget, which is the accounting of all gains and losses of a system to measure the signal strength at the receiver:

$$Received\ power\ =\ transmit\ power\ +\ gains - losses \qquad (4\text{-}3)$$

The goal is to make the received power greater than the receiver sensitivity.

In free space (an ideal condition), the inverse square law is the only factor affecting range. In the real world, however, the range also can be degraded by other factors:

• Obstacles such as walls, trees, and hills can cause significant signal loss.

• Water in the air (humidity) can absorb RF energy.

• Metal objects can reflect radio waves, creating new versions of the signal. These multiple waves reach the receiver at different times and destructively (and sometimes constructively) interfere with themselves. This is called multipath.

7.7 Intervention Units

7.7.1 IUs System Architecture

Each intervention unit is supposed to send the location, text, and voice messages to the CS and shall receive the same from the CS. (The architecture is changeable upon the change of requirements).

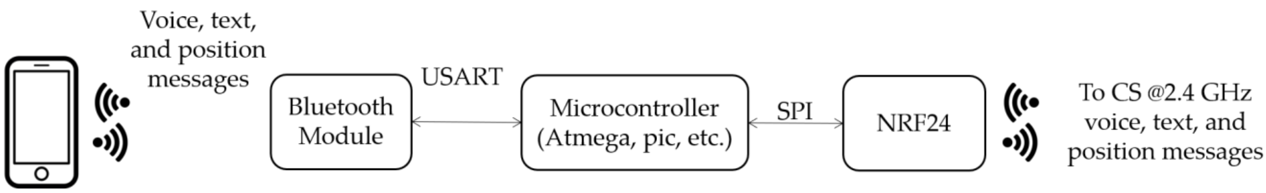
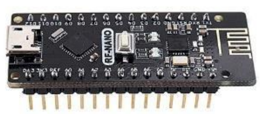


Figure 5-1 Intervention unit basic architecture

Each intervention unit shall connect to an operator's cell phone via Bluetooth interface. The user shall send text and voice data to the unit which will be processed by a microcontroller and fed to the NRF24 module in order to be transmitted by air.

7.7.2 IUs System Components

Table 5-1 Intervention unit system components (for prototyping)

Component	**Description**
Nano-RF24 	• Microcontroller: ATmega328P-MU QFN32 • Bootloader: Newest 1.8.8 • Wireless: Nrf24L01+ 2.4G • BLE chip: TI CC2540 • Work channel: 2.4G – 2.528G • Transmission distance: ~100m • Architecture: AVR • Input voltage: USB power supply, Vin 6-12V • Operating Voltage: 5V • Flash Memory: 32KB of which 2KB used by bootloader • SRAM: 2KB • Clock Speed: 16MHz

7.7.3 NRF24 Signal Interpretation

The first step to establish a communication channel between the base station and the IUs is to detect and decode the signal from the NRF24L01 module and be able to replay a signal back to it. This module modulates its signal using Gaussian Frequency Shift Keying (GFSK) with user-defined channel (2.400 to 2.528 GHz separated at 1 MHz @ 1 Mbps and 2 MHz @ 2 Mbps), air rate (1 or 2 Mbps), and antenna power. The module uses Enhanced **Shockburst packet format** for data transmission which contains a preamble field, address field, PCF field, payload field, and CRC field.

Preamble 1 byte	Address 3-5 byte	Packet Control Field 9 bit	Payload 0 - 32 byte	CRC 1-2 byte

Figure 5-2 An Enhanced Shockburst packet with payload (0 - 32 bytes)

The preamble (1-byte long) is a bit sequence used to detect the 0 and 1 levels in the receiver. It is either 01010101 or 10101010. This is done to ensure there are enough transitions in the preamble to stabilize the receiver.

The address is for the receiver. An address ensures that the correct packet is detected by the receiver. It can be configured as 3 to 5 bytes long. When choosing the address, one shall avoid using repetition in characters or cyclic codes (e.g., 000FFFFFFF or 0101010101) which will raise the packet-error-rate.

The Packet Control Field (PCF) is a 9-bit long sequence which gives information about the payload. This field is only used if the Dynamic Payload Length function is enabled.

Payload length 6bit	PID 2bit	NO_ACK 1bit

Figure 5-3 Packet control field

Payload length gives the information about the user-defined message length sent by the transmitter. It can be either static or dynamic.

The 2-bit PID (Packet Identification) field is used to detect if the received packet is new or retransmitted. PID prevents the receiver from presenting the same payload more than once to the. The PID field is incremented at the transmitter side for each new packet received through the SPI. The PID and CRC fields are used by the receiver device to determine if a packet is retransmitted or new. When several data packets are lost on the link, the PID fields may become equal to the last received PID. If a packet has the same PID as the previous packet, nRF24L01 compares the CRC sums from both packets. If the CRC sums are also equal, the last received packet is considered a copy of the previously received packet and discarded.

The Cyclic Redundancy Check (CRC) is the error detection mechanism in the packet. It my either be 1 or 2 bytes and is calculated over the address, PCF, and payload:

The polynomial for 1 byte CRC is $X^8 + X^2 + X + 1$, Initial value 0xFF. The polynomial for 2-byte CRC is $X^{16} + X^{12} + X^5 + 1$, Initial value 0xFFFF

The goal is to develop a transceiver with the same packet format described by the NRF24 module. In reception mode, the receiver (SDR at the CS side) shall locate the presence of a known address to start interpreting the data. While in transmitting, the CS shall assemble the same packet format in order for the module to receive successfully.

7.7.4 Developing the Transceiver

The IUs are responsible for exchanging data with the base station. Hence, a transceiver architecture is required.

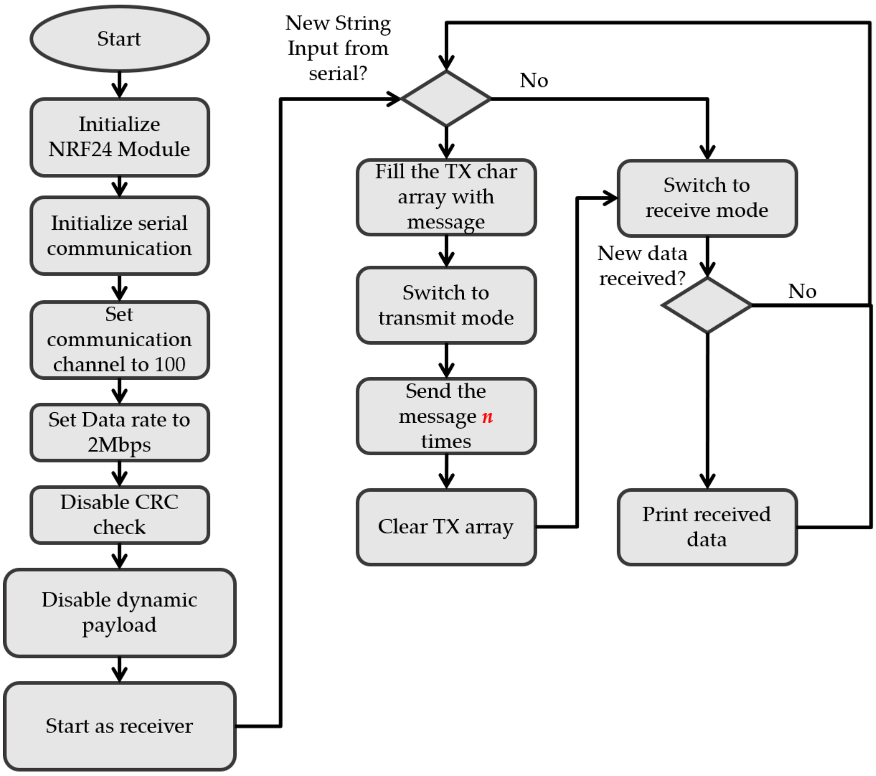


Figure 5-4 IU transceiver flowchart

7.8 Channel Coding

Channel coding, or Forward Error Correction (FEC), is a technique used in wireless communication in order to handle transmission channel uncertainties and errors. The various coding methods that can be employed are achieved by interweaving additional binary digits into the transmission. When decoded on the receiving end, the transmission can be checked for errors that may have occurred and, in many cases, repaired. Other times, the recipient simply asks for the transmission again.

7.9 Scouting Units

Still under implementation

7.9.1 Developing the Transmitter

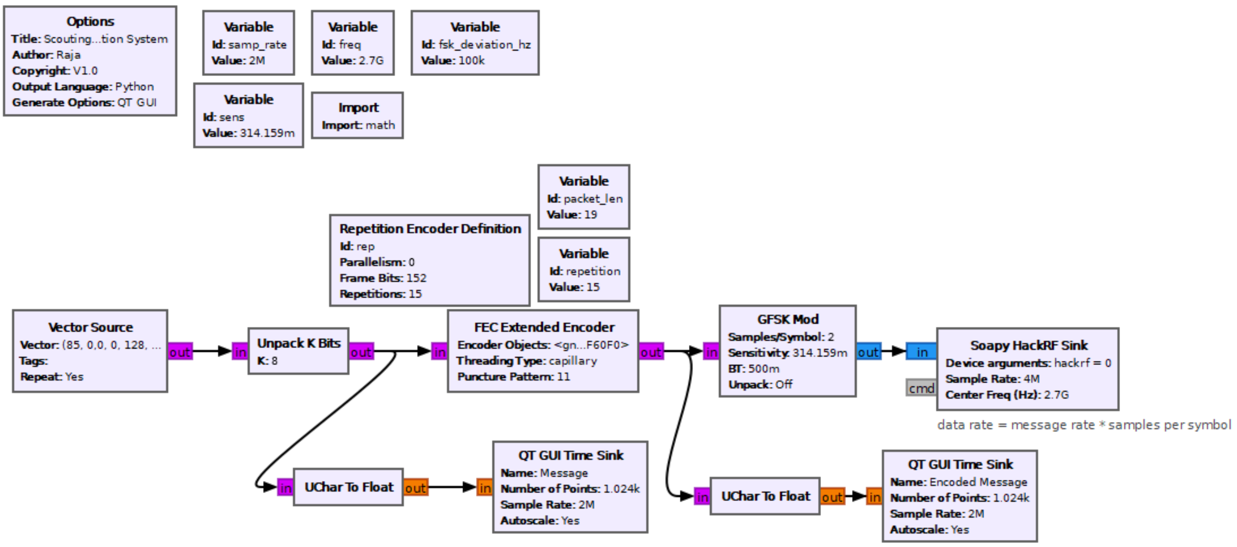
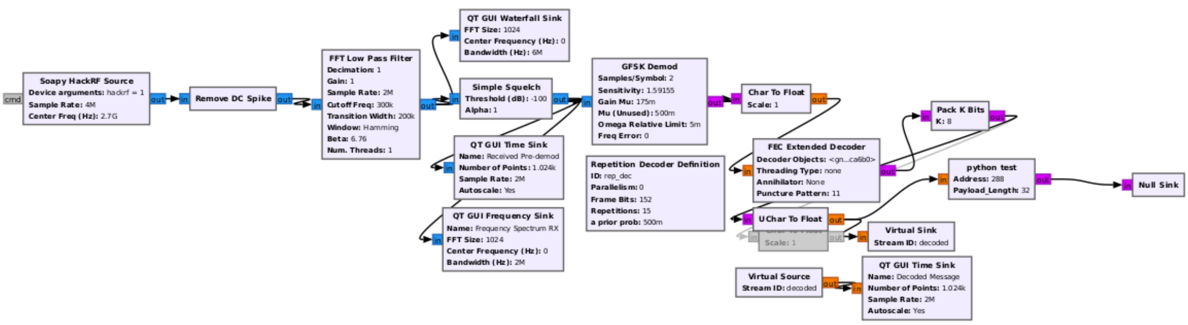


Figure 7-1 SU transmitter flowgraph

The SUs transmitter uses the GFSK modulation to transmit data. The following factors must be considered when designing a robust transmitter:

- **Air data rate:** it is expressed in samples per second.
- **Bandwidth:**
- **Packet length:**
- **Channel coding algorithm:**

7.9.2 Developing the Receiver



We have added the receiver part that plays the role of the base station that receives the data from the transmitter . first of all , we want to check if the preamble is equal to 85 in the packet received because preamble is the header of the packet that indicated the beginning of the packet , then after checking the preamble we want to check the access address to indicate the destination of the packet .

```
import numpy as np

from gnuradio import gr
```

```
import time



class blk(gr.sync_block):  # other base classes are basic_block, decim_block,
interp_block

    """Embedded Python Block example - a simple multiply const"""


    def __init__(self,  address = '0x00', payload_length = 32):  # only default
arguments here

        """arguments to this function show up as parameters in GRC"""

        gr.sync_block.__init__(

            self,

            name='python test',   # will show up in GRC

            in_sig=[np.float32],

            out_sig=[np.byte]

        )

        # if an attribute with the same name as a parameter is found,

        # a callback is registered (properties work, too).

        self.address = address

        self.payload_length = payload_length


    def work(self, input_items, output_items):


        input = input_items[0]

        addr= self.address


        if 85 in input and len(input) > 17:
```

```
            for i in range (len(input)):

                if            input[i]            ==            85            and
int(input[i+1]+input[i+2]+input[i+3]+input[i+4]+input[i+5]) == int(addr):

                    print("Found packet")

        return len(output_items[0])
```

This is our code above , where we are taking the input and checking if the packet contains the preamble (which is 85) then we check the access address if it is the same in the packet .

7.10 Base Station

7.10.1 Base Station Components

The base station components are listed in Table 7-1 below.

Table 7-1 Base Station Components

Component	Description
Hackrf one 	RF Device: • 30 MHz to 6 GHz operating frequency • Half-duplex transceiver • Up to 20 million samples per second • 8-bit quadrature samples (8-bit I and 8-bit Q) • Software-configurable RX and TX gain and baseband filter • Software-controlled antenna port power (50 mA at 3.3 V) • Power amplification up to 15 dBm.
PC	Processing Device: • 64-bit operating system • At least 4 GB RAM • SSD is preferred. • Linux-based OS (Ubuntu 20.04)

7.10.2 Base Station Software Packages

The following software packages must be installed on the ubuntu-based machine for development.

1. GNU Radio Companion: Used to control the HackRf transceiver which will be used between the control station and the scouting units.

```
sudo add-apt-repository ppa:gnuradio/gnuradio-releases

sudo apt-get update
```

```
sudo apt-get install gnuradio python3-packaging
```

2. Arduino IDE (Prototyping): Used to program the nrf24-based controllers for Intervention units.

```
https://www.arduino.cc/en/software
```

3. Hackrf board packages.

```
sudo apt-get install hackrf
```

4. to make sure the hackrf is working correctly, open the terminal and type:

```
hackrf_info
```

7.10.3 Communicating with IUs

IUs use the GFSK modulation technique for message transmission. Hence, a GFSK modulator and demodulator are to be designed and implemented on the base station to handle this communication. Both the receiver and transmitter are put in the same flowgraph but they are separated in this section for explanation reasons.

7.10.3.1 Building the Receiver and Decoder

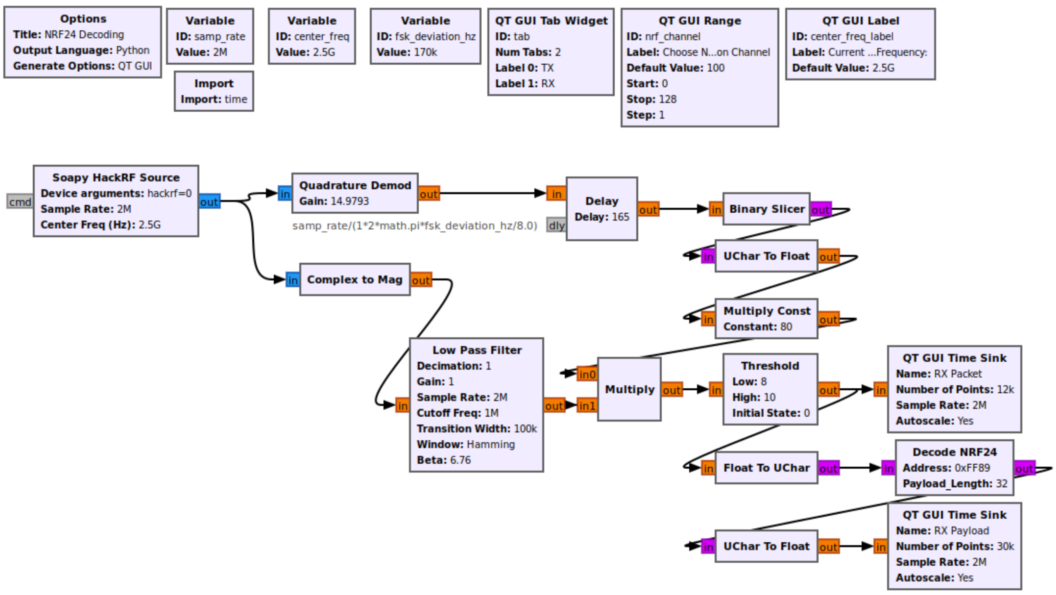


Figure 8-1 Gnuradio NRF24 demodulator and decoder (V1.02)

The soapy HackRF Source block receives the IQ signal from the air. The user should determine some configuration parameters for this block.

Table 7-2 Soapy HackRF source block parameters

Parameter	Description	Type	Variable	Value

Device arguments	Determines the address of the SDR connected to the PC.	String	-	'hackrf = 0'
Sample rate	Determines the required sample rate, in samples per second, of the hackrf device. It should be chosen according to the received air sample rate.	Float	samp_rate	2e6
Center frequency	Determines the center frequency, in Hz, in which the transmitter is transmitting on.	Float	2.4e9 + nrf_channel * 1e6	2.5e9
Bandwidth	This is the expected bandwidth, in Hz, of the received message.	Float	-	1e6
IF Gain	Intermediate frequency gain.	Float	-	25
VGA Gain	Variable Gain Amplifier.	Float	-	30

The received IQ signal is fed into a magnitude calculator and then into a low pass filter to make it smoother. The magnitude calculator transforms the complex IQ signal into just a magnitude. In other words, it only highlights the necessary part of the signal. All the obvious high frequency noises are smoothed via a low-pass filter. The output of the low pass filter is shown below in Figure 7-2 which corresponds to TX identical packets received from the transmitter. The LPF parameters are listed in Table 7-3 below.

Table 7-3 Low pass filter block parameters

Parameter	**Description**	**Type**	**Variable**	**Value**
FIR type	The type of the input and output of this block.	-	-	Float -> Float (Decimating)
Decimation	Decimates the sample rate. i.e., divides the sample rate by this value.	Int	-	1
Sample rate	The sample rate of the message entering this filter.	Float	Samp_rate	2e6
Cutoff frequency	Determines the LPF cutoff freq. Any frequencies above this value will not pass.	Float	-	1e6
Transition width	Is a range of frequencies that allows a transition between a passband and a stopband of a signal processing filter.	Float	-	100e3
Window		-	-	Hamming
Beta	Low pass filter constant	Float	-	6.76

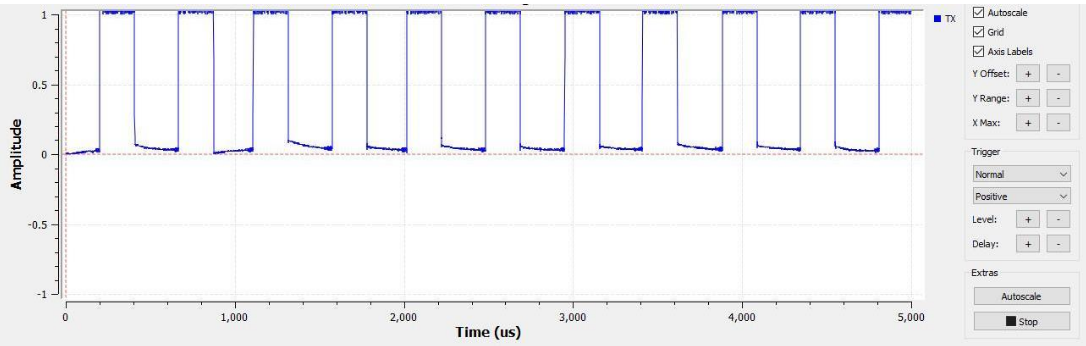


Figure 8-2 Low pass filter response to TX packets being received

On the other side, the IQ signal received, which is GFSK-modulated, undergoes quadrature demodulation which demodulates the FSK signals. This block retrieves the message sent from the transmitter. The output of this block is the signal frequency in relation to the sample rate, multiplied with the gain. Mathematically, this block calculates the product of the one-sampled delayed input and the conjugate un-delayed signal and then calculates the argument of the resulting complex number:

$$Y[n] = arg\ arg\ \left(A^2 e^{j2\pi\frac{f}{f_s}}\right) \quad (7\text{-}1)$$

Where, A is real, and so is A^2, and hence it only scales, therefore $arg\ arg\ (.)$ is invariant which leads to:

$$arg\ arg\ \left(A^2 e^{j2\pi\frac{f}{f_s}}\right) = \frac{f}{f_s} \quad (7\text{-}2)$$

The gain of this block is determined by the sample rate and the FSK frequency deviation (which is 170 KHz) and pugging the numbers:

$$= \frac{f_s}{2\pi\left(\frac{f_{dev}}{8}\right)} = \frac{2e6}{2\pi\left(\frac{170e3}{8}\right)} = 14.9793 \quad (7\text{-}3)$$

The binary slicer block slices a float value producing 1 bit output. Positive input produces a binary 1 and negative input produces a binary zero. In other words, it scales the float input from the quadrature demodulator into 1s and 0s.

The incoming signal may be weak, so a soft amplification has to be done in order to better interpret the signal. Figure 7-3 below shows the output of the binary slicer followed by the gain multiplication of 80. It is clear that the output now is swinging between 0 and 80 instead of 0 and 1.

The following step is to only take the required part of this messy signal. By multiplying the demodulated and amplified signal and the output of the low pass filter (which only focuses on the magnitude of the IQ signal), we can only extract what is necessary and remove what is noise. Hence, the multiply block takes the output of the low pass filter and the output from the amplified quadrature demodulation and outputs the baseband signal.

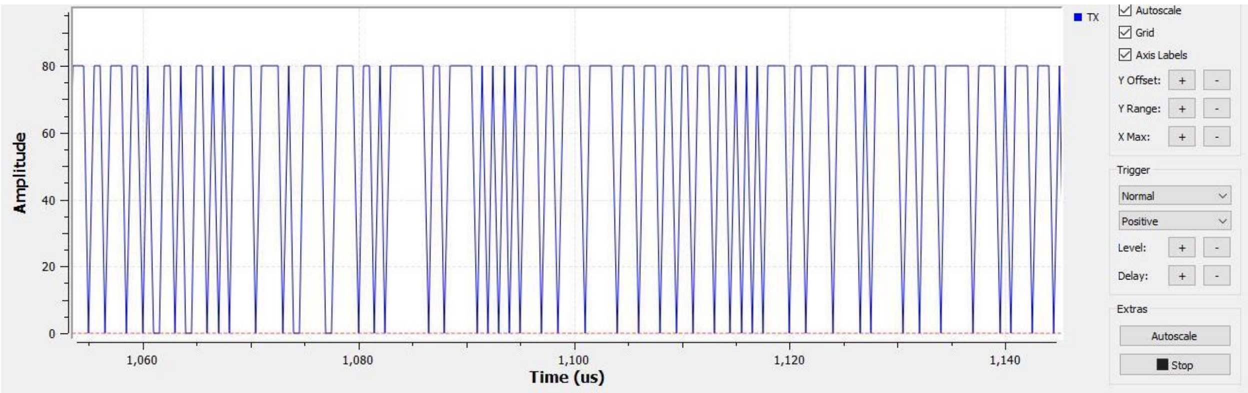


Figure 8-3 Binary slicer and multiply constant output

The output after multiplying the LPF output with the demodulated and amplified signal is shown below in Figure 7-4 and Figure 7-5. It is clearly visible that the noise's amplitude (before receiving the NRF packet) is somewhere between 0 and 4 in the amplitude unit.

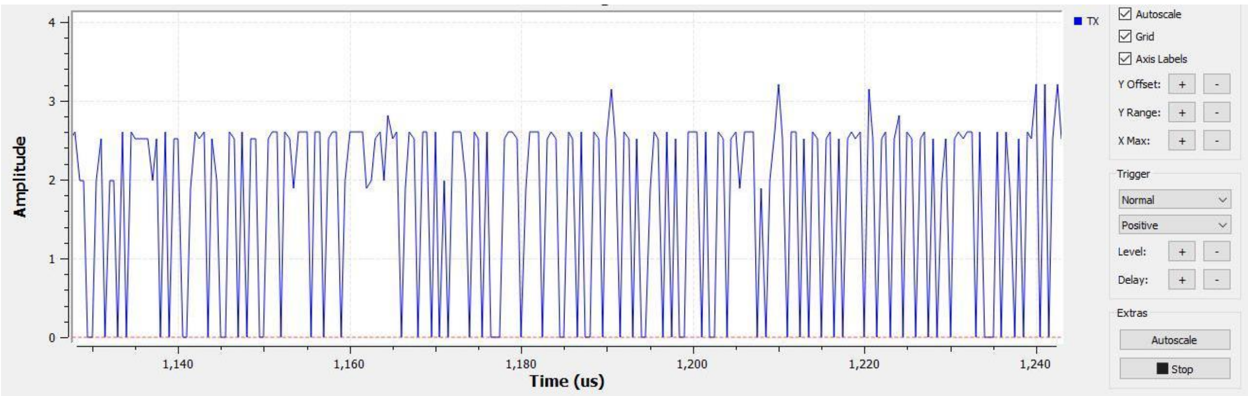


Figure 8-4 Before a NRF packet is received

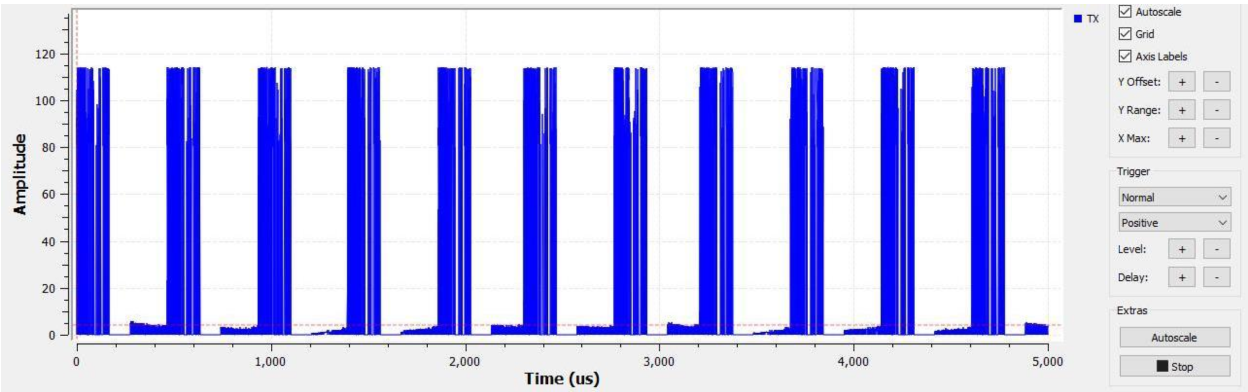


Figure 8-5 After NRF packets are received

Zooming in the above figure:

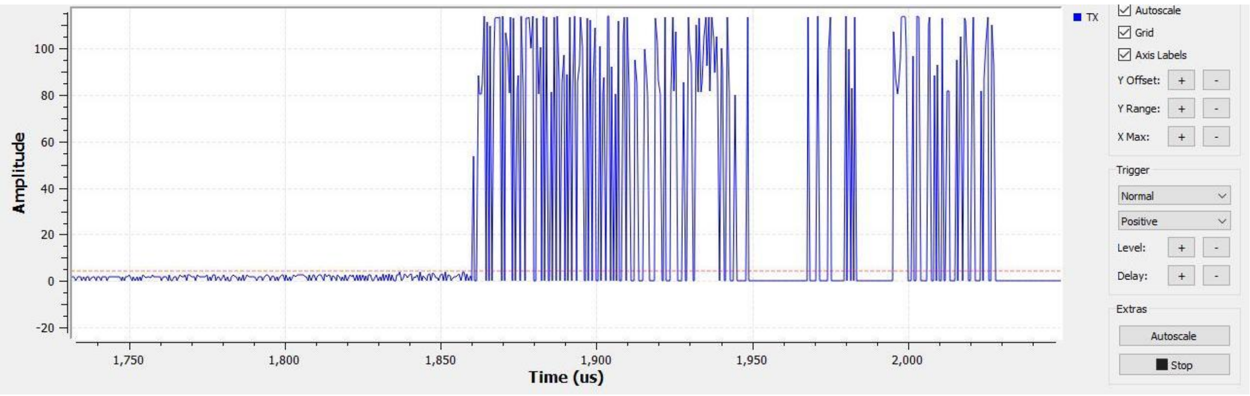


Figure 8-6 After NRF packet is received zoomed in

Figure 7-6 above holds the entire packet within it. But it is still hard to distinguish the packet received from the NRF. Hence, a threshold block is added to filter out the noises from this signal. This block outputs a 1 if the input is greater than its high threshold and a 0 if the input is below its low threshold. The high and low thresholds have been chosen as 10 and 8 respectively. In other words, the necessary information in the signal lying above the amplitude of 10 and anywhere below 8 is considered as noise.

The packet received after adding threshold is shown below in Figure 7-7. The address field is chosen in the Arduino sketch which is completely custom.

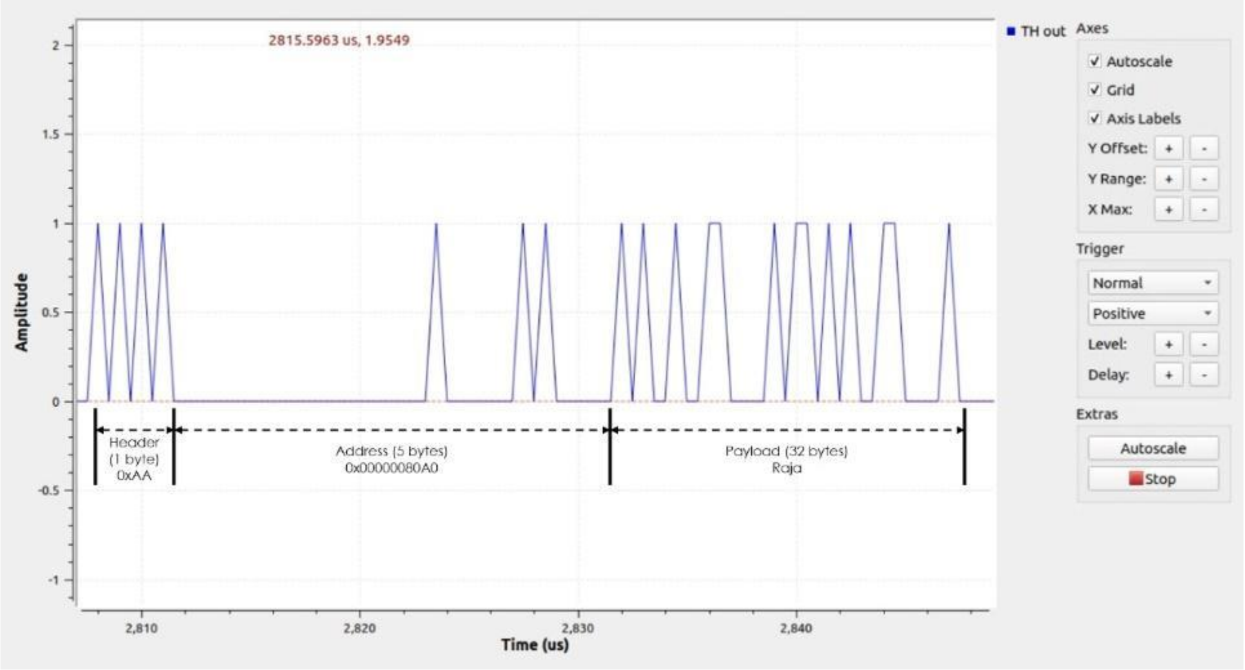


Figure 8-7 NRF24 packet received with PCF field disabled

After successful reception of the packet, the packet must be decoded to extract the payload. So, a custom python block (decode NRF24) is added. The input to this block is a stream of bytes, it accepts the address and the payload length as parameters, and outputs the payload as a stream of bytes. When writing this block, several factors must be considered to ensure proper packet-decoding:

- The input stream is received as a complete list and not element by element.
- The input stream of unpacked bytes (0s and 1s) is in the form of a list, input = [0,1,0,0,1,0,….,1,0], with variable length.

- The input stream sample rate is determined by the block before it, where it is 2Mbps.
- Processing the input stream and manipulating it with complex CPU functions will result in a delayed-sampling period which affects the functionality of the flowgraph.
- The block has 2 main functions; *"init ()"* and *"work ()"*. The *"init"* function is called once when the flowgraph is active and the *"work ()"* function is called whenever there is a new input stream present at the input terminal.

The SW design of this block took into consideration the user custom addresses set by the NRF24 transmitter and the payload size determined also by the TX side.

Table 7-4 Decode NRF24 python block parameters

Parameter	**Type**	**Comments**
Input	Stream of bytes	-
Output	Stream of bytes	-
Address	String of HEX	Max: 5 bytes. Example: FFEEDDCCBB
Payload_length	Int	Min: 1 byte – Max: 32 bytes

As mentioned before, manipulating the input stream with complex functions will absolutely result in delayed-sampling of the input stream and eventually result in bad packet decoding in addition to gnuradio freezing. To solve this problem, the input stream must not be processed when not necessary or when the TX side transmits nothing. The SW design flowchart of this block is shown below in Figure 7-8.

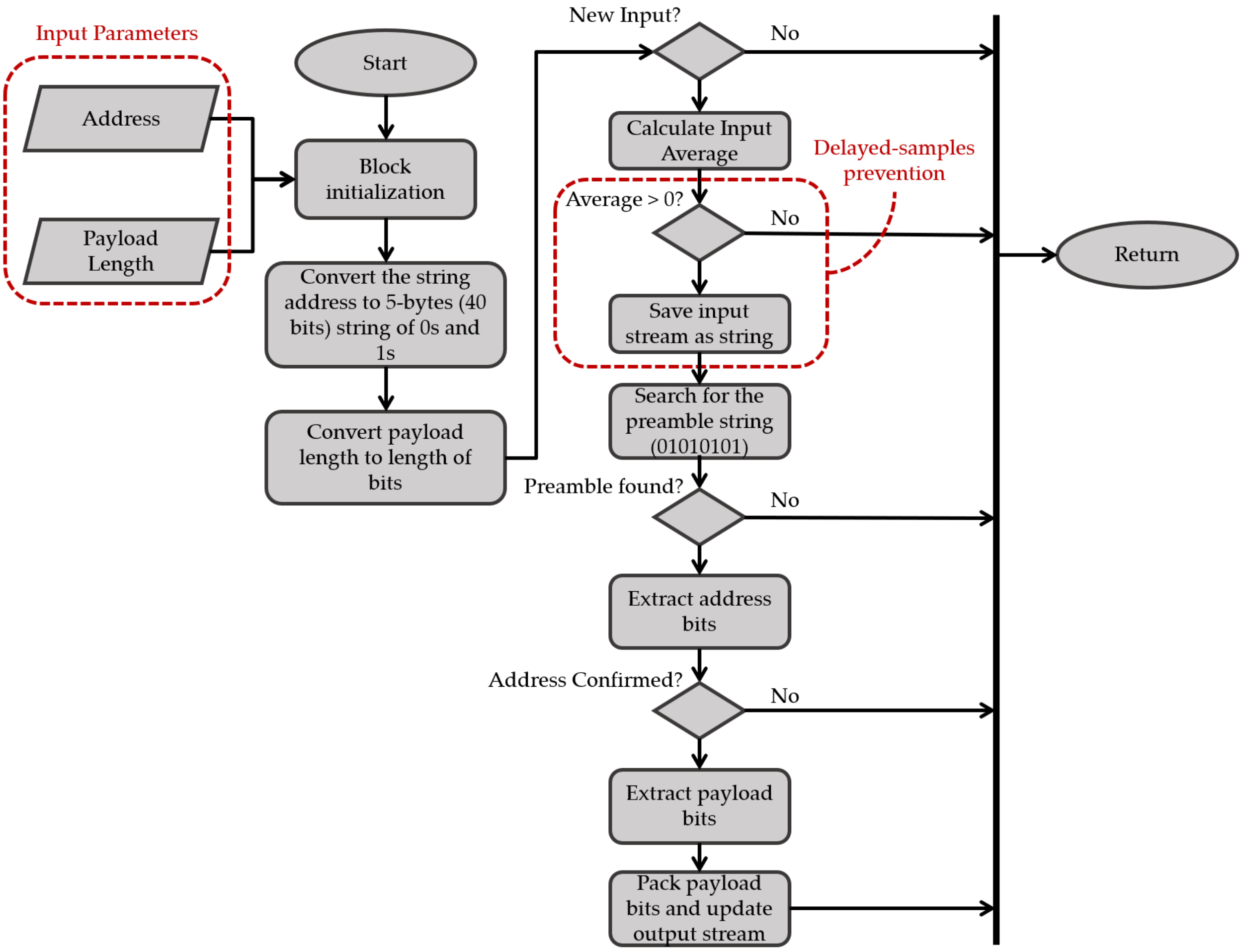


Figure 8-8 Decode NRF24 python block flowchart (V1.02)

The methodology illustrated above processes the input stream as a whole string of bits ("0100101010…"). Thus, the address parameter, which is input from the user, should be expressed as bits and not bytes (ex: 0x80A0 = 1000000010100000). A mapping function is used to map the address string to its equivalent string of bits which is 5-bytes long.

When a new input is present, the average of the list is calculated. If the average is 0, this means that all the input list is 0 and no new packet has arrived and no further processing is required. However, when at least 1 packet is received, the average of the input list becomes different than 0. So, the packet can be processed. This small change calculation will robustly remove the delayed-sampling and process only what is important.

The Decode NRF24 block code is written using VS code and is shown below:

```
"""
Decode NRF24 Packet:
This block receives NRF24 packets at 2 Mbps. If the average of the input list is 0, this
block will ignore the input. Otherwise, useful packet is received, the block will process
the data as a string. This is done to prevent data delayed-sampling.
"""
import numpy as np
from gnuradio import gr
import re
import binascii
import time
```

```
import pmt

class blk(gr.sync_block):  # other base classes are basic_block, decim_block, interp_block

    """NRF24 Decoding - This block decodes the stream coming from NRF24 BTLE
       Parameters:
       self.address: Is the self.address put by the transmitter. The self.address for both 
ends must match to decode
            the message.
       Payload Length: The message length in bytes
    """
    def __init__(self, address = '0x00', payload_length = 32):  # only default arguments 
here
        """arguments to this function show up as parameters in GRC"""
        gr.sync_block.__init__(
            self,
            name='Decode NRF24',   # will show up in GRC
            in_sig=[np.byte],
            out_sig=[np.byte]
        )

        self.address = address
         # Map the address to its equivalent string of bits
        self.address = str(bin(int(self.address, 16))[2:].zfill(8))
        # Complete the string with MSB 0s to complete the 40 bits
         self.address = '0' * (40 - len(self.address)) + self.address
         # Express the payload length in bits instead of bytes
        self.payload_length = payload_length * 8
        self.text_rec_prev = ""

    def work(self, input_items, output_items):
        bits = "" # Variable string to store input stream
        input = input_items[0] # save the input stream
        avg = sum(input) / len(input) # Calculate Input Average

        # if the input stream is different than 0
        # this avoids delayed-sampling
        if avg > 0:
            for i in range (len(input)):
                bits += str(input[i])

        preamble_header = [m.start() for m in re.finditer('01010101', bits)] # Extract the 
preamble sequence
        output_items[0][:] = '0' # Clear all residual data in the output buffer
        if len(preamble_header) > 0:
            for header_index in preamble_header:
                text_rec = "" # Variable to store the characters of the payload

                if  bits[header_index  +  8:  header_index  +  8  +  len(self.address)]  == 
self.address:
                    PCF = bits[header_index + 8 + len(self.address): header_index + 8 + 
len(self.address) + 9]

                    PLD_LENGTH = int(PCF[0:6], 2)
                    PID = int(PCF[6:8], 2)
                    NO_ACK = int(PCF[8], 2)

                    #print(PLD_LENGTH, PID, NO_ACK)
```

```
                    payload = bits[header_index + 8 + len(self.address) + 9: header_index
+ 8 + len(self.address) + 9 + self.payload_length]

                    for x in range(self.payload_length):

                     # Update the output stream
                        output_items[0][x] = payload[x]

                    for pld in range(0,len(payload) - 1, 8):
                        an_integer = int(payload[pld:pld + 8], 2)
                        ascii_char = chr(an_integer)
                        text_rec += ascii_char

                    if self.text_rec_prev == text_rec:
                        print(text_rec) # Print the payload

                    self.text_rec_prev = text_rec

            return(len(output_items[0]))

        return(len(output_items[0]))
```

7.10.3.2 Building the Transmitter

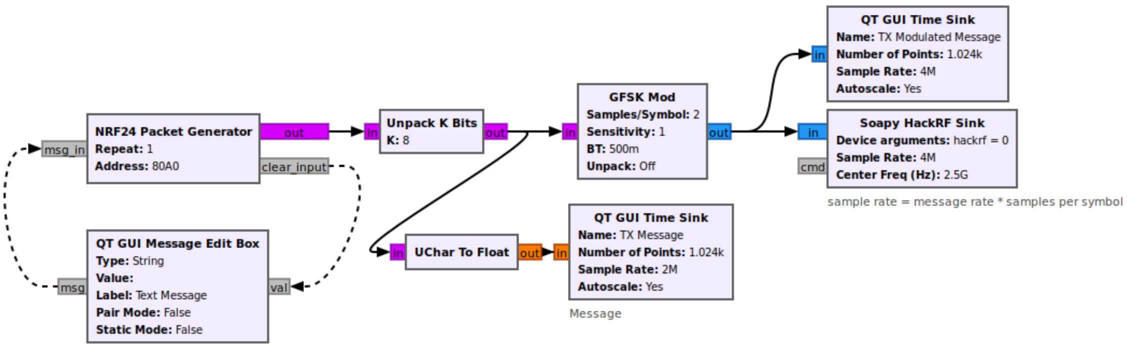


Figure 8-9 Gnuradio NRF24 Transmitter (V1.02)

The transmitter design is the complete opposite of the receiver. The payload must be packetized in the same form of the packet format in Figure 5-2. The whole packet then should undergo GFSK modulation to match the format of the NRF24 packet. Figure 7-9 above shows the gnuradio flowgraph to transmit a text message to the hackRF. The "NRF24 Packet Generator" is a custom block used to combine the payload with the preamble and the address. Note that the CRC checking is disabled in both the gnuradio TX and the NRF24 RX for the aim of simplicity but will be added in the future. This block has the following user-defined parameters:

Table 7-5 NRF24 Packet Generator block parameters

Parameter	**Type**	**Comments**
Address	string	The write address. This must match with the receiver in order to accept the message. Max: 5 bytes
Repeat	int	The number of times to repeat the TX message.
Msg_in	PMT	Receives the payload from a text field editor in form of pmt message.

Clear_input	PMT	Used to clear the text field editor after processing.
Out	byte	Byte of packed bits.

The TX flowgraph must be executed whenever a new string message is entered in the text field in order to not interfere with the RX flowgraph. Thus, the SW design architecture of the packet generator is as follows:

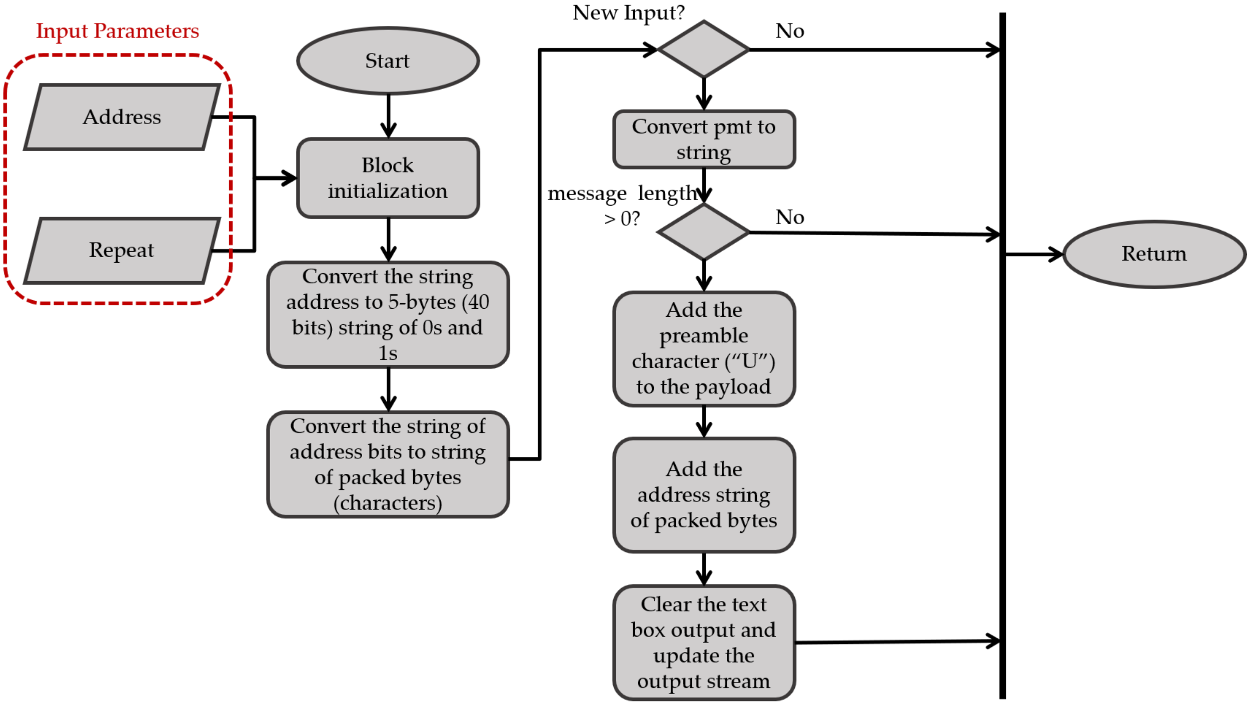


Figure 8-10 NRF24 Packet Generator block flowchart (V1.02)

When writing a message in the textbox editor and set the repeat parameter to 1, the output byte stream looks like:

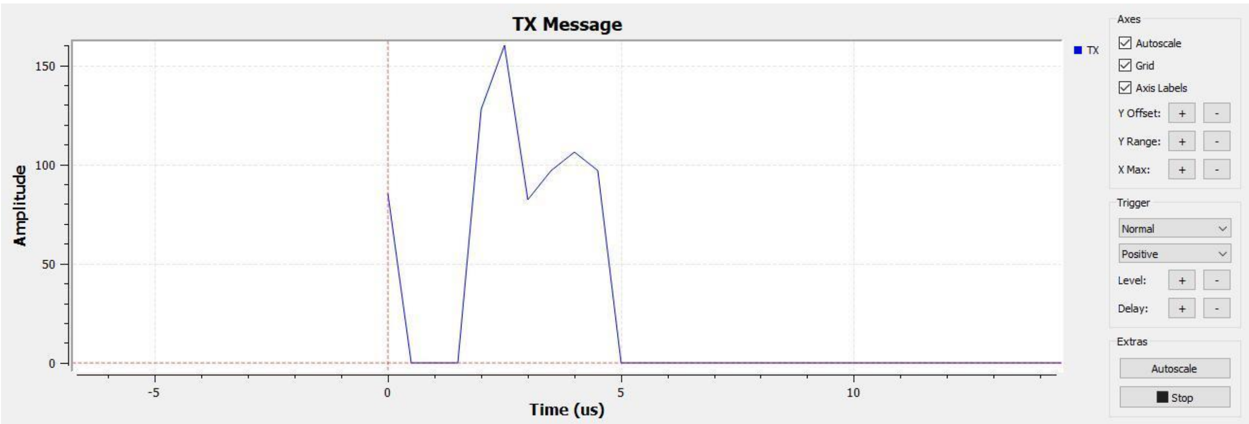


Figure 8-11 Packed TX output byte stream

The code of the NRF24 packet generator block is as follows:

```
import numpy as np
from gnuradio import gr
import pmt
import time

textboxValue = ""

class blk(gr.sync_block):

    """This block generates the NRF24 packet format."""

    def __init__(self, repeat = 5, address = str(00)):  # only default arguments here
        """arguments to this function show up as parameters in GRC"""
        gr.sync_block.__init__(
            self,
            name='NRF24 Packet Generator',   # will show up in GRC
            in_sig=None,
            out_sig=[np.byte]
        )
        self.message_port_register_in(pmt.intern('msg_in'))
        self.message_port_register_out(pmt.intern('clear_input'))
        self.set_msg_handler(pmt.intern('msg_in'), self.handle_msg)

        self.repeat = repeat

        self.address = address
        self.address = str(bin(int(self.address, 16))[2:].zfill(8))
        self.address = '0' * (40 - len(self.address)) + self.address

        self.address_str = ""
        for i in range (0, len(self.address), 8):
            an_integer = int(self.address[i:i + 8], 2)
            self.address_str += chr(an_integer)

    def handle_msg(self, msg):
        global textboxValue
        textboxValue = pmt.symbol_to_string(msg)

    def work(self, input_items, output_items):
        global textboxValue

        for i in range (len(output_items[0])):
            output_items[0][i] = 0

        # get length of string
        _len = len(textboxValue)
        if (_len > 0):
            # Add the preamble and the address to the payload
            textboxValue = "U" + self.address_str + textboxValue
            _len += 1 + len(self.address_str)

            for i in range (self.repeat):
                # store elements in output array
                for x in range(_len):
                    output_items[0][x + (i * _len * self.repeat)] = ord(textboxValue[x])



            textboxValue = ""
```

```
        self.message_port_pub(pmt.intern('clear_input'), pmt.intern(''))
        return (len(output_items[0]))
    else:
        return (0)
```

The packet needs to unpacked before undergoing modulation. Thus, an unpack block is added which unpacks the packet to (8 bits per byte). The packet after unpacking looks like:

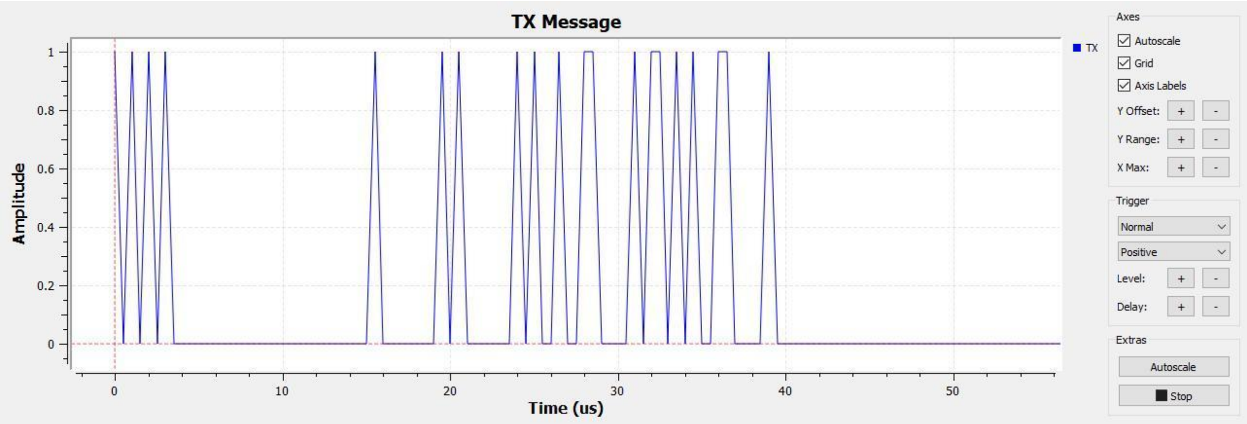


Figure 8-12 Unpacked TX output byte stream

Where the above packet looks very similar to the NRF24 packet detected in Figure 7-7. Thus, the packet is now ready for modulation.

A GFSK modulator block is used to do all the magic. This block converts the stream of unpacked bytes to 2 frequencies:

$$f_{out} = \{f_{base} + f_{\Delta}, \quad if\ message = 1\ f_{base} - f_{\Delta}, \quad if\ message = 0 \qquad (7\text{-}4)$$

Where, f_{base} is the center frequency in Hz and f_{Δ} is the FSK deviation frequency of 170 KHz. The input sample rate of the GFSK modulation block is 2Mbps. The output sample rate of the GFSK mod block can be found as:

$$f_{s_{out}} = samples\ per\ symbol \times f_{s_{in}} \qquad (7\text{-}5)$$

Where, the samples per symbol determine how many frequency deviations are there. By default, this is set to 2 and cannot be less than 2. The output of the GFSK mod block, of the same message as above, is as follows:

Figure 8-13 GFSK modulation block output

Finally, a soapy hackRF block is used to transmit this message. The parameters of the hackRF sink are:

Table 7-6 Soapy hackRF sink block parameters

Parameter	Description	Type	Variable	Value
Device arguments	Determines the address of the SDR connected to the PC.	String	-	'hackrf = 0'
Sample rate	Determines the required sample rate, in samples per second, of the hackrf device. It should be chosen according to the transmitted air sample rate.	Float	samp_rate * 2	4e6
Center frequency	Determines the center frequency, in Hz, in which to transmit the message at.	Float	2.4e9 + nrf_channel * 1e6	2.5e9
Bandwidth	This is the expected bandwidth, in Hz, of the transmitted message.	Float	-	1e6
VGA Gain	Variable Gain Amplifier.	Float	-	47

7.11 How to Use

To run the program, follow the following steps:

1. Open the Arduino transceiver V1.02 sketch using Arduino IDE.
2. Choose Arduino nano from the select board selection bar along with the COM port and then click the upload button (must be done once). The configurable parameters are the address, the communication channel in the radio.setChannel() function (by default it is 100), and the baud rate.

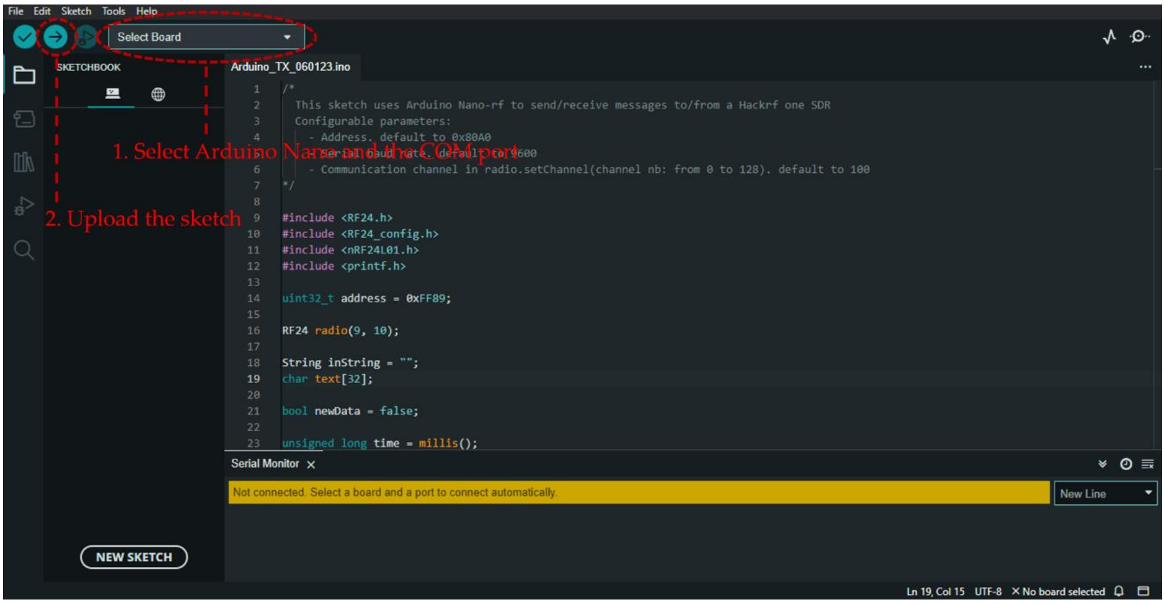


Figure 9-1 Arduino Nano configuration in Arduino IDE

If everything is OK, the Arduino IDE must show a “success” message in the serial monitor.

Figure 9-2 Successful connection to NRF24

3. Type the message you wish to send in the serial monitor.
4. Open GNUradio flowgraph NRF_transceiver_V1_02 and click on RUN.
5. Enter the desired transmit message in the edit box as shown in Figure 8-3.
6. The RX menu in the GUI shown in Figure 8-3 views the received packets from the NRF24 module.
7. The user can change the NRF communication channel from the counter input.

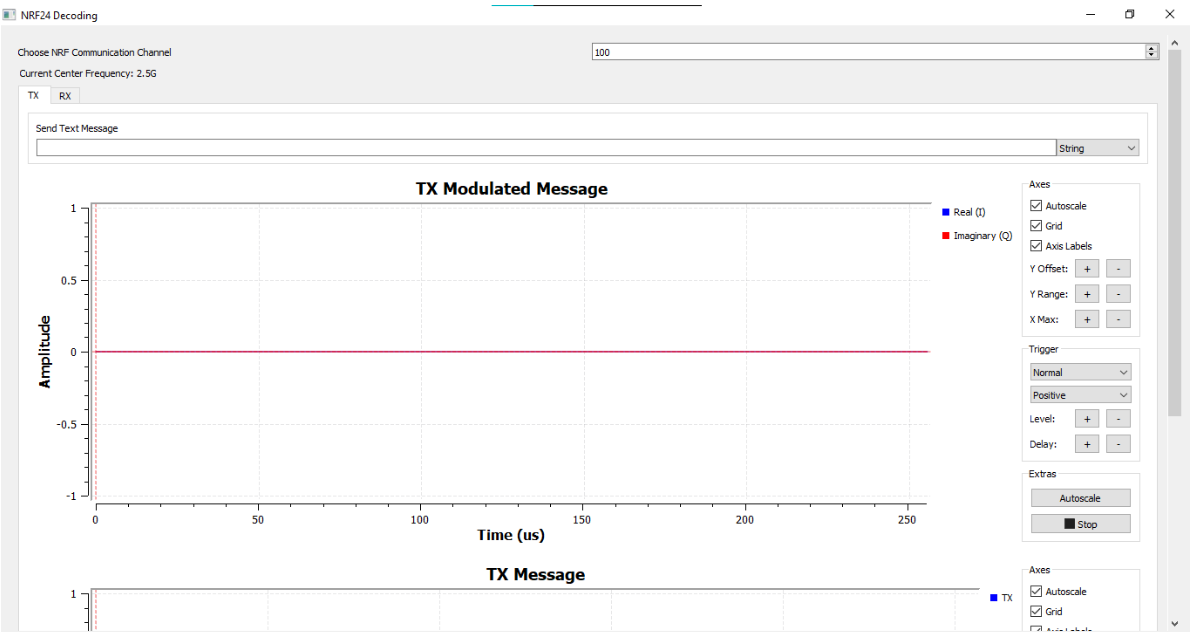


Figure 9-3 Gnuradio GUI TX menu

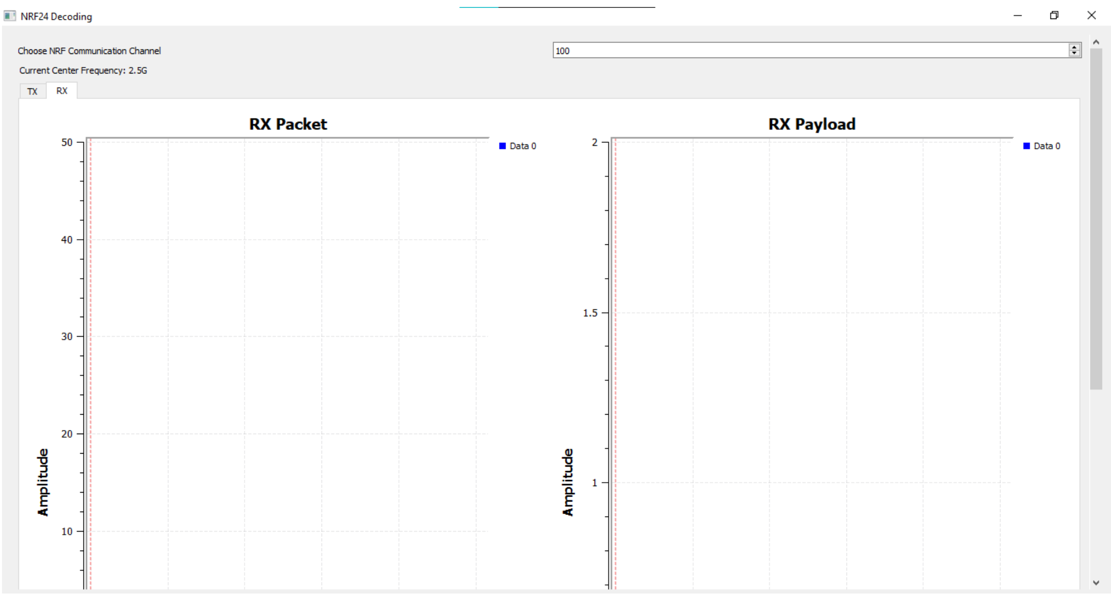


Figure 9-4 Gnuradio GUI RX menu

7.12 Test Cases

7.12.1.1 TS_SU_0001 Range Test, Test Run 1

TS_SU_0001	**Communication Range Test**	**Test Run : 1**
Description	we want to see maximum range between the transmitter HackRf and the receiver HackRf	
Precondition	- 2 hackRfs are on - frequency deviation = 200 Khz - sample rate = 2 Msps - center frequency = 2.7 Ghz	

TS_SU_0001	**Communication Range Test**	**Test Run : 1**
	- VGA gain (TX) = 47 dB - VGA gain (RX) = 62 dB - IF gain (RX) = 20 dB	
Steps	- connect the HackRf (TX and RX) with the usb cable each HackRf on 2 different computers. - check if the HackRf is connected by writing in the terminal "hackrf_info". - then open GNU radio , on the first computer to run the transmitter code and on the other computer we run the receiver code. - after running the code we check the 2 graphs of the receiver and transmitter if they are the same - Then we keep the transmitter HackRf fixed and take the receiver HackRf further away from the transmitter to check the signal at every distance .	
Expected result	We are expecting the signal to be received well until we reach 1 km	

7.12.1.2 TS_SU_0001 Range Test, Test Run 2

TS_SU_0001	**Communication Range Test**	**Test Run : 2**
Description	we want to test different antenna on one of the HackRf	
Precondition	- 2 hackRfs are on - frequency deviation = 200 Khz - sample rate = 2 Msps - center frequency = 2.7 Ghz - VGA gain (TX) = 47 dB - VGA gain (RX) = 62 dB - IF gain (RX) = 40 dB	
Steps	we put on the transmitter a black antenna and on the receiver one the same antenna remains as the first test run	
Expected result	We are expecting the signal to be received well until we reach 1 km , then the signal	

7.12.1.3 TS_SU_0001 Range Test, Test Run 3

TS_SU_0001	**Communication Range Test**	**Test Run : 3**
Description	we want to test different antenna on both of the HackRf	
Precondition	- 2 hackRfs are on	

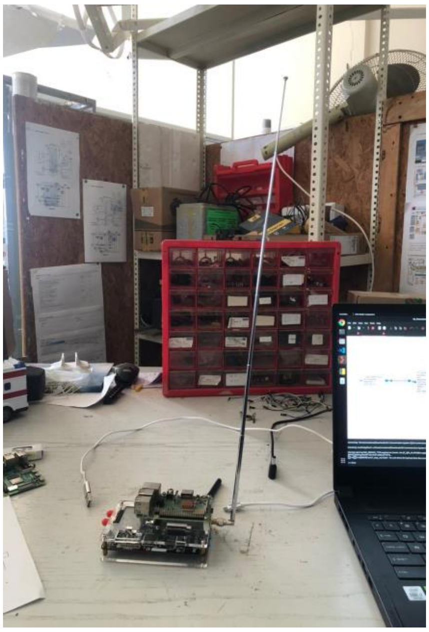
TS_SU_0001	Communication Range Test	Test Run : 3
	- frequency deviation = 200 Khz - sample rate = 2 Msps - center frequency = 2.7 Ghz - VGA gain (TX) = 47 dB - VGA gain (RX) = 62 dB - IF gain (RX) = 40 dB	
Steps	we put on both of the transmitter and the receiver a black antenna	
Expected result	We are expecting the signal to be received well until we reach 1 km	

7.12.2 Test Cases Results

7.12.2.1 TS_SU_0001 Range Test, Test Run 1

TS_SU_0001	Communication Range Test Results	Test Run : 1
Results	After the transmitter and the receiver ran well , we had a problem with the range . We expected the range to be at least 1 km, but after we reached 30 m there was noise in the signal and when we exceeded 40 m , the signal was completely corrupted.	
Expected Reasons	we think that : - we have to change the antenna - we have problem in channel coding	

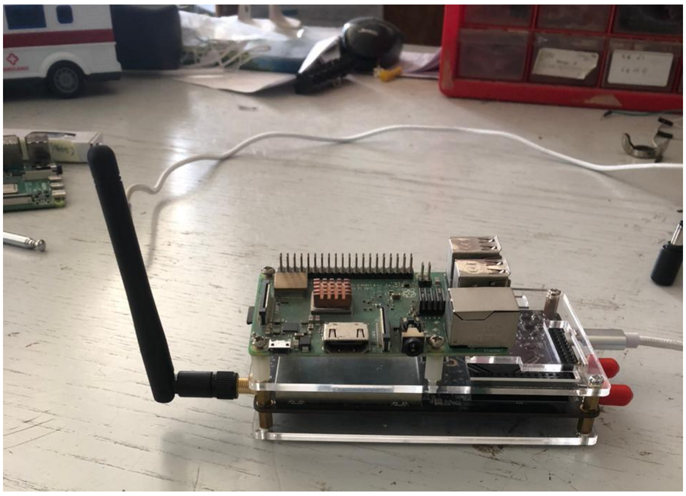
We used this antenna on both the transmitter and receiver HackRf in the test run : 1



7.12.2.2 TS_SU_0001 Range Test, Test Run 2

TS_SU_0001	**Communication Range Test Results**	**Test Run : 2**
Results	After changing on of the antenna the signal was more stable then the test run 1, and the range was so far , on line of sight	
Expected Solutions	we think that : - we have to change both antenna to the black ones	

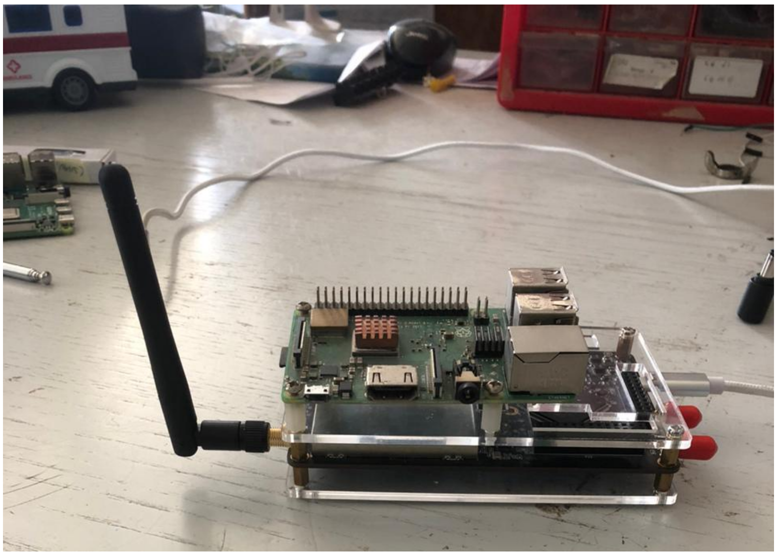
We used this antenna on the transmitter HackRf in the test run : 2



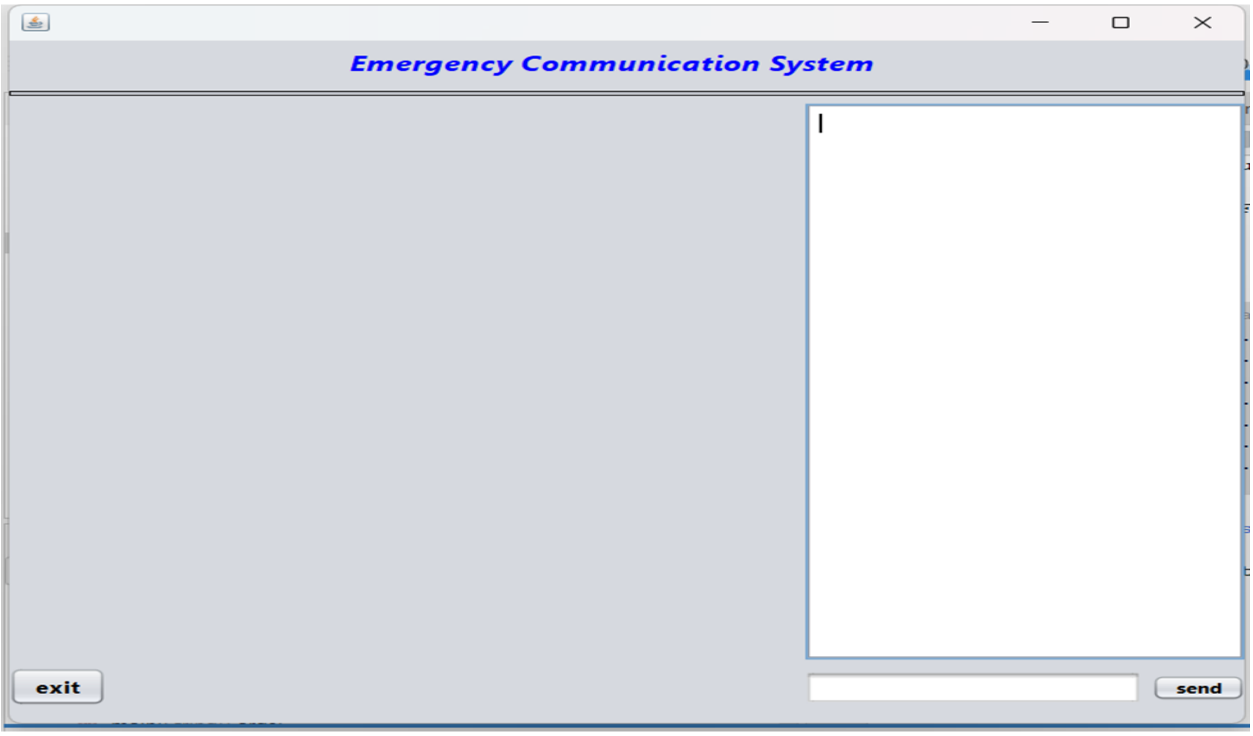
7.12.2.3 TS_SU_0001 Range Test, Test run 3

TS_SU_0001	**Communication Range Test Results**	**Test Run : 3**
Results	After changing on of the antenna the signal was more stable then the test run 2, and the range way greater the the previous test run , on line of sight	

We used this antenna on both of the transmitter and the receiver HackRfs in the test run : 3.



7.13 GUI for base station



The GUI here used to be a programme for the whole emergency communication system to control all actions like the order to the SUs and the IUs to give the gps and the pictures of any needed unit.

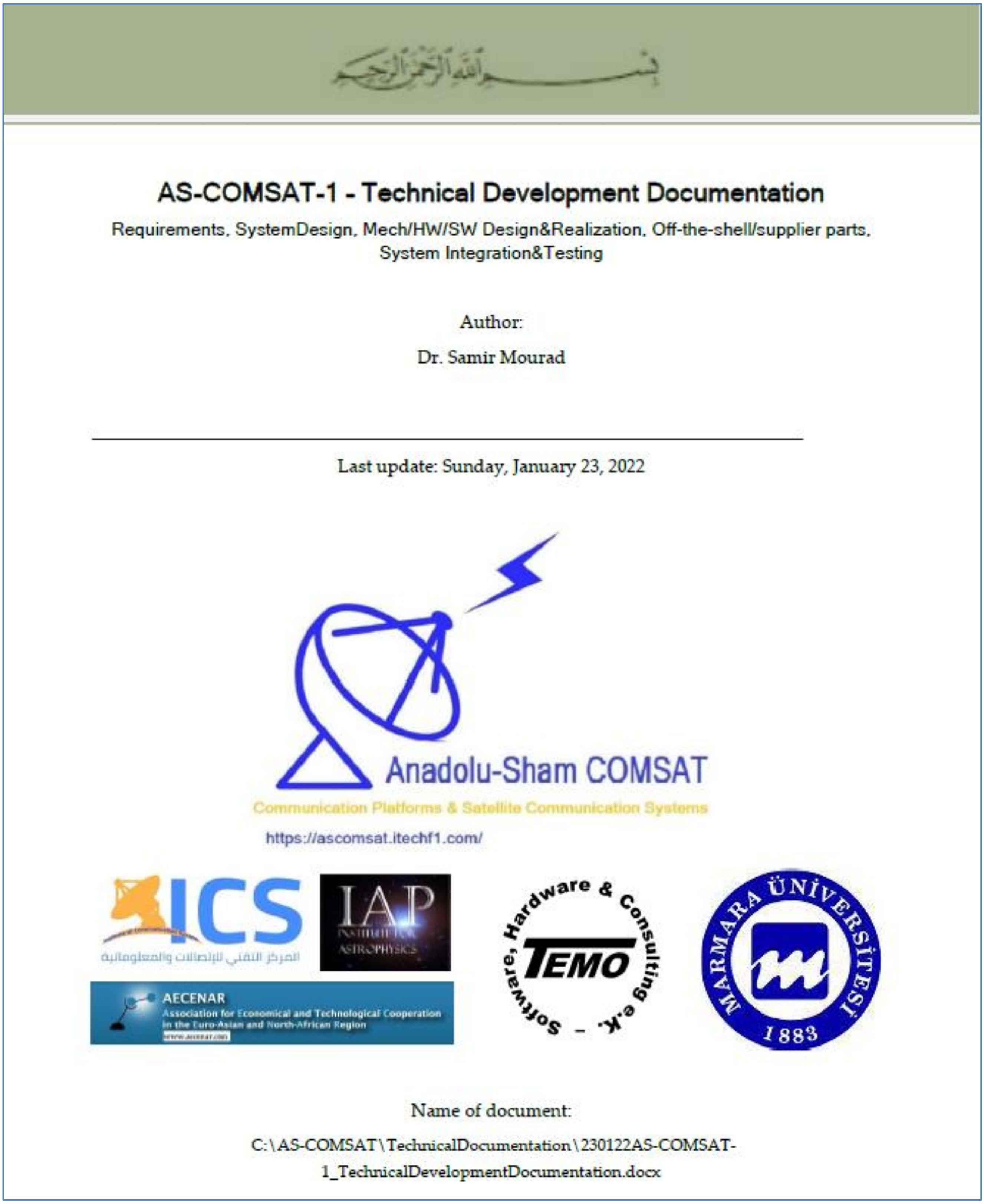
7.13.1 How it works

First of all we have to open the system we have a list view that contains all the active units each unit has her own channel when you press on the desired channel you can now write in the text field the command navigation to SUS(drones) and the voices or the text messages IUS(ambulance) to give their data in GPS, PICTURES, VOICES and text messages.

7.13.1.1 the reached level

We put the text field, text area, and 2 buttons which are programmed in the exit button to close the system and the send button that we programmed till now just all we write in the text field to the text area once it sends all the written words will disappear automatically.

8 Cubesat AS-COMSAT-1, Version 2021: Technical Development Documentation (Development with off-the-shell components)



بسم الله الرحمن الرحيم

AS-COMSAT-1 - Technical Development Documentation

Requirements, SystemDesign, Mech/HW/SW Design&Realization, Off-the-shell/supplier parts, System Integration&Testing

Author:

Dr. Samir Mourad

Last update: Sunday, January 23, 2022

Anadolu-Sham COMSAT

Communication Platforms & Satellite Communication Systems

https://ascomsat.itechf1.com/

Name of document:

C:\AS-COMSAT\TechnicalDocumentation\230122AS-COMSAT-1_TechnicalDevelopmentDocumentation.docx

8.1 Requirements Database (RDD)

8.1.1 System Requirements

8.1.1.1 Orbit Specification, Number of required satellites[12]

For AIS application:

- 650 km altitude, elliptic semimajor axis a = 7027.748 km
- 10 minutes per satellite visibility - > 10 satellites
- - > 10 satellites
- - inclination: 100° - 127°

Analysis of required number of satellites, inclination und height

The ground station can receive AIS signals in a radius of 40-40 km when using an antenna 15 m above sea level. However, Terrestrial stations placed on a higher altitude may be able to extend the radius and get the signal from to 70-100 km subject to some other factors like weather, elevation, external antenna, and obstacles around it (1).

The altitude of the station antenna plays an important role to get better reception. Base stations in higher elevation extend the range and provide efficient reception and get signals from distant ships and vessels.

However, terrestrial network is not yet efficient for vessel tracking services because of its limitations considering seas and oceans. They are inevitable to be used in ports for clustering, traffic management, path prediction, identification and to get better understating of the current situation at ports and coastal areas.

But when going far into the open sea, it is hard to get the overall picture of what is going on there using terrestrial network-based receivers, but it does help in navigation for the vessels themselves and ensure the safety of the ships by providing collision avoidance solutions so the ships will be aware about each other and the location of each vessel. Thus, this can only work in coastal zones and ship-to-ship zone.

Satellite based AIS data can be more useful in the seas and oceans when no terrestrial base station can be found to receive and analyze AIS signals. It can provide a global and yet complete picture of the world's maritime network.

When satellites are used to handle these data, term S-AIS is used. While terrestrial network can be ideal solution for real-time vessel tracking and positions coverage at thousands of coastal areas and ports, satellites are promising solution to the next generation the AIS devices which need to be improved also to work better with the satellite-based networks. Because current AIS devices can not make better use of all the solutions, features and functions that satellites can provide.

CubeSat AIS receivers can be used to detect AIS signals in LEO orbit at the altitude of 650km (2). Satellites in higher altitudes may suffer from the propagation, AIS packets collision due to Faraday effect which causes the signal's polarization plane to rotate subject to the elevation

[12] [5]

angle and magnetic flux intensity. Doppler effect also makes the signal to overlap. Hence, the detection probability has to be at its highest which can be obtained by satellites operating in LEO.

Orbital Period

The orbital period is defined by the time taken by an astronomical object to rotate in its orbit around another for one complete revolution.

A satellite's period is the amount of time it takes to make one full orbit around a planet, Earth. If the satellite is placed high above the surface of the planet, it will take a long time to complete its orbit. However, if the satellite gets closer to the surface of the planet, or to a lower altitude, it will take less time to complete its revolution - and its period will be shorter.

The time needed for a satellite for one period in the orbit is given by the formula:

$$T= \frac{2\pi}{\sqrt{\mu}}(R_e+z)^{3/2}$$

Where μ is the gravitational constant for the Earth equal to 398600 km^3/sec^2.

- Re is Earth's equatorial radius 6378 km.
- z is the altitude.

Thus, a satellite launched into a sun-synchronous circular orbit with altitude 650 km will need a period of 97.72 minutes.

The period determines the semimajor axis a:

$$T= \frac{2\pi}{\sqrt{\mu}}(a)^{3/2}$$

Thus, the semimajor axis for the altitude 650 km and period 97.72 minutes is:

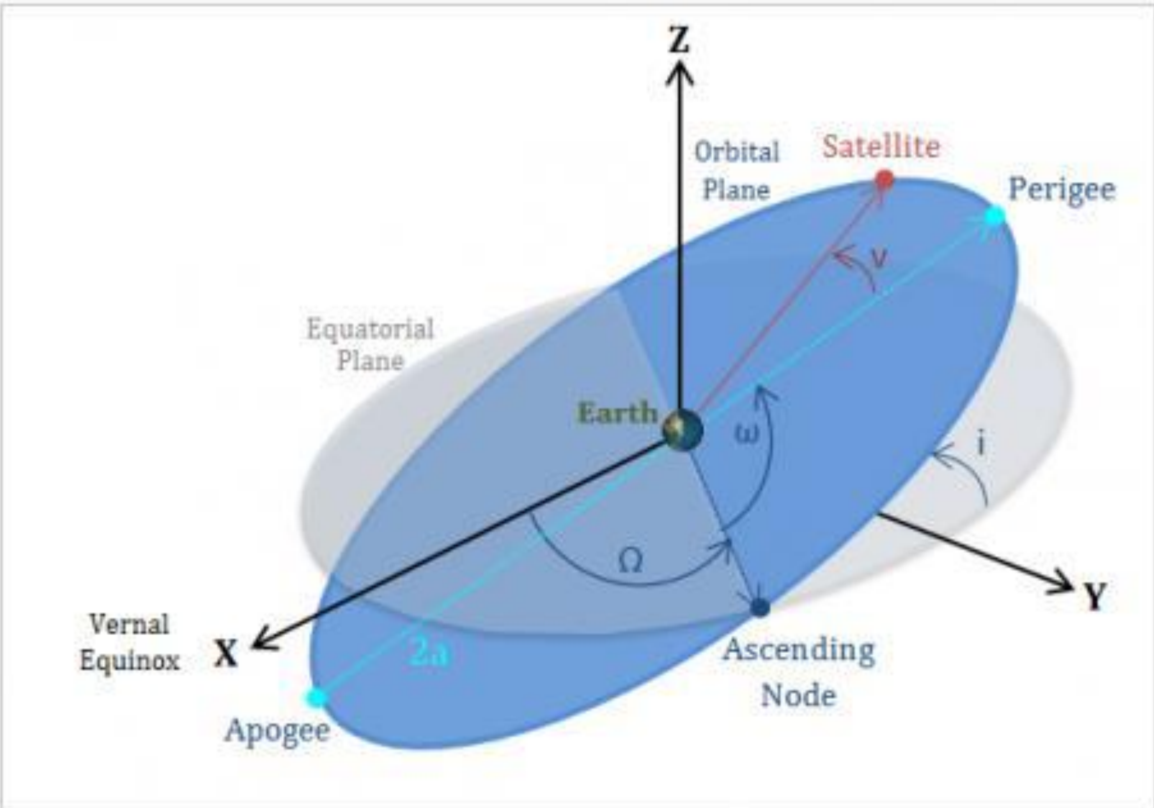
a= 7027.748766 km

Inclination Angle

The angle of inclination is the angle between the reference plane and the direction axis. It is an orbital element that determine the shape and direction of astronomical orbits.

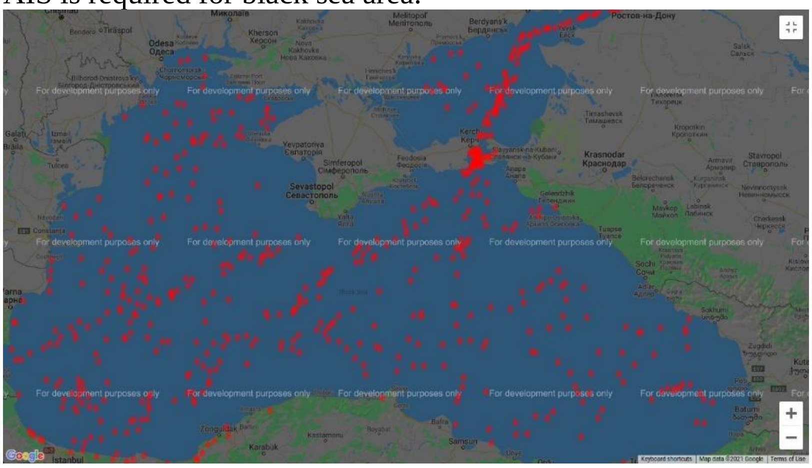
Description	The orbit plane of a satellite that rotate around the Earth straight above the equator line is the same as the equatorial plane of the Earth, and the orbital inclination in this case is 0°. **Inclination Angle**
The orbital object has a prograde orbit in the equatorial plane of the planet.	0°
Prograde orbit but not as same as the equatorial plane.	0°<∝<90°
Polar orbit, satellite passes over the poles of the planet.	90°
Critical inclination, zero apogee drift in elliptical orbits.	63.4°

Retrograde orbit, in which the satellite's position on the equatorial plane is projected in the opposite direction of the Earth's rotation.	90°<∝<180°
Retrograde equatorial orbit.	180°



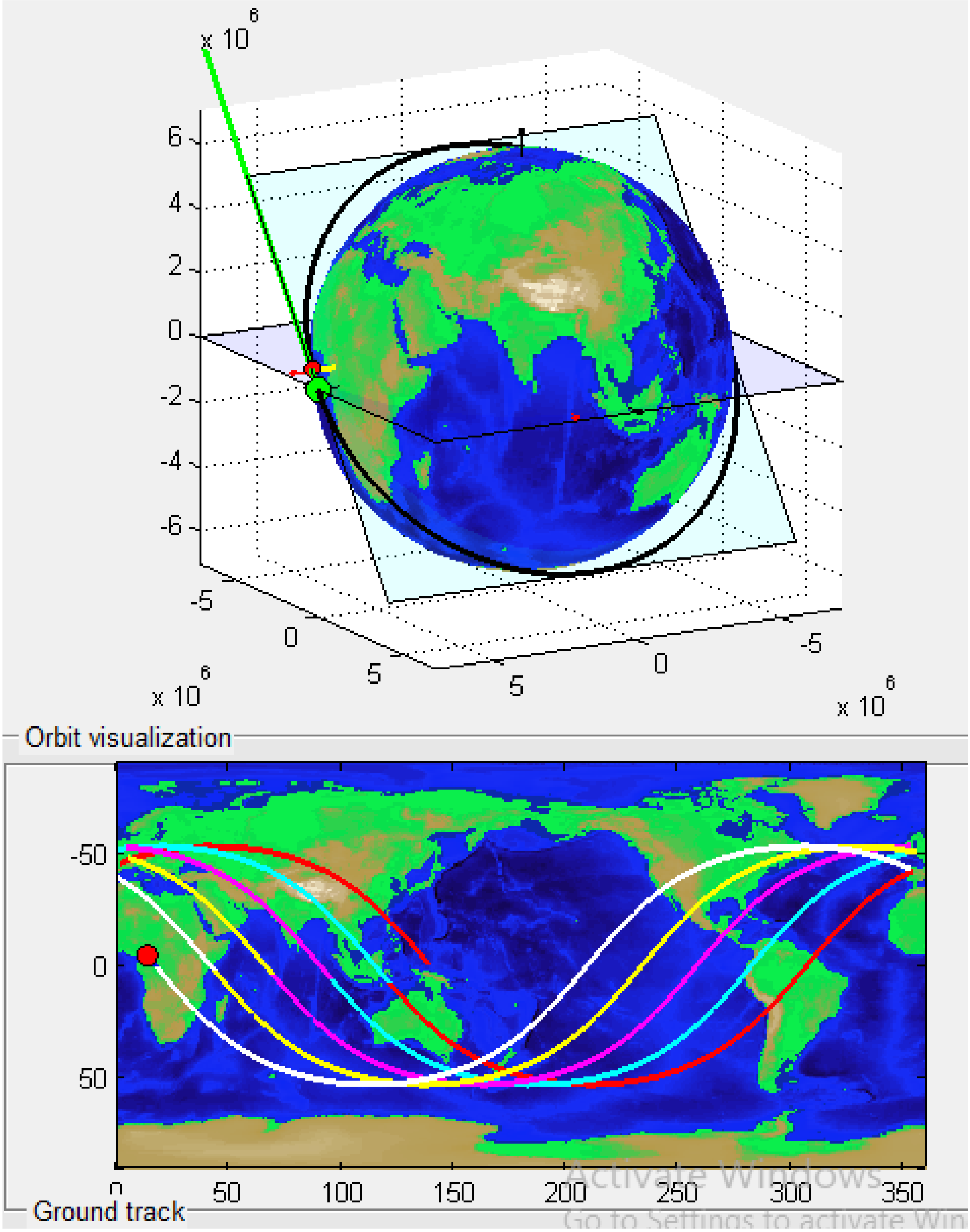
Orbital plane with its inclination (i) with respect to the equatorial plane, right ascension of the ascending node (Ω), argument of Perigee (ω) and true anomaly (v). The vernal equinox, the ascending node, the perigee and the equatorial plane are represented.

AIS is required for black sea area:



Considering Black Sea area, polar sun-synchronous orbits with 127° inclination at altitude 650 km and semimajor axis of 7027.748766 km with a revolution period of approximately 97.72 min is by far accepted to handle AIS data from this area.

Below, is a graph with five periods of the calculated orbital elements



1: The automatic identification system (AIS) : a data source for studying maritime traffic : The case of the Adriatic Sea. Arnaud Serry, 2017

2: Design of Automatic Identification System (AIS) Receiver for Low Earth Orbit (LEO) Satellite

Analysis with MATLAB AddOn "sat-lab":

SAT-LAB: A MATLAB Graphical User Interface for simulating and visualizing Keplerian satellite orbits

D. Piretzidis and M.G. Sideris
Department of Geomatics Engineering, University of Calgary
dimitrios.piretzidis@ucalgary.ca

SCHULICH School of Engineering

GEOMATICS ENGINEERING

INTRODUCTION

SAT-LAB is a MATLAB-based Graphical User Interface (GUI), developed for simulating and visualizing satellite orbits. The primary purpose of SAT-LAB is to provide a software with a user-friendly interface that can be used for both academic and scientific purposes. The calculation of the satellite state vector (position and velocity) is done using a Keplerian propagator. After selecting the six Keplerian elements, the computation and visualization of the satellite orbit is performed simultaneously and in real time. Both the satellite orbit and the state vector at each epoch are given in two reference frames, i.e., the Inertial Reference Frame (IRF) and the Earth-Fixed Reference Frame (EFRF). For the EFRF, both the 3D Cartesian coordinates and the ground tracks of the orbit are provided. Other visualization options include selecting the appearance of the coastline, topography/bathymetry, satellite orbit, position, velocity and radial distance, and IRF and EFRF axes. SAT-LAB is also capable of visualizing orbits of operational satellites and real-time tracking of their position.

CURRENT LIMITATIONS

SAT-LAB users should use the current version only for educational purposes or for low-accuracy simulations of satellite orbits. Scientific usage is not yet advised. Satellite positions are calculated with an accuracy of several kilometers due to the following limitations:

- A Keplerian propagator is used, which cannot accurately describe the motion of a satellite orbiting a planet, especially in a low orbit. In low orbital altitudes, planet flattening and non-gravitational forces highly affect the satellite orbit.
- Two-line element (TLE) ephemerides are used for the computation and real-time tracking of operational satellites. Although TLE ephemerides provide the six Keplerian elements, they are suitable for use only with simplified perturbation models (e.g., SGP, SGP4, SDP4, etc.) and not with a Keplerian propagator. Even using simplified perturbation models, the accuracy of the satellite position using TLE ephemerides can reach several kilometers per day.
- Only the Earth's rotation is taken into account for the transformation of satellite position and velocity from IRF to EFRF. Formally, precession, nutation and polar motion effects should be taken into account as well.

SOFTWARE DESCRIPTION

The SAT-LAB GUI is presented in Figure 1. The main form consists of the 9 elements provided below, following the same numbering as in Figure 1:

1) Menu bar, which contains the "Satellite Data" menu and two submenus that allow the user to download orbital data from operational satellites and access their orbital elements, as well as other information.
2) "Inertial Reference Frame" panel, which shows the satellite orbit in the IRF.
3) "Earth Fixed Reference Frame" panel, which shows the satellite orbit in the EFRF.
4) "Earth Fixed Reference Frame (Ground Tracks)" panel, which shows the satellite orbit in the EFRF after converting the 3D Cartesian coordinates of the satellite to geodetic coordinates.
5) "Select Keplerian elements" panel. When the user selects or changes the Keplerian elements, SAT-LAB produces and simulates the orbit in real time.
6) "Select appearance" panel. The user can control which orbital and geometric components should be shown and can select their color. The changes are implemented in real time.
7) "Select resolution" panel. The user can define the spatial resolution of the topography/bathymetry (terrain), coastline and day/night map (for real-time positioning of operational satellites). For optimal performance, a medium resolution is recommended.
8) "Select satellite" panel. The user can select the orbit and real-time position of selected operational satellites.
9) "Animation" panel. Selecting at least one reference frame and clicking on the "Animate orbit" button, SAT-LAB produces an animation of the current satellite orbit.

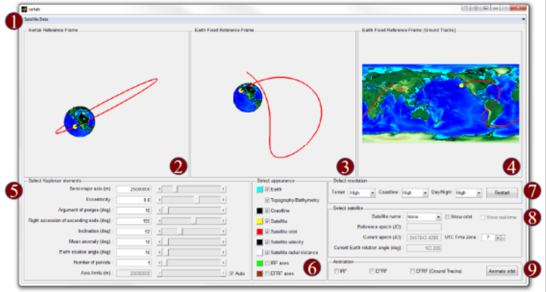


Figure 1 – SAT-LAB main form

ORBIT SIMULATION AND VISUALIZATION

Orbit simulation and visualization examples using SAT-LAB are given in Figures 2 to 5. The following types of orbits are presented:

- Geostationary orbit. A geostationary orbit is considered a special case of geosynchronous orbit. A satellite in geostationary orbit has a period of 1 sidereal day, an inclination of 0° (equatorial orbit) and an angular velocity equal to the Earth's angular rate.
- Tundra orbit. Tundra orbit is a geosynchronous, elliptical orbit with an inclination of 63.4°.
- Molniya orbit. Molniya orbit is an elliptical orbit with an inclination of 63.4°, an argument of perigee of 270° and a period of 0.5 sidereal days. A satellite in Molniya orbit spends the majority of its orbit in the northern hemisphere and, therefore, Molniya orbit is suitable for satellite observations over North America, Russia and northern Europe.
- Low Earth orbit. Low Earth orbit is any satellite orbit with an altitude of 200-2000 km. Earth observation satellites, such as altimetry and gravimetry satellites, are orbiting in low Earth orbits.

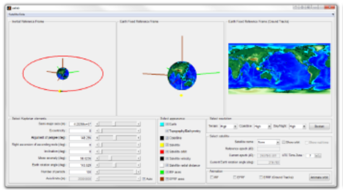


Figure 2 – Geostationary orbit

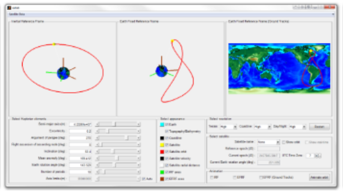


Figure 3 – Tundra orbit

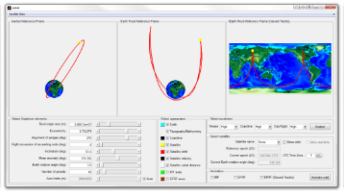


Figure 4 – Molniya orbit

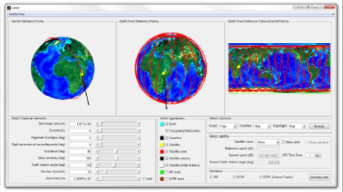


Figure 5 – Low Earth orbit

ORBIT VISUALIZATION OF OPERATIONAL SATELLITES

SAT-LAB provides the ability to download Keplerian elements and other information of operational satellites in the form of TLE ephemerides from https://celestrak.com/. The user can choose among the 41 satellite categories presented in Figure 6. Figure 9 provides the TLE information for GRACE-1 satellite. Here, the orbit of the following satellites is chosen to be visualized using SAT-LAB:

- LAGEOS 1 (Figure 7). LAGEOS is a satellite mission dedicated to providing geodetic and geodynamic data using the method of satellite laser ranging. The LAGEOS 1 satellite is a sphere of 60 cm diameter with 426 reflectors on its surface. LAGEOS 1 was lunched in 1976 and has an orbital altitude of 5860 km and an inclination of 109.84°.
- BEIDOU IGSO 2 (Figure 8). BEIDOU is the navigation satellite system of China, currently consisting of 21 operational satellites. The BEIDOU IGSO 2 satellite was launched in 2010 and operates at an altitude of 35700 km with an inclination of 55°.
- GPS BIIF-1 (Figure 10). GPS is a navigation satellite system operated by the U.S Department of Defense and currently consists of 31 operational satellites. The GPS BIIF-1 (PRN 25) satellite was launched in 2010 and has an altitude of 20000 km, an inclination of 55° and a period of approximately 12 hours.
- GRACE-1 (Figure 11). The GRACE satellite mission is designed to observe the temporal variations of the Earth's gravity field. GRACE-1 was launched in 2002 in a low Earth, circular and near-polar orbit.

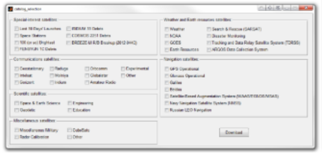


Figure 6 – Satellite catalogs

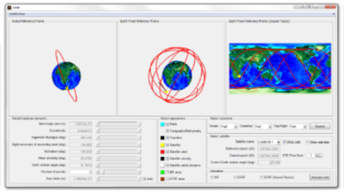


Figure 7 – Visualization of LAGEOS 1 satellite orbit

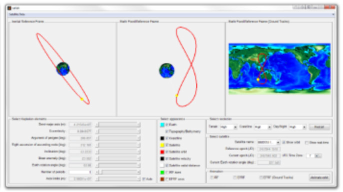


Figure 8 – Visualization of BEIDOU IGSO 2 satellite orbit

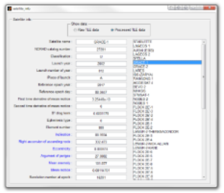


Figure 9 – Satellite information

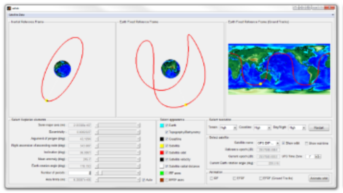


Figure 10 – Visualization of GPS BIIF-1 satellite orbit

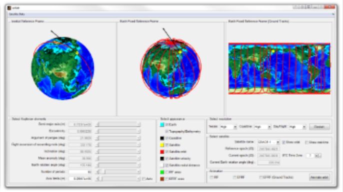


Figure 11 – Visualization of GRACE-1 satellite orbit

REAL-TIME SATELLITE TRACKING

The capability of real-time tracking of operational satellites is also implemented in SAT-LAB. Accurate real-time satellite tracking is possible only when the correct date/time settings and UTC time zone are selected by the user, depending on their location. A day and night map is also calculated and superimposed only for real-time satellite tracking. Examples of real-time tracking of GRACE-1 and International Space Station (ISS) satellites are given in Figures 12 and 13, respectively. The satellite position can be visually compared with results obtained from http://www.heavens-above.com/ , for both cases.

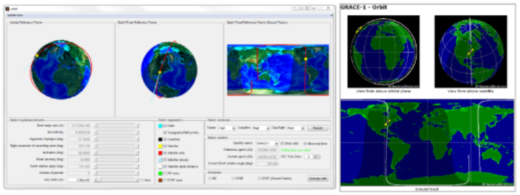


Figure 12 – Real-time tracking of GRACE-1 by SAT-LAB (left) and heavens-above.com (right)

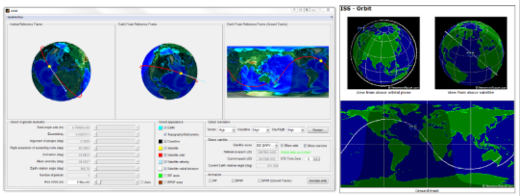


Figure 13 – Real-time tracking of ISS by SAT-LAB (left) and heavens-above.com (right)

SUMMARY

- The SAT-LAB software provides a friendly and easy-to-use GUI for orbit simulation, orbit visualization and real-time tracking of operational satellites.
- SAT-LAB is developed using MATLAB R2012 and tested in later releases (i.e., R2013a) in both 32- and 64-bit operating systems. Overall, SAT-LAB consists of 20 .m files (17 functions and 3 script files).
- The current version of SAT-LAB is suitable for educational purposes and low-accuracy orbit simulations only.
- More rigorous approaches to satellite orbit simulation and, possibly, more visualization options will be implemented in future releases of SAT-LAB.
- SAT-LAB is freely available for download at http://www.dimitriospiretzidis.com/satlab_home.html .

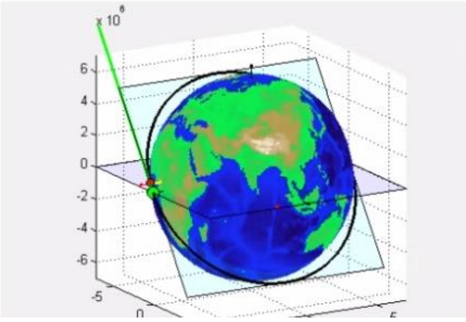
REFERENCES

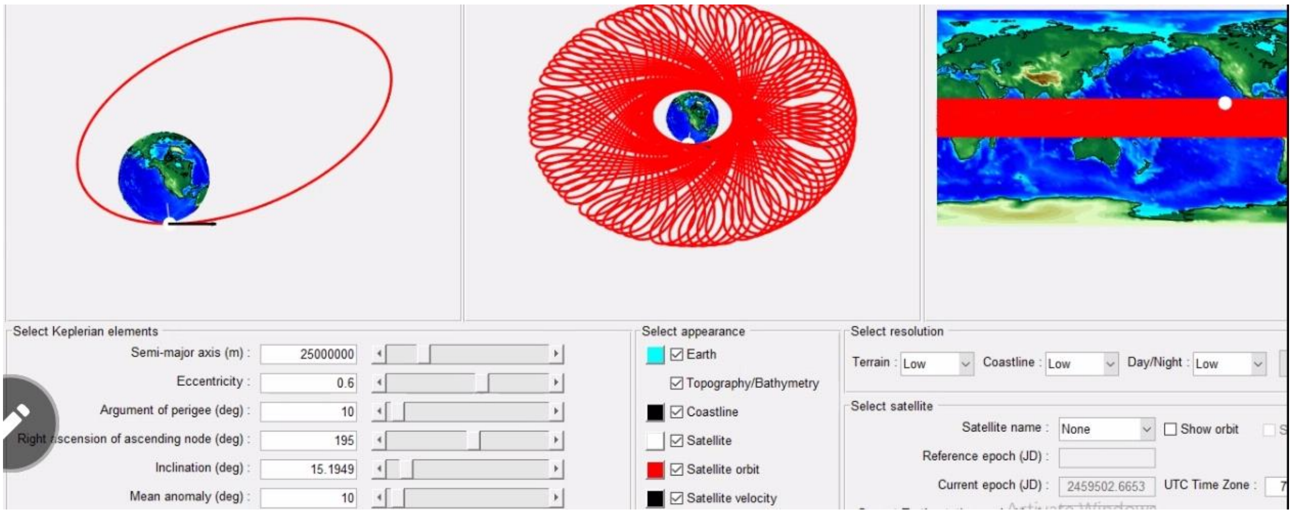
Heavens-Above. http://heavens-above.com/. Accessed 3 Apr 2017a
CelesTrak. https://celestrak.com/. Accessed 3 Apr 2017b

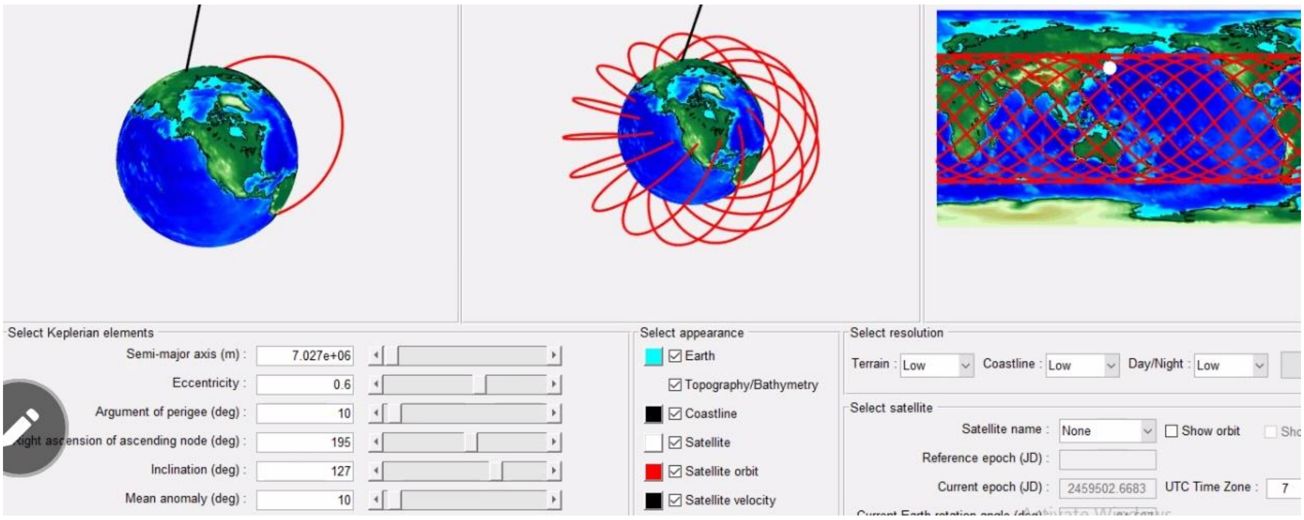
EGU General Assembly 2017, Vienna, Austria, 23 – 28 April 2017

Considering Black Sea area, polar sun-synchronous orbits with 127° inclination at altitude 650 km and semimajor axis of 7027.748766 km with a revolution period of approximately 97.72 min is by far accepted to handle AIS data from this area.

Below, is a graph with five periods of the calculated orbital elements







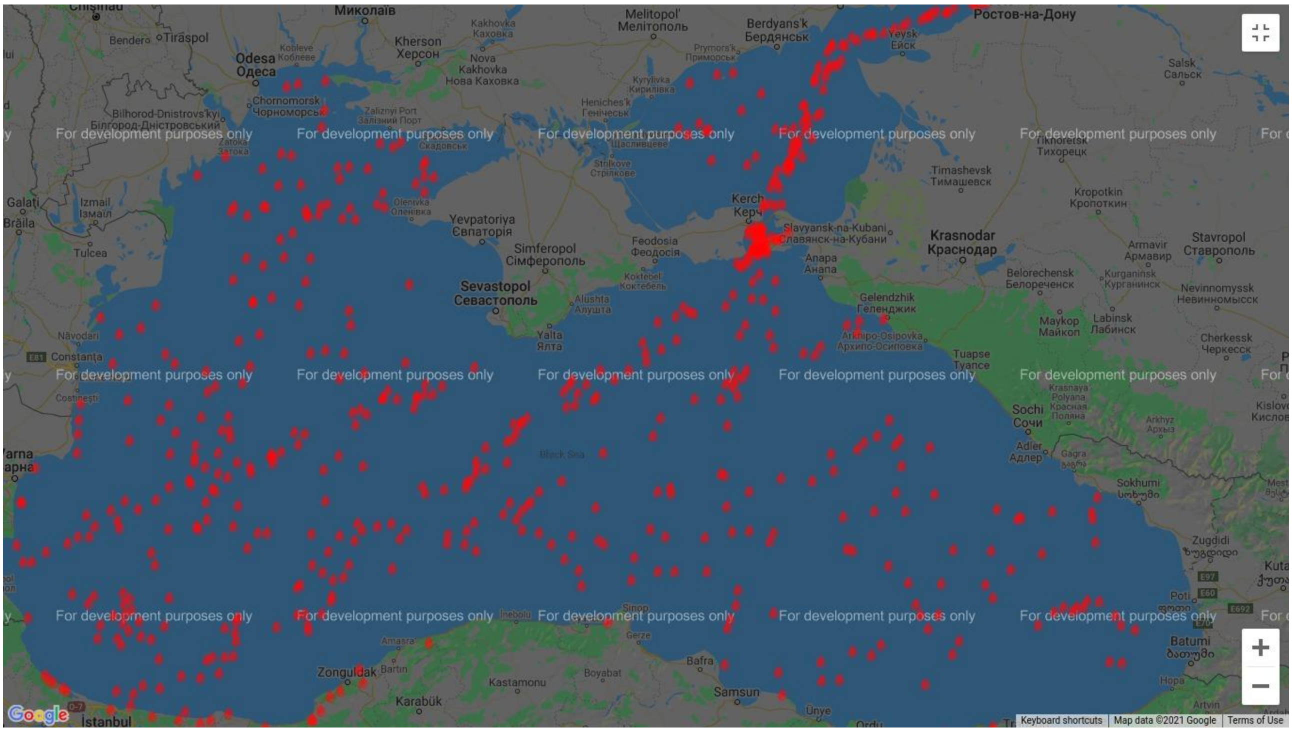
8.1.2 Mission purpose and Payload

8.1.2.1 X-Ray sensor

[SRQ 20] There shall be an astronomical x-ray sensor on board.

8.1.2.2 Satellite based AIS system

[SRQ 30] There shall be a AIS transponder system on board for coordinating AIS signals from ships in the black sea.



Tasks:

- Satellite Footprint investigation to know how many visible satellites we need for AIS for black sea

Result: Requirements for satellite system:

- Orbit height: 500-600 km (over 600 km we get problems with AIS signals)
- Sun-synchronous orbit (restriction from launcher)
- Inclination: ?

For AIS application:

- 650 km altitude, elliptic semimajor axis a = 7027.748 km
- 10 minutes per satellite visibility - > 10 satellites
- - > 10 satellites
- - inclination: 100° - 127°
-

8.1.3 ACS

[SRQ60] Shall be low cost. For this reason only magnetorquer and sun-sensor is shall be used.

8.1.4 TT&C System

[SRQ10] TT&C shall be realized with the same hardware card as payload COM

OBC -> TT&C/COM transceiver (SDR) -> flat antenna

		
On-Board Computer (RaspberryPi)	HackRF SDR Card	Antenna

8.2 System Design Document (SDD)

8.2.1 On-Board Telemetry, Tracking & Control (TT&C)

8.2.1.1 Overview - Bottum-Up Approach

[SRQ10] TT&C shall be realized with LimeSDR Mini.

8.2.1.2 Transceiver Card

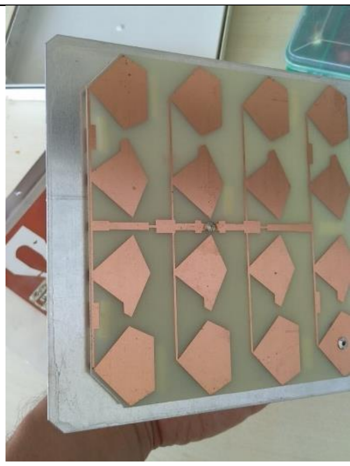
[SRQ50] The Transceiver Card has to be maximum 10 cm x 10 cm x 1 cm.

8.2.2 Telemetry, Tracking & Control (TT&C) Ground Station

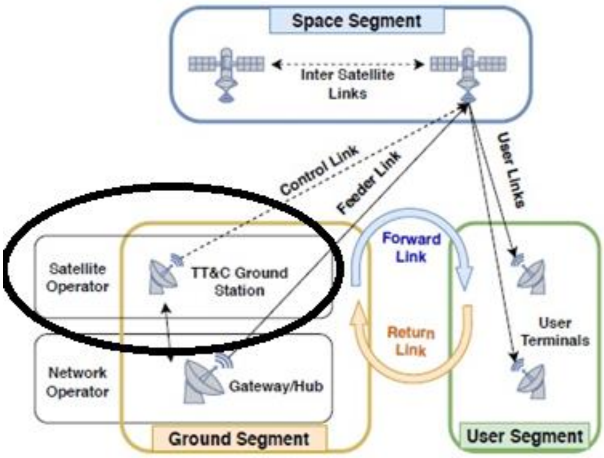
8.2.2.1 Bottum-Up Approach

At ground station the off-the-shell HackRF is suitable.

OBC -> TT&C/COM transceiver (SDR) -> flat antenna

		
On-Board Computer (RaspberryPi)	HackRF SDR Card	Antenna

8.2.2.2 Requirements (A DESCRIPTION OF A STANDARD SMALL SATELLITE GROUNDSTATION FOR USE BY WMO MEMBERS [4])



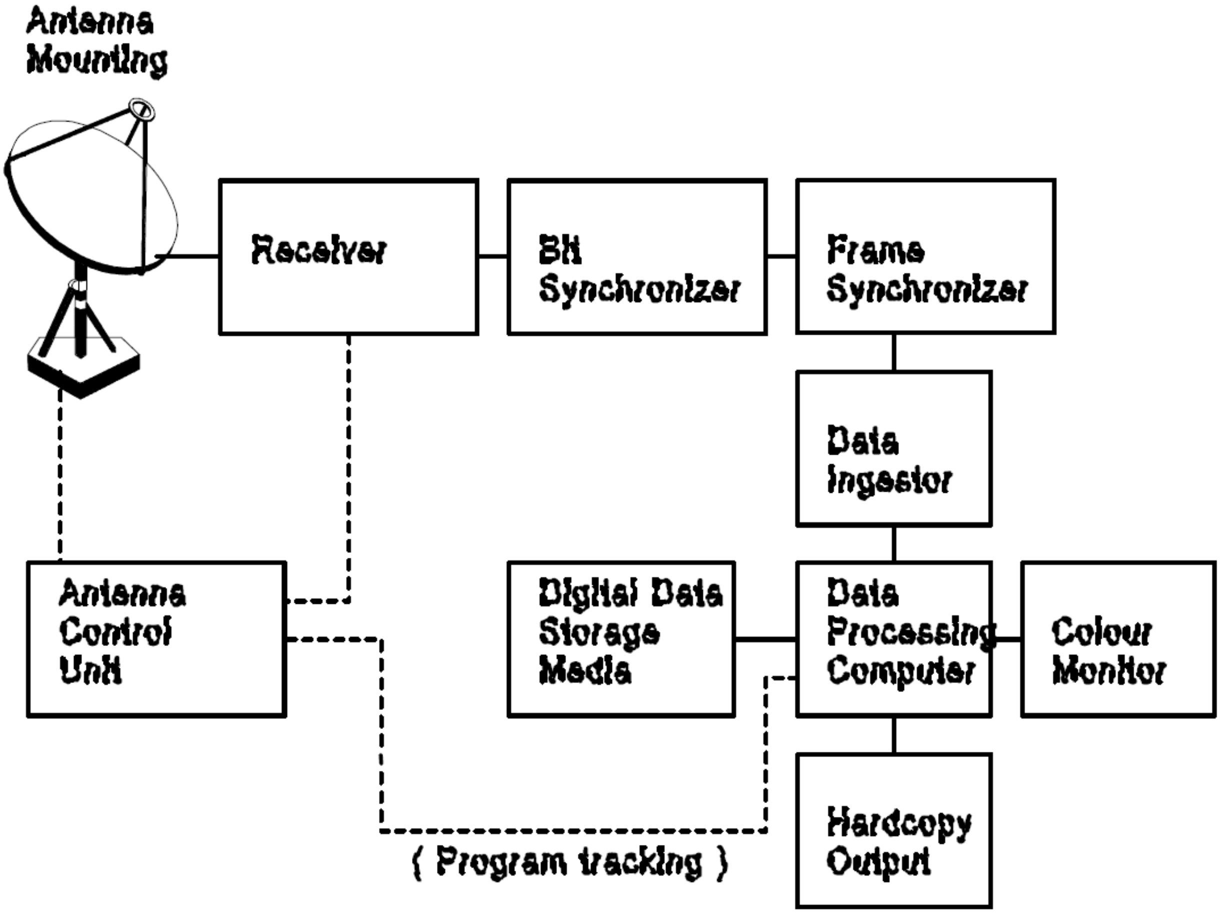


Fig.1 Block Diagram of a small satellite ground station

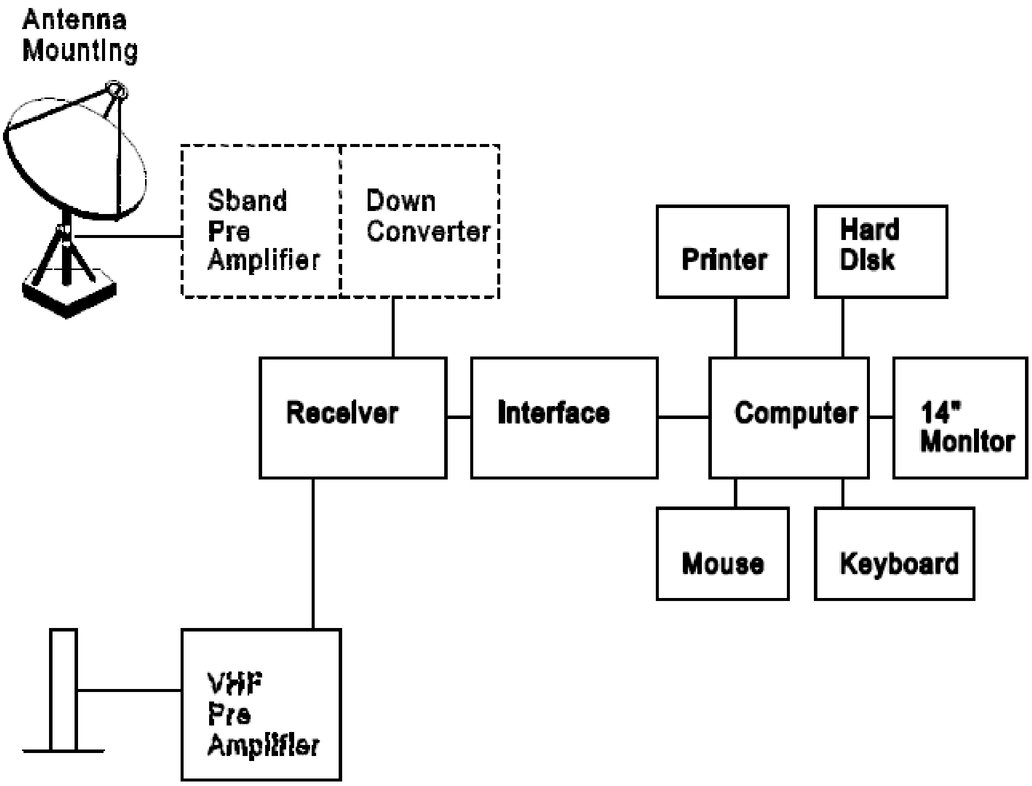


Fig.2 APT/WEFAX Receiving and Processing Station

5.1.1 S-band parabolic antenna

5.1.1.1	Diameter of antenna:	: 1.5 m
5.1.1.2	Gain	: 26.0 db
5.1.1.3	Beamwidth	: 8.1°
5.1.1.4	Frequency	: 1691.0 MHz
5.1.1.5	Polarization	: linear
5.1.1.6	Mount	:fixed, variable adjust, elevation 90°, azimuth (20° - 70°)
5.1.1.7	VSWR	: 1.5
5.1.1.8	Impedance	: 50 Ω

5.1.2 S-band preamplifier and down converter

5.1.2.1	RF input	: 1691.0 MHz
5.1.2.2	IF output	: 137.5 MHz or other
5.1.2.3	Band width	: 6 MHz
5.1.2.4	IF/RF gain	: 30 db
5.1.2.5	Noise figure	: 1.5 db
5.1.2.6	Stability	: 5×10^{-6}
5.1.2.7	Impedance	: 50 Ω
5.1.2.8	Cable length	: 60 m

5.1.3 VHF antenna

To reduce the price and for ease of maintenance, an OMNI directional non-tracking antenna is recommended. An OMNI directional non-tracking antenna must be able to receive data above an elevation of 5°. This requirement will reduce interference while maximizing the possibility for coverage of synoptic scale meteorological phenomena.

OMNI directional antenna

5.1.3.1	Frequency	: 137.5 MHz
5.1.3.2	Polarization	: right hand circular
5.1.3.3	Impedance	: 50 Ω
5.1.3.4	VSWR	: 2.1 max
5.1.3.5	Gain	: 3 dbi
5.1.3.6	Beamwidth	: 180°

Depending on the user's situation and requirements, an omni-directional antenna may not be sufficient for proper APT reception. Under these circumstances, the use of a directional antenna, such as a crossed Yagi, would provide higher performance and greater coverage. Note that use of program tracking and other antenna pointing methods would be required. The following information describes an alternative to the OMNI directional antenna described in sections 5.1.3.1 through 5.1.3.6.

Directional antenna (Yagi)

5.1.3.7	Centre frequency	: 137.5 MHz
5.1.3.8	Polarization	: right hand circular
5.1.3.9	VSWR	: 2.0 max
5.1.3.10	Gain	: 20 dBi or greater
5.1.3.11	Beamwidth	: 20 degrees at 20 dBi
5.1.3.12	Mount	: Elevation over azimuth
5.1.3.13	Program track	: Program track

5.1.4 VHF Preamplifier

5.1.4.1	Centre frequency	: 137.5 MHz
5.1.4.2	Gain	: 30 db
5.1.4.3	Noise figure	: 2 db
5.1.4.4	Installation	: in antenna base

5.1.5 Receiver

5.1.5.1	Type	: FM phase lock loop
5.1.5.2	Input frequency	:Switch selectable crystals for reception of APT and WEFAX
5.1.5.3	IF bandwidth	:50 KHz and 30 KHz --switch selectable
5.1.5.4	Noise figure	: 5 db

5.1.6 Computer

5.1.6.1	486 computer with TVGA card (1024 x 768 resolution)	
5.1.6.2	Colour monitor	:14"
5.1.6.3	Hard disk	: 120 MB
5.1.6.4	Memory	: 4 MB
5.1.6.5	Clock : 33 MHz	
5.1.6.6	Keyboard	
5.1.6.7	Mouse	
5.1.6.8*	Printer (optional)	

5.1.7 Outdoor environment

5.1.7.1	Temperature	: -40° C - +50° C
5.1.7.2	Humidity	: 98%
5.1.7.3	Wind	: operational 20 m/s, survival 35 m/s

5.1.8 Power

5.1.8.1	110v/220v +- 10%
5.1.8.2	50Hz/60Hz

5.2 Specification requirement, for high resolution ground station

The stations should be reliable and easy to operate. The block diagram of the high resolution ground station is shown in Fig.3. If the station receives geostationary satellite data, it should be equipped with a fixed antenna. If the station receives polar orbiting satellite data then it must be equipped with a tracking antenna and antenna control unit.

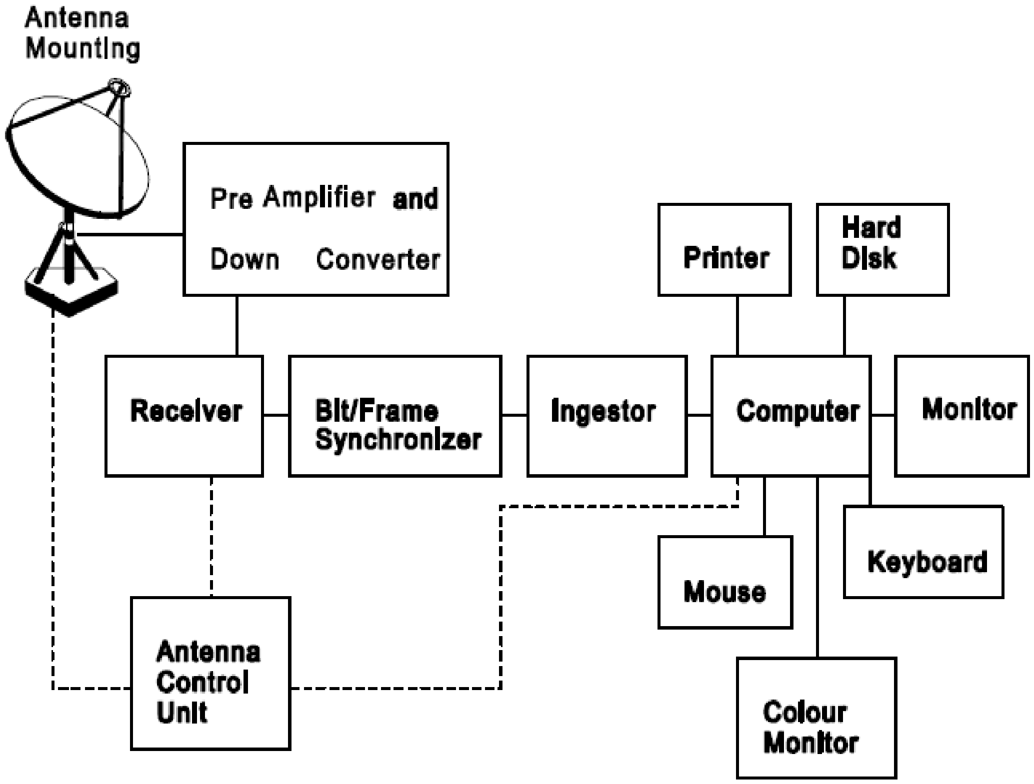


Fig.3 High Resolution Data Receiving and Processing Station

5.2.1	**S-band tracking antenna and antenna control unit for receiving HRPT data**	
5.2.1.1	G/T merit of antenna system	:>6 db/K bit error rate is better than $1x10^{-6}$ at 5 degree elevation
5.2.1.2	Frequency	:1670~1710 MHz
5.2.1.3	Polarization	: RH, LH
5.2.1.4	Impedance	: 50/75 Ω
5.2.1.5	VSWR	: 1.5
5.2.1.6	Antenna mounting	: azimuth-elevation
5.2.1.7	Tracking coverage	: full geometric coverage including overhead passes (Z-pass), good performance of Z-pass tracking
5.2.2	**S-band fixed antenna for receiving geostationary satellite**	
5.2.2.1	G/T merit of antenna system	:G/T depending on which satellite to be received, G/T must guarantee that bit error rate is less then 10^{-6} in the worst case
5.2.2.2	Frequency	: 1670~1710 MHz
5.2.2.3	Polarization	: linear
5.2.2.4	Mounting	: fixed, variable adjust elevation 90, azimuth 20° ~70°
5.2.2.5	VSWR	: 1.5
5.2.2.6	Impedance	: 50/75 Ω

8.2.2.3 As reference a ground station design from 2007 ([3]):

Abstract of [3] :

The CubeSat satellite ground station at the University of Wuerzburg is built with "commercial of the shelf" low cost amateur radio hardware. It opens up opportunities for students to receive and operate CubeSats, including Wuerzburgs UWE-1. As any other satellite ground station, it is built up on essential hardware, as there are Antenna, Antenna Rotator, Radio, Modem and Computers. Furthermore software is used to afford basic control over the ground station and provide tracking abilities to follow a satellite passing over the ground station.

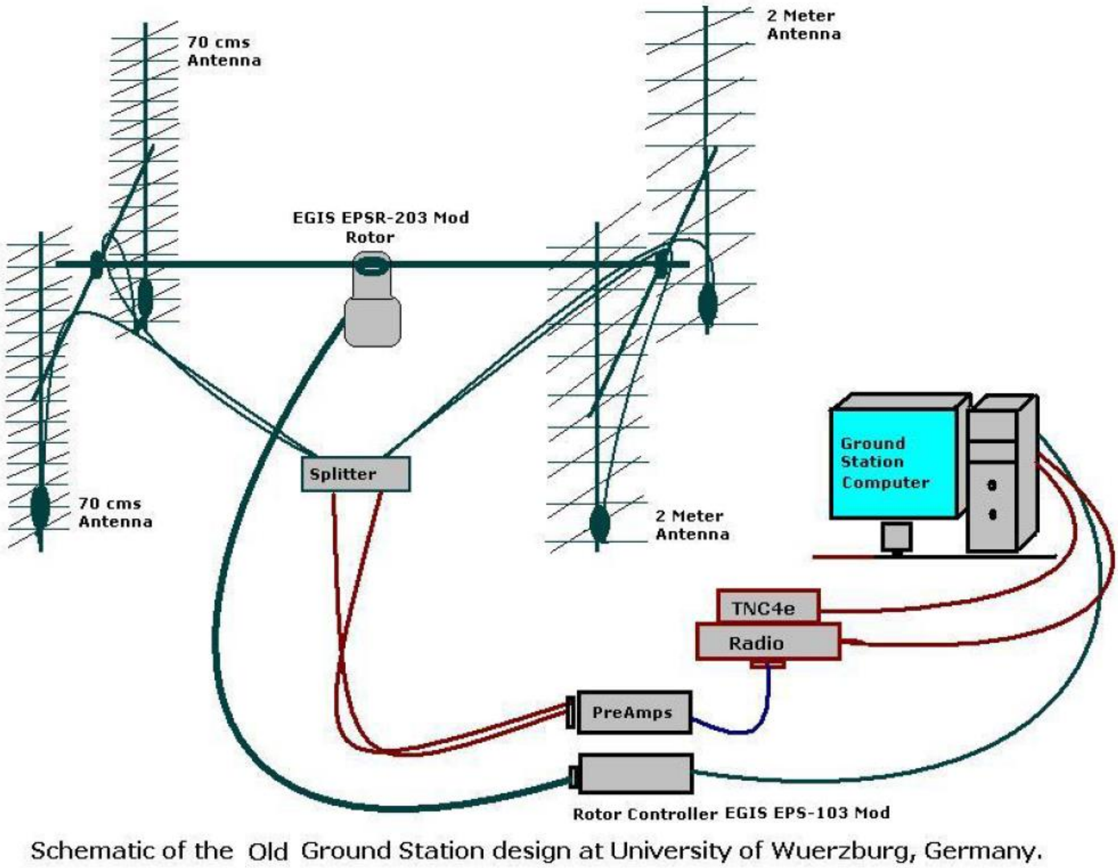
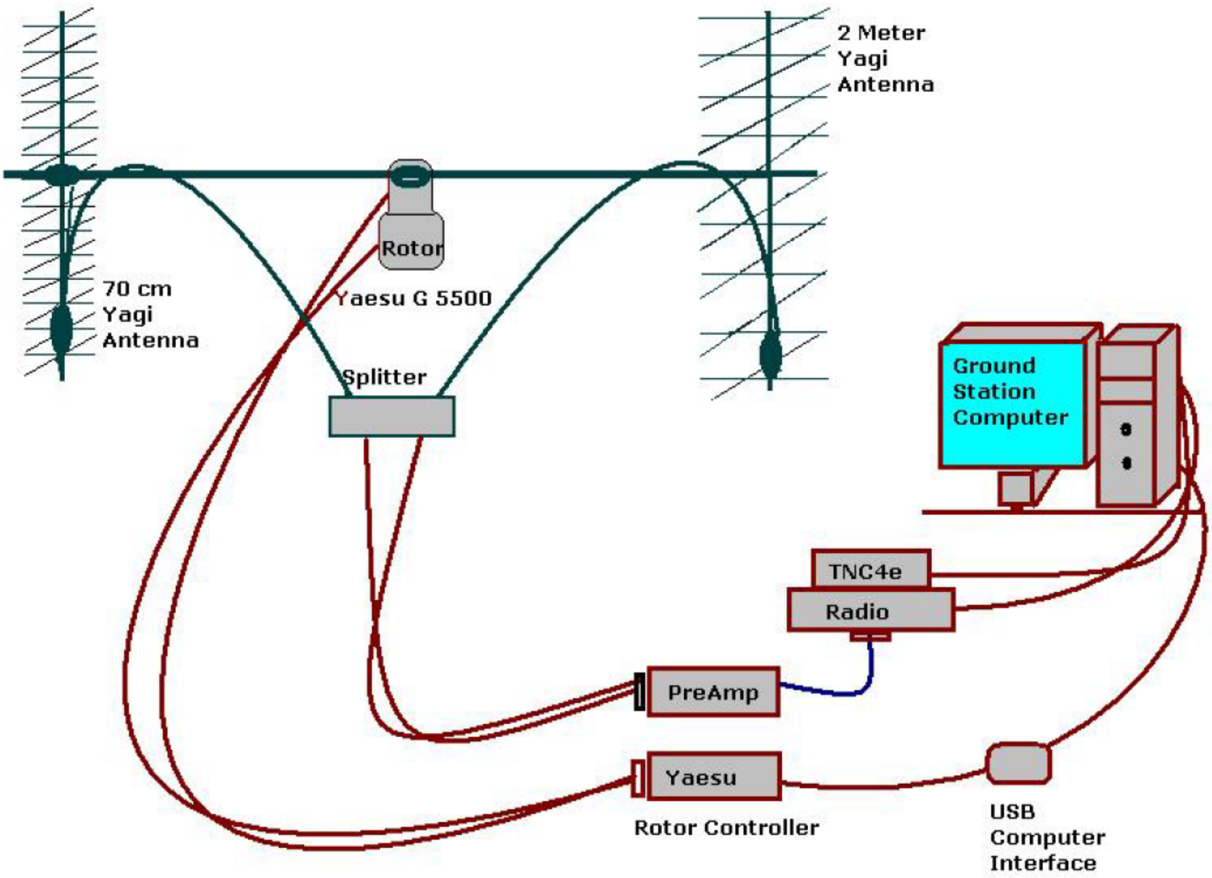


Figure 1.1: Old design of the Ground Station.

Ground Station structural study

Related CubeSat.	***University.***	***Antenna Rotor.***	***Rotor Computer Controller & Tracking Software.***
Delfi-C3	Delft University of Technology	Yaesu G-5500	NOVA for Windows
AAU Cubesat	Aalborg University of Technology	Yaesu G-5500	Predict
CubeSat	University of Arizona	Yaesu G-5500	NOVA for Windows with Uni_Trac
PolySat	California Polytechnic State University, U.S.	Yaesu G-5500	SatPC32
Cubesat	University of Tokyo, Japan	Elevation Rotator ERC5A (Creative Design) Azimuth Rotator RC5A-3 (Creative Design)	Orbital calculation software (Virtual Ground Station 3)
Cubesat	TU-Berlin	Yaesu G-5500	SatPC32 and ARSWIN
---	Kagawa University, Japan	ERC-5A (El) And RC5B-3P (Az)	Satellite Tracker is RAC825
---	Kyusyu University Ground Station.	EMOTATOR EV-800D	No avail info
---	Kyushu Institute of Technology.	Yaesu G-5500	GS-232A
---	Nara National College of Tech, Japan	Yaesu G-5500	GS-232A and Nova for Windows.
---	Soka University Ground Station Unit.	Yaesu G-550 (El) And Yaesu G-2300DXA (Az)	GS-232B

Figure 1.2: Ground Station Structural Study.



Schematic of the new Ground Station design at University of Wuerzburg, Germany.

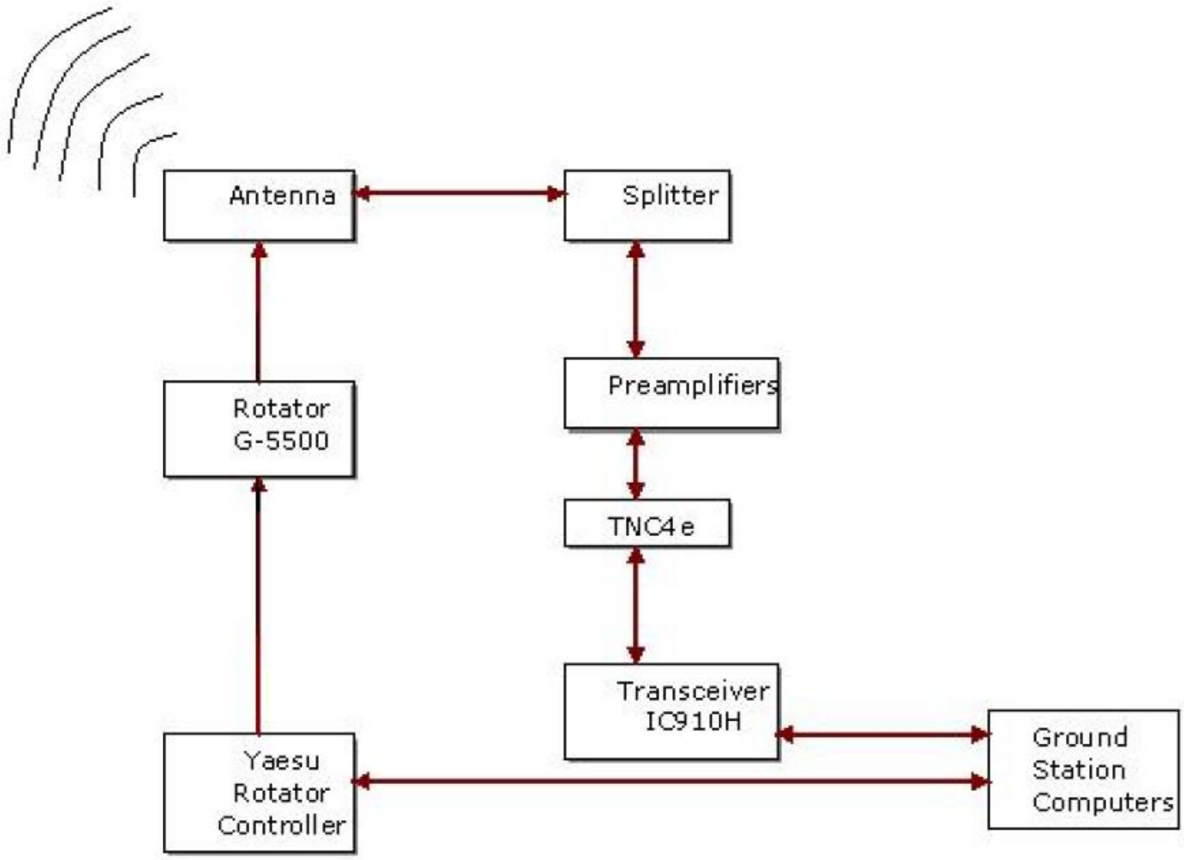
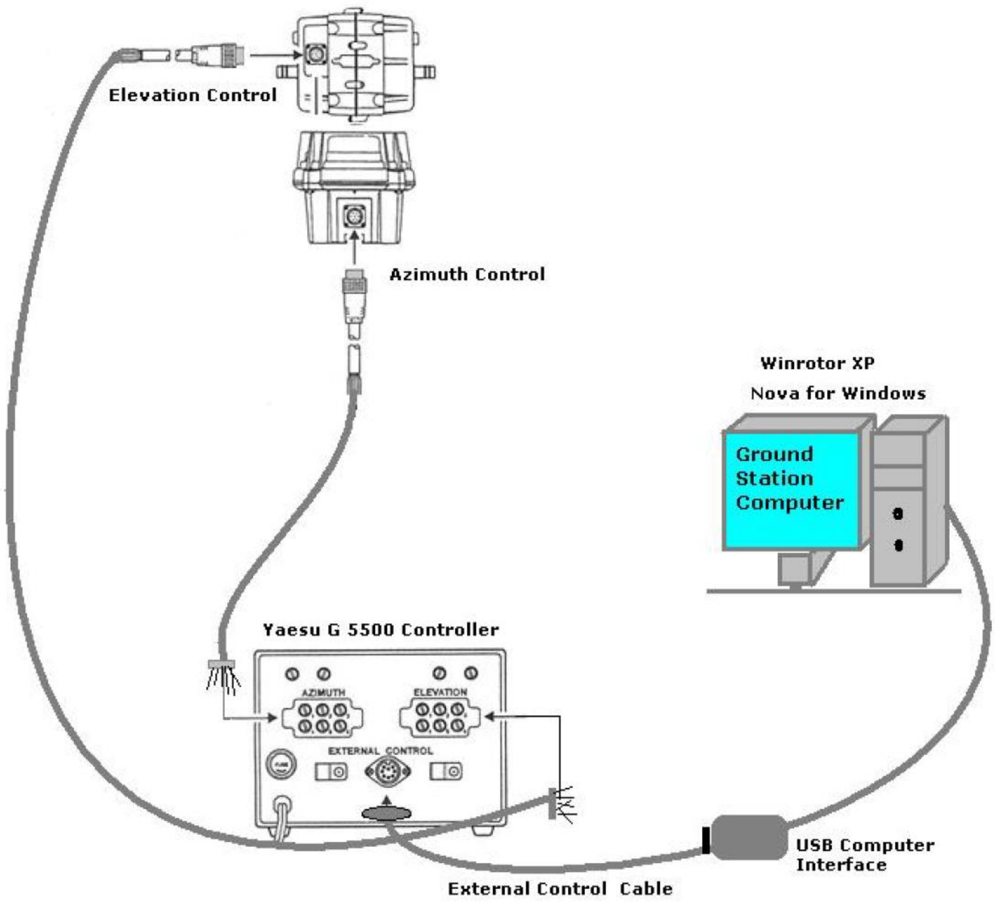


Figure 1.4: Block Diagram of the Ground Station.

Altitude	310 meters
Tower	Hummel Teletower Jumbo III
Operating frequencies	2m and 70 cms amateur bands
One 2 meter antenna	M2 2MCP22
One 70 cms antenna	M2 436CP42U/G
Antenna rotator and Controller	Yaesu G-5500
Rotator-Computer Interface	WinRotor
Rotator-Computer Interface driver	WinRotor XP
Radio	TNC4e
Polarisation switch	WiMO
Preamplifiers	LNA-145, SLN Series
Tranceiver	ICOM IC-910H
Power Supply	Microset 13.5 Volts
Two PCs	Fujitsu Siemens

Table 2.1: Hardware Specifications Table.



Rotor-Controller-Computer Interfacing.

8.2.2.4 Graphical User Interface

Installing Nova for Windows

(a) Insert the Nova for Windows CD into the CD-ROM drive of your computer.

(b) If the setup program doesnt start automatically, click on the Start button (lower left corner of the desktop).

(c) Click on Run.

(d) In the file name box, type **Setup.EXE**.

(e) Follow the directions in the Nova for Windows Setup.

Important:

Be sure to enter the serial number carefully. Serial number must include the NLD- prefix.

First step is to set the type of Map. In the screenshots shown below "Large Rectangular Map" is selected for convenience.

To choose the new map setting the path is-
"Views" then "Configure current view" and then Choose "Map display" and "Map Size".
Refer figure 2.11.

Second step is to set the position of the Ground Station in "Nova for Windows". The path is- "Setup" and then "Observers".

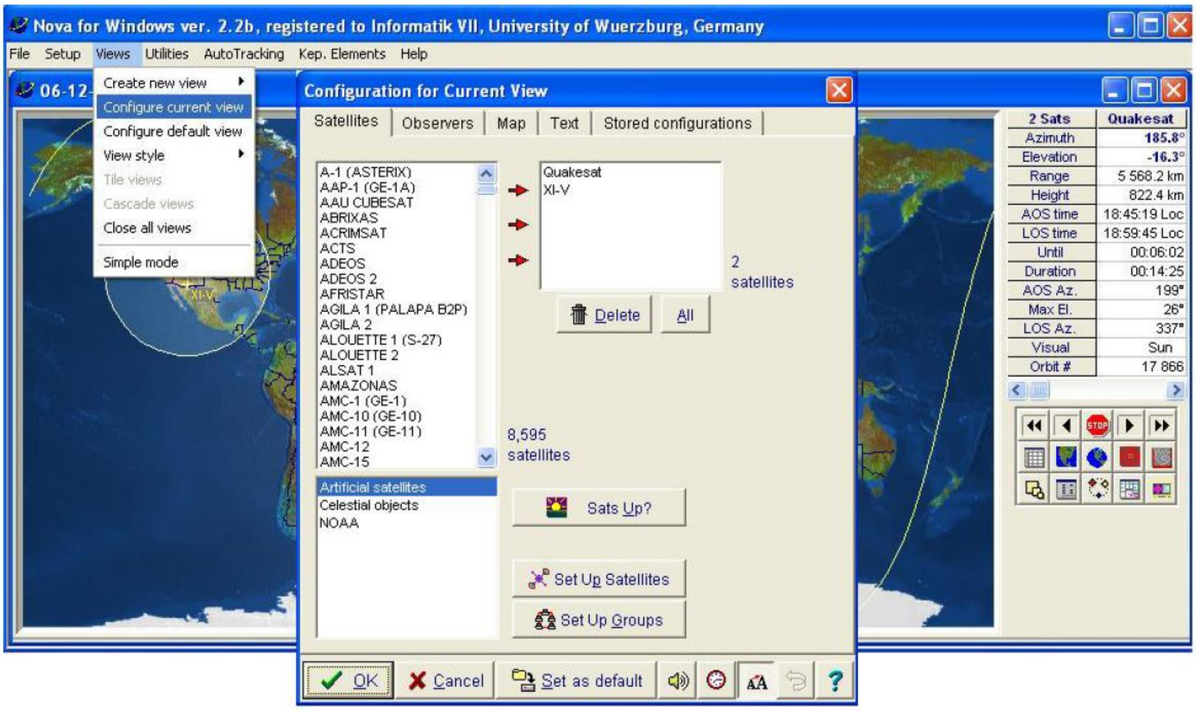


Figure B.1: Nova for Windwos - Configuring View.

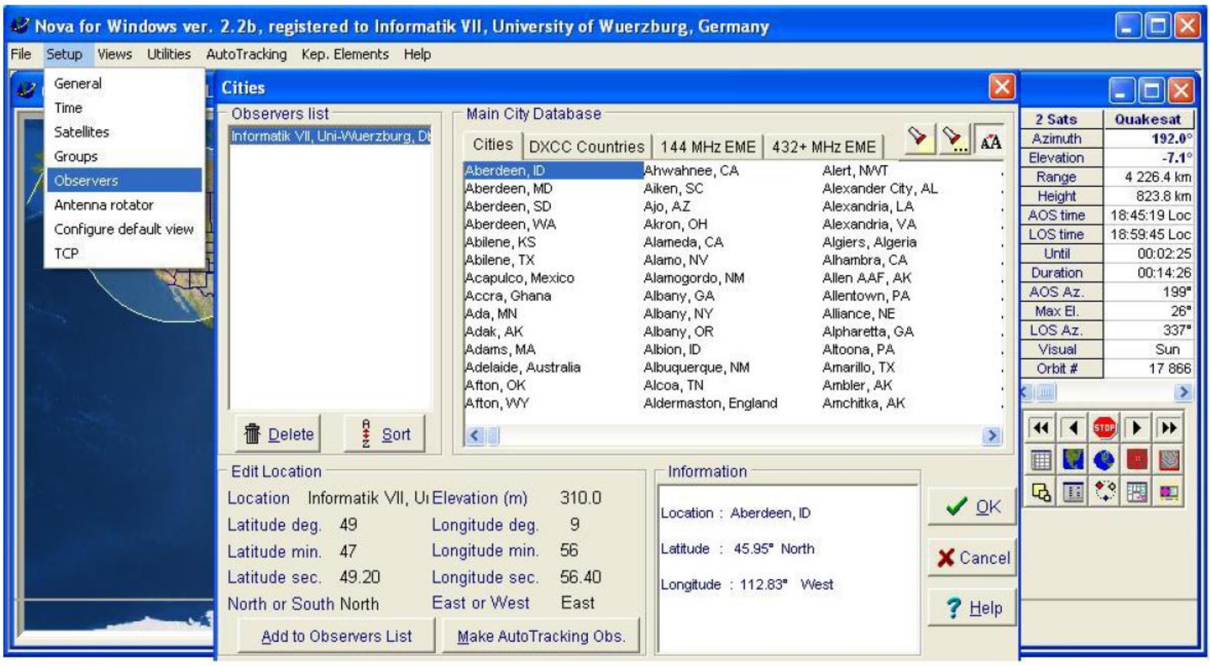


Figure B.2: Nova for Windows - Configuring Observer.

In our case it is:
Location: *"Informatics VII, Uni-Wuerzburg, Germany."*
Elevation is of *310 meters.*
Latitude is *49 degrees 47 minutes 49.20 seconds North.*

Longitude is *9 degrees 56 minutes 56.40 seconds East.*
Refer figure 2.12.

Third step is to check the availability of the specific satellite from the Satellite Editor in the database of "Nova for Windows".
In this editor, new satellite names and its Keplerian elements can also be added. Also "Update Keplerian Elements" button provides the online update.
The path is "Setup" and then "Satellites".
Refer figure 2.13.

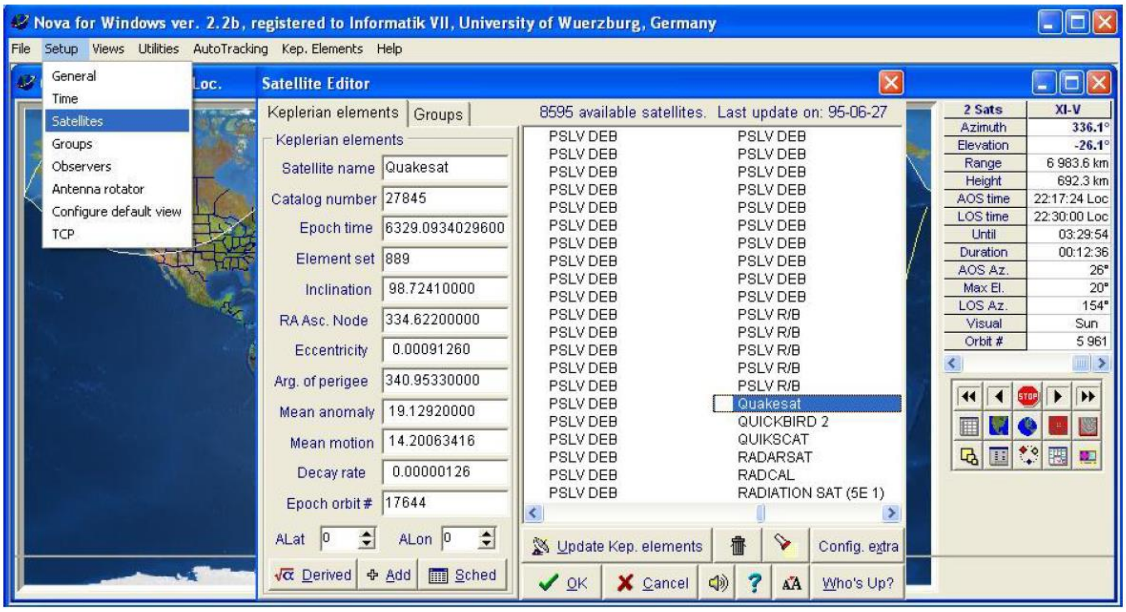


Figure B.3: Nova for Windows - Configuring Satellites.

To update Keplerian elements or to get related help click on "Kep Elements". Refer figure 2.14.

Fourth step is to choose the "Current View" in order to see Satellite and Observer (Ground Station position) all together.
This provides a feature of selecting multiple Satellites and Observation points on the map at the same time.
The path is "Views" and then "Configure current view" and then "Satellites" or "Observes" or "Map" or "Text". Refer figures 2.15 and 2.16.

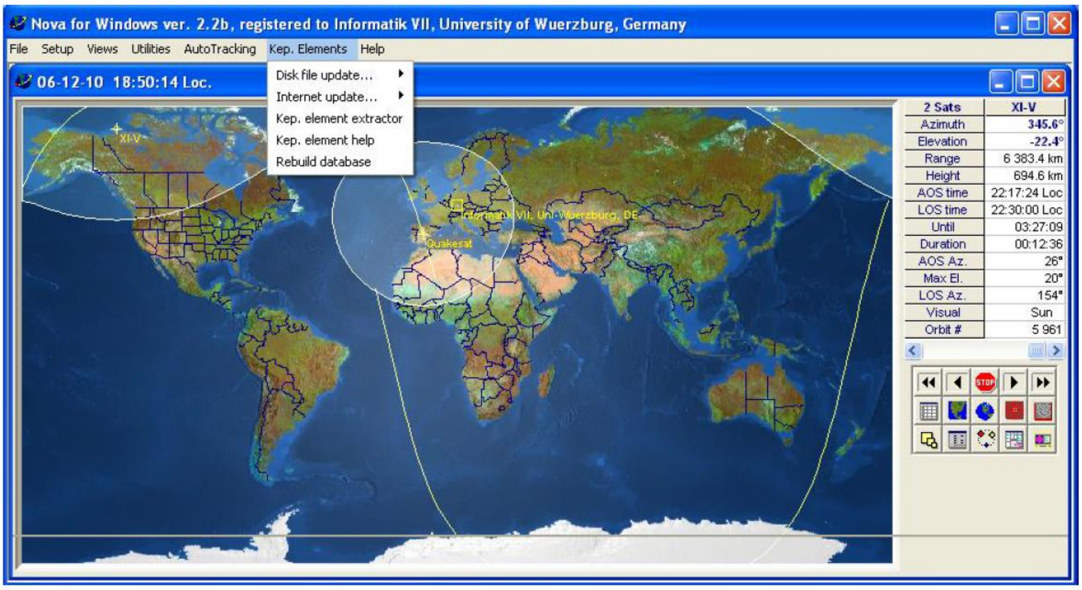


Figure B.4: Nova for Windows - TLE Updation

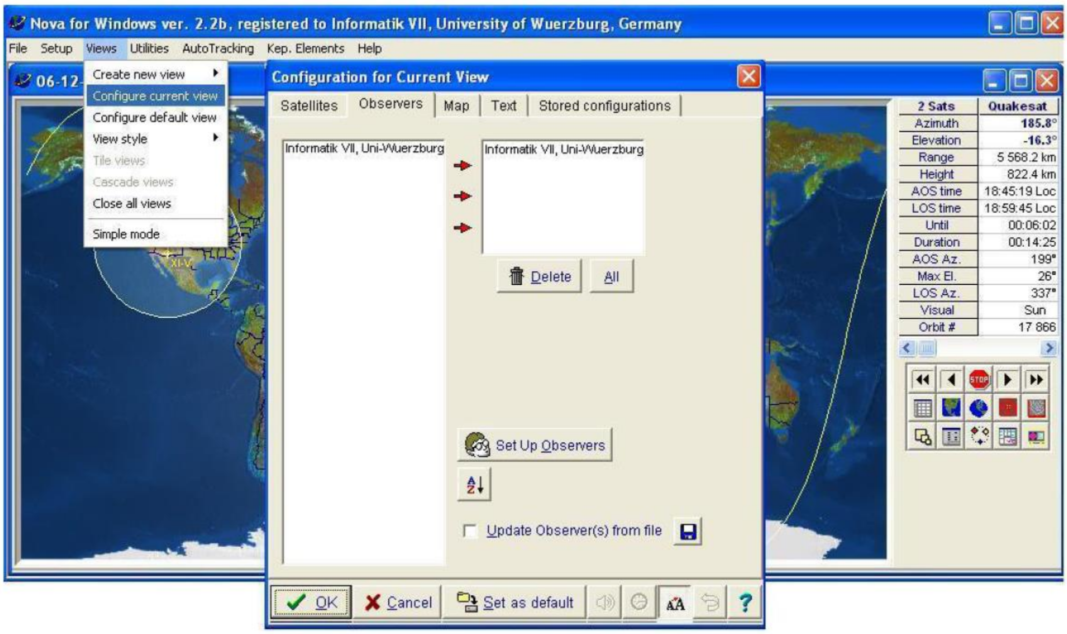


Figure B.5: Nova for Windows - Current View Observer.

On the Map, Footprint of the satellite/s and the Ground Station's position/s can be easily found. Refer figure 2.17.

On the Right hand side of the screen, Real-time text data of the con-

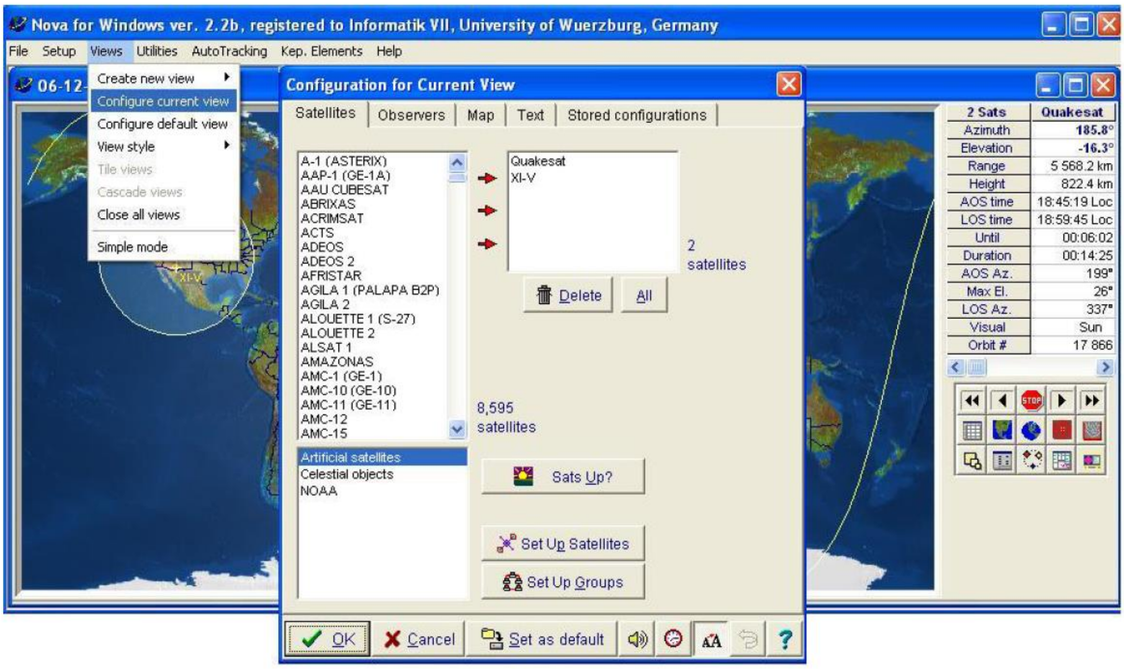


Figure B.6: Nova for Windows - Current View Satellite.

cerning satellite is available. The number of columns in the real-time text window depends on the number of satellites in the view.

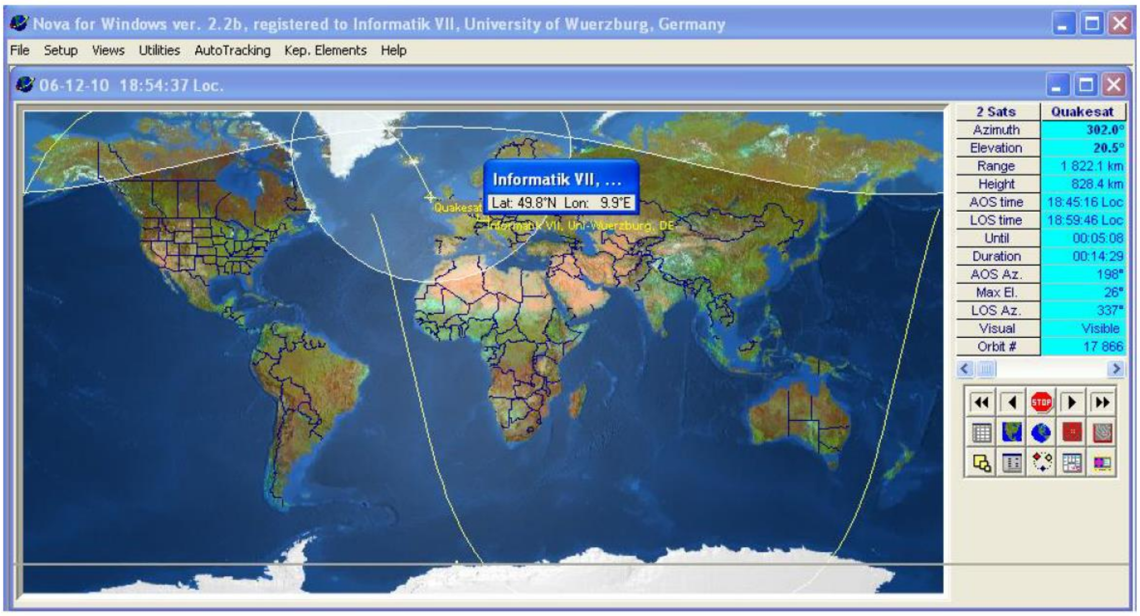


Figure B.7: Nova for Windows - Satellite Footprints.

Satellite Script.

"Satellite Script" features the prediction of the flyby time of the satellite or satellites over a particular Observer (Ground Station) up to 48 hours in advance.
This also enables "automatic script tracking". Refer figure 2.19.

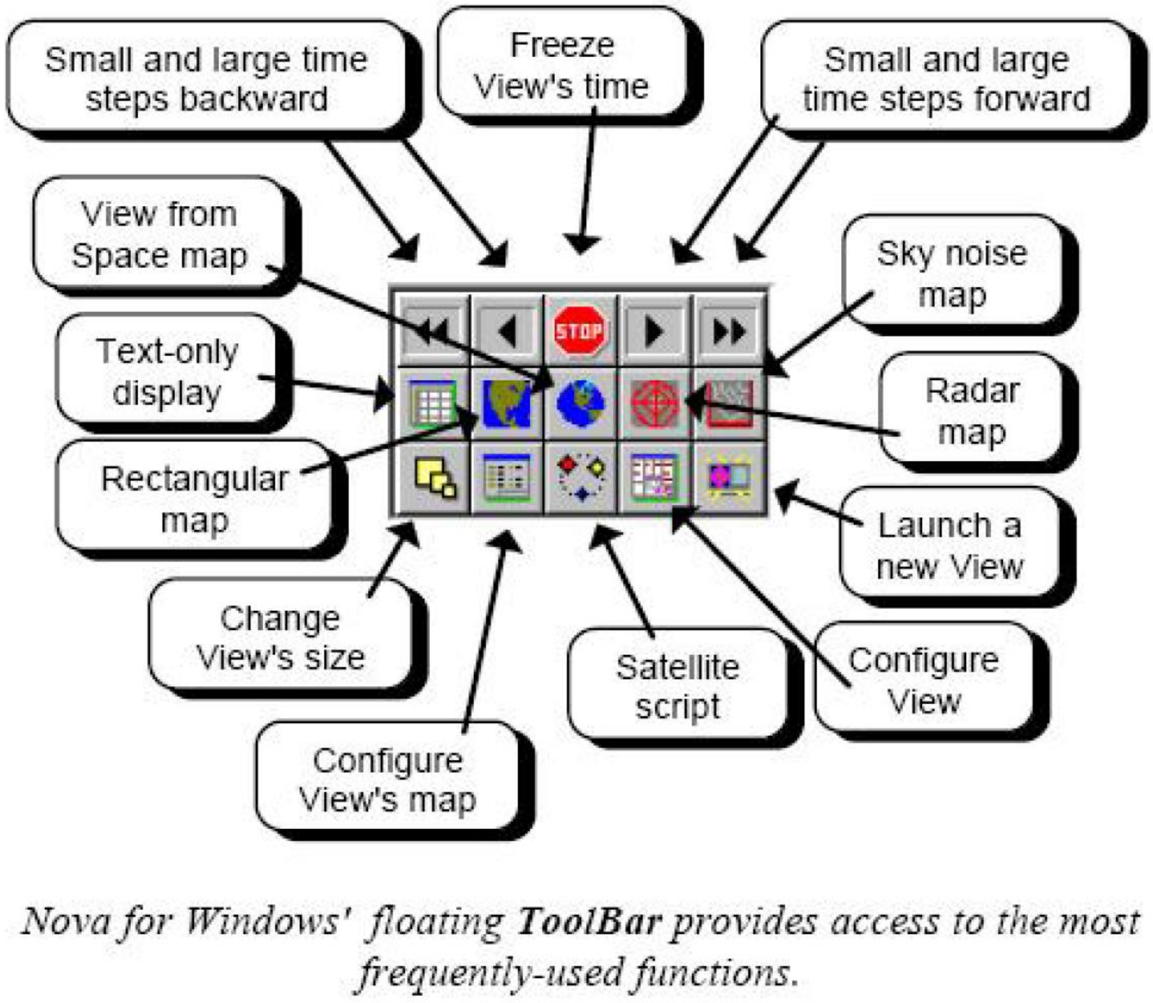
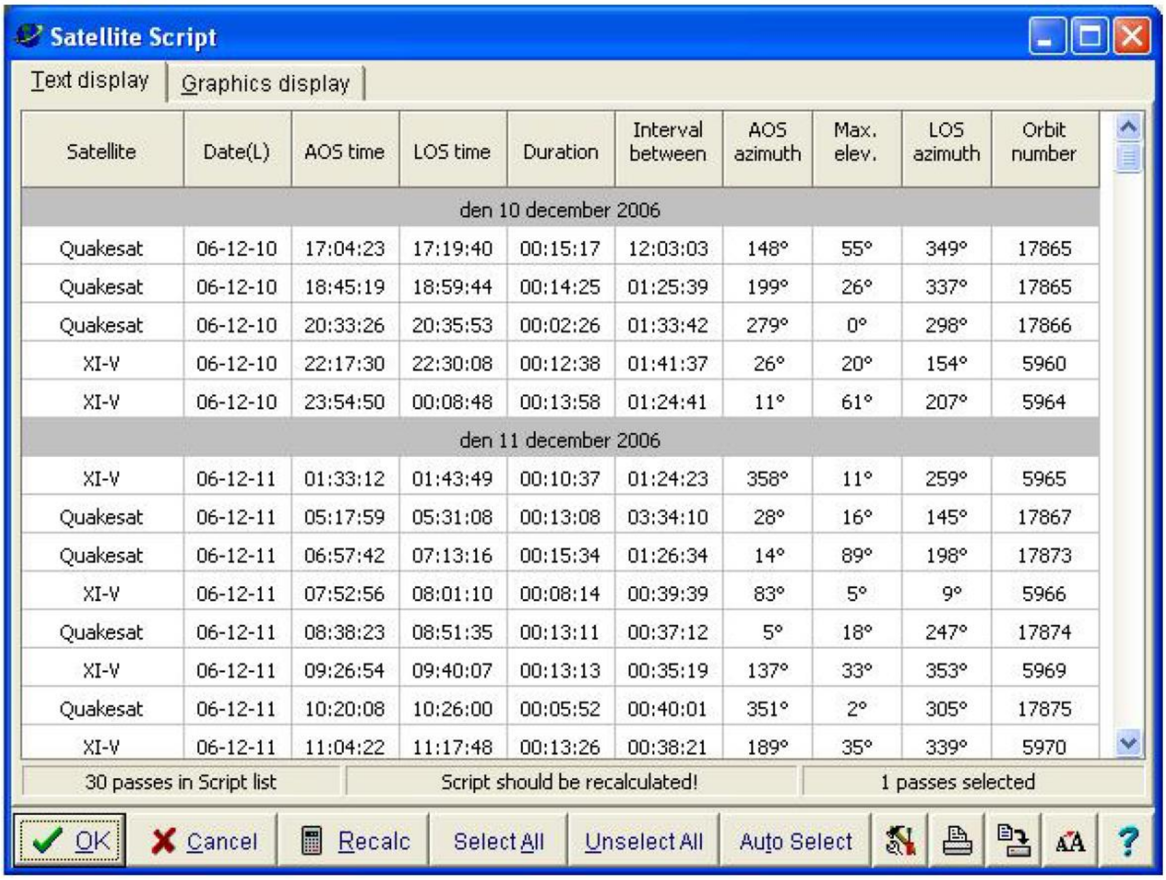


Figure B.8: Nova for Windows - Floating Toolbar.



Satellite	Date(L)	AOS time	LOS time	Duration	Interval between	AOS azimuth	Max. elev.	LOS azimuth	Orbit number
den 10 december 2006									
Quakesat	06-12-10	17:04:23	17:19:40	00:15:17	12:03:03	148°	55°	349°	17865
Quakesat	06-12-10	18:45:19	18:59:44	00:14:25	01:25:39	199°	26°	337°	17865
Quakesat	06-12-10	20:33:26	20:35:53	00:02:26	01:33:42	279°	0°	298°	17866
XI-V	06-12-10	22:17:30	22:30:08	00:12:38	01:41:37	26°	20°	154°	5960
XI-V	06-12-10	23:54:50	00:08:48	00:13:58	01:24:41	11°	61°	207°	5964
den 11 december 2006									
XI-V	06-12-11	01:33:12	01:43:49	00:10:37	01:24:23	358°	11°	259°	5965
Quakesat	06-12-11	05:17:59	05:31:08	00:13:08	03:34:10	28°	16°	145°	17867
Quakesat	06-12-11	06:57:42	07:13:16	00:15:34	01:26:34	14°	89°	198°	17873
XI-V	06-12-11	07:52:56	08:01:10	00:08:14	00:39:39	83°	5°	9°	5966
Quakesat	06-12-11	08:38:23	08:51:35	00:13:11	00:37:12	5°	18°	247°	17874
XI-V	06-12-11	09:26:54	09:40:07	00:13:13	00:35:19	137°	33°	353°	5969
Quakesat	06-12-11	10:20:08	10:26:00	00:05:52	00:40:01	351°	2°	305°	17875
XI-V	06-12-11	11:04:22	11:17:48	00:13:26	00:38:21	189°	35°	339°	5970

Figure B.9: Nova for Windows - Satellite Script.

Frequency display.
It also displays the Uplink and Downlink Frequencies, with the Doppler value for the particular selected satellite.
To check this, the path is "Utilities" and then "Frequency display". Refer figure 2.20.

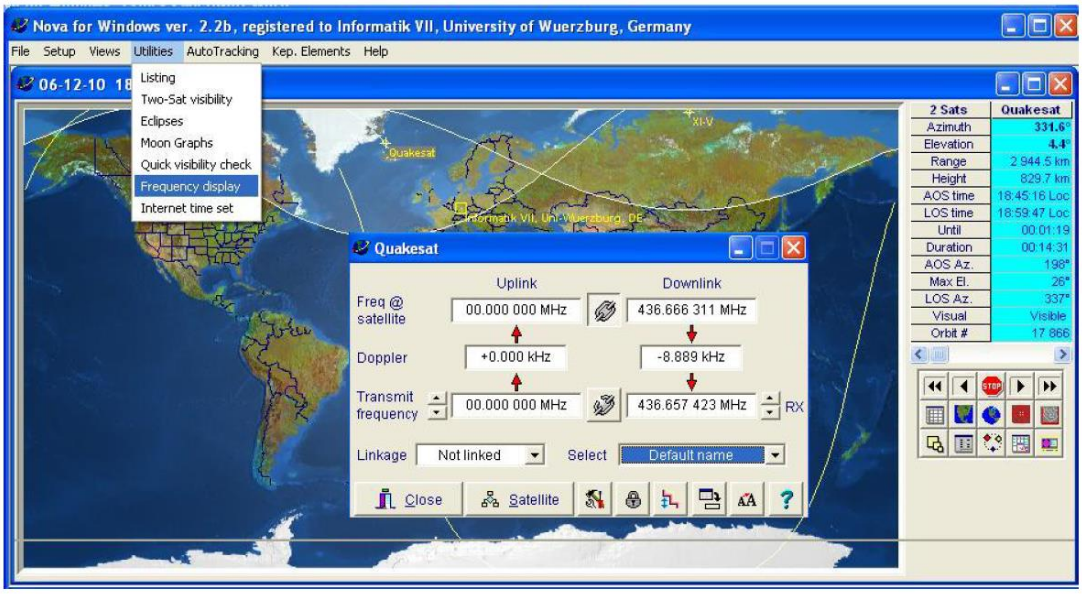


Figure B.10: Nova for Windows - Frequency Display.

To enable **Auto-Tracking** with "Nova for Windows", the first step is to select the type of Antenna Rotator from the Rotator Interface list. The path is "AutoTracking" and then "Antenna Rotator Setup" and then "Interface".
Select the Rotator Interface from the available list.
For the Ground Station at Informatics VII, University of Wuerzburg, "WinRotor" is the Rotator Interface.

For **Yaesu G-5500** azimuth rotator range is 0 to 360 degrees and elevation rotator range is 0 to 180 degrees. Refer figure 2.21.

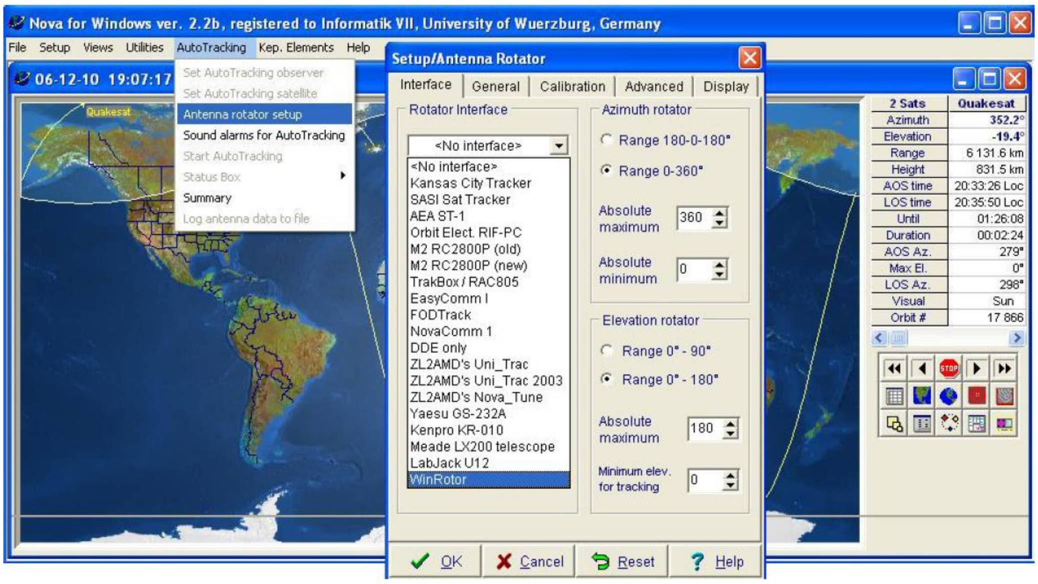


Figure B.11: Nova for Windows - Antenna-Rotator Setup.

More ***help*** regarding “Nova for Windows” can be available from “help” of the display window or please refer its detailed brochure. Refer figure 2.22.

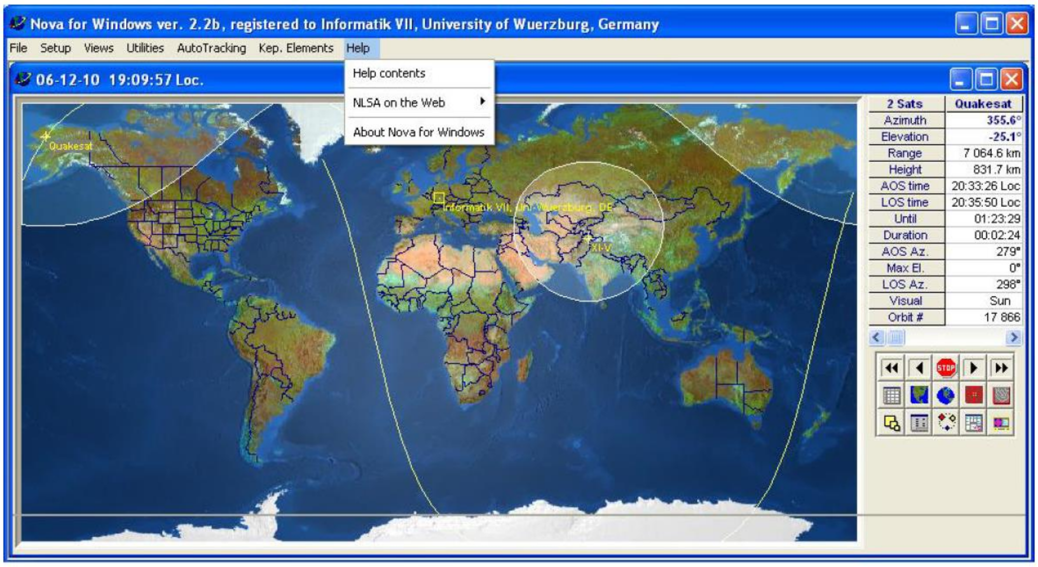


Figure B.12: Nova for Windows - Further Help.

Test Results

Eleven tests are documented in a duration of 10 days from 08 Dec 2006 to 18 Dec 2006. Testing summary is as follows:

08 December 2006	
AOS Time	16:04:08 Local Time
LOS Time	16:17:42 Local Time
Duration	00:13:33 hrs
AOS Azimuth	118 degrees
Maximum Elevation	21 degress
LOS Azimuth	355 degress

Table C.1: Testing - Satellite Script 08Dec2006.

Number of Beacons received : 3.

tnc4e2: fm KD7OVB to QST ctl UI pid=BB len 255 16:10:18
tnc4e2: fm KD7OVB to QST ctl UI pid=BB len 255 16:10:28
tnc4e2: fm KD7OVB to QST ctl UI pid=BB len 255 16:10:57

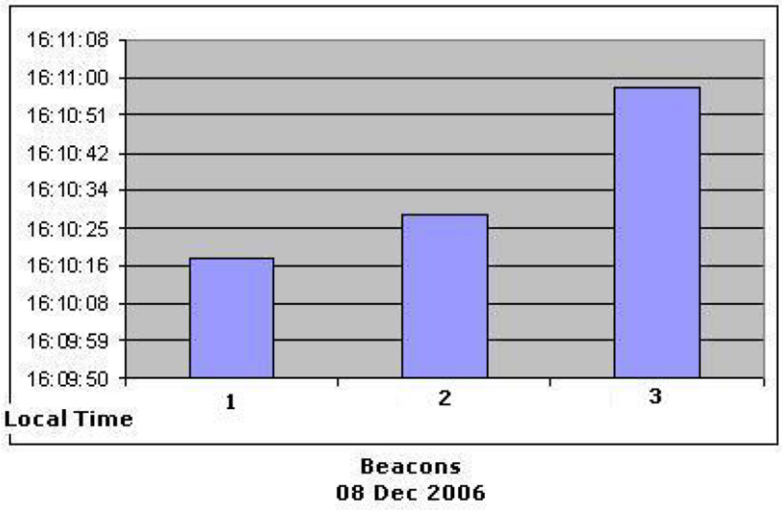


Figure C.1: Test Beacons on 08Dec2006.

11 December 2006	
AOS Time	15:08:46 Local Time
LOS Time	15:19:13 Local Time
Duration	00:10:26 hrs
AOS Azimuth	87 degrees
Maximum Elevation	8 degress
LOS Azimuth	1 degress

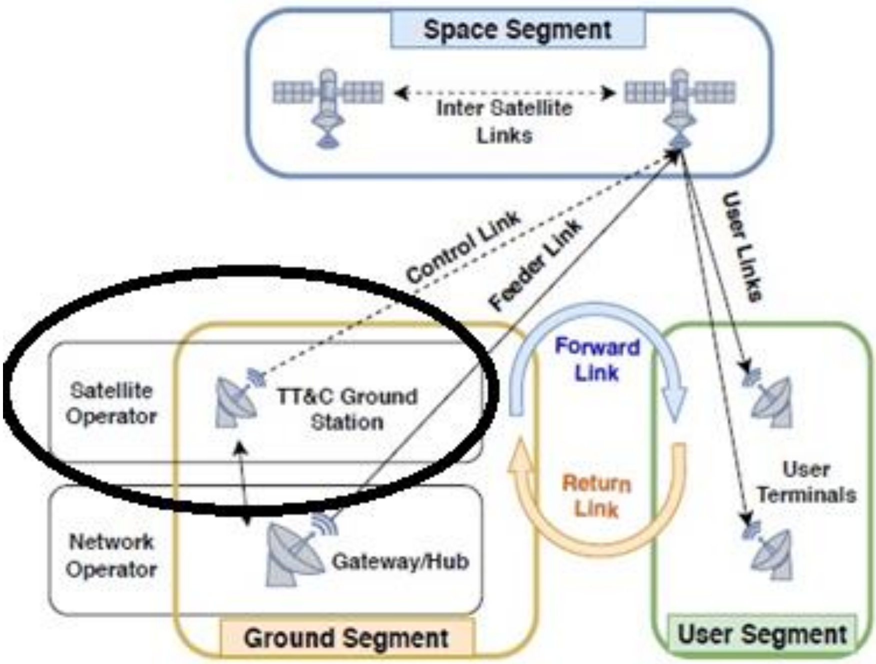
Table C.2: Testing - Satellite Script 11Dec2006.

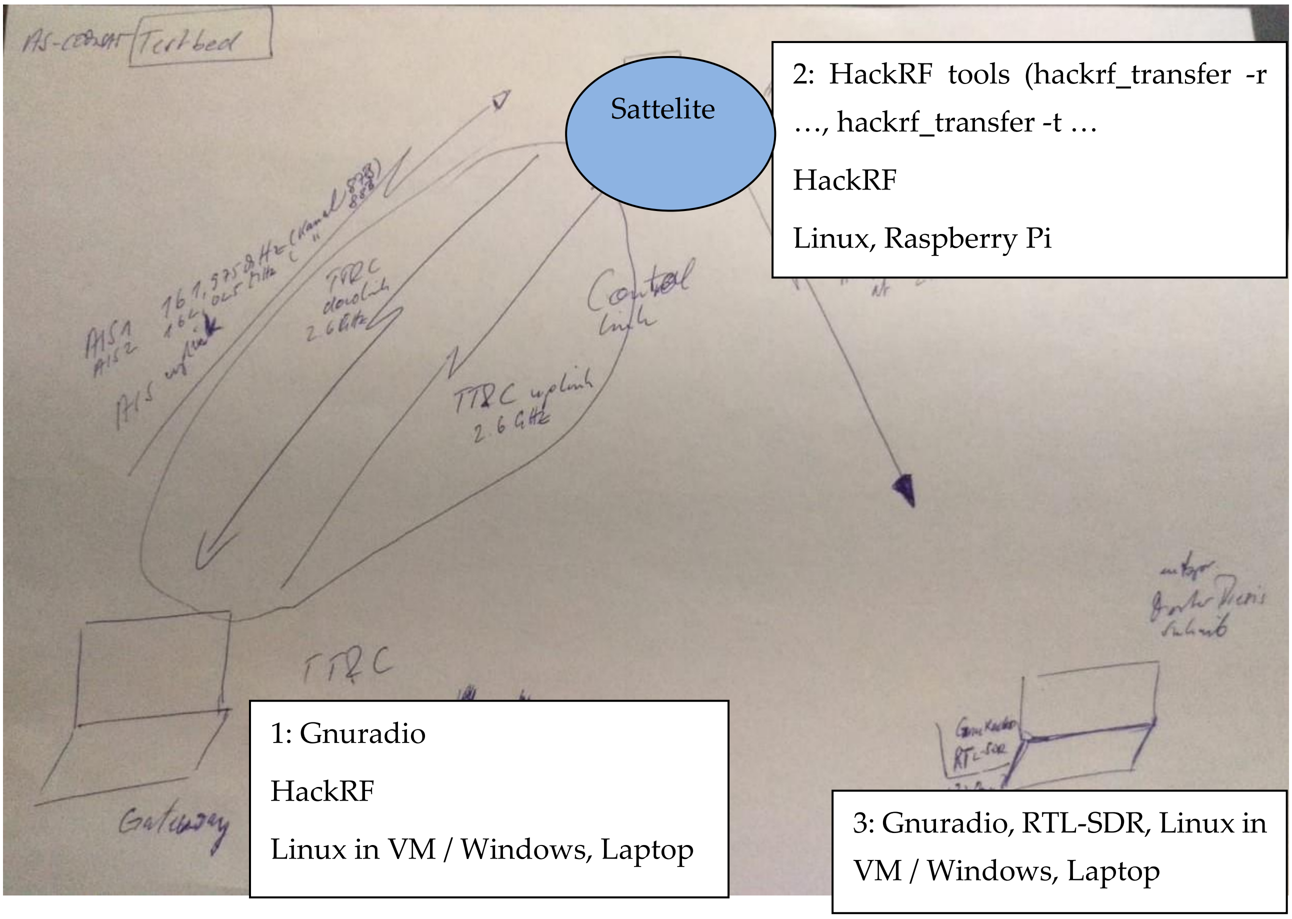
11 December 2006	
AOS Time	16:45:23 Local Time
LOS Time	17:00:18 Local Time
Duration	00:14:55 hrs
AOS Azimuth	139 degrees
Maximum Elevation	40 degress
LOS Azimuth	351 degress

Table C.3: Testing - Satellite Script 11Dec2006.

8.2.3 Hardware in the loop (HIL) Test System

8.2.3.1 System Design of HIL





Payload: Sending from 1 to 2 an AIS file on 161.975 MHz – Sending from 2 to 3 this file on 2.6 GHz

Telemetry, Tracking & Control (TT&C): Sending from 1 to 2 a control command file on 2.6 GHz, sending from 2 to 1 a file with sensor information on 2.6 GHz

8.2.3.2 HackRF - RTL-SDR - GNU RADIO Setup[13]

Orange Pi – Raspberry Pi

HackRF One is an SDR (Software Defined radio) working in transmission and reception in a wide frequency range. 1MHz to 6GHz. RTL-SDR is the cheapest and most well-known SDR working in reception only. Here we will discuss the connection of these 2 SDRs to an Orange Pi or a Raspberry PI and their operation with the GNU Radio software.

Connection of the HackRF One SDR to an Orange Pi Pc2 or Orange PI One Plus running on ARMBIAN or a Raspberry PI.

HackRF Library

In console mode, update your system:

sudo apt-get update

sudo apt-get upgrade

[13] HackRF – RTL-SDR – GNU RADIO Setup – F1ATB (https://f1atb.fr/index.php/2020/08/06/hackrf-orange-pi-gnuradio-setup/)

Install the library for hackrf:

sudo apt-get install hackrf

Connect the hackrf to an USB port and check it:

hackrf_info

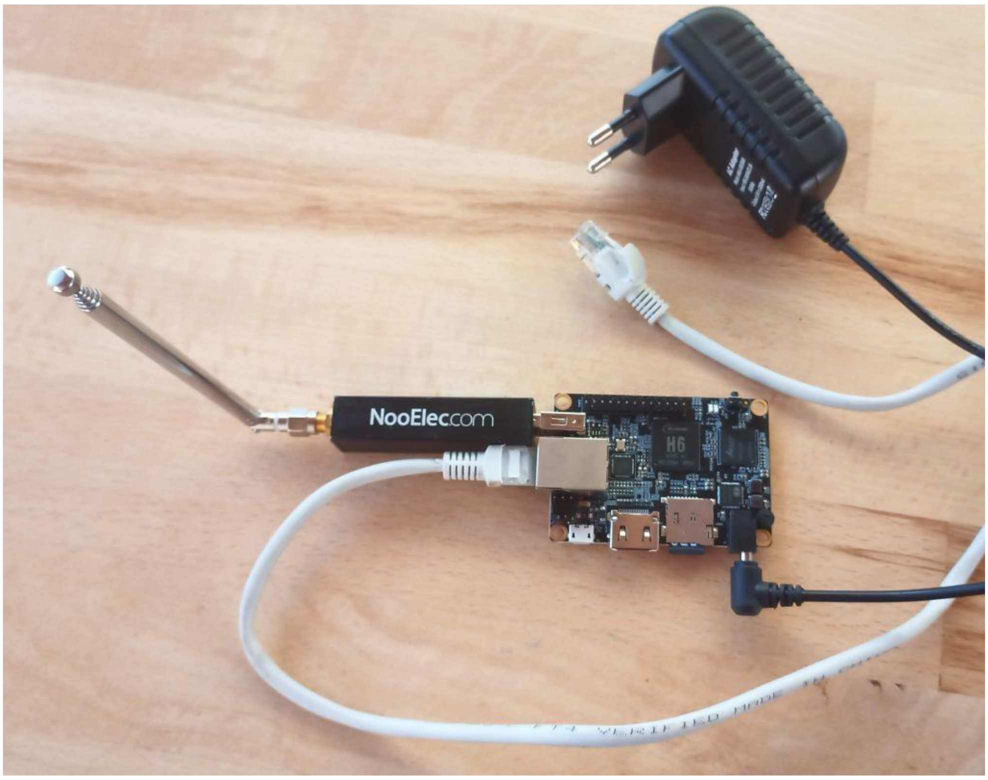
If the installation is good, you will get a response "Found HackRF" etc … Sometimes with some USB cables that connect the HackRF, there is no response. Change the cable in this case.

Installation GNU Radio Companion

GNU Radio is a very powerful tool for modelling signal processing chains. GNU Radio Companion is a complementary tool allowing to build radio processing chains graphically without writing a line of code. So far for Debian Buster there is only version 3.7.13 which is considered stable.

sudo apt-get install gnuradio

Installation Osmocom SDR and RTL-SDR Drivers



RTL-SDR and Orange PI One Plus

sudo apt-get update

Installation of the USB library:

sudo apt-get install cmake build-essential libusb-1.0-0-dev

We clone Osmocom in the user's Downloads folder for example:

cd ~/Downloads

sudo git clone https://github.com/osmocom/rtl-sdr.git

Go to the folder rtl-sdr:

cd rtl-sdr

sudo mkdir build

cd build

sudo cmake ../ -DINSTALL_UDEV_RULES=ON -DDETACH_KERNEL_DRIVER=ON

At this point I sometimes got an error because the pkg-config package was not found. Make:

sudo apt-get install pkg-config

and relaunch the cmake from above.

sudo make

sudo make install

sudo ldconfig

Osmocom source module installation

sudo apt-get install gr-osmosdr

GNU Radio launch

You have to be in graphics mode to be able to use this superb signal processing design tool. Personally, I use my orange-pi or raspberry in remote mode. I connect to them by enabling VNC in setup and using the VNC extension in chrome on my PC. On a terminal in graphics mode type:

gnuradio-companion

A message "RANDR" missing … is not important.

Error Xterm executable is missing

If you have when launching a model in gnuradio companion a message of the type Xterm executable is missing and if you are on Armbian or Raspbian (raspberry) go to the configuration folder.

cd ~/.gnuradio

edit grc.conf

sudo nano grc.conf

and add at the end of the file:

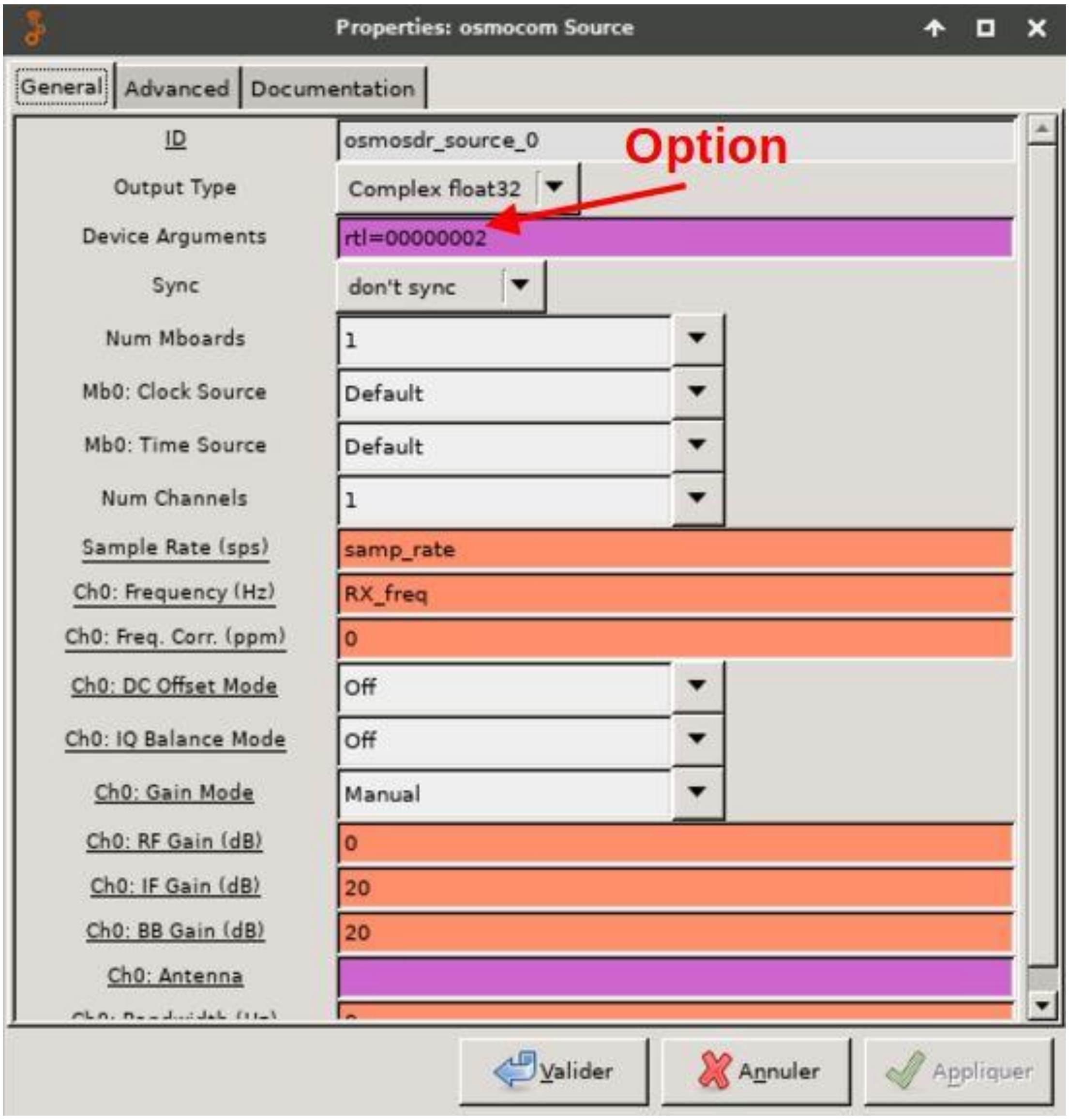
```
[grc]

xterm_executable = /usr/bin/lxterminal
```

First of all, check that the terminal is installed on your machine. Otherwise installed it with:

sudo apt install lxterminal

Osmocom module

To enter data from HackRF One or RTL-SDR, use the Osmocom source module in GNU-Radio.



It is not necessary to identify the SDR if only one is connected to the processing board. For the gains, it is necessary to make tests to find the good values according to the model of SDR.

Note On Hack RF Gains

HackRF One provides:

- RX : three different analog **gain** controls
  - RF ("amp", 0 or 14 dB),
  - IF ("lna", 0 to 40 dB in 8 dB steps),
  - baseband ("vga", 0 to 62 dB in 2 dB steps)
- TX : two gain controls
  - RF (0 or 14 dB) ,
  - IF (0 to 47 dB in 1 dB steps)

The RX and TX, RF amplifiers have two settings: on or off. In the off state, the amps are completely bypassed. They nominally provide 14 dB of **gain** when on, but the actual amount of gain varies by frequency. In general, expect less gain at higher frequencies.

Note on RTL-SDR Gains

Only the RF gain parameter acts on the RTL-SDR. You can get the available gain values provided that the RTL-SDR was not started by an application. In a terminal window type:

rtl_test

```
root@opi-onep-70:~# rtl_test

Found 1 device(s):

  0:  Realtek, RTL2838UHIDIR, SN: 00000001



Using device 0: Generic RTL2832U OEM

Detached kernel driver

Found Rafael Micro R820T tuner

Supported gain values (29): 0.0 0.9 1.4 2.7 3.7 7.7 8.7 12.5 14.4 15.7 16.6 19.7 20.7 22.9 25.4
28.0 29.7 32.8 33.8 36.4 37.2 38.6 40.2 42.1 43.4 43.9 44.5 48.0 49.6

[R82XX] PLL not locked!

Sampling at 2048000 S/s.



Info: This tool will continuously read from the device, and report if

samples get lost. If you observe no further output, everything is fine.
```

Multi-SDR

In the case of several SDRs connected to the processing card, they must be identified. For a hackrf type in a terminal hackrf_info and retrieve the serial number and put it in the 'Device Arguments' box. Ex:

```
hackrf=000000000000000075b068dc3122a607
```

For an RTL-SDR, type rtl_eeprom and put the serial number. Ex:

```
rtl=00000002
```

The difficulty with RTL-SDRs is that they all carry the serial number 1 when they are manufactured. You can reprogram this number by typing:

```
rtl_eeprom -s 'numéro de série'
```

With the rtl_eeprom -h command, we have the corresponding help.

Purchases

The HackRF being developed in Open Source can be bought in China at Aliexpress for a hundred Euros. To use it, the Orange Pi solution is very interesting. We forget the Orange Pi Zero which, following the treatment defined in GNU-RADIO, quickly risks running out of power. We are going to move towards 64-bit 4-core processors like the H5 or the H6. I have successfully tested Orange Pi PC2 H5 and Orange PI One Plus H6 for SSB reception and SSB transmission.

Orange PI PC2

The Orange PI PC2 has the following advantages:

- several USB2 if you want to connect different SDRs
- An audio output on headphone jack, useful for an HF receiver

Orange PI One Plus

The Orange PI One Plus has the following advantages:

- The minimum required to control a Hack RF or a RTL-SDR from the network (USB, Ethernet 1Gb / s)
- A very low price: less than 30 € with food and transport (Ebay or Aliexpress)

Posts on Remote-SDR

- Remote SDR v3
- Gpredict — Remote SDR
- Remote SDR – Raspberry Pi 4B or Orange Pi Zero 2 image installation
- Remote SDR v3 – Manual Installation
- SA818 – RTL-SDR
- Remote SDR – Examples of realization
- Transmit over QO-100 satellite with a Smartphone
- Remote SDR V2 – Software Architecture
- Remote SDR v3 – Tips
- Remote SDR V1- Purchase
- Remote SDR V1 – Man Machine Interface
- Remote SDR V1 – Signal Processing
- Web Client to GNU Radio
- GNU Radio to Web client

- Remote SSB Transmitter
- Remote SSB Receiver
- GPIO on Orange PI One Plus H6
- TCXO installation on HackRF
- Q0-100 Transceiver with 2 SDR – Remote SDR V1

8.2.3.3 Other Gnuradio/HackRF instructions

Installing HackRF on Raspberry:

HackRF One installation

- Prerequisites

  If you did not install these for SDRplay:

```
sudo apt install libusb-1.0-0-dev libfftw3-3 libfftw3-dev
```

- Install

```
• cd ~/
• git clone https://github.com/mossmann/hackrf
• cd hackrf/host
• mkdir build
• cd build
• cmake ..
• make -j 4
• sudo make install
```

```
sudo ldconfig
```

- reboot
- You may need to update the firmware to match the version of the driver. As of this writing, the latest firmware/driver version was 2021.03.1

Most HackRF devices have firmware version 2018.01.1 or earlier. Use hackrf_info to see if the firmware version matches the driver that you installed.

```
hackrf_info

# If you see "Firmware Version: 2021.03.1 (API:1.04)", then it is up to date.

# Transfer rate test:
hackrf_transfer -r /dev/null -s 21500000

#Test sending data:
hackrf_transfer -t /dev/zero
```

- Firmware updating instructions are based on this: https://hackrf.readthedocs.io/en/latest/updating_firmware.html#updating-the-spi-flash-firmware Note that the zip and tar.xz archives have the firmware files. The git clone procedure does not.

  Download hackrf-2021.03.1.zip

  Extract only the firmware-bin directory into the ~/hackrf directory

```
cd ~/hackrf/firmware-bin

# To update the firmware on a working HackRF One, use the hackrf_spiflash program:
```

```
• hackrf_spiflash -w hackrf_one_usb.bin
•
• # Press the reset button on the HackRF.
• hackrf_info
•
```

```
# The new firmware version should now displayed.
```

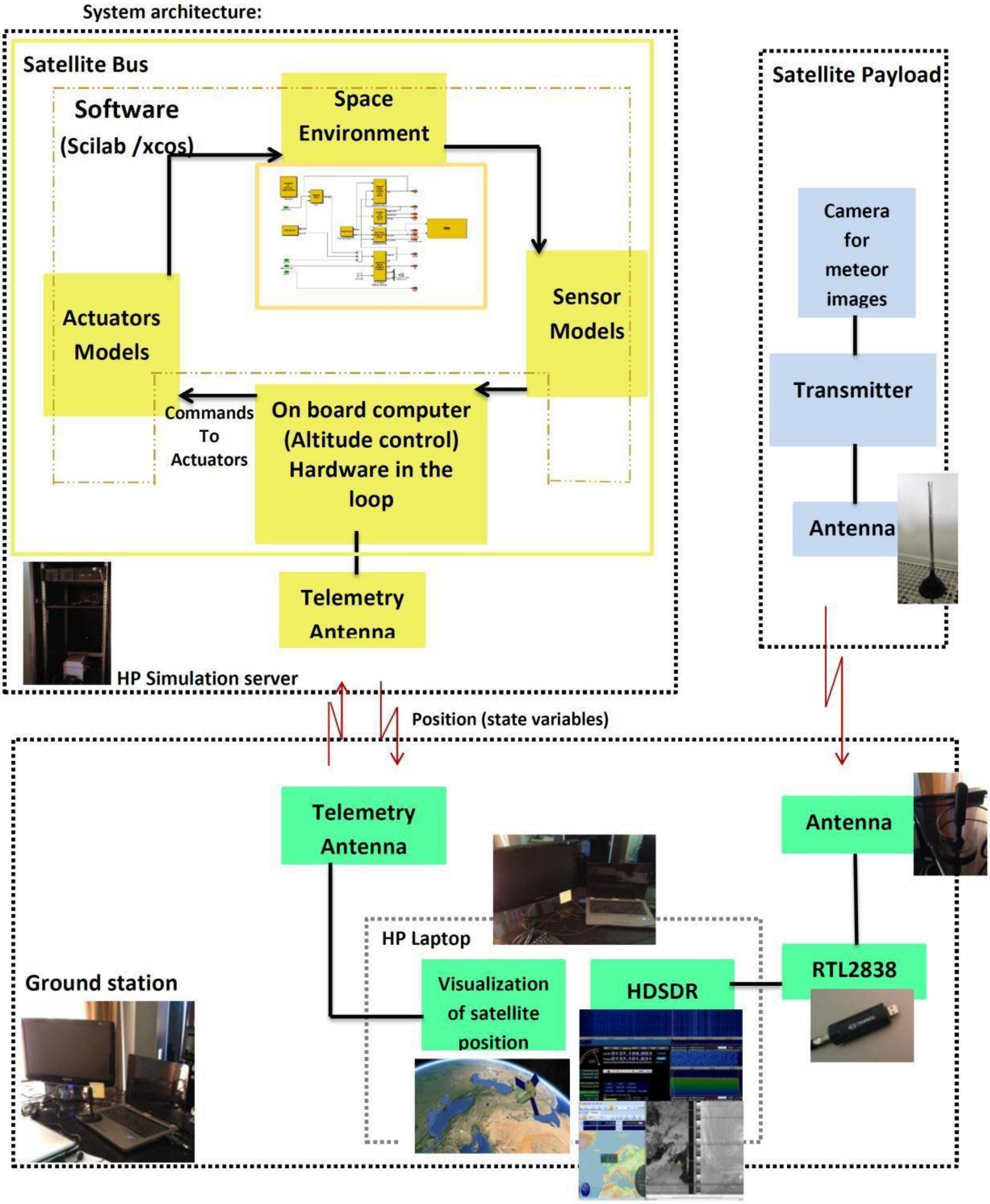
Installing gnuradio on raspberry:

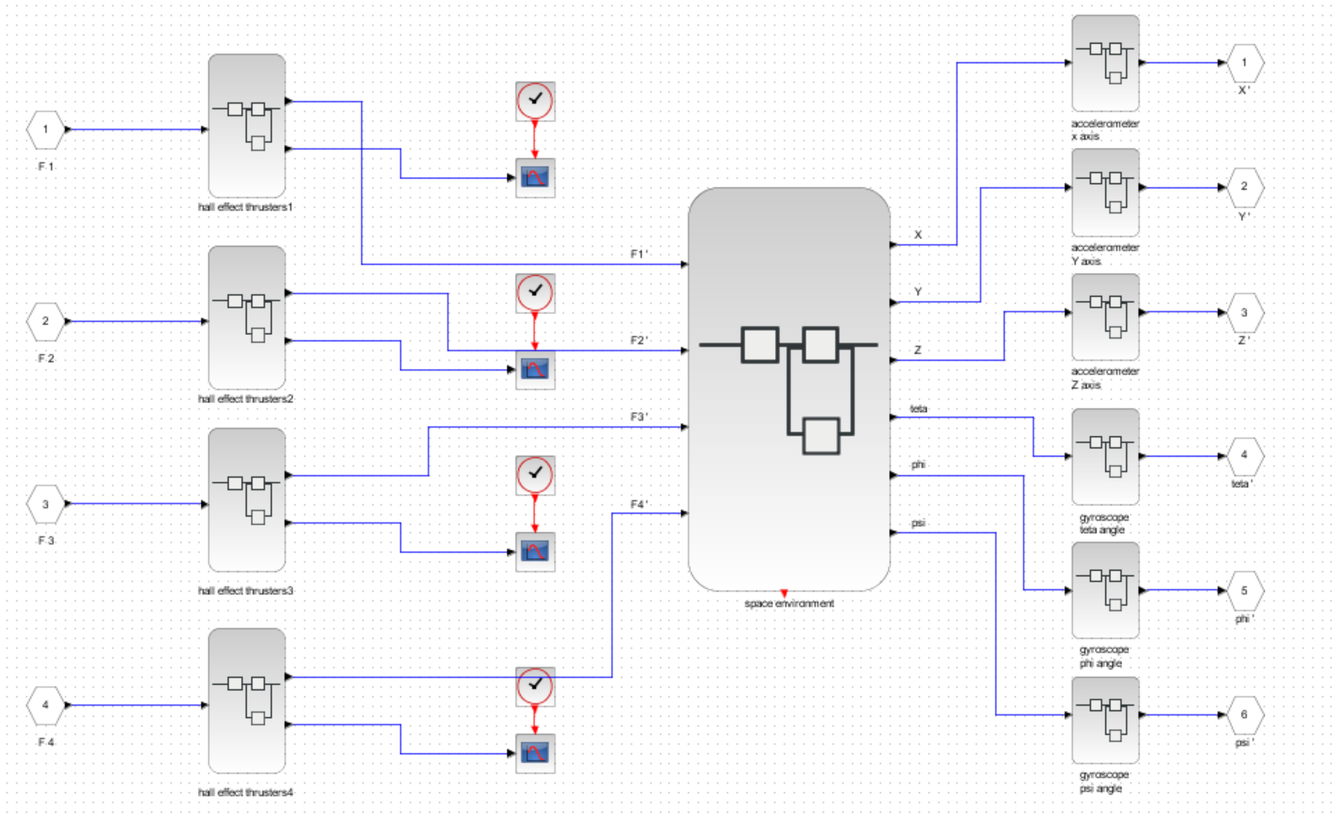
apt-get install -y gnuradio

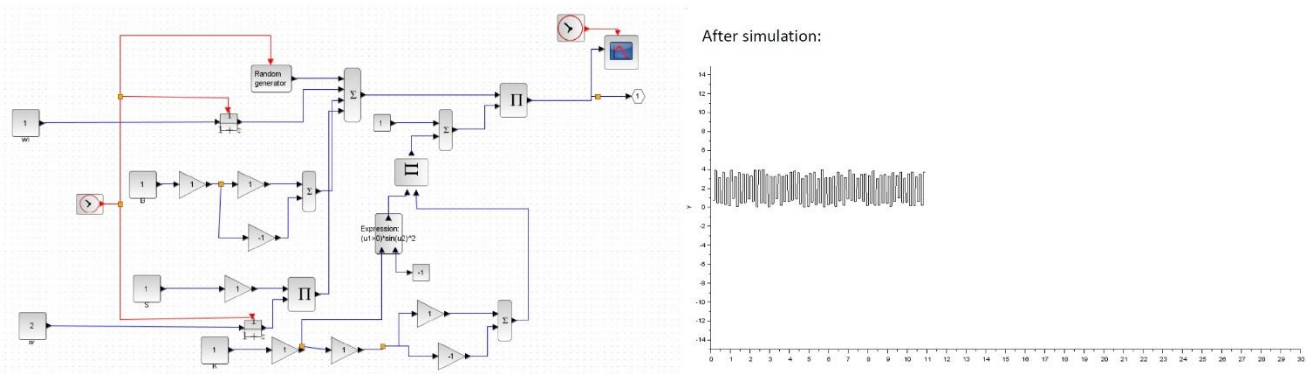
running gnuradio:

gnuradio-companion

8.2.3.4 Hardware-in-the-Loop test rig for IAP-SAT (Overview) (2015)







In this figure, input 1 is the angular speed wi, and imput 2 is the acceleration. These inputs is passed by the scale factor, the noise, transfer function and the misalignement of the axis to measure the final angular speed as output of gyroscope.

Overview is including Simulation model of actuators, space model and sensors of IAP-SAT (for 1. xcos model, 2. Graphs of simulation results please refer to documentation)

- Interface between Simulation Server and Board Computer of IAP-SAT (for 1. xcos model, 2. Graphs of simulation results): please refer to documentation)

- Meteorological Images supply by HDSDR

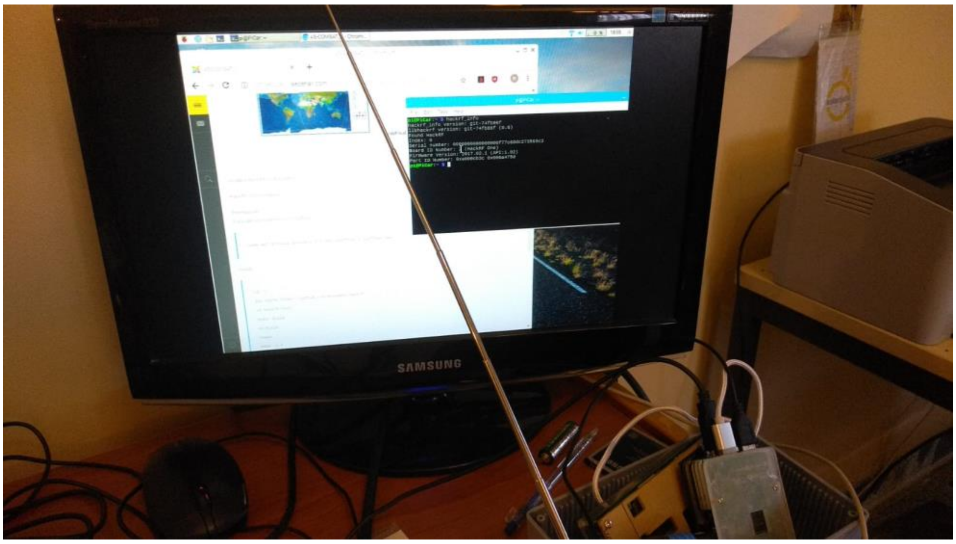
8.2.3.5 HIL 2021

Space Segment:

- Raspberry (Linux, Gnuradio, NASA coreFlightSystem)
- ACS (magnetorquer, IMU from IAP-SAT))
- HackRF

- Antenna
- Structure

Gnuradio program and HackRF drivers successfully installed on raspberry pi



Ground Segment:

- Laptop (Windows or Linux, GnuRadio/PothosSDR)
- Antenna
- HackRF

Also possible with

Programm CubicSDR, PothosFlow (instead of gnuradio)

Task Sending and Receiving file

1. On 2.6 GHz band: sending a file from space segment
2. Receiving this file on ground segment and visualize it

   Especially:

1. Recording file from remote control of car opening (probably 433,92 MHz) by raspberry
2. Sending this file to ground station

8.3 Hardware Design Document (HDD) & Hardware Realization Document (HRD)

8.3.1 TT&C Tranceiver

8.3.1.1 Crowd Supply LimeSDR Mini Boards

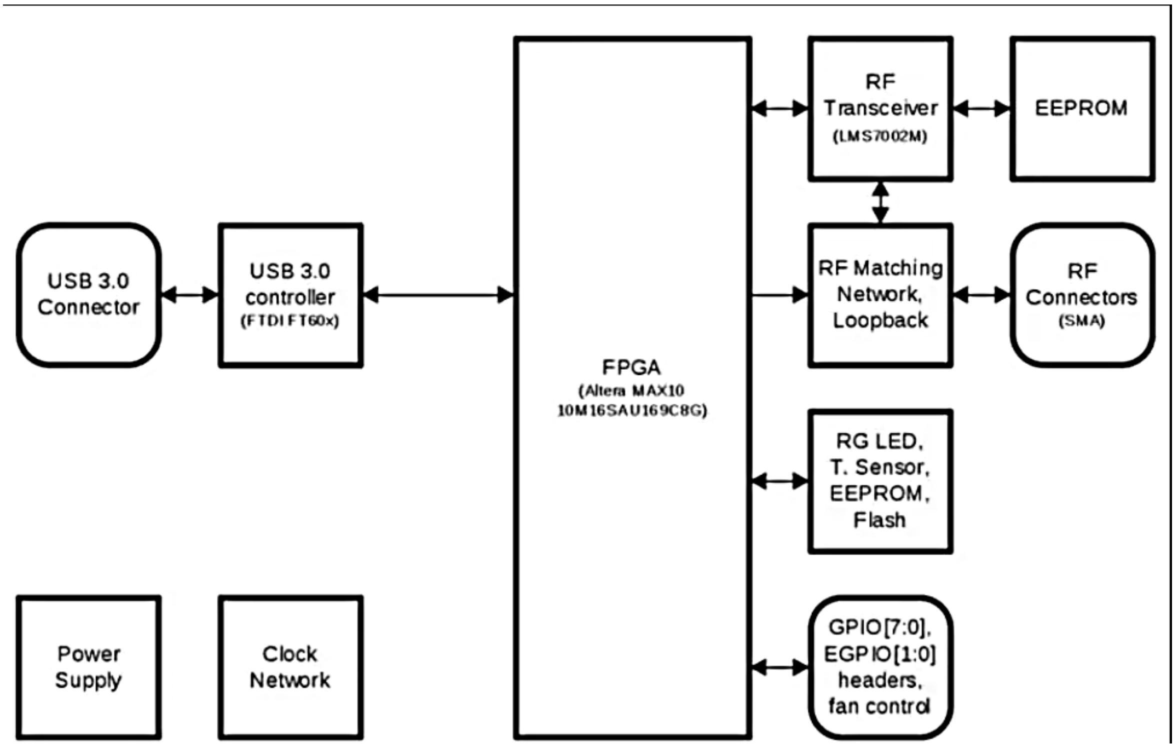
Crowd Supply Lime Software Defined Radio (SDR) Mini Boards are hardware platform for developing and prototyping high-performance and logic-intensive digital and RF designs. These boards use Altera's MAX 10 FPGA and Lime Microsystems' LMS7002M RF transceiver. The LimeSDR Mini boards are smaller, and less expensive when compared to the LimeSDR. These mini boards feature two 128KB for RF transceiver MCU firmware, and a 4MB flash memory for data.

The LimeSDR Boards feature 2 SMA (SubMiniature A) connectors for connecting external transmit and receive antennas, such as the Taoglas TG.09.0113. A U.FL connector is provided for an external clock source, such as GPSDO or atomic clock, via the Taoglas CAB.721 antenna.

Features

- **Lime Microsystems LMS7002M** MIMO FPRF transceiver
- **Altera MAX 10** (10M16SAU169C8G) FPGA:
    - 169-pin FBGA package
    - 549KB M9K memory
    - 2368KB user flash memory
    - 4 x fractional Phase Locked Loops (PLLs)
    - 130 x general purpose input/output (GPIO)
    - Single supply voltage
    - Flash feature
    - FPGA configuration via JTAG
- 2 x 128KB for RF transceiver MCU firmware and data
- 1 x 4MB flash memory for data
- General user inputs/outputs:
    - 2 x dual color (red and green) LED
    - 8 x FPGA GPIO pin header (3.3V)
- Connectivity:
    - USB 3.0 Type-A (FTDI FT601 controller)
    - 2 x coaxial RF SMA connectors
    - U.FL connector for external clock source
    - FPGA GPIO headers
    - FPGA JTAG connector
- Clock system:
    - 30.72MHz onboard VCTCXO
    - Possibility to tune VCTCXO with onboard DAC
    - External clock input via U.FL connector
- 69mm x 31.4mm dimensions
- Weighs about 20g

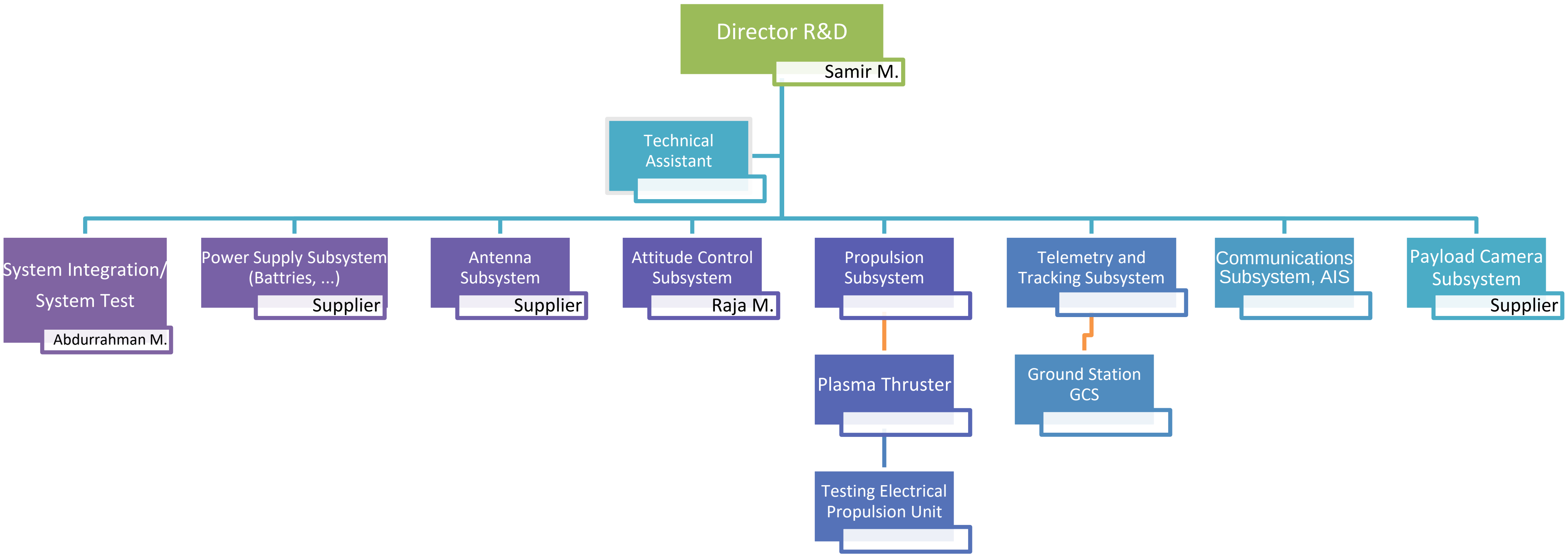
8.3.1.2 Lime SDR mini



8.3.2 TT&C Antenna

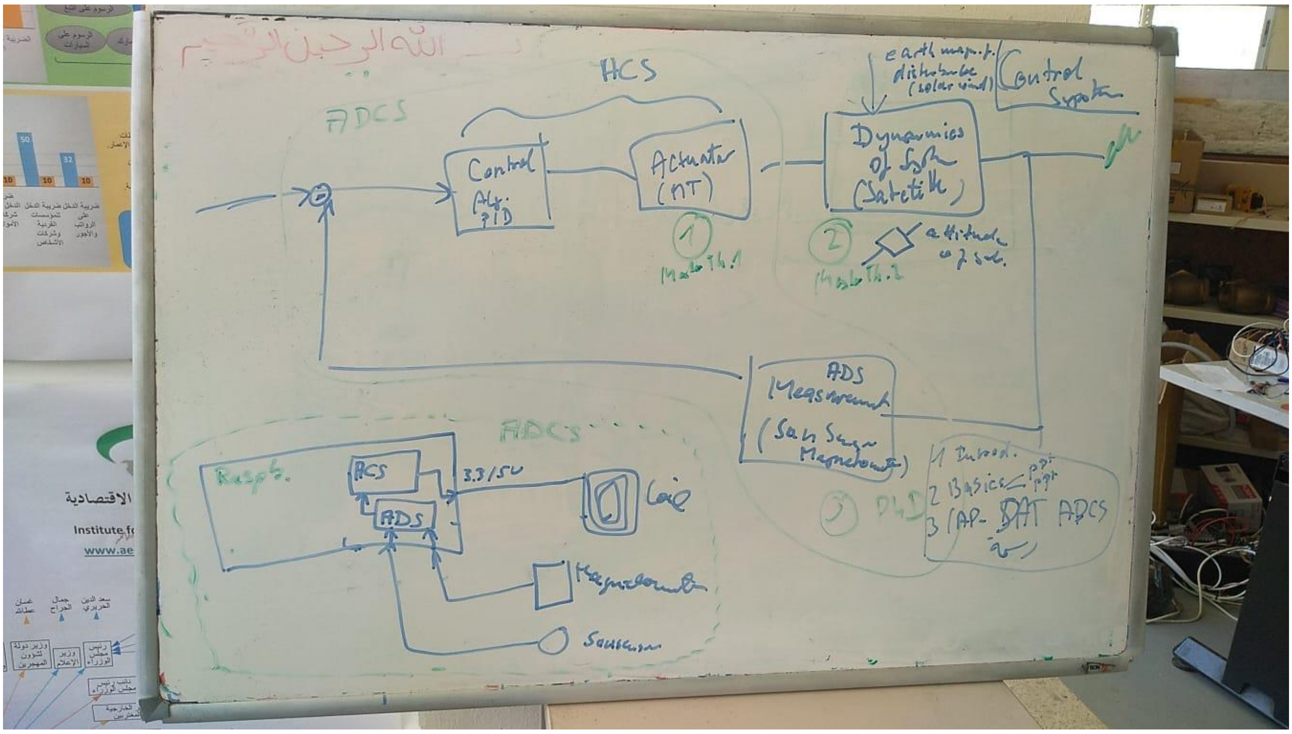
8.4 Software Design Document (SWDD) & Software Implementation Document (SWID)

8.4.1 Organizational chart (current and future)



8.5 AS-COMSAT-1 System (Hardware&SW)

8.5.1 System Concept for Attitude Determination and Control System (ADS, ACS)



8.5.2 On-Board Computer (Raspberry Pi)

On this computer the NASA core flight system shall be implemented. System design: Jana Othman (Internship AECENAR July-Aug 2021).

8.5.2.1 Physical System Overview

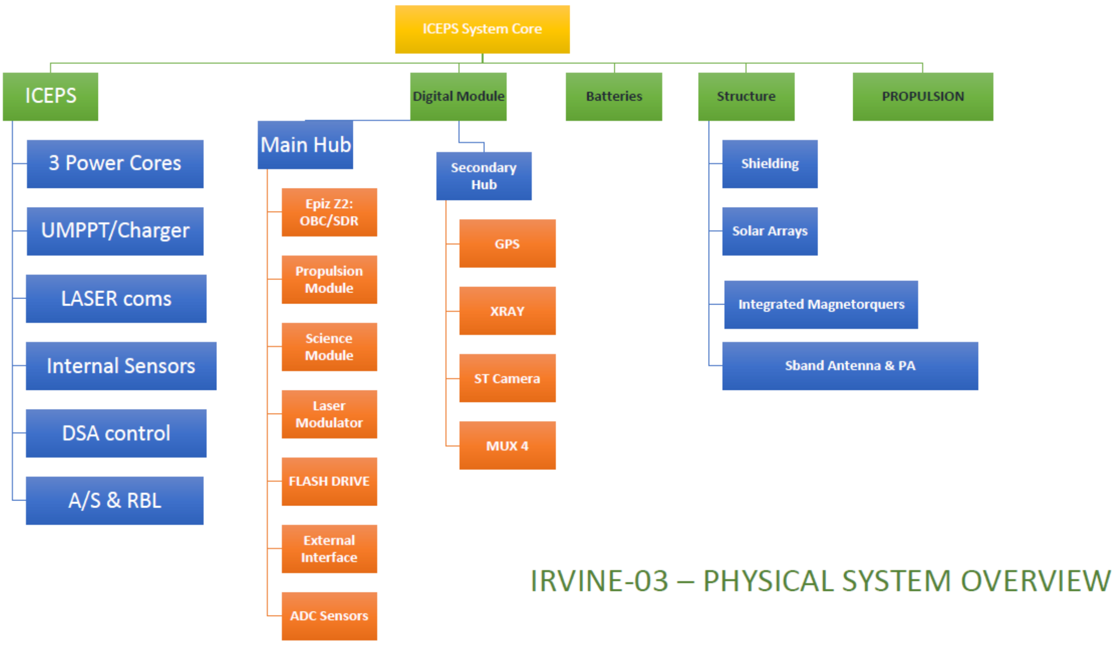


Figure 35: physical overview

8.5.2.2 How communication is done

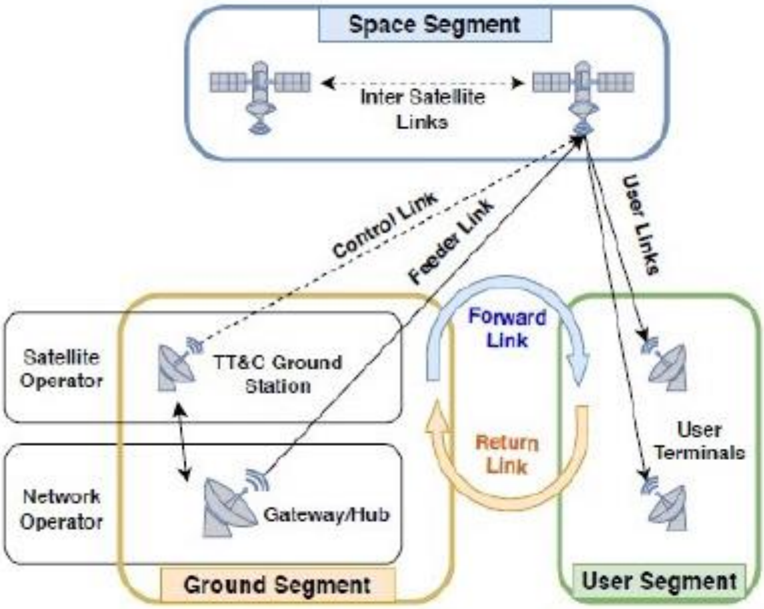


Figure 36: Communication and TT&C

8.5.2.3 OBC System

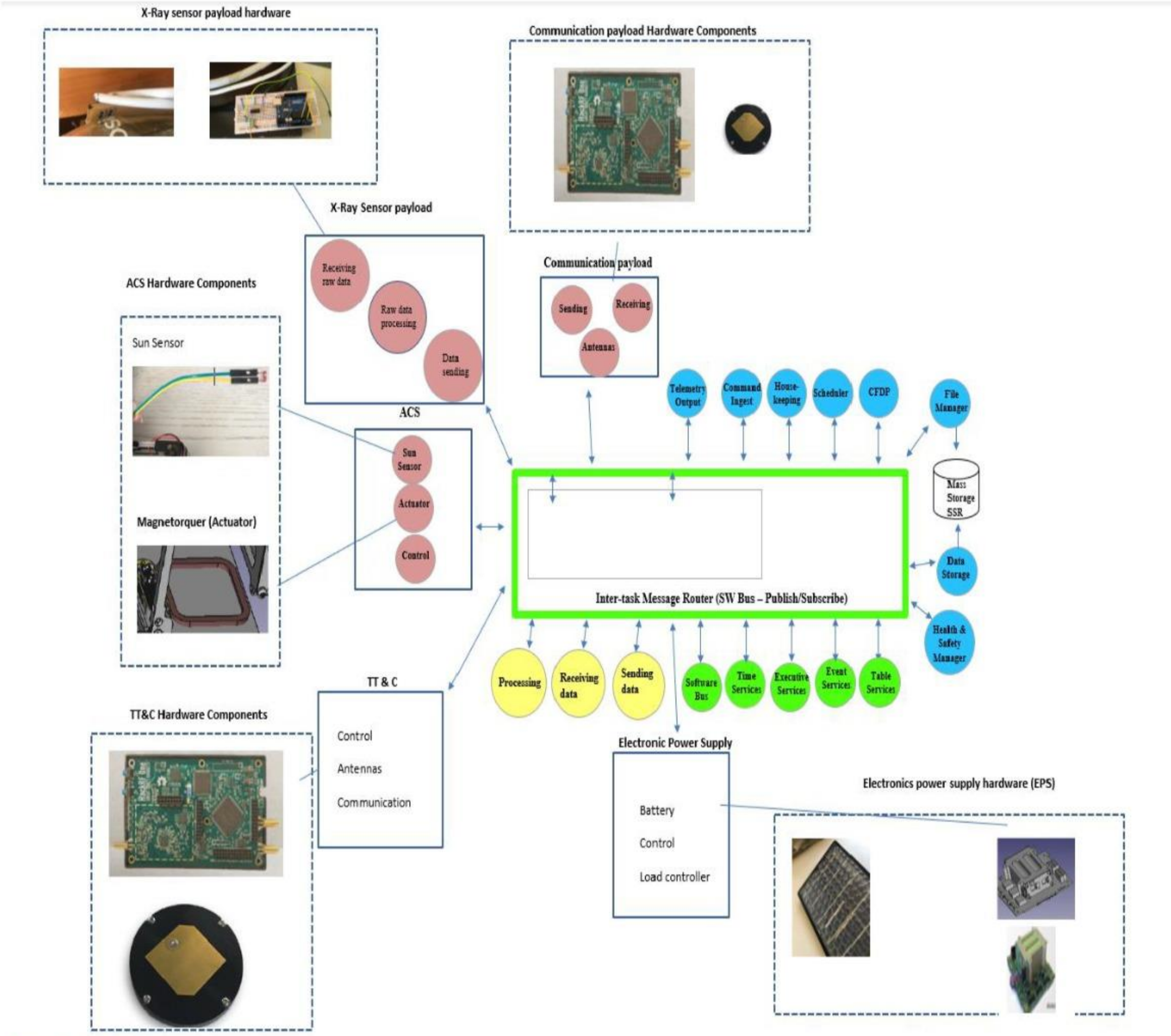


Figure 9: System Design

Legend:

: Components of a block

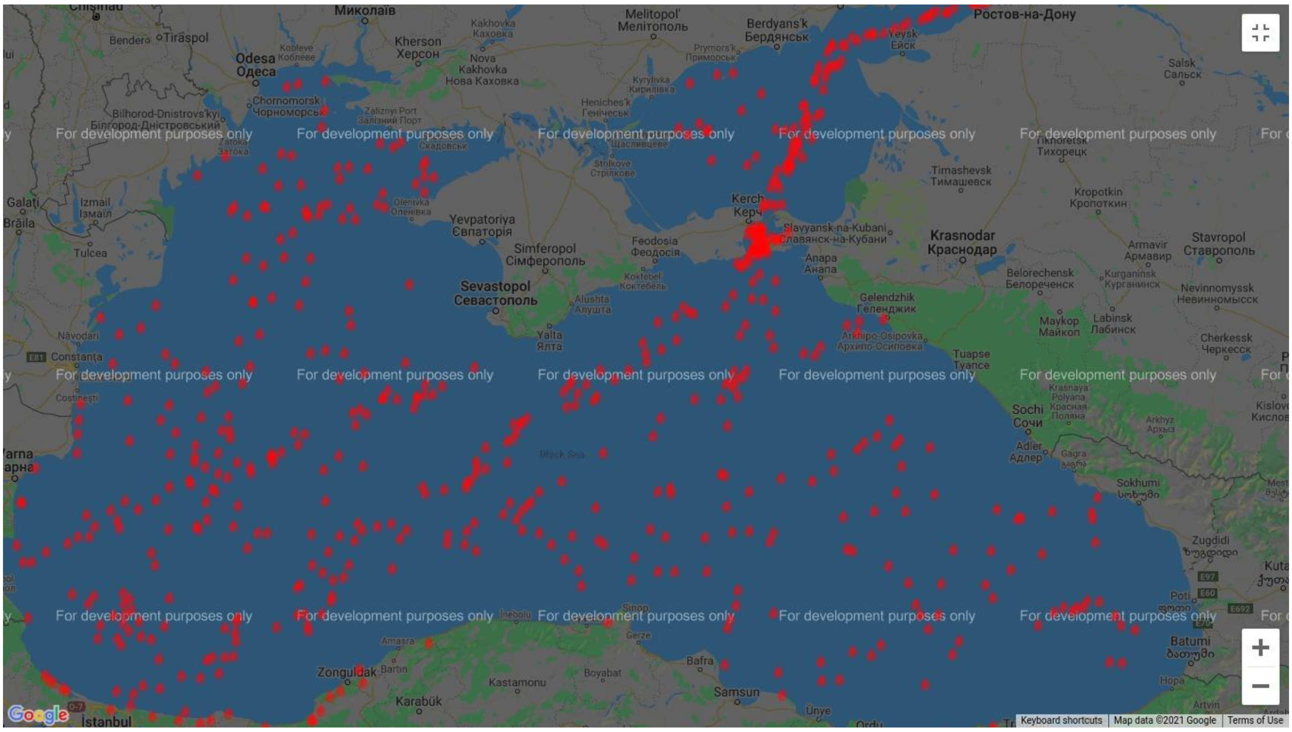
: CFS Configurable Applications

: cFE Core Services

: X-Ray Specific Apps

As payload also the AIS system shall be implemented (Rozan Mustafa as master thesis at Marmara Univ.)

8.5.2.4 AIS system on the OBC



8.5.3 HackRF Card (Responsible: Abdurrahman)

Establishing digital communication between HackRF and On-Board Computer

8.5.4 Attitude Control System (ACS) (Responsible: Raja)

Sensors: Sun sensor, IMU (see FCS of TEMOLeb-Mintad 2018 in TEMOLeb-Mintad Final Report[14])

Actuators: Magnetorquer

8.5.5 X-Ray Sensor (Responsible: Yahya, Raja)

This project will present the design and implementation of an X-ray detector system that is based on low-cost PIN diodes (BPW34). General procedure of designing the preamplifier and shaping amplifier will be presented as well as the practical approach to implement this system.

Typically, the PIN diodes (detectors) have low-voltage input signal and have to be amplified via an extremely low-noise preamplifier. The preamplifier acts as a first stage in a chain of amplifiers that has very sensitive characteristics. Then, a shaping amplifier is used to make a reasonable signal shape that has the shape of a tail which can be processed and interpreted by a microcontroller. An ADC will then be used to transform these analog quantities to digital. Figure 37 below shows the overall process of converting input charges from a detector into digital readable signal proportional to input charges.

[14] See [2]

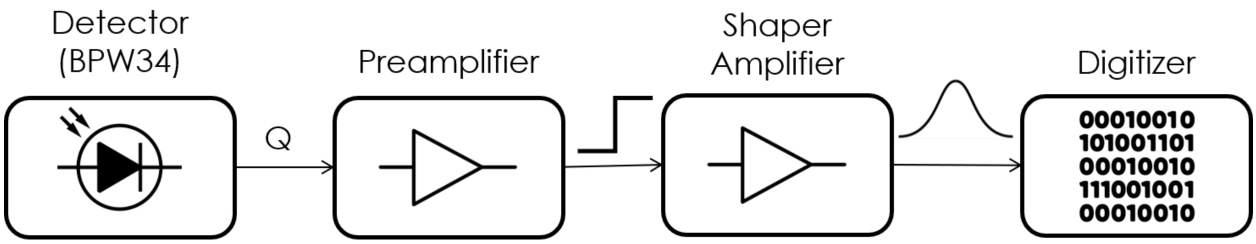


Figure 37. Silicon Detector Amplification Stages

There are mainly 3 types of preamplifiers and are listed below. However, we are just interested in charge-sensitive preamplifiers which are the type used for charge detector applications.

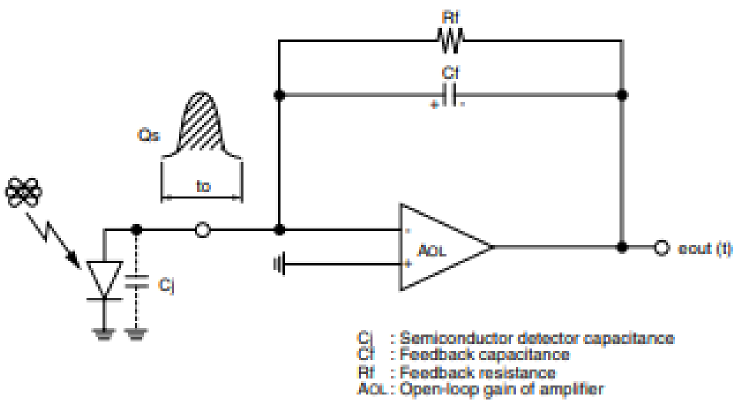
- Voltage-sensitive preamplifier.
- Current-sensitive preamplifier.
- Charge-sensitive preamplifier.

8.5.6 Charge Amplifier

As mentioned before, the PIN diode, when used to detect X-rays, outputs a very weak charge pulse having a pulse width of several tens of nano seconds. Since the detector is can be modeled as a capacitive device, its impedance is high and the preamplifier should be designed according to this criterion. For these applications, an integrator amplifier (using capacitor feedback) is used since it has high impedance, integrates weak charge pulses and converts them to voltage pulses ready for further amplification.

The first stage of charge amplifier (preamplifier) is usually a low-noise FET and its open loop gain must be set high in a way that the second amplification stage is not influenced by the detector capacitance.

When X rays strike into the detector, signal charges Q_s are generated with an amplitude proportional to the particle energy. These signal charges are all integrated in the feedback capacitance C_f (since it can be approximated that the current entering the op amp is equal 0) and then a pulse output $e_{out}(t)$ is generated.



In general, the following characteristics are required for the design of a good X ray charge amplifier:

- High gain
- Low noise (Excellent signal-to-noise ratio)
- Excellent integration linearity
- High speed rise time
- High temperature stability

8.5.7 Gain

The gain of charge amplifier is given in 2 ways; amplifier gain alone or the gain for detector/amplifier together. Amplifier gain (G_c) which is also referred to charge gain is given as:

$$G_c = \frac{V_{out}}{Q_s} = \frac{1}{C_f} \left[\frac{V}{Columb}\right] or \left[\frac{V}{picoColumb}\right] \tag{1}$$

The charge fall time of the amplifier can be determined by the feedback resistance and capacitor. $\tau_f = R_f C_f$

Amplifier with detector gain can be referred to as sensitivity (R_s). Sensitivity is expressed as output voltage mV per one MeV of energy particle irradiated onto the detector. The amplitude of the signal charge obtained with a semiconductor detector is determined by the input particle energy such as X-rays and also by the material of the semiconductor.

$$Q_s = \frac{E.e^-}{\varepsilon} \tag{2}$$

Where, E is the particle energy (MeV), e^- is the elementary charge ($1.6e^{-19}$ C), and ε is the energy required to create one electron/hole pair. For silicon, ε varies between 3.62 eV and 3.71 eV. From equations (1) and (2), amplifier's sensitivity can be written as:

$$R_s = \frac{V_{out}}{E} = \frac{e^-}{C_f}.\frac{1}{\varepsilon}\left[\frac{mV}{MeV}\right] \tag{3}$$

Noises:

Noises in charge amplifiers come generally from 3 sources: Thermal noise of first-stage FET, shot noise caused by the gate current of the first stage FET and dark current of the detector, and thermal noise caused by the feedback resistance. The noise of the first-stage FET is given as:

$$en_1 = \sqrt{\frac{8}{3}.\frac{KT}{g_m}} \tag{4}$$

Where, K is Boltzmann constant, T is the absolute temperature in Kalvin, and g_m is the mutual conductance of first-stage FET. The second noise source, shot noises, can be expressed as:

$$in = \sqrt{2q(I_G + I_D)}\ (A/\sqrt{Hz}) \tag{5}$$

Where, q is the elementary charge, I_G is the gate leakage current of first-stage FET, and I_D is the dark current of the detector. The third and final source of noises can be presented as:

$$en_2 = \sqrt{4kTR_f}\ (V/\sqrt{Hz}) \tag{6}$$

Where, R_f is the feedback resistance. From equations (4), (5), and (6), the total noise of a charge amplifier can be written as:

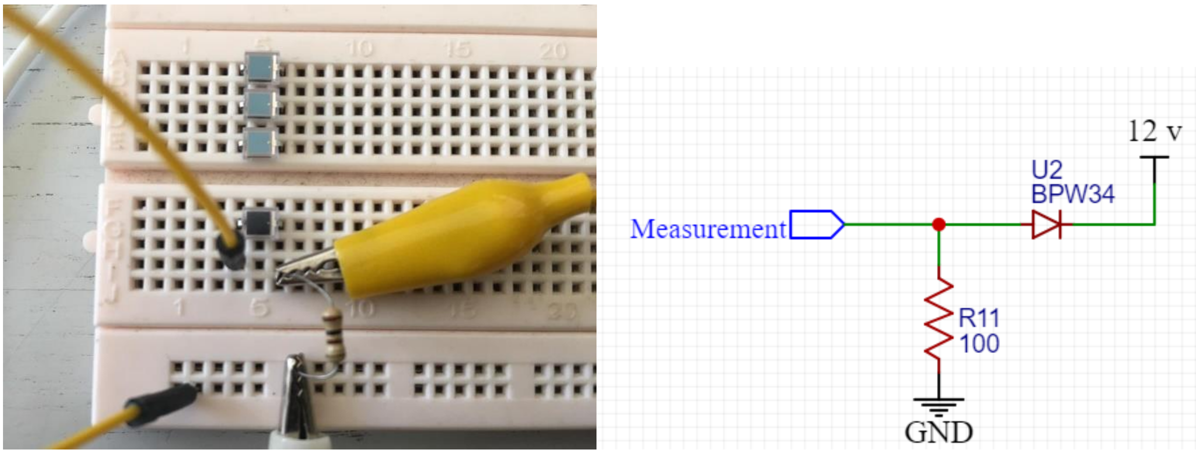
$$en_t(jw) = en_1^2 . \left(1 + \frac{C_{in}}{C_f}\right)^2 + \left[in^2 + \left(\frac{en_2}{R_f}\right)^2\right] . \frac{1}{\left(jwC_f\right)^2} \tag{7}$$

The first term is constant over the entire frequency range and amplified by the noise gain $(1 + \frac{C_{in}}{C_f})$ determined by the input capacitance. The second term component is constant regardless the input capacitance but it decreases with the frequency.

8.5.8 Experiments

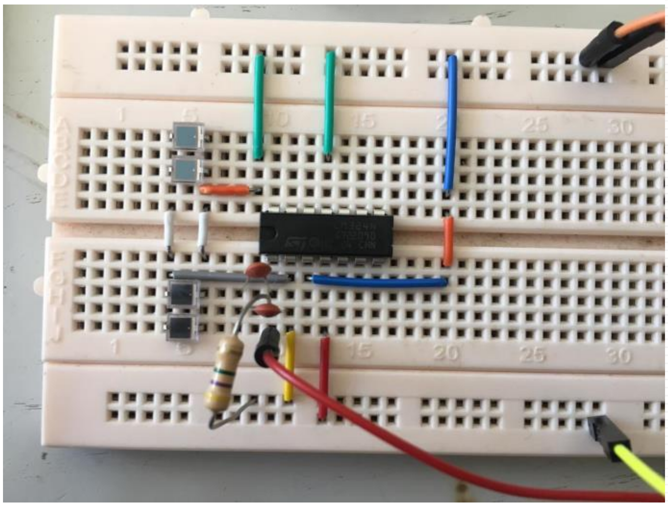
8.5.8.1 Experiment 1: I-V characteristics of BPW34 diode: (28-6-2021)

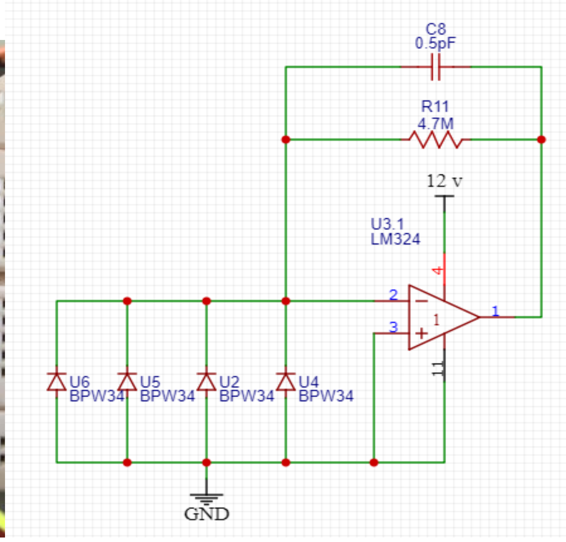
The I-V characteristics of a photodiode are studied with the same manner of an ordinary diode. The diode is studied under reverse voltage of 12 V to ensure its ionization and with a 100 ohms resistor to measure the current across it using an oscilloscope. A flash light from the phone is used as luminance input.

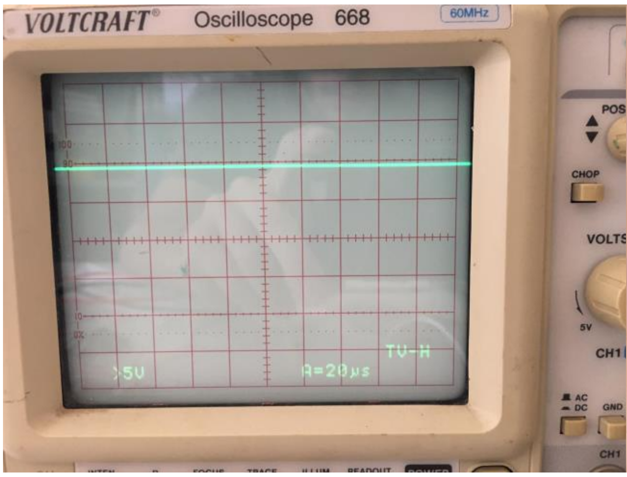


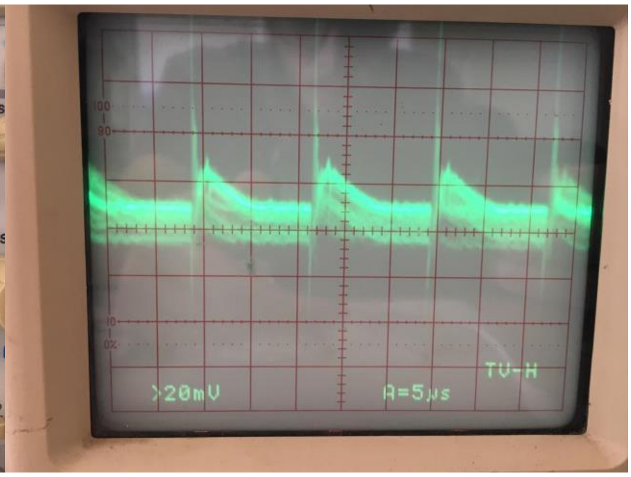
8.5.8.2 Experiment 2: Preamplifier Design: (28-6-2021)

This experiment uses an ordinary feedback integrator circuit with pulling down the non-inverting pin of the op amp. The charge op amp fall time is approximated to be 23.5 micro-seconds (5 pF * 4.7 M).





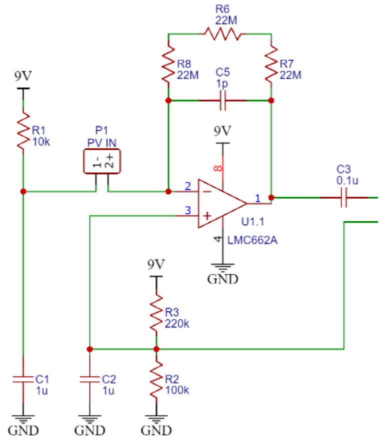


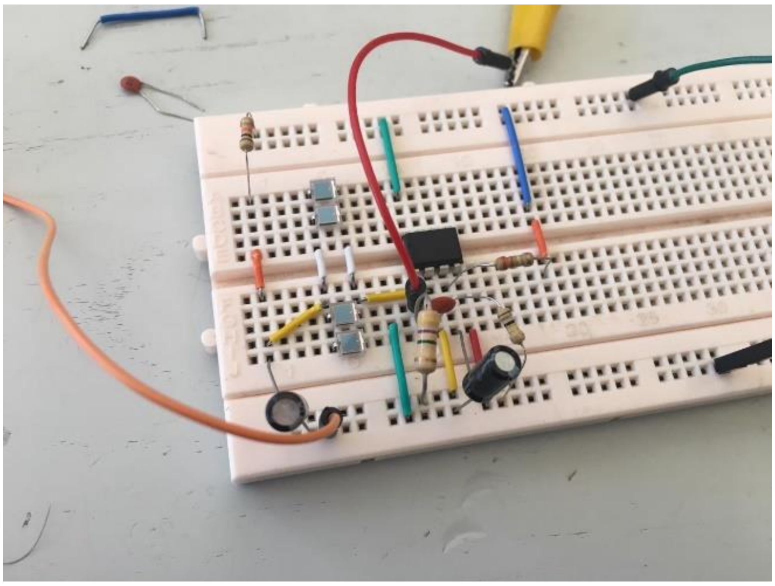


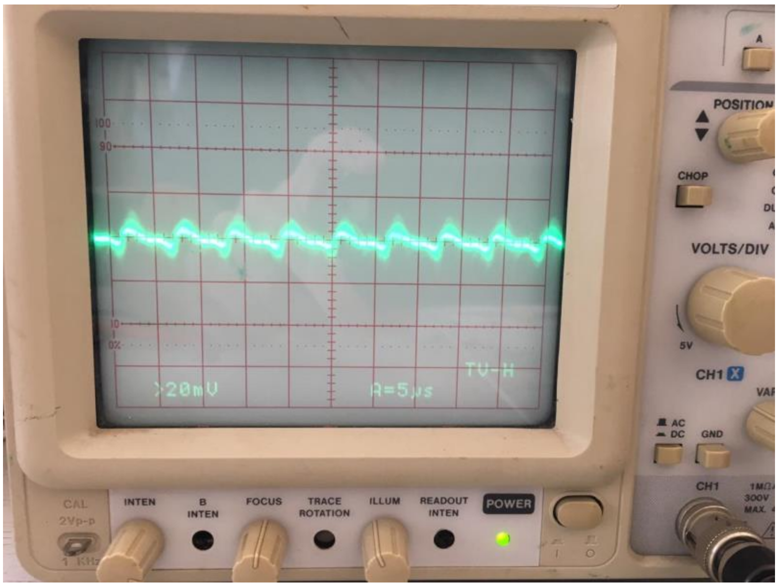
However, the output is constant (left figure) no matter how much light is presented at the photodiodes. After some modifications, a 10-pF capacitor is mounted between the photodiodes and the operational amplifier and it can be seen from the right figure above a nearly charging and discharging signal shape. **(FAILED)**

8.5.8.3 Experiment 3: Preamplifier Design: **(29-6-2021)**

Another experiment was done by presenting a bias voltage for the photo diodes by connecting a voltage divider network into the non-inverting operational amplifier pin. The output can still be seen as very noisy. **(FAILED)**



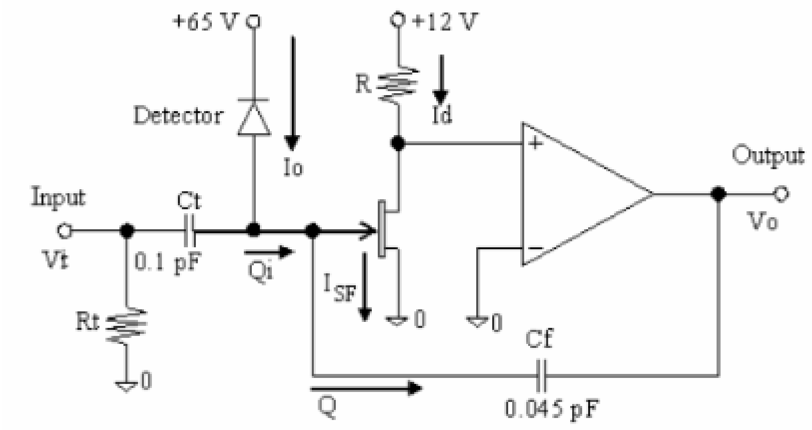




8.5.8.4 Conclusion: Preamplifier Design: (2-7-2021)

The previous experiments gave unsatisfactory results due to several reasons, such as:

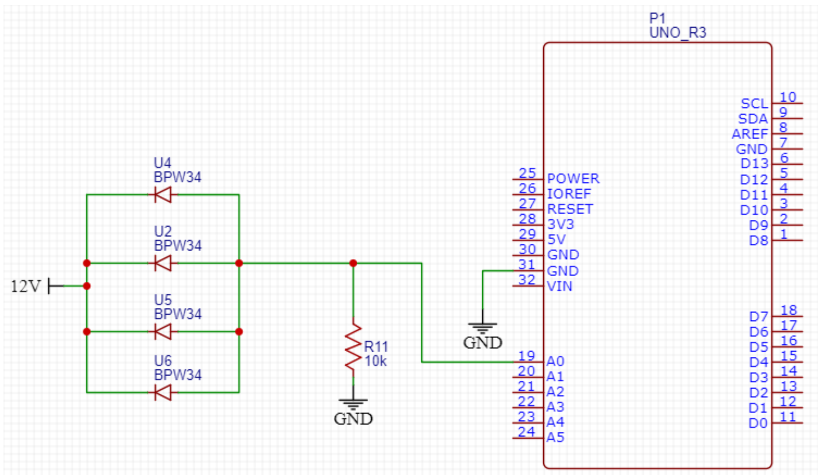
- A JFET transistor should be used at the output of the detector with low capacitance junction to collect the charges from the detector. JFET transistor suggestions can be: 2N4416, 2SK152, 2N6550.



- Another operational amplifier with JFET input should be used such as TLE2072 or OPA324.

8.5.9 Monitoring (6-7-2021)

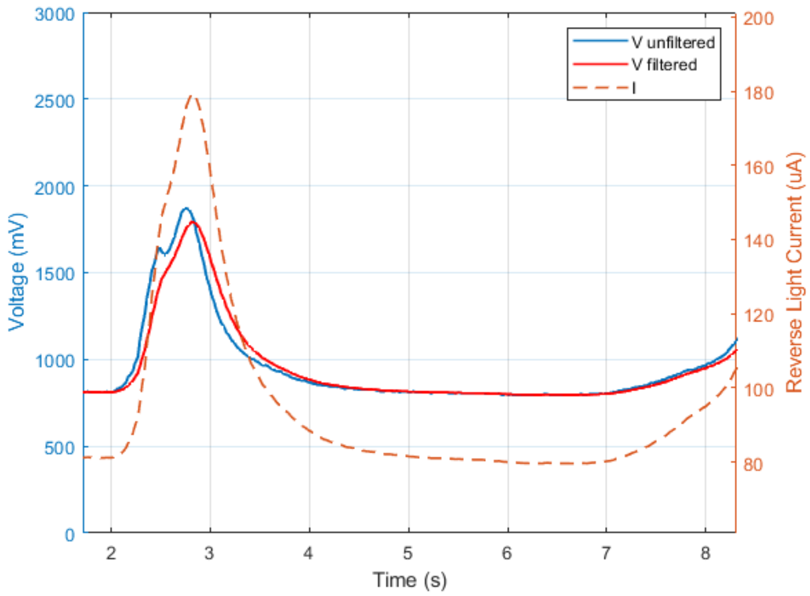
Another approach was taken in order to monitor the energy received by the photodiodes. The photodiodes in this experiment were dealt with as current sources. A resistor is connected to the reverse diodes in order to get a voltage image of the current. An Arduino Uno is used to read the voltage coming from the photo diodes through its analog pin A0 which will be then converted into current.



The nominal voltage, when the diodes are exposed to normal room lights, was approximately 750 mV. When an external light source (phone flashlight) is exposed to the diode array at a distance of approximately 8 cm, the voltage rises to 1780 mV. A complimentary filter is used to filter out measurement noises which has the following form:

$$V_{filt} = \alpha . V_{filt} + (1 - \alpha) . V_{measured} \tag{8}$$

Where, α is a wheighing constant between 0 and 1 which prefers one variable over the other, where it was chosen as 0.8. The current is calculated by dividing the output voltage by the load resistance of 10 k ohms.



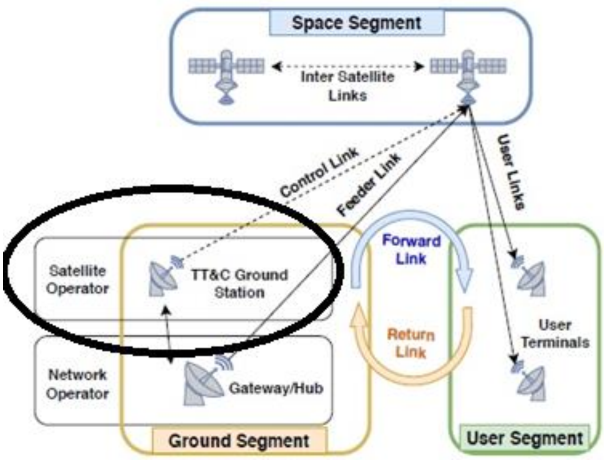
The light intensity, at this stage, can be calculated from the current graphs given by the datasheet of the BPW34 photodiodes. It is noteworthy that there are 4 diodes connected in parallel so the current is added.

8.5.10 Testing X-Ray Sensor with e-beam on cupper

You can find the full vidoe on the following link:
https://drive.google.com/file/d/1ftpkXedKj8o95gi0vmIOGuOIiWHcSMjj/view?usp=drivesdk

8.5.11 Telemetry, Tracking & Control (TT&C) Ground Station

8.5.11.1 Requirements (A DESCRIPTION OF A STANDARD SMALL SATELLITE GROUNDSTATION FOR USE BY WMO MEMBERS [4])



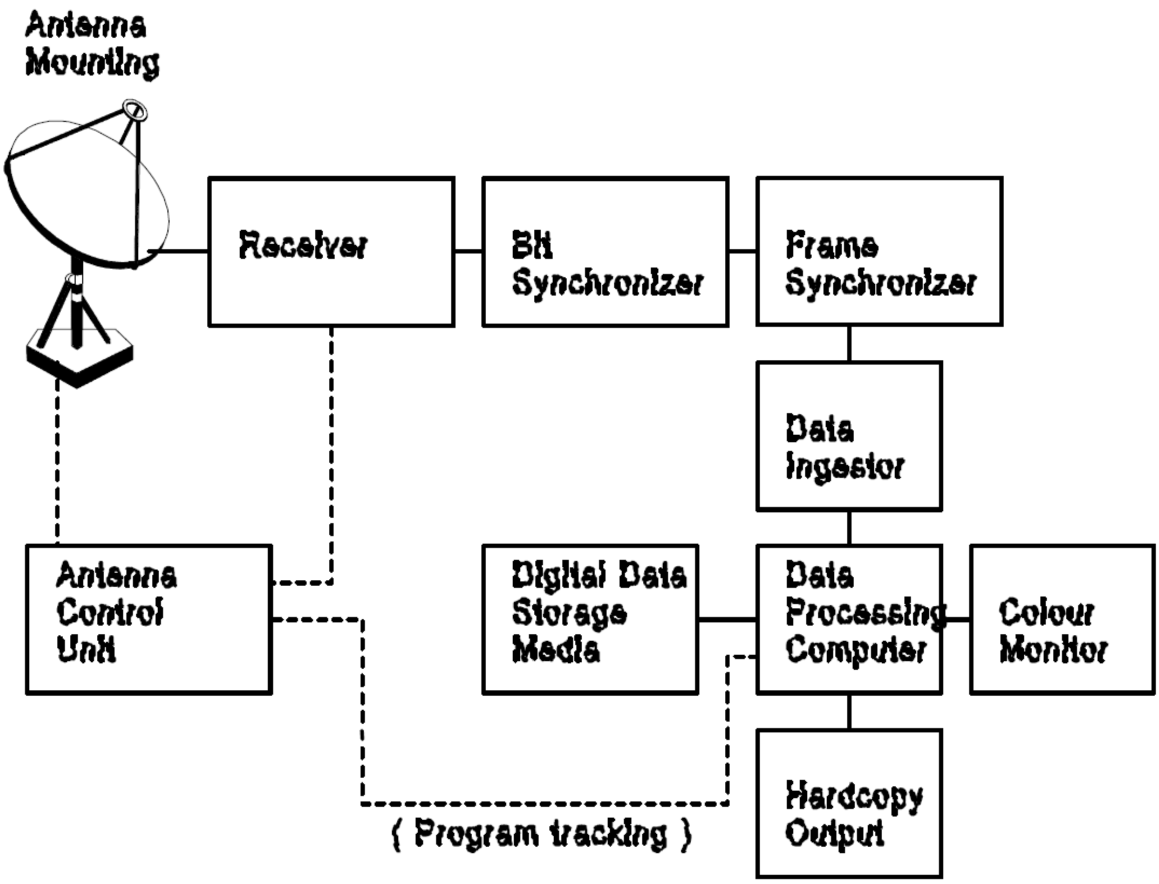


Fig.1 Block Diagram of a small satellite ground station

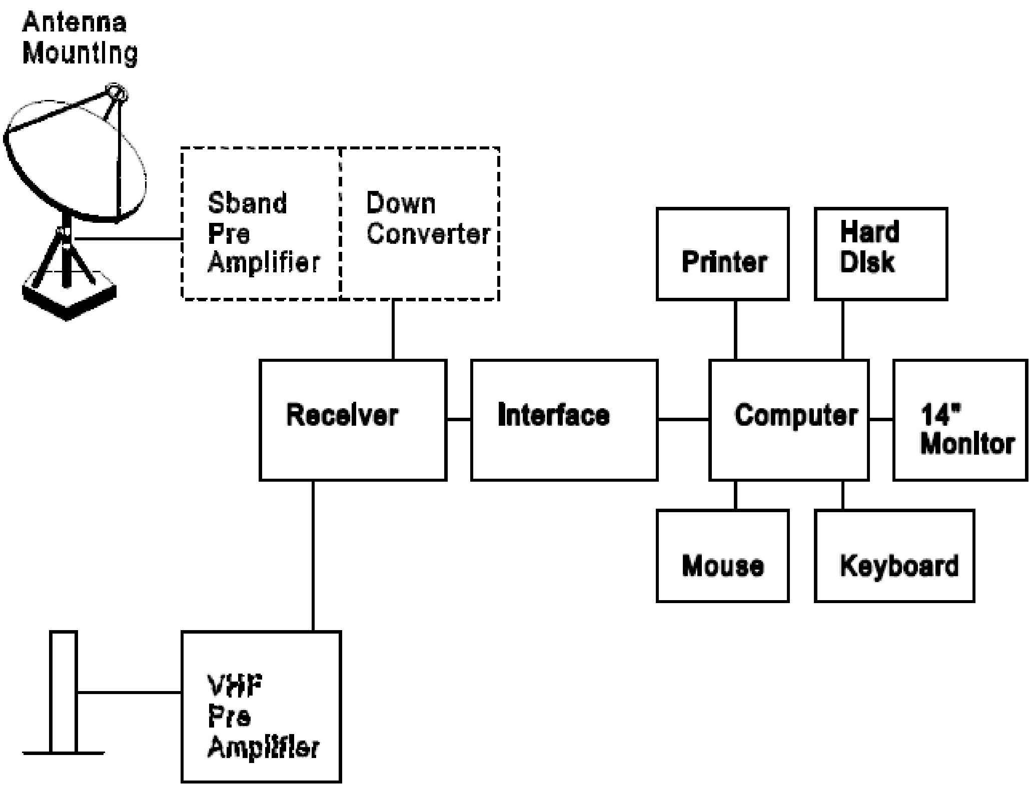


Fig.2 APT/WEFAX Receiving and Processing Station

5.1.1 S-band parabolic antenna

5.1.1.1	Diameter of antenna:	: 1.5 m
5.1.1.2	Gain	: 26.0 db
5.1.1.3	Beamwidth	: 8.1°
5.1.1.4	Frequency	: 1691.0 MHz
5.1.1.5	Polarization	: linear
5.1.1.6	Mount	:fixed, variable adjust, elevation 90°, azimuth (20° - 70°)
5.1.1.7	VSWR	: 1.5
5.1.1.8	Impedance	: 50 Ω

5.1.2 S-band preamplifier and down converter

5.1.2.1	RF input	: 1691.0 MHz
5.1.2.2	IF output	: 137.5 MHz or other
5.1.2.3	Band width	: 6 MHz
5.1.2.4	IF/RF gain	: 30 db
5.1.2.5	Noise figure	: 1.5 db
5.1.2.6	Stability	: 5×10^{-6}
5.1.2.7	Impedance	: 50 Ω
5.1.2.8	Cable length	: 60 m

5.1.3 VHF antenna

To reduce the price and for ease of maintenance, an OMNI directional non-tracking antenna is recommended. An OMNI directional non-tracking antenna must be able to receive data above an elevation of 5°. This requirement will reduce interference while maximizing the possibility for coverage of synoptic scale meteorological phenomena.

<u>OMNI directional antenna</u>

5.1.3.1	Frequency	: 137.5 MHz
5.1.3.2	Polarization	: right hand circular
5.1.3.3	Impedance	: 50 Ω
5.1.3.4	VSWR	: 2.1 max
5.1.3.5	Gain	: 3 dbi
5.1.3.6	Beamwidth	: 180°

Depending on the user's situation and requirements, an omni-directional antenna may not be sufficient for proper APT reception. Under these circumstances, the use of a directional antenna, such as a crossed Yagi, would provide higher performance and greater coverage. Note that use of program tracking and other antenna pointing methods would be required. The following information describes an alternative to the OMNI directional antenna described in sections 5.1.3.1 through 5.1.3.6.

<u>Directional antenna (Yagi)</u>

5.1.3.7	Centre frequency	: 137.5 MHz
5.1.3.8	Polarization	: right hand circular
5.1.3.9	VSWR	: 2.0 max
5.1.3.10	Gain	: 20 dBi or greater
5.1.3.11	Beamwidth	: 20 degrees at 20 dBi
5.1.3.12	Mount	: Elevation over azimuth
5.1.3.13	Program track	: Program track

5.1.4 VHF Preamplifier

5.1.4.1	Centre frequency	: 137.5 MHz
5.1.4.2	Gain	: 30 db
5.1.4.3	Noise figure	: 2 db
5.1.4.4	Installation	: in antenna base

5.1.5 Receiver

5.1.5.1	Type	: FM phase lock loop
5.1.5.2	Input frequency	:Switch selectable crystals for reception of APT and WEFAX
5.1.5.3	IF bandwidth	:50 KHz and 30 KHz --switch selectable
5.1.5.4	Noise figure	: 5 db

5.1.6 Computer

5.1.6.1	486 computer with TVGA card (1024 x 768 resolution)	
5.1.6.2	Colour monitor	:14"
5.1.6.3	Hard disk	: 120 MB
5.1.6.4	Memory	: 4 MB
5.1.6.5	Clock : 33 MHz	
5.1.6.6	Keyboard	
5.1.6.7	Mouse	
5.1.6.8*	Printer (optional)	

5.1.7 Outdoor environment

5.1.7.1	Temperature	: -40° C - +50° C
5.1.7.2	Humidity	: 98%
5.1.7.3	Wind	: operational 20 m/s, survival 35 m/s

5.1.8 Power

5.1.8.1	110v/220v +- 10%
5.1.8.2	50Hz/60Hz

5.2 Specification requirement, for high resolution ground station

The stations should be reliable and easy to operate. The block diagram of the high resolution ground station is shown in Fig.3. If the station receives geostationary satellite data, it should be equipped with a fixed antenna. If the station receives polar orbiting satellite data then it must be equipped with a tracking antenna and antenna control unit.

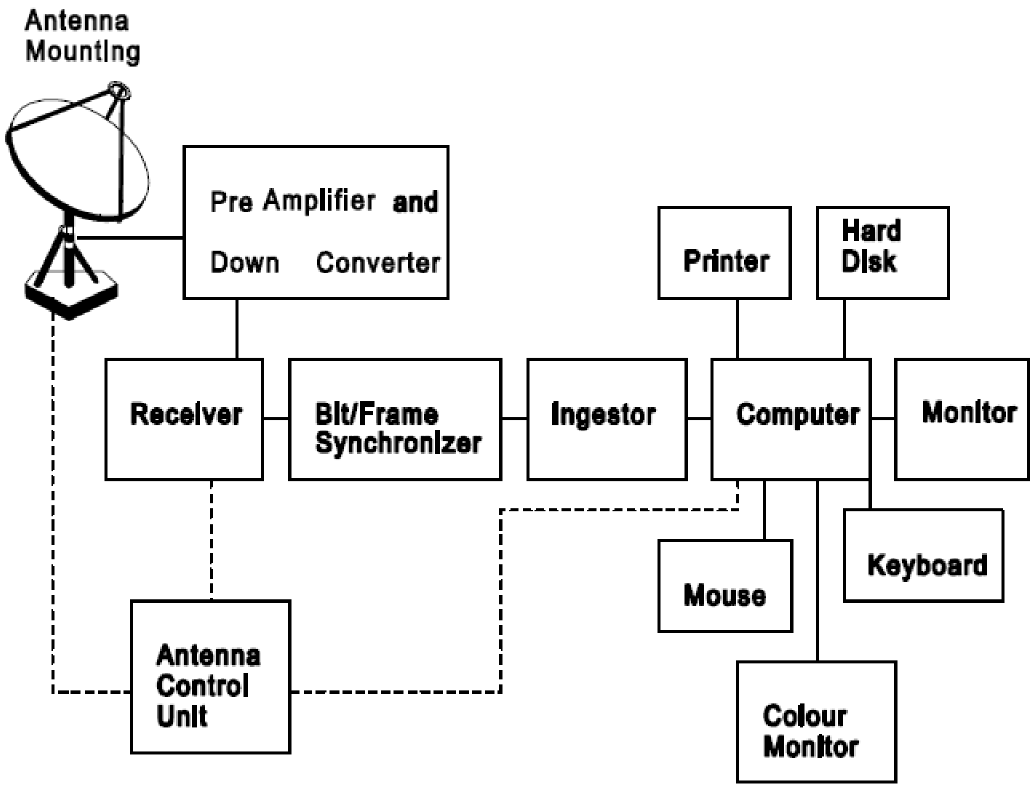


Fig.3 High Resolution Data Receiving and Processing Station

5.2.1 S-band tracking antenna and antenna control unit for receiving HRPT data

5.2.1.1	G/T merit of antenna system	:>6 db/K bit error rate is better than $1x10^{-6}$ at 5 degree elevation
5.2.1.2	Frequency	:1670~1710 MHz
5.2.1.3	Polarization	: RH, LH
5.2.1.4	Impedance	: 50/75 Ω
5.2.1.5	VSWR	: 1.5
5.2.1.6	Antenna mounting	: azimuth-elevation
5.2.1.7	Tracking coverage	: full geometric coverage including overhead passes (Z-pass), good performance of Z-pass tracking

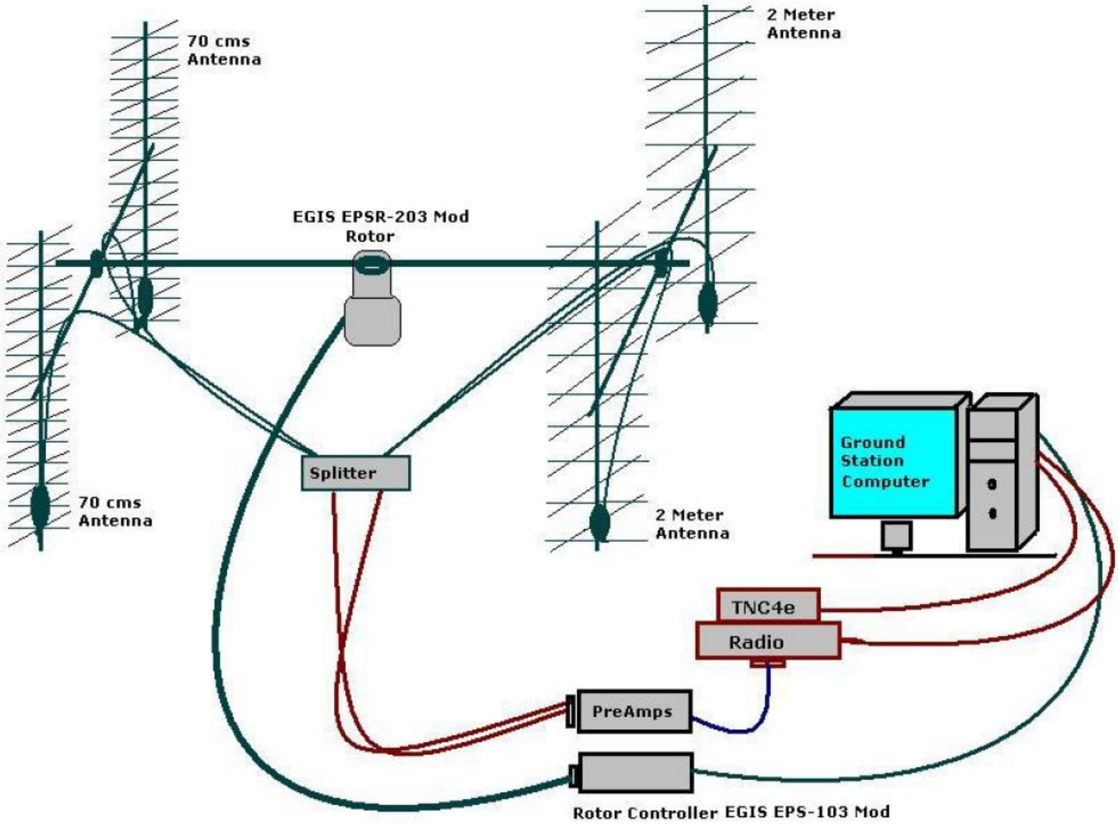
5.2.2 S-band fixed antenna for receiving geostationary satellite

5.2.2.1	G/T merit of antenna system	:G/T depending on which satellite to be received, G/T must guarantee that bit error rate is less then 10^{-6} in the worst case
5.2.2.2	Frequency	: 1670~1710 MHz
5.2.2.3	Polarization	: linear
5.2.2.4	Mounting	: fixed, variable adjust elevation 90, azimuth 20° ~70°
5.2.2.5	VSWR	: 1.5
5.2.2.6	Impedance	: 50/75 Ω

8.5.11.2 As reference a ground station design from 2007 ([3]):

Abstract of [3] :

The CubeSat satellite ground station at the University of Wuerzburg is built with "commercial of the shelf" low cost amateur radio hardware. It opens up opportunities for students to receive and operate CubeSats, including Wuerzburgs UWE-1. As any other satellite ground station, it is built up on essential hardware, as there are Antenna, Antenna Rotator, Radio, Modem and Computers. Furthermore software is used to afford basic control over the ground station and provide tracking abilities to follow a satellite passing over the ground station.



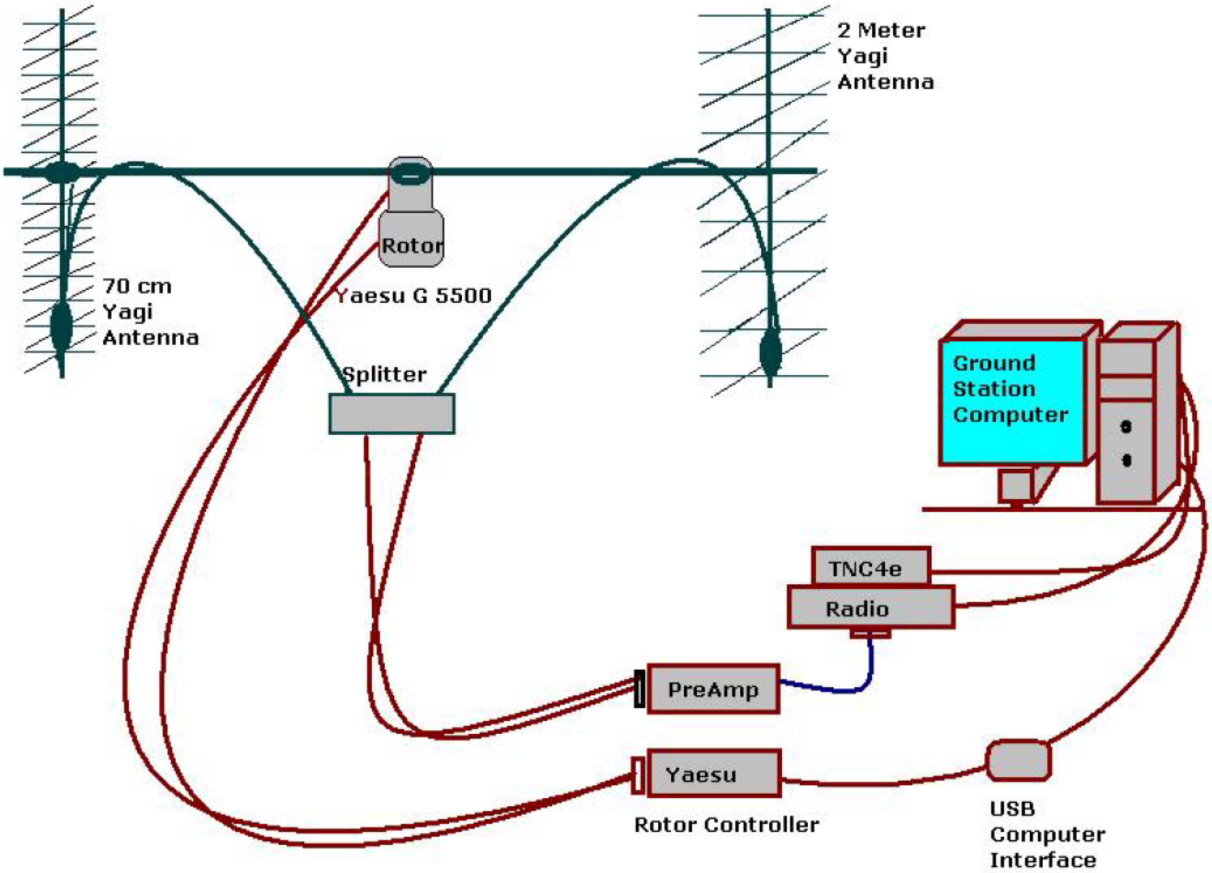
Schematic of the Old Ground Station design at University of Wuerzburg, Germany.

Figure 1.1: Old design of the Ground Station.

Ground Station structural study

Related CubeSat.	**University.**	**Antenna Rotor.**	**Rotor Computer Controller & Tracking Software.**
Delfi-C3	Delft University of Technology	Yaesu G-5500	NOVA for Windows
AAU Cubesat	Aalborg University of Technology	Yaesu G-5500	Predict
CubeSat	University of Arizona	Yaesu G-5500	NOVA for Windows with Uni_Trac
PolySat	California Polytechnic State University, U.S.	Yaesu G-5500	SatPC32
Cubesat	University of Tokyo, Japan	Elevation Rotator ERC5A (Creative Design) Azimuth Rotator RC5A-3 (Creative Design)	Orbital calculation software (Virtual Ground Station 3)
Cubesat	TU-Berlin	Yaesu G-5500	SatPC32 and ARSWIN
---	Kagawa University, Japan	ERC-5A (El) And RC5B-3P (Az)	Satellite Tracker is RAC825
--- ---	Kyusyu University Ground Station. Kyushu Institute of Technology.	EMOTATOR EV-800D Yaesu G-5500	No avail info GS-232A
---	Nara National College of Tech, Japan	Yaesu G-5500	GS-232A and Nova for Windows.
---	Soka University Ground Station Unit.	Yaesu G-550 (El) And Yaesu G-2300DXA (Az)	GS-232B

Figure 1.2: Ground Station Structural Study.



Schematic of the new Ground Station design at University of Wuerzburg, Germany.

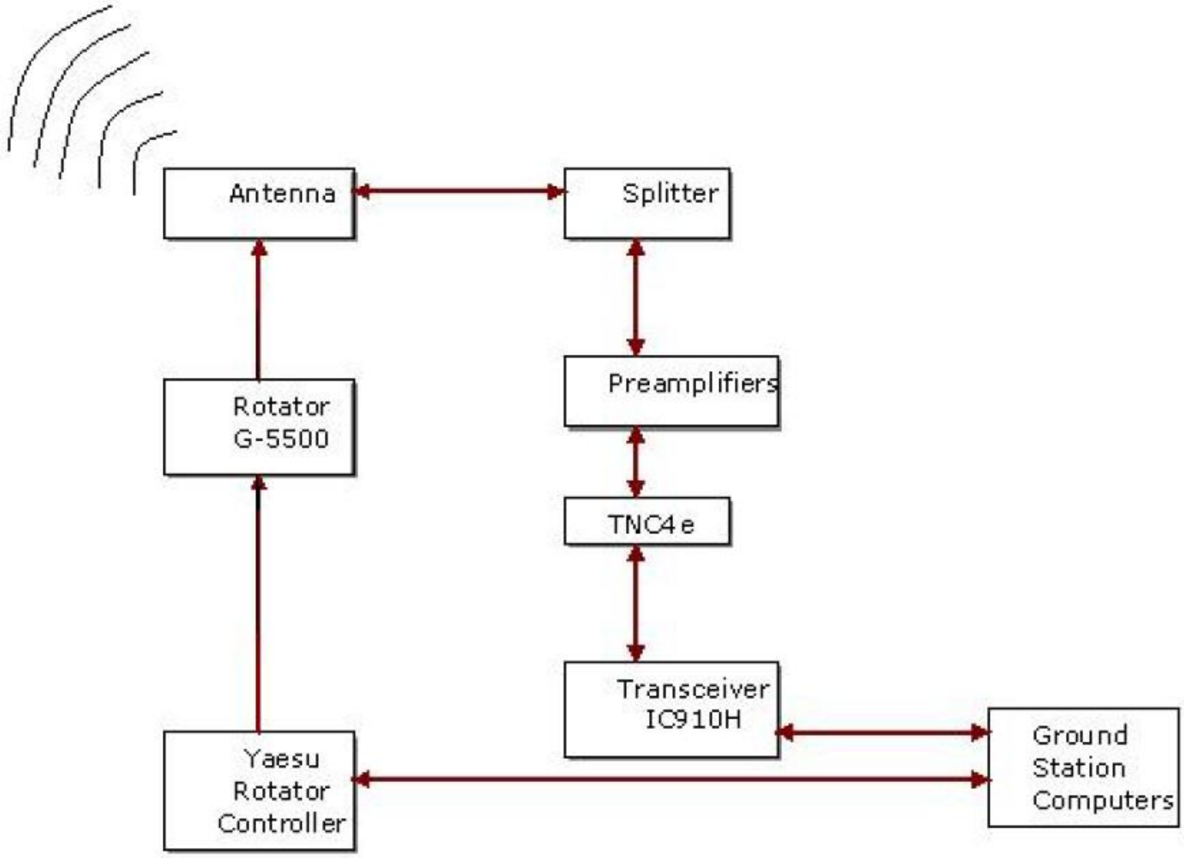
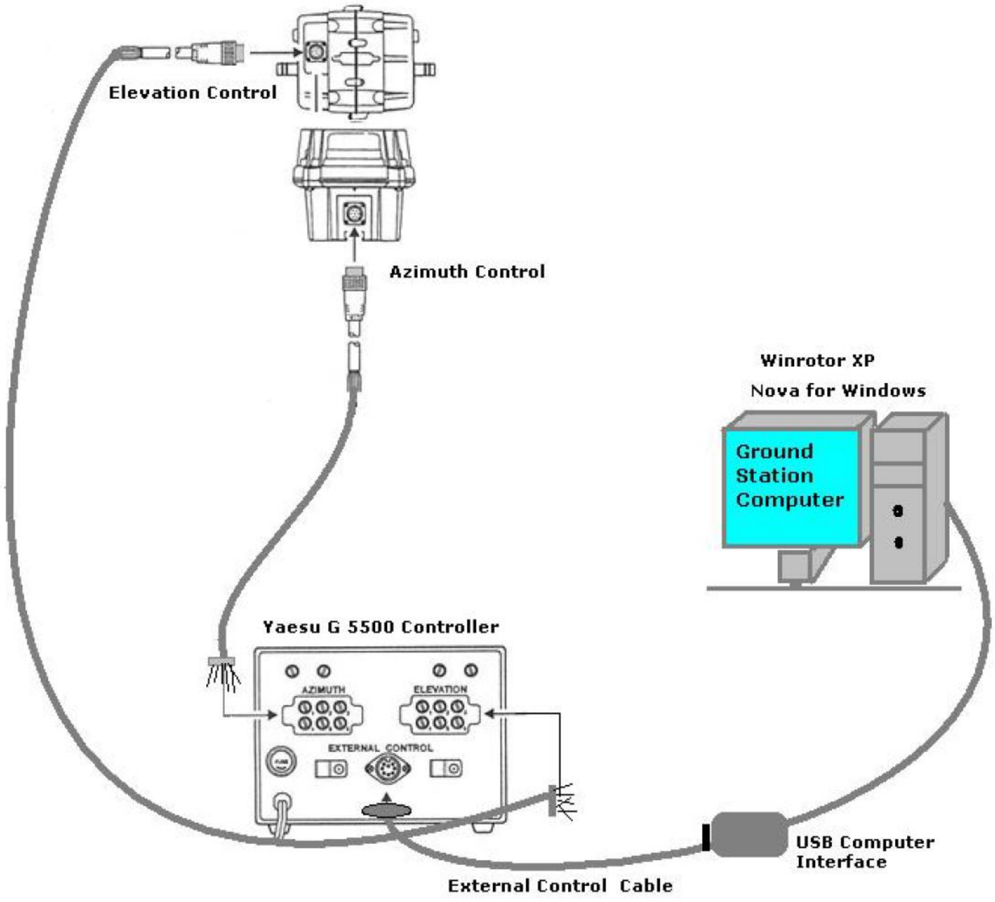


Figure 1.4: Block Diagram of the Ground Station.

Altitude	310 meters
Tower	Hummel Teletower Jumbo III
Operating frequencies	2m and 70 cms amateur bands
One 2 meter antenna	M2 2MCP22
One 70 cms antenna	M2 436CP42U/G
Antenna rotator and Controller	Yaesu G-5500
Rotator-Computer Interface	WinRotor
Rotator-Computer Interface driver	WinRotor XP
Radio	TNC4e
Polarisation switch	WiMO
Preamplifiers	LNA-145, SLN Series
Tranceiver	ICOM IC-910H
Power Supply	Microset 13.5 Volts
Two PCs	Fujitsu Siemens

Table 2.1: Hardware Specifications Table.



Rotor-Controller-Computer Interfacing.

8.5.12 Graphical User Interface

Installing Nova for Windows

(a) Insert the Nova for Windows CD into the CD-ROM drive of your computer.

(b) If the setup program doesnt start automatically, click on the Start button (lower left corner of the desktop).

(c) Click on Run.

(d) In the file name box, type **Setup.EXE**.

(e) Follow the directions in the Nova for Windows Setup.

Important:

Be sure to enter the serial number carefully. Serial number must include the NLD- prefix.

First step is to set the type of Map. In the screenshots shown below "Large Rectangular Map" is selected for convenience.

To choose the new map setting the path is-
"Views" then "Configure current view" and then Choose "Map display" and "Map Size".
Refer figure 2.11.

Second step is to set the position of the Ground Station in "Nova for Windows". The path is- "Setup" and then "Observers".

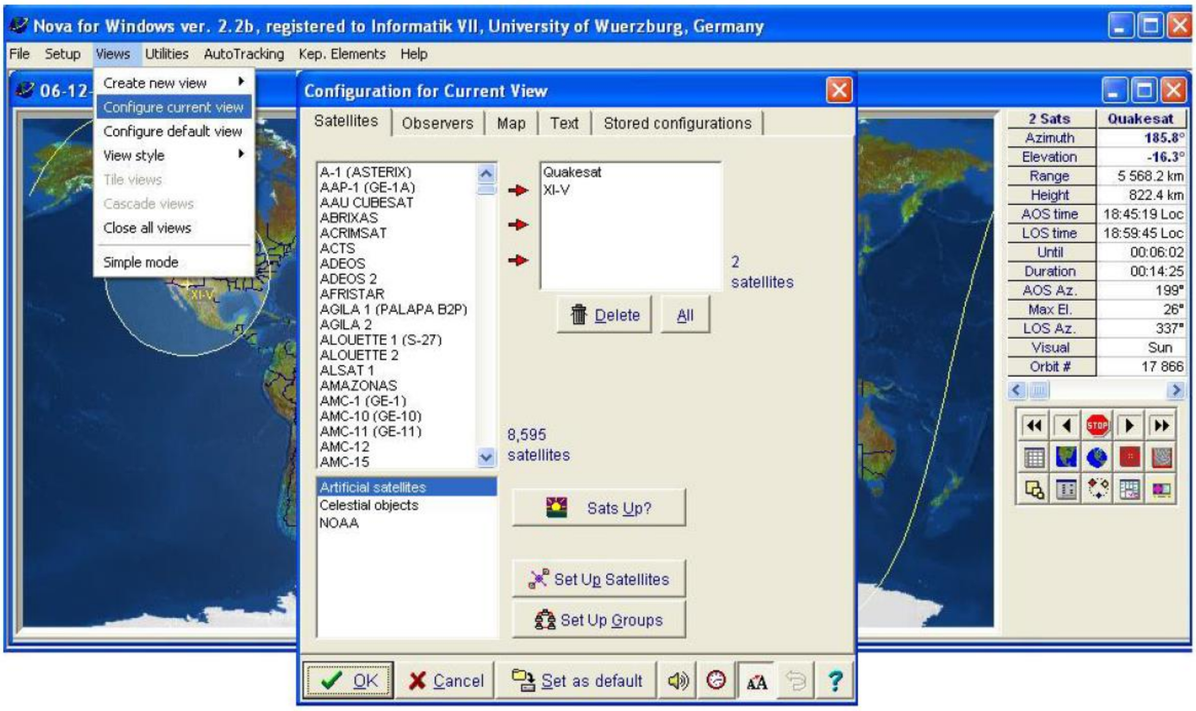


Figure B.1: Nova for Windwos - Configuring View.

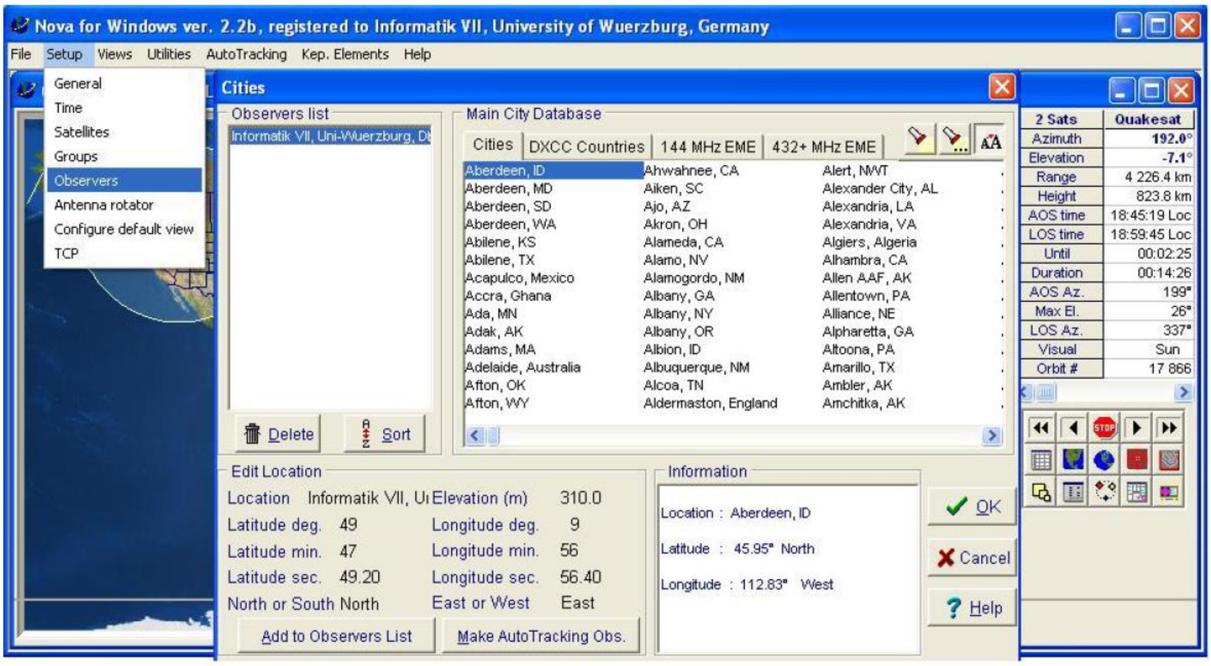


Figure B.2: Nova for Windows - Configuring Observer.

In our case it is:
Location: "*Informatics VII, Uni-Wuerzburg, Germany.*"
Elevation is of *310 meters*.
Latitude is *49 degrees 47 minutes 49.20 seconds North*.

Longitude is *9 degrees 56 minutes 56.40 seconds East.*
Refer figure 2.12.

Third step is to check the availability of the specific satellite from the Satellite Editor in the database of "Nova for Windows".
In this editor, new satellite names and its Keplerian elements can also be added. Also "Update Keplerian Elements" button provides the on-line update.
The path is "Setup" and then "Satellites".
Refer figure 2.13.

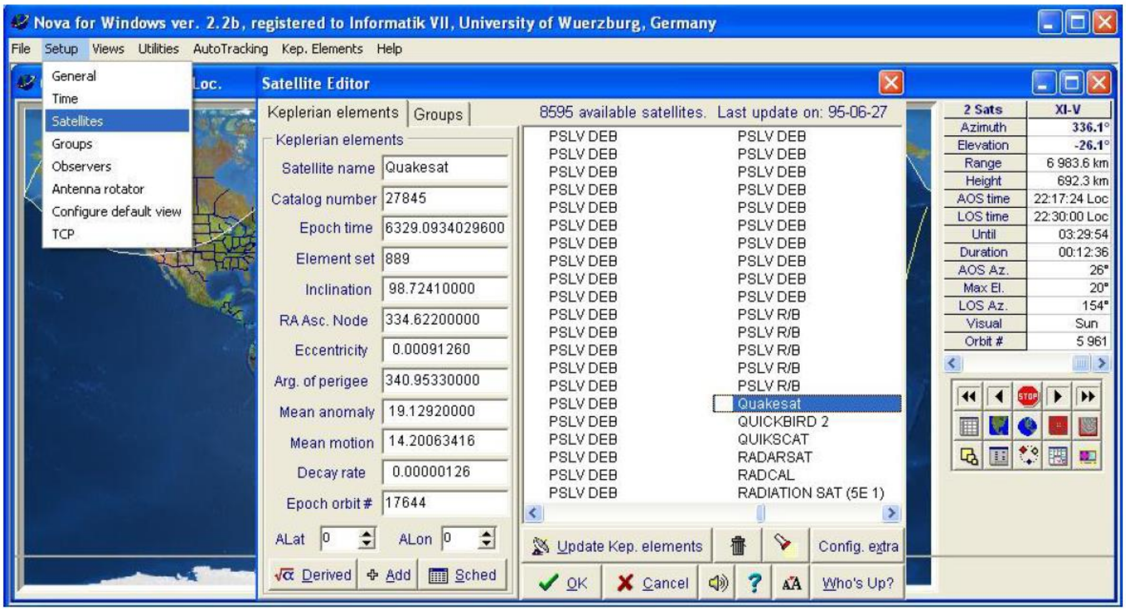


Figure B.3: Nova for Windows - Configuring Satellites.

To update Keplerian elements or to get related help click on "Kep Elements". Refer figure 2.14.

Fourth step is to choose the "Current View" in order to see Satellite and Observer (Ground Station position) all together.
This provides a feature of selecting multiple Satellites and Observation points on the map at the same time.
The path is "Views" and then "Configure current view" and then "Satellites" or "Observes" or "Map" or "Text". Refer figures 2.15 and 2.16.

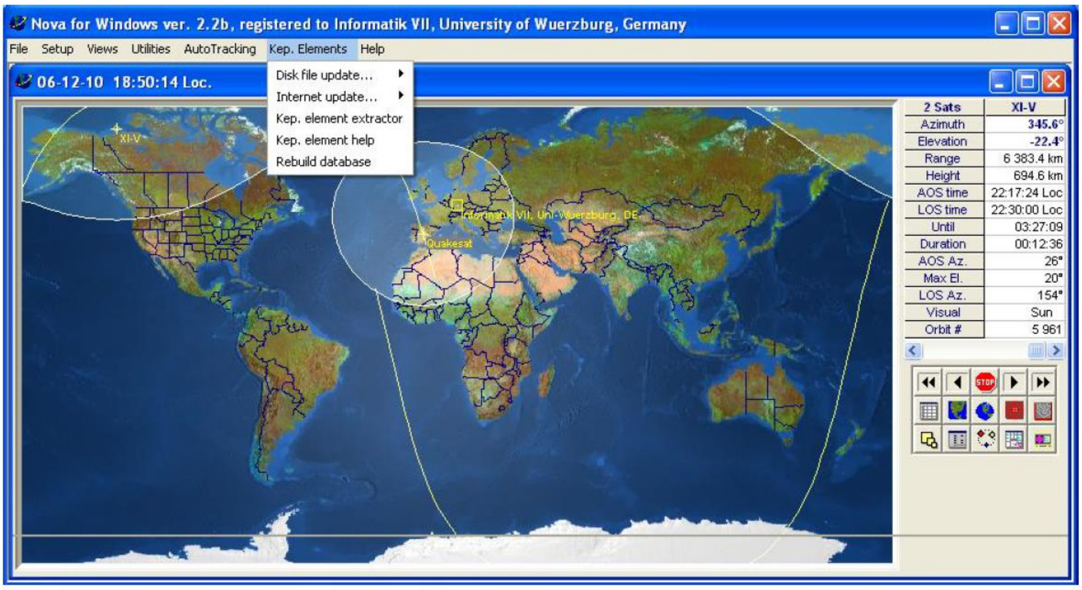


Figure B.4: Nova for Windows - TLE Updation

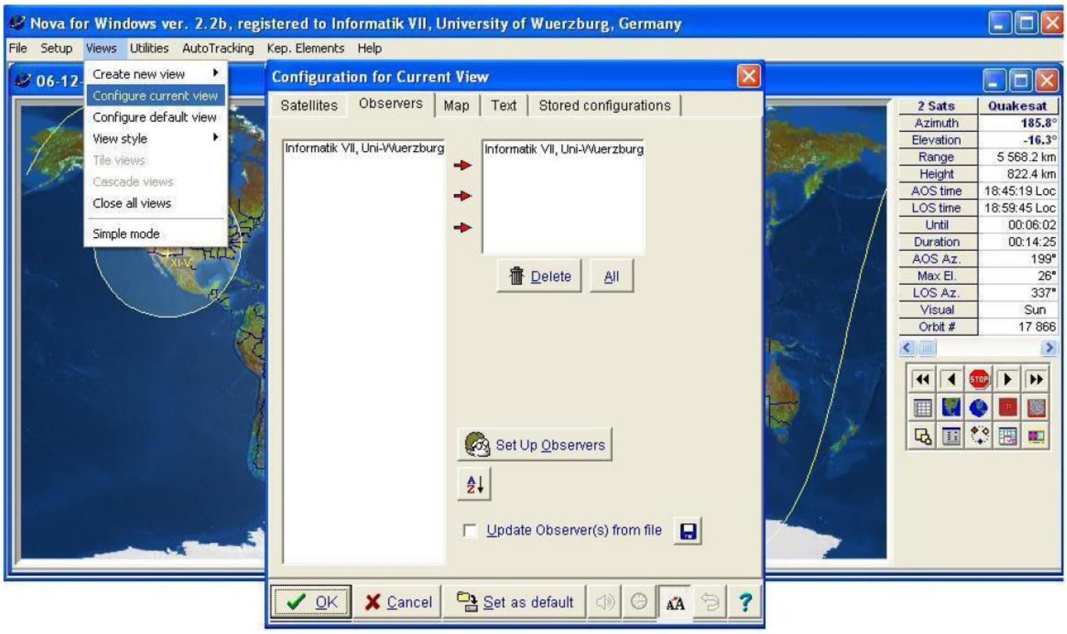


Figure B.5: Nova for Windows - Current View Observer.

On the Map, Footprint of the satellite/s and the Ground Station's position/s can be easily found. Refer figure 2.17.
On the Right hand side of the screen, Real-time text data of the con-

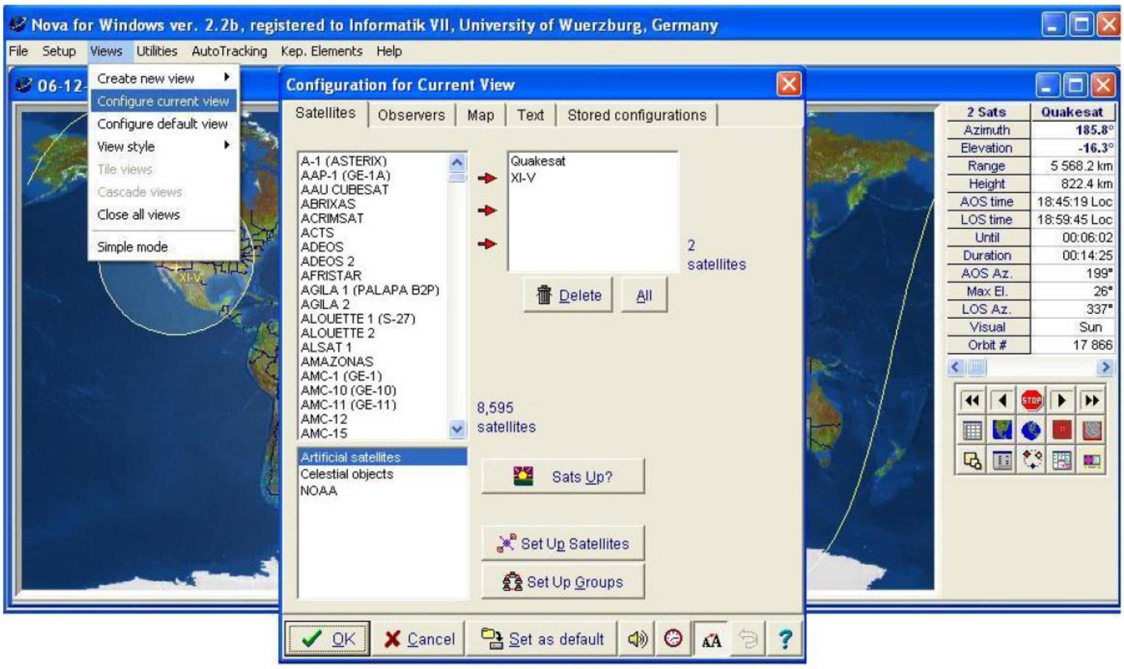


Figure B.6: Nova for Windows - Current View Satellite.

cerning satellite is available. The number of columns in the real-time text window depends on the number of satellites in the view.

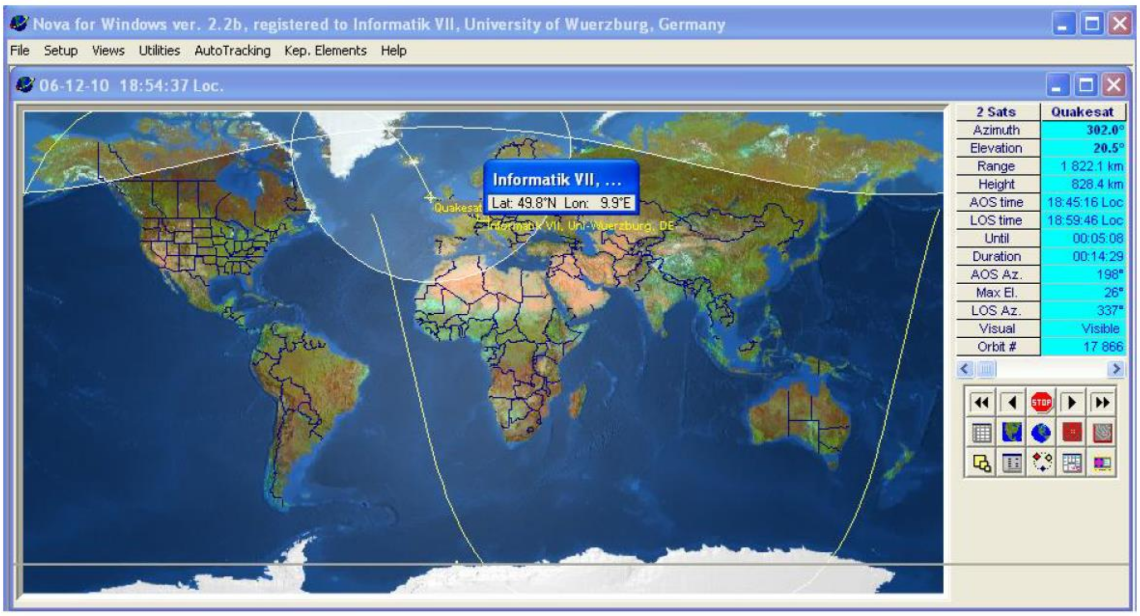


Figure B.7: Nova for Windows - Satellite Footprints.

Satellite Script.

"Satellite Script" features the prediction of the flyby time of the satellite or satellites over a particular Observer (Ground Station) up to 48 hours in advance.
This also enables "automatic script tracking". Refer figure 2.19.

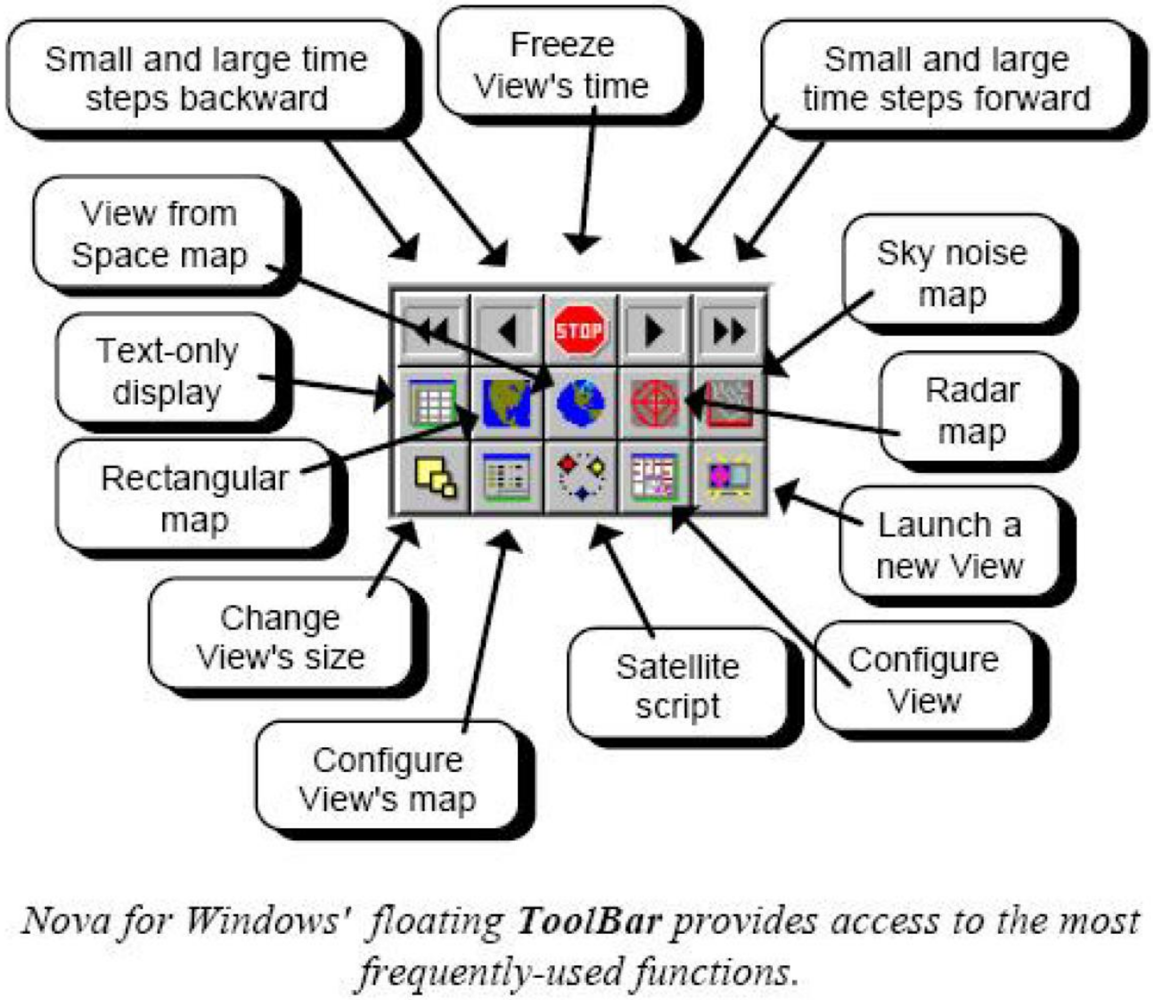
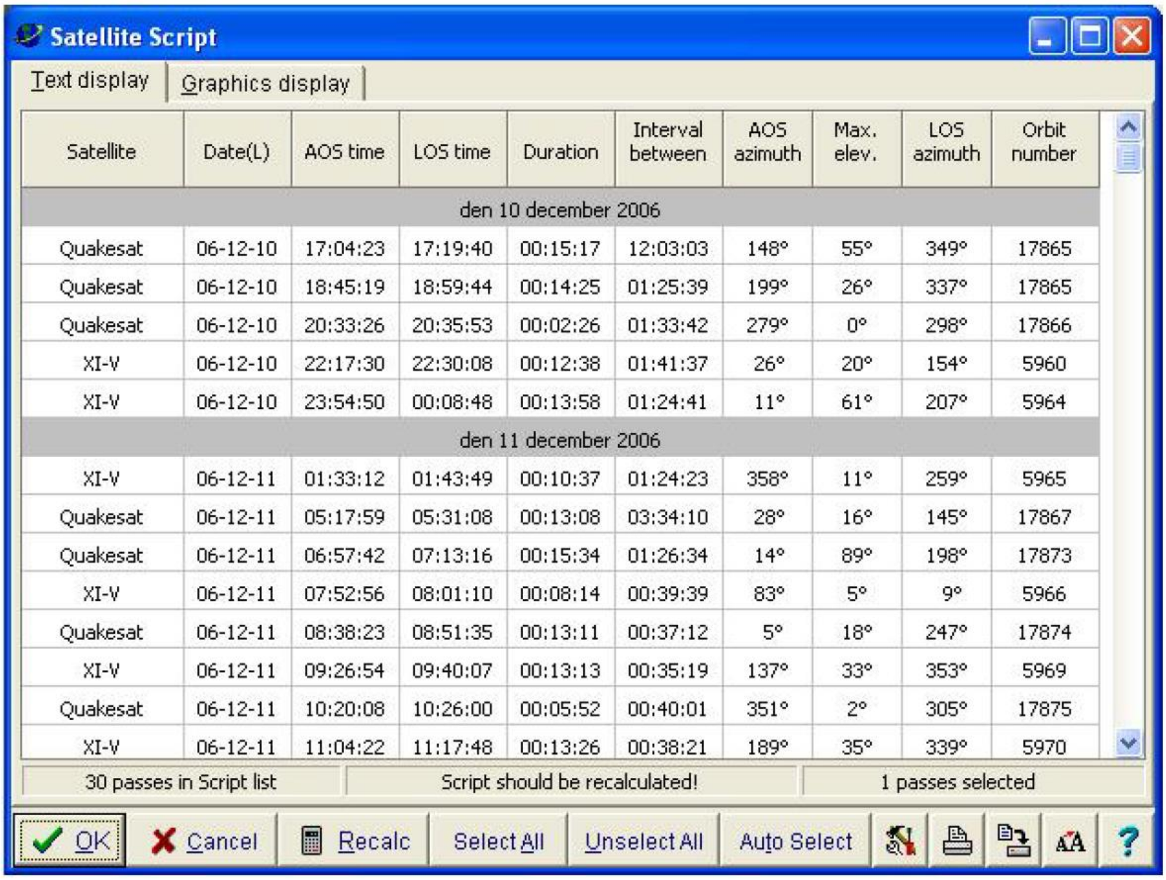


Figure B.8: Nova for Windows - Floating Toolbar.



Satellite	Date(L)	AOS time	LOS time	Duration	Interval between	AOS azimuth	Max. elev.	LOS azimuth	Orbit number
den 10 december 2006									
Quakesat	06-12-10	17:04:23	17:19:40	00:15:17	12:03:03	148°	55°	349°	17865
Quakesat	06-12-10	18:45:19	18:59:44	00:14:25	01:25:39	199°	26°	337°	17865
Quakesat	06-12-10	20:33:26	20:35:53	00:02:26	01:33:42	279°	0°	298°	17866
XI-V	06-12-10	22:17:30	22:30:08	00:12:38	01:41:37	26°	20°	154°	5960
XI-V	06-12-10	23:54:50	00:08:48	00:13:58	01:24:41	11°	61°	207°	5964
den 11 december 2006									
XI-V	06-12-11	01:33:12	01:43:49	00:10:37	01:24:23	358°	11°	259°	5965
Quakesat	06-12-11	05:17:59	05:31:08	00:13:08	03:34:10	28°	16°	145°	17867
Quakesat	06-12-11	06:57:42	07:13:16	00:15:34	01:26:34	14°	89°	198°	17873
XI-V	06-12-11	07:52:56	08:01:10	00:08:14	00:39:39	83°	5°	9°	5966
Quakesat	06-12-11	08:38:23	08:51:35	00:13:11	00:37:12	5°	18°	247°	17874
XI-V	06-12-11	09:26:54	09:40:07	00:13:13	00:35:19	137°	33°	353°	5969
Quakesat	06-12-11	10:20:08	10:26:00	00:05:52	00:40:01	351°	2°	305°	17875
XI-V	06-12-11	11:04:22	11:17:48	00:13:26	00:38:21	189°	35°	339°	5970

Figure B.9: Nova for Windows - Satellite Script.

Frequency display.
It also displays the Uplink and Downlink Frequencies, with the Doppler value for the particular selected satellite.
To check this, the path is "Utilities" and then "Frequency display".
Refer figure 2.20.

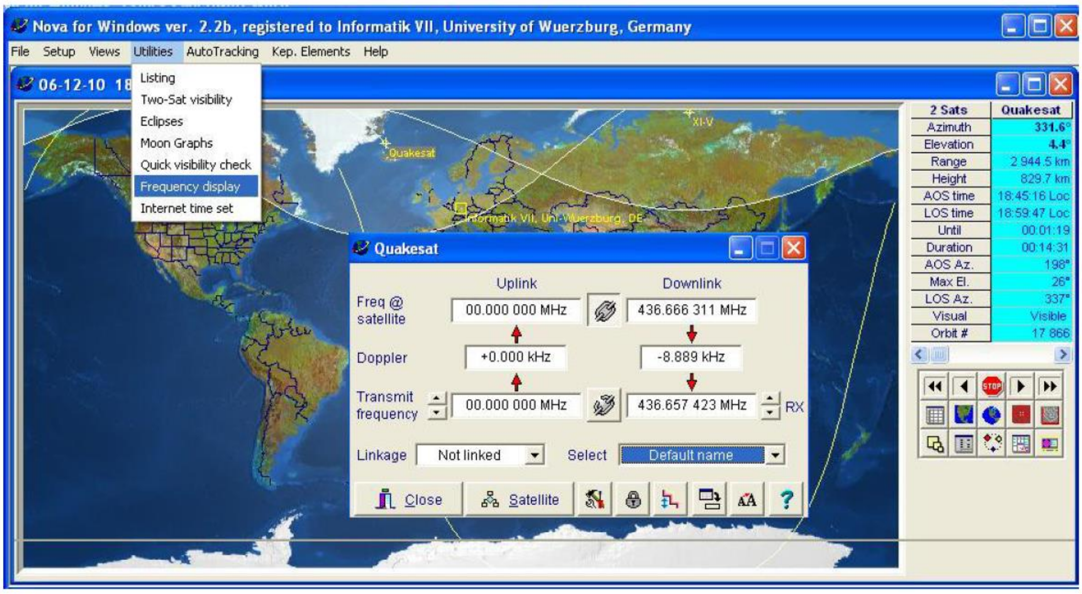


Figure B.10: Nova for Windows - Frequency Display.

To enable **Auto-Tracking** with "Nova for Windows", the first step is to select the type of Antenna Rotator from the Rotator Interface list. The path is "AutoTracking" and then "Antenna Rotator Setup" and then "Interface".
Select the Rotator Interface from the available list.
For the Ground Station at Informatics VII, University of Wuerzburg, "WinRotor" is the Rotator Interface.

For **Yaesu G-5500** azimuth rotator range is 0 to 360 degrees and elevation rotator range is 0 to 180 degrees. Refer figure 2.21.

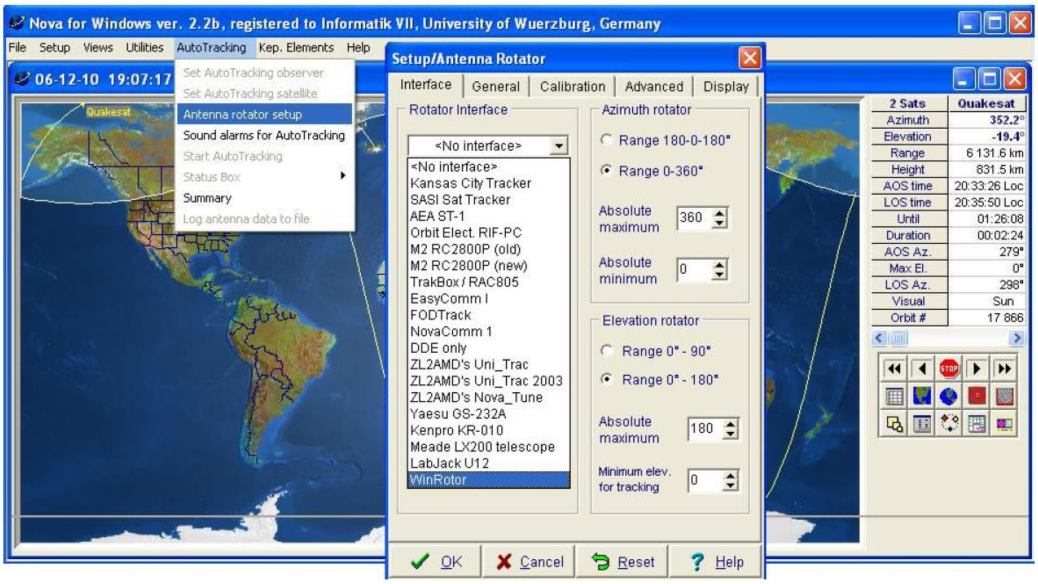


Figure B.11: Nova for Windows - Antenna-Rotator Setup.

More ***help*** regarding "Nova for Windows" can be available from "help" of the display window or please refer its detailed brochure. Refer figure 2.22.

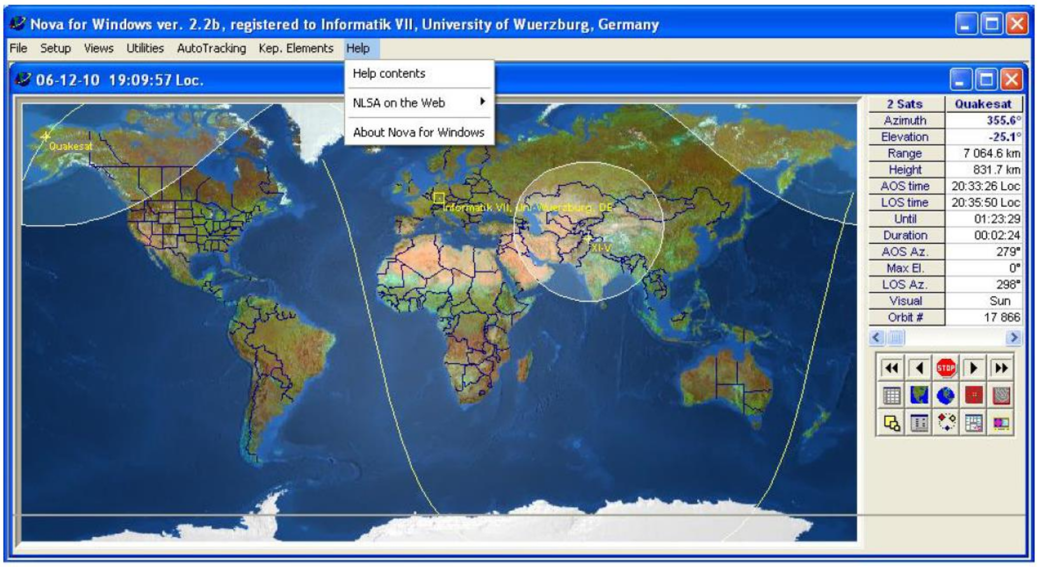


Figure B.12: Nova for Windows - Further Help.

Test Results

Eleven tests are documented in a duration of 10 days from 08 Dec 2006 to 18 Dec 2006. Testing summary is as follows:

08 December 2006	
AOS Time	16:04:08 Local Time
LOS Time	16:17:42 Local Time
Duration	00:13:33 hrs
AOS Azimuth	118 degrees
Maximum Elevation	21 degress
LOS Azimuth	355 degress

Table C.1: Testing - Satellite Script 08Dec2006.

Number of Beacons received : 3.

tnc4e2: fm KD7OVB to QST ctl UI pid=BB len 255 16:10:18
tnc4e2: fm KD7OVB to QST ctl UI pid=BB len 255 16:10:28
tnc4e2: fm KD7OVB to QST ctl UI pid=BB len 255 16:10:57

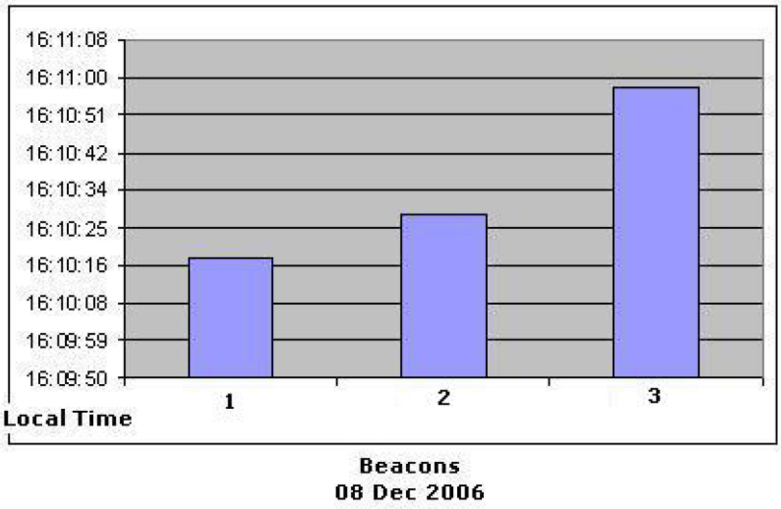


Figure C.1: Test Beacons on 08Dec2006.

11 December 2006	
AOS Time	15:08:46 Local Time
LOS Time	15:19:13 Local Time
Duration	00:10:26 hrs
AOS Azimuth	87 degrees
Maximum Elevation	8 degress
LOS Azimuth	1 degress

Table C.2: Testing - Satellite Script 11Dec2006.

11 December 2006	
AOS Time	16:45:23 Local Time
LOS Time	17:00:18 Local Time
Duration	00:14:55 hrs
AOS Azimuth	139 degrees
Maximum Elevation	40 degress
LOS Azimuth	351 degress

Table C.3: Testing - Satellite Script 11Dec2006.

8.6 Research plan to improve SDR communication system

- Research plan to improve SDR communication system

(Salih hocam lütfen:
Please put here your actual presentation slides)

1/23/2022 AS-COMSAT, Jan-Feb 2021 116

TT&C Subystem – On-Board Part

Master Thesis Task at Marmara Univ./
Prof. Dr. Salih Bayar

Tarih: 14/01/2021

T.C.
MARMARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZ ÖNERİSİ FORMU

i. ÖĞRENCİ BİLGİLERİ / *STUDENT INFORMATION*	
Ad Soyadı / *Name- Surname:*	Aytunç Polat
TC Kimlik Numarası / *Identity Number:*	12598626098
Öğrenci Numarası / *Student Number:*	525018020
Bir öğe seçin. **Dalı** / *Department:*	Anabilim – Anasanat Dalı giriş için tıklayın veya dokunun.
Programı / *Program:*	Program giriş için tıklayın veya dokunun.
Program Dili / *Language of Program:*	☐ Türkçe / *Turkish* ☒ İngilizce (Diğer / *Other*)
E-Posta / *E-Mail:*	aytuncpolat07@gmail.com
Telefon / *Phone Number:*	543-763-2013
Adres / *Address:*	Bağlarbaşı mah. Cemalbey cad. No12-14H Ablok D9 Maltepe İstanbul
Tez Danışmanı / *Supervisor:*	Dr.Öğr.Üyesi SALİH BAYAR
İkinci Tez Danışmanı / *Co-Supervisor:*	-

1/23/2022 AS-COMSAT, Jan-Feb 2021 117

TT&C Subystem – On-Board Part

Master Thesis Task at Marmara Univ./Prof.Dr. Salih Bayar

İngilizce Tez Başlığı / *Title of Thesis in English*

Software-defined Radio (SDR) based Data-Link Design for Small Mobile Platforms

İngilizce Tez Önerisi Özeti

Small mobile robot platforms designed to operate in air, land and water are today used in defense, aviation and space; It is used in medical, logistics and agricultural activities. Such mobile platforms, which are acted autonomously, semi-autonomously or manually, should generally be controlled or controlled from one or more centers. Between mobile platforms and the center, by the mission of the mobile platform or platforms; Real-time data such as motion, telemetry, video and audio must be transported safely with the help of wired or wireless communication tools insufficient channels, bandwidths and distances.

Service times are expected to be long due to the costs of the mentioned mobile robots. The software of the robots can be changed in the field to increase service times. In this way, new features can be added to these products and software errors can be corrected, even software solutions can be brought to hardware errors. In a mobile robot platform, communication equipment is also one of the important hardware that may need to be changed and developed during its lifetime. In traditional radio topologies, the fact that components such as mixer, filter, amplifier, modulator and demodulator are implemented with software is defined as Software-defined Radio (SDR). Thanks to the software-defined radio concept, wireless communication devices can be reconfigured throughout their lifetime and even during operation, so parameters such as broadcast frequency, modulation type, and bandwidth can be changed without interfering with the hardware. Besides, in today's world where data encryption methods are developing day by day, the ability to reconfigure crypto algorithms in-depth to ensure data security meets a critical need, especially in areas such as defense and aviation.

The aim of this thesis is to design a software-based data link with optimized physical dimensions that can communicate in **a high-frequency band and high data rate for use on small mobile robot platforms**. The design will consist **of two main units, a radio frequency front unit and a digital processing unit**. The radiofrequency front unit will consist of low noise amplifier (LNA), power amplifier (PA), mixer and filter groups. The digital processing unit will consist of analog to digital converter (ADC), digital to analog converter (DAC), Field Programmable Gate Array (FPGA) and memory units.

İngilizce Anahtar Kelimeler / *Keywords in English*

Software defined radio, digital radio, cognitive radio, digital signal processing

TT&C Subystem – On-Board Part

Master Thesis Task at Marmara Univ./Prof.Dr. Salih Bayar

1. AMAÇ ve HEDEFLER / AIM AND OBJECTIVES

The aim of this thesis is to design a software-based data link with optimized physical dimensions that can communicate in a high-frequency band and high data rate for use on small mobile robot platforms. The design will consist of two main units, a radio frequency front unit and a digital processing unit. The radiofrequency front end unit will consist of low noise amplifier (LNA) power amplifier (PA), mixer and filter groups. The digital processing unit will consist of analog to digital converter, digital analog converter and FPGA and memory units. The software-defined radio device hardware will be designed as a stand-alone device to use both mobile platform side and base station side.
Our design goals listed below;

- less than 500gr
- less than 100cm3
- At least 2 hours operation without fan cooler (passive cooling).
- 1MHz to 6GHz operating frequency. (Both ISM and non-ISM bands)
- At least 0.05W Output Power @ 900MHz
- 20MHz channel bandwidth

TT&C Subystem – On-Board Part

Master Thesis Task at Marmara Univ./Prof.Dr. Salih Bayar

3. ÖZGÜN DEĞER / ORIGINALITY OF STUDY

There are studies on software-defined radio using ready-to-use devices like the HackRF. However, there are not any domestic software-defined radio hardware development works. In this context, it is clear that a software radio hardware design is needed domestically. Also, our work will be an open-source project for both hardware and firmware wise.

4. YÖNTEM / METHODOLOGY

We can roughly divide this work into two different parts; the digital signal processor unit and the radio frequency transceiver unit. We plan to use Zynq 7000 System-on-Chip (SoC) from Xilinx as a digital signal processor in our design. Also, we intend to use AD9361 or similar wideband RF transceiver integrated circuit will for RF microwave part which can operate at 6GHz. Having these two parts, we plan to place them on different stacked printed circuit boards. So, we will have flexibility in our design and decrease device volume.

1/23/2022 AS-COMSAT, Jan-Feb 2021 120

Research plan to improve SDR communication system

Customized Software Defined Radio (SDR) Hardware Design

Asst. PROF. DR. Salİh Bayar

1/23/2022 AS-COMSAT, Jan-Feb 2021 121

Agenda

- Personal information
- Business idea
- Facts that support the business idea
- Existing System Problems and Proposed Solution
- Innovative aspects of the project
- Business packages
- Budget
- Target customers & Competitors

1/23/2022 AS-COMSAT, Jan-Feb 2021 122

Asst. Prof. Dr. Salih Bayar

- ❖ Birth year and place: 1980, Balıkesir
- ❖ BS: Yıldız Technical University - Electronics and Communication Eng., 2003
- ❖ MS: Karlsruhe Institute of Technology (KIT), 2007
- ❖ PhD: Boğaziçi Uni. - Computer Eng, 2015
- ❖ Worked places: Beko, Türk Telekom, KIT, TEMO Soft-, Hardware & Consulting e.K., Molex and Daimler Chrysler AG, Boğaziçi University, Beyken University, İstanbul Kültür University, Maltepe University, Idea Technology Solutions, Marmara University
- ❖ Specializations: Embedded Systems, FPGA, Parallel Software and Hardware, Multiprocessor and core architectures
- ❖ Contact:
  - ❖ http://ee.eng.marmara.edu.tr/akademik-kadro/ogretim-uyeleri
  - ❖ https://avesis.marmara.edu.tr/salih.bayar
  - ❖ http://www.salihbayar.com/

1/23/2022 AS-COMSAT, Jan-Feb 2021 123

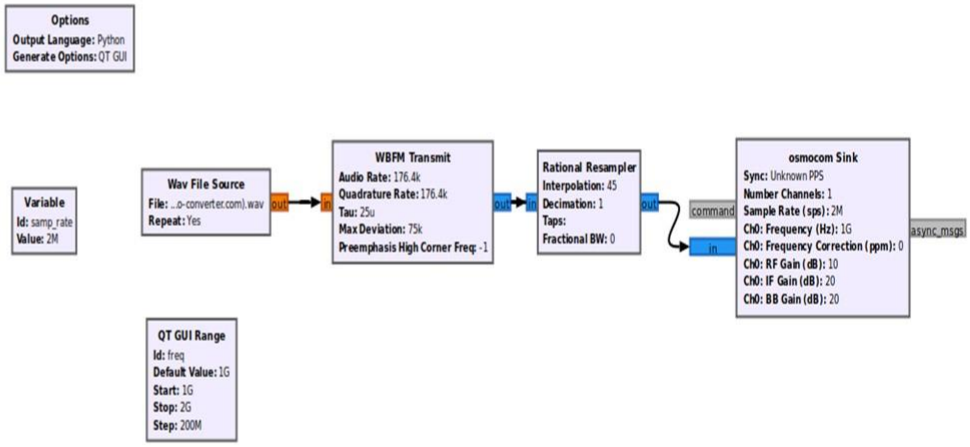
Business Idea

- A single software-based radio transmitter receiver that can replace separate hardware that can implement different frequency and modulation types for radio communication purposes.

Facts that support the business idea

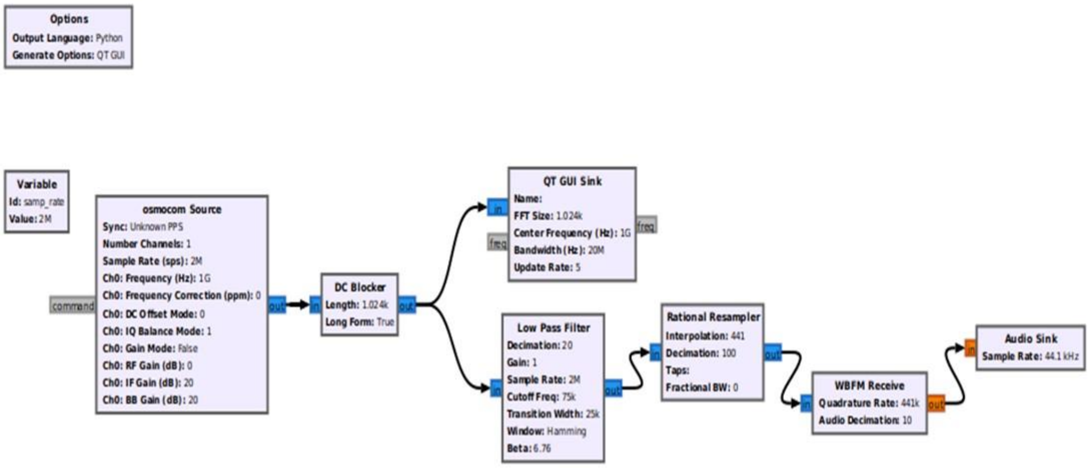
- Satellite communication and secure radio link production for our defense and industry needs. In this context, the development of a powerful communication card, which can be used in both satellite and defense industry at high frequency ranges (at least up to 20 GHz).

Work done so far: GnuRadio Reciever



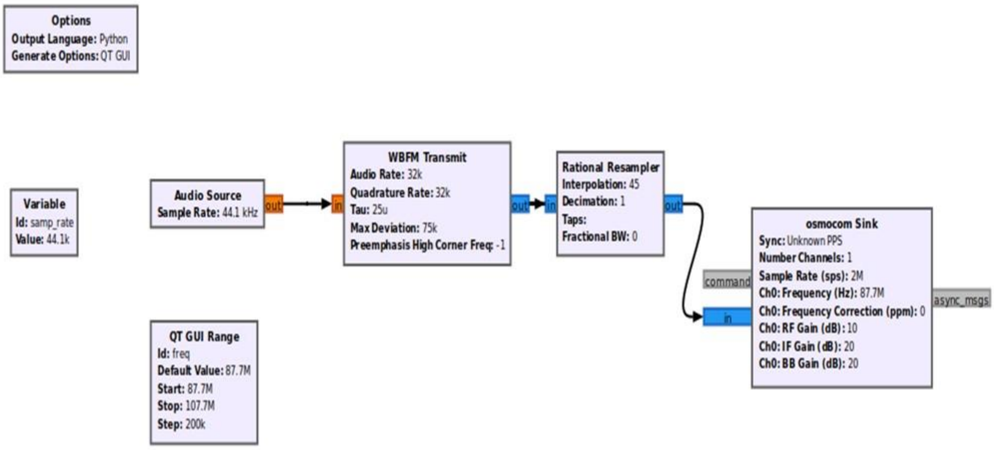
1/23/2022 AS-COMSAT, Jan-Feb 2021 126

Work done so far: GnuRadio Transmitter



1/23/2022 AS-COMSAT, Jan-Feb 2021 127

Work done so far: GnuRadio Live Broadcast



1/23/2022 AS-COMSAT, Jan-Feb 2021 128

Work done so far – Analog broadcasting

1/23/2022 AS-COMSAT, Jan-Feb 2021 129

HackRF Problems

- Until now, communication tests have been carried out on HackRF style low-performance products that are already available related to the business idea. Such a requirement has arisen due to the fact that the frequencies of such devices are not very high (up to 6GHz) and the FPGA cards used on them are very weak and insufficient.

The specific aspects of the project - Novelty

- It will have its own specific Linux-based operating system
- Thanks to the Linux-based operating system, many different features can be added
- High performance working capacity
- High frequency working capacity
- Having high data processing and transmission capacity

Work Packages

Work Package 1 — 3 Months

Reporting System
Literature Review
Design Document
Test Scenarios
Procurement Study

Work Package 2 — 2 Months

……….

Work Package 3 — 3 Months

……………..

Work Package 4 — 3 Months

Embedded System Design and Implementation

Work Package 5 — 3 Months

Prototype Production and Tests

Budger

General Operating Expenses	Amount	Amount Requested
Company / Company establishment expenses	1	4.000,00 TL
Electricity	12	2.400,00 TL
Dues	12	2.400,00 TL
Accounting	12	6.000,00 TL
Telephone and internet	12	1.800,00 TL
Travel expenses	2	4.500,00 TL
Fixture Expenses	1	4.000,00 TL

Budget cont.

Service Procurement and Consultancy	Amount Requested
Card design consultancy	15.000,00 TL
Embedded System and Software Consultancy	15.000,00 TL

Machine-Hardware-Software-Publication	Amount Requested
Laptop	16.000,00 TL

Personnel Expenses	Amount Requested
Business Owner – Salih Bayar	60.000,00 TL
Personnel	18.900,00 TL

Budget cont.

Consumables	Amount	Amount Requested
FPGA	2	8.000,00 TL
Peripherals	12	12.000,00 TL
HackRF	2	5.000,00 TL
Antennes	4	3.600,00 TL

Target Customers

- Government Agencies(TUSAŞ, TÜBİTAK, TÜRKSAT vb)
- Private Institutions
- Municipalities
- Personal Customers

1/36/2022 AS-COMSAT, Jan-Feb 2021

Rivals

- HackRF
  - ....
  - ....

1/37/2022 AS-COMSAT, Jan-Feb 2021

References

1. HackRF, https://greatscottgadgets.com/hackrf/

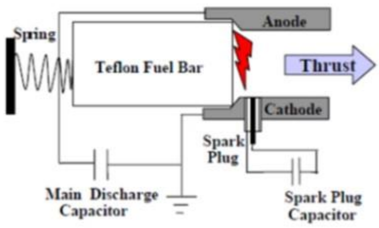
8.6.1 Electrical Propulsion Unit

Electrical Propulsion Unit for Orbit Control

Electrical Propulsion Unit for Orbit Control

Pulsed Plasma Thruster (PPT)

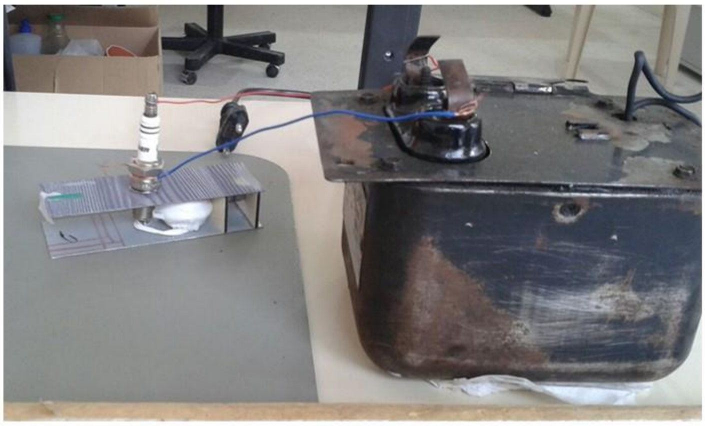
A pulsed plasma thruster (PPT) is a type of electromagnetic propulsion system , with a high specific impulse and low power and fuel requirements, that has been used on a number of satellites for station-keeping maneuvers.
A PPT works by ablating and ionizing material from a fuel bar (typically consisting of a chlorofluorocarbon such as Teflon) with the current from a discharging capacitor. The positive ions released are then accelerated between two flat-plate electrodes – one positive, the other negative – arranged in the form of two long parallel rails which are connected across the capacitor. Escaping from the spacecraft, the accelerated ions produce a thrust of some several hundred newtons. The capacitor is then charged up again from a power supply and the pulse cycle repeated.



1/23/2022 AS-COMSAT, Jan-Feb 2021 143

Electrical Propulsion Unit for Orbit Control

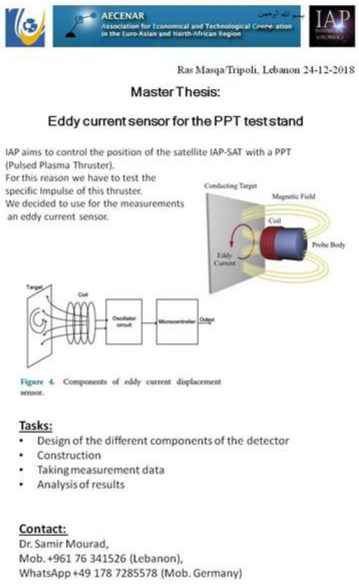
- Test Rig for Pulsed Plasma Thruster (PPT)



1/23/2022 AS-COMSAT, Jan-Feb 2021 144

Electrical Propulsion Unit for Orbit Control

- Eddy Current Sensor for the Pulsed Plasma Thruster Test Rig



AECENAR
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region
IAP

Ras Masqa/Tripoli, Lebanon 24-12-2018

Master Thesis:

Eddy current sensor for the PPT teststand

IAP aims to control the position of the satellite IAP-SAT with a PPT (Pulsed Plasma Thruster).
For this reason we have to test the specific Impulse of this thruster.
We decided to use for the measurements an eddy current sensor.

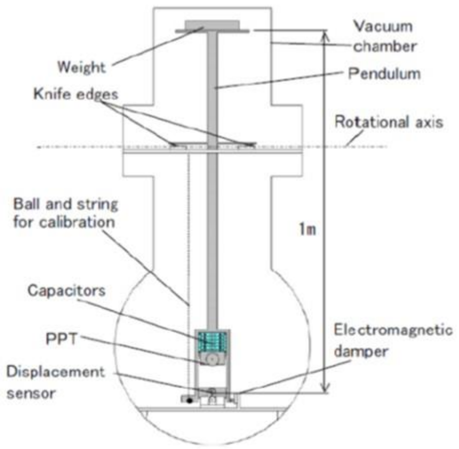
Figure 4. Components of eddy current displacement sensor.

Tasks:
- Design of the different components of the detector
- Construction
- Taking measurement data
- Analysis of results

Contact:
Dr. Samir Mourad,
Mob. +961 76 341526 (Lebanon),
WhatsApp +49 178 7285578 (Mob. Germany)

1/23/2022 AS-COMSAT, Jan-Feb 2021 145

Electrical Propulsion Unit for Orbit Control



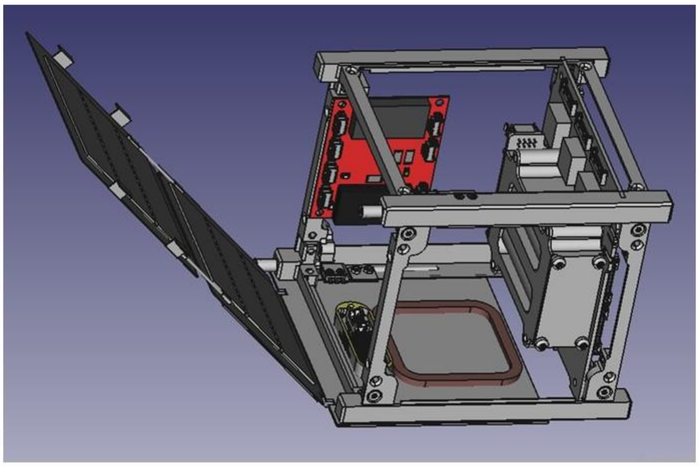
C:\Users\Aecenar1\Documents\IAP-SAT_2018\Mariam\IAP.Abgabe30.1.18.00..41\Pulsed Plasma Thruster (PPT) /engine

- Vacuum Chamber for the Pulsed Plasma Thruster Test Rig

1/23/2022 AS-COMSAT, Jan-Feb 2021 146

8.7 CAD Model

AS-COMSAT 3D Model



Maximum 1.3 kg

1/23/2022 AS-COMSAT, Jan-Feb 2021 157

8.8 Simulation platform

Simulation platform

- https://www.rtl-sdr.com/building-a-fossasat-1-lora-iot-ground-station/

1/23/2022 AS-COMSAT, Jan-Feb 2021 158

8.9 Thermal Control

Source:

Follow this and additional works at: https://scholarsmine.mst.edu/masters_theses

Part of the Aerospace Engineering Commons Department:

Recommended Citation

Boushon, Katelyn Elizabeth, "Thermal analysis and control of small satellites in low Earth orbit" (2018). Masters Theses. 7755. https://scholarsmine.mst.edu/masters_theses/7755

8.9.1 INTRODUCTION

Spacecraft thermal control is integral to mission success. The process of thermal control for a spacecraft involves managing the energy entering and leaving the spacecraft to ensure that the components of the spacecraft remain within an acceptable temperature range.

8.9.1.1 THERMAL CONTROL HARDWARE

The thermal control system on a satellite generally uses two basic approaches fortemperaturemanagement: passive and active thermal control. ...4.1.1. Passive Thermal Control. Passive thermal control techniques includematerialproperty selection, controlling the path of heat transfer, and using insulation systems to ensure that temperatures remain within acceptable limits [22]. Techniques including the use of multilayer insulation (MLI) and thermal coatings have a long heritage on traditional satellites, but may require modifications for use in small satellites.

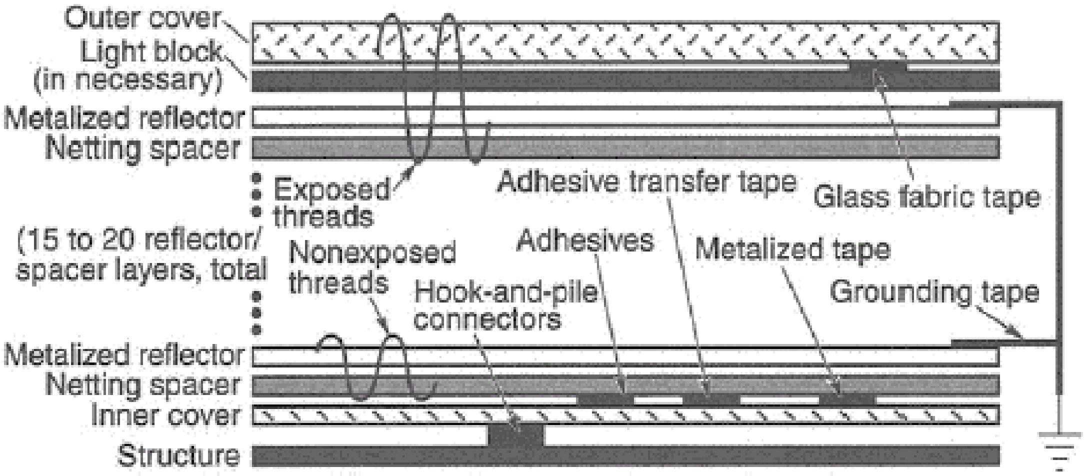


Figure 4.1. Typical MLI Blanket Composition [14]

Materials used for fabrication of an MLI blanket should always be treated as flight-critical hardware from the moment they are received. The materials should never be handled with bare hands and should never be exposed to uncontrolled and corrosive environments in order to avoid contamination and material degradation. Actions such as pulling or unnecessarily wrinkling the material should be avoided as this can cause stress in the layers and defects that may not appear until launch. Fabrication should occur in a temperature and humidity monitored Class 100,000 clean room to preserve the cleanliness and optical properties of the material. The fabrication area requires tables large enough to support the largest blanket being manufactured. All tools, equipment, templates, holding fixtures, and tables should be cleaned with a solvent that has a nonvolatile residue that does not exceed 0.02 g/L. The solvent must be compatible with the materials to avoid damaging the materials during normal cleaning operations. Clean white gloves or powder-free latex gloves suitable for clean room use must be used when handling the material, and clean room lab smocks must be worn [14].

8.9.1.2 Sunshields.

Sunshields offer shading for a satellite from direct solar

•impingement and the radiation environment of space [6]. Traditional sunshields have been made from a thin aluminum, titanium, or stainless-steel substrate, with a low absorptivity and high emissivity coating of silvered Teflon or white paint on the outer surface [1]. Sunshields for small satellites must unfold from a smaller form factor than their traditional counterparts. The implementation of sunshields for small satellites applications is fairly new, though Sierra Lobo has flown deployable sunshields on a few small satellite missions [10].

8.9.1.3 Radiators.

Waste satellite heat is rejected to space through the use of

•radiators. Regardless of the radiator configuration, be it a satellite structural panel or a flat plate radiator mounted to the satellite exterior, radiators reject heat from their surfaces by IR radiation. The optical properties determine the power of the radiator. Radiators must reject waste heat from the satellite while also rejecting heat impinging on the satellite. Most radiators have a high emissivity to maximize heat rejection and low absorptivity to limit heat loads from the space environment. Typical finishes include quartz mirrors, white paint, and silvered or aluminized Teflon [14].

•The simplest and most common radiators are the existing panels of the satellite exterior. For example, an exterior aluminum honeycomb panel can serve as a structural panel as well as a radiator. The face sheets of the panel distribute away from electronics boxes that are mounted to it, with the outside panel face acting as the radiating surface. The face sheets can also be made thicker to increase the heat distribution. Separate plates called “doublers,” typically made of aluminum, can also be added under high heat dissipating electronics boxes to help distribute the heat. These measures may result in mass increases that will not fit within the satellite mass budget. Heat pipes can be considered in this situation to distribute spread the heat.

4.1.2. Active Thermal Control

- **4.1.2.1 Heaters.** Heaters are often the simplest device to use for active
- thermal control [19]. Their main function is to maintain satellite components in the required temperature range, but they can also be used to warm up components that are dormant before their activation, to control temperature differences to greater stability, and to dissipate excess satellite power [1]. Heaters are the only active thermal control hardware that have been successfully miniaturized for use on small satellites [6].
- **4.1.2.1.1 Heater types.** The most commonly used type of heater is the patch
- heater. Patch heaters consist of an electrically resistant element bonded between two sheets of flexible electrically insulating material [14]. The electrically resistant element is typically an etched foil, such as Nichrome, and the insulating material is typically a Kapton film [1].

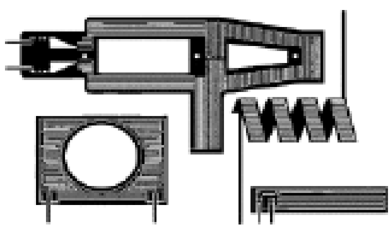


Figure 4.7. Custom Shaped Patch Heaters [14]

8.9.2 THERMAL DESIGN PROCESS

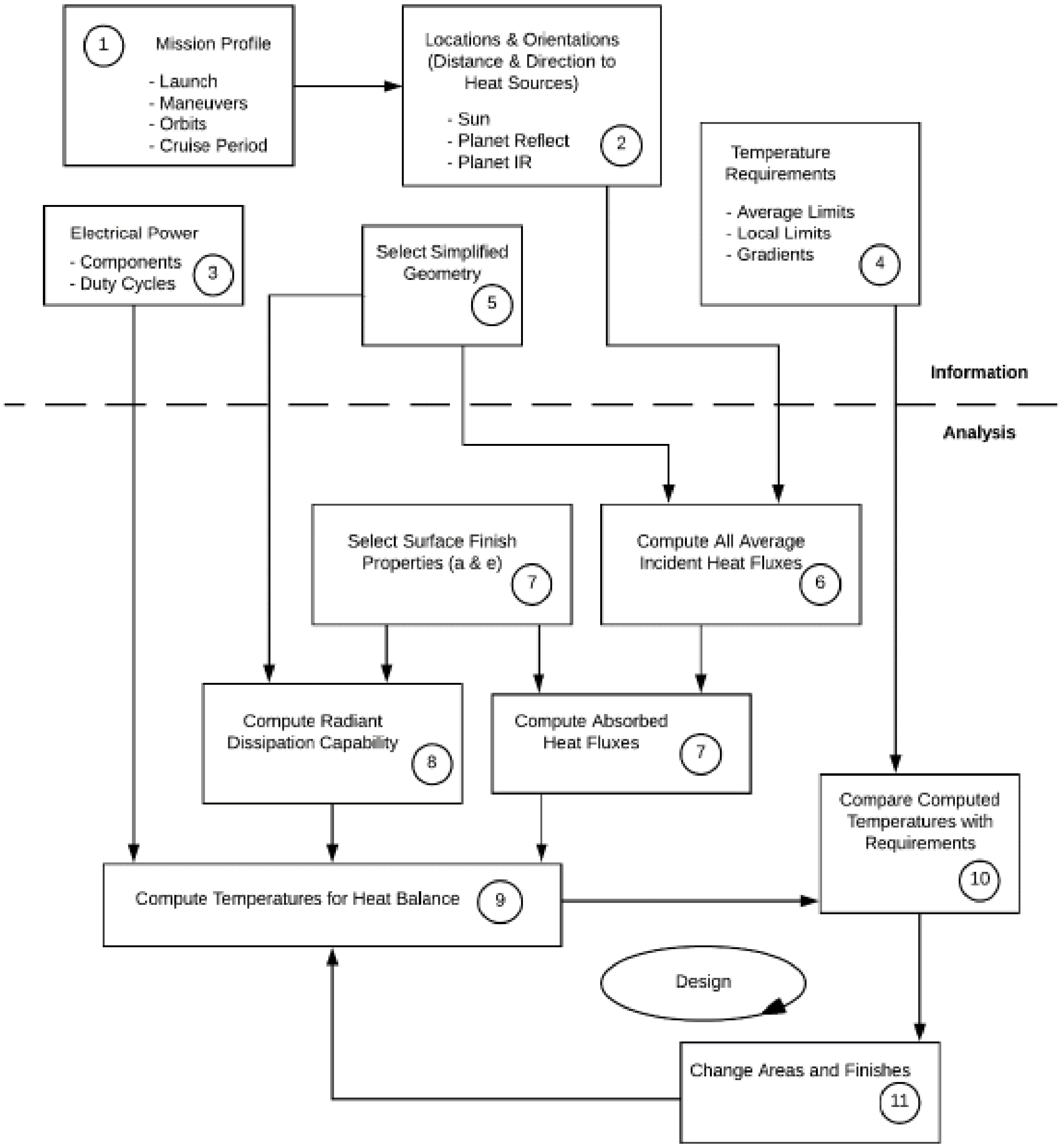


Figure 5.1. Thermal Analysis and Design Process [18]

Case Study

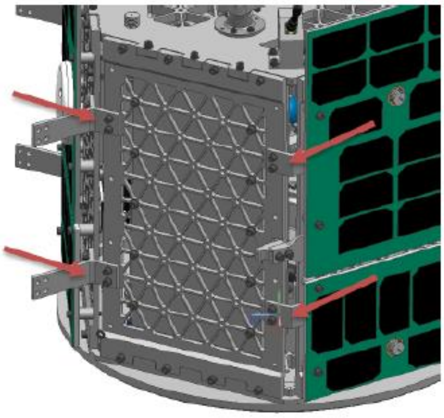


Figure 6.15. MR SAT Side Panel Brackets

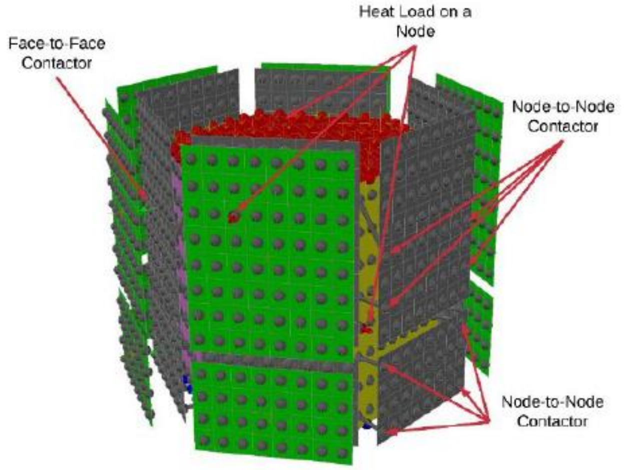


Figure 6.16. MR SAT Model Revision C – Conduction and Heat Load Locations

AS-COMSAT, Jan-Feb 2021

Case Study

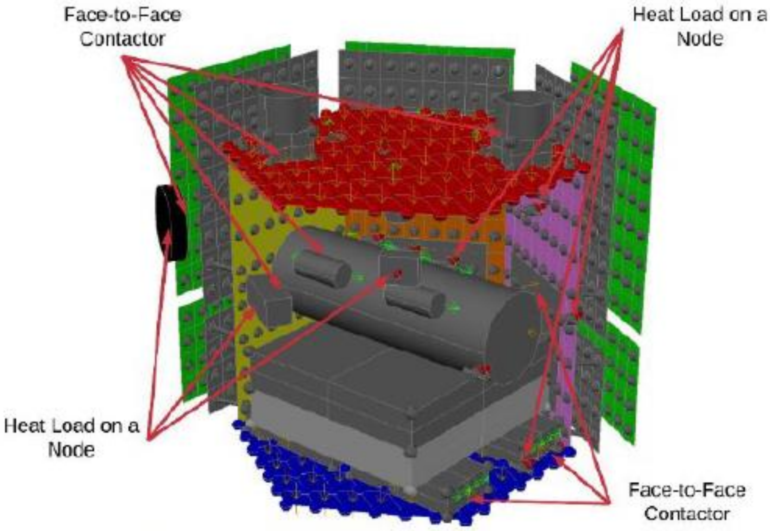


Figure 6.24. MR SAT Model Revision D – Conduction and Heat Load Locations

Case Study

Table 6.11. Model Revision E – Transient Results for Components

Component	Max Temp [°C] - Hot Case	Min Temp [°C] - Cold Case	Max Operating Temp [°C]	Min Operating Temp [°C]	Max Survival Temp [°C]	Min Survival Temp [°C]
TiNi 1	42.3084	2.4930	70	-65	-150	150
TiNi 2	38.6156	2.7749	70	-65	-150	150
TiNi 3	39.9796	2.7160	70	-65	-150	150
MCU Digital	46.7935	10.9856	85	-40	-65	150
Isovalve 1	47.0767	10.9767	49	-18	-	-
Isovalve 2	47.0771	10.9768	49	-18	-	-
Isovalve 3	47.0768	10.9767	49	-18	-	-
Thruster 1	47.5563	1.8898	49	-18	-	-
Thruster 2	46.8478	1.4100	49	-18	-	-
Thruster 3	48.0069	1.5618	49	-18	-	-
Thruster 4	48.9021	1.5535	49	-18	-	-
Thruster 5	34.8042	1.9214	49	-18	-	-
Thruster 6	35.5939	2.1649	49	-18	-	-
Thruster 7	37.3262	2.4853	49	-18	-	-
Thruster 8	36.0827	2.3751	49	-18	-	-
Thruster 9	41.0231	3.5942	49	-18	-	-
Thruster 10	42.4893	3.4262	49	-18	-	-
Thruster 11	38.4595	2.6354	49	-18	-	-
Thruster 12	42.1236	2.2227	49	-18	-	-
Pressure Transducer 1	46.6006	10.9703	82	-29	-	-
Pressure Transducer 2	46.5972	10.9711	82	-29	-	-
Comm Radio	46.6040	11.3703	70	-30	-30	85
GPS Receiver	46.8809	11.3115	85	-40	-55	95
GPS Antenna	39.1379	-1.9531	85	-55	-55	85
Flight Raspberry Pi	49.2888	13.7380	80	-25	-	-
IMU	46.7473	11.1784	85	-40	-	-
Sun Sensor Camera 1	23.1908	-2.2587	70	-10	95	-40
Sun Sensor Camera 2	58.0436	-3.2861	70	-10	95	-40

8.9.2.1 Lessons Learned.

Many lessons were learned throughout the process of thermal model construction, analysis, and application of thermal control. It is very important to keep up-to-date documentation during the process. Documentation should include information regarding all model input parameters including orbit profile, satellite dimensions, and material properties. It should be very clear in the documentation which parameters are used for which model revision. Each new model revision should be saved as a separate file and should be accompanied by its own documentation.

8.9.3 CONCLUSION

THESIS SUMMARY

Proper thermal analysis and control for spacecraft is essential for successful mission performance. Thermal control methods for traditional satellites are well documented, but many methods for small satellite applications are still in the development stages. This thesis study presents proven methods of thermal analysis and control specifically relating to small satellites in low Earth orbit in order to act as a resource for future reference.

Satellite thermal analysis typically involves using analytical processes assisted by computer software to determine temperatures at nodes in the model by applying a numerical approximation method, typically the finite difference method. The solar vector, albedo factors, satellite component dissipation, orbit beta angle, and orbit altitude all affect the outcome of the thermal analysis. Thermal extrema cases define the upper and lower bounds on temperature predictions. The results of thermal models are then verified through testing, and the thermal model adjusted to more closely reflect the test results.

Satellite thermal control methods are used to regulate temperatures to ensure that components function properly throughout the mission. Thermal control systems on a satellite can use both passive and active thermal control. Small satellites most commonly employ passive methods as they tend to be lighter, more reliable, and do not require power. Passive thermal control methods include the use of multilayer insulation, thermal surface coatings and finishes, tapes, sunshields, radiators, heat pipes, phase change materials, and heat switches.

8.9.4 CONTRIBUTIONS TO THE SMALL SATELLITE COMMUNITY

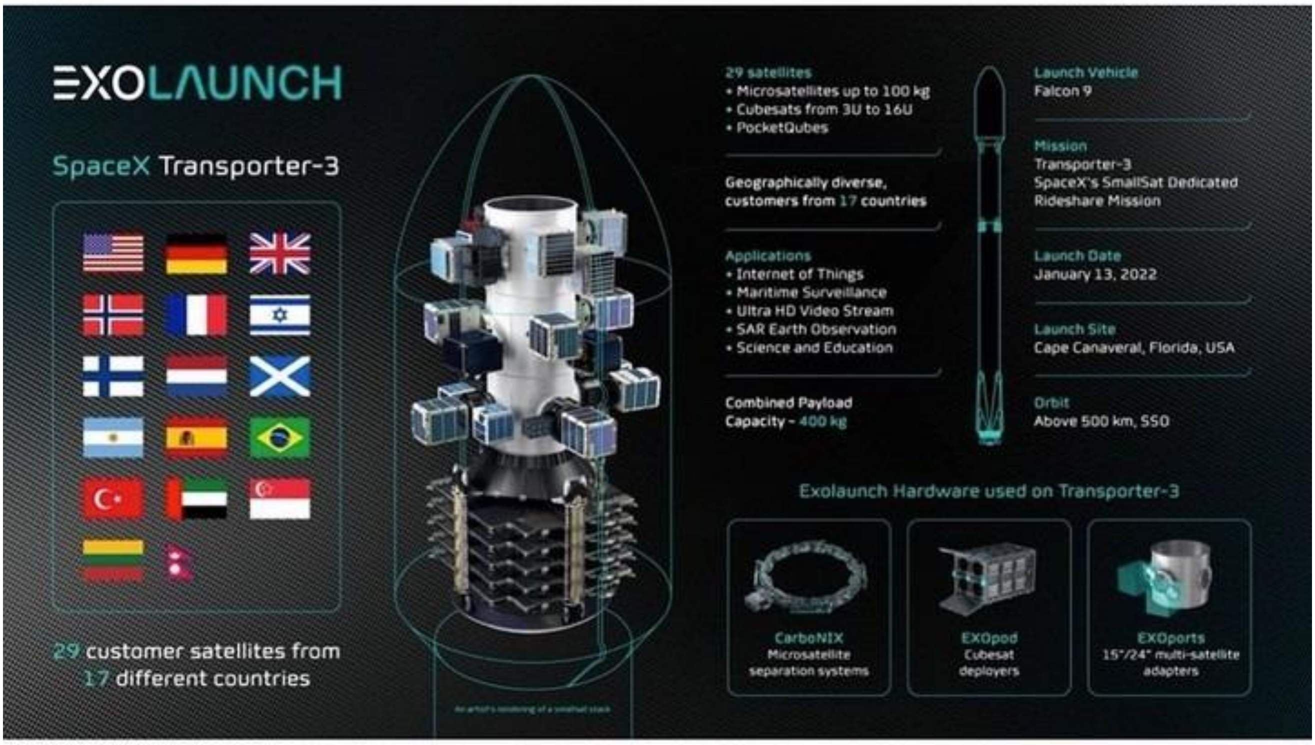
The goal of this thesis study has been to provide a resource to guide the small satellite thermal control system design and analysis process. Inexperienced engineers and academic teams will be able to use this thesis study as the starting point for their work in the thermal analysis and control of their small satellite designs. Basic heat transfer concepts and satellite heating environments are discussed for the benefit of student engineers still learning about the topics. Thermal analysis processes from various sources are summarized and presented, as well as a case study to demonstrate the use of these practices and an outline of the practical application of model construction, analysis, and design in Appendix A.

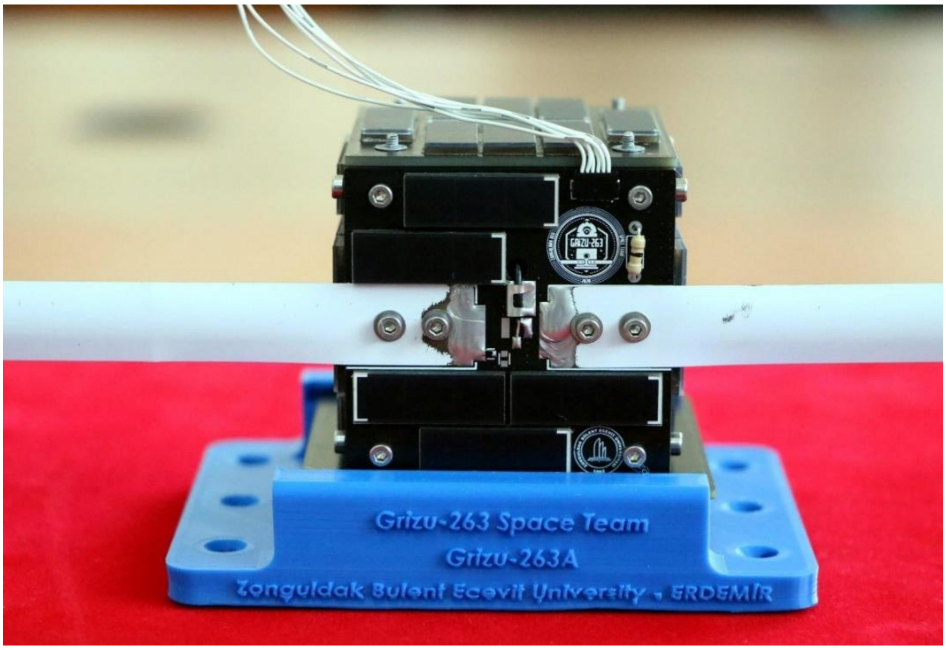
Optical Properties of Common Surface Coatings and Finishes [14]

Material	Absorptivity	Emissivity
Optical Solar Reflectors		
Teflon, Aluminized, 0.5 mm	0.14	0.4
Teflon, Aluminized, 1 mm	0.14	0.6
Teflon, silvered, 2 mm	0.08	0.68
Teflon, silvered, 10 mm	0.09	0.88
Black Coating		
Chemglaze Z306 Black Paint	0.96	0.91
Black Z306 polyurethane paint, 3 mm	0.95	0.87
Ebanol C Black	0.97	0.73
Rough black matte, black paint	0.9	0.9
Films and Tapes		
Kapton, aluminized, 0.25 mm	0.31	0.45
Kapton, black (carbon loaded), 1 mm	0.92	0.88
Tape, 235-3M, black	0.95	0.9
Tape, aluminum	0.1	0.04
White Coatings		
Chemglaze A276 white paint	0.24	0.9
Hughson A-276 white paint	0.26	0.88
Magnesium oxide white paint	0.09	0.9
Polyurethane white paint	0.27	0.84
Other Paints		
Aluminum Paint	0.3	0.31
Chromacoat aluminum paint	0.28	0.05
Silicone aluminum paint	0.29	0.3
Metals		
Aluminum, buffed	0.16	0.03
Aluminum, polished	0.15	0.05
Beryllium copper	0.31	0.03
Copper, buffed	0.3	0.03
Copper foil tape, tarnished	0.55	0.04
Gold, electroplated	0.23	0.03
Silver, polished, unoxidized	0.04	0.02
Stainless steel	0.47	0.14
Titanium	0.4	0.55
Anodized Aluminum		
Black anodize	0.65	0.82
Chromic anodize	0.44	0.56
Clear anodize	0.27	0.76
Plain anodize	0.26	0.04

8.10 Launch issues

8.10.1 Exolaunch





I think they used a tape meter for the antennas. They can be folded and they can open like a switchblade by the tension of the material which is a nice idea. I have seen yagi type antennas from cut tape measures made by RF amateurs before .

5x5x5 cm pocket sat Grizu 263A is on air https://www.aa.com.tr/tr/bilim-teknoloji/turkiyenin-ilk-cep-uydusu-grizu-263a-uzay-yolculuguna-hazirlaniyor/1861764

8.10.2 Offer from Russian Company (Launch with Soyuz) from May 2021

www.gklaunch.ru, info@gklaunch.ru

GK LAUNCH SERVICES

CONFIDENTIAL
#00548 06.05.2021

2 U, 3 kg: 110,000$

Dear Samir! Thank you for your launch quote request. We have prepared a ROM price proposal based on your satellite's characteristics and selected a suitable mission.

Mission:

> Cluster launch
> Launch period: 2 quarter of 2022
> Orbit: SSO, LTAN 11:00
> Baikonur Cosmodrome
> Primary payload: contracted
> Secondary payload: available

Launch price:

for 3kg satellite is 110000$

The price includes:

1. Program management and program documents;
2. Support of Customer personnel visits;
3. Administrative support to the Customer personnel at GK facilities in accordance with the terms and conditions to be defined in the contract;
4. Program reviews and meetings as may be necessary;
5. Interface Control Document with the results of analyses and reports as may be necessary;
6. Hardware (adapter and separation system, umbilical connector, harness for testing), personnel and equipment for fit-check to be performed at the NPOL facility (SC dummy for testing to be provided by Customer);
7. Hardware (adapter and separation system, umbilical connector, harness), personnel and equipment for integration of the flight SC with the launcher;
8. Hardware, ground support equipment (Space Head Module (SHM), Fregat upper stage, launch vehicle) and personnel for processing of the SHM with SC and execution of launch;
9. Customs clearance of SC/GSE on entry into Russia, customs clearance of GSE on exit from Russia;
10. Transportation of satellite and GSE from the entry/exit port to payload preparation

facility and back of GSE, including their customs clearance;
11. Performance of launch campaign and provision of:
Work place in AITB;
Administrative and storage premises;
Power supply;
Provision of logistics to Customer’s personnel whilst at the launch site (payment for the services to be made by Customer);
Communications services (international telephone calls to be paid for in accordance with the terms and conditions to be defined in the contra).
Launch of deployer with CubeSat(s) into the required orbit;
12. Provision to Customer of LV telemetry data confirming the SC separation and initial orbit parameters;
13. Post-launch services;
14. Photographic and video documentation;
15. Linguistic support;
16. Procurement of third party liability insurance for the damage due to the launch activities and support in obtaining the satellite insurance;
17. Launch observation.

You can add insurance of the satellite ground and space related risks and insurance of
the launch service. To do so, call us or write a response letter.

This is a ROM launch price proposal to be finalized after we receive all the documents. Some mission parameters may change.

Standard Milestone payment plan:

	Milestone	Payment, USD (%)	Preliminary Date of milestone completion and submission of invoice
1	Manifesting of Payload on a Launch Mission	15%	Manifesting of Payload
2	IRD review	15%	Completion of IRD review
3	ICD approval	25%	Approval of ICD
4	Ground Tests	20%	Completion of ground tests
5	Start of Launch Campaign	15%	Upon Payload arrival at Launch Site
6	Successful launch	10%	L+2 weeks

8.10.3 Companies launch satellite

From

AECENAR Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region

IAP INSTITUTE FOR ASTROPHYSICS

Steps for Launching Satellite

Author:
Siham Aisha

Last Update: 27.12.2021

8.10.3.1 GK laucnh (Russian)[15]

GK Launch Services is a company established by Glavkosmos (a subsidiary of ROSCOSMOS State Space Corporation) and International Space Company Kosmotras. GK Launch Services is an operator of Soyuz-2 commercial launches from the Russian spaceports.

The key targets this joint venture aims at include advancing of commercial launch services, promotion of Russian launch vehicles on the world market, and strengthening the positions of Russia as the most competitive launch service provider.

A solid competence of the two partners facilitates achieving these ambitious goals. Glavkosmos has been participating in global international space projects over 30 years and has already had experience in provision of launch services with Soyuz-2 rocket. Kosmotras has lofted over 100 payloads within 22 commercial launches.

8.10.3.2 Swarm company (U.S)[16]

Swarm Technologies, Inc. is a private company building a low Earth orbit satellite constellation for communications with Internet Of Things (IOT) devices using a Store and forward design. An early investor was Craft Ventures. On July 16, 2021, Swarm entered into an agreement to become a direct wholly-owned subsidiary of SpaceX.[2]

They have an Federal Communications Commission (FCC) licence for low bandwidth communications satellites in low Earth orbit.[3]

In 2018 Swarm became the first ever company found to have deployed satellites without regulatory approval after an FCC investigation into the startup's launch on an Indian PSLV rocket of its first four picosatellites in January that year.[4]

By December 2020, Swarm had launched 9 test satellites and 36 of a planned 150 low Earth orbit satellites to provide communication with IOT devices.[5]

[15] http://gklaunch.ru/en/

[16] https://en.wikipedia.org/wiki/Swarm_Technologies

In February 2021 Swarm announced that its commercial services were now live using 72 commercial satellites providing its global low cost data to customers.[6]

The Swarm Tile is its dedicated satellite two-way data modem designed to be low energy and embedded on the PCB of third party products. Other products include a data plan and development kit.[7]

8.10.3.3 Exolaunch company (Germany)[17]

The protocol of launching is:

10) LAUNCH PLANNING

Every mission is unique. We listen to you and offer solutions that will enable the successful launch of your satellite. With precision, knowledge and expertise, we make the complex work of your specific campaign a simple and affordable experience. No stress.

11) MISSION MANAGEMENT

Next, we produce your event. We provide technical management of the satellite adaptation to a launch vehicle, interface control document development, mission analysis delivery, and launch schedule coordination.

12) SEPARATION SYSTEMS

EXOpod deployers for cubesats and CarboNIX, the shock-free separation systems for microsatellites, are designed and produced by Exolaunch to support your mission needs. Later, we adapt the deployment systems to the launch vehicle to safely deliver your satellite into its target orbit.

13) DEPLOYMENT SEQUENCER

EXObox is essential for smallsat cluster launches. It is a unique, highly reliable and modular deployment sequencer to manage the deployment of up to 50 satellites with just one EXObox unit. It will precisely and safely separate small satellites into their target orbits.

14) LICENSING

In this phase, we handle the complex legal and regulatory support documents that are required for launch. Your desk is now clean!

15) ENVIRONMENTAL TESTING

Our fully-fledged environmental testing services are tailored to the requirements of any launch vehicle, ensuring your satellite has made the grade. All of the tests are performed in Berlin, and yEnvironmental tests profiles and types:

f. Vibration and shock testing

We provide a full range of mechanical testing to cover the qualification, proto-flight or acceptance requirements of all launch vehicles.

g. Thermal and vacuum testing

[17] https://exolaunch.com/

Thermal cycling and thermal vacuum testing is available to meet your mission requirements. Whether testing survivability limits or simply performing a vacuum bake-out, we can perform the tests that fit your needs.

h. Qualifation and acceptance test

Qualification of your design based on the composed mechanical loads in order to meet the requirements of most common launch vehicles. Tailored acceptance testing profiles of your flight models to the launch vehicle of your choice, ensuring the function of your spacecraft while increasing confidence in its reliability.

i. Test profiles design

We offer support to develop individual specifications for mission tailored test profiles with optimized loads and durations. Inclusive our acceptance guarantee of the test profiles by the launch authority.

j. Adapters & additional hardware

We offer TestPods and other test fixtures with the test interfaces identical to the launch vehicle interfaces, ensuring the validity of all test results. ou are offered our cleanroom for satellite checkouts.

16) SHIPMENT

Our expertise in global logistics and experience with customs clearance allows us to process worldwide shipping of payloads and equipment in a safe, convenient and timely fashion. We will ensure your satellite reaches the launch site safely and without hassle.

17) INTEGRATION SERVICES

When your satellite meets the launch vehicle. We will seamlessly integrate your satellite with the support hardware onto its launch vehicle.

18) LAUNCH

The launch vehicle has left Earth. Shortly afterwards, we receive a positive signal that your satellite is in orbit and is ready to start changing the world.

8.10.3.4 Gunter's space page[18]

The four **SpaceBEE**, formerly known as **BEE**s (**Basic Electronic Elements**), picosatellites, built to the 0.25U CubeSat form factor are to demonstrate two-way satellite communications and data relay for Swarm Technologies Inc.

The mission is to test the world's smallest two-way communications satellites to serve as a cost-effective low-data rate Internet of Things (IoT) network connectivity solution for remote and mobile sensors. The initial experimental space deployment is comprised of four satellites, each with a 1/4U form factor employing radar signature enhancement technology, which enables them to be passively

[18] https://space.skyrocket.de/doc_sdat/spacebee.htm

tracked, and using VHF band frequencies for communications. There will also be an experimental deployment of ground stations for communications with the space units.

The mission is to demonstrate the capabilities of these picosatellites for serving low data rate communication relays for remote sensors and data collectors. Experimental operations is scheduled to begin upon launch for a period of at least 6 months and up to 2 years

The tiny satellites have very small radar cross section, which might complicate the tracking. Therefore they featured a GPS device in each satellite that would broadcast its position on request. Also the four smallest faces of the satellites are covered with an experimental passive radar reflector developed by the U.S. Navy's Space and Naval Warfare Systems Command, which according to the FCC application would increase the satellites radar profile by a factor of 10.

The FCC dismissed Swarm's application. Nevertheless, the satellites have been launched, apparently without a valid licence, in January 2018 on an Indian PSLV-XL rocket under the name **SpaceBEE**. The ownership of the SpaceBEEs remained obscure, until in an IEEE Spectrum article the identity of the SpaceBEEs with Swarm's BEE satellites was revealed.

A follow-on mission, SpaceBEE 5 to 8, with larger 1U CubeSats was also not granted a licence after this. Later the licesnce was granted. SpaceBEE 1 to 4 were also granted an operation licence.

The operational 2nd generation SpaceBEE satellites reverted back to the 0.25U form factor.

8.10.3.5 Antrix corporation[19]

I contact this company and the emails is below:

8.10.3.6 RE: FW: question for procedure to launch a satellite

November 25, 2021 7:44 am 28 KB

From:

Ganesh Mohan <ganesh_mohan@antrix.co.in>

To:

siham.aisha@temo-group.com

Hi Siham,

Could you please elaborate on what is it that you're looking for in administrative and legal procedures?

I presume all the permits, authorisations and notices of non-opposition including frequency filing / other regulatory mandates would be already done by you, during the course of the project.

We will support in any administrative matters during the import and re-export of the payload and the auxiliaries, including transportation, testing facilities, accommodation of personnel etc during the launch base. Once the satellite is separated in a low earth orbit, the control gets passed over to you. All the other

[19] https://www.antrix.co.in/

aspects like Indemnity, Insurances etc would be covered in the launch contract and we can discuss over it during the course of execution of the contract.

Thanks

Ganesh

From: Siham [mailto:siham.aisha@temo-group.com]
Sent: 22 November 2021 15:53
To: Ganesh Mohan
Subject: Re: FW: question for procedure to launch a satellite

Hello,
Thank you for your response.

We need know what is the administrative and legal procedure in details, I mean if we want to keep some legal files we wish to inform us.

I prefer an email contact to be clearly.
our location is in Lebanon - Tripoli - Ras masqa, name of my organization is North Alternative Power departement TemoGroup

Regards

On November 22, 2021 at 11:57:24 am +02:00, Ganesh Mohan <ganesh_mohan@antrix.co.in> wrote:

Hi Siham,

Thanks for your interest in Antrix.

The procedure is that

1. there will be a launch services agreement that we will have to execute, wherein the said satellite will be accommodated as a ride share with one of the upcoming PSLV / SSLV missions.

2. There will be an Interface control document, where all the details of the testing, Dynamic studies, sequencing, power supply etc would be addressed and cleared.

3. The launch would happen from Sriharikotta, where the necessary testing / operations / safety procedures would be happening.

With respect to the pricing, it will vary. May I know where are you based at and the company that you're working for? We can probably discuss it over a call.

Thanks

Ganesh Mohan

Manager, Antrix Corporation

From: Siham [mailto:siham.aisha@temo-group.com]
Sent: 16 November 2021 15:39
To: sonali@antrix.co.in
Subject: question for procedure to launch a satellite

Hello,

I want to launch a satellite, Could I have a quote of price and what is the procedure to launch it?

1- the time frame it's about in the middle of 2022

2- launch to the same inclination, I mean in the same orbit and the inclination is 100 degre - 127 degre

3- our spacecraft don't have propulsion

4- size of our spacecraft is : 10X10X20 cm 2U for 2 satellites

5- mass is: 3Kg

6- the purpose: small pilot system for communication satellite.

7- altitude: 650 km

8- elliptic semimajor axis a = 7027.748 km

9- orbit heigh: 500- 600 km

Regards,

Siham

8.10.3.7 Procedure to launch a satellite in India[20][21]

Norms, Guidelines and Procedures for Satellite Communications Announced

The Government has approved a policy that envisages allocation of INSAT system capacity for non-governmental users, registration of Indian satellite systems by private Indian companies and limited use of foreign satellites in special circumstances. The Department of Space (DOS) will be the administrative ministry in all matters related to satellite systems in India.

As per the policy, the Indian National Satellite System (INSAT) capacity will be made available to non-government (private) Indian Service Providers on a commercial basis subject to availability after meeting the government needs. The DOS will allocate INSAT capacity for private users. DOS may also build capacity in INSAT system for private users on request on commercial basis.

Private Indian companies with a foreign equity less than 74 percent are now allowed to establish Indian Satellite Systems. These companies can submit their applications for registering their satellite systems to the Committee for Authorising the establishment and operation of Indian Satellite Systems (CAISS). The office of CAISS is set up at the SatCom Programs Office at ISRO Headquarters, Antariksh Bhavan, New BEL Road, Bangalore- 560 094. The authorisation to operate the Satellite System and the Orbit spectrum notification/registration will be done by CAISS. However, operating licenses for services to be provided by the Indian Satellite Systems will be issued only by the concerned administrative departments like Department of Telecommunication for telecom services and Ministry of Information and Broadcasting for TV/Radio broadcasting.

Foreign satellites will also have allowed to be used in special circumstances for satellite communication services in India. The service licensing departments may allow the use of foreign satellites only in consultation with the Department of Space. If suitable capacity/capability is available in INSAT or Indian Satellite Systems, operations with foreign satellites will not be

---

[20] https://www.isro.gov.in/update/08-may-2000/norms-guidelines-and-procedures-satellite-communications-announced

[21] https://www.isro.gov.in/contact-us-0

permitted. For the use of foreign satellites for Internet Service Provider (ISP) gateways, the existing procedures established by Telecom Commission will apply.

India's Space Policy[22]

Remote sensing

Recognizing that Remote Sensing data provides much essential and critical information - which is an input for developmental activities at different levels, and is also of benefit to society.

Noting that a large number of users - both within and outside government, use Remote Sensing data from Indian and foreign remote sensing satellites for various developmental applications.

Taking into consideration the recent availability of very high-resolution images, from foreign and commercial remote sensing satellites, and noting the need for proper and better management of the data acquisition/ distribution from these satellites in India.

Recognizing that national interest is paramount, and that security consideration of the country needs to be given utmost importance.

The Government of India adopts the Remote Sensing Data Policy (RSDP) - 2011 containing modalities for managing and/ or permitting the acquisition / dissemination of remote sensing data in support of developmental activities. Department of Space (DOS) of the Government of India shall be the nodal agency for all actions under this policy, unless otherwise stated.

6. For operating a remote sensing satellite from India, license and/ or permission of the Government, through the nodal agency, shall be necessary.
   - As a national commitment and as a "public good", Government assures a continuous and improved observing/ imaging capability from its own Indian Remote Sensing Satellites (IRS) programme.
   - The Government, through the nodal agency, shall be the sole and exclusive owner of all data collected/ received from IRS. All users will be provided with only a license to use the said data, and add value to the satellite data.
   - Government reserves the right to impose control over imaging tasks and distribution of data from IRS or any other Indian remote sensing satellite, when it is of the opinion that national security and/ or international obligations and/ or foreign policies of the Government so require.
   -
7. For acquisition/ distribution of remote sensing data within India, license/ permission from the Government of India, through the nodal agency, shall be necessary.
   - Government reserves the right to select and permit agencies to acquire/ distribute satellite remote sensing data in India. DOS shall be competent to decide on the procedure for

[22] https://www.isro.gov.in/indias-space-policy-0

granting license/ permission for dissemination of such data, and for the levy of necessary fees.

- To cater to the developmental needs of the country, the National Remote Sensing Centre (NRSC) of the Indian Space Research Organisation (ISRO)/ DOS is vested with the authority to acquire and disseminate all satellite remote sensing data in India, both from Indian and foreign satellites.
  - NRSC shall enter into appropriate arrangements with DOS for acquiring/ distributing data from IRS within the visibility circle of NRSC's receiving station(s).
  - NRSC and/ or Antrix Corporation Ltd., shall be competent to enter into agreements with foreign satellite operator(s) for acquisition/distribution of foreign satellite data in India. However, NRSC will distribute the data as per terms agreed to with Antrix Corporation Ltd.
- NRSC shall maintain a systematic National Remote Sensing Data Archive, and a log of all acquisitions/ sales of data for all satellites.

8. For acquisition and distribution of IRS data for use in countries other than India, the Government of India, through the nodal agency, shall grant license to such bodies/ agencies of those countries as are interested in the acquisition/ distribution of IRS data, as per specific procedures.
   - The Antrix Corporation Ltd. (of DOS) is vested with the authority for receiving the applications for grant of license for acquisition/ distribution of IRS data outside of India; to consider and decide on the granting of license within the policy considerations of the Government, and to enter into licensing agreements with the prospective users on behalf of the Government. Antrix Corporation Ltd. shall also be competent to levy such fees for granting licenses as may be considered appropriate by it. It shall also be responsible, where necessary, for rendering any further help/ guidance needed by the license.
   - The Government reserves right to impose restrictions over imaging tasks and distribution of IRS data in any country when it is of the opinion that national security and/ or international obligations and/ or foreign policies of the Government so require.
9. The Government prescribes the following guidelines to be adopted for dissemination of satellite remote sensing data in India:
   - All data of resolutions up to 1 m shall be distributed on a non-discriminatory basis and on "as requested basis"
   - With a view to protect national security interests, all data of better than 1 m resolution shall be screened and cleared by the appropriate agency prior to distribution; and the following procedure shall be followed:
     - Government users namely, Ministries/ Departments/ Public Sector/ Autonomous Bodies/ Government R&D institutions/ Government Educational/ Academic Institutions, can obtain the data without any further clearance.
     - Private sector agencies, recommended at least by one Government agency, for supporting development activities, can obtain the data without any further clearance.
     - Private sector agencies, recommended at least by one Government agency, for supporting development activities, can obtain the data without any further clearance.
     - Specific requests for data of sensitive areas, by any user, can be serviced only after obtaining clearance from the HRC.

- Specific sale/ non-disclosure agreements to be concluded between NRSC and other users for data of better than 1 m resolution.

10. This Policy (RSDP-2011) comes into effect immediately, and may be reviewed from time-to-time- by Government.

I contact spaceflight:

8.10.3.8 RE: [External] - Sales - Website Submission

November 17, 2021 6:31 am 52 KB

From:

Keiko Nasu <knasu@spaceflight.com>

To:

siham.aisha@temo-group.com

Hello,

Sorry, for some reason your e-mail has been in the spam box and it took me a while to find your response. I deeply apologize for that.

We are just to close the manifest for June 2022 SpaceX rideshare. We need to provide a good portion of deliverables to SpaceX by December 1, but if you will be able to do that, we might be able to launch your satellite with that mission.

The issue is how you would like to separate the 2 satellites. Our Sherpa-OTV could provide in-plane phasing but it's not going to be very economical for 2U satellite (and I heard that our Sherpa capacity is already filled).
We might be able to separate a little bit by delay the second satellite deployment (30min or so at most), but it will not give you a lot of separation.

Please confirm SSO is the orbit you would like to go, and will check other possible launch options.

Let me know if you prefer to have a brief call.

Warm regards,

Keiko Nasu
Business Development, Spaceflight Inc.
KNasu@spaceflight.com
Mobile:+1-206-384-0678

From: Siham <siham.aisha@temo-group.com>
Sent: Monday, November 15, 2021 1:36 AM
To: Keiko Nasu <knasu@spaceflight.com>
Subject: RE: [External] - Sales - Website Submission

CAUTION: This email originated from outside of the organization. Do not click links or open attachments unless you recognize the sender and know the content is safe.

hello,
I'm waiting for response.
another question: what is the procedure to launch a satellite ?

Regards,
Siham

On October 2, 2021 at 9:20:37 am +03:00, Siham <siham.aisha@temo-group.com> wrote:
Hello Keiko,

1- the time frame it's about in the middle of 2022
2- launch to the same inclination, I mean in the same orbit
3- our spacecraft don't have propulsion
4- size of our spacecraft is : 10X10X20 cm 2U for 2 satellites
5- mass is: 3Kg
6- the purpose: small pilot system for communication satellite.

Regards,
Siham

On September 29, 2021 at 1:34:32 am +03:00, Keiko Nasu <knasu@spaceflight.com> wrote:

Hello Siham,

Thank you so much for reaching out to us.

In order to figure out the launch options and pricing, could you provide below?

- What is the time frame you are looking to launch?
- What orbit do you want to launch your spacecraft to?

-> Do you mean to launch to Mid-inc? 45 degrees inclination, Or you would like to launch to the same inclination but do the plane phasing?

- Will your spacecraft have propulsion?
- What is the size of your spacecraft?

->2U x 2 satellites

- What is the mass of your spacecraft?
- What is the purpose of your spacecraft?

Thank you!

Keiko Nasu

Business Development, Spaceflight Inc.

KNasu@spaceflight.com

Mobile:+1-206-384-0678

8.11 Tasks and Responsibilities, Technical Documentation

Working Package	Responsible	Name of Technical Documentation	Department/ Stakeholder
	Samir	**System Design**	**IAP**
		ACDS Technical Report	**IAP**
	Rozan	AIS Clustering	**Marmara University, Istanbul, Faculty of Computer Science**
		Orbit and Altitude Specification	
	Abd	**COM/OBD**	
	Yahya	X-Ray Sensor	**IAP**
		Vibration damper	
		Thermal Isolation	

Packages:

1. CoreFlightSystem, on-board computer
2. Attitude control system
3. Telemetry and payload COM system, intersatellite communication
4. Ground station
5. Launching issues
6. AIS

Staff:

For 1: new Turkish bachelor student group

For 2: Raja

For 3: Abdurrahman

For 4: new Turkish bachelor students

For 5: Siham

For 6: Rozan

8.11.1 Summary of System Parts

MAGNETOMETERS - Magnetometers sense magnetic field strengths and direction. The measurements are compared to the Earth's magnetic field map (which is dependent on the spacecraft position) to determine the attitude. Moreover, it can only be used at low altitude orbits, where the magnetic field is strong enough.

8.11.2 Status of Hardware

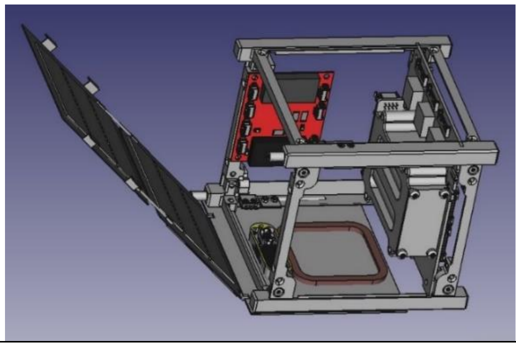
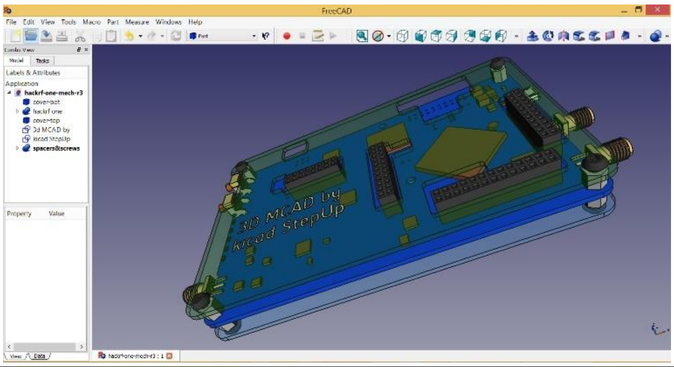
Hardware	Status
Onboard Computer	Received
Solar Cells	Received
Power System	Ordered – 10 Week Lead Time
Chassis	Ordered – Unknown Lead Time
ITC Designed Solar Panel PCBs	Designed – Out for Quote
Radio	Ordered – 6 Month Lead Time
Clean Room	Procured and Setup for Ribbon Cutting
Deployable Antenna	Ordered – Unknown Lead Time
Camera	Received

8.11.3 Project Documents & Databases for AS-COMSAT-1 (1 Satellite 10cmx10cmx10cm) (Last update: 8.4.21)

8.11.3.1 Development Documents

Development Phase	Name of Document	Purpose/Content
Analysis	Fire Detection System Description Presentation	about 130 pages Project Description, Supplier Parts
	Initial Cost Estimation	381,000 $ for satellite and ground station (including launch)
Systemdesign	System Design Document	
Mechanical Design		
Hardware Design	HackRF	
SW Specification	Software Specification Document (SDS)	on OBC and HackRF
SW Design	Software Design Document (SDD)	

8.11.3.2 Mechanical CAD Models

AS-COMSAT- 1 Integration	
Payload 1 (X-Raysensor)	9,5 cm x 2,5 cm x 7 cm
PV Cells and Controller	
Load Controller&Batteries	
Magnetorquer	
Sun Sensor (Photocell)	
On-Board-Computer (OBD)	Raspberry Pi 3
SDR (HackRF) (TT&C and Payload COM)	

8.12 Suppliers

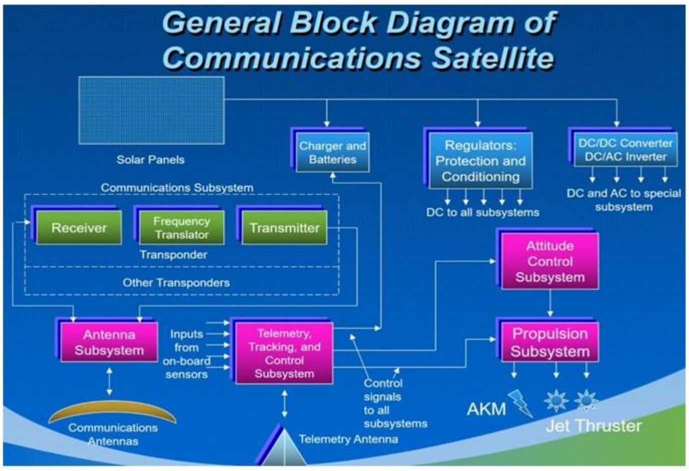
8.12.1 Satellite Parts

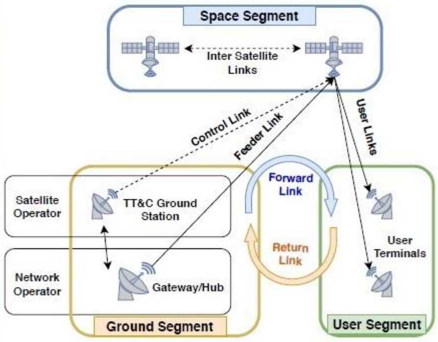
www.cubesatshop.com

8.13 Parts from Suppliers

8.13.1 2U Sommunication Satellite System

Communication Satellite System

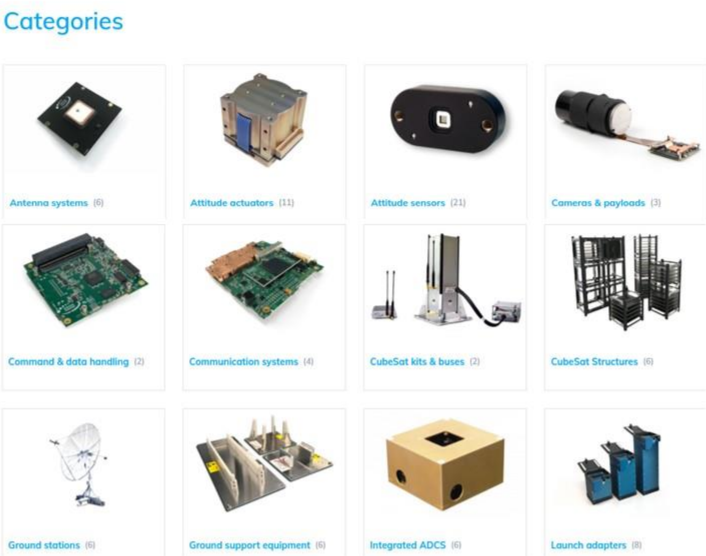




1/23/2022 AS-COMSAT, Jan-Feb 2021 14

Cubesat off-the-shelf satellite elements

Suppliers:
https://www.cubesatshop.com

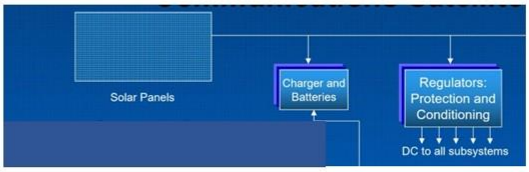


1/23/2022 AS-COMSAT, Jan-Feb 2021 15

8.13.2 SAT - Power Supply System

Power Supply

- Solar Panels
- Charger and Batteries
- Regulators
- DC/AC Converter

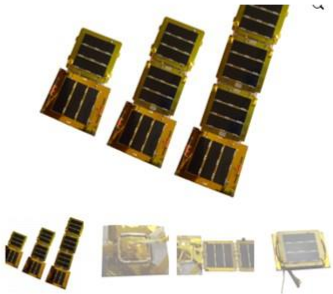


1/23/2022 AS-COMSAT, Jan-Feb 2021 16

Solar Panel Suppliers

1/23/2022 AS-COMSAT, Jan-Feb 2021 17

Solar Panel Suppliers



The EXA DSA/1A (Titanium Deployable Solar Array for 1U) is the entry-level product of a family of deployable solar arrays based on artificial muscles for CubeSats in the range of 1U to 6U. The arrays are composed by 5 panels, 3 on top 2 on bottom that attached to the CubeSat structure just as another solar panel and once in orbit deploys to full extension, includes deploy and release contact sensors and custom options are available on requests like sun and temperature sensors, 7-panel configuration and your choice of solar cells like our low-cost solar cells to AzurSpace 3G-30 for very high power missions.

Every array is tested and qualified in our own facilities and shipped with full reports, their very thin yet robust Titanium scaffold of only 0.25mm thickness allows a 1U CubeSat to pack as much power as a full 3U mission. The DSA/1UA yields the best results when coupled with high capacity batteries like our BA01/S and they are fully customizable to your mission needs.

Vendor: EXA

Configuration 1-Panel Array, deploys as 3 panels (2 on top side, 1

Extra Options None

Clear

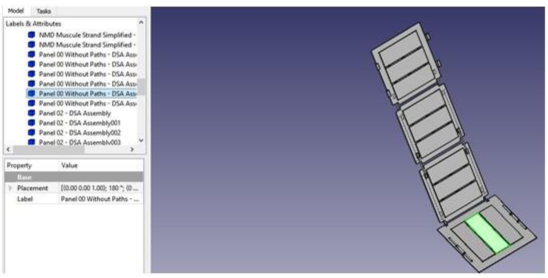
€3,500

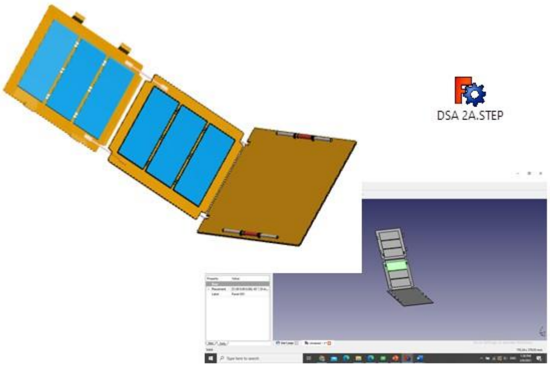
2 in top side, 1 on bottom side,
Low cost cells 2.75 W each

1/23/2022 AS-COMSAT, Jan-Feb 2021 18

EXA Solar Panel

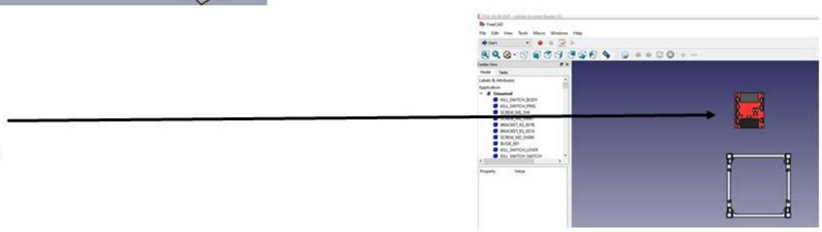
DSA 3A.STEP





DSA 2A.STEP

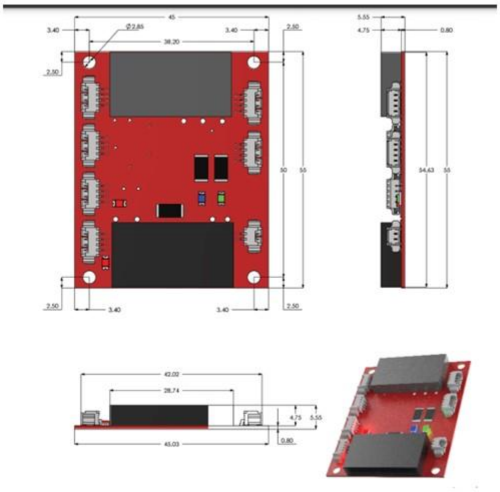
DSA Control.STEP



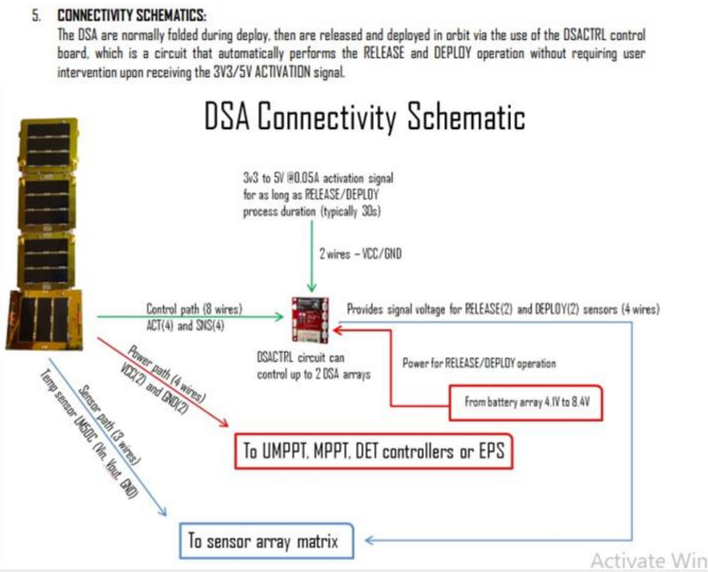
1/23/2022 AS-COMSAT, Jan-Feb 2021 19

EXA Solar Panel

DSA Control



EXA Solar Panel

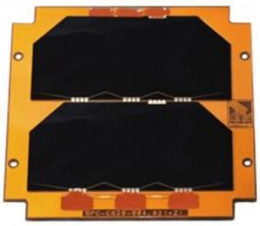


Solar Panel Suppliers

DHV 1U Solar Cell

Home / Solar panels / CubeSat Solar panel DHV-CS-10

Flight Heritage since 2015



€1,450

Solar panels have been tested in qualified laboratories for space applications, as well as solar cells are fully qualified. Different mechanical and electrical designs can be manufactured to meet with subsystems on board. The PCB substrates are made in conformance with ECSS-Q-ST-70-11C. Solar panels are manufactured following the standard ECSS-E-ST-20-08C.
DHV-CS-10 products are solar arrays for 1U CubeSats, there are available top, bottom and side versions. The solar cells are Azur Space 3G30C, qualified solar cells for space applications with 30% efficiency using triple-junction technology. Two solar cells are connected in series to get about 4.8 V. Magnetometer and temperature sensor are integrated on the PCB. Wires and connectors are included.
Availability: 4 – 5 weeks

Vendor: DHV Technology

1/23/2022 AS-COMSAT, Jan-Feb 2021 22

Solar Panel Suppliers

DHV 1U Solar Cell

Performance

Solar Cell String (Spectrum: AM0 WRC = 1367 W/m2; T = 28 °C; Series configuration)
- Type: TJ Solar Cell 3G30C – Advanced
- Base material: GaInP/GaAs/Ge on Ge substrate
- Open Circuit Voltage (Voc): 5.4 V
- Short Circuit Current (Isc): 0.52 A
- Voltage at max. Power (Vmp): 4.82 V
- Current at max. Power (Imp): 0.5 A
- Efficiency: 30 %

Temperature sensor:
- Manufacturer: MAXIM
- Model: LM75BIMM-3+
- Package: μMAX® (μSOP)
- Type: Temperature sensor /Over-temperature detector
- Supply Voltage: 3.3V
- Current: 4μA (Shutdown Mode), 250μA (typ), 1mA (max)
- Communication: I2C (configurable address)
- Conversion time: 100ms
- Accuracy: ±2°C
- Resolution: 9bits
- Temperature range: -55°C to +125°C

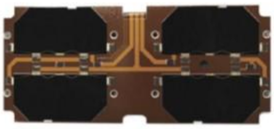
1/23/2022 AS-COMSAT, Jan-Feb 2021 23

Solar Panel Suppliers

1-Unit-Solarpanel 2500 EUR

CONFIGURATIONS

- Top/Bottom 1U panels
- On top of the ISIS AntS Antenna System
- 1U/2U/3U/6U configuration
- Deployable Panels (optional hold down release mechanism)
- Dummy panels with sensors without cells
- Custom configurations available on request



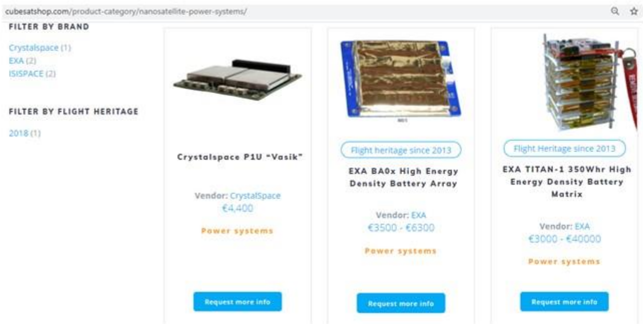
2U solar panels for CubeSats

1/23/2022 AS-COMSAT, Jan-Feb 2021 24

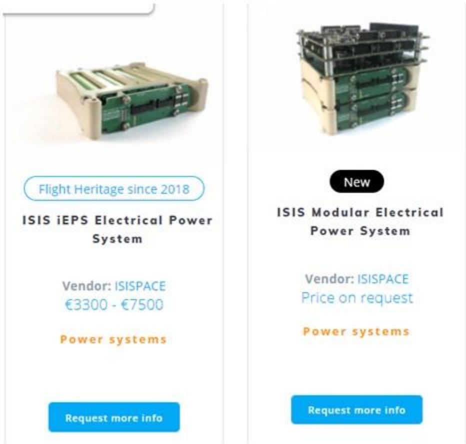
Charger and Batteries Suppliers

- EXA TITAN-1
- Price: 3000 EUR
- Supplier: EXA

- CrystalSpace:
- 1PU
- Price: 4400 EUR



1/23/2022 AS-COMSAT, Jan-Feb 2021 25



1/23/2022 AS-COMSAT, Jan-Feb 2021 26

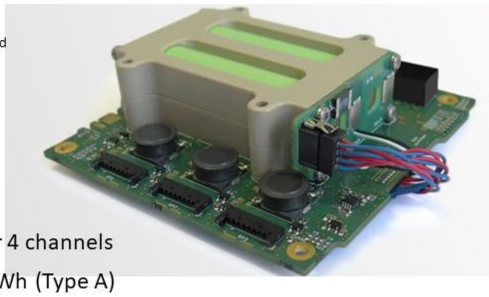
Electrical Power System (EPS)

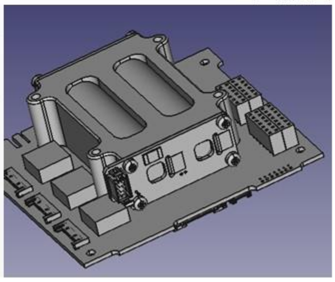
The ISIS Electrical Power System (iEPS) is the second-generation compact power system for nanosatellites, ideal for 1U up to 3U CubeSats. The system leverages wide bandgap semiconductor technologies, implementing GaN-FETs to improve solar power conversion efficiency and performance. It is equipped with an integrated heater, hardware-based Maximum Power Point Tracking (MPPT) and hardware voltage and over-current protection. The iEPS provides 3.3V and 5 V regulated buses, as well as an unregulated bus. An add-on daughter board allows additional configurations to suitably power the system and payload instruments.

Type A (2-cell battery pack): 4,600 EUR

icepsv2_type_a.stp

Compact single PC/104 form factor board solution
FRAM based MCUs for improved radiation tolerance
Hardware voltage, over-current protection and hardware-based maximum power point tracking
Designed for low (idle) power consumption
Solar Panel interface utilizes GaN-FETs
Allows customizations through mountable daughter-board





Power delivered	20W @ 5V over 4 channels
Energy storage	6300mAh/22.5Wh (Type A)

1/23/2022 AS-COMSAT, Jan-Feb 2021 27

Electrical Power System (EPS)

Flight Heritage since 2018

ISIS iEPS Electrical Power System

Vendor: ISISPACE
€3300 - €7500

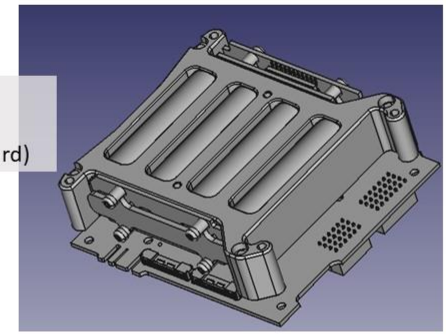
Power systems

The ISIS Electrical Power System (iEPS) is the second-generation compact power system for nanosatellites, ideal for 1U up to 3U CubeSats. The system leverages wide bandgap semiconductor technologies, implementing GaN-FETs to improve solar power conversion efficiency and performance. It is equipped with an integrated heater, hardware-based Maximum Power Point Tracking (MPPT) and hardware voltage and over-current protection. The iEPS provides 3.3V and 5 V regulated buses, as well as an unregulated bus. An add-on daughter board allows additional configurations to suitably power the system and payload instruments.

Type B (4-cell battery pack): 6,400 EUR

Mass	Type A 184 ± 5 grams (2 cell battery pack) Type B 310 ± 5 grams (4 cell battery pack) Type C 360 ± 5 grams (4 cell battery pack + daughterboard)

Power delivered	20W @ 5V over 4 channels
Energy storage	6300mAh/22.5Wh (Type A) 12800mAh/22.5Wh (Type B/C)

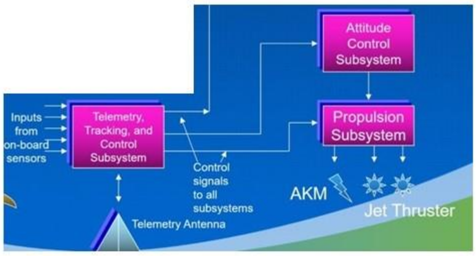


1/23/2022 AS-COMSAT, Jan-Feb 2021 28

8.13.3 SAT -Attitude Control System

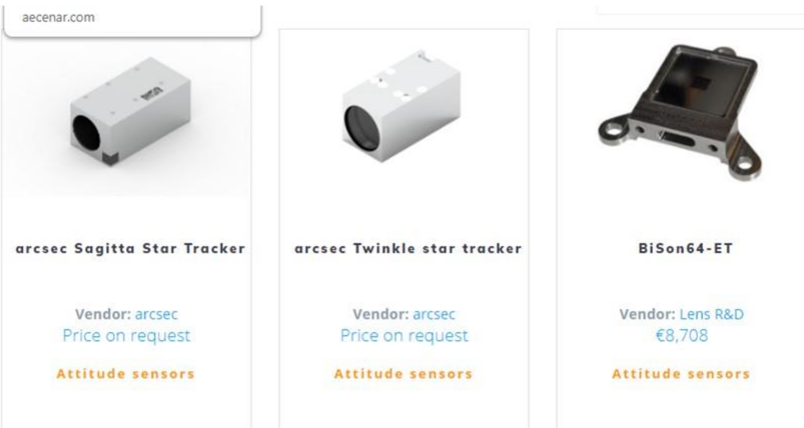
Satellite Bus and Attitude Control Sytem

- Antenna Subsystem
- Telemetry Tacking and Control Subsystem
- Attitude Control Subsystem
- Propulsion Subsystem



1/23/2022 AS-COMSAT, Jan-Feb 2021 29

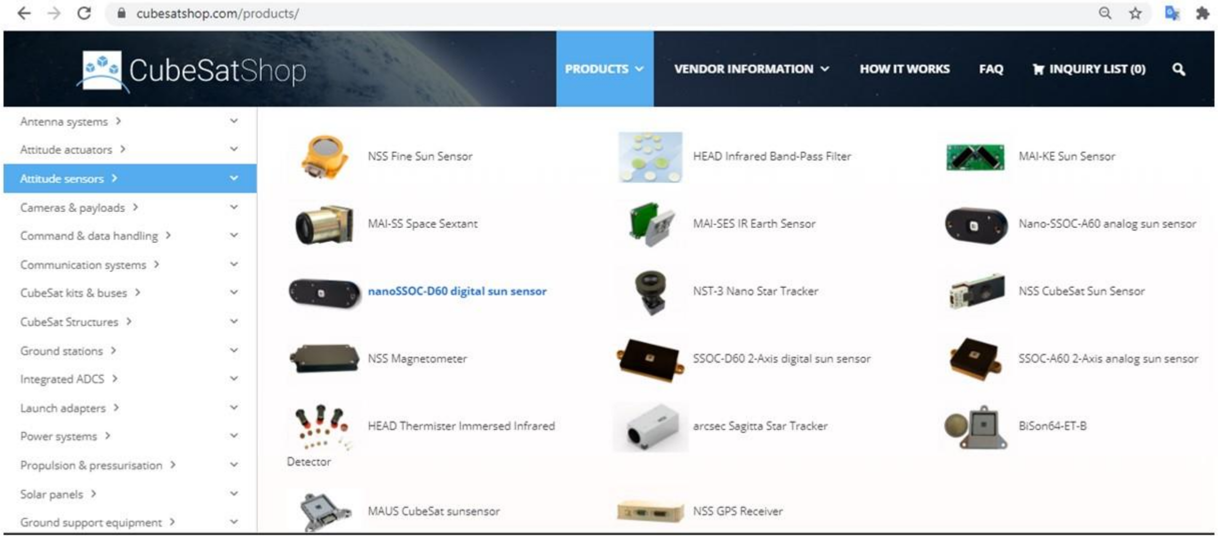
Attitude Control Subsystem



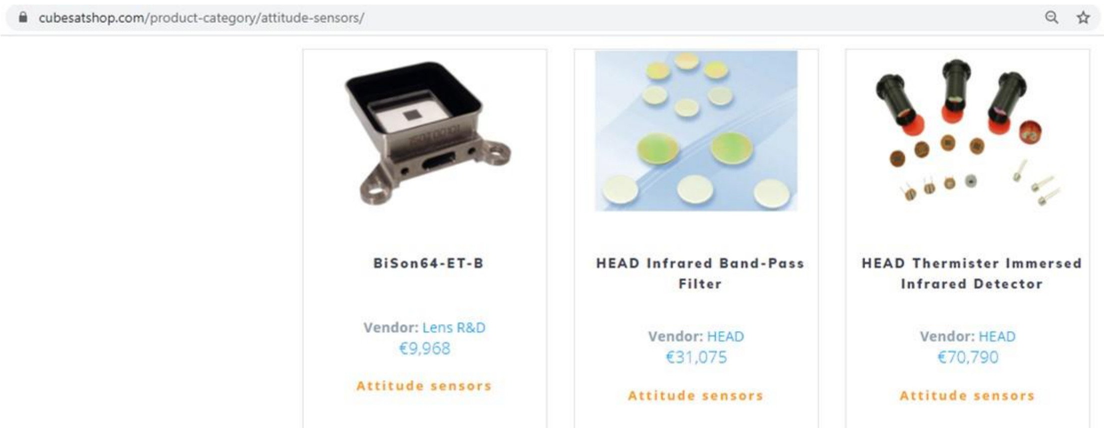
1/23/2022 AS-COMSAT, Jan-Feb 2021 30

Attitude Control Subsystem

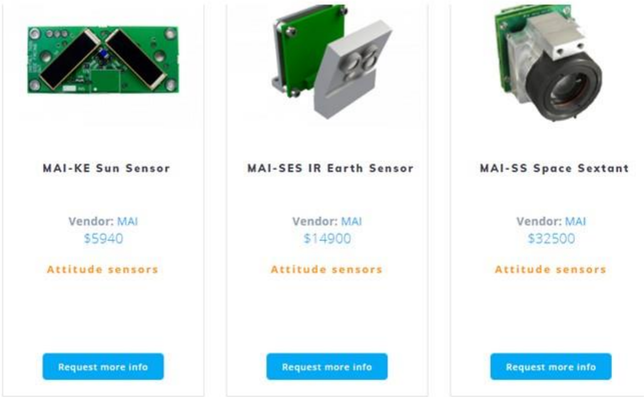
Attitude Sensors



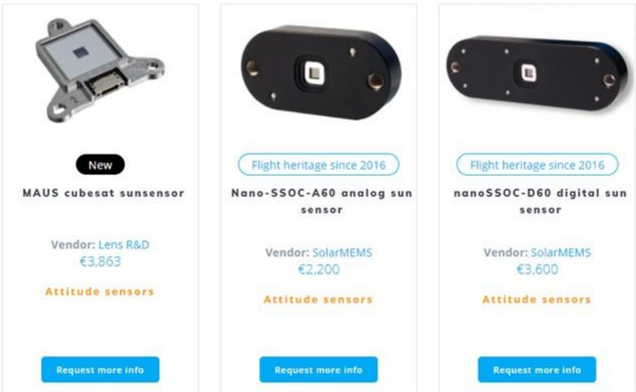
1/23/2022 AS-COMSAT, Jan-Feb 2021 31



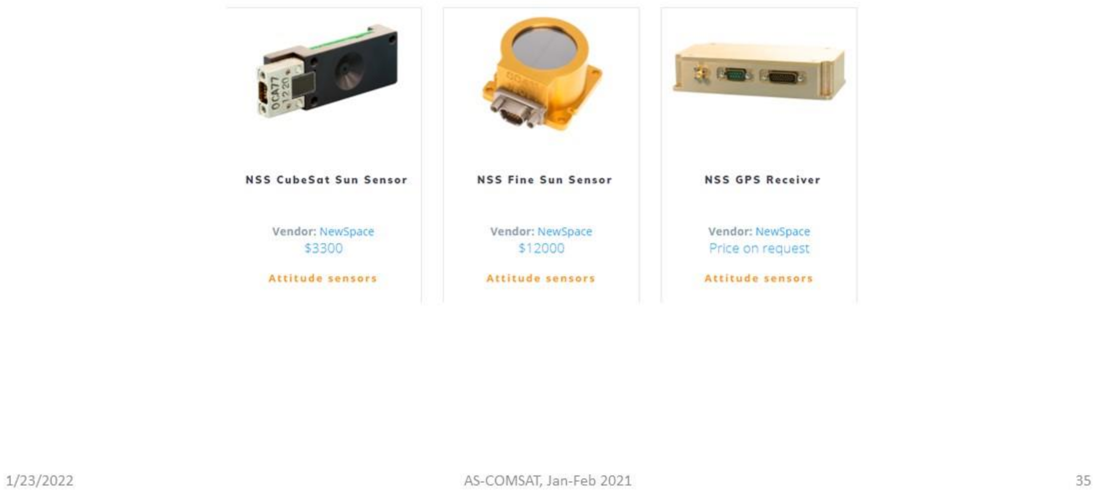
1/23/2022 AS-COMSAT, Jan-Feb 2021 32

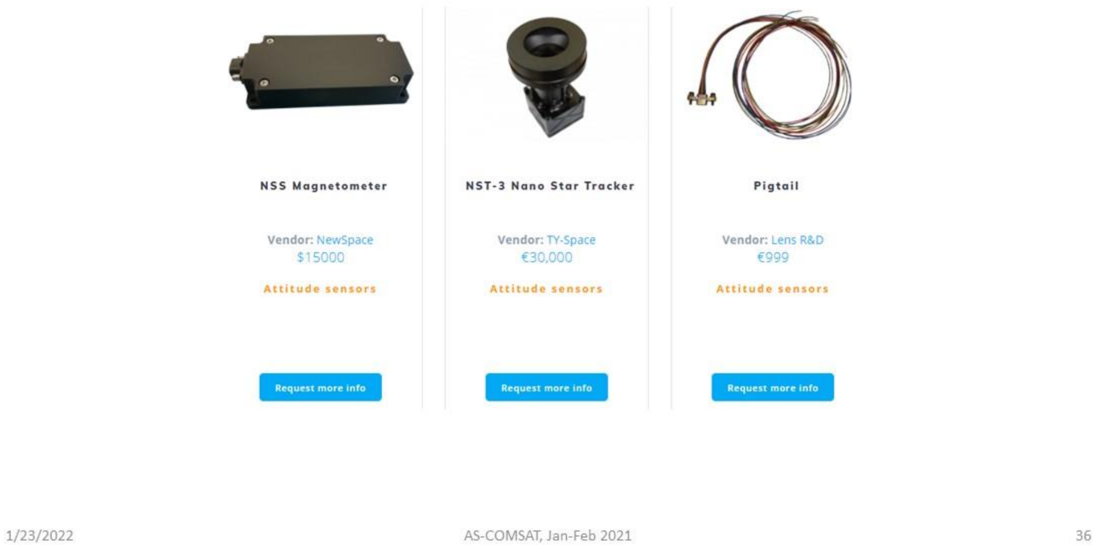


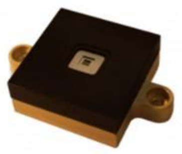
1/23/2022 AS-COMSAT, Jan-Feb 2021 33



1/23/2022 AS-COMSAT, Jan-Feb 2021 34


NSS CubeSat Sun Sensor
Vendor: NewSpace
$3300
Attitude sensors
NSS Fine Sun Sensor
Vendor: NewSpace
$12000
Attitude sensors
NSS GPS Receiver
Vendor: NewSpace
Price on request
Attitude sensors
1/23/2022
AS-COMSAT, Jan-Feb 2021
35


NSS Magnetometer
Vendor: NewSpace
$15000
Attitude sensors
Request more info
NST-3 Nano Star Tracker
Vendor: TY-Space
€30,000
Attitude sensors
Request more info
Pigtail
Vendor: Lens R&D
€999
Attitude sensors
Request more info
1/23/2022
AS-COMSAT, Jan-Feb 2021
36



Flight heritage since 2009

SSOC-A60 2-Axis analog sun sensor

Vendor: SolarMEMS
€7,200

Attitude sensors

Flight heritage since 2009

SSOC-D60 2-Axis digital sun sensor

Vendor: SolarMEMS
€12,200

Attitude sensors

Transport box

Vendor: Lens R&D
€159

Attitude sensors

1/23/2022 AS-COMSAT, Jan-Feb 2021 37

Attitude Control Subsystem

Attitude Sensors

SOLARMEMS

nanoSSOC-D60
Sun Sensor for Nano-Satellites
Digital Interface

ITAR FREE

Technical Chracteristics:

Type	2 orthogonal axes
Field of View	±60°
Accuracy	< 0.5 ° (3sigma) < 0.1 ° (precision)
Electrical interface	UART, I2C or SPI 10-pin micro-connector
Power supply	3.3V / 5V < 23mA consumption
Mechanical interface	43 x 14 x 5.9 mm 6.5 g
Housing	Aluminum 6082 Black anodizing

Digital
Space Qualified

Sun Sensor on a Chip (SSOC)

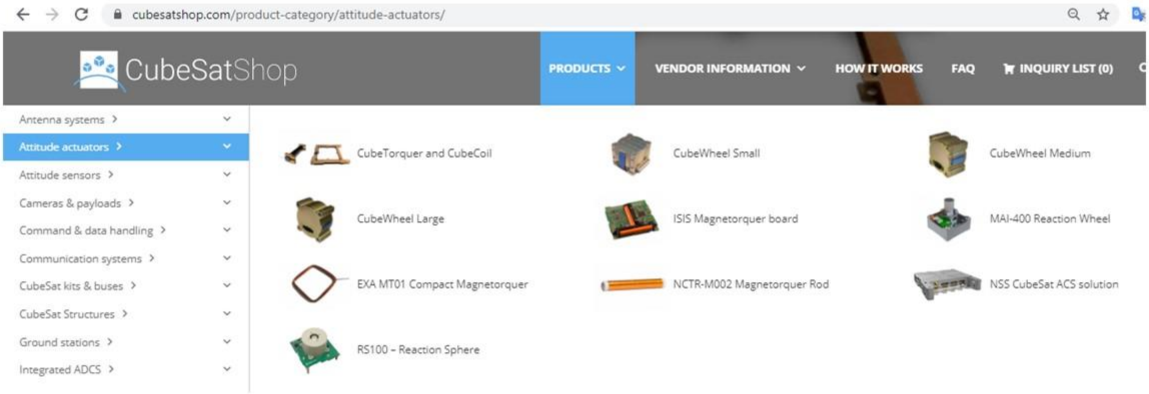
nanoSSOC-D60 digital sun sensor

€3,600

Sun Sensor on a Chip (SSOC) is a two-axes and low cost sun sensor for high accurate sun-tracking, pointing and attitude determination. The device measures the incident angle of sun ray in two orthogonal axes, providing a high sensitivity based on the geometrical dimensions of the design. nanoSSOC sun sensor is based on MEMS fabrication processes to achieve high integrated sensing structures.
nanoSSOC-A60 has tiny size, low weight and low power consumption to be the perfect ADCS solution for nanosatellite platforms like Cubesats.

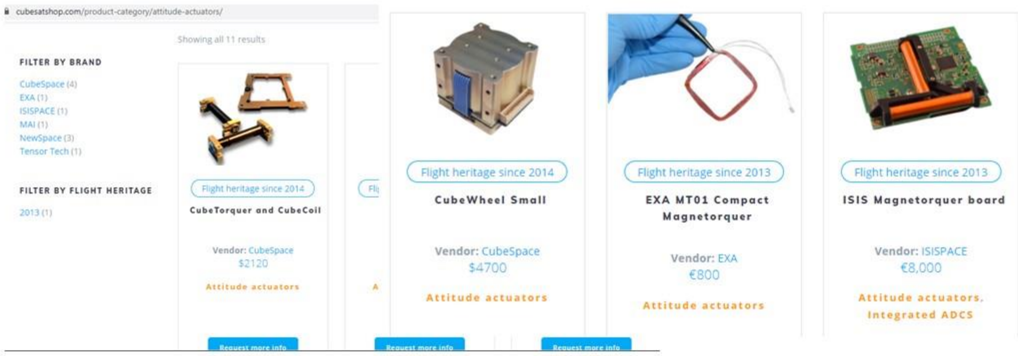
1/23/2022 AS-COMSAT, Jan-Feb 2021 38

Attitude Actuators



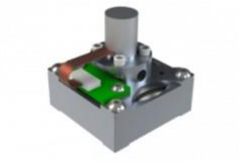
1/23/2022 AS-COMSAT, Jan-Feb 2021 39

Attitude Actuators



1/23/2022 AS-COMSAT, Jan-Feb 2021 40

Attitude Actuators



MAI-400 Reaction Wheel

Vendor: MAI
$7100

Attitude actuators

NCTR-M002 Magnetorquer Rod

Vendor: NewSpace
$1200

Attitude actuators

NCTR-M012 Magnetorquer Rod

Vendor: NewSpace
$2000

Attitude actuators

1/23/2022 AS-COMSAT, Jan-Feb 2021 41

Attitude Actuators

Home / Attitude actuators / EXA MT01 Compact Magnetorquer

€800

With only 7.5 grams and 3.2 millimeters thickness, the MT01 Compact Magnetorquer is a vacuum core magnetic coil designed fo ADCS control in cubesat mission from 1U to 3U that boast an impressive performance compared to its small footprint over the mass, power and area budget of the spacecraft. Even with that small dimensions the MT01 is capable of greater magnetic moments, turn speeds and angular accelerations than comparable products on the market, yet the power usage is kept to a minimum: It can turn a 1U mass 90 degrees in 60 seconds using only 0.2 Watts at a LEO orbit of 500kms.

MT01 can be integrated in to our BA0x family of high capacity compact batteries and our DSA Deployable Solar Array family too, the biggest advantage of the MT01 is that it can be easily affixed anywhere on your spacecraft using a minimal area.

Every coil is tested and qualified in our own facilities and shipped with full reports and packed with additional match connectors interfaces.
Availability: 2 weeks

Vendor: EXA

1/23/2022 AS-COMSAT, Jan-Feb 2021 42

Attitude Actuators

EXA Magnetorquer

Performance

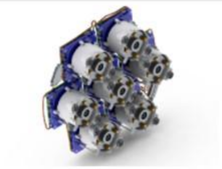
- •Working Voltage: From 1.25V to 7.5V
- •Working Current: From 100mAh to 2000 mAh
- •Nominal Magnetic moment: >0.19 Am2
- •Saturation Magnetic moment: >0.85 Am2
- •Linearity: +/- 4% across operating design rang
- •Residual moment: <0.0045 Am2
- •Torque: 5.36 μNm @ 7.2-3 Tesla (1U mass)
- •Angular acceleration: 3.2-3 Rad/sec-2 (1U mas
- •B-center = 8.9 Gauss
- •B-corners = 14.5 Gauss
- •Supply Power: From 250mW to 1750mW
- •Typical resistance: 4.1 to 4.7 ohms @ 25°C
- •Random Vibration: 16g rms
- •Lifetime: >10 years

Features | Performance | Product properties | Materials | Testing | Configurations | Documents

Configuration	Price
MT01 Compact Magnetorquer 50x50x3.2 mm	€ 800
Optional: Integrated to BA0x High Capacity Battery Array	- € 100
Optional: Integrated to DSA Deployable Solar Array	- € 100
Optional: Integrated thermal sensor	+ € 150

1/23/2022 AS-COMSAT, Jan-Feb 2021 43

Propulsion Unit (Thruster)



Cluster of IFM Nano Thruster for Smallsats

Vendor: ENPULSION
€50000 - €210000

Propulsion & pressurisation

Flight Heritage since 2018

IFM Nano Thruster for CubeSats

Vendor: ENPULSION
€38,400

Propulsion & pressurisation

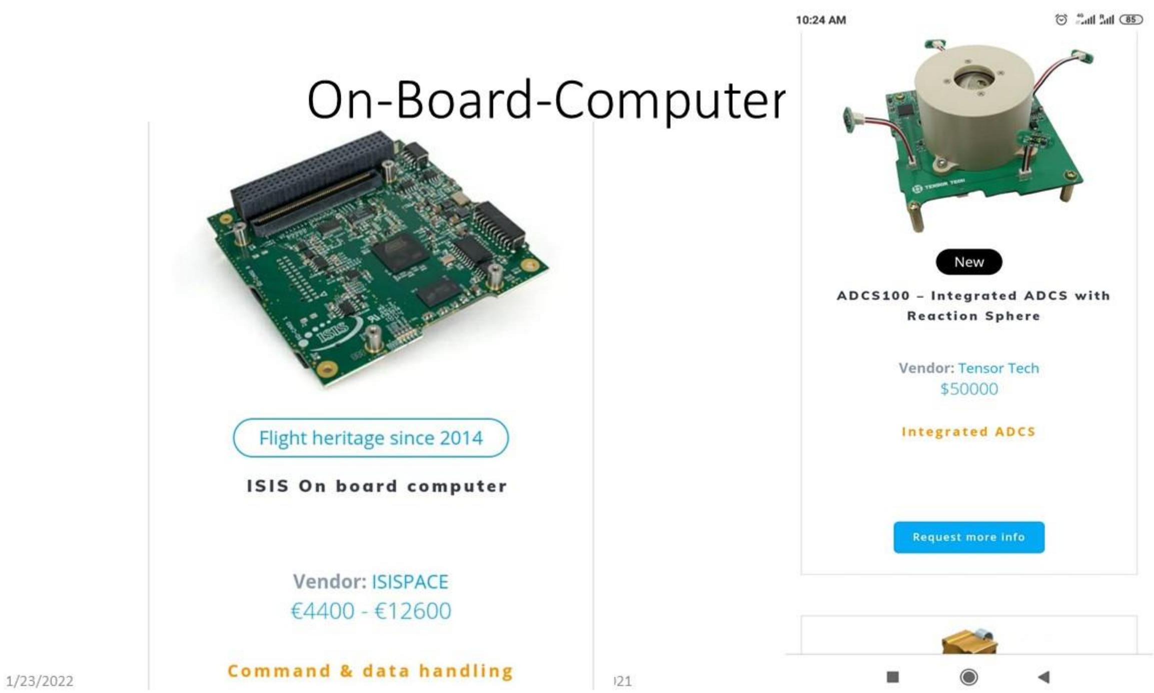
Nanosatellite Micropropulsion System

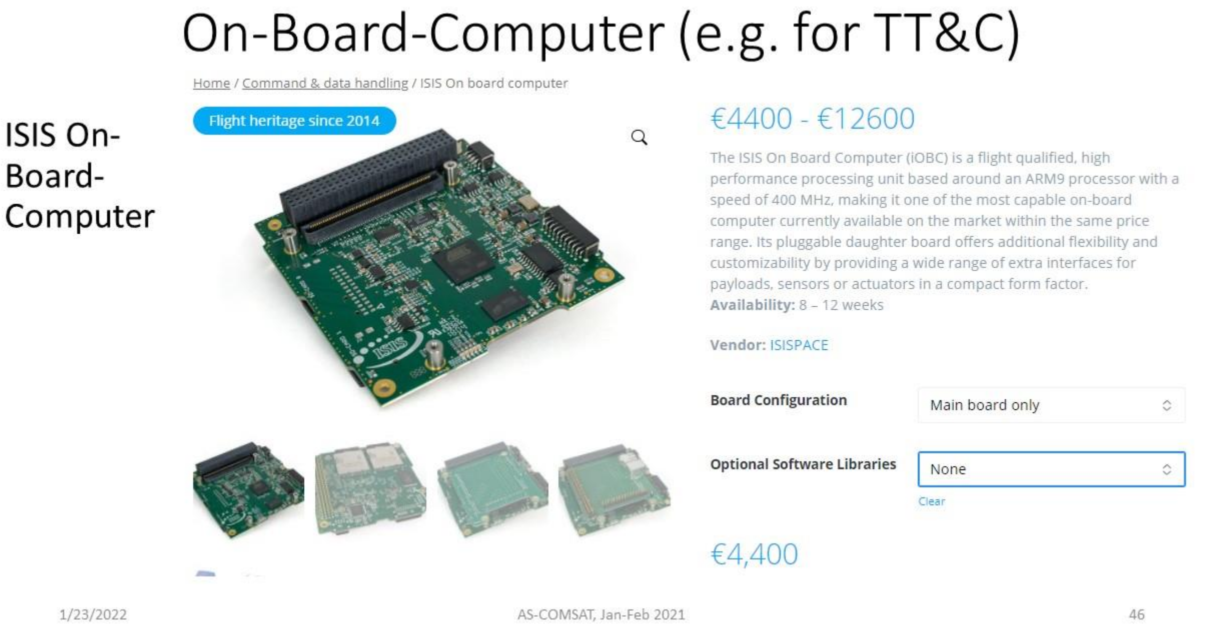
Vendor: MicroSpace
€81000 - €129000

Propulsion & pressurisation

1/23/2022 AS-COMSAT, Jan-Feb 2021 44

8.13.4 Command&Data Handling (On-Board Computer)





On-Board-Computer

Flight heritage since 2014

ISIS On board computer

Vendor: ISISPACE
€4400 - €12600

Command & data handling

The ISIS on-board computer (iOBC) is a flight proven, high performance processing unit based around an ARM9 processor with a clock speed of 400 MHz and offers a multitude of standardized interfaces. Combined with its daughterboard architecture, allowing for easy addition of mission specific electronics or interfaces, this makes the iOBC the ideal candidate for your main mission computer or payload processing unit.

FEATURES

- 400 MHz, power efficient ARM9 processor
- Multiple OS options available:
  → FreeRTOS operating system for simple and lightweight cooperative multitasking
  → KubOS Linux (coming in 2017)
- On-board telemetry: voltages, currents, and temperature
- External on-board watchdog, power-controller, and real time clock
- High reliability data storage and fail safe filesystem
- Flexible daughterboard architecture
- Robust design

INTERFACES

- I²C master or slave mode
- SPI master mode (up to 8 slaves)
- 2x UART (RS232 + RS232 / RS485 / RS422)
- General Purpose Input / Output pins (GPIO)
- ADC (10-bit, 8 channels)
- Pulse Width Modulation (PWM)
- JTAG for programming and debugging
- Dedicated debug LEDs and UART
- USB host and device
- Image Sensor Interface

1/23/2022 AS-COMSAT, Jan-Feb 2021 47

On-Board-Computer

Flight heritage since 2014

ISIS On board computer

Vendor: ISISPACE
€4400 - €12600

Command & data handling

PRODUCT PROPERTIES

- Operating Temperature:-25 °C to +65 °C
- Power Supply: 3.3V
- Dimensions: 96 x 90 x 12.4mm (incl. FM daughter board)
- Mass: 76g mainboard only, 100g with EM daughter board
- Power Consumption: 400mW average

DAUGHTERBOARD ARCHITECTURE

- The pluggable daughterboard offers flexibility and customizability by providing a wide range of interfaces for payloads, sensors, actuators in a compact form factor
- EM daughterboard: all interfaces for development and debugging
- FM daughterboard: all interfaces in compact form factor using high reliability connectors
- Custom daughterboard: design your own daughterboard with additional interfacing and electronics based on mission requirements

AVAILABLE SOFTWARE LIBRARIES

SD card slots

EM Daughter Board

1/23/2022 AS-COMSAT, Jan-Feb 2021 48

On-Board-Computer

AVAILABLE SOFTWARE LIBRARIES

iOBC Hardware Abstraction Layer library (included)
- Library that supports all iOBC hardware peripherals and includes FreeRTOS and the fail-safe FAT32 filesystem

Subsystems interface library (optional)
- Library for interfacing the iOBC with the most commonly used satellite subsystems over the I²C databus

Mission Support library (optional)
- Library providing high level mission software functionality, includes flight parameter storage and logging modules

Custom solutions on request

iOBC with FM Daughter Board

PRODUCT CONTENTS AND ACCESSORIES

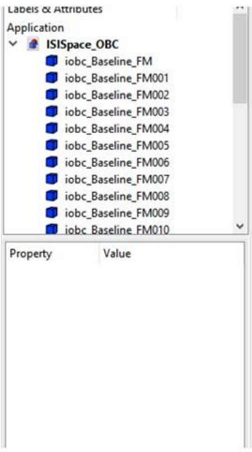
- iOBC main board
- JTAG programmer / debugger + USB cable
- Adapter board (including debug UART to USB conversion) + USB cable
- 2 x 2 GB high reliability SD cards
- Power break-out board for easy connection to power supply
- USB drive containing manuals, iOBC SDK installer, and applicable software libraries

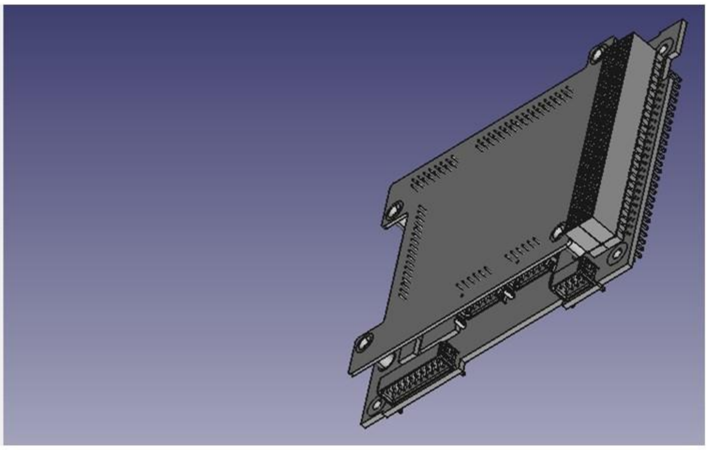
1/23/2022 AS-COMSAT, Jan-Feb 2021 49

On-Board-Computer

CAD File:

iobc_Baseline_FM.zip





1/23/2022 AS-COMSAT, Jan-Feb 2021 50

On-Board-Computer (from ISIS)

ISIS On Board Computer
€4.400,00 – €6.850,00
•400 MHz, power efficient ARM9 processor
•Multiple OS options available:
- FreeRTOS operating system for simple and lightweight cooperative multitasking
- KubOS Linux

•On-board telemetry: voltages, power-controller, and real time clock
•High reliability data storage and fail safe filesystem
•Flexible daughterboard architecture
•Robust design
•Includes Hardware Abstraction Layer Library

1/23/2022 AS-COMSAT, Jan-Feb 2021 51

On-Board-Computer

Flight heritage since 2014

Cube ADCS

Vendor: CubeSpace
$24700 - $37400

Integrated ADCS

$24700 - $37400

CubeADCS bundles offer custom solutions for a variety of satellite requirements. The ADCS OBC can also serve as a main satellite OBC. Each bundle consists of an integrated stack of CubeSpace components with UART, I2C, and CAN interfaces to other satellite subsystems. High-level ADCS software libraries are also available on any CubeADCS bundle. The bundles are compact and have low power consumption.

Vendor: CubeSpace

1/23/2022 AS-COMSAT, Jan-Feb 2021 52

On-Board-Computer inclusive other components

New

EXA ICEPS Spacecraft System Core

Vendor: EXA

€16500 - €53000

All in one USB 2.0 bus OBC, SDR radio, Laser comms, EPS, Solar panel manager, Deployables manager Cubesat system core

1/23/2022 AS-COMSAT, Jan-Feb 2021 53

On-Board-Computer inclusive other components

All in one USB 2.0 bus OBC, SDR radio, Laser comms, EPS, Solar panel manager, Deployables manager Cubesat system core

Configuration: Choose an option

Choose an option

- 1 - FULL EPS ONLY (I2C INTERFACE) + 25W BATTERY (1 single deck)
- 2 - FULL EPS (USB/I2C)+ OBC/SDR RADIO/32GB SSD
- 3 - FULL EPS (USB/I2C) + OBC/SDR RADIO/256GB SSD + LASER COMMS
- 4 - FULL EPS (USB/I2C) + OBC/SDR RADIO/256GB SSD + LASER COMMS + D/R CONTROL
- 5 - FULL EPS (USB/I2C) + OBC/SDR RADIO/512GB SSD + LASER COMMS + D/R CONTROL + 25W BATTERY
- 6 - FULL EPS (USB/I2C)+ OBC/SDR RADIO/512GB SSD + LASER COMMS + D/R CONTROL + 50W BATTERY (2 single deck or 1 double deck)
- 7 - FULL EPS (USB/I2C)+ OBC/SDR RADIO/512GB SSD + LASER COMMS + D/R CONTROL + 100W BATTERY (2 double deck)

Configuration: 2 - FULL EPS (USB/I2C)+ OBC/SDR RADIO/32GB SSD

Clear

€29,000

1/23/2022 AS-COMSAT, Jan-Feb 2021 54

On-Board-Computer inclusive other components

New

EXA ICEPS Spacecraft System Core

Vendor: EXA
€16500 - €53000

ICEPS System Core Specifications:

Bus type:	USB 2.0, I2C
OBC:	Xilinx Zynq XC7Z010-2I System-on-Chip: Programmable Logic (PL) Specification: -28K Logic Cells -2.1 Mbits BlockRAM -80 DSP slices Processor System (PS) Specification: -Dual-core ARM Cortex A9 CPU running up to 733 MHz -Linux 4.11 • 512 MB of DDR3L RAM • 32 MB of QSPI flash storage for uboot bootloader, kernel, and root file system Temperature Sensor: TMP103AYFFR Accuracy: -40 deg C to +125 deg C (+/- 1 deg C typ), Resolution: 1 deg C
OBC/OS:	Linux computer running IIOS
Radio:	SDR from 70Mhz to 6GHz
EIRP	+28.5 dBm integrated LNB
Sensitivity:	-110 dBm
Antenna ports:	2 RX and 1 Transceiver (TX/RX)
System Storage	From 32 to 512 Gigabytes 60MB/s r/w (user chooses)
Number of ports:	USB2.0 60MB/s 14 total: 6 external, 8 internal; 1 I2C port
High Speed Laser communications	405nm or 450 nm solid state laser, temperature stabilized , 10 Mbps, variable beam aperture, variable focus. Integrated temperature sensor range -50C to +125C
Power rails:	5V@3A; 12V@3A(adjustable), 3v3@3.6A(2 redundant); 4.2 ~ 3.6 (unregulated) @ 12A ; 1 auxiliary APU port
Battery packs	One or two at 3.7@6A, 50W max nominal
Power delivery	50W Nominal (continuously) , 65W Maximum, 100W peak for 2.5 sec.
Solar mgmt:	4 UMPPT channels 16V@2A max each
Solar charger:	Based on TPS100 2A continuously, 1S, 2S,3S
Internal sensors:	20 internal sensors; integrated IMU
Actuators:	Integrated automatic management of Release/deploy mechanisms ; integrated automatic LNB/PA switching
Built in protection	RBL; 10A Activation switch w/ MTBF>1000, 2A and 7A resettable fuses
Inertial Measurement Unit:	6-axis MotionTracking Device: (3-axis gyroscope, 3-axis accelerometer) TDK / InvenSense ICM-20602 - Gyroscope sensitivity error: ±1% - Gyroscope noise: ±4 mdps/√Hz - Accelerometer noise: 100 µg/g/√Hz
Mass	100 grams
Operating temp.	-50C to +125C
Dimensions	96x96x25mm

- The system is modular, this is a full description, but the user can choose what components to be added or subtracted.

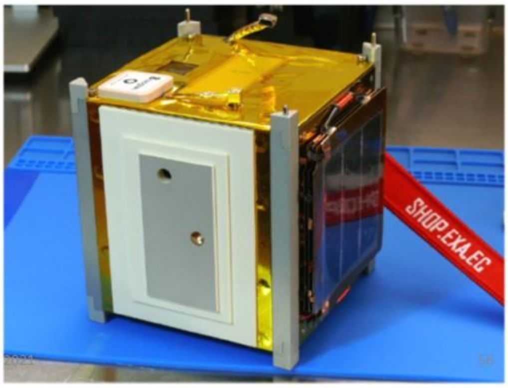
1/23/2022 AS-COMSAT, Jan-Feb 2021 55

On-Board-Computer inclusive other components

New

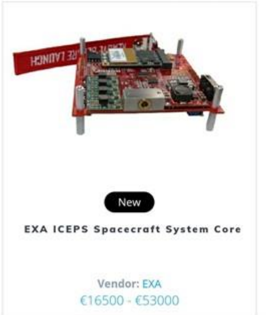
EXA ICEPS Spacecraft System Core

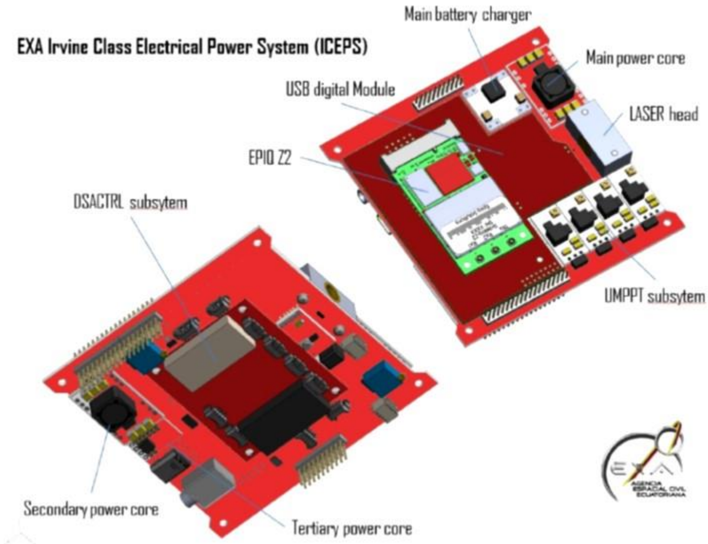
Vendor: EXA
€16500 - €53000



1/23/2022 AS-COMSAT, Jan-Feb

On-Board-Computer inclusive other components



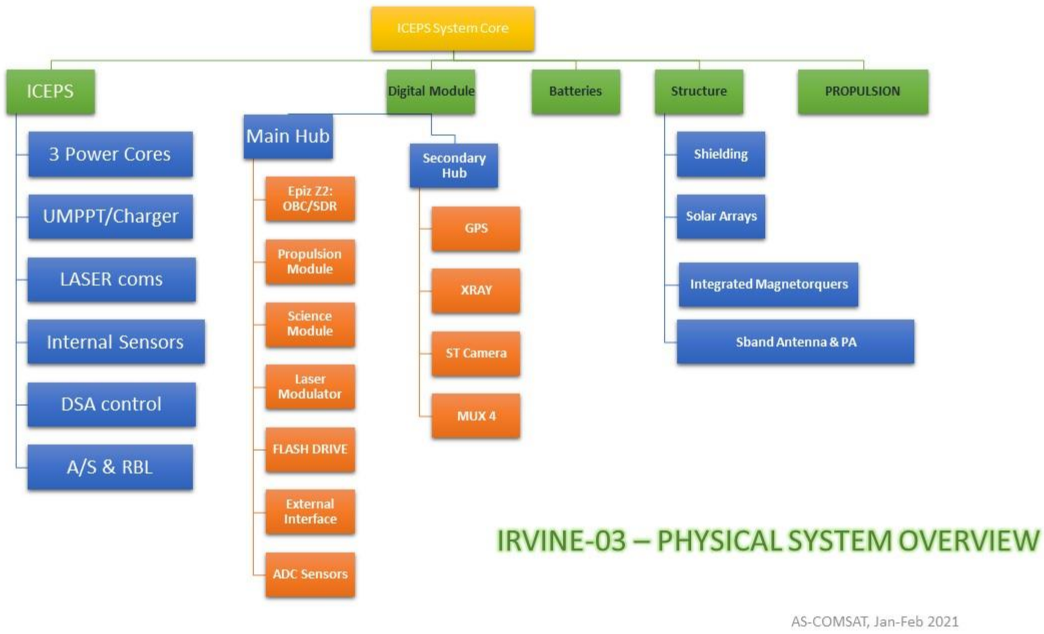


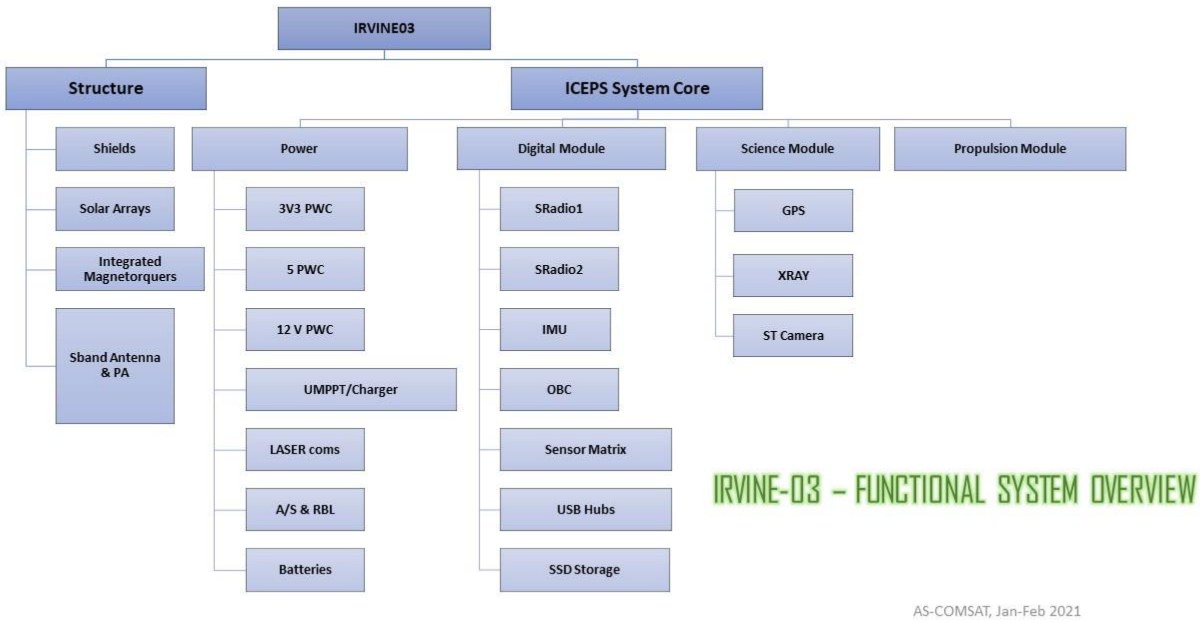
1/23/2022 AS-COMSAT, Jan-Feb 2021 57

DESIGN:

A. USB architecture
B. Solar charge (UMPPT)
C. Laser Communications Module
D. Power Supply
E. High Capacity SSD Storage
F. Epiq Sidekick Z2 OBC
G. Software Defined Radio
H. DSA Deployment Control
I. Sensors and actuators

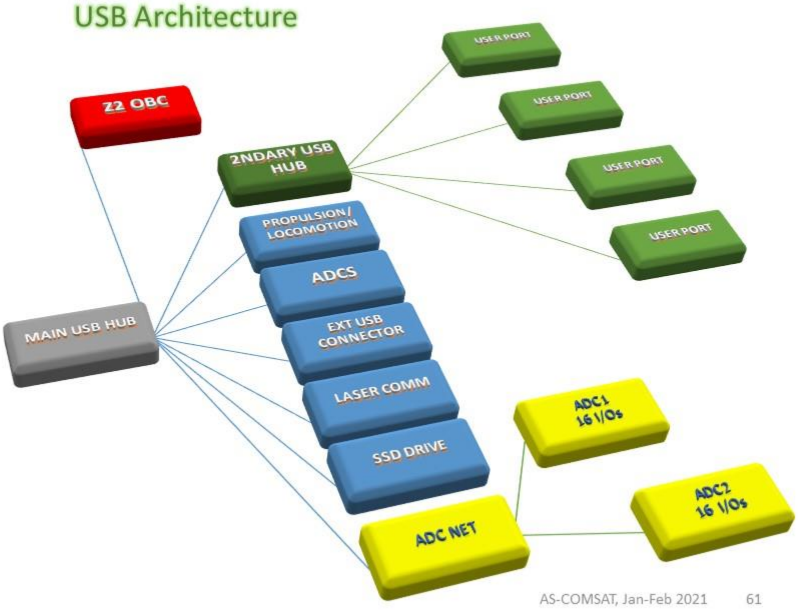
AS-COMSAT, Jan-Feb 2021 58


ICEPS System Core
ICEPS
Digital Module
Batteries
Structure
PROPULSION
3 Power Cores
UMPPT/Charger
LASER coms
Internal Sensors
DSA control
A/S & RBL
Main Hub
Epiz Z2: OBC/SDR
Propulsion Module
Science Module
Laser Modulator
FLASH DRIVE
External Interface
ADC Sensors
Secondary Hub
GPS
XRAY
ST Camera
MUX 4
Shielding
Solar Arrays
Integrated Magnetorquers
Sband Antenna & PA
IRVINE-03 – PHYSICAL SYSTEM OVERVIEW
AS-COMSAT, Jan-Feb 2021


IRVINE03
Structure
ICEPS System Core
Shields
Solar Arrays
Integrated Magnetorquers
Sband Antenna & PA
Power
3V3 PWC
5 PWC
12 V PWC
UMPPT/Charger
LASER coms
A/S & RBL
Batteries
Digital Module
SRadio1
SRadio2
IMU
OBC
Sensor Matrix
USB Hubs
SSD Storage
Science Module
GPS
XRAY
ST Camera
Propulsion Module
IRVINE-03 – FUNCTIONAL SYSTEM OVERVIEW
AS-COMSAT, Jan-Feb 2021

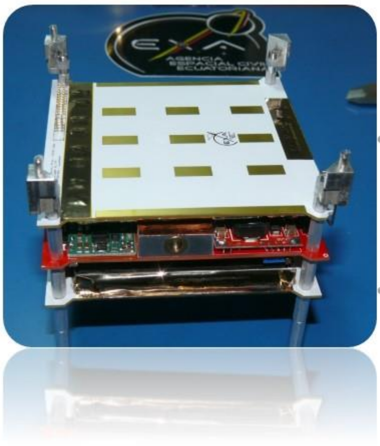
USB Architecture

- From I2C/SPI to USB 2.0
- The USB core has up to 14 ports, 8 of them used by ICEPS, 6 available for user.
- The OBC is on upstream port of main USB hub
- The ADC network has room for 32 GPIOs
- USB-C connector is backward compatible and normally uses USB 2.0 protocol.

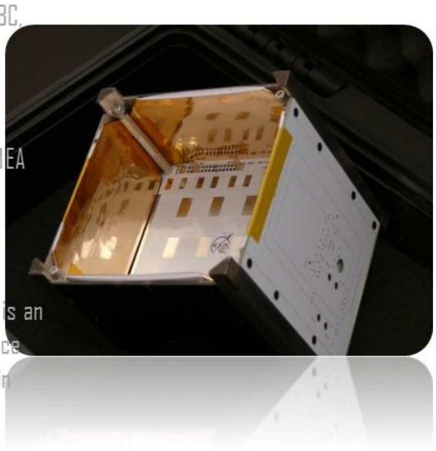


AS-COMSAT, Jan-Feb 2021 61

USB bus resilience against radiation



- To mitigate the impact of SLEs and SUEs in the OBC, the ICEPS board is encased between 2 BA01/S battery packs
- The second radiation protection is the SEAM/NEMEA shielding MLI that protects all the sides of the cubesat.
- The Space Environment Attenuation MLI (NEMEA) is an MLI of 27 layers that has spaceflight heritage since 2013 and has been demonstrated very effective in regulating radiation, temperature and plasma environments.



AS-COMSAT, Jan-Feb 2021 62

Solar charge using Unified Maximum Power Point Tracking

- The power generation architecture of IRVINE-03 includes 2 Deployable Solar Arrays (DSA), each one with two nanomorphodynamic actuator sets
- The DSAs have two sides and each one is equipped with solar cells, hence the need for four UMPPT channels, in order to harness as much power as possible without sacrificing the other partially illuminated side.
- The UMPPT bus to which the channels are connected feed the charging circuit to which the battery arrays are directly connected.

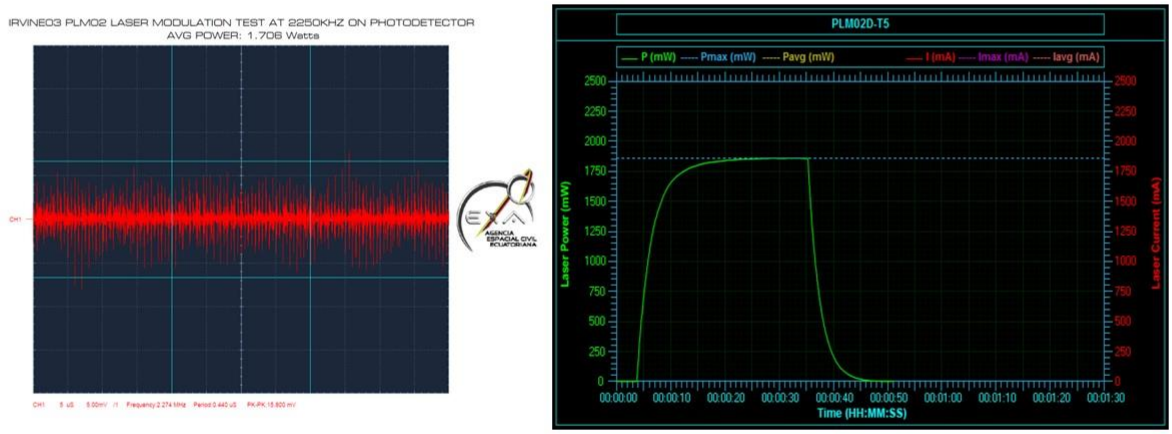
AS-COMSAT, Jan-Feb 2021 63

Laser Communications Module PLM02

- This laser module has higher data transfer rates that can handle large payload data sizes and can potentially achieve downloads as big as 100MB in 1 to 5 minutes.
- The laser communications module can be set to transmit as low as 115Kbps and as high as 10Mbps.
- The wavelength of the communications laser is 450nm or 405nm.

AS-COMSAT, Jan-Feb 2021 64

PLM-02 Power plot modulation 2.25Mbps at 1.7 W



AS-COMSAT, Jan-Feb 2021 65

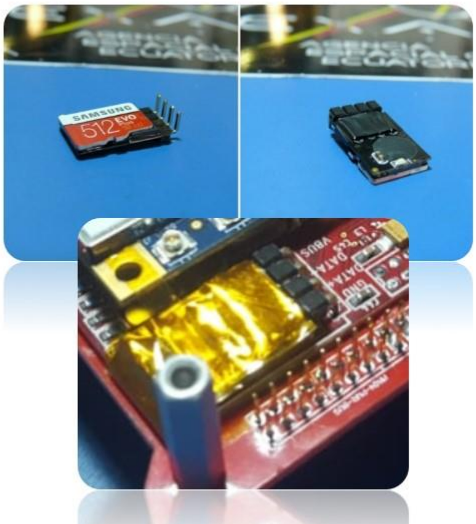
ICEPS USB 2.0 Satellite System Core

PLM02: Pointing accuracy vs. Power flux density

1/23/2022 AS-COMSAT, Jan-Feb 2021 66

512GB of high speed, radiation tolerant SSD storage

- ICEPS can be fitted with a minimum of 32GB and a maximum of 512GB of high speed (60 MB/s read/write) USB 2.0 SSD storage, which is natively radiation tolerant made by Samsung.
- In addition to that native tolerance, the SSD is protected by its own radiation shielding, encased between 2 battery banks and protected by the NEMEA shielding.
- The SSD is connected to a high priority USB 2.0 channel, so it can drive at a top speed of 480Mbps if needed.

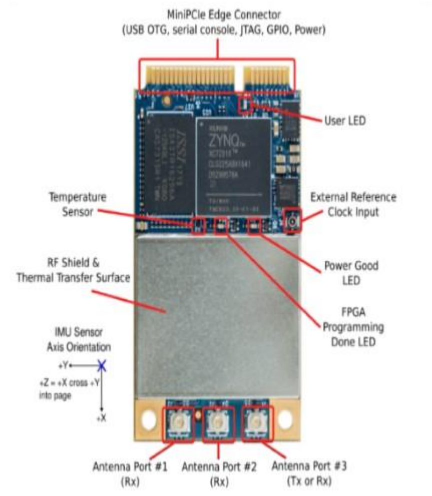


AS-COMSAT, Jan-Feb 2021 67

Epiq Sidekiq Z2 OBC

ICEPS computing core is an Epiq's Sidekiq Z2:

- Dual-core ARM Cortex A9 CPU running up to 733 MHz
- Linux computer running IIOS
- 512 MB of DDR3L RAM
- 32 MB of QSPI flash storage for uboot bootloader, Linux kernel, and root file system
- USB 2.0 main network port
- Internal independent watchdog and temp. sensor
- Internal IMU, 6 axis sensor, ADCS ready

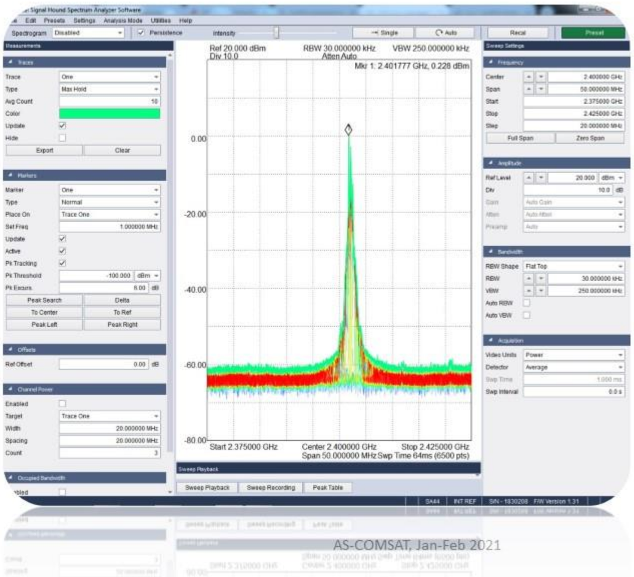


AS-COMSAT, Jan-Feb 2021

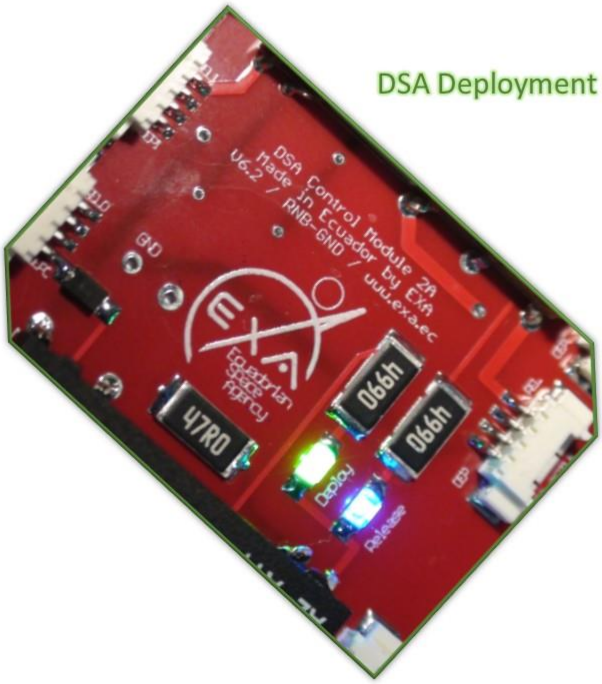
Z2 Software Defined Radio OBC

The ICEPS's Sidekiq Z2 is an SDR Engine:

- SDR range: 70Mhz to 6GHz
- EIRP: +28.5 dBm w/ext. LNB
- Sensitivity: -110 dBm
- Antenna ports: 2 RX
  1 Transceiver (TX/RX)
- Complete libraries in C++ and Linux programming environment.
- Automated RF Switch integrated into the S/C bus



AS-COMSAT, Jan-Feb 2021



DSA Deployment Control

The Deployable Solar Array Deployment control allows the user to rely on ICEPS for deployment and release control.

- Just one simple command from the user.
- DSA/DC automates the release and deploy sequence.
- If using EXA DSA arrays, release/deploy is guaranteed.
- Natively manages artificial muscles and Nitinol-based R/D devices
- Can also manage thermal knife user devices using up to 15W.

AS-COMSAT, Jan-Feb 2021

Sensors and Actuators Control

ICEPS has 24 internal sensors:

- Laser(1), OBC(1), External (2) and battery(2) temperature
- Solar panel voltage (4)
- Power rails (3) voltage
- Solar array deploy/release (4)
- UMMPT bus (1)
- 6 Axis gyroscope (6)

And 4 actuators:

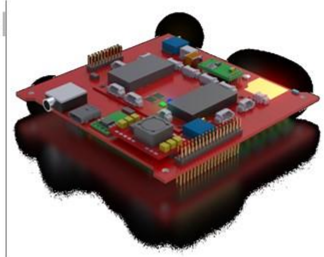
- DSA R/D sequence, Power amplifier, power rail control (2)

AS-COMSAT, Jan-Feb 2021

MODULARITY

ICEPS can be configured from an EPS configuration only to a full System Core configuration at user request:

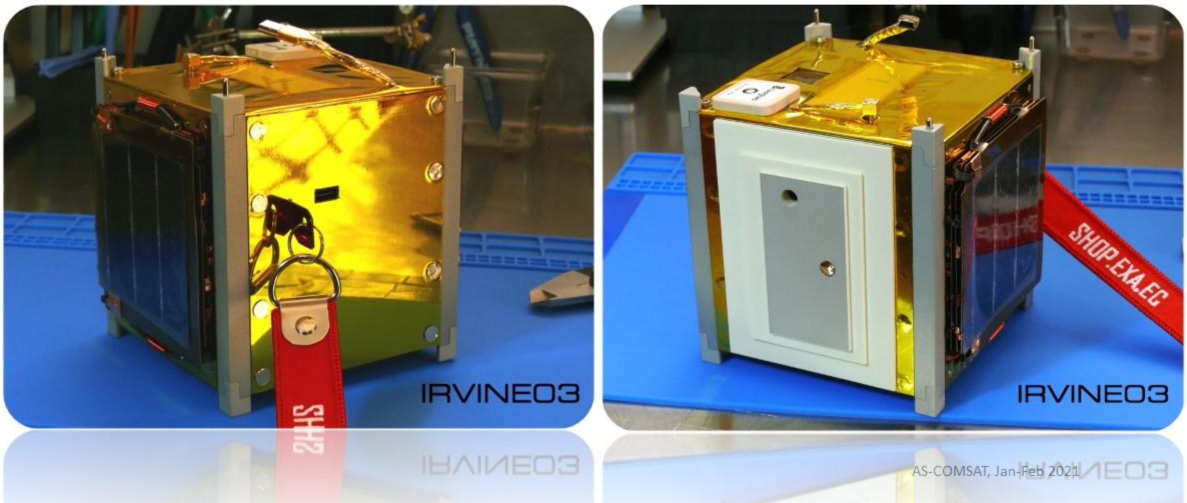
- Power rails, 1 to 6, user configurable output
- Solar channels, 1 to 4
- Solar charging manager, single or dual
- Release/Deploy circuitry, 0 to 1
- Laser comm module, 0 to 1
- Onboard OBC, 0 to 1
- Battery arrays, 1 to 4
- USB 2.0 bus or just GPIO



AS-COMSAT, Jan-Feb 2021

IN-ORBIT DEMONSTRATION

ICEPS ONBOARD IRVINE03 TO BE LAUNCHED ON NASA ELaNA SLOT IN Q2/2020



LUNAR DEMONSTRATION

ICEPS ONBOARD QBWALKER TO LAND ON THE MOON ON BOARD ASTROBOTIC'S PEREGRINE IN Q3/2021

AS-COMSAT, Jan-Feb 2021

CONCLUSIONS

Pros:

- Can serve missions from 1U to 24U.
- Takes the cubesat community beyond the limitations of the I2C, SPI, CAN-bus etc.
- High computing power, throughput, high capacity storage and high speed laser communications are included. SDR naturally embedded into the system.
- Modularity allows great configuration and cost flexibility.
- USB bus enables cubesats to interface with the millions of devices on the ground that have long since standardized to it.

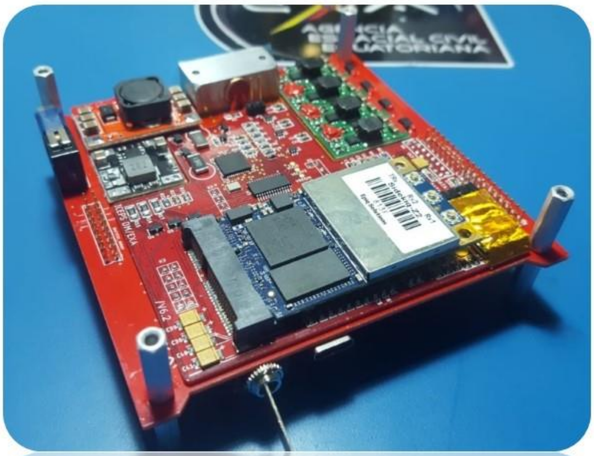
Cons:

- Needs heavier radiation shielding, but EXA's NEMEA solves this need
- The payload development needs to strictly follow USB 2.0 developing guidelines

AS-COMSAT, Jan-Feb 2021

Thank you for your attention

info@exa.ec



AS-COMSAT, Jan-Feb 2021

On-Board-Computer inclusive other components

New

EXA ICEPS Spacecraft System Core

Vendor: EXA
€16500 - €53000

1/23/2022 AS-COMSAT, Jan-Feb 2021 77

On-Board-Computer inclusive other components

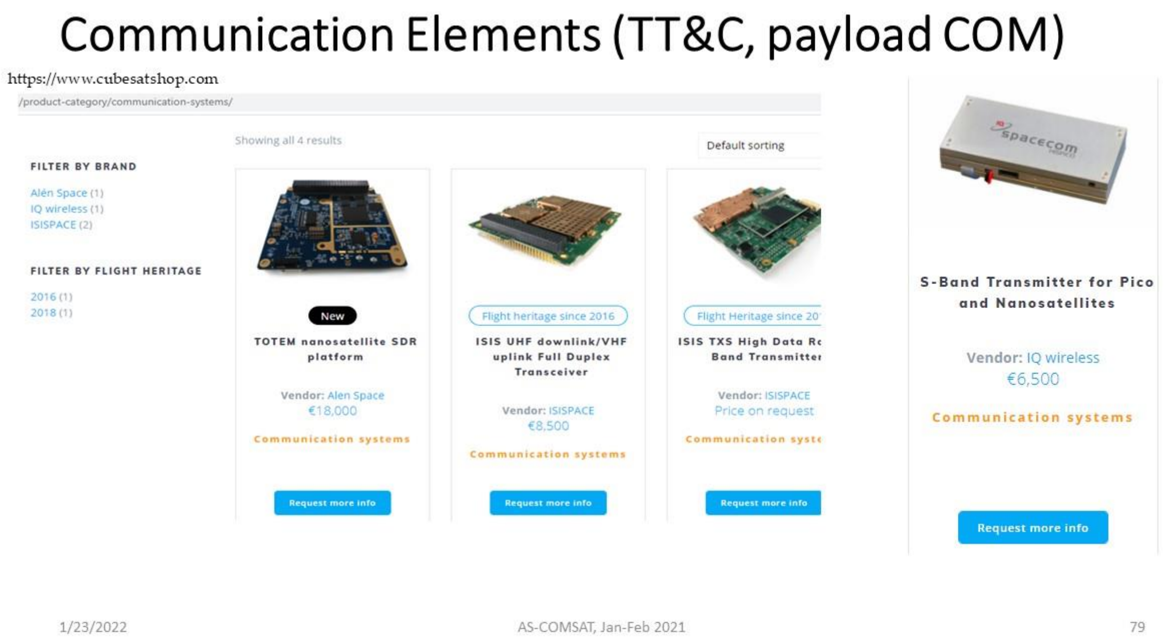
New

EXA ICEPS Spacecraft System Core

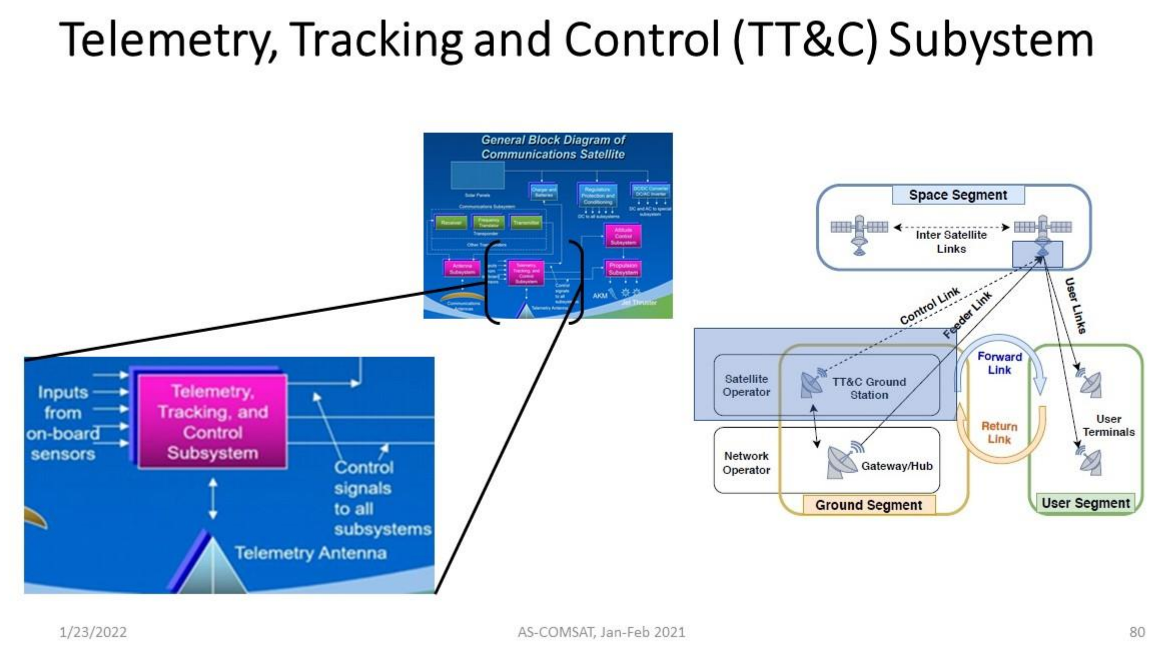
Vendor: EXA
€16500 - €53000

1/23/2022 AS-COMSAT, Jan-Feb 2021 78

8.13.5 COM Elements: TT&C and payload COM



8.13.6 SAT - TT&C System



8.13.6.1 TT&C GCS

TT&C Subystem – Ground Control System (GCS)

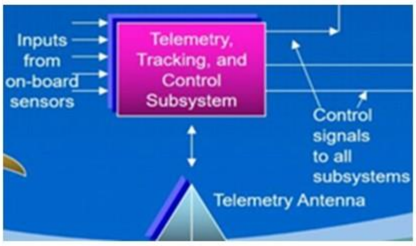
1/23/2022 AS-COMSAT, Jan-Feb 2021 81

TT&C Subystem – Ground Control System (GCS)

1/23/2022 AS-COMSAT, Jan-Feb 2021 82

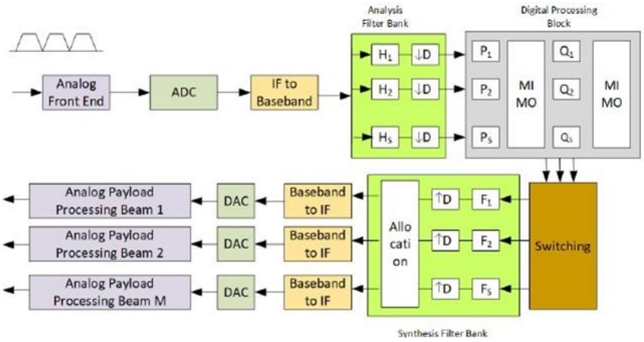
8.13.6.2 TT&C On-board part

TT&C Subystem – On-Board Part



1/23/2022 AS-COMSAT, Jan-Feb 2021 83

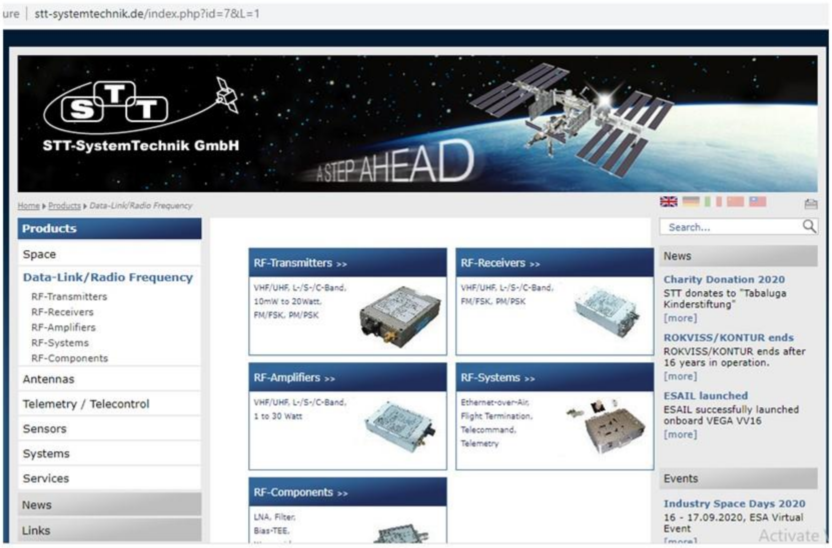
TT&C Subystem – On-Board Part



On-board processing architecture (from: A. I. Prez-Neira et. al., "Signal processing for high throughput satellites: Challenges in new interference-limited scenarios, IEEE Signal Processing Magazine, vol. 36, no. 4, pp. 112–131, 2019.)

1/23/2022 AS-COMSAT, Jan-Feb 2021 84

TT&C Subystem – On-Board Part



stt-systemtechnik.de

1/23/2022 AS-COMSAT, Jan-Feb 2021 85

TT&C Subystem – On-Board Part

TOTEM nanosatellite SDR platform

- SDR + UHF front end platform
  - 5 W @ 30 dBm in 437 MHz
- SDR tunnable from 70 MHz to 6 GHz
- UHF front end as a piggyback board
  - Unregulated voltage supply from EPS and 3V3
  - Multiple GPIOs and DACs available
- Embedded Linux
- Multiple interfaces: CAN, UART, Ethernet
- PC/104 standard
- Physical properties
  - Mass: 131 g (shieldings included)
  - Dimensions : 89.3 mm x 93.3 x 13.9 mm
  - Power consumption
  - ~ 5 W @ 30 dBm output power
  - < 2 W in RX mode
  - 1.36 W with front end OFF

1/23/2022 AS-COMSAT, Jan-Feb 2021 86

TT&C Subystem – On-Board Part

ISIS VHF uplink/UHF downlink transceiver

The ISIS VHF uplink/UHF downlink transceiver is a full duplex communication system for CubeSat **TT&C** applications. The radio can operate in commercial and amateur bands of the VHF/UHF frequency spectrum. It is low power, low mass, and highly configurable, offering the flexibility of changing data rates and frequencies in flight. This radio is tailored for CubeSat missions and cross-compatible with other subsystems such as onboard computers and antenna systems. Flight proven since 2016.
Availability: 8 – 12 weeks
Vendor: ISISPACE

1/23/2022 AS-COMSAT, Jan-Feb 2021 87

TT&C Subystem – On-Board Part

ISIS VHF uplink/UHF downlink transceiver

FEATURES

- Full duplex communication
- Data rate re-configurable in-flight
- FM transponder mode available
- Safety watchdog
- Low power consumption
- Single PCB radio
- Single board Telemetry, Telecommand and Beacon capabilities

PRODUCT PROPERTIES

Dimensions:	90 x 96 x 15 mm
Mass:	75g
Supply voltage range:	6.5 – 20 V DC
Power consumption:	0.48W (receiver only) 4 W (transmitter on)
Operating temperature:	-20 to +60 deg C
RF interfaces:	MMCX (50 ohm)
Data interfaces:	I²C

1/23/2022 AS-COMSAT, Jan-Feb 2021 88

TT&C Subystem – On-Board Part

ISIS VHF uplink/UHF downlink transceiver

PERFORMANCE

Transmitter	
Frequency range:	435 – 438 MHz (amateur-satellite UHF allocation). Other ranges are available on request
Transmit power: Modulation options:	27 dBm Binary Phase Shift Keying (BPSK) with G3RUH scrambling Gaussian Minimum Shift Keying (GMSK) with G3RUH scrambling
Data rate selectable:	1200, 2400, 4800 and 9600 bps
Data link layer protocol:	AX.25 or HDLC

Receiver	
Frequency range:	145.8 MHz – 146 MHz
Modulation:	Frequency Shift Keying (FSK) with G3RUH scrambling
Data rate:	1200, 9600 bps
Sensitivity:	-104 dBm Sensitivity for BER 1E-5
Data link layer protocol:	AX.25

1/23/2022 AS-COMSAT, Jan-Feb 2021 89

TT&C Subystem – On-Board Part

ISIS VHF uplink/UHF downlink transceiver

QUALIFICATION AND ACCEPTANCE TESTING

Test	QT	AT
Functional	✓	✓
Vibration	✓	-
Mechanical Shock	✓	-
Thermal Cycling	✓	✓
Thermal Vacuum	✓	-

*QT is performed on the design/qualification model
*AT is performed on the unit to be shipped

Configuration

- •Receiver/Transmitter operating frequency
- •Downlink data rate
- •Custom beacon message (AX.25)
- •CSKB connector type and location
- •RF connector position and orientation
- •I^{2}C watchdog implementation

1/23/2022 AS-COMSAT, Jan-Feb 2021 90

TT&C Subystem – On-Board Part

ISIS.TXS.DS.001, version 1.0 TXS S-band Transmitter Datasheet

ISIS_TXS_RevC.stp

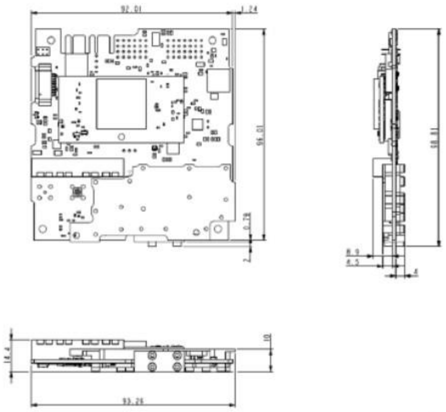
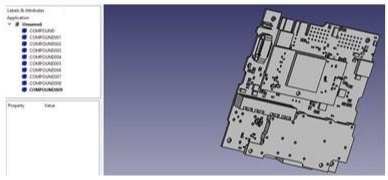


Figure 8 Mechanical outline[1]



Physical layout

Figure 6 Top view

Figure 7 Bottom view

TT&C Subystem – On-Board Part

TXS S-band Transmitter Datasheet

Absolute Maximum Ratings

Stresses at or above the absolute maximum ratings in Table 3 may cause permanent damage to the product. Operation at or beyond the maximum operating ratings may affect product reliability.

Table 3 Absolute Maximum Ratings

Parameter	Symbol	Min	Max	Unit
Supply voltage	V_{CC}	6	26	V
Operating temperature range	T_{amb}	-40	70	°C
Storage temperature range	$T_{storage}$	-40	85	°C
Voltage on I²C pins	V_{I2C}	-0.5	7	V
I²C pull up resistor value	R_{pu}	1.2		kOhm
LVDS input pin voltage	V_{IN_LVDS}	-0.3	3.6	V
LVDS output pin voltage	V_{OUT_LVDS}	-0.3	3.6	V
GPIO input voltage, any GPIO pin	V_{IN_GPIO}	-0.3	3.6	V

TT&C Subystem – On-Board Part

TXS S-band Transmitter Datasheet

Block diagram

TXS is based on a MicroSemi SmartFusion2 SoC. A separate supervisor MCU takes care of power switching, telemetry gathering and watchdog functionality. An LVDS interface is provided for high speed payload data, although (low speed) data to be transmitted can also be routed via the I_2C bus.

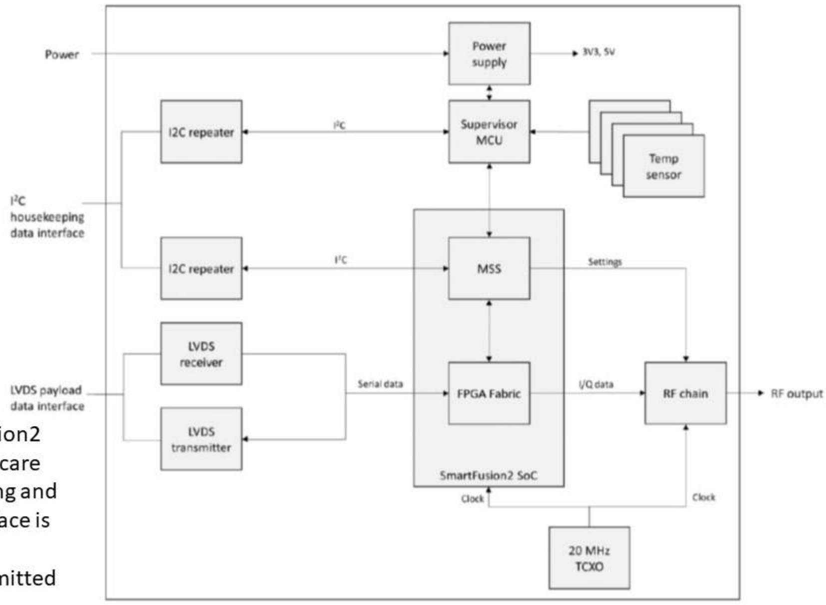


Figure 1 TXS high level block diagram

1/23/2022

TT&C Subystem – On-Board Part

TXS S-band Transmitter Datasheet

Typical link budget

Table 4 provides a typical link budget achieved with TXS to a small groundstation (1.9 m diameter), for a link with these parameters, 2.1 Mbit/s OQPSK can be supported from 5 degrees elevation. With higher groundstation G/T, larger usable datarates can be supported.

Table 4 Typical TXS link budget

Parameter	Value	Unit	Rationale
Frequency	2245.0	MHz	2200-2290 MHz SOS / EESS / SRS space-to-Earth allocation
Satellite transmitter power	3.0	dBW	2W / 33 dBm
Satellite TX losses	1.0	dB	Assumption
Satellite antenna gain	0.0	dBi	Typical patch antenna gain for 5 deg elevation and Nadir pointing satellite
Satellite EIRP	2.0	dBW	
Satellite pointing loss	0.5	dB	Assumption
Orbital altitude	600000.0	m	Typical LEO orbit
Elevation angle	5	deg	Minimum elevation for communication
Range	2329031.4	m	
Path loss	166.8	dB	
Atmospheric losses	0.5	dB	Source: RD3, 99% of the time, Madrid DSN
Ionospheric losses	0.1	dB	Approximate mean values for low earth station elevation angle
Polarization losses	0.0	dB	No polarization mismatch assumed
Earth station pointing loss	1.0	dB	Assumption
Earth station figure of merit	9.0	dB/K	Typical S-band station figure of merit (1.9 m diameter antenna)
Channel symbol rate	2500000.0	sym/s	2.5 Msym/sec
Code rate	0.430502	-	CCSDS RS (255, 223) + conv R = 1/2
Information bitrate	2152510	bit/s	5 Msym/sec OQPSK, RS (255, 223) + conv R = 1/2, interleaving depth = 1
Information bitrate	63.3	dBHz	in dBHz
Implementation loss	2.0	dB	Pessimistic assumption for a typical demodulator
Eb/N0	5.4	dB	
Required Eb/N0	2.4	dB	OQPSK, RS(255, 2223) + C(7, 1/2) for a BER 1E-5
Link margin	3.0	dB	

Note: in the above table, losses are denoted by a positive number.

1/23/2022 AS-COMSAT, Jan-

TT&C Subystem – On-Board Part

Antenna system

VHF/UHF Antenna

The ISIS deployable antenna system for 1U/3U CubeSats contains four tape spring antennas of up to 55 cm length. The deployment system relies on a thermal knife composed of one wire and two redundant heating elements per tape. RF phasing / BalUn circuitry ties the antennas together in a turnstile configuration.
Depending on the configuration, one or two radios in the CubeSat can connect to the antenna system by means of miniature RF connectors. The top face of the antenna system can accommodate a two solar cell solar panel and it can be customized for accommodating sensors or other systems to protrude to the exterior, e.g. camera apertures. The antenna is compatible with any UHF and/or VHF radio system. It can be mounted on all ISIS CubeSat structures and Pumpkin rev C and rev D CubeSat structures. For custom made structures, which adhere to the CubeSat standard mechanical envelope, mounting should also be possible.
Availability: 8-12 weeks
Vendor: ISISPACE

1/23/2022 AS-COMSAT, Jan-Feb 2021 95

TT&C Subystem – On-Board Part

Antenna system

IEEE S band	
Frequency range	2 – 4 GHz
Wavelength range	15 – 7.5 cm
Related bands	•E / F bands (NATO) •UHF / SHF (ITU)

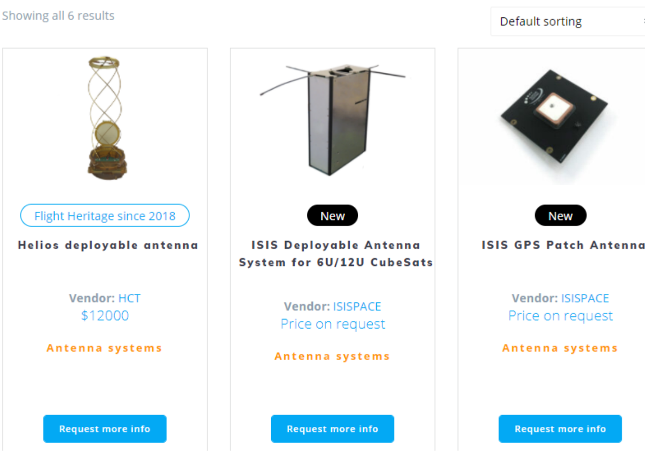
Frequency Allocations

Band	Uplink (GHz)	Downlink (GHz)	Bandwidth (MHz)
C	5.9 – 6.4	3.7 – 4.2	500
X	7.9 – 8.4	7.25 – 7.75	500
Ku	14 – 14.5	11.7 – 12.2	500
Ka	27 – 30	17 – 20	-----
	30 – 31	20 – 21	-----

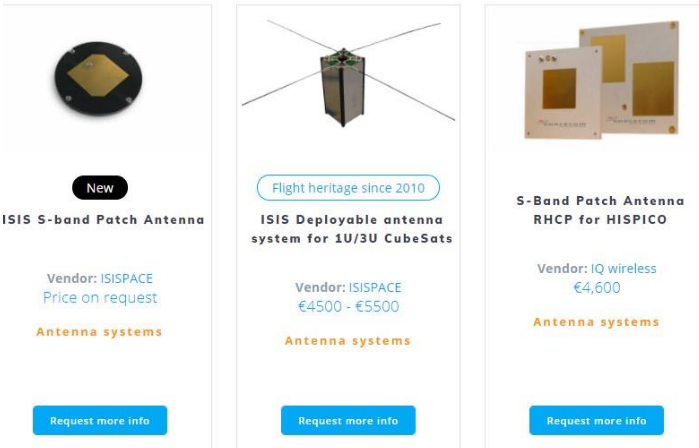
1/23/2022 AS-COMSAT, Jan-Feb 2021 96

Antenna system

Antenna System



1/23/2022 AS-COMSAT, Jan-Feb 2021 139



1/23/2022 AS-COMSAT, Jan-Feb 2021 140

TT&C Subystem – On-Board Part

€4500 - €5500

Antenna system ISIS Antenna

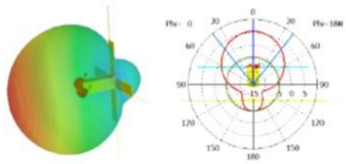
1/23/2022 AS-COMSAT, Jan-Feb 2021 97

TT&C Subystem – On-Board Part

Antenna system ISIS Antenna

PERFORMANCE

- Antenna main beam gain:
  - → UHF: 0 dBi
  - → VHF: 0 dBi
- Max RF Power: 2W
- Bandwidth:
  - → UHF: >50 Mhz (-10db bandwidth)
  - → VHF: >10 Mhz (-10db bandwidth)
- Antenna element deployment duration: <3s at 15°C

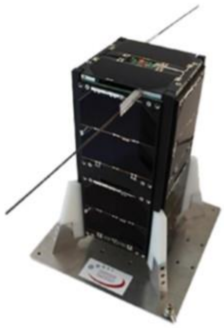


Radiation pattern simulation

PRODUCT PROPERTIES

- Mass: 77-85g (depends on configuration)
- Envelope stowed (l x w x h): 98x98x7mm³
- Antenna length
  - → UHF: 17cm average
  - → VHF: 55cm average
- 30mm diameter through-hole for pass-through of payload or other interfaces (not available for turnstile configuration)
- Power consumption
  - → Nominal: < 40 mW
  - → During deployment: < 2W
- Interfaces:
  - → Electrical: Miniature 9 pin OMNETICS connector
    - → Power: 3.3V or 5V
    - → Data: I2C
  - → RF input/output: MMCX and SSMCX, female 50 ohm
- Qualified operational temperature range:-20°C to +60°C

CONFIGURATIONS



Antenna mounted on CubeSat

1/23/2022 AS-COMSAT, Jan-Feb 2021 98

TT&C Subystem – On-Board Part

Antenna system ISIS Antenna

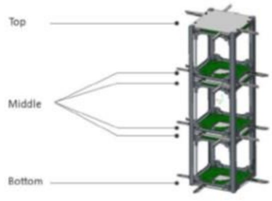
CONFIGURATIONS

- RF Antenna configurations
  - → Single/multiple monopoles (UHF and/or VHF)
  - → Single/dual dipoles (UHF and/or VHF)
  - → Combination of monopole and dipole
  - → UHF or VHF turnstile
- Supply voltage 3.3V or 5V
- RF Harness length and connector type and orientation (MMCX, MCX, SMA)
- Top lid accommodation (solar panel, through hole, mounting points, sensors etc.)
- Customization and simulation on request

DELIVERABLES

- Hardware: antenna, RF harness, refurbishment kit for flight preparation
- Documentation: user manual, test and build reports
- Services: fine tuning, functional and thermal testing

Antenna mounted on CubeSat

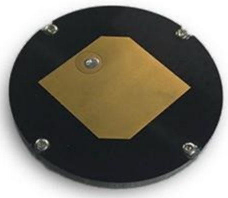


Antenna position configuration

TT&C Subystem – On-Board Part

Antenna system

ISIS S-band Patch Antenna



Description

The ISIS S-band patch antenna is a part of a new generation of antennas designed for S-band communications on nanosatellites and CubeSats. It is an off the shelf compact antenna designed to complement the ISIS S-band transmitter for telemetry and payload data transmissions. The antenna is mounted onto a Rogers PCB, ideal for high-frequency RF circuitry ensuring fewer losses than traditional FR-4 PCBS.

The ISIS S-band patch antenna is a compact, low mass solution suitable for the commercial S-band frequency range of 2200-2290 MHz. This passive antenna is suitable for any CubeSat platform.

ISIS S-band transmitter for telemetry and payload data transmissions

TT&C Subystem – On-Board Part

Antenna system

- Applications:
– CubeSat TT&C
– CubeSat RF Payloads
- Lefthand or Righthand Circular antenna polarization
- Dual Modular Redundant Release Mechanism
- Designed for combination with multiple Receiver/Transceivers
- Compatible with ISIS products and recent Pumpkin, ClydeSpace and GomSpace products
- Compliant to CubeSat standard

RF Impedance (deployed): 50 Ohms

- Max RF Power: 1 Watt
- Frequency Range: 400-3000 MHz
- Electrical Power: 8 VDC at 7 Amps for 1 minute to deploy
- Envelope Stowed (l x w x h): 100mm x 100mm x 35mm
- Antenna Axial Height (deployed): 330mm
- Supply Voltage: 8 VDC at 7 Amps for 1 minute to deploy
- Operational Temperature Range: -40°C to 85°C
- Antenna main beam gain: 3dBi+
- Deployment Duration: 60-90s
- Antenna Return Loss at resonance frequency: >10 dB
- Power Consumption:
– Nominal: 0

1/23/2022 AS-COMSAT, Jan-Feb 2021 101

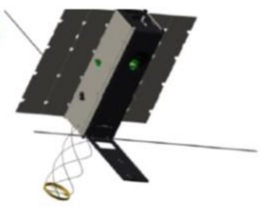
TT&C Subystem – On-Board Part

Antenna system

HCT
Helical Communication Technologies

HCT 400 or 437 MHz Heritage QHA

400 or 437MHz Heritage UHF QHA



634 Barnes Boulevard Suite #206, Rockledge, FL 32955
O: 321-208-8978
E: HelicalCommunicationTech@gmail.com

https://www.helicomtech.com/

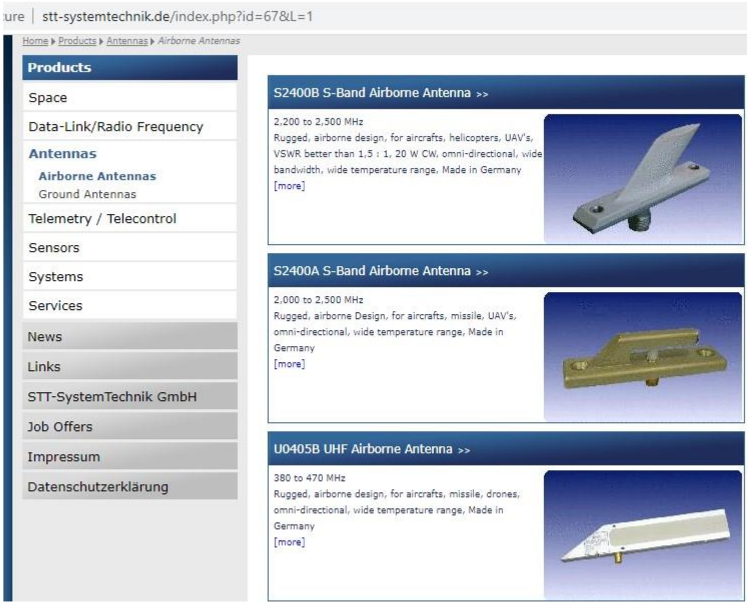
About Us

Helical Communication Technologies was formed to serve the increasing need for specialized antennas for use with ground-based and space-based communication with satellites placed in low earth orbit and deep space.

1/23/2022 AS-COMSAT, Jan-Feb 2021 102

TT&C Subsystem – On-Board Part

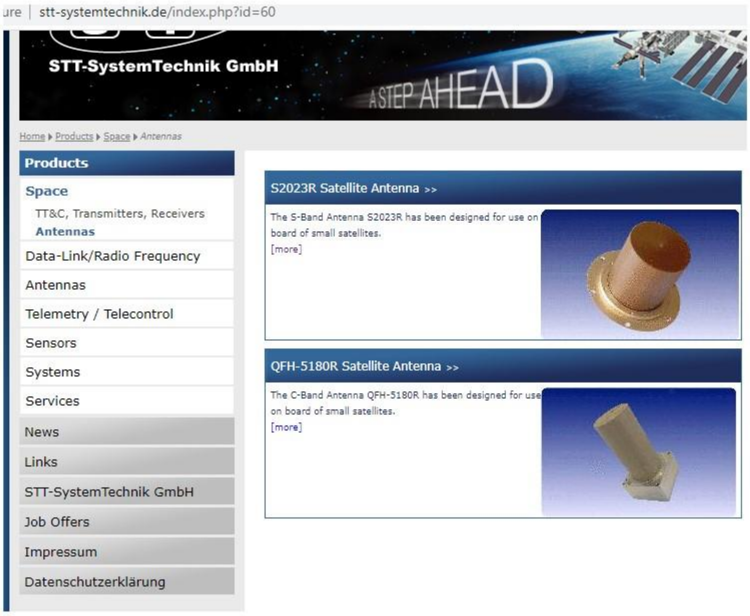
Antenna system



1/23/2022 AS-COMSAT, Jan-Feb 2021 103

TT&C Subsystem – On-Board Part

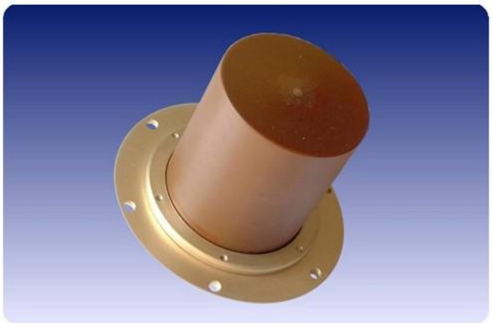
Antenna system



1/23/2022 AS-COMSAT, Jan-Feb 2021 104

TT&C Subystem – On-Board Part

Antenna system



S2023R S-Band Satellite Antenna

The S-Band Antenna S2023R has been designed for use on board of small satellites. The pattern shape has been optimized particularly for low earth orbit (LEO) missions with NADIR orientation. The antenna is broadband and operates at both the standard S-Band frequencies for satellite missions, i.e. downlink at 2200 to 2290 MHz and uplink at 2025 to 2110 MHz.
The antenna design is compact, solid and very rugged.
To achieve spherical coverage the S2023R could be used in pairs with opposite placement at the satellite body.

TT&C Subystem – On-Board Part

Antenna system

S2023R S-Band Satellite Antenna

frequencies for satellite missions, i.e. downlink at 2200 to 2290 MHz and uplink at 2025 to 2110 MHz.

The antenna design is compact, solid and very rugged.

To achieve spherical coverage the S2023R could be used in pairs with opposite placement at the satellite body.

Electrical Specifications

Frequency Range	2025 MHz...2110 MHz uplink 2200 MHz...2290 MHz downlink
Gain	3 dBic boresight, typ. > 0 dBic for -45° < Θ < +45°, typ. > -6 dBic for -90° < Θ < +90°, typ.
Coverage	Hemispherical
HP Bandwidth	140° typical
Polarization	Right circular (Left circular optional)
Power	40 dBm CW, max.
Impedance	50 ohms
VSWR	better than 1,5 : 1
Connector	SMA female

Environmental Specifications

Operating Temperature	-60°C...+120°C (extended range upon request)
Vibration	20...2,000 Hz; 25g rms random, 3-axis
Shock	100g (100 Hz), 3,500g (>1000 Hz)

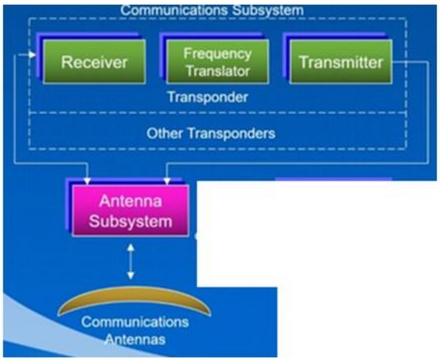
Mechanical Specifications

Dimensions	Length approx. 70 mm w/o connector Reflector d = 100 mm Radome d = 60 mm
Mounting	6 holes each 4.2 mm
Weight	ca.140g
Radome	PEEK (beige) VESPEL (gold brown) optional
Reflector	AW6082 with finish Alodine1200 or SURtec650

8.13.7 SAT - Payload Communications Subsystem

Payload Communications Subsystem

- Receiver
- Frequency Translator
- Transmitter



1/23/2022 AS-COMSAT, Jan-Feb 2021 107

COM Payload – On-Board Part (Downlink)

ISIS.TXS.DS.001, version 1.0 TXS S-band Transmitter Datasheet

ISIS

Overview

The ISIS High Data-rate S-band Transmitter is a CubeSat compatible transmitter designed to meet the needs of high data-rate downlinks of up to 4.3 Mbps (usable information bit-rate at CCSDS TM Transfer Frame level). **The transmitter can be used for both TT&C or Payload Data downlinks.** The S-band transmitter is flexible, implementing CCSDS as data link layer protocol and allowing in-flight configuration of data-rate, modulation scheme, frequency, and RF output power.

1/23/2022 AS-COMSAT, Jan-Feb 2021 108

SDR (Software Defined Radio): HackRF Hardware Platform and GnuRadio Software Library - Introduction

Tutorials

First steps and analog modulation

This tutorial/guide is a good start into HackRF and Gnuradio, including installation instructions, theoretical basics, and simple examples for analog (de)modulation.

After installing and trying out the software HDSDR and SDRSharp, we made 3 tutorials out of this guide: Receiving, Transmitting, and Live Broadcasting FM Radio Signals.

1/23/2022 AS-COMSAT, Jan-Feb 2021 109

Figure 4: equipment: 2x HackRF One, smartphone as FM radio (receiver), laptop(not in picture), traditional radio(not in picture)

1/23/2022 AS-COMSAT, Jan-Feb 2021 110

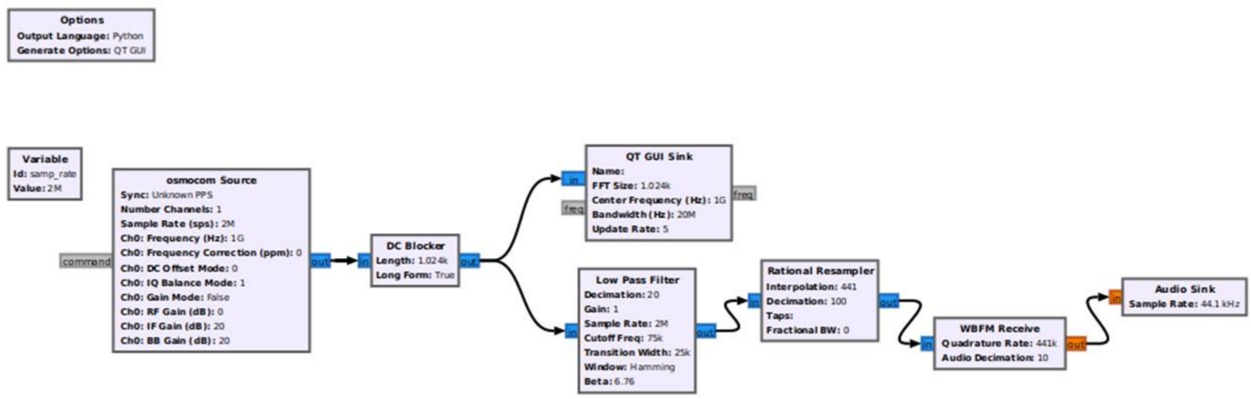


Figure 1: Gnuradio flowgraph – Analog Receiving

1/23/2022 AS-COMSAT, Jan-Feb 2021 111

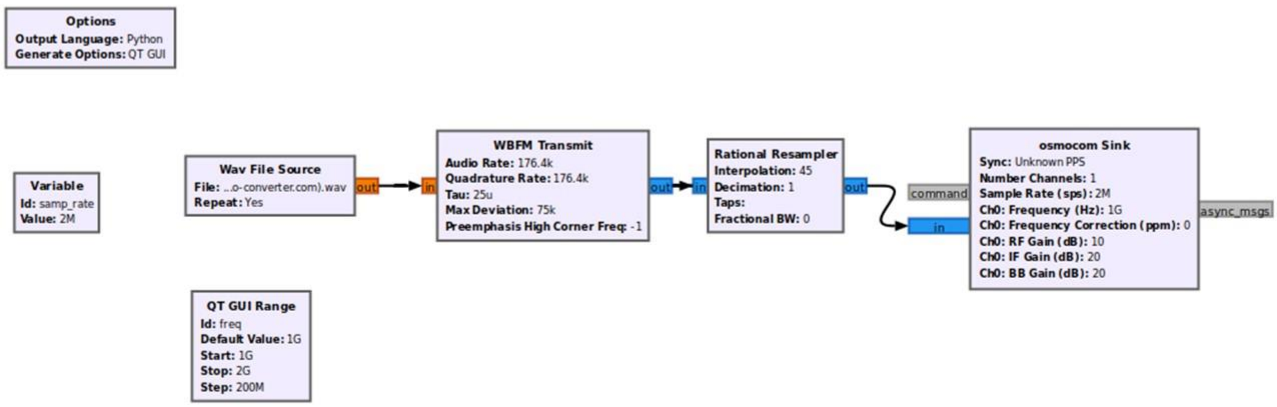


Figure 2: Gnuradio flowgraph – Analog Transmitting

1/23/2022 AS-COMSAT, Jan-Feb 2021 112

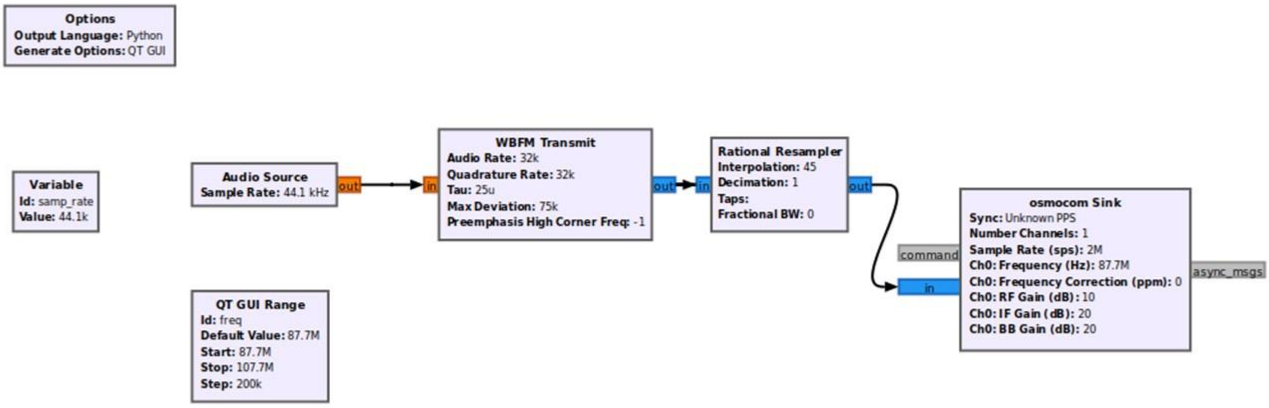


Figure 3: Gnuradio flowgraph - Live Broadcast

1/23/2022 AS-COMSAT, Jan-Feb 2021 113

Technical info

GNU Radio Companion: 3.8.1.0 (Python 3.8.5)

OS: Ubuntu 20.04.1 LTS as Virtual machine on Windows 10 Host (using VirtualBox)

HackRF One Firmware version: 2015.07.2 (API:1.00)

2. HackRF One Firmware Version: 2018.01.1 (API:1.02)

1/23/2022 AS-COMSAT, Jan-Feb 2021 114

Digital modulation

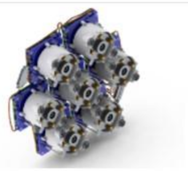
Here is our current stand. In addition to other well-known digital (de)modulations, there is the PSK (de)modulation, we are currently trying the following tutorial:

https://wiki.gnuradio.org/index.php/Guided_Tutorial_PSK_Demodulation

1/23/2022 AS-COMSAT, Jan-Feb 2021 115

8.13.8 Electrical Propulsion Unit

Propulsion Unit



Cluster of IFM Nano Thruster for Smallsats

Vendor: ENPULSION

€50000 - €210000

Propulsion & pressurisation

Flight Heritage since 2018

IFM Nano Thruster for CubeSats

Vendor: ENPULSION

€38,400

Propulsion & pressurisation

Nanosatellite Micropropulsion System

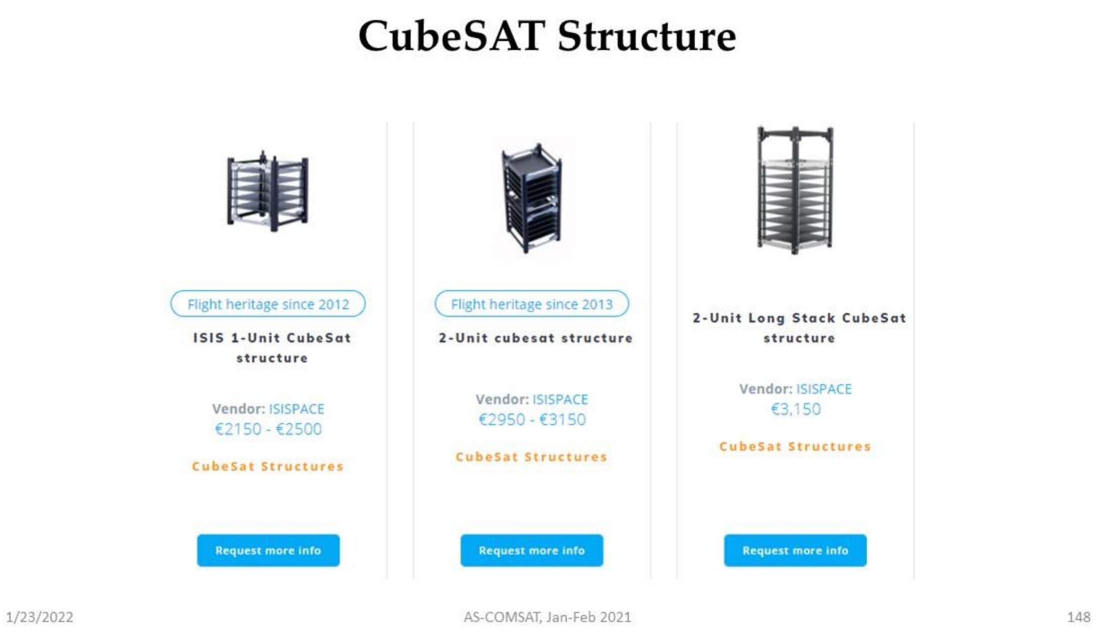
Vendor: MicroSpace

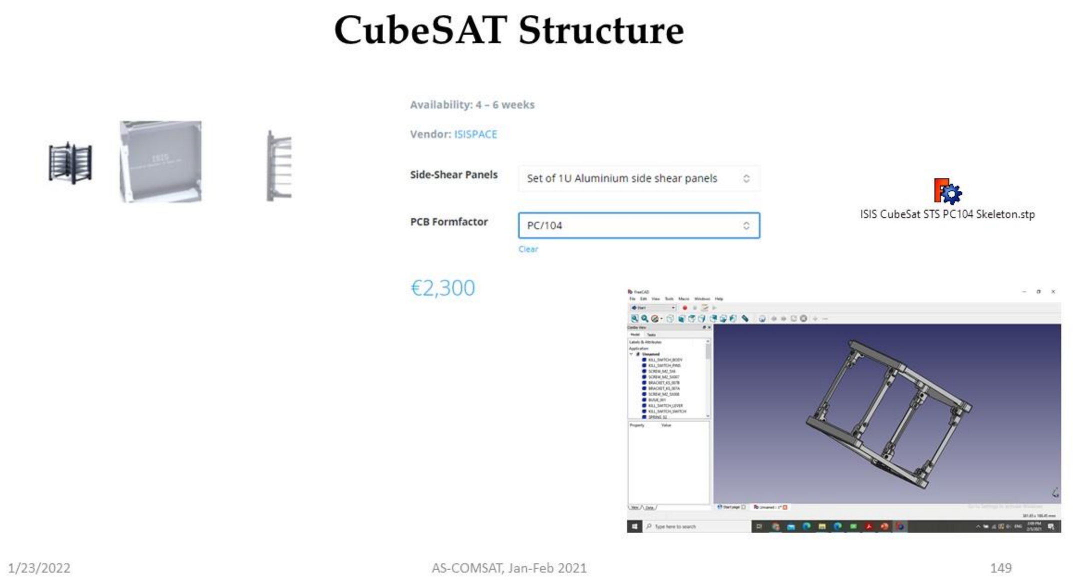
€81000 - €129000

Propulsion & pressurisation

1/23/2022 AS-COMSAT, Jan-Feb 2021 147

8.13.9 CubeSAT Structure





TT&C Subystem – Ground Support Equipment

1/23/2022 AS-COMSAT, Jan-Feb 2021 150

Launch adapter

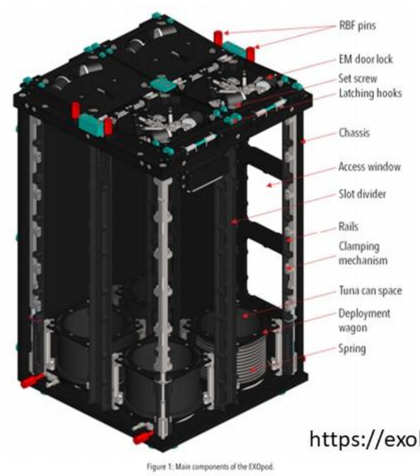
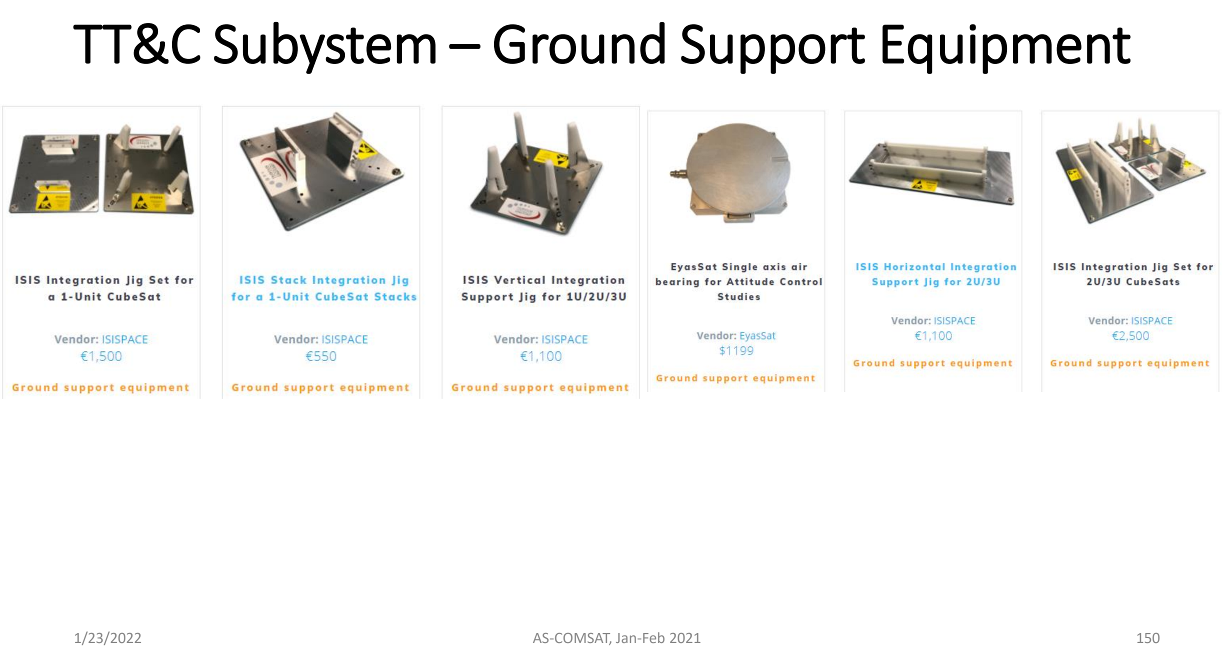


Figure 1: Main components of the EXOpod.

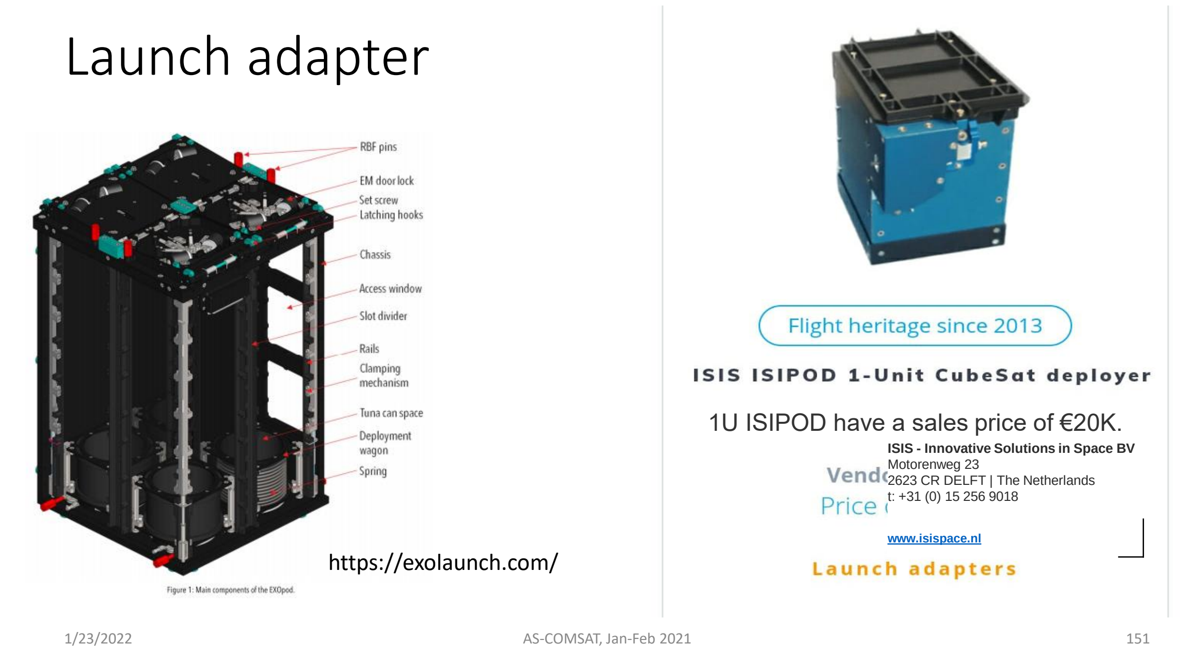
https://exolaunch.com/

1/23/2022 AS-COMSAT, Jan-Feb 2021 151

8.13.10 Ground Support Equipment

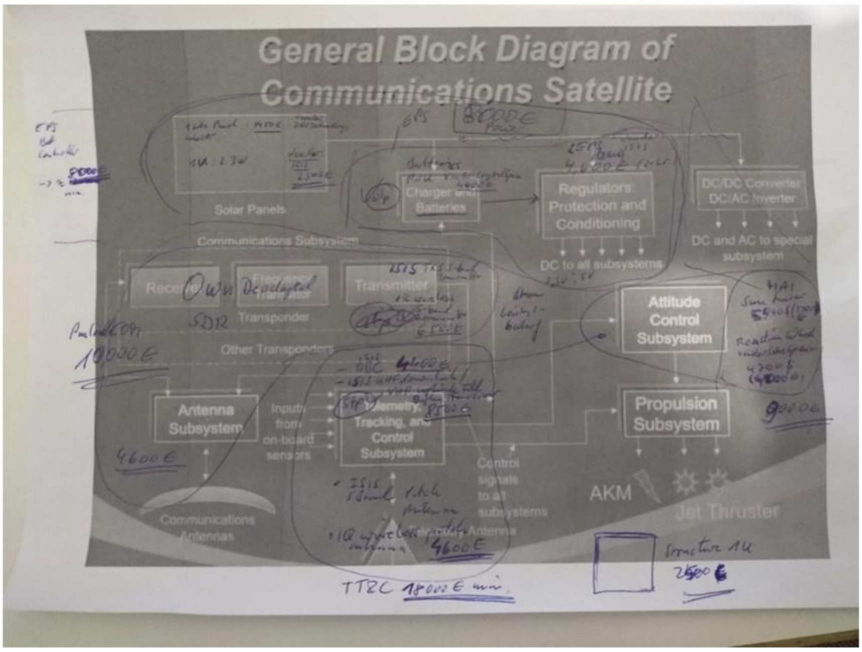


8.13.11 Launch adapter (SAT deployment)



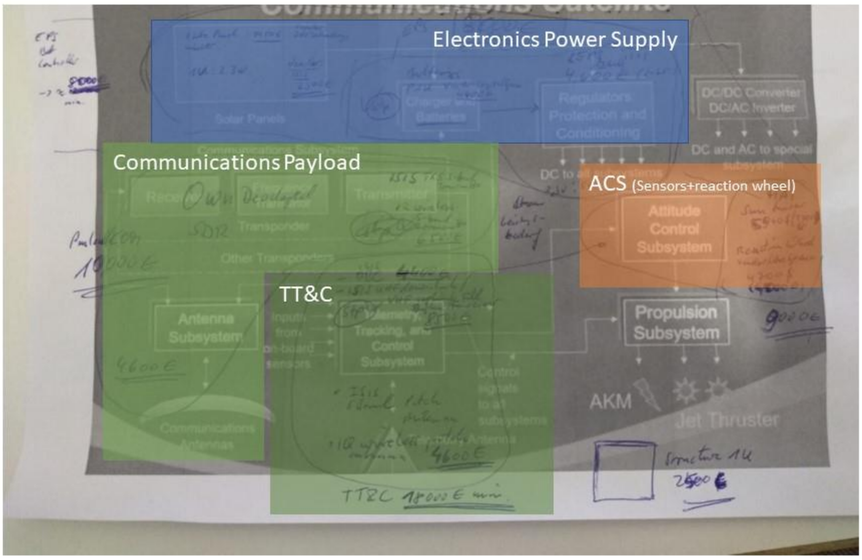
8.14 Minimal System with suppliers components

Minimal System for Demonstration



1/23/2022 AS-COMSAT, Jan-Feb 2021 152

Minimal System for Demonstration



1/23/2022 AS-COMSAT, Jan-Feb 2021 153

Minimal System for Demonstration

The parts for a minimal communication 1U cubesat (without camera, only communication payload) will cost inscha Allah about 48,000 EUR (See above sheet). The launch also about 40,000 EUR.

That means we need about 90,000 EUR for a demonstration satellite system.

The communication payload costs about 10,000 EUR from cubesatshop.With the master thesis proposal from Salih hoca this will be reduced to the cost of an S-band patch antenna (about 4600 EUR)

When we have this demonstration system working, then we will find inscha Allah investors for larger systems.

Next step:

Verifying power requirements for electronic parts. (Is the 1U Power Solar Panels + Batteries System enough?)

1/23/2022 AS-COMSAT, Jan-Feb 2021 154

Minimal System for Demonstration
Next working packages

Scientific:

- ...

Technical:

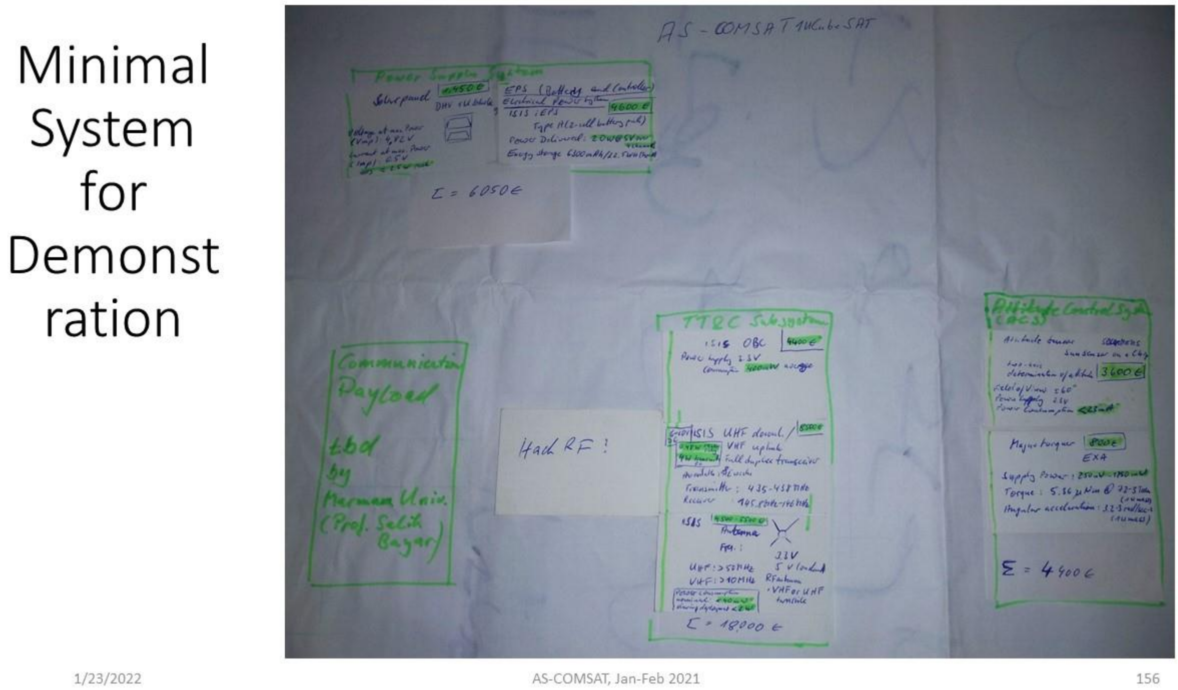
- Verifying power requirements for electronic parts. (Is the 1U Power Solar Panels + Batteries System enough?)
- ...

Administrative:

- Business Plans (ROI for municipalities, …)

TODO: Specifying working packages

1/23/2022 AS-COMSAT, Jan-Feb 2021 155



8.15 References

[1] Jana Othman, AECENAR Internship report, 3.Sep 2021

[2] http://aecenar.com/index.php/downloads/send/16-ics/666-temoleb-mintad-final-report-1999-2020

[3] *Development of a GS Package suited for Spacecraft Operation Control and Optimization methods for Satellite flyby over the Ground Station*, Raj Gaurav Mishra, Master Thesis at University of Wuerzburg, Germany, 2007

[4] A DESCRIPTION OF A STANDARD SMALL SATELLITE GROUNDSTATION FOR USE BY WMO MEMBERS, TECHNICAL DOCUMENT WMO/TD No. 660, 1995

[5] Rozan Mustafa, Draft of Master Thesis (Marmara University Istanbul), Oct 2021

9 Distributed Vessel Clustering for S-AIS Based Maritime Network

MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES
IN PURE AND APPLIED SCIENCES

S-AIS Based Maritime Network: Distributed Vessel Clustering

CANDAR KARABULUT

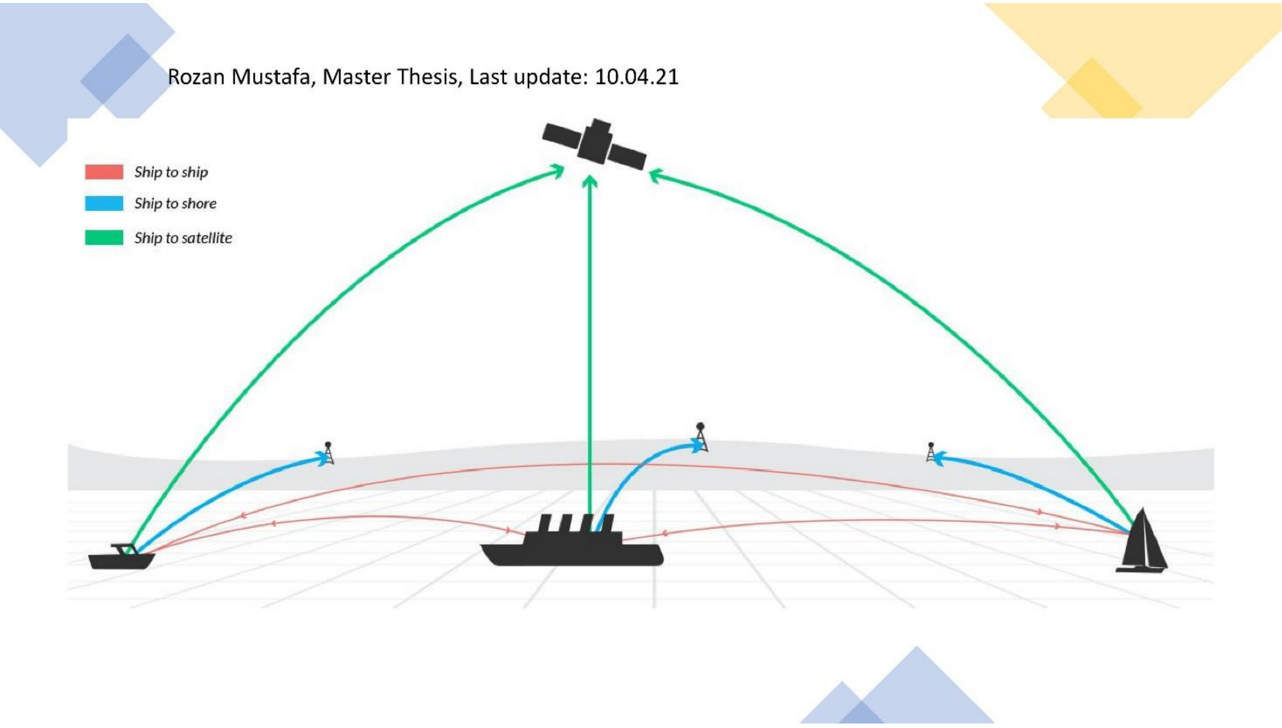
Rozan Mustafa

MASTER THESIS
Department of Computer Engineering

Thesis Supervisor
Assoc. Prof. ÖMER KORÇAK

ISTANBUL, 2022

9.1 AIS Presentation



Automatic Identification System

- Required on any international ship with gross tonnage of 300 or more tons.
- Required on ALL passenger ships regardless of size
- Tracking system for ships
  - Ship-to-ship communication in open-sea
- Some Applications:
  - Vessel Traffic Services
  - Collision Avoidance
  - Maritime Security
  - Search and rescue, Accident investigation
- AIS 1: Works on 161.975 MHz (ship to ship)
- AIS 2: 162.025 MHz (ship to shore)

Automatic Identification System

- Static Information (Every 6 minutes and on request):
  - MMSI Number (Unique)
  - Name
  - Length
  - Ship Type
- Dynamic Information (Depends on speed and course alteration)
  - Positon (GPS)
  - Speed

!AIVDM,1,1,,A,13aEOK?P00PD2wVMdLDRhgvL289?,0*26


```
{
    'type': 1,
    'repeat': 0,
    'mmsi': '244670316',
    'status': <NavigationStatus.Undefined: 15>,
    'turn': -128,
    'speed': 0.0,
    'accuracy': True,
    'lon': 4.379285,
    'lat': 51.89475,
    'course': 70.60000000000001,
    'heading': 511,
    'second': 14,
    'maneuver': <ManeuverIndicator.NotAvailable: 0>,
    'raim': True,
    'radio': 33359
}
[Finished in 0.1s]
```

Challenges

- There are many challenges with satellite AIS since the signals were never meant to be received from space.
- The biggest one is that the satellites will receive transmissions from many, many vessels - up to thousands of vessels - simultaneously. So the big challenge is the signal processing challenge of being able to make sense of all that information.
- Ship can turn off their AIS (Only a ship can tell whether its neighbour turned off its AIS or not, satellite cannot be sure when it is in crowded areas eg. signal interference, overloading.)
- Horizontal Range up to 74Km.

Vessel Clustering

- Cluster ships into some groups.
- Each cluster has its own cluster head.
- Cluster head is responsible for communicating with satellite after processing its cluster members messages.
- Instead of processing thousands of signals, satellite is now processing fewer data.
- Clustering based on speed, type of vessels, position, transmission power…

Applications Of S-AIS

- **Fishing fleet monitoring and control**

AIS is widely used by national authorities to track and monitor fisihing activities depending on location and quality of coast based receivers/base stations with supplementary data from satellite-based networks.

- **Maritime security**

Improves maritime domain awareness – Vessel types – Pirating activities.

- **Search and rescue**

AIS Classes – Class A

- Send and receive data over longer distance.
- Uses SOTDMA.
- The transmission of AIS data is also more frequent.
- SOTDMA requires a transceiver to maintain a constantly updated slot map in its memory such that it has prior knowledge of slots which are available for it to transmit. SOTDMA transceivers will then pre-announce their transmission, effectively reserving their transmit slot.

AIS Classes – Class B

- Less expensive, short distances, lower reporting rate.
- Uses CSTDMA.
- Time slot allocation is not guaranteed (if not found it waits 30s)

AIS Classes

- AIS information from a class A transponder will always be prioritized and, thus, be shown to other ships in the area.
- AIS information from a class B transponder will not be shown until or if there is room (free time slot) on the AIS channel.

- The fundamental challenge for AIS satellite operators is the ability to receive very large numbers of AIS messages simultaneously from a satellite. There is an inherent issue within the AIS standard; the TDMA radio access scheme defined in the AIS standard creates 2250 available time-slots in each minute but this can be easily overwhelmed by the large satellite reception footprints and the increasing numbers of AIS transceivers, resulting in message collisions, which the satellite receiver cannot process.
- Lost AIS Signal vs Dark Activity
- ***The original purpose of AIS was solely collision avoidance***

9.2 ACKNOWLEDGEMENTS

God Almighty says in his Noble Book: “Be grateful to God. Whoever is grateful does so only for his own good”. At the beginning of my thesis, I must first thank God Almighty, who enabled me to reach this high scientific stage, and paved the way for me to be among you today to discuss my master’s thesis. I would like to express my sincere gratitude to my beloved father, my honorable mother, and my dear brothers for their ongoing support. I would also like to extend my thanks and gratitude to Assoc. Prof. Ömer KORÇAK for his enthusiasm for the project, for his support, encouragement, and patience. May God protect him, take care of him, and prolong his life. His supervision had the first hand in getting this scientific thesis out in the way it appeared. His guidance and advice were instrumental in completing my scientific studies. Thanks are due to the honorable members of the discussion committee, Assoc. Prof. Müjdat Soytürk and Assist. Prof. Mehmet Tahir Sandıkkaya. for accepting the discussion of this master's thesis. I also extend my thanks and gratitude to Marmara University and all its member's staff along with the university administration, which had the greatest role in providing facilities and services to complete this study. I would also give my special thanks to Türkiye Scholarships, also known as Türkiye Burslari (YTB), the government-funded higher education scholarship program run by the Republic of Türkiye for international students.

August, 2022 **Candar KARABULUT**

TABLE OF CONTENTS

9.4 ÖZET: S-AIS Tabanlı Denizcilik Ağı: Dağıtılmış Gemi Kümelenmesi

B˙ILG˙ISAYAR MÜHENDİSLİĞ˙I Yüksek Lisans derecesi için CANDAR KARABULUT, Ağustos 2022, Marmara Üniversitesi, ˙Istanbul, Türkiye'de sunulmus¸tur.

AIS (Otomatik Tanımlama Sistemi) 1990'larda gemilerde kullanılmak üzere ilk kez geliştirildiğinden beri, bu periyodik olarak gemilerin yayınladıkları verilerin gemi trafik hizmetlerinde (VTS) çok önemli olduğu ve denizlerde ve okyanuslarda nelerin olduğunun anlas¸ılmasında çok yararlı olduğu görülmüs¸tür. Bu değerli sinyal verileri, diğer birçok uygulamanın yanı sıra izleme amaçları, arama ve kurtarma, navigasyon ve çarpıs¸madan kaçınma için kullanılabilir. AIS sinyalinin yatay menzili yaklas¸ık 74 km'dir, bu da trafik yönetiminin sadece kıyı bölgelerinde ve limanlarda veya gemiler arasında yapılmasını kısıtlamaktadır. Bu sınırlama, denizcilik ağı için daha istikrarlı bir iletis¸ime sahip olmak için alternatif çözümlerin aras¸tırılmasını kaçınılmaz hale getirmis¸ ve bu ihtiyaca kars¸ılık vermek için S-AIS terimi ortaya çıkmıs¸tır. AIS verilerini toplamak, is¸lemek ve analiz etmek için uydular kullanıldığında; Satellite AIS anlamına gelen S-AIS terimi kullanılır. Ancak, uydu tabanlı sistemleri kullanmanın temel sorunu, kalabalık alanlarda uydunun es¸zamanlı olarak is¸lemesi gereken çok sayıda AIS sinyali olmasıdır. Böyle durumlarda, çarpıs¸ma veya diğer sinyaller ile giris¸imler nedeniyle birçok önemli gemilerin AIS sinyalları kaybolabilir. Ayrıca, uydunun ayak izinin dıs¸ında kalan gemilerin takibi, denizcilik ağında birçok gemiyi yetkililere görünmez hale getiren bir diğer önemli konudur. Bu açıkları kapatmak için, bu çalıs¸mada gemiler arası is¸ birliği ile dağıtık kümeleme yaklas¸ımı kullanılmasını öneriyoruz. Gemilerin gruplanması ve kümelenmesi, sinyalleri uyduya iletmeden önce ayak izi alanı dıs¸ında bazı yararlı veri ön is¸leme yöntemlerini uygulamamıza yardımcı olabilecek ve küme üyelerinin verilerine toplu olarak kapsamlı bir genel bakıs¸ sunmamızı sağlayacaktır. Açık denizlerin ortasında gemiler arasında is¸ birliğine dayalı çalıs¸manın, genel denizcilik ağını ve gemi trafiği hizmetlerini iyiles¸tirmeye yardımcı olacağına ve ayrıca yetkililere açık denizlerde neler olup olmadığını daha iyi anlamalarına yardımcı olacağına inanıyoruz.

9.5 ABSTRACT: S-AIS Based Maritime Network: Distributed Vessel Clustering

Rozan Mustafa, for the Master of Science degree in COMPUTER ENG˙INERR˙ING, presented on July 01, 2022, at Marmara University, Istanbul, Türkiye.

Automatic Identification System (AIS) is crucial in vessel traffic services (VTS) and beneficial for getting a better understanding of what is happening in the middle of the seas and oceans. AIS signals are being used for many applications including ship tracking, search and rescue, navigation, and collision avoidance. Since the horizontal range of the AIS signal is limited, it is inevitable to utilize satellite communications on open seas which introduced satellite-based AIS (S-AIS). However, there are significant challenges with satellite-based systems, such as the huge number of AIS signals that the satellite has to process concurrently in crowded areas which cause signal interference making some important vessel's signals be lost or dropped. Moreover, tracking vessels that are out of the satellite's footprint is another essential issue in the maritime network making many vessels to be invisible to the authorities. To bridge these gaps, a new methodology is presented based on distributed clustering approach for the vessels so that they can work cooperatively to transmit their signals to the satellite. Grouping and clustering ships will

allow us to collectively take a comprehensive overview of the data of the cluster members that can help us to implement some useful data pre-processing methods outside the footprint area to examine the signals before transmitting to the satellite. We implement the proposed methodology in a realistically generated vessel traffic in the Mediterranean and the Black Sea and propose some optimization approaches to increase the coverage area. Test results show that the cooperation between vessels in the middle of the open seas will help to improve the overall maritime network and vessel traffic services and assist the authorities as well to get a better understanding of what is happening on the open seas.

9.6 ABBREVIATIONS

AIS : Automatic identification System

VTS : Vessel Traffic Services

S-AIS : Satellite-Based Automatic Identification System

VANET : Vehicular Ad-Hoc Networks

S2S : Ship-to-Ship

SOG : Speed Over Ground

SNR : Signal-to-Noise Ratio

DBSCAN : Density-Based Spatial Clustering of Applications with Noise

IMO : International Maritime Organization **MMSI**

: Maritime Mobile Service Identity **GPS**

: Global Positioning System

LEO : Low Earth Orbit

HD : Hop Distance

TLE : Two Line Elements

DVCA : Distributed Vessel Clustering Algorithm **RAAN**

: Right Ascension of the Ascending Node **CH :**

Cluster Head

9.7 LIST OF FIGURES

9.8 LIST OF TABLES

9.9 List of Algorithms

9.10 INTRODUCTION

Since the AIS (Automatic Identification System) was first introduced in the 1990s to be used in ships and vessels, the data that these transceivers broadcast periodically has proven to be crucial in vessel traffic services (VTS). Vessels use AIS signals to communicate with other ships and with the shore using different frequencies (161.975 MHz for ship-to-ship and 162.025 MHz Duplex for the ship-to-shore) [1]. Initially the main idea behind AIS was only collision avoidance based on S2S (ship-to-ship) communication, but nowadays almost all maritime and vessel traffic services applications are relied on these signals. The horizontal range of AIS signal is about 74 kilometers which makes the traffic management obtainable only by the shore in the coastal areas and ports or in the S2S zone [2]. This limitation makes investigating alternative solutions inevitable to have more stable communication for maritime networks and the term S-AIS is finally introduced.

S-AIS stands for Satellite AIS, where Low Earth Orbit (LEO) satellites are used to detect, analyze, and process AIS data. However, the key issue with using satellite-based systems is the huge number of AIS signals that satellite has to process concurrently in crowded areas or from the satellite's very wide signal footprint. This leads to many lost AIS signals because of the collision or the interferences with other signals and thus, AIS is still not a perfect data source [3]. The other issue is that a vessel is not always in the satellite's footprint especially in the oceans, making tracking operations unreliable. Thus, A vessel is not always in the satellite's footprint especially in the oceans, but even if it is in the coverage area, its signal may not be detected when the satellite is passing over crowded area. Also some of the ships may perform dark activities by intentionally switching off their AIS signal transmitters.

Since initially AIS signals never meant to be transmitted to the space and the main idea behind it was only collision avoidance based on S2S communication, rethinking is necessary to better utilize satellite-based AIS systems considering all the aforementioned limitations.

In this paper, we aim to examine how satellite-based AIS systems can detect and analyze the AIS signals in open seas beyond the reach of terrestrial-only networks using a fully distributed approach to guarantee safety and security for the vessels in addition to ensure better tracking operations providing better time interval for position report. We design and present a novel distributed clustering algorithm for the vessels to prove how the coordination betweenships can be useful to overcome some limitations of AIS and make communication more reliable and stable for maritime monitoring systems.

Relying on data broadcasted by other ships, we intend to group a set of vessels into multiple clusters. Instead of clustering based on similar characters or homogeneous dynamic/static data like most the clustering algorithms do, we cluster the vessels based on their position to closest satellite's footprint. We will see how it is important to neglect the fundamental rule of clustering in Vehicular Ad Hoc Networks (VANETs) or other networks which is; nearest neighbors into same cluster as much as possible. However, we intentionally have to avoid grouping closest pairs into same cluster. Dark Activity is another issue in Maritime Network which we believe cooperative vessels can overcome.

Our aim is not to bring a new product, but to illustrate the impact of **cooperative vessels** in maritime networks and how ships working together can bring new features and improvements to the network. Thus, we intend to suggest the necessary modifications with which our presented clustering approach can work.

9.10.1 Contributions of the Thesis

Clustering vessels using a proper algorithm based on dynamic factors to encourage cluster members to work cooperatively, to exchange data and share and manage the available resources efficiently and fairly, will not only lead to better network management, but will also give the vessels another reason to obtain an AIS transceiver.

Clustering can play an important role to improve the maritime network and the term of Cooperative Vessels can be unavoidable for the next core updates of the network.

The main contribution of this study is to elaborate on how cooperative vessels can improve the maritime network in terms of vessel traffic services by presenting a novel distributed clustering algorithm to help cargo firms and authorities to monitor the network in a better way.

When AIS is adapted to execute the algorithms in the presented approach, ships can benefit from this method to deliver their signals in circumstances in which they can not due to the satellite not being in the area or weak transmission range in adverse weather conditions. Indeed, our distributed approach relies only on AIS signals and does not require any additional signaling or infrastructures.

Furthermore, presenting the concept of cooperation of vessels and the formation of cooperative groups will encourage researchers to investigate new methods to enhance the overall performance of the network from a different point of view when we have clusters.

- We discuss the limitations of AIS and why the maritime network cannot rely on the current system to provide better tracking services.
- We present a new term in maritime network called **Cooperative Vessels** to overcome some limitations of AIS to make ships coordinate and collaborate for the first time.
- We introduce an novel distributed clustering algorithm for maritime network in the middle of the seas and oceans to improve the overall performance of the network in terms of tracking services and reliability of AIS.
- The simulation environment is built from scratch to run the proposed clustering approach with applied optimization techniques.

9.10.2 Outline of the Thesis

The rest of the thesis is organized as follows. Chapter II mentions some related clustering approaches in both maritime network and VANET. The key differences between the both networks are discussed in this chapter to better understand why clustering methods in VANET cannot be applied in maritime network. Chapter III recalls the current principles of the network and describes the limitations of AIS. A novel clustering algorithm for maritime network is presented in Chapter IV followed by optimization techniques. Section VI focuses on numerical study in terms of experimental setup, data collection, simulation parameters, implementation of the algorithm and the obtained results.

9.10.3 RELATED WORKS

Over the last few years, the conception of clustering has come to be an important field of study in many networks especially mobile ad hoc networks. To the best of our knowledge, there are no vessel clustering algorithms in the open sea and ships work on their own without any collaboration with other ships in the area unlike VANET where many clustering methods are presented based on different approaches (partitioning, hierarchical, density-based method, etc.). However, we intend to present a novel vessel clustering algorithm in the open sea to illustrate how cooperative work between ships will make the network and vessel traffic management applications that rely on AIS signals more dependable and safer. Indeed, there exists some studies on maritime networks regarding clustering in ports and coastal areas and also extracting shipping route patterns.

9.10.3.1 Extracting shipping route patterns by path clustering

The maritime routes are known for irregular traffic paths unlike urban roads which are easy to define and follow. Thus, the authors in [4] tried to overcome this issue and extract ships trajectory routes considering dynamic parameters from AIS data. The authors aimed to enhance the safety and security of maritime environment for more stable ports by analyzing AIS datasets to extract shipping route patterns from raw AIS messages and proposed a trajectory clustering model that relies on the location of the vessels which is the minimal set of information required for clustering mobile nodes. However, to obtain better classification results, they enriched the model with other parameters like speed and direction. This study aimed to convert AIS messages to reliable information to help the port managers to get better understanding of the port situation and make related decisions in advance and react immediately in compliance with the current conditions to decrease the risk and accidents especially in crowded ports. The authors in [5] presented a route extraction method based on density clustering algorithm by analyzing S-AIS data. They first grouped the data into sub-trajectories and then DBSCAN method is used for clustering according to structural similarity calculation based on the similarity distance of the sub-clusters. The AIS parameters utilized by this approach are the actual direction of the vessel, vessel position, and speed over ground. The clustering in this case is helpful to notify vessels and the authorities with unusual or irregular ships behavior to improve vessel traffic services. Extracting shipping routes could help also in determine whether the vessel obeys the recommended sailing lane or not.

9.10.3.2 Clustering vessels in the ports and coastal areas

Additionally, some other researchers focused on clustering vessels not the trajectories, but also in the ports and coastal areas [6]. They used a behavioral clustering approach based on k-means theory to classify ships into groups using AIS signals from Rotterdam port to understand the behavior of the ship and predict its pattern for better traffic management. The vessel properties (Vessel Type, Gross Tonnage, Length and Beam) were analyzed to investigate vessel behavior (Heading, Speed Over Ground, Course Over Ground and Position) to prove later that the behaviors can be grouped into clusters based on the defined properties. The only classification parameter is determined by the correlation between these factors, and they have shown that vessel position and course over ground are strongly affected by the beam. On the other hand, the speed is related to the gross tonnage. Thus, the behavioral factors are classified in compliance with the vessel properties. The clustering based on revealed behavioral analysis leads to better prediction of the vessel intention which will help the authorities to monitor ships and estimate their future behavior patterns.

9.10.3.3 Clustering in VANET

As mentioned before, clustering is a wide research topic in VANET which is a fast-changing network in terms of topology comparing with maritime network. Here, the nodes are faster and have higher mobility characteristics than vessels. Thus, routing the messages to their final destination is considered a big challenge in this domain. However, this issue is also addressed by clustering techniques based on dynamic parameters like speed, direction, density, and location of the vehicles. However, the authors in [7] presented innovative approach for clustering in mobile networks which is Coalitional Game Theoretical Approach considering only two parameters which are speed and link quality to ensure higher signal-to-noise ratio (SNR) and establish stable clusters. In such a game as discussed by the researchers, a vehicle can select any other vehicle for cooperating or forming a cluster depending on the gain they expect to get and the cost of the coalition. The vehicles have many strategies depending on traffic density and the neighbors, and it selects to form/join a cluster which provides maximum revenue. The gain function is defined by signal-to-noise ratio, while the cost is determined by the connection lifetime and speed difference. This is similar to our algorithm, the vessel cooperates with a neighbor that provides the best deal but considering the distance, density, position, available resources, and satellite's position as well. Another clustering algorithm in VANET [8] focused more on the continuous connectivity between vehicles even at the cost of higher network overhead. They consider other goals of clustering as secondary objectives and primary goal is to increase connection reliability to improve the network in emergency situations. They proposed a multi-homing clustering algorithm in which each vehicle is connected to multiple cluster heads through multiple connections and channels to avoid connection-less states. However, the overhead in the maritime network is already an issue for the satellite which may not be able to process all AIS data due to its huge footprint and that leads to signal loss and interference when multi-homing approach is applied to vessel clustering. The authors in [9] presented a novel and distributed multi-hop clustering method in VANET based on neighborhood follow approach between vehicles. The proposed algorithm decreases number of isolated clusters, number of cluster heads, and communication cost. The vehicles can exchange information in multi-hop manner through intermediate nodes rather than one-hop that requires cluster members to be directly available to the cluster head. Thus, multi-hop approach can extend cluster coverage but requires complex routing protocol to fulfil the requirements of multi-hop communication between members in multiple clusters not to mention the higher cost of forming and maintaining those clusters. This approach is better from the previous one because each node is connected to only one node which can be a cluster member or a cluster head directly. The majority of other well-known clustering algorithms in VANET are described in [10] and almost all approaches relies on the availability of GPS. Yet, Basu et al. [11] proposed a novel mobility based clustering algorithm in mobile ad-hoc networks that doesn't utilize any absolute location providers. They use the received signal strength as an indicator of the distance between transmitting and receiving node pairs. The mobility metric is the only parameter for cluster formation and cluster head selection procedures.The node with lowest aggregate relative mobility value compared to the surrounded nodes is selected as cluster head, otherwise, it announces itself as a cluster member.

9.11 Methodological Background

The current maritime network and vessel traffic services are based on AIS signals where ships are responsible to broadcast their signals to other ships and satellites as well. However, since maritime transport is getting wider and becoming essential in many industries [12], the network has to be improved and new solutions are required to deal with the increasing number of vessels in the water despite all the limitations of AIS. However, we believe that cooperative vessels can overcome many of them.

9.11.1 Vessel Traffic Services

VTS is traffic monitoring system in maritime network set to help authorities to coordinate traffic in coastal areas by identifying, locating, and observing vessels to increase the safety of maritime traffic and to protect the environment and react to different traffic situations [13]. We did mention that AIS signals were intentionally planned to be used for collision avoidance, but until today not all ships have the device installed because there are specific criteria that have to be met, or else ships do not have to obtain AIS. Therefore, the existence of one ship not having AIS in a crowded area will not guarantee collision avoidance. However, Vessel traffic service is still among the biggest applications of AIS signals at least for ships that are required to have AIS according to IMO (International Maritime Organization) which are passenger vessels irrespective of size, all ships engaged on international voyages with size of 300 gross tonnage, and cargo ships of 500 gross tonnage [14]. Nevertheless, Clustering vessels using a proper algorithm based on dynamic factors to encourage cluster members to work cooperatively, exchanging data and share and manage the available resources efficiently and fairly, will not only lead to better vessel traffic management but will also give the vessels another reason to obtain an AIS transceiver.

9.11.2 Automatic Identification System

The data broadcasted by AIS in a single message can be divided into three groups in terms of information types: static information, dynamic information, and voyage-related information [15]. The dynamic information includes latitude, longitude, speed over ground, course over ground, etc.; these parameters are updated automatically throughout specific time intervals depending on vessel’s movement and message type. Static information which contains

Table 3.1. AIS Signals Time Intervals

Navigation Behavior (Velocity V in knots)	Time Interval (s)
In the anchorage (V= 0)	180
$0 \leq V \leq 14$	12
$14 < V \leq 23$	6
$0 \leq V \leq 14$ and changing course	4

V > 23	3
14 < V ≤ 23 and changing course	2
V > 23 and changing course	2

vessel type, name, dimensions, gross tonnage, etc. is set when the device is first installed. Voyage-related information is defined and updated manually by the crew before the sailing like estimated time off arrival, destination, and draught. However, our research only needs the most dynamic property which is position data. The position-related AIS signal is broadcasted by the vessel at a time interval between 2 and 180 seconds as described in Table 3.1 [16].

Indeed, AIS signals have proven to be valuable source of information for all applications of maritime networks. Yet, it suffers from some limitations.

(i) The accuracy of information received from AIS is depended on the quality of the data broadcasted by that AIS which may not be updated especially voyage-related parameters. The vessel may be sailing at 15 knots at the middle of the sea ant yet showing “at anchor” status.

(ii) AIS is not compulsory on every vessel; even though, the vessels that are enforced by the IMO to obtain these devices can deactivate and turn off the AIS anytime.

(iii) The horizontal range of AIS signals in the coastlines for ship-to-shore area is 50 nautical mile (nm) and for vessel-to-vessel at sea is 20-30 nm under most atmospheric conditions [17] subject to some other factors like elevation, external antenna, and obstacles around it.

(iv) The AIS system could become overwhelmed in crowded areas.

The current infrastructure of AIS allows 64 message types, 27 of them are already reserved [18]. In this paper, we recommend using additional message type called *cluster related data* including information about clusters situation to be used by our presented algorithms.

9.11.3 Satellite vs Terrestrial Networks for Receiving AIS Signals

The terrestrial network is not yet efficient for vessel tracking services because of its range limitations considering seas and oceans. They might be inevitable to be used in ports for clustering, traffic management, path prediction, identification and to get a better understanding of the current situation at ports and coastal areas. But when going far into the open sea, it is hard to get the overall picture of what is going on there using terrestrial network-based receivers, but it does help in navigation for the vessels themselves and ensure the safety of the ships by providing collision avoidance solutions so the ships will be aware of each other and the location of each vessel. Thus, this can only work in coastal zones and ship-to-ship zone. Satellite-based AIS data can be more useful in the seas and oceans when no terrestrial base station can be found to receive and analyze AIS signals. It can provide a global and yet complete picture of the world's maritime network. When satellites are used to handle these data, the term S-AIS is used. While the terrestrial networks can be ideal solution for real-time vessel tracking and positions coverage at thousands of coastal areas and ports, satellites are a promising solution to the next generation of AIS devices [19]. However, we believe that these devices need to be improved also to work better with satellite-based networks since current devices cannot make better use of all the solutions, features, and functions that satellites can provide. CubeSat AIS receivers can be used to detect AIS signals in LEO orbit at the altitude of 650km [20]. Satellites in higher altitudes may suffer from the propagation, AIS packets collision due to the Faraday effect which causes the signal's polarization plane to rotate subject to the elevation angle and magnetic flux intensity [21]. Doppler effect also makes the signal overlap. Hence, the detection probability has to be at its highest which can be obtained by satellites operating in LEO. Thus, to detect and analyze and transmit AIS signals in the open seas, a satellite-based network will help for tracking ships that are beyond the ports and coastal zones where shores cannot receive the signals due to AIS limitations. Hence, S-AIS is an up-and-coming solution to get around the constraints of the terrestrial networks [22]. However, S-AIS still lacks the continuous real-time coverage making the network less reliable for the authorities not to mention the latency and signal collision due to huge satellite's footprint [23].

9.11.4 Cooperative Vessels

Presenting a novel distributed clustering algorithm for vessels that can be functional for maritime networks will prove how cooperative vessels can make the network more reliable and safer than when each one is working on its own. When ships work together to deliver AIS signals to satellite or shores, VTS can be more dependable, and the authorities can have better picture of the sea even beyond both terrestrial and satellite-based networks. Maritime network has become larger since 90 percent of global trade volumes is being carried by the sea currently [24]. Thus, the development in the structure and presenting original approaches and solutions are inevitable in the network to keep pace with its increasingly crucial role in the global economy. Cooperative vessels can only be the first step to improve reliability and stability of the network.

9.11.5 Dark Activity

Nothing can stop vessels from turning off their AIS and go dark due to many legitimate or illegitimate reasons. The authorities cannot tell for sure whether a ship switched off its AIS to hide its location for some illicit dark activities (e.g., fishing in restricted areas and dark trade) [25]. The ships can always deny this activity because the signal can be lost the way to the satellite especially in congested waters. Thus, only a nearby ship can be aware of the dark activity. The ratio between lost signals unintentionally due to conditions beyond AIS and deactivating the transmitter on purpose is something between 1:10 and 1:20 depending on ship type and geographical area according to Ron Crean, vice-president for commercial at Windward Maritime Analytics [26].

However, we believe that vessels working cooperatively can overcome this issue; obviously, not if the AIS was turned off before sailing in the first place.

9.11.6 Orbital Elements

The classical orbital elements are the six Keplerian elements, the proportion of Johannes Kepler and his laws in relation to planetary motion, are the parameters that can be determined for each stage of the orbit, usually indicated by a particular time, epoch, and point of reference.

9.11.6.1 Definitions

Orbital Period

The orbital period is defined by the time taken by an astronomical object to rotate in its orbit around another for one complete revolution.

A satellite's period is the amount of time it takes to make one full orbit around a planet, Earth. If the satellite is placed high above the surface of the planet, it will take a long time to complete its orbit. However, if the satellite gets closer to the surface of the planet, or to a lower altitude, it will take less time to complete its revolution - and its period will be shorter.

The time needed for a satellite for one period in the orbit is given by the formula:

$$P = \frac{2\pi}{\sqrt{\mu}} \left(R_g + z\right)^{3/2} \qquad (3.1)$$

- μ is the gravitational constant for the Earth equal to $398600 km^3/sec^2$.
- R_e is Earth's equatorial radius $6378 km$.
- z is the altitude.

Mean Motion

Mean motion for a satellite is described as the number of orbits traversed in one day around earth.

It can be calculated using the formula:

$$n = 2\pi/P \tag{3.2}$$

Where n is the mean motion in rad/sec and P is the period of the satellite in seconds.

TLE Data

Position of satellites is usually calculated by some ground stations mechanisms like radar and laser reflectors.

However, satellite location can be approximately determined by numerical approach making use of some data called the TLE with orbital elements which we are going to describe in this section. Tow-Line Element or as called TLE is used to encode the orbital elements of a body orbiting the earth for a specific time, Epoch. This data can be used to predict the position and velocity of the body at any time in the future or the past accurately to an acceptable extent [27]. TLE is also used for maneuvering operations, risk analysis, and collision avoidance for the objects orbiting the earth.

A TLE file is made of two lines of data, sometimes preceded with another line defining title or name of the satellite, but it is not a mandatory line. These two lines are actually Brouwer-Lyddane mean orbital elements and not Keplerian elements which defines the parameters of the satellite orbit [28].

The main difference that Keplerian elements have six parameters describing together the shape, size and orientation of the orbit and the position of the object in the orbit presuming there are not any forces apart from the gravitational force eliminating all perturbing forces.

On the other hand, Brouwer-Lyddane Mean element is set of the same parameters used by Keplerian elements. But the values are not constant, instead they are averaged over time considering perturbing gravitational forces. Thus, it is an advantageous orbit element type for orbital design. Figure 3.1 explains the TLE data and the definition of each element [27].

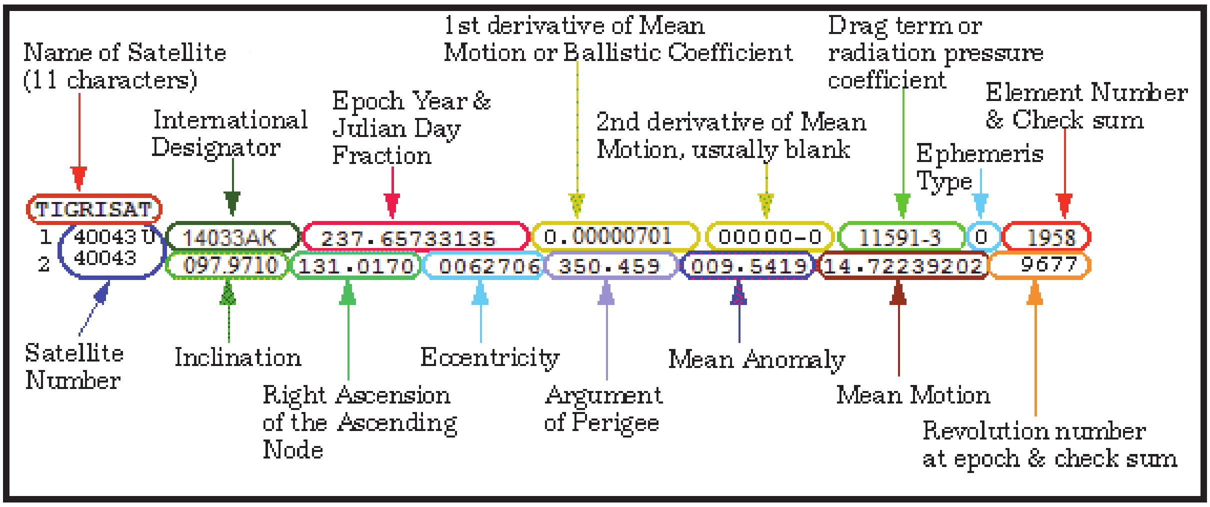


Figure 3.1. TLE elements

9.11.6.2 Semi-Major Axis

Semi major axis represented by letter *a* describes size of the orbit it which the object is rotating around earth and the time each revolution takes to complete one orbit [29].

Since the earth is not a perfect circle, satellite is sometimes closer to the earth and at other times farther away. Thus, we have got two definitions [30]:

- Apogee: The point in the orbit of the satellite in which the satellite is at its farthest from earth r_a.
- Perigee: Nearest point from the earth on the orbit of the satellite r_p.

$$r_a = a(1 + e) \tag{3.3}$$

$$r_p = a(1 - e) \tag{3.4}$$

Where e defines the eccentricity of earth, the third orbital element.Thus, Semi major axis is half of the major axis which is the distant from apogee to perigee.

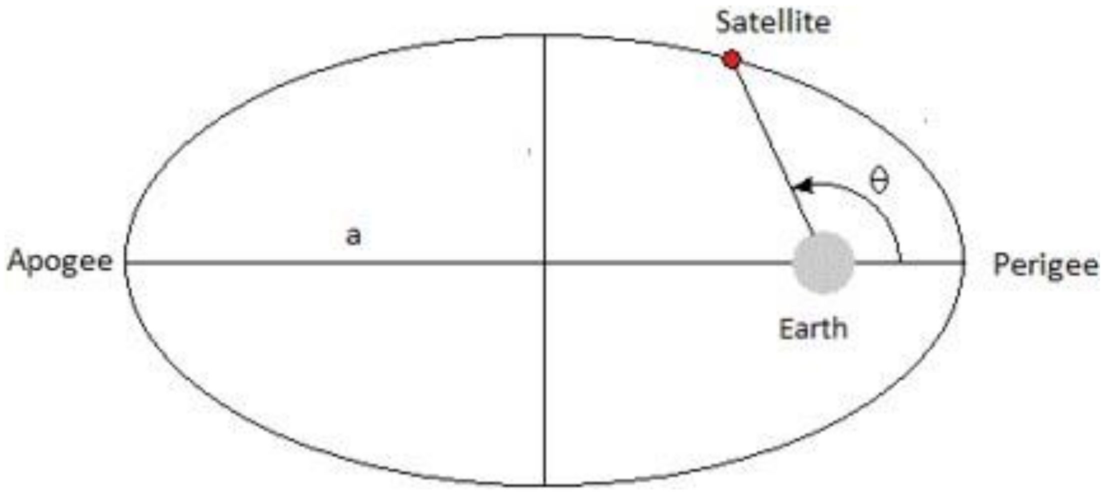


Figure 3.2. Semi-Major Axis

9.11.6.3 Inclination Angle

The angle of inclination is the angle between the reference plane and the direction axis [31]. It is an orbital element that determine the shape and direction of astronomical orbits. Thus, the angle made by the orbit of the satellite with equator line of the earth defines the inclination of the orbit The orbit plane of a satellite that rotate around the Earth straight above the equator line is the same as the equatorial plane of the Earth, and the orbital inclination in this case is 0°.

Table 3.2. Inclination Angles

Inclination Angle *i*	**Description**
0°	The orbital object has a prograde orbit in the equatorial plane of the planet
$0° < \theta < 90°$	Prograde orbit but not as same as the equatorial plane
90°	Polar orbit, satellite passes over the poles of the planet
63.4°	Critical inclination, zero apogee drift in elliptical orbits
$90° < \theta < 180°$	Retrograde orbit, in which the satellite's position on the equatorial plane is projected in the opposite direction of the Earth's rotation
180°	Retrograde equatorial orbit

9.11.6.4 Eccentricity

Table 3.3. Eccentricity of Shapes

Eccentricity Value	**Description**
$e = 0$	Circle
$0 < e < 1$	Ellipse
$e = 1$	Parabola
$e > 1$	Hyperbola
$e = \infty$	Line

The eccentricity is a positive real number defines how much a conic section deviates from being a circle. Thus, circle has eccentricity of 0, while eccentricity for earth has a value between 0 and one since it is elliptical. In orbital elements, eccentricity describe the shape of the orbit [32].

Eccentricity is the ratio of the distances between the two focus points of the ellipse and the length of the major axis.

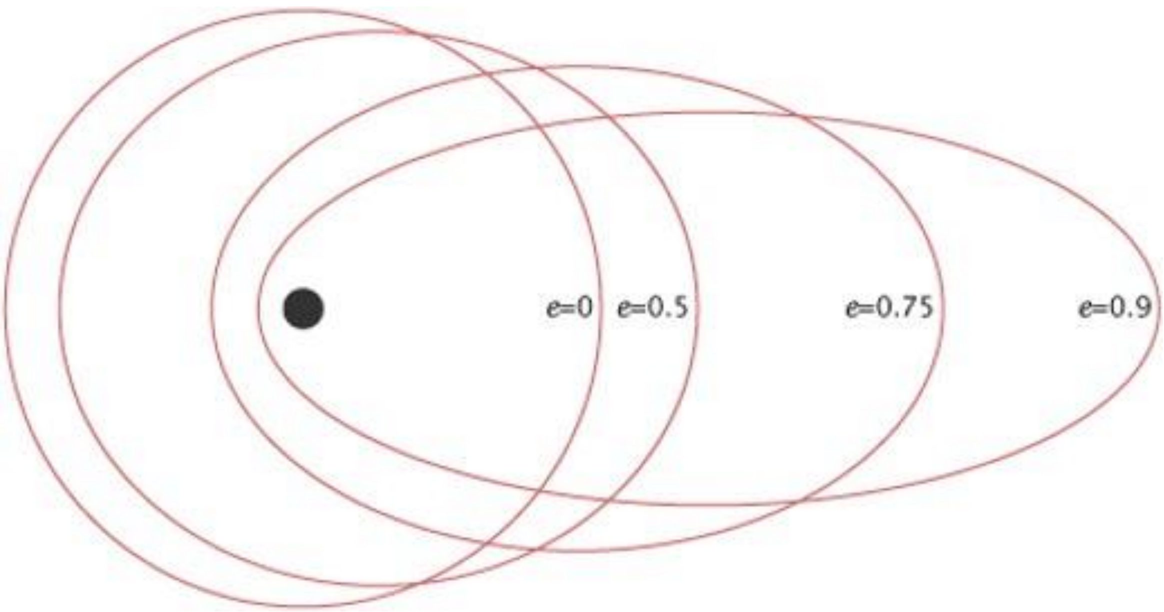


Figure 3.3. Satellite orbits with different inclination angles

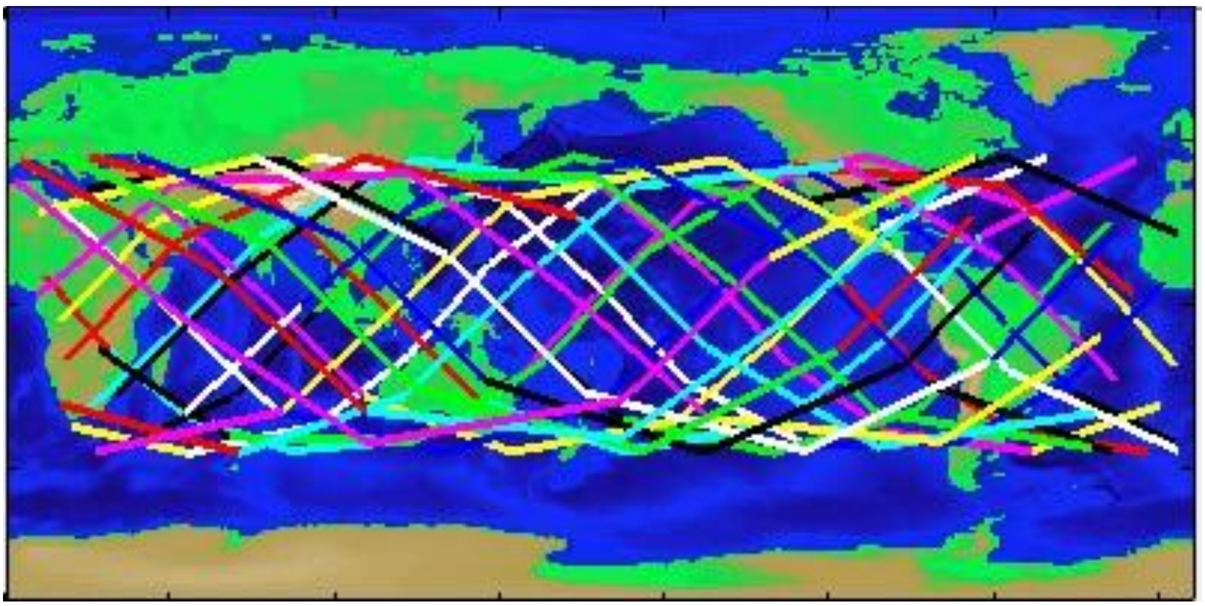


Figure 3.4. A total of 100 satellite periods with $e = 0.1$, $i = 45$

9.11.6.5 Mean Anomaly

The mean anomaly M can be defined as an angle θ ranging from 0 to 2π radians which describes the location of the satellite it its orbit at a specific epoch time relative to perigee as shown in Figure 3.2. It is a fraction of period elapsed since the satellite passed perigee. Mean anomaly at perigee is zero and reaches to 180 degrees at apogee before getting back to perigee at 2π.

The mean anomaly M at any given epoch time t is calculated by adding the previous known mean anomaly M_0 to the mean motion multiplied by the time elapsed since M_0.

$$M(t) = M_0 + n(t - t_0) \tag{3.5}$$

It can also be calculated if we know time of the previous perigeal passage t_p using the formula:

$$M = n(t - t_p) \tag{3.6}$$

9.11.6.6 Argument of the perigee

Satellite drops through the equatorial plane in two points, Ascending node and descending node.

- Ascending node where satellite intersects the equatorial from south to north.
- Descending node when the satellite passes the equatorial plane from northern hemisphere to the south.

Argument of the perigee is the angle along the orbital path between the Ascending node and the perigee point and measured always in the orbital plane at earth's center in the direction of the object motion [31]. it ranges from 0 to 360 degrees, and for circular orbits, the value is zero.

9.11.6.7 Right ascension of ascending node

Right ascension of ascending node RANN is also called longitude of the ascending node represented with Greek letter Ω and define the orientation of the orbit in the space [33].

It is an angle measured counterclockwise between the first point of aries and the ascending node of the orbit.

9.12 Distributed Vessel Clustering Algorithm (DVCA)

We present a novel and fully distributed vessel clustering algorithm (DVCA) for maritime networks so that vessels are able to deliver their AIS signals to the ground stations even if they are not in the satellite's footprint area or beyond terrestrial networks. Before the satellite passes over a region, we intend to do some signal pre-processing operations to get a better understanding of the situation in the water and how the vessels are positioned by grouping vessels into clusters based on their location with respect to the footprint area. Thus, each ship depending on its distance to the closest footprint will carry specific number of other ships' signals which are not in the footprint or will not be anytime soon.

9.12.1 Assumptions

For a ship equipped with AIS transponder to be involved in the clustering operation, it is expected that:

(i) The ship knows when the next closest satellite will be passing over its region or a nearby region.
(ii) The ship can calculate the distance between any two locations (Distance to its neighbors and distance to the closest footprint) which can be done by Haversine formula [34].
(iii) The ship is able to store and track some data about its neighbors and cluster situation.
(iv) The *cluster related data* AIS signature is defined with a minimum set of fields including cluster ID, position data, empty value, priority and satellite access bit. These parameters will be described in details in the main text.

9.12.2 Functions

Figure 4.1 illustrates high-level flowchart of the distributed clustering approach. The ship who wants to carry a signal is named as *the carrier*, and the ship whose signal is being carried is named as *the acceptor*. Obviously, a ship can be a carrier and an acceptor at the same time, it may be carrying some signals and has a carrier as well. Each ship has Cluster ID which is initially set to nil. A ship accepts a proposal by updating its Cluster ID to carrier's Cluster ID. If the proposal is accepted, the `Carry` function takes place.

The last carrier in the cluster who sets the *clusterid* carrying all previously carried signals forwarding to the satellite is called *cluster head*. The cluster head which is in the footprint and

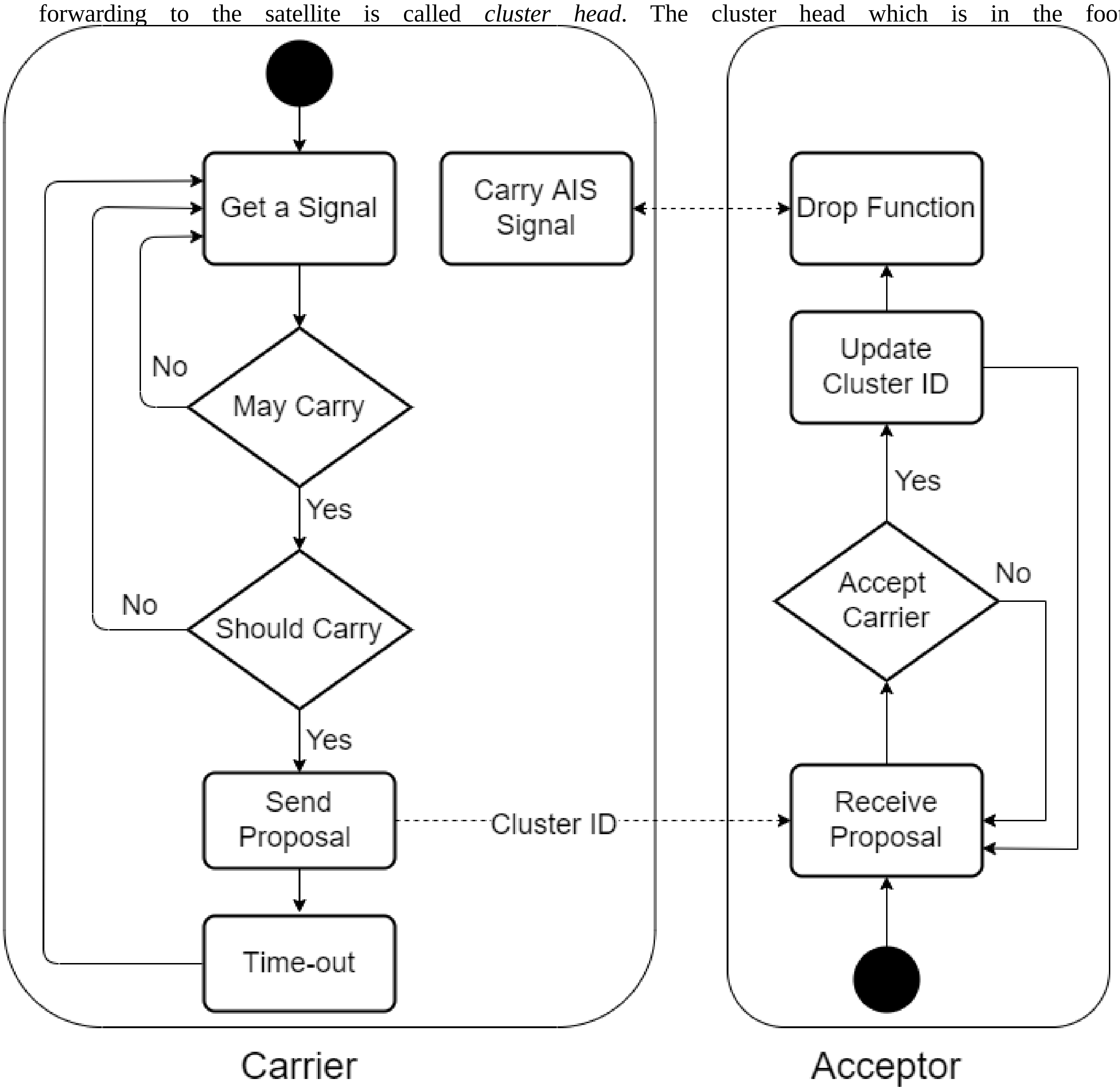


Figure 4.1. Distributed Vessel Clustering Algorithm

is actually delivering all carried signals to the satellite is then called the *cluster carrier*. Thus, the cluster may not have a cluster carrier.

There are three main functions a carrier has to consider while making decisions upon carrying another ship's signal or not. These functions are `May_Carry`, `Should_Carry` and `Carry`. Another function in the acceptor side, `Accept_Carrier`, is applied against the proposal to decide whether to accept the carrier or not. If the carrier is accepted, acceptor may have to drop some signals it has already carrying, according to the available resources at the carrier side, which is done in the `Drop` function.

Some conditions in the functions related to carrying or accepting signals may be reconsidered after a while and the decision might be changed in the future according to specific circumstances and scenarios since the third assumption allows us to track data which is necessary for optimization purposes as will be described later.

9.12.2.1 May_Carryfunction

This function decide whether a carrier "may carry" a signal or not. If the result of this function is "no", the signal is simply discarded. If it is "yes", then the second decision step will take place to decide whether the signal should be carried or not. Before describing the `May_Carry` function, we define the following parameters:

r: AIS range, maximum horizontal range of AIS signal. Typically r may change according to

the AIS class of the ships. We set r as the minimal horizontal range of vessels involved in the clustering algorithm.

md: Distance from satellite to the farthest ship allowed to create or join a cluster. It is desirable to have $md \mod r = 0$.

f: Number of signals the farthest ship can carry. It is reasonable to set this value to one by

default, because in order to carry more signals, md can be increased instead of f. However, it can still be adjusted for crowded areas.

d: Distance from the ship to satellite. If the ship is in the footprint, then $d = 0$.

hd: Hop distance from the ship to satellite. It is actually equal to the minimum number of ships that is needed to send the signal to the footprint area, and defined as

$$rd = \lceil d/r \rceil \tag{4.1}$$

$n(hd)$: Number of signals each ship can carry depending on its hop distance (hd) to satellite. This value is updated dynamically in time.

If $d > md$, then $n(hd) = 0$, i.e. the ship cannot carry any signal. Otherwise, the number of signals the ships are allowed to carry increases when the ship gets closer to the footprint area. We use the following equations to set the value of $n(hd)$:

$$n(hd) = \begin{cases} 0, & \textbf{if } hd > md/r \\ f, & \textbf{if } hd = md/r \\ n(rd+1)+h, & \textbf{otherwise.} \end{cases}$$

where we set *h* as

$$h = \begin{cases} 2, & \textbf{if } hd >= md/2r, \\ 1, & \textbf{otherwise.} \end{cases}$$

For the ships that are far away from the footprint, we increment $n(hd)$ by 2, while for the ships closer to the footprint we increment by one. The particular reason behind this is the fact that the ships that are closer to the footprint area may have more opportunity to find other alternatives for sending their signals, as well as they have more probability to be covered by a satellite in a near future. Therefore we decide to favor the vessels that are far away from the footprint area.

For the standard parameters in this research $r = 40km$, $md = 160km$ and $f = 1$, we need at least 5 ships with the n values shown in Table 4.1.

Table 4.1. *n* values for the standard parameters

Distance from the vessel to the nearest satellite	*n*
d>160	0
120<d≤160	1
80<d≤120	3
40<d≤80	5
0<d≤40	6
d=0	7

If we define *empty* as the capacity parameter considering the number of signals a ship can carry, then its initial value is *n* and it is decremented by one for every carried signal and incremented by one when a signal is released.

We want to draw attention to the fact that the maximum number of signals that can be carried by a vessel in the footprint area could be limited according to technical reasons or regulations. This requirement is satisfied by adjusting two parameters, namely *md* and f.

However, in crowded areas it is better to have lower *md* with higher *f*. While in the middle of the oceans higher *md* is recommended.

Algorithm 1: `May_Carry` Function

Data: Carrier ship *c*, signal to carry *s*.

Result: Decide whether *c* may carry *s* or not. $carry \leftarrow true$;

if $c.empty > 0$ ***or*** $s.d = 0$ ***or*** $s.d > md$ ***or*** $s.d < c.d$ ***or*** $|s.d - c.d| < 2km$ **then**

  $carry \leftarrow false$;

return carry

Algorithm 1 gives the decision factors on the possibility of carrying a signal considering the parameters and current situation of both vessels. A signal is simply discarded if there is no empty space in the carrier, or if signal to carry is already in the footprint or closer to it than the carrier, or if the difference between their distances to the satellite is less than 2 kilometers. The latter condition is included to not overwhelming the network, and will be illustrated in detail later while describing the scenarios.

Figure 4.2 represents a set of ships in an area with one satellite around. According to the previous function we can see which ship may carry another ship's signal:

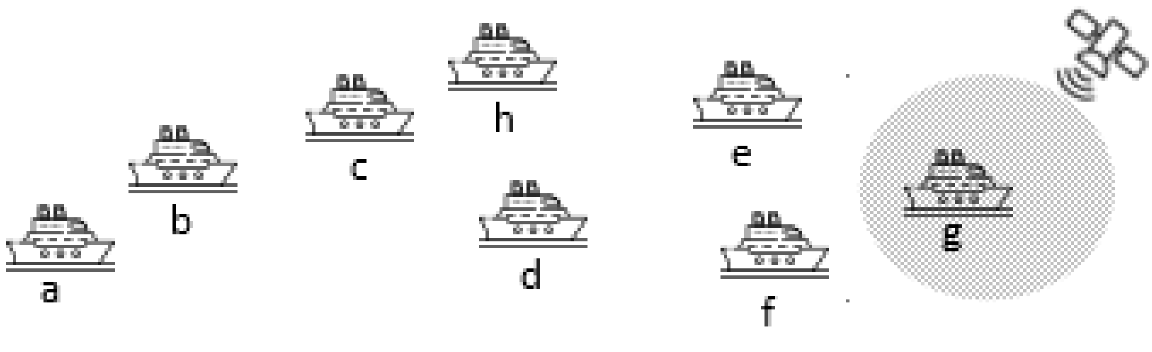


Figure 4.2. `May_Carry` Function example

- *b* will not carry *a* ($a.d > md$);
- *c* may carry *b*;
- *d* may carry *h*;

Table 4.2. Distances for Figure 4.2

	a	b	c	d	e	f	g	h	sat
a	0	30	-	-	-	-	-	-	185
b	30	0	35	-	-	-	-	-	155
c	-	35	0	36	-	-	-	20	115
d	-	-	36	0	35.5	37	-	33	75
e	-	-	-	35.5	0	25	36	38	28
f	-	-	-	37	25	0	34.5	-	26.5
g	-	-	-	-	36	34.5	0	-	0
h	-	-	20	33	38	-	-	0	90

- f will not carry e ($|f.d - e.d| < 2$km) (in coming scenarios it may, when optimization techniques are applied);
- g may carry e and f;
- f will not carry g ($g.d = 0$);
- c will not carry d ($d.d < c.d$) (in coming scenarios it may, when optimization techniques are applied).

9.12.2.2 Should_Carryfunction

This function will also be run by the carrier ship against a signal it may carry to investigate the shared area between carrier and that signal. The primary goal is to analyze the neighbors of both ships and check whether there exists another ship with empty space in the range of the signal to carry, that is closer to the satellite. If such neighbor exists, then the signal it may carry is discarded to give the other neighbor opportunity to carry it. This way we also make sure that every ship tries to carry farthest signals as much as possible. Algorithm 2 illustrates final decision procedures on carrying a signal or not, by exploring the shared environment.

Consider again the Figure 4.2. *d* should not carry *h* because *e* is a shared neighbor, and it is closer to the satellite. Thus, it is better for *d* to carry *c* instead, even if it has space for both signals.

When the carrier takes decision in favor of carrying a signal, it broadcasts this signal under its own *clusterid* indicating updated *empty* value of the acceptor which is the number of

Algorithm 2: `Should_Carry` Function

Data: Carrier ship *c*, signal it may carry *s*.

Result: Decide whether *c* should carry *s*.

carry ← *true*;

foreach $n \in c.neighbors$ **do**

  if $n.d < c.d$ ***and*** $n.empty > 0$ ***and*** $n \in s.neighbors$ **then**

    carry ← *false*;

return carry

additional signals the acceptor is allowed to carry. The acceptor is asked to drop some carried signals by setting the *empty* value to negative. Then it waits for the acceptor to take action. The proposal sending operation to nearby vessels is repeated within an interval defined by the authorities, although it needs further investigation.

9.12.2.3 Accept_CarrierFunction

After a ship decide to carry a vessel's signal, it sends proposal to it by repeating the signal with its own *clusterid* as mentioned above. If the acceptor accepts the proposal, it will update its *clusterid* accordingly and all the signals the acceptor already carrying will do the same thing. Indeed, the carrier may not carry (repeat) all signals the acceptor carries. Thus, some signals may by dropped and they will update their *clusterid* to nil to receive proposals from new carriers.

In Algorithm 3, *s* is looking for a better carrier if the proposed carrier is not already in the footprint. The *access_bit* is used by a vessel in the footprint to inform another vessel -directly or through cluster members recursively- that its signal has been sent to the satellite by broadcasting its signal with *access_bit* = 1 until it get out of the satellite's coverage area.

If there exists a neighbor with *access_bit* = 1 and has empty space more than the proposed carrier has, it will reject the proposal and wait for new one from that neighbor *n*. This might be risky because the neighbor may be filled up before it realizes *s* or it may send proposals to other ships after waiting for a timeout period. However, *s* can accept the proposal anytime and the probability that both became filled up in the same time is low, so it is better for *s* not to wait too much for a proposal from *n* and follow the updates carefully.

Algorithm 3: `Accept_Carrier` Function

Data: Proposal by *c*, signal to be carried *s*

Result: Decide for *s* whether to accept carrier *c*

if *s.clusterid==nil* **then**
    accept ← *true*;
if *c not in footprint* **then**
    foreach $n \in s.neighbors$ **do**
        if *n.access_bit=1* ***and*** $n.empty \geq c.empty$ **then**
            accept ← *false*;

if *accept* **then**
    s.clusterid ← *c.clusterid*;

return accept

9.12.2.4 `Carry` Function

Carry function is the core function in the proposed clustering algorithm for maritime network. The function takes place when the acceptor accepts the proposal and broadcast its AIS signal under the potential carrier's Cluster ID.

Intuitively, the carrier should carry all the signals the acceptor already carrying and the fact that the empty space at the carrier may not allow to carry all those signals is demonstrably discernible. Thus, some of them will have to be dropped in the `Drop` function shown in Figure 4.1.

There are three mechanisms for dropping signals:

(i) Drop at the carrier: The carrier repeats the acceptor's carried signals one by one until it get filled up. The acceptor consequently drops the unrepeated signals and so will do the other members of the cluster. If the acceptor is carrying 5 signals and the the carrier has only 4 empty spaces, meaning that two signals has to be dropped. However, the carrier may drop a vessel that is carrying 2 signals, which in turn, will lead to dropping 3 signals. Another solution for the carrier is to drop the farthest signals to overcome this problem, but as mentioned, it is better to carry farthest signals. This mechanism may be fast but not effective especially when the priority function (that will be mentioned in Section 5) is concerned.

(ii) Drop at the acceptors: The acceptor knows how many empty place the carrier has, and it knows the conditions of its own acceptors as well. Thus, it chooses a vessel (vessels) considering priority values, if that vessel has no carried signals, it is dropped directly, otherwise, the mission is passed to it. This is effective but time consuming.

(iii) Hybrid: Only the current acceptor decides which signals to be dropped without involving the acceptors behind. This approach is fast but not always effective, especially when its acceptors are not empty.

The analysis of each of these methods requires further studies. In this work, we adopt the second mechanism.

Algorithm 4: `Carry` Function

Data: Accepted proposal from *s*, carrier *c*

Result: *c* carrying *s*

```
/* icarry= carried signals by a vessel                              */
```

$c.icarry.push(s)$;

$c.empty \leftarrow c.empty - 1$;

s.drop();

foreach $n \in s.icarry$ **do**

  $c.icarry.push(n)$;

  $c.empty \leftarrow c.empty - 1$;

foreach $n \in c.icarry$ **do**

  $n.empty \leftarrow min(n.empty, c.empty)$;

Setting acceptor's empty value by the carrier while sending proposal is crucial for the adopted dropping mechanism. If this value is negative, the acceptor will drop that number of signals by the recursive flow chart shown in Figure 4.3. Indeed, the carrier may carry all the signals carried by its new acceptor but ask other acceptors (if any) to drop signal(s) according to priority values. When an acceptor is told to drop a signal, it first checks the priority values of its own acceptors. If the one with lowest priority has no carried signals, it is dropped or else the mission is delivered to it by setting its empty value to negative to run the recursive `Drop`

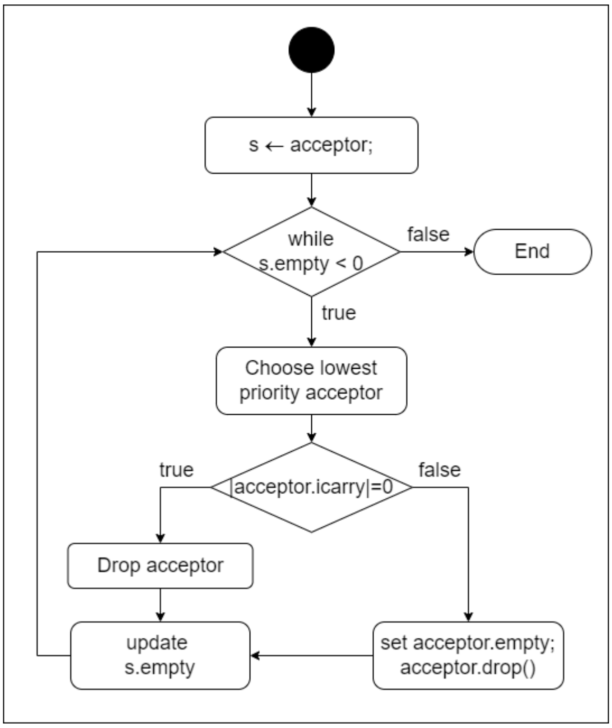


Figure 4.3. Drop Function

function, and so on. For every dropped signal, the negative empty value is increased by one. To drop this signal, carrier repeats it for the last time, by setting its *clusterid* to nil. The number of signals to be dropped can be divided between multiple acceptors if they have same priority values. Algorithm 5 describes the operation of choosing an acceptor to drop signals. If the priorities are the same, then the number of carried signals and distance to satellite is considered respectively.

Algorithm 5: `Lowest_Priority_Acceptor` Function

Data: Acceptors *acceptors*[] ordered by *d* in ascending order

Result: Lowest Priority Acceptor For Signal *s*

`/* R is priority value */`

lowest = acceptors[0];

foreach $n \in acceptors$ **do**

if $n.R < lowest.R$ ***or*** $(n.R = lowest.R$ ***and*** $|n.icarry| < |lowest.icarry|)$ **then**

$lowest \leftarrow n;$

Fairness may come to be an issue here since we do not consider the priorities of all the signals carried by the acceptors. An acceptor may be dropped although its priority is higher than at least one ship carried by another acceptor. However, this way we still have some preferable aspects:

- We avoid dropping signals to the greatest extent in chain topologies which are the best form of clusters since they carry signals from remoter distances.
- We boost the term of cooperative vessels which ensures higher priority values for the carriers as discussed in section 5.

9.12.3 Failing Scenarios

9.12.3.1 Closer to Satellite Failure

Let's refer to the fourth condition in `May_Carry` function, $s.d < c.d$. Here, if the signal to carry is closer to satellite than the carrier, the carrier will not carry it. However, this may cause a problem if there are no other ships to carry that signal the way to satellite as in the following examples:

c will not carry *b* because *b* is closer to satellite than *c*. *c* will not carry *d* because *d* is closer

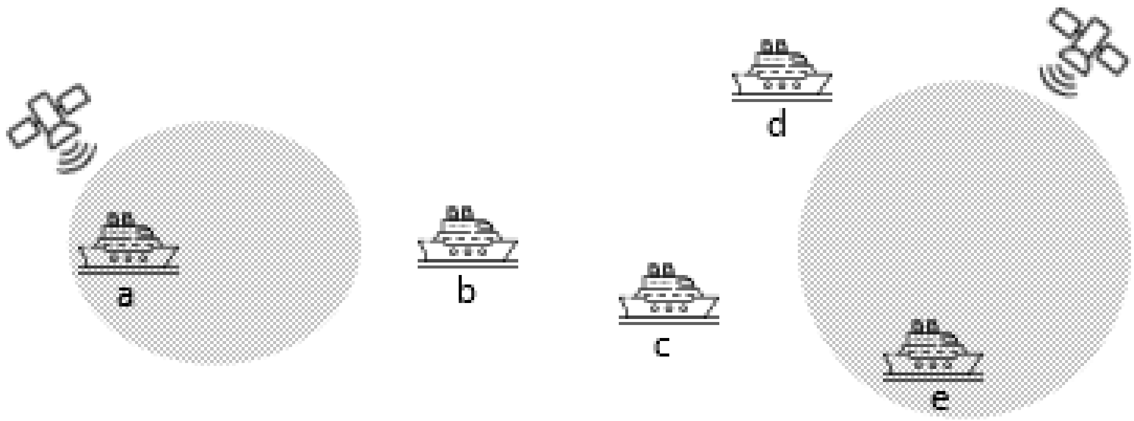


Figure 4.4. Closer to Satellite Failure

to satellite than *c*. The result is that both *b* and *d* will not be able to deliver their signals to the satellite while in a better case they should.

Table 4.3. Distances for Figure 4.4

	a	**b**	**c**	**d**	**e**	**sat**
a	0	-	-	-	-	0
b	-	0	13	-	-	20
c	-	13	0	37	38	20
d	-	-	37	0	-	10

e	-	-	38	-	0	0

9.12.3.2 Low Distance Failure

The last condition in `May_Carry` function $|s.d - c.d| < 2$km may produce similar issue. This condition is there to not overwhelming the network because it is better to carry farthest signals since a signal who is behind *c* by only 2 km may get proposal from another ship who is in front even from *c*'s carrier itself. Therefore it would be nonsense to carry a signal *a* to *b* through *c* where *b* can carry *a* directly without carrier in middle. But if there is no vessel to carry signal *a* other than *c*, *a* will not be delivered.

9.12.3.3 Poor Carriers

In `Accept_Carrier` function, if signal to carry's *clusterid* is not nil -meaning that it already has a carrier-, the proposal will be rejected, but what if the current carrier could not deliver the signal and dropped it. Thus, there need to be a mechanism for tracking data to not accept proposals from previous carriers which has dropped the signal before, and wait for other proposals.

9.12.3.4 Overloaded Carriers with Undistributed Signals

This scenario is critical, and it is the biggest issue in the clustering algorithm so far as explained in the bellow diagram. As it can be seen, ship *b* could not deliver its signal because ship *c* is already carrying 7 signals. However, in the best case, *e* should carry *a* to let space in *c* to allow *f* to carry *b*. This happened because *e* was last broadcasted, and without *e* in middle, *d* cannot carry *a*. We can be sure that this scenario will happen in so many areas in a real map.

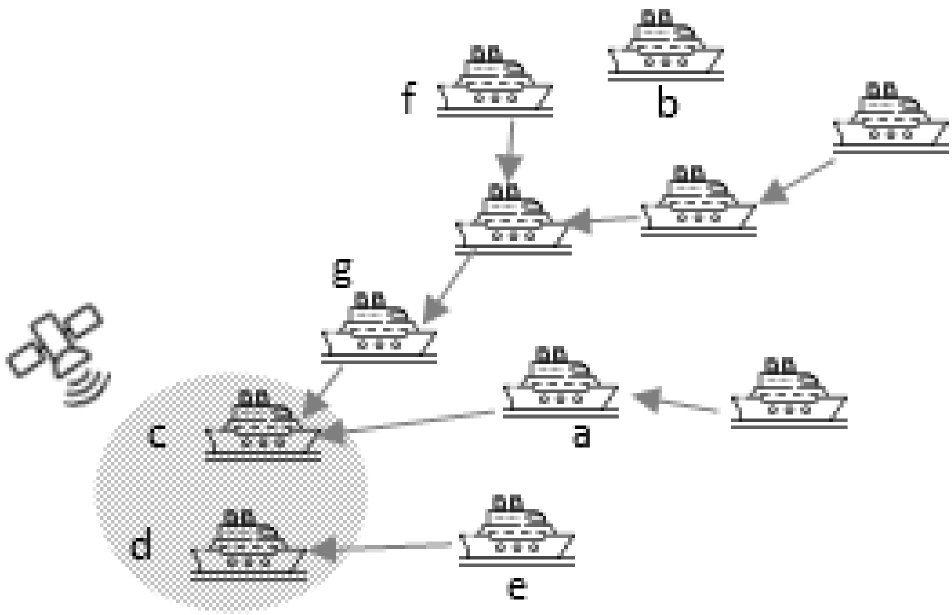


Figure 4.5. Overloaded Carriers with Undistributed Signals

9.12.4 Dark Activity Detection

Cooperative vessels can play an important role in dark activity detection because when a ship turns off its AIS transmitter, only its neighbors can be aware of this behavior. Every ship's role in the water will come to be crucial sooner or later in clustering algorithm. In the previous diagram, if ship *g* turns off its AIS, it can affect 5 other ships to be invisible to the authorities. However, *c* can detect this activity at some probability if the ship is supposed to be in its range for specific period. When *c* receives *g*'s signal, it will know *g*'s speed and direction from AIS data (speed over ground, course over ground, heading). This way, *c* can compare this data with its own and calculate how long *g* is going to be in the range. This can also be an additional parameter to be used in `May_Carry` function in future work.

Exponentially weighted moving average is better choice to calculate the period.

a: calculated period using the last AIS signal.

α: alpha value between 0.80-0.85.

p_l: current period given by:

$$p_l = a * \alpha + (1 - \alpha) * p_{l-1} \tag{4.2}$$

if $t = min(p_l, 180s)$ passes since the last check, the probability that the ship went dark is given by:

$$P_d = (n - l * n - s)/(t) * (i) * \omega \tag{4.3}$$

t: time in which the accused ship should have been in range.

n: number of signals should have been received in t.

l: percentage of lost AIS signals in the area.

i: Weighted average of signal intervals.

ω: Number between 0-1. This value is related to the risk of the area, nearby forbidden areas, and type of the ship.

s: number of received signals in t.

The idea behind 180s is that the period might be long, yet the ship should check the dark activity frequently at least once every 3 minutes.

Figure 4.6. Dark Activity

We can see that after 00:00:30, signal *b* isn't received anymore but it is supposed to be in *a*'s range for more 76 seconds.

If no signal from *b* is received until 00:01:48, ship *a* will check the dark activity for ship *b*:

P_d= (12.74-12.74*0.07-0)/(76.46)*(6)*0.9= 0.84

If the last period was more than 150s, let's say 160s, then the *a* would check the dark activity twice:

- The first time after 180s passes and the probability would be:

  n= 25+7 (150/6=25; number of signals should have been received after 00:00:30. 7; number of signals should have been received before that. Please note that the interval changes to 6s after the speed becomes under 24knots).

Table 4.4. Dark Activity Detection

Time	a (km/h)	b (km/h)	Distance (km)	p (s)
00:00:00	70	50	35.00	150.00
00:00:03	74	45	35.06	149.54
00:00:06	75	41	35.15	150.32
00:00:09	78	-	35.28	144.03
00:00:12	80	40	35.43	138.07
00:00:15	85	-	35.62	127.88
00:00:18	88	40	35.85	118.48
00:00:21	90	-	36.10	109.47
00:00:24	95	41	36.40	97.42
00:00:27	97	-	36.73	87.03
00:00:30	100	40	37.10	76.46
00:00:33	101	-		-
00:00:36	100	-		-
00:00:39	102	-		-
00:00:42	104	-		-
00:00:45	105	-		-
—	104	-		-

—	105	-	-
00:01:48	104	-	-

P_d= (32-32*0.07-7)/(180)*(5.71)*0.9= 0.65

- The second check is after the period passes:

P_d= (26.6-26.6*0.07-0)/(160)*(6)*0.9= 0.84

If no signal is received at all in any period, the probability is $(1-l)*\omega$.

9.13 Optimization Techniques

9.13.1 Tracking Data

As a carrier, `May_Carry` and `Should_Carry` functions prevent a ship from sending proposals to neighbors that are behind by only two kilometers or less. And also, to those who are closer to satellite. However, sometimes it has to send proposals because there may not be other ships around to carry their signals. Thus, the carrier stores data related to these ships and sends proposal to them after waiting for a time-out value, *td_timeout*. This parameter was set to 60 seconds in our study.

Another reason for tracking data is to help ships making better decisions when accepting proposals. A ship may drop a signal for one of the following reasons:

(i) It could not deliver the signal to the satellite because it is a dead node, there are no ships in its range to deliver signals forward.

(ii) All of its neighbors are full.

(iii) Its own carrier could only carry some of signals because $|s.icarry| + 1 > s.carrier.empty$, and the ship had to drop some signals.

(iv) The ship preferred some other signals to carry (see priority function).

A vessel may store information about the carriers which dropped its signal before to reject coming proposals from them and wait for a better proposal. If there is no other, it waits for *td_timeout* and accepts any proposal.

9.13.2 Load Balancing

It is not a good practice if a ship is carrying 5 signals and its neighbor is carrying only 1, or if it can carry 6 but is carrying only 4 because its carrier has no empty space. Even though `Should_Carry` function investigates the area before sending proposal to help making the network more distributed, the topology may change anytime. Thus, a mechanism allowing the acceptor to change its carrier will provide better distribution value.

Distribution value in any given area *A* is given by:

$$D(A) = sd((i.n - i.icarry) \quad i \in A \ \&$$

$$\exists j \in i.neighbors : j.clusterid \neq nil) \tag{5.1}$$

which is the standard deviation of unused capacity of the ships in area *A*, considering only the vessels that sent their signals by the clustering algorithm and their neighbors. Unused capacity of a ship is given by the number of signals it can carry minus the number of signals it is actually carrying. The aim is to decrease *D* value to distribute signals between vessels and obtain higher functionality.

Better *D* value can be achieved by `Load_Balance` function given in Algorithm 6, which aims to decrease loads on current carrier and choose another one to balance the load as much as possible. The ship will analyze the area looking for a neighbor with cluster or a neighbor in footprint and compare the number of signals carried by its current carrier and the new candidate, and decide accordingly. Obviously, it should first receive a proposal from that neighbor.

Algorithm 6: `Load_Balance` Function

Data: Ship s, its neighbors

Result: Ship s changes its carrier

foreach $n \in s.neighbors$ **do**

  if $(|n.icarry| + |s.icarry| + 1) < |s.carrier.icarry|$ ***and*** $n.empty > (|s.icarry| + 1)$ **then**

    $wait\_and\_accept\_proposal\_from(n);$

9.13.3 Local Optimality

The `Load_Balance` function does not guarantee local optimality since a ship may have a neighbor without cluster but cannot carry it because its carrier's *empty* is zero. In spite of that, it may not change its carrier because there would be no change in terms of load balance.

Consider the scenario in Figure 5.1. *a* cannot carry *d* because *empty* value of *a*'s carrier is zero (hence its own *empty* value is zero as well). If any of the ships *a* or *b* changes its carrier to *c*, *a* will be able to carry *d*. But they will not do this in `Load_Balance` function because in both cases the load on their carrier will not change (for e.g. *b*'s current carrier *e* carries two signals, if *b* changes its carrier to *c*, then *c* will carry two signals as well).

However, to ensure local optimality, this case should be considered. Thus, If a ship's *empty* value is zero and its $n - icarry$ is positive, and if it has a neighbor without a cluster, it will just change its carrier

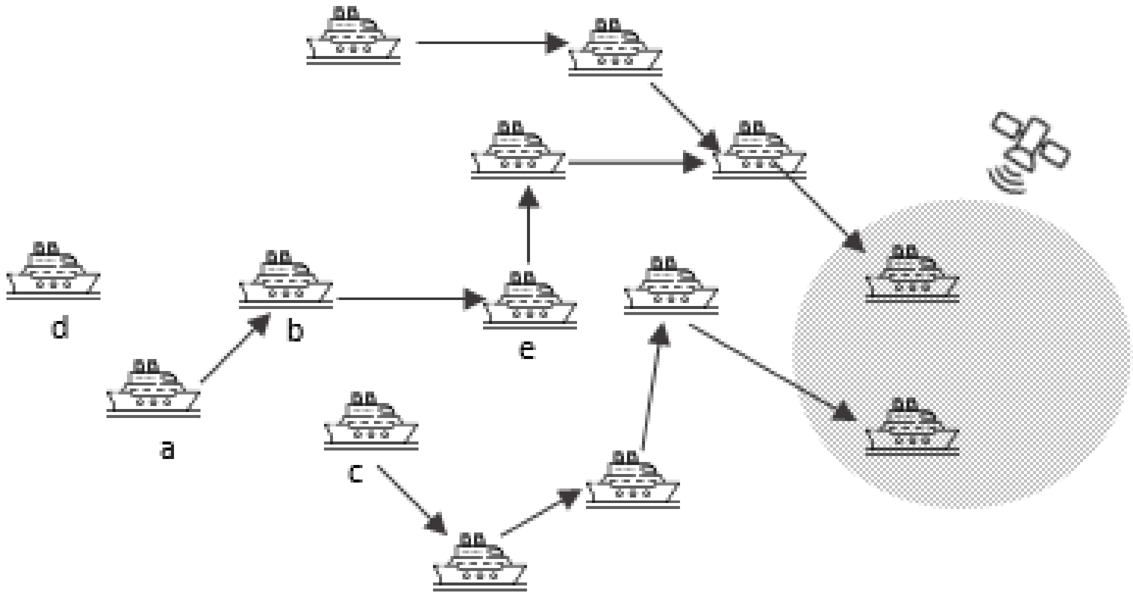


Figure 5.1. A scenario with load balancing, but there is still room for improvement

even if the conditions in load balance are not met (it is enough if the new carrier has enough space to carry its signals and that neighbor). After switching to the new carrier, the new neighbor would be able to join to its cluster. Algorithm 7 illustrates the `Local_Optimality` function.

Here, one may ask why do we need `Load_Balance` function, and why not a ship changes its carrier only if it has a neighbor without cluster? The answer to this question is that if a ship changes its carrier to increase load balancing even if it does not have a neighbor without cluster, it may give its cluster members in front and behind more spaces to carry other signals. The global optimality is still a discussion.

Algorithm 7: `Local_Optimality` Function

Data: Ship s, its neighbors

Result: Ship s changes its carrier

if $\exists\, j \in s.neighbors : j.clusterid = nil$ ***and*** $(s.n - |s.icarry|) > 0$ **then**
  foreach $n \in s.neighbors$ **do**
    if $n.empty > (|s.icarry| + 1)$ **then**
      $wait\_and\_accept\_proposal\_from(n)$;

9.13.4 Priority Function

This function is to make preferences between signals to carry more important ones. Each vessel will have a dynamic priority parameter R ranging from 0-1 according to specific situations:

(i) Emergency e: In case of emergency, $e = 1$, otherwise $e = 0$. A ship may have a priority of 1 if only if it has an active emergency situation. Furthermore, the md condition will not be considered in this case.

(ii) Ship type p: each ship will have default priority $p \in [0, 0.5]$ according to the ship itself (search and rescue vessels, passenger vessels, cargo ships, coast guard ships, etc.).

(iii) Hop distance to the footprint (hd): To prevent drop function from dropping farther ships and protect chain topologies, vessels are given an additional priority value $\alpha \in [0, 0.1]$ according to the hd value.

(iv) Time t: Time passed since it delivered its signal to the satellite last time. This is known by utilizing *access_bit*.

(v) Priorities of the signals it carries.

$$R = max(e, c) \tag{5.2}$$

α value is calculated according to the ratio of hop distance to the maximum hop distance by the following equation.

$$\alpha = 0.1hd \cdot r/md \tag{5.3}$$

In case of emergency, $R = 1$, otherwise $0 \le R = c < 1$. If a ship does not carry any signal and t is not more than threshold value δ, then c is equal to $p + \alpha$.. If $t > \delta$, c is updated according to a logistics growth function G as follows.

$$G(t, \delta) = \frac{1 - e^{-k(t-\delta)}}{1 + be^{-k(t-\delta)}} \tag{5.4}$$

$$c = (p + \alpha) + (1 - (p + \alpha))G(t, \delta) \quad (5.5)$$

G is always in range [0,1). Using such a sigmoid function is reasonable, since it starts with a slow growth just after t exceeds δ, followed by a moderate growth, and then back to a period of slow growth while converging to 1 for high t values. b and k parameters define growth rate of the function. If t and δ values are considered in terms of minutes, we set b and k values to 5 and 0,1 respectively. Typical value for δ is 3 minutes. Growth rate for these values and $p = 0.3$ is shown in Figure 5.2.

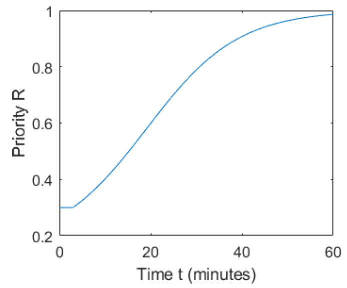


Figure 5.2. Growth function for $p + \alpha = 0.3$, $\delta = 3$ in 60 minutes

Once the priority parameter R is calculated, it is updated if the vessel carries other signals. Let us define R_c as the priority of carrier c that is calculated by equations (5.2) and (5.5). R_c is updated for every acceptor a, according to the acceptor's priority R_a as follows.

$$R_c = R_a + (1 - R_a)R_c \quad (5.6)$$

This update mechanism has some desirable features:

- If any vessel is in active emergency condition (with $e = 1$), all its carriers will have a priority value of 1.
- Priority value is never 1 if there is no emergency condition in any of the carried signals.
- Priority of a carrier is never less than the priority of its acceptor.
- If a carrier has multiple acceptors, the order of applying equation (5.6) for each of the acceptors does not matter.

Figure 5.3 illustrates an example on priority updates when forming or joining a cluster. Furthermore, when a ship updates its priority value due to any reason such as changes in t or e parameters, its carrier also follow the updates accordingly.

9.13.5 Divide Signals

To improve the network, we believe that more ships in footprint should play role to deliver signals to satellite. Until now, for default parameters, the CH in the footprint carries up to 7 signals. But it would be a good practice if these signals were divided between some

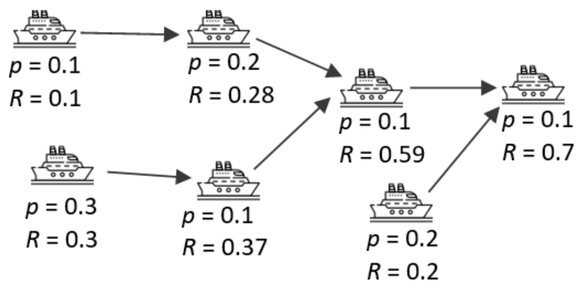


Figure 5.3. Cluster priorities

other ships in the satellite's coverage area that are carrying no signals. The other vessels in the footprint who join the cluster are called *cluster supporters*. The first cluster supporter carries $\lceil |cluster\_carrier.icarry|/2 \rceil$ signals. Each new supporter uses the same formula against the previous supporter.

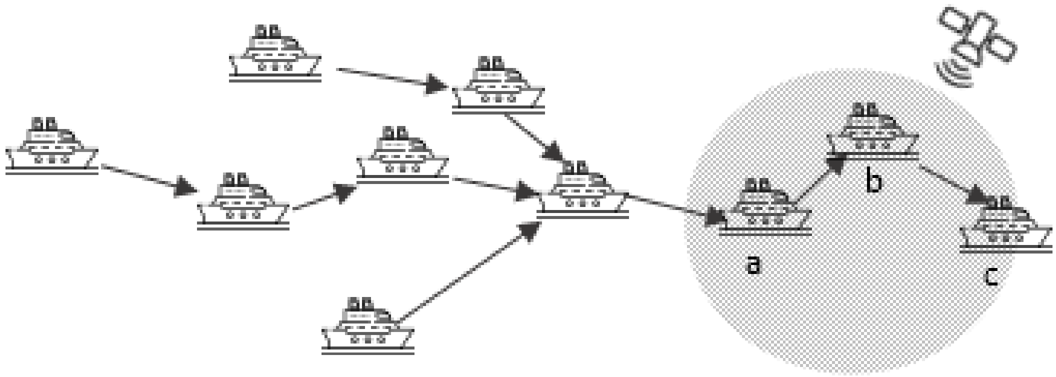


Figure 5.4. Divide Signals

Consider the scenario shown in Figure 5.4. All *a*, *b* and *c* are in the footprint and *a* is carrying 7 signals. In order to do division of labor, other satellites in the footprint decide to carry some signals of *a*. Here *b* will carry 4 from *a*, *c* will carry 2 from *b*. Indeed, *a* is not dropping the 4 signals *b* is carrying but will broadcast them within higher intervals than the other 3 signals. Otherwise, if it does not broadcast them, its acceptor may think that those signals are dropped so they will look for other carriers. Similarly, *b* also broadcasts 2 signals carried by *c* within higher intervals, to inform *a* that those signals are not dropped.

9.13.6 Global Optimality

Although local optimality is achieved by `Load_Balance` and `Local_Optimality` functions, global optimality cannot be guaranteed because vessels are not aware about the situation in other areas and do not have the full picture.

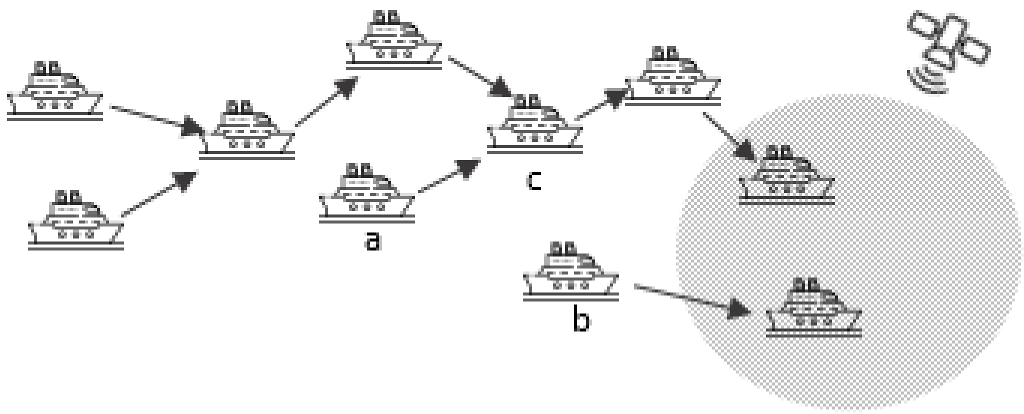


Figure 5.5. Global Optimality Discussion

The example scenario shown in Figure 5.5 is an illustration of why global optimal cannot be guaranteed. In this scenario, *c* can carry 5 signals and *b* can carry 6. According to `Load_Balance` function, *a* will change its carrier to *b* to decrease load on *c*. However, if 6 more ships appear near *b* after some time and they are not in *a*'s range, 2 of them will not be able to join any cluster.

9.14 Numerical Study

9.14.1 Data Collecting

To gather empirical data for the simulation, there were two sources of AIS which are terrestrial and satellite-bases AIS signals. However, we needed both type of data since our experiment extends the coasts and involves all seas surrounding Türkiye which are Black Sea, Mediterranean, Aegean and Marmara Seas. Obtaining historical real terrestrial data was easy but not enough. Thus, the simulation utilized more than fifty mathematical functions with different density parameters to produce AIS signals in the middle of sea in a way to look similar to the real-world map at its most congested time. According to the United Nations Environment Programme, some two thousand vessels of size one hundred metric tons (98.42 gross tons) each are there in the Mediterranean Sea at any given time [35]. In our experiment, 2150 vessels located in the Mediterranean Region, while 850 are in the Black Sea, meaning that three thousand total vessels are used by our study for clustering purposes.

9.14.2 Satellites Orbit

A total of five satellites are used by the simulator to track vessels over the investigated region. The TLE data files are defined in Figur 6.1. The other parameters are defined in Table 6.1.

Table 6.1. Orbital parameters for the satellites used by the simulator

Semi-Major (km)	Inclination (deg)	Eccentricity Value	Mean Ano. (deg)	Mean Mot. (rev/day)	Arg. Peri. (deg)	RAAN (deg)	Altitude (km)
7000	97.98	0.012	264.0374	14.82	130.58	275.77	536-707
6850	97.98	0.012	260.0374	15.31	130.58	286.76	388-555
6960	97.98	0.012	259.0374	14.95	130.58	298.16	497-666
6950	97.98	0.012	259.0374	14.98	130.58	309.16	487-656
6940	97.98	0.012	269.0374	15.01	130.58	319.06	477-646

Considering the region, polar sun-synchronous orbits with inclination $i = 97.9827$ and semi-major axis of $a \in [6850, 7000]$ with a revolution period of approximately $P = 97.72min$ is

```
00001S 97037A   22182.54166667  .00000340  00000-0  33196-4 0  9998
24883  97.9827 275.7718 0121450 130.5814 264.0374 14.95163415974056

00002S 97037A   22182.54166667  .00000340  00000-0  33196-4 0  9998
24883  97.9827 286.7618 0121450 130.5814 260.0374 14.95163415974056

00003S 97037A   22182.54166667  .00000340  00000-0  33196-4 0  9998
24883  97.9827 298.1618 0121450 130.5814 259.0374 14.95163415974056

00004S 97037A   22182.54166667  .00000340  00000-0  33196-4 0  9998
24883  97.9827 309.1618 0121450 130.5814 259.0374 14.95163415974056

00005S 97037A   22182.54166667  .00000340  00000-0  33196-4 0  9998
24883  97.9827 319.0618 0121450 130.5814 269.0374 14.95163415974056
```

Figure 6.1. TLE data for the satellites used by the simulator

by far accepted to handle AIS data from this area. The first satellite passes over the investigated region at Tue Mar 2022 02:06:40 GMT+3:00 when the simulation is set to run. The first five periods are shown in Figure 6.2.

9.14.3 Results

The simulation was done using the following parameters:

- 3000 ships: 2150 in the Mediterranean, 850 in Black Sea.
- Five satellites crossing the investigated area at the simulation time with different orbital elements.
- Seven parameters sets for different *r*, *md*, and *f* values.

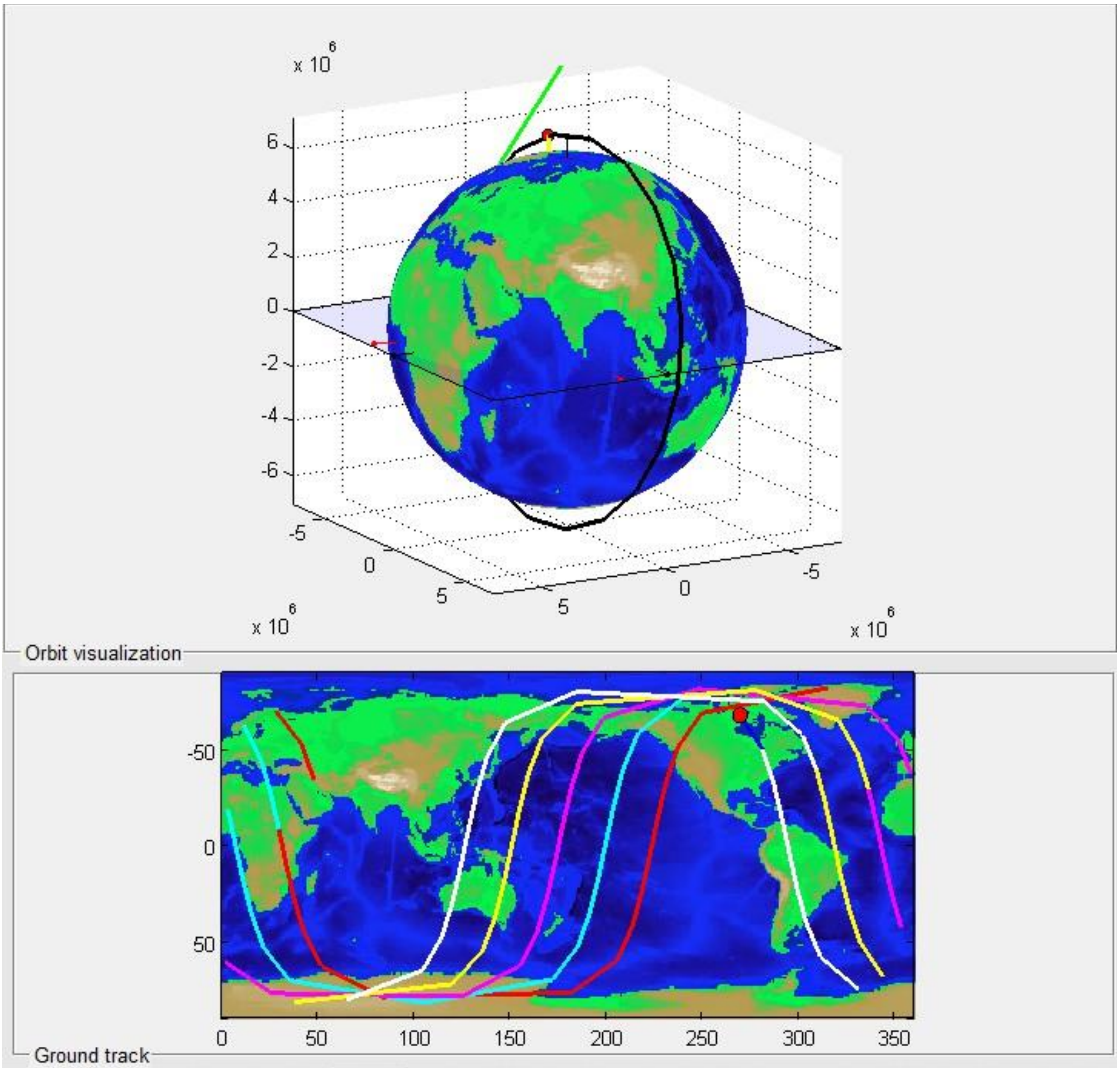


Figure 6.2. Orbit visualization for first satellite in five periods

9.14.3.1 Results without clustering

Figure 6.3 illustrates all the vessels situation in the studied region considering default parameters $r = 40km$, $md = 160km$, and $f = 1$, after the 5 satellites passes the area (fifth satellite has passed the easternmost region some time ago, so it is not seen in the figure). The vessels marked by dark blue delivered their signals, while the vessels that could not deliver their signals are represented by dark red color. Before applying the presented clustering algorithm, we have got 2165 ships delivered their signals and 835 other ones which were not in the footprint area could not.

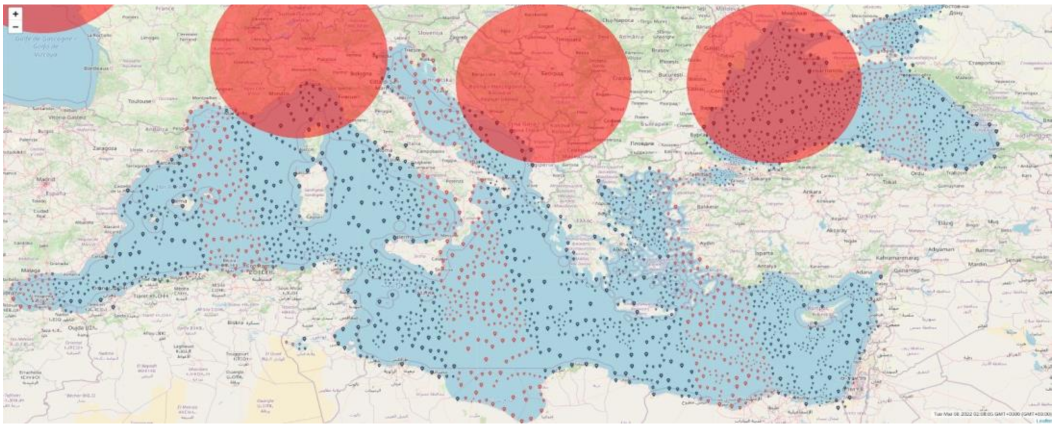


Figure 6.3. Results without clustering

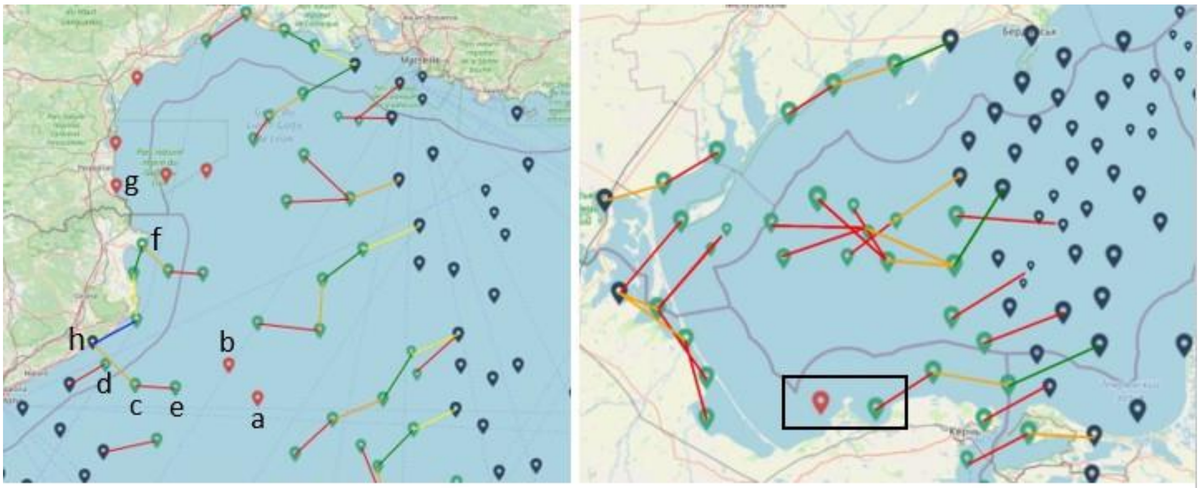


Figure 6.4. Cluster Examples for plain DVCA

9.14.3.2 Results for plain DVCA

After running DVCA without optimization techniques, we have 650 (77.84%) more ships managed to deliver their signals thanks to the cooperative vessels. In Figure 6.4 we present some examples for clusters. Vessels marked by green color delivered their signals by joining or forming a cluster. Members of same cluster are shown by connecting lines between vessels. We can obviously see some of the failing scenarios in both sides of Figure 6.4. For example, if we look at the rectangle in the right side, we can see a ship without a cluster (marked by red) although it has another ship in range which is in green. There were two satellites passed the west and the east of the area, which can be noted by dark blue

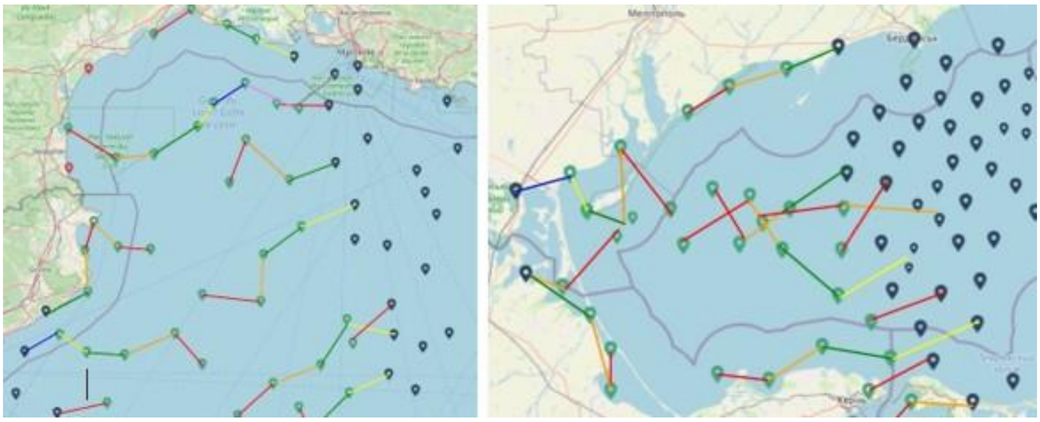


Figure 6.5. Load Balance and Tracking Data Results

vessels. The green ship is 59 km away from the nearest footprint (east) and can carry 5 signals. While the red one is 58 km away from the nearest satellite (west). Thus, the ship in green did not send proposal to it because it is closer to another satellite.

Another failing scenario is in the left side. It is better for ship *c* to change its carrier from *h* to *d* to provide some empty space for *e* to carry *a* and *b*. Not to mention that by this operation, the load on *h* will be decreased by 2 allowing *f* to carry 2 more signals from the north where we have 3 vessels without cluster when *g* becomes in *f*'s range (it is not yet).

There are 3 possible reasons why *c*'s current carrier is *h*, not *d*:

- Ship *d* appeared after the cluster is set for *c*.
- *c* did not accept proposal from *d* because it has a neighbor in footprint with empty space.
- *d* did not send proposal to *c* due to `Should_Carry` function.

However, the actual reason was the third one.

9.14.3.3 Optimization Results

With tracking data and improved distribution value using `Load_Balance` function, number of ships delivered their signal using the proposed clustering algorithm is increased to 704 (84%) ships. Now we can have a look at Figure 6.5 to see the improvements. We can notice that 6 more signals are delivered from both sides.

As mentioned before, tracking data and load balance functions do not guarantee local

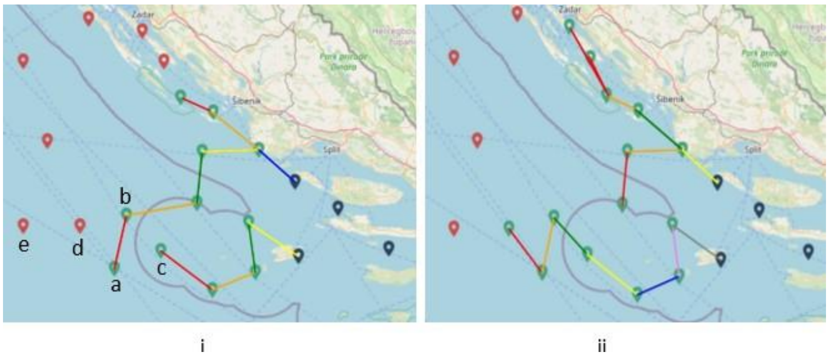


Figure 6.6. (i) A failing scenario for load balance; (ii) Improvement by local optimization optimality. Figure

6.6.i illustrates one real example. If *a* changes its carrier to *c*, then both *e*

and *d* will be delivered. If *b* change its carrier to *c*, then only *d* will be carried. It really depends on who gets the proposal and acts first. Let's remember that in the standard load balance function without adding the local optimality condition, nothing is going to happen, as illustrated before in Figure 5.1.

In Figure6.6.ii with local optimality condition considered, *b* changed its carrier to *c* allowing *a* to carry *d*. Indeed, this provided 2 empty spaces for *b*'s previous cluster carrier. Thus, total 3 more signals are delivered. Total number of ships delivered their AIS signals to the satellite using all optimization techniques is increased to 720 (86.22%). There remain 115 ships without cluster and 13 of them are away from the nearest satellite by a distance more than

md = 160km. Thus, they by default cannot create or join a cluster.

9.14.3.4 Dividing signals

Having more ships involved in the clustering algorithm increases the efficiency of the network and prove the term of cooperative vessels. Now, we have 193 cluster heads in the footprint (cluster carriers) carrying signals of vessels that are not in the satellite's coverage area. However, this can be increased to 373 vessels utilizing dividing signal function. Figure 6.7 illustrates how new ships in footprint are involved in sharing signals with cluster carrier.

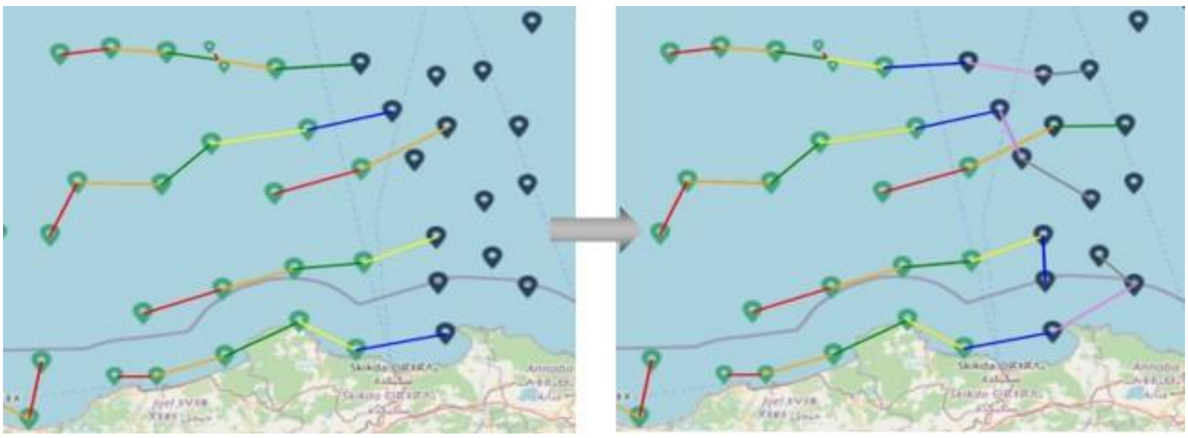


Figure 6.7. Dividing Signals Results

9.14.3.5 Comparison Table

If n_c is the number of vessels delivered their AIS signals by clustering and D is the distribution value for the investigated area, then the simulation results for seven different parameter sets are presented in Table 6.2.

The impact of optimization techniques is outstanding improving the results by at least 9%. Furthermore, the load balance function is proven to play an important role in decreasing the distribution value by around 12% in most cases. Comparing the first set with the last one (where n values for cluster carriers are the same), increasing md by 80km added no more than 36 members to the clusters. Therefore, increasing f instead of md was a better choice since our map is crowded. Comparison of sets 1 and 3 is another confirm for this fact. The clustering algorithm is more influenced by r as expected. Yet, increasing f with low AIS ranges is desirable as can be noted comparing sets 5 and 6. When r is small, a vessel has fewer alternatives of carriers to choose between. This restricts the power of load balance function which could only decrease the distribution value by 8.2% in set 4. The number of vessels carrying full capacity (n) for the first set is 59 and no vessel in the footprint is full. While for set 4, there are 66 vessels with full capacity, 17 of them are in the footprint. This means that the carried signals along all clusters could not be shared among more cluster carriers.

Set No.	Parameter Sets	# Ships with $0 < d \leq md$	Standard Clustering	Load Balance	Tracking Data	Local Opt.	Max. d : $clusterid = nil$
1	$md = 200$km $r = 40$km, $f = 1$	833	n_c=673, D=2.08	n_c=709, D=1.80	n_c=730, D=1.95	n_c=738, D=1.97	171km
2	$md = 160$km $r = 40$km, $f = 1$	822	n_c=650, D=1.97	n_c=692, D=1.77	n_c=704, D=1.81	n_c=720, D=1.84	156km
3	$md = 160$km $r = 40$km, $f = 2$	822	n_c=671, D=2.15	n_c=705, D=1.87	n_c=725, D=1.99	n_c=732, D=2.00	156km
4	$md = 160$km $r = 32$km, $f = 1$	822	n_c=485, D=2.07	n_c=497, D=1.90	n_c=519, D=1.95	n_c=529, D=1.93	142km
5	$md = 144$km $r = 36$km, $f = 1$	803	n_c=593, D=2.03	n_c=616, D=1.81	n_c=631, D=1.84	n_c=648, D=1.88	143km
6	$md = 144$km $r = 36$km, $f = 2$	803	n_c=612, D=2.22	n_c=625, D=1.93	n_c=656, D=2.06	n_c=668, D=2.07	143km
7	$md = 120$km $r = 40$km, $f = 3$	758	n_c=637, D=1.88	n_c=673, D=1.65	n_c=684, D=1.70	n_c=702, D=1.74	120km

Table 6.2. Comparison Table

9.15 Conclusion and Future Work

The current maritime network is based on AIS signals and ships are responsible to broadcast their own signals to other ships and satellites as well independently. However, since maritime transport is getting wider and becoming essential in many industries, the network has to be improved and innovative solutions are required to deal with the increasing number of vessels in the water. The presented distributed vessel clustering algorithm (DVCA) proves the importance of cooperative vessels so they can deliver their position report in shorter time intervals offering a conclusive answer to the question of "What should be considered in the next generation of AIS devices?".

When vessels coordinates and work in clusters, the network becomes more dependable and safer than when each vessel works on its own. The clustering algorithm in standard parameters increased number of delivered signals for ships which are not in the satellite's coverage area by 77.84%. The optimization techniques provided better results by making the network more distributed and efficient increasing the percentage to 86.22%. We believe that the presented approaches are easily deployable, since they do not require any core update in the infrastructure, and rely in software updates and definition of new AIS message type for cluster-related data. It is important to mention that the only parameters used in the clustering were the distance between vessels, AIS range, and the nearest satellite but yet the findings that we have presented in this research are precursor to more advanced work and it would be fruitful to pursue further research in this area. Clustering based on certain characteristics and parameters like AIS classes, direction, and speed will ensure stability in the network which is an important term in clustering. The mobility of the nodes and the changing topology should be considered in future works. Furthermore, serving multiple clusters and sharing carried signals with more than one ship/cluster considering load balance function will decrease the distribution value and make the network more reliable. Giving incentives to the ships to carry other's signals is another issue. To cover these gaps, we intend to present a game theoretical approach for clustering in maritime network in future work. Moreover, we believe that some presented functions like `May_Carry`

and `Accept_Carrier` can be enhanced by machine learning models. It is expected that this no vel study will have pioneering effect for many valuable researches in maritime network.

9.16 REFERENCES

[1] Tu Ho et al. "Internet of Things at Sea: Using AIS and VHF over Satellite in Remote Areas". In: Apr. 2018.

[2] Maurantonio Caprolu et al. "Vessels Cybersecurity: Issues, Challenges, and the Road Ahead". In: *IEEE Communications Magazine* 58 (June 2020), pp. 90–96. DOI: 10 . 1109/MCOM.001.1900632.

[3] Nathan Greig et al. “Using Satellite AIS to Analyze Vessel Speeds Off the Coast of Washington State, U.S., as a Risk Analysis for Cetacean-Vessel Collisions”. In: *Frontiers in Marine Science* 7 (Feb. 2020), p. 109. DOI: 10.3389/fmars.2020.00109.

[4] Pan Sheng and Jingbo Yin. “Extracting Shipping Route Patterns by Trajectory Clustering Model Based on Automatic Identification System Data”. In: *Sustainability* 10 (July 2018), p. 2327. DOI: 10.3390/su10072327.

[5] Wang Yitao, Yang Lei, and Song Xin. “Route Mining from Satellite-AIS Data Using Density-based Clustering Algorithm”. In: *Journal of Physics: Conference Series* 1616 (Aug. 2020), p. 012017. DOI: 10.1088/1742-6596/1616/1/012017.

[6] Yang Zhou et al. “Ship classification based on ship behavior clustering from AIS data”. In: *Ocean Engineering* 175C (Mar. 2019), pp. 176–187. DOI: 10.1016/j.oceaneng.2019.02.005.

[7] Selo Sulistyo, Sahirul Alam, and Ronald Adrian. “Coalitional Game Theoretical Approach for VANET Clustering to Improve SNR”. In: *Journal of Computer Networks and Communications* 2019 (July 2019), pp. 1–13. DOI: 10.1155/2019/4573619.

[8] Samo Vodopivec, Janez Bester, and Andrej Kos. “A Multihoming Clustering Algorithm for Vehicular Ad Hoc Networks”. In: *International Journal of Distributed Sensor Networks* 2014 (Mar. 2014), pp. 1–8. DOI: 10.1155/2014/107085.

[9] Yuzhong Chen et al. “Distributed multi-hop clustering algorithm for VANETs based on neighborhood follow”. In: *EURASIP Journal on Wireless Communications and Networking* 2015 (Dec. 2015). DOI: 10.1186/s13638-015-0327-0.

[10] Craig Cooper et al. “A Comparative Survey of VANET Clustering Techniques”. In: *IEEE Communications Surveys and Tutorials* PP (Sept. 2016), pp. 1–1. DOI: 10.1109/COMST.2016.2611524.

[11] Prithwish Basu, Naved Khan, and Thomas Little. “A Mobility Based Metric for Clustering in Mobile Ad Hoc Networks”. In: May 2001, pp. 413–418. ISBN: 0-7695-1080-9. DOI: 10.1109/CDCS.2001.918738.

[12] Alexandra (Adam) et al. “The Importance of Maritime Transport for Economic Growth in

the European Union: A Panel Data Analysis". In: *Sustainability* 13 (July 2021), p. 7961. DOI: 10.3390/su13147961.

[13] Michael Baldauf and Gianiti Claresta. "Vessel Traffic Services (VTS) and e-Navigation to safely and efficiently connect Regions". In: 6 (Nov. 2020), pp. 29–38. DOI: 10.34647/jmv.nr6.id46.

[14] imo.org. *AIS Transponders*. Retrieved 22 June 2022. URL: https://www.imo.org/en/OurWork/Safety/Pages/AIS.aspx.

[15] Dong Yang et al. "How big data enriches maritime research – a critical review of Automatic Identification System (AIS) data applications". In: *Transport Reviews* 39 (July 2019), pp. 1–19. DOI: 10.1080/01441647.2019.1649315.

[16] Martin Redoutey et al. "Efficient Vessel Tracking with Accuracy Guarantees". In: Dec. 2008, pp. 140–151. ISBN: 978-3-540-89902-0. DOI: 10.1007/978-3-540-89903-7_13.

[17] Donna Kocak and Peggy Browning. "Real-time AIS tracking from space expands opportunities for global ocean observing and maritime domain awareness". In: Oct. 2015, pp. 1–7. DOI: 10.23919/OCEANS.2015.7404635.

[18] Rec ITU. "M. 1371-5-Technical characteristics for an automatic identification system using time-division multiple access in the VHF maritime mobile band". In: *International Telecommunications Union* (2014).

[19] Sushma Shankarappa and V. Rao. "Wire monopole antenna for low earth orbit satellite applications". In: (Dec. 2017), pp. 70–74. DOI: 10.1109/CCUBE.2017.8394160.

[20] Emir Husni. "Design of Automatic Identification System (AIS) Receiver for Low Earth Orbit (LEO) Satellite". In: *International Review on Modelling and Simulations (IREMOS)* 9 (Dec. 2016), p. 435. DOI: 10.15866/iremos.v9i6.10958.

[21] Roman Wawrzaszek et al. "Detection and Decoding of AIS Navigation Messages by a Low Earth Orbit Satellite". In: Jan. 2019, pp. 45–62. ISBN: 978-3-319-94516-3. DOI: 10.1007/978-3-319-94517-0_4.

[22] Mélanie Fournier et al. "Past, present, and future of the satellite-based automatic identification system: areas of applications (2004–2016)". In: *WMU Journal of Maritime Affairs* 17 (Sept. 2018). DOI: 10.1007/s13437-018-0151-6.

[23] Davor Šakan et al. "Near Real-time S-AIS: Recent Developments and Implementation Possibilities for Global Maritime Stakeholders". In: *Pomorstvo* 32 (Dec. 2018),

pp. 211–218. DOI: 10.31217/p.32.2.6.

[24] Zuzanna Kosowska-Stamirowska, César Ducruet, and Nishant Rai. "Evolving structure of the maritime trade network: evidence from the Lloyd's Shipping Index (1890–2000)". In: *Journal of Shipping and Trade* 1 (Sept. 2016). DOI: 10 . 1186 / s41072 - 016 - 0013-3.

[25] Fabio Mazzarella et al. "A Novel Anomaly Detection Approach to Identify Intentional AIS On-Off Switching". In: *Expert Systems with Applications* 78 (Feb. 2017). DOI: 10. 1016/j.eswa.2017.02.011.

[26] JOHN BASQUILL at Global Trade Review. *Dark activity: As sanctions pressure rises, maritime trade turns to tech*. Retrieved 22 June 2022. 2020. URL: https :// www. gtreview . com / magazine / volume - 18 - issue - 3 / dark - activity - sanctions-pressure-rises-maritime-trade-turns-tech/.

[27] Mohammed Chessab Mahdi. "TIGRISAT Orbital MotionSimulation and Analysis". In: *Journal of Control Engineering and Technology* 5 (Jan. 2015), pp. 1–8. DOI: 10 . 14511/jcet.issue.050101.

[28] Dirk. Brouwer. "SOLUTION OF THE PROBLEM OF ARTIFICIAL SATELLITE THEORY WITHOUT DRAG". In: *The Astronomical Journal* 64 (1959), p. 378.

[29] Nicholas Crisp, Katharine Smith, and Peter Hollingsworth. "Launch and Deployment of Distributed Small Satellite Systems". In: *Acta Astronautica* 29 (Apr. 2015). DOI: 10 . 1016/j.actaastro.2015.04.015.

[30] M. M. Martinovic and S. D. Segan. "SATELLITE ORBIT UNDER INFLUENCE OF A DRAG–ANALYTICAL APPROACH". English. In: *Serbian Astronomical Journal* (July 2017). Report, pp. 53+. ISSN: 1450698X. URL: https://link.gale.com/apps/doc/A529864240/AONE?u=anon~44133411&sid=googleScholar&xid= 242c6689.

[31] Larisa Trichtchenko et al. "Highly Elliptical Orbits for Arctic observations: Assessment of ionizing radiation". In: *Advances in Space Research* 54 (Dec. 2014). DOI: 10.1016/ j.asr.2014.09.012.

[32] Ram Krishan Sharma. "Contraction of high eccentricity satellite orbits using KS elements in an oblate atmosphere". In: *Advances in Space Research* 23.4 (1999). Satellite Dynamics, Orbit Analysis and Combination of Space Techniques, pp. 693–698. ISSN: 0273-1177. DOI: https:// doi. org/ 10 . 1016 / S0273 - 1177(99)00139- 1. URL: https:// www. sciencedirect. com/ science/ article/ pii/ S0273117799001398.

[33] Shraddha Gupta. "Effect of Altitude, Right Ascension of Ascending Node and Inclination on Lifetime of Circular Lunar Orbits". In: *International Journal of Astronomy and Astrophysics* 1 (Jan. 2011). DOI: 10.4236/ijaa.2011.13020.

[34] Prasetya, Dwi Arman and Nguyen, Phong Thanh and Faizullin, Rinat and Iswanto, Iswanto and Armay, Edmond Febrinicko. "Resolving the shortest path problem using the haversine algorithm". eng. In: Journal of Critical Reviews 7.1 (2020), 62–64. ISSN: 2394-5125. DOI: {10.22159/jcr.07.01.11}. URL: %7Bhttp://dx.doi. org/10.22159/jcr.07.01.11%7D.

[35] SARAH EVERTS. "The Mediterranean: Beneath The Surface". In: *Chemical & Engineering News Archive* 90 (Mar. 2013), pp. 12–17. DOI: 10.1021/cen-09015- cover.

10 Satellite System Development 2023 and TT&C Testrig 2023-Apr 24

Satellite System Development

Anadolu-Sham COMSAT

Communication Platforms & Satellite Communication Systems

Written By:

Raja Mourad

Ahmad Awwad

Ahmad Dannawi

Hana Mourad

Khalil Shateh

Riyad Mourad

Asmaa Dhaybi

Bachir Merehbi

Last updated : 1.4.24

Table of Content

10.1 Introduction

This document aims to explain the development process of the satellite system designed by AS-COMSAT.

10.2 On-Board System Design and Realization

10.2.1 Introduction

This chapter is intended to introduce the design procedure of the onboard system. The onboard system is composed of all necessary units for the satellite to function, which will be discussed in details.

10.2.2 Overview

The onboard system consists of several units to ensure a stable operation. Figure 2-1 below shows all the units and how they interact with each other. The main components are:

- **Attitude Control System**: This controller board establishes the stability of the satellite. It is responsible for navigating (through its onboard propulsion components) the satellite with respect to user commands (received from the ground station) and sun sensors equipped on-board to track the sun's position.
- **Charge Control/Power Supply Unit**: This unit is responsible for feeding the satellite units with their suitable power. It also manages the battery charging process via the solar panels.
- **Telemetry, Tracking, and Commanding (TT&C) Transceiver unit**: This unit sends to the ground station all the telemetry data measured from the onboard sensors. It is also responsible for receiving the command setpoints from the ground station.
- **On-board computer (OBC)**: This is the heart of the satellite system. It serves as a complete on-board data sharing server between all the units. All the units will be connected to the OBC via serial bus (USB bus) and data would be shared between all the units.
- **Payload Data Transceiver:** The Payload transceiver is responsible for user data exchange between the ground station and the satellite such as images, text, video streaming, etc…
- **Inter-satellite transceiver:** This unit is responsible for transferring data in between the satellites located in the LEO. The objective is to cover the widest area possible to transmit messages.

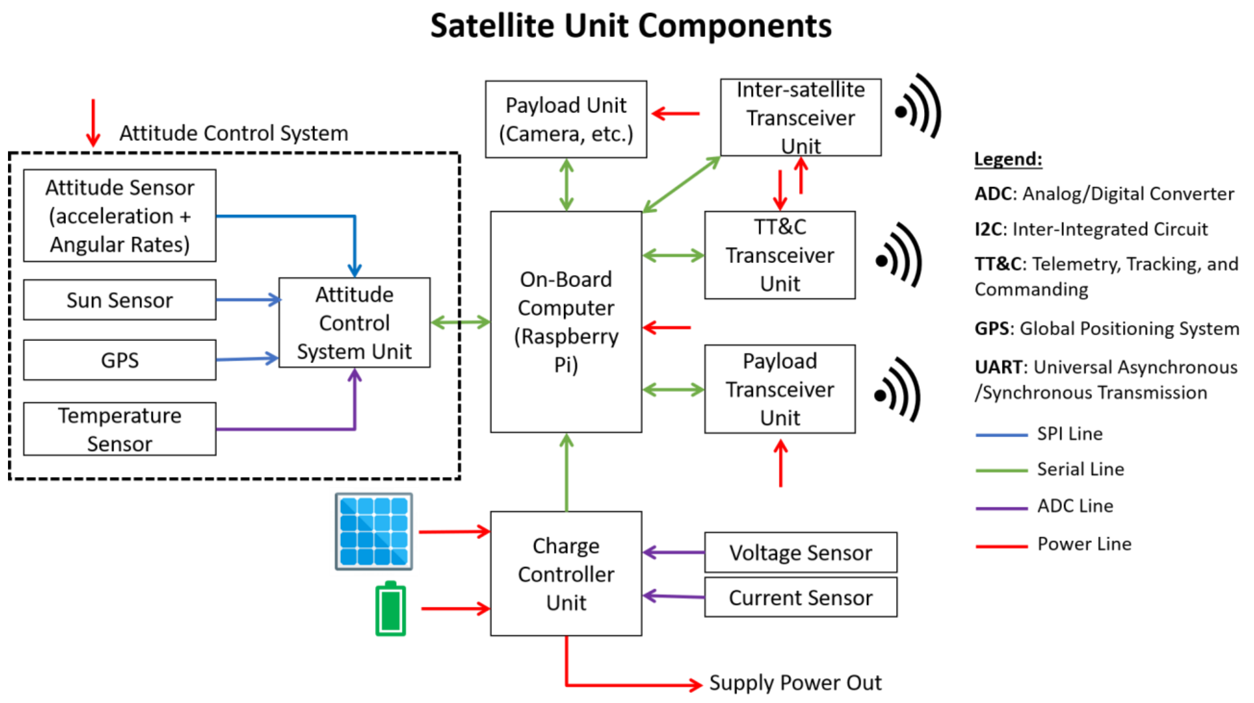


Figure 2-1 Satellite Unit Components

10.2.3 Bill of Material

To assemble a single unit of CubeSat, the components listed in Figure 2-2 below are needed. N.B: The last column "Implemented In" shows in which CubeSat version the related component was used (for now, there are 2 CubeSat versions).

Component	Type	Model/Rating	Unit price ($)	Quantity	Total	Comments	Implemented In
Aluminum Rails	Mechanical	2020 (20 mm x 20 mm)	5	3	15	price is per meter	V1, V2
Aluminum Plate	Mechanical	25cm x 25cm	1.666	3	4.998		V1, V2
Aluminum corners	Mechanical	2020 (20 mm x 20 mm)	1.5	24	36		V1, V2
Bolts + nuts	Mechanical		2.05	1	2.05	per 64 bolts and 64 nuts	V1, V2
Pressure gauge	Mechanical	1/2"/4 bar	6.5	1	6.5		V1
Pex tubes	Mechanical	1/2"	0.65	1	0.65	price is per meter	V1
Manual valve	Mechanical	1/2"	8	1	8		V1
T-connector	Mechanical	1/2"	2	3	6		V1
Pex Corner connector	Mechanical	1/2"	2	4	8		V1
Pex connector	Mechanical	1/2"/male	1	1	1		V1
Teflon	Mechanical		0.35	1	0.35		V1
Accessories	Mechanical		6	1	6	Includes connectors between pips, extensions, etc.	V1
Solar Panel	Electrical	0.5A-6W	7.5	8	60		V1, V2
Valves	Electrical	DCF-HS11/220V	5	4	20		V1,V2
Relay box	Electrical	Songle/12VDC	1	3	3		V1
Wires	Electrical		2	1	2		V1,V2
Terminal Junction	Electrical		3	1	3		V1
Battery	Electrical	18650/3.7V per cell	6	2	12		V1,V2
Battery holder	Electrical		2	2	4		V1,V2
Antenna	Communication	Patch antenna	5.67	1	5.67		V1,V2
Raspberry Pi	OBC	Pi 3	40	1	40		V2
Charge Controller Unit	Electrical	Refer to technical doc.	32	1	32		V2
TT&C Transceiver Unit	Communication	Refer to technical doc.	25	1	25		V2
ACS Unit	Electrical	Refer to technical doc.	30	1	30	Missing: GPS receiver	V2

Figure 2-2 Bill of material for different satellite assembly versions

10.2.4 Mechanical Realization

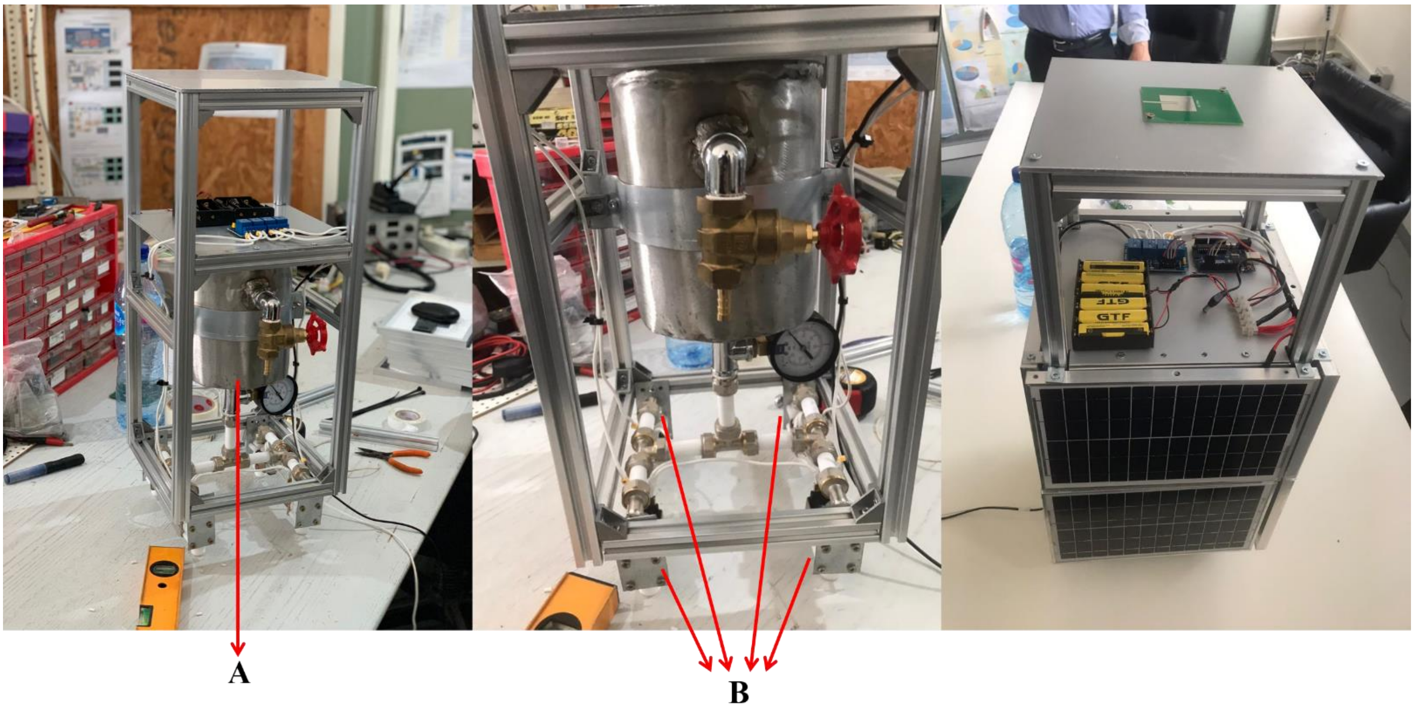


Figure 2-3 Mechanical realization of satellite system V1(cold gas propulsion unit)
)

- A: Pressurized gas container: used to store a high-pressure gas such as nitrogen to generate the propulsion for the Attitude Control System (ACS).
- B: Solenoid valves used to distribute the pressure and generate thrust for the CubeSat.

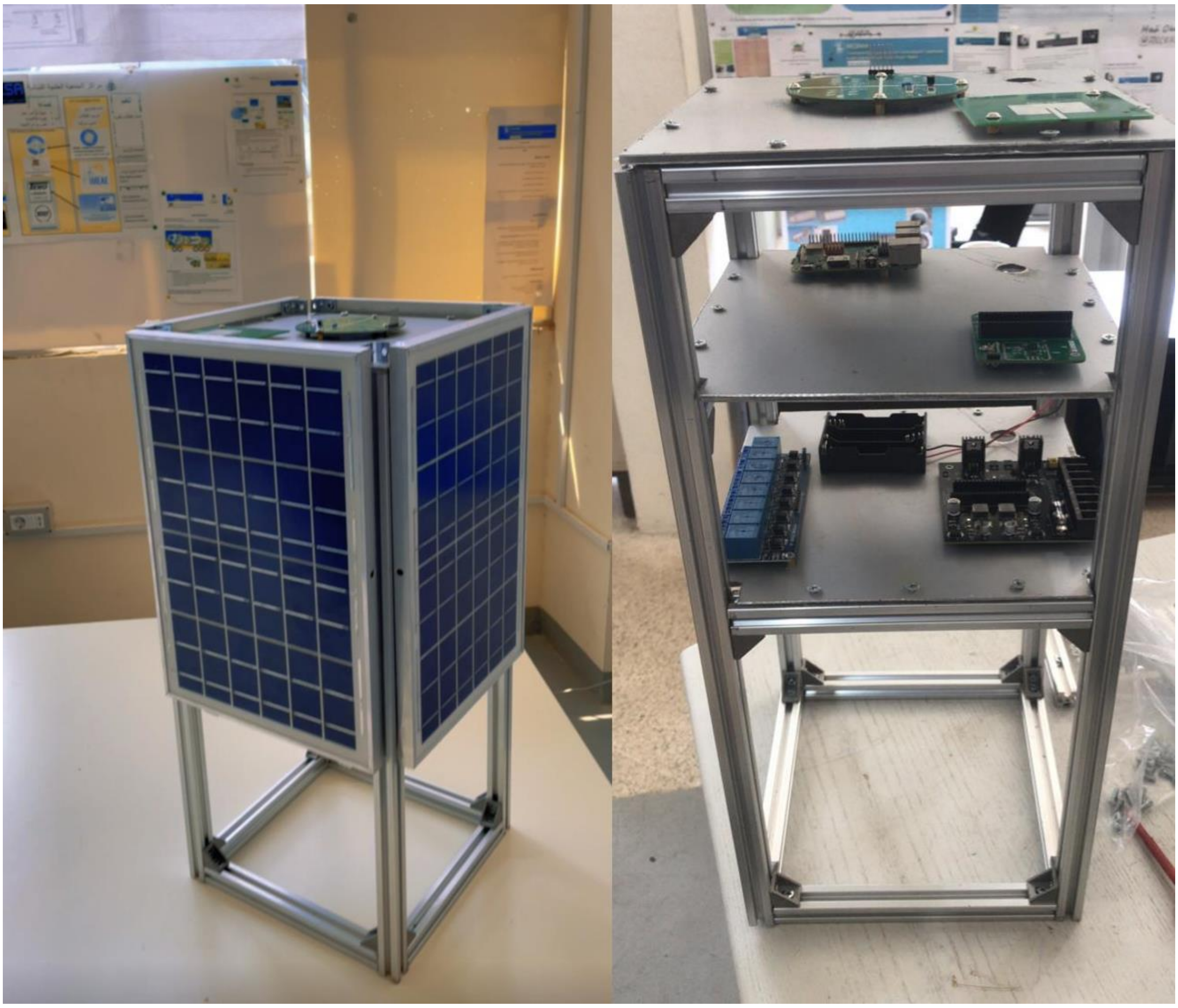


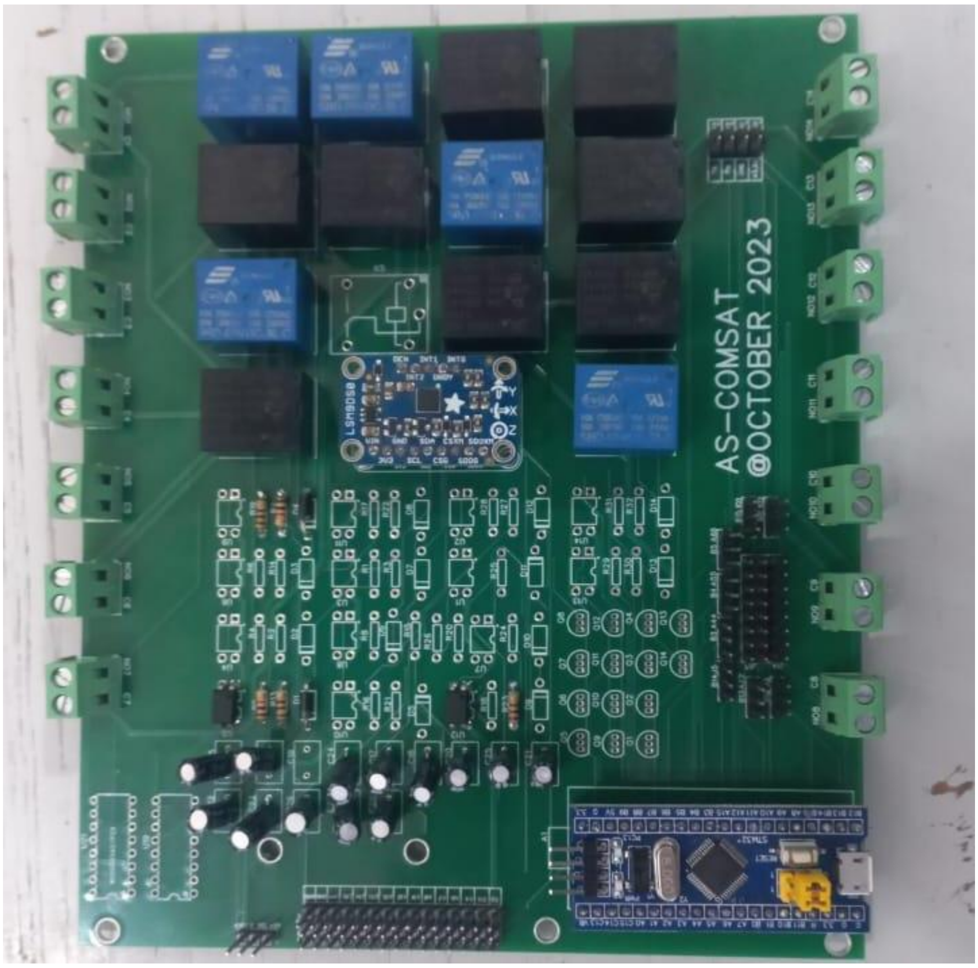
Figure 2-4 Mechanical realization of satellite system V2 (orbit change Propulsion unit)

In V2 system, the propulsion unit is built outside the CubeSat and is controlled through the Attitude Controller.

10.2.5 Attitude Control System (ACS)

10.2.5.1 Introduction

The attitude control system is responsible for receiving the high-level navigation commands and stabilize the satellite at a specific orientation.



The schematic/PCB files are attached below:

drive-download-20231228T084755Z-001.zip

10.2.5.2 Propulsion System

Pressurized Gas (V1)

The satellite navigates in the LEO level by having a tank filled and pressurized with cold gas (In our case, it was 6 bars of Helium gas). The cold gas container is attached to 4 valves to ensure proper thrusting and thus, we can obtain a 3-DOF translation in space.

Fuel Burner (V2)

See document done by Ahmad Dannawi and Abdulla Qassem

(https://www.aecenar.com/index.php/companies/as-comsat/ics-iap-sat/as-comsat-1-attitude-control-system-testrig)

10.2.5.3 Inertial Measurement Unit (IMU)

Communication protocol used between accelerometer & stm32:

- The communication between an accelerometer and STM32 microcontroller on an ACS board typically involves the use of a I2C communication protocol.
- This protocol allows the accelerometer to send data to the STM32 microcontroller, which can then process and use the data as needed.
- I2C is a serial communication protocol only uses two wires to transmit data between devices. so, data is transferred bit by bit along a single wire (the SDA line).
- Data is transferred in messages. Messages are broken up into frames of data. Each message has an address frame that contains the binary address of the slave. One address frame is 7 bits**.**
- Now we shall program the accelerometer & gyro-meter using IDE programming and sending data to STM32 using I2C by SDA with address 0F (depending on data sheet of LSM9DS0). And check if there is connection between accelerometer and microcontroller.
- The microcontroller shall response by 49 (HEX) for accelerometer & D4 for Gyro meter, if no response so no connection between the 2 devices. (based on datasheet of accelerometer LSM9DS0)
- After the connection between STM32 & Accelerometer is approved, now we shall implement some code to test the accelerometer & Gyro meter along 3-axis (X, Y, Z)
- The results shall be in form of Hexadecimal, we put for each axis 2 bytes (16 bites)
- Therefore, when we change the location of the sensor along any axis, the results will change and the test will be complete.

Add the sensor communication flowchart

10.2.5.4 Sun Sensor

Sun sensors are navigation instruments that are used in space to establish the direction and position of the sun in relation with the satellite or spacecraft. These devices are used to determine attitude in space, providing data about the orientation of the satellite in relation with the solar vector, which is the angle at which the sun rays reach the satellite or spacecraft.

The sun sensor board consist of 4 identical stages of Figure 2-5 separated by 4.5 cm between each stage.

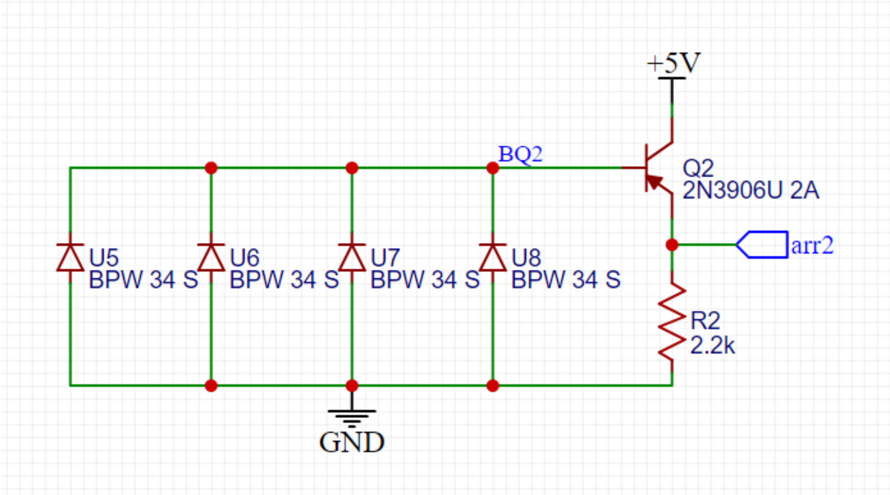


Figure 2-5 Sun sensor single array schematic

The following BOQ is used when ordering the sun sensors.

ID	Name	Designator	Footprint	Quantity	Manufacturer Part	Manufacturer	Supplier	Supplier Part	Price	Pins
1	NC	R5,R6,R7,R8	R_AXIAL-0.4	4						2
2	2.2k	R1,R2,R3,R4	R_AXIAL-0.5	4						2
3	2N3906-AP	Q1,Q2,Q3,Q4	TO-92-3_L4.8-W3.7-P2.54-L	4	2N3906-AP	MCC(美微科)	LCSC	C2987037	0.035	3
4	BPW 34 S	U1,U2,U3,U4,U5,U6,U7,U8,U9,U10,U11,U12,U13,U14,U15,U16	SENSOR-TH_BPW34	16	BPW 34 S	OSRAM	LCSC	C2900590	0.819	2
5	HDR-M-2.54_1x6	J1	HDR-M-2.54_1X6	1			LCSC	C190819	0.034	6

Figure 2-6 Sun sensor BOQ

BOM_PCB_Sun Sensor_2023-07-18.c

The heart of the sun sensor is the BPW34 photodiode. It acts as a dependent current source when exposed to light (the higher the light is, the higher the current is). From its datasheet, the variation of its reverse current with respect to its exposed light is shown in Figure 2-7. It is noticed that the relationship between both quantities is almost linear. Hence, the relation between both quantities can be expressed as:

$$I_{ra} = 0.08\frac{\mu A}{lx}\,.\,sun\ exposure\ (lx) \quad 2\text{-}1)$$

When connecting 4 parallel photo-diodes, the equation becomes:

$$I_{ra} = 0.32\frac{\mu A}{lx}\,.\,sun\ exposure\ (lx) \quad 2\text{-}2)$$

Hence, the limits of the generated current via the 4 parallel diodes would become:

$$I_{ra} = \{3.2\ \mu A \quad if\ sun\ exposure = 10^1\ lx\ 3200\ \mu A\ \ if\ sun\ exposure = 10^4\ lx \quad 2\text{-}3)$$

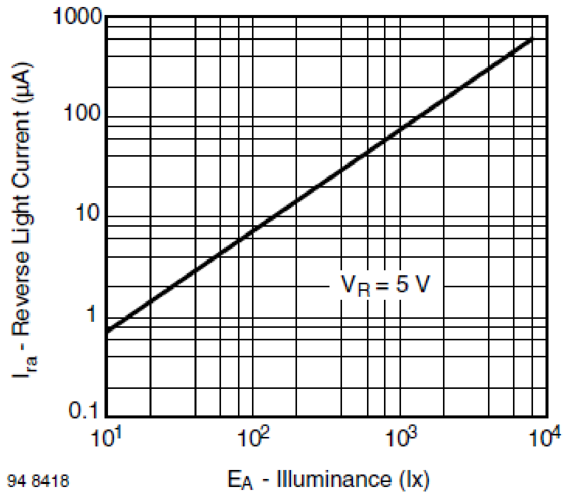


Figure 2-7 Sun sensor reverse current vs. exposed light illuminance

The use of resistor in emitter part of transistor: For Stability and Biasing: to act as amplifier when the emitter junction is forward biased and the collector junction is reverse biased, the transistor is in the active region The resistor limits the current flowing through the emitter, protecting the transistor from excessive current that could damage it.

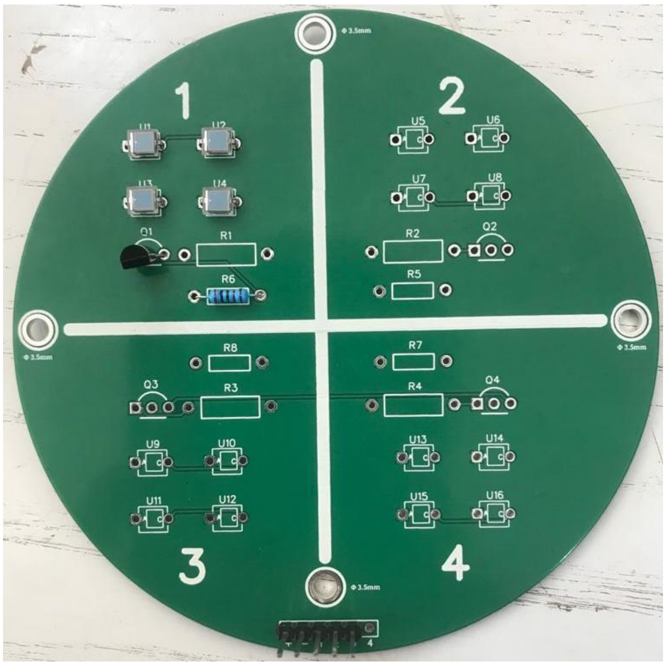
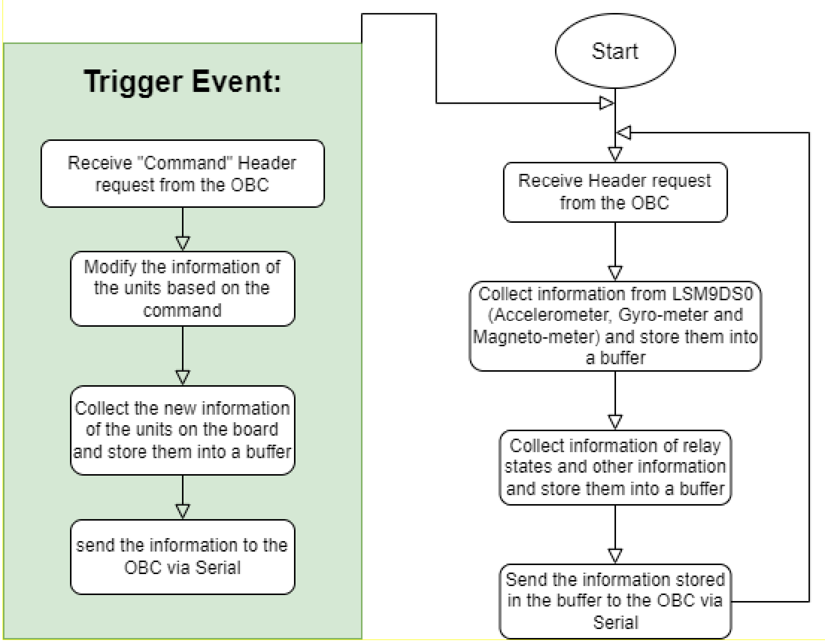
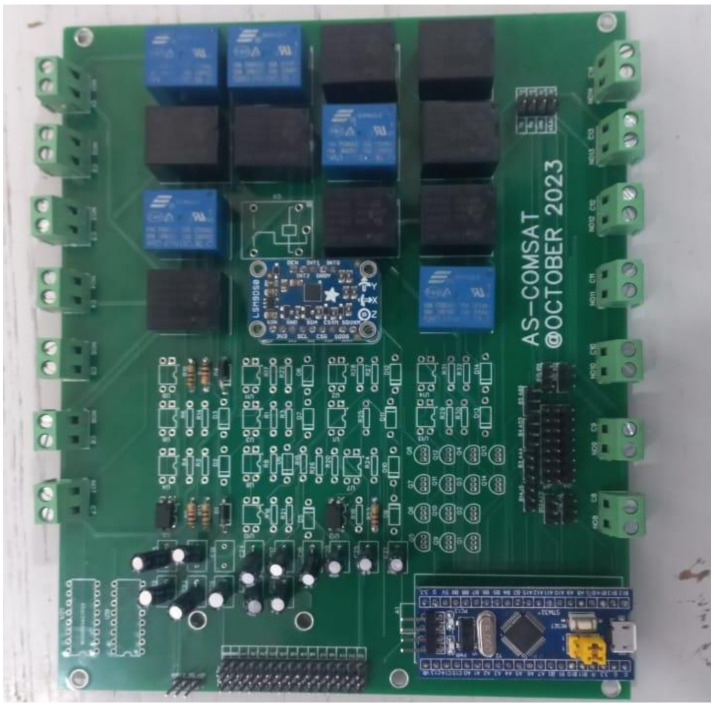


Figure 2-8 Sun sensor PCB board

10.2.5.5 ACS Software Design



The software can be accessed from https://drive.google.com/drive/folders/1GAl7GWIgGsKQ98mlGo1DK3z6c-Tu0ZIL (Attitude_Controller_SW101) which shall be uploaded to the STM32 controller.



https://www.aecenar.com/index.php/companies/as-comsat/as-comsat-tripoli-ecs/system-design-of-attitude-control-system-acs

10.2.6 TT&C Transceiver 2,4 GHz Development

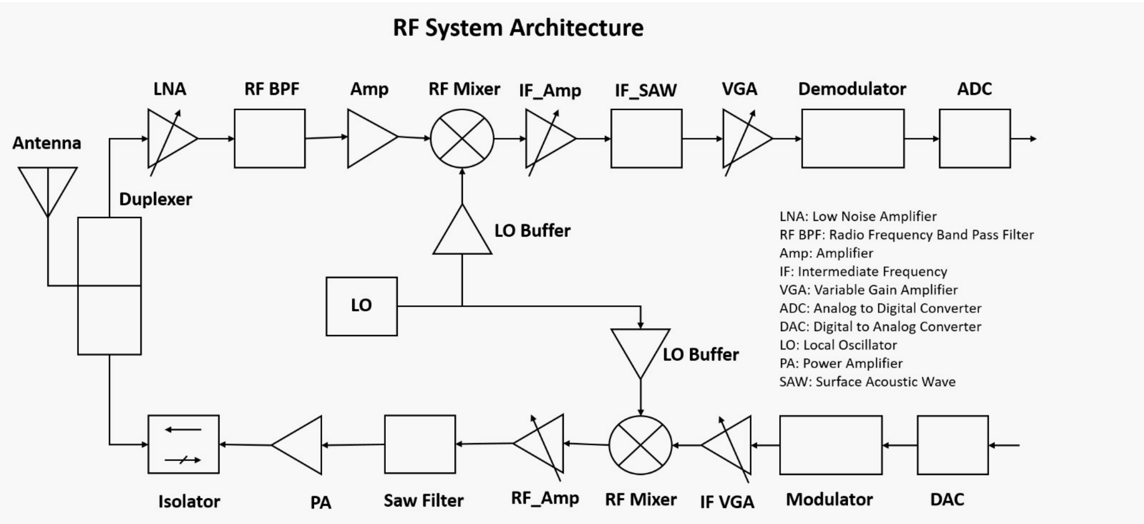
10.2.6.1 Introduction

The Telemetry, Tracking, and Control (TT&C) transceiver is responsible for transmitting back to the ground station the telemetry and tracking data such as: battery health, solar panels health, position updates, acceleration, etc. It is also responsible for receiving navigation commands from the ground station.

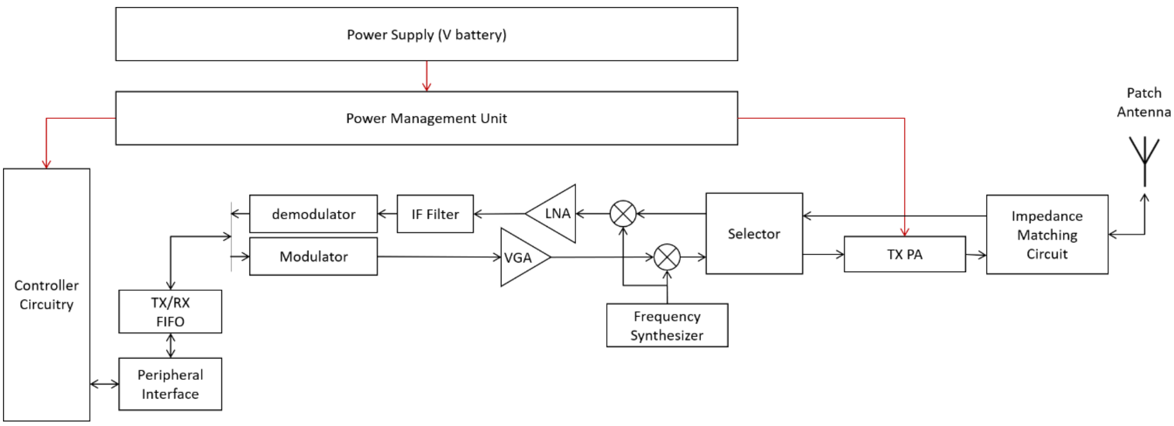
This chapter will discuss in details the design procedure of the 2 versions of TT&C transceivers. Version 1 uses the at86rf233 from microchip along with rfx2401c power amplifier while version 2 uses nrf24l01+ from Nordic semi-conductors along with rfx2401c power amplifier for range extension.

10.2.6.2 System Architecture

UPDATE March 16, 2023



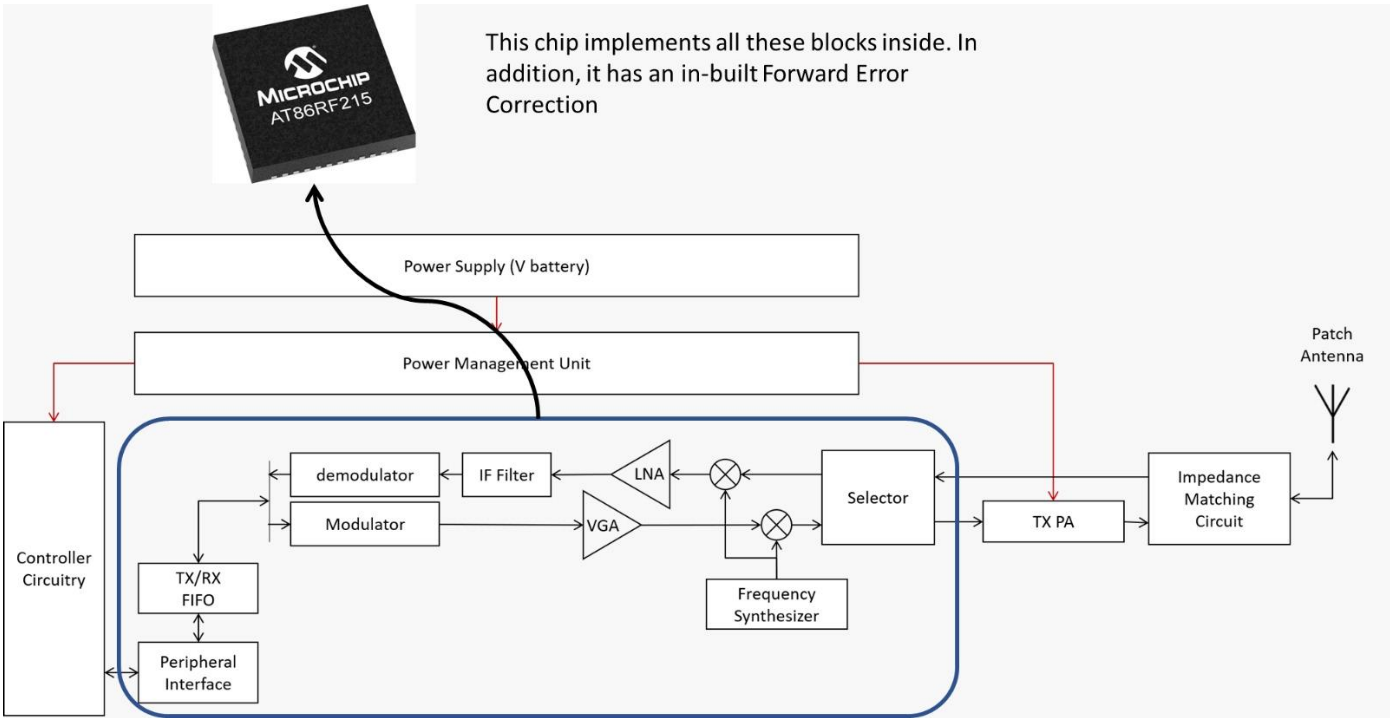
UPDATE March 25, 2023



Frequency: 2.4 GHz

Update 2.6.23 (Transceiver with Microchip AT86RF215):

This is a chip from Microchip. It offers the transmission/receiving over 2.4 - 2.483.5 GHz band.



The chip uses Multi-Rate (MR) FSK, MR-O-QPSK Modulations

10.2.6.3 Bill of Material

ID	Value	Designator	Footprint	Quantity	Manufacturer Part	Manufacturer	Supplier	Supplier Part	Price
1	HDR-M-2.54_2x4	J1	HDR-M-2.54_2X4	1			LCSC	C58363	
2	HDR-M-2.54_1x3	J2	HDR-M-2.54_1X3	1			LCSC	C180248	0.209
3	XL-2012WWC	TX	LED0805-RD_WHITE	1	XL-2012WWC	XINGLIGHT(成兴光)	LCSC	C965820	0.011
4	220	RRX,RTX	R0805	2					
5	LED-0805_R	RX	LED0805_RED	1	17-21SURC/S530-A3/TR8	EVERLIGHT(台湾亿光)		C72037	0.018
6	10p	C10	C0402	1	GRM1555C1H200JA01D	muRata(村田)	LCSC	C76959	0.005
7	HJ-SMA076	ANT	SMA-SMD_HJ-SMA076	1	HJ-SMA076	皇捷	LCSC	C1509221	0.91
8	2.4GHz ~ 2.5GHz	U8	QFN-16_L3.0-W3.0-P0.50-BL-EP1.7	1	RFX2401C	SKYWORKS	LCSC	C19213	0.586
9	1k	R1,R2	R0402	2			LCSC		
10	0.1u	CB2,CB1,CB3,CB4	C0402	4			LCSC	C141382	0.003
11	AT86RF233-ZU	U1	VQFN-32_L5.0-W5.0-P0.50-TL-EP	1	AT86RF233-ZU	MICROCHIP(美国微芯)	LCSC	C177148	3.907
12	16MHz	X1	CRYSTAL-SMD_4P-L3.2-W2.5-BL	1	TAXM16M4RDBCCT2T	TAE(雅晶鑫)	LCSC	C164044	0.073
13	10uF	C1,C3,C4,C5,C6,C8	C0603	6	GRM188R61C106MA73D	muRata(村田)	LCSC	C86277	0.039
14	20pF	C2	C0402	1	GRM1555C1H200JA01D	muRata(村田)	LCSC	C76959	0.005
15	12pF	CX1,CX2	C0402	2	GRM1555C1H120FA01D	muRata(村田)	LCSC	C161526	
16	M7	D1	SMA_L4.4-W2.8-LS5.4-RD	1	M7	BORN(伯恩半导体)	LCSC	C266550	0.011
17	10k	R3,R4	R0402	2			LCSC		
18	10k	R5	R0402	1					
19	BD2327N50100AHF	T1	XFMR-SMD_4P-L1.1-W1.1-P0.50-BL	1	BD2327N50100AHF	Anaren	LCSC	C502744	1.282
20	STM32F103C8_BLUEPILL COPY	U2	STM32F103C8T6-BLUEPILL	1					
21	AMS1117-5.0	U3	SOT-223_L6.5-W3.5-P2.30-LS7.0-BR	1	AMS1117-5.0	PUOLOP(迪浦)	LCSC	C351787	0.07
22	AMS1117-3.3V	U4	SOT-223-4_L6.5-W3.5-P2.30-LS7.0-BR	1	AMS1117-3.3V	WPMtek(维攀微)	LCSC	C2688239	0.07

Figure 2-9 TT&C Transceiver V1 BOM

No.	Quantity	Comment	Designator	Footprint	Value	Manufacturer Part	Manufacturer	Supplier Part	Supplier
1	1		A1	STM32F103C8T6-BLUEPILL-PCB			STM	STM32F103C8T6	
2	2	22pF	C1,C2	C0603	22pF	CC0603GRNPO9BN220	YAGEO(国巨)	C327276	LCSC
3	1	2.2nF	C3	C0603	2.2nF	CC0603KRX7R9BB222	YAGEO(国巨)	C107082	LCSC
4	1	4.7pF	C4	C0603	4.7pF	CC0603CRNPO0BN4R7	YAGEO(国巨)	C527037	LCSC
5	5	1.5pF	C5,C21,C22,C23,C24	C0603	1.5pF	CC0603BRNPO9BN1R5	YAGEO(国巨)	C519100	LCSC
6	1	1pF	C6	C0603	1pF	CC0603BRNPO9BN1R0	YAGEO(国巨)	C115062	LCSC
7	1	33nF	C7	C0603	33nF	CC0603KRX7R9BB333	YAGEO(国巨)	C106856	LCSC
8	1	1nF	C8	C0603	1nF	CC0603KRX7R9BB102	YAGEO(国巨)	C100040	LCSC
9	1	10nF	C9	C0603	10nF	CC0603KRX7R9BB103	YAGEO(国巨)	C100042	LCSC
10	1	0.3pF	C10	C0402	0.3pF	CQ0402BRNPO9BNR30	YAGEO(国巨)	C723570	LCSC
11	1	2.2uF	C11	C0603	2.2uF	C0603C225K8RAC7867	KEMET(基美)	C599715	LCSC
12	1	10nF	C12	C0603	10nF	GCJ188R71H103KA01D	muRata(村田)	C344025	LCSC
13	7	10uF	C13,C14,C15,C16,C17,C18,C20	C0603	10uF				
14	1	20pF	C19	C0603	20pF				
15	1		D1	DO-41_BD2.4-L4.7-P8.70-D0.9-RD					
16	1		H2	HDR-TH_8P-P2.54-V-M-R2-C4-S2.54					
17	1	8.2nH	L1	L0603	8.2nH	VHF160808H8N2JT	FH(风华)	C90204	LCSC
18	1	2.7nH	L2	L0603	2.7nH	SDCL1608C2N7STDF	Sunlord(顺络)	C13817	LCSC
19	1	3.9nH	L3	IND-SMD_L1.8-W1.1	3.9nH	FHW0603UC3N9JST	FH(风华)	C317962	LCSC
20	2	2.4nH	L4,L5	L0603	2.4nH	SDCL1608C2N4STDF	Sunlord(顺络)	C370235	LCSC
21	1		LED1	LED0805-RD_GREEN					
22	1	22kΩ	R1	R0603	22kΩ	0603WAF2202T5E	UNI-ROYAL(厚声)	C31850	LCSC
23	3	1k	R2,R3,R4	R0603	1k				
24	1	22k	R5	R0603	22k				
25	1	6.2k	R6	R0603	6.2k				
26	1	10k	R7	R0603	10k				
27	1	HJ-SMA716	RF1	SMA-SMD_HJ-SMA716		HJ-SMA716	皇捷	C2874826	LCSC
28	4	0	SB1,SB2,SB4,SB5	R0603	0				
29	1	NRF24L01P-R	U1	QFN-20_L4.0-W4.0-P0.50-BL-EP		NRF24L01P-R	NORDIC	C8791	LCSC
30	1	RFX2401C	U2	QFN-16_L3.0-W3.0-P0.50-BL-EP1.7	2.4GHz~2.5GHz	RFX2401C	SKYWORKS	C19213	LCSC
31	1		U3	SOT-223_L6.7-W3.5-P2.30-BR					
32	1		U4	SOT-223-4_L6.5-W3.5-P2.30-LS7.0-BR					
33	1	16MHz	X1	CRYSTAL-SMD_4P-L3.2-W2.5-BL	16MHz	TAXM16M4RDBCCT2T	TAE(泰晶艺)	C164044	LCSC

Figure 2-10 TT&C Transceiver V2 BOM

In Figure 2-9 and Figure 2-10, the listed BOMs are generated from EasyEDA software. All parts were purchased from https://www.lcsc.com/ and the complete excel sheets of the BOQs are attached.

TRX_V2_BOQ_111123.xlsx

TRX_V1_060723.csv

10.2.6.4 Hardware Design (V1) (Canceled Due To Defect in Design)

When designing RF circuits, plenty of measures must be taken into account in order to design a fully-effective RF system.

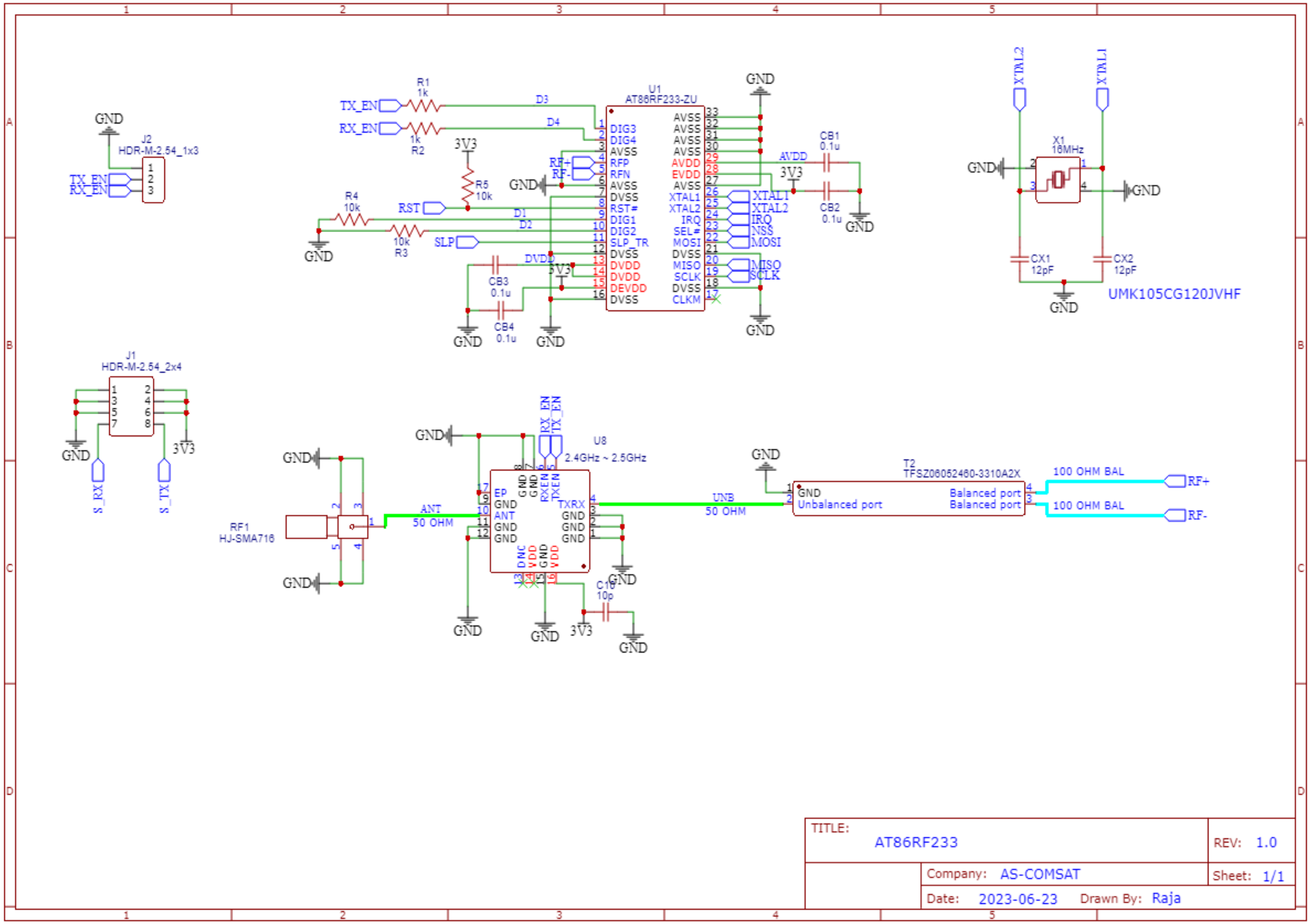


Figure 2-11 TT&C transceiver schematic (AT86RF233 interface)

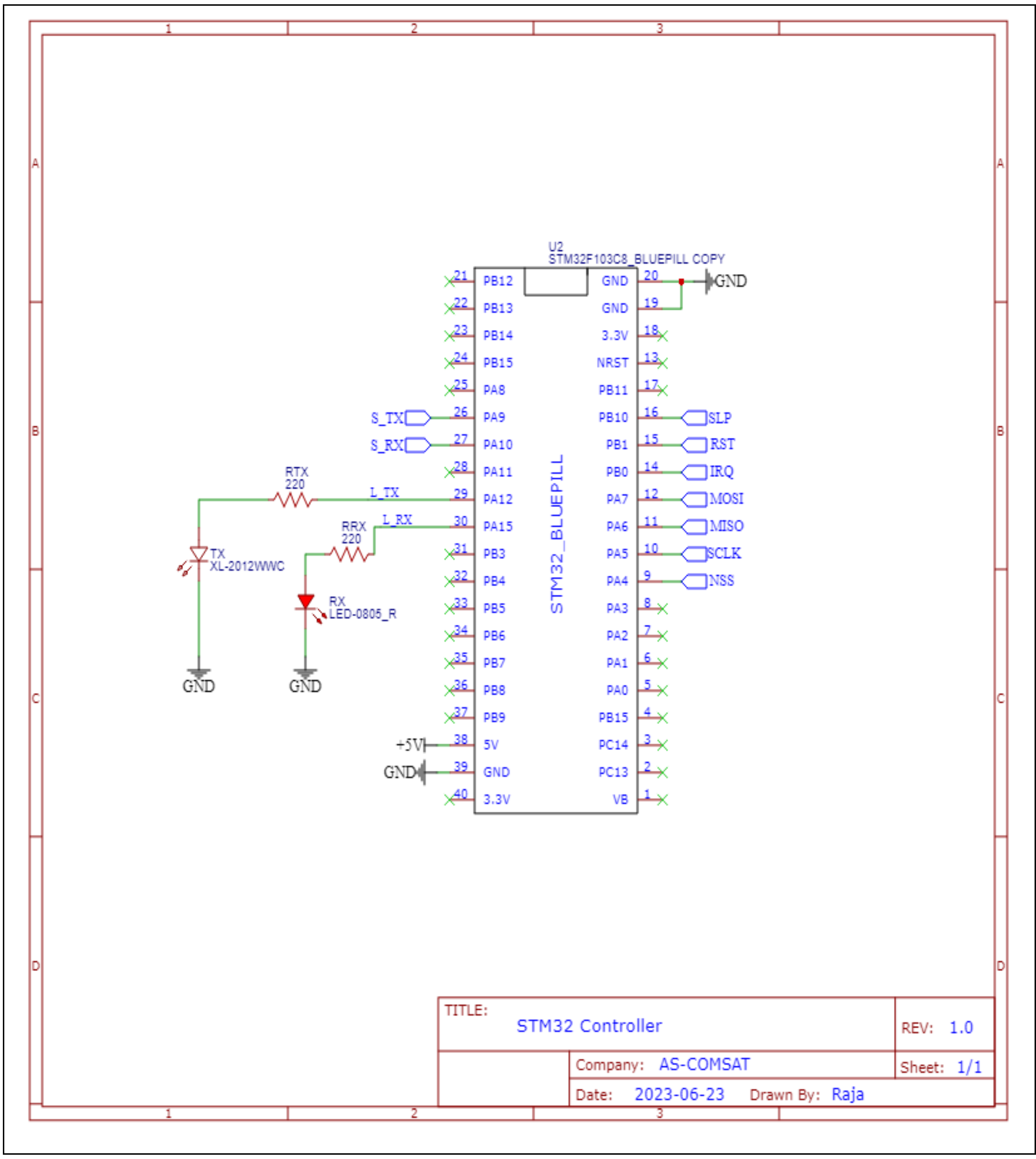


Figure 2-12 TT&C transceiver schematic (Controller interface)

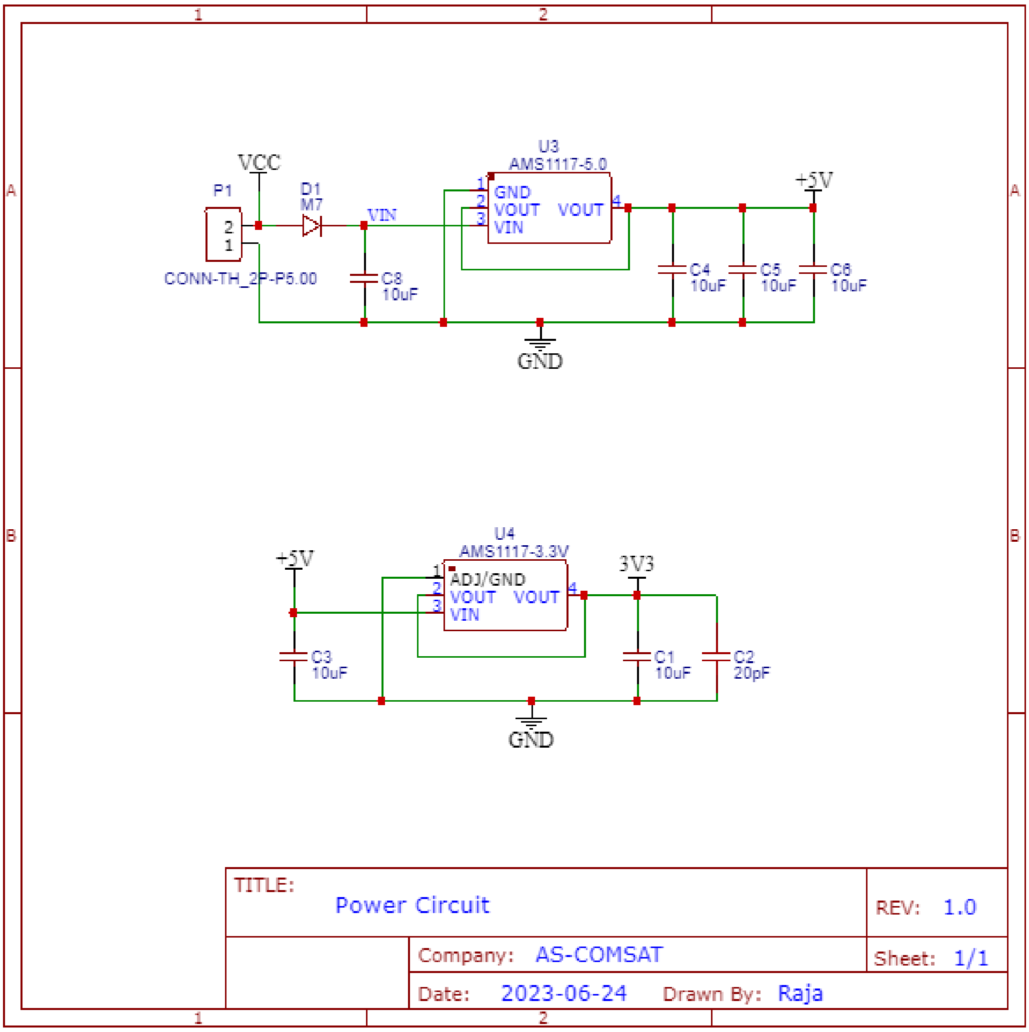


Figure 2-13 TT&C transceiver schematic (Power Management)

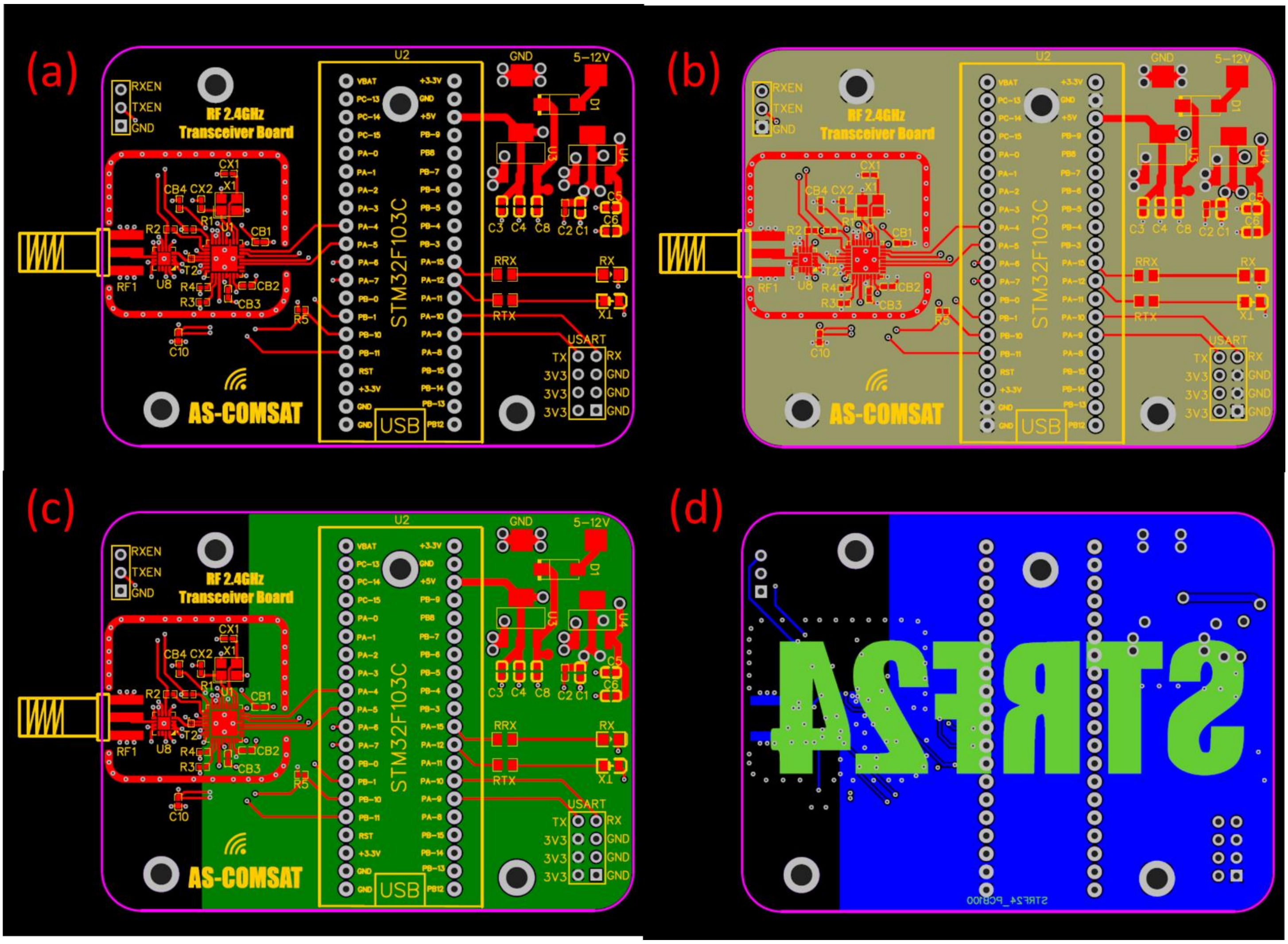


Figure 2-14 TT&C transceiver PCB design. (a) Top layer (signal layer), (b) second layer (GND plane), (c) Third later (3V3 plane), (d) Bttom layer (signal layer)

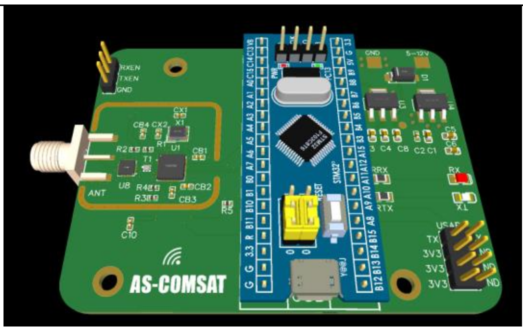


Figure 2-15 TT&C transceiver 3D model

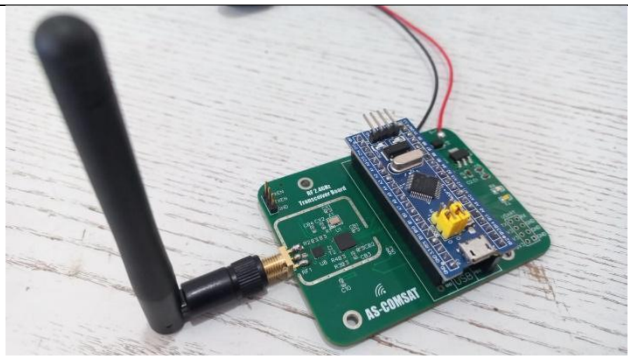


Figure 2-16 TT&C implemented PCB

This version of the transceiver was canceled on November 8, 2023 due to several reasons including:

- Noisy connections of SPI traces: The SPI's MISO pin is generating noisy data and the master controller (STM32) was not able to process them.

- Inefficient impedance matching circuit: Since the transceiver chip is too old, majority of recommended parts for the impedance matching network were not available.

Hence, to solve the aforementioned problems, a version 2 of the transceiver was designed and implemented.

10.2.6.5 Hardware Design (V2)

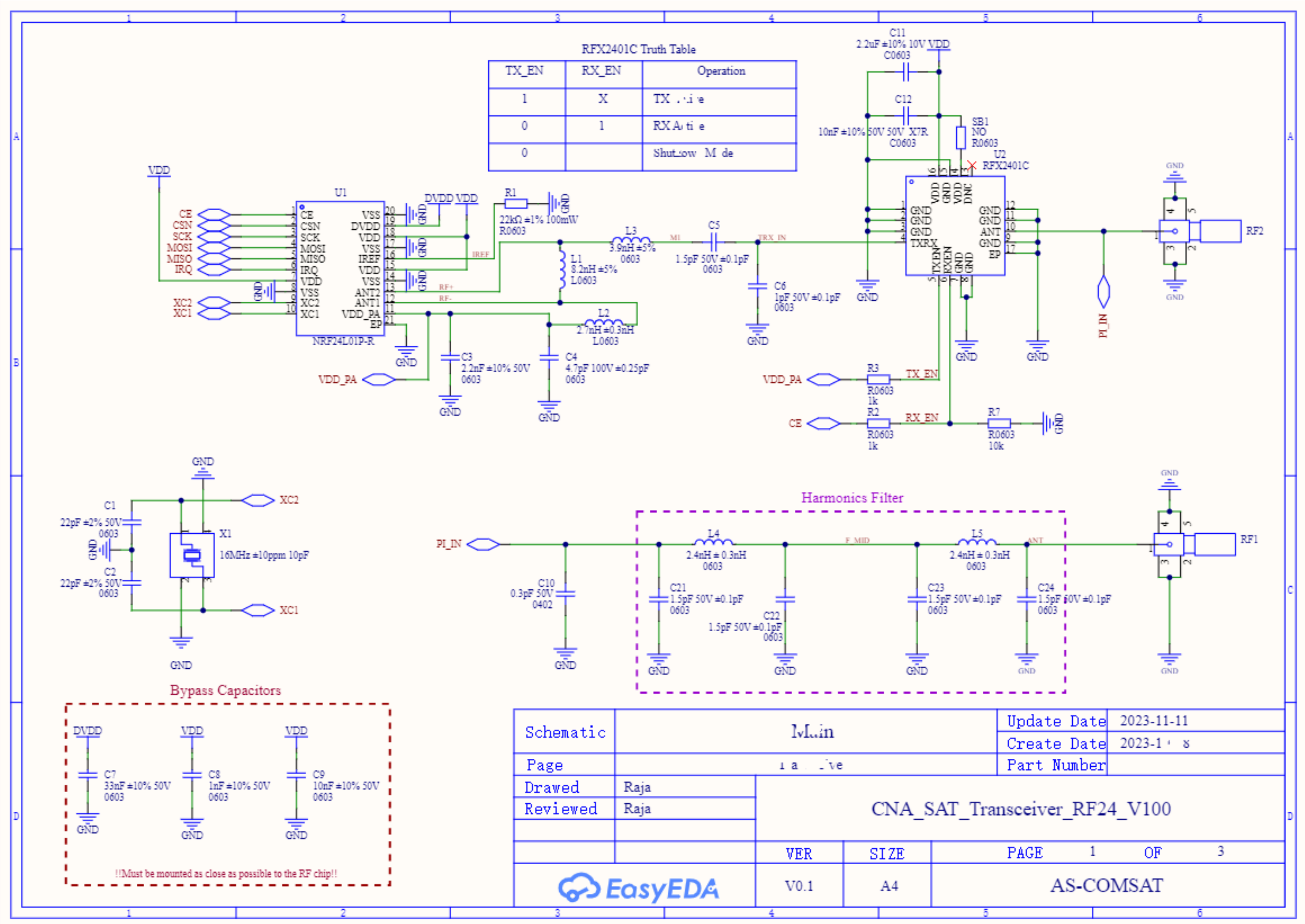


Figure 2-17 TT&C transceiver V2 main schematic

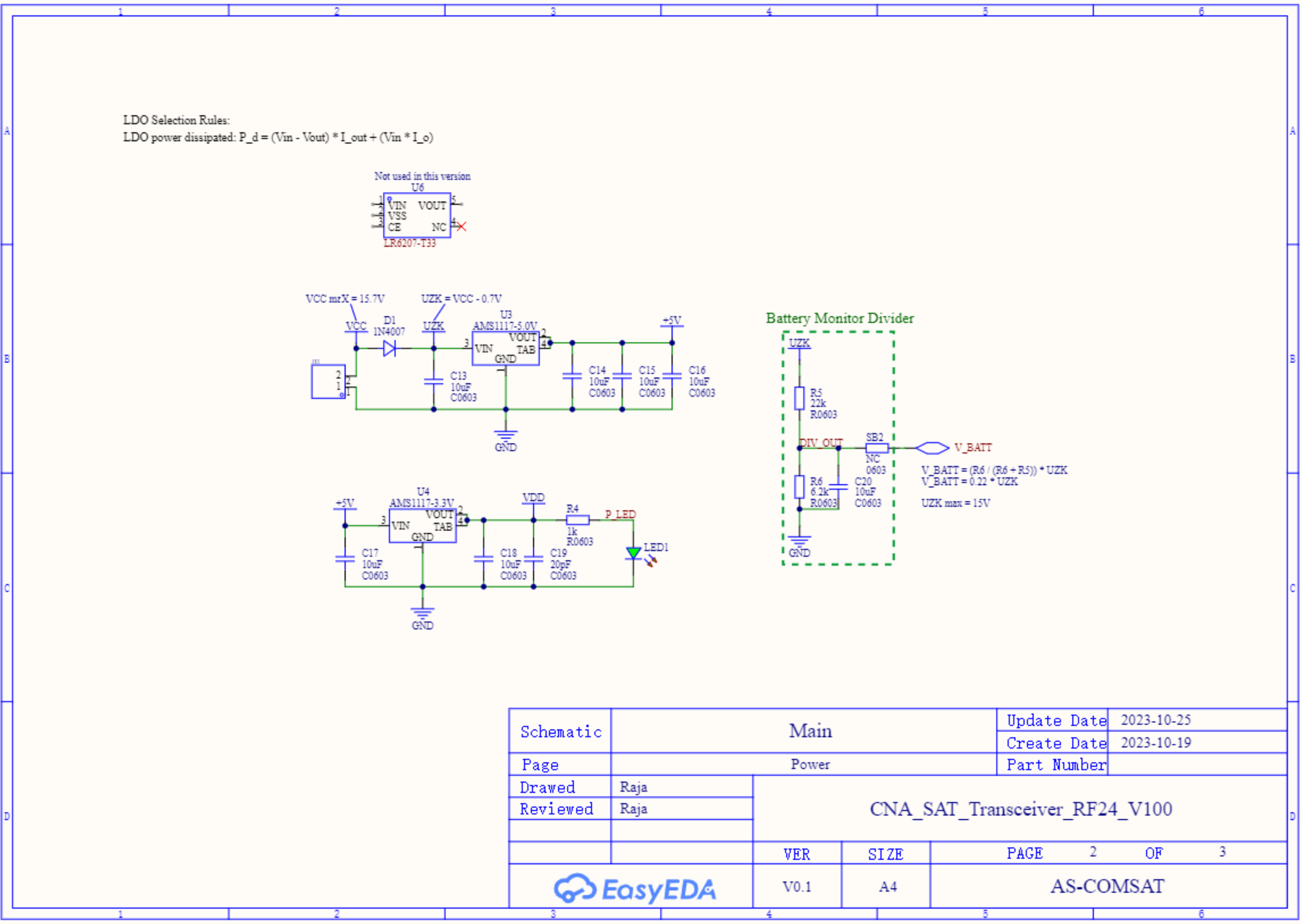


Figure 2-18 TT&C transceiver V2 power schematic

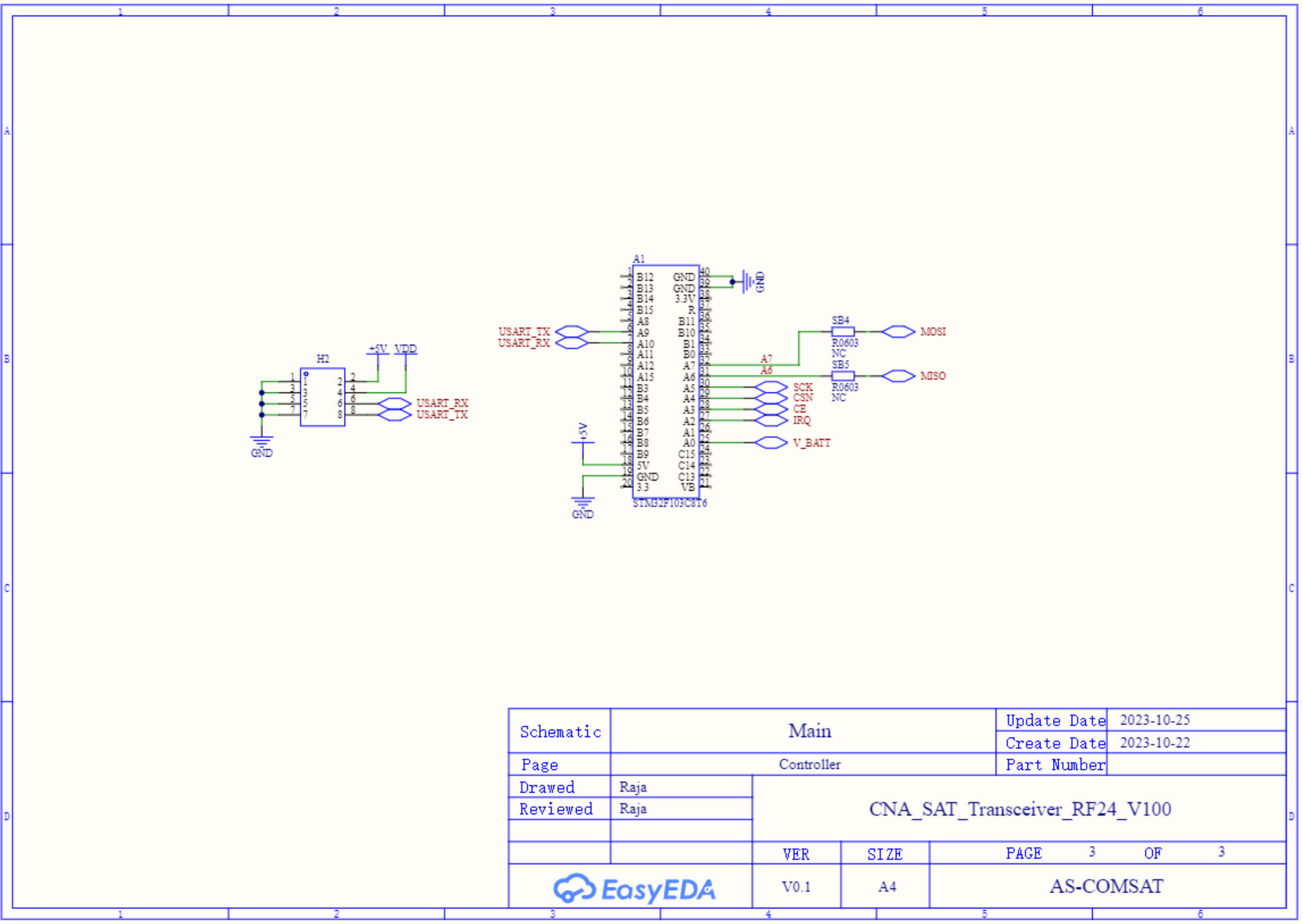


Figure 2-19 TT&C transceiver V2 controller schematic

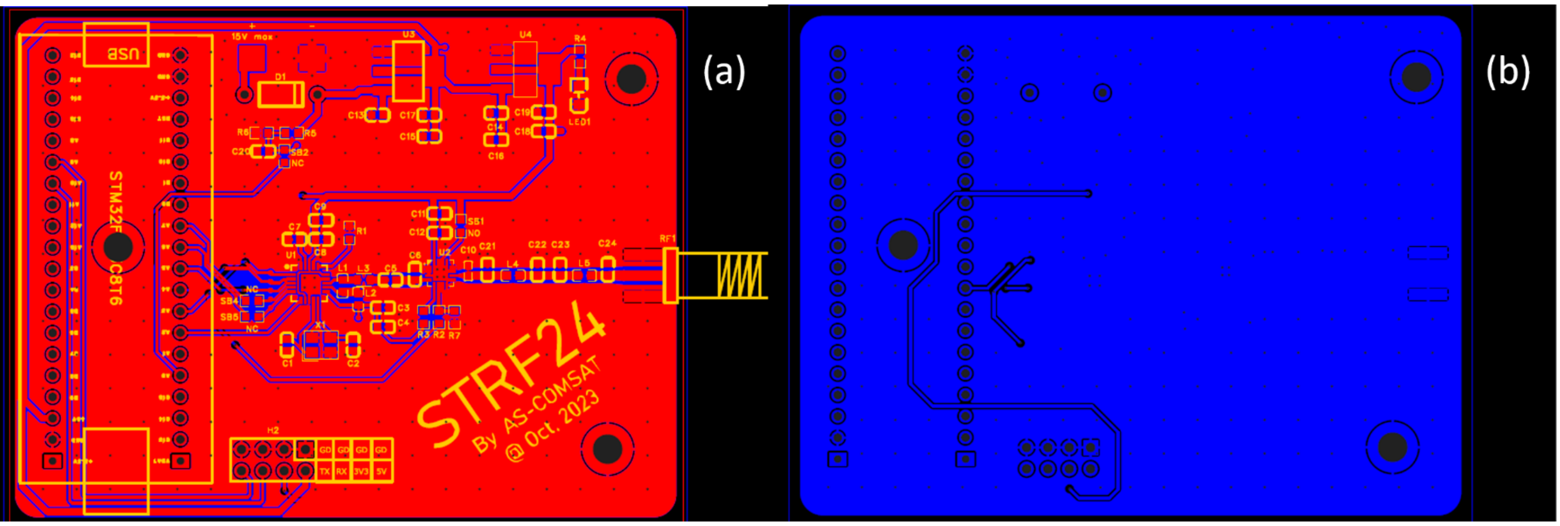


Figure 2-20 TT&C transceiver V2 PCB layout. (a): top layer, (b): bottom layer

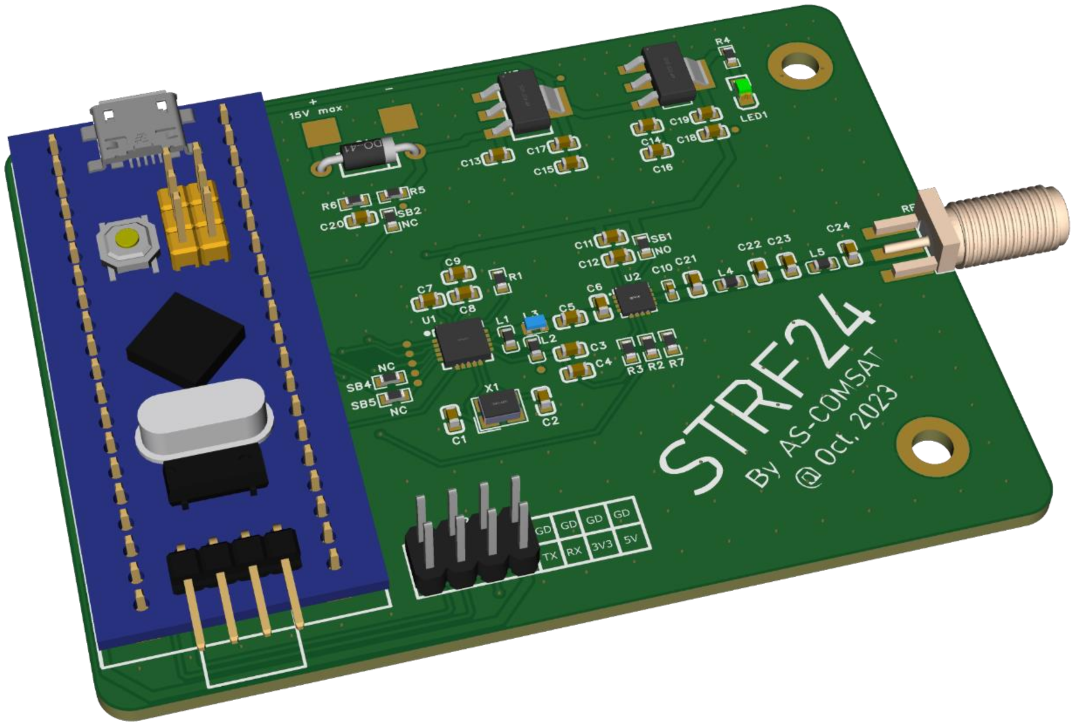


Figure 2-21 TT&C transceiver 3D model

Power Section

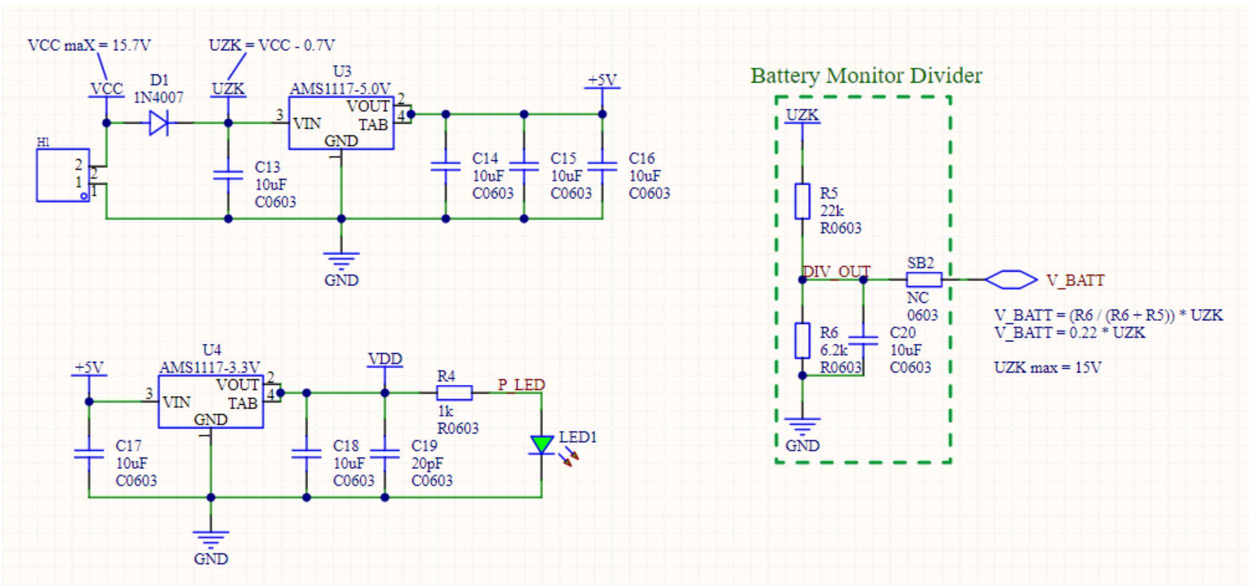


Figure 2-22 TT&C transceiver V2 power supply part snippets

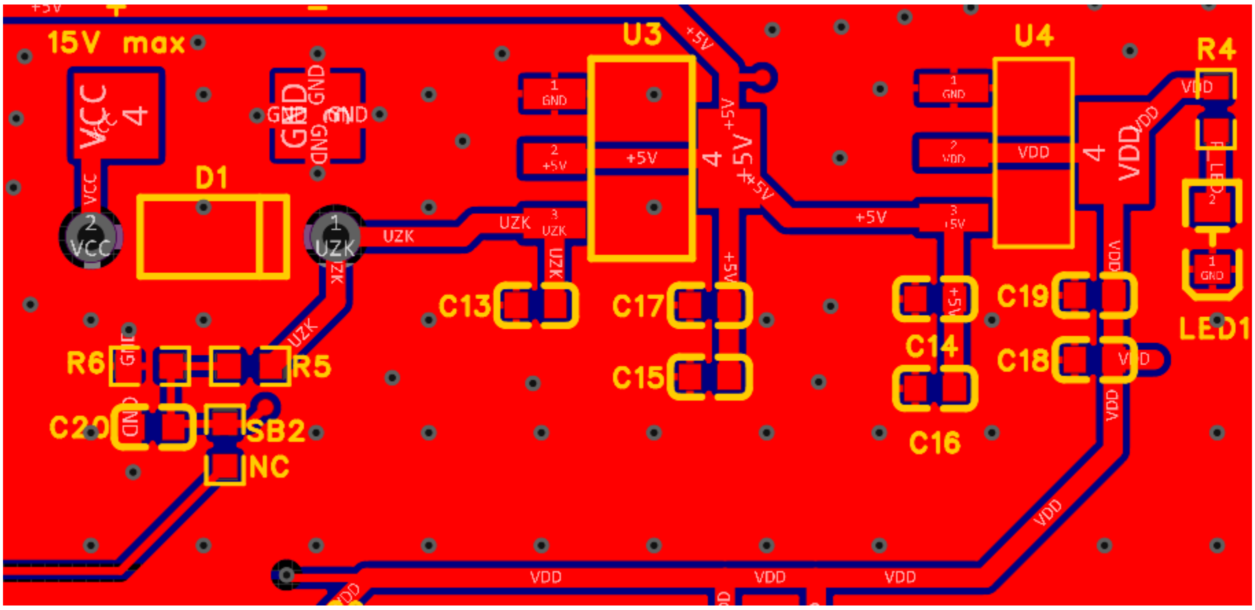


Figure 2-23 TT&C transceiver V2 power supply PCB part snippets

The power section is responsible for delivering sufficient power for all the circuit components. Figure 2-22 shows the schematic of the power system designed for this transmitter.

When designing a power circuit, several factors must be taken into consideration:

- Required output voltage: There are 2 stages of Low Drop-Out (LDO) regulator where it is required to supply 2 voltage levels to the transceiver; 5V and 3.3V. The 5V is used to powerup the microcontroller and the 3.3V is used to powerup all the other components.
- Required output power: Table 2-1 shows the required output power for each subcircuit.
- LDO self-power dissipation: This is an important factor which reflects the power dissipated in the LDO itself when delivering/not delivering power to the output. The quality of the LDO can be reflected from what is called the Quiescent Current (Iq or Icc). This current expresses the current dissipation when there is no load connected. The lower Icc is, the better the LDO is. The LDO power dissipation can be calculated as follows:

$$P_D = (V_{in} - V_{out}) \times I_{out} + V_{in} \times I_{cc} \qquad 2\text{-}4)$$

  Table 2-2 shows the specifications of AMS1117 LDOs extracted from their datasheets. From equation 2-4), we can calculate the maximum power dissipated through the LDOs and their value is inserted in Table 2-1.
- Reverse polarity protection: This aspect is important to mitigate faulty polarity connection of the battery. Diode D1 is used as reverse polarity protector diode. If the supply voltage is connected correctly, the diode allows the current to pass from its anode to its cathode. However, if the input battery is reversed, the diode prevents current from passing and the circuit remains off.

- PCB trace width: power lines in the PCB shall be designed according to the required current passing through them. The trace width can be calculated as follows:

$$I_{trace} = K \times \Delta T^{0.44} \times (W \times H)^{0.725} \qquad 2\text{-}5)$$

Where, $I_{trace}$is the required current, K is a constant (0.024 for internal traces and 0.048 for external traces), ΔT is the temperature rise above ambient temperature (~ 10°), W is the trace width, and H is the trace height (typically 0.035 mm). The track width of the power traces was calculated (assuming ΔT = 10 and K = 0.048) as **0.7mm** which withstands approximately **1.9 A**.

Table 2-1 Components power dissipation

Component	Max Power Dissipation
Nrf24l01	49.5 mW @ 3.3V
RFX2401C	390 mW @ 3.3V
Controller	75 mW @ 5V
Total	440 mW @ 3.3V @ 134mA 75 mW @ 5V @ 15 mA
LDO power dissipation	AMS1117-5V @ Vin = 15V, Iout = 15mA: 315mW AMS1117-5V @ Vin = 15V, Iout = 0: 165 mW AMS1117-5V @ Vin = 11.3V, Iout = 15 mA: 220 mW AMS1117-5V @ Vin = 11.3V, Iout = 0: 125 mW AMS1117-3.3V @ Vin = 5V, Iout = 134mA: 283 mW AMS1117-3.3V @ Vin = 5V, Iout = 0: 55 mW
Power dissipated from input battery (Vin = 11.3V)	P_{total} (Vin = 15V) = 283 + 315 ~ 600 mW P_{min} (Vin = 15V) = 55 + 165 = 220 mW P_{total} (Vin = 11.3V) = 283 + 315 ~ 503 mW P_{min} (Vin = 11.3V) = 55 + 165 = 180 mW

Table 2-2 AM1117 specifications

LDO		
	5V Output	3.3V Output
Min Input voltage	6.5V	5V

Max input voltage	15V	15V
Max out current	1.5A	1.5A
Quiescent current	11 mA	11mA

The second section of the power section is the battery monitoring sub-circuit. It is used to monitor the health of the connected power supply and repots its status. It consists of a voltage divider network that shifts the voltage down by a ratio of 1:4.45. Given the constraint of V battery = 15V max, the output of the divider gives 3.3V (the maximum acceptable voltage by the STM32's pins). SB2 (shown in Figure 2-22) is a 0 Ohm resistor solder bridge used to connect/disconnect the divider network.

Controller Section (STM32 Microcontroller)

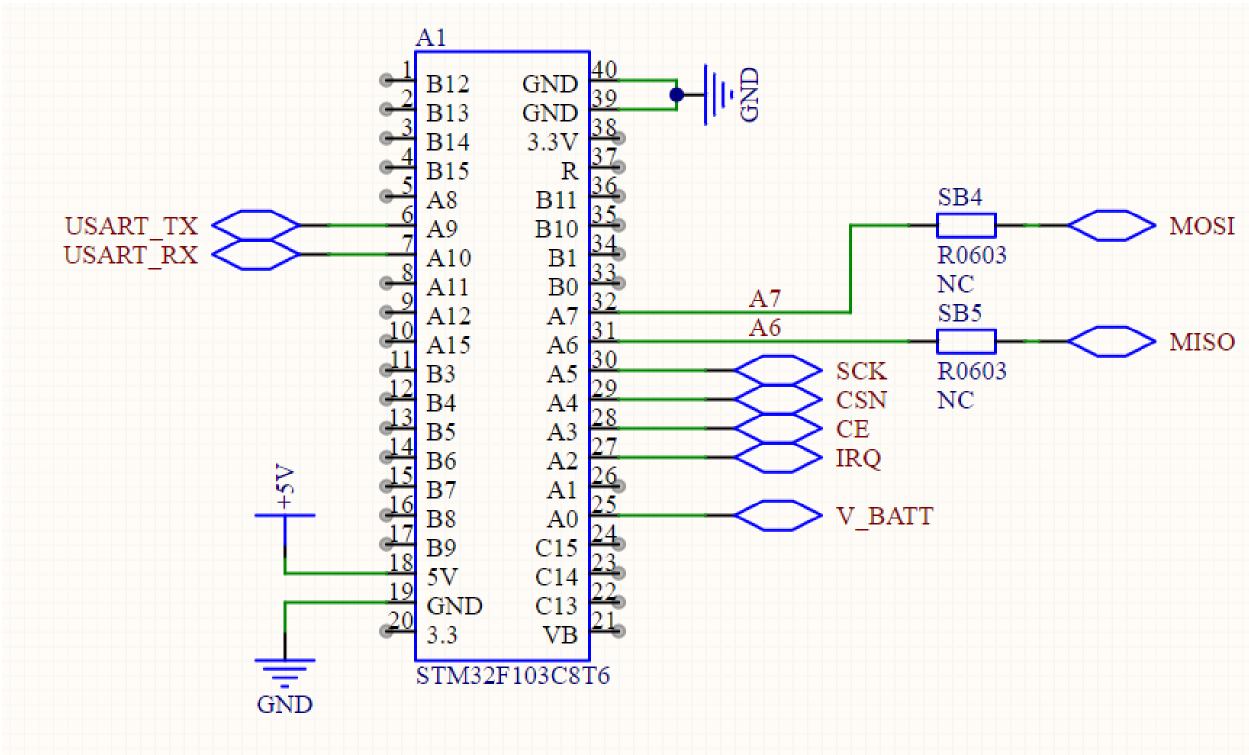


Figure 2-24 TT&C transceiver V2 controller part snippets

The controller section uses STM32 as the microcontroller driving the transceiver. SB4 and SB5 are Solder Bridges expressed as 0 Ohm resistors. They are used in case of faulty connection. Thus, it is easy to desolder them and correct the connections accordingly.

Transceiver Section

To begin with the transceiver itself, the nrf24l01+ chip, Figure 2-26 shows the transceiver part only. The transceiver communicates with the microcontroller via SPI connection (MOSI, MISO, CSN, SCK) mode 0 (see Figure 2-25).

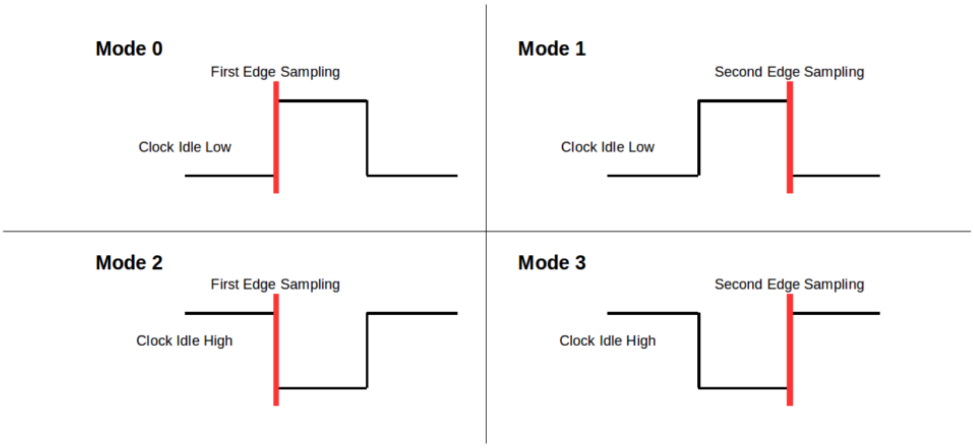


Figure 2-25 SPI communication modes

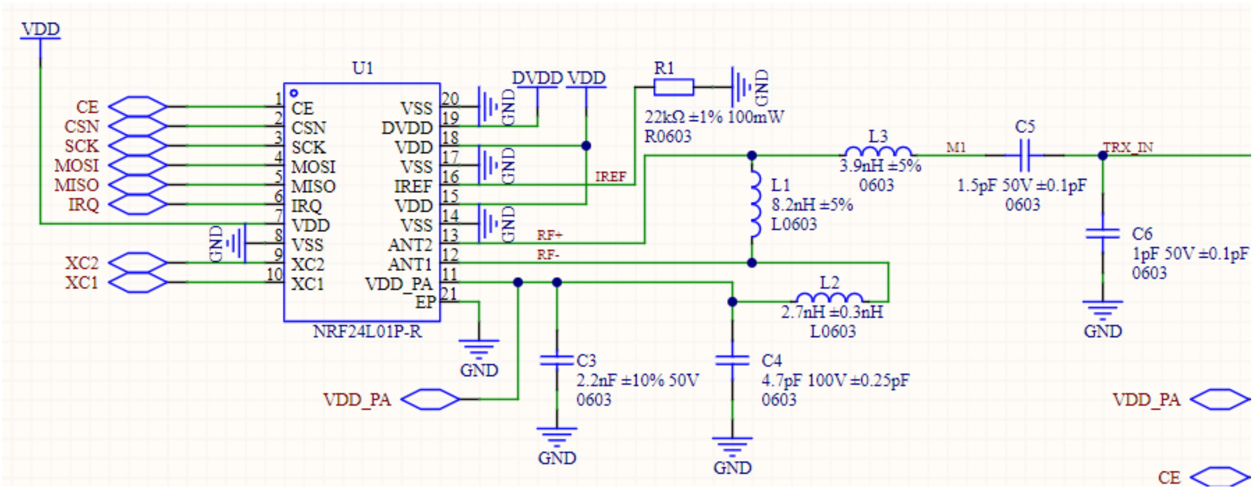


Figure 2-26 TT&C transceiver V2 transceiver part snippets

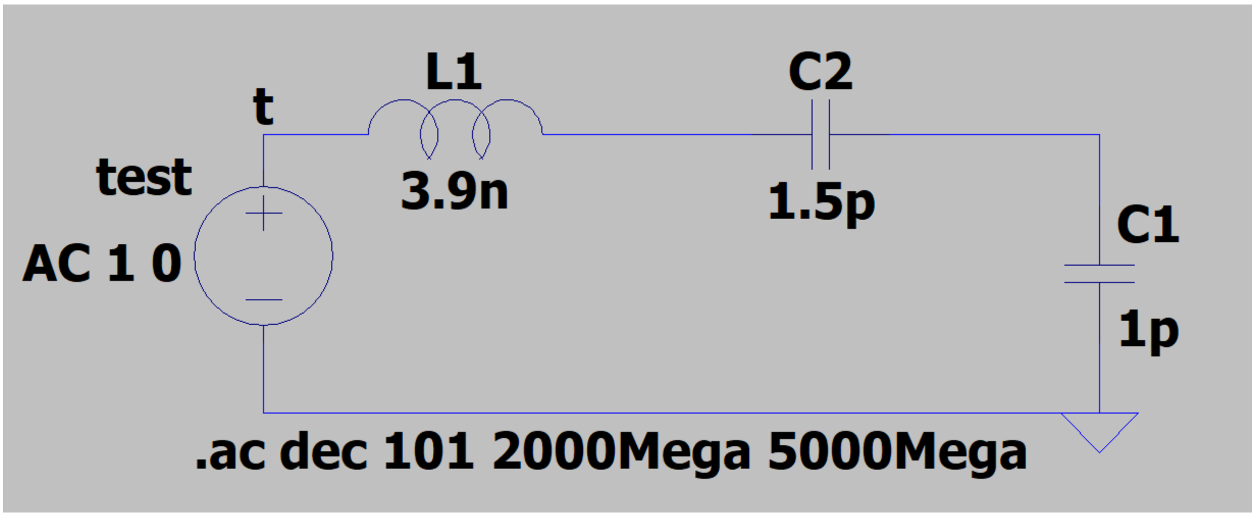


Figure 2-27 Impedance matching stage 1 simulation

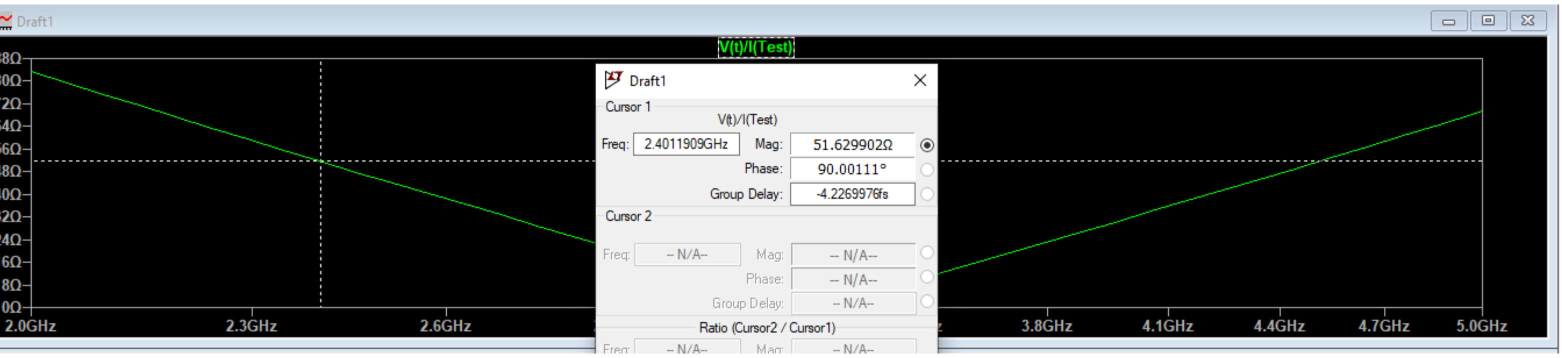


Figure 2-28 Impedance matching stage 1 results

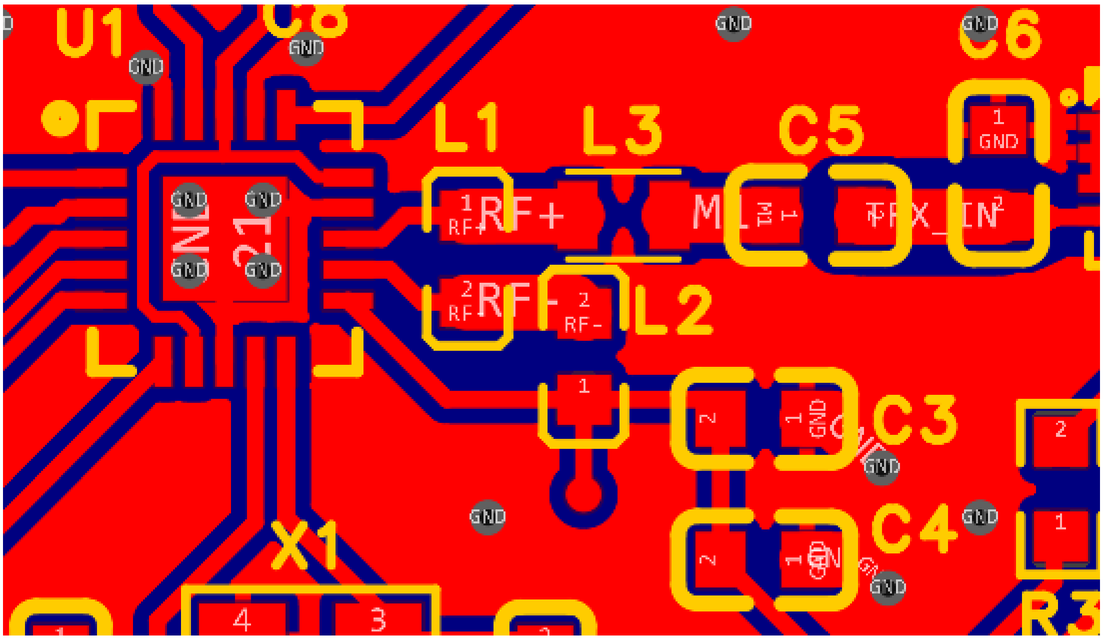


Figure 2-29 TT&C transceiver V2 transceiver PCB part snippets

Oscillator Section

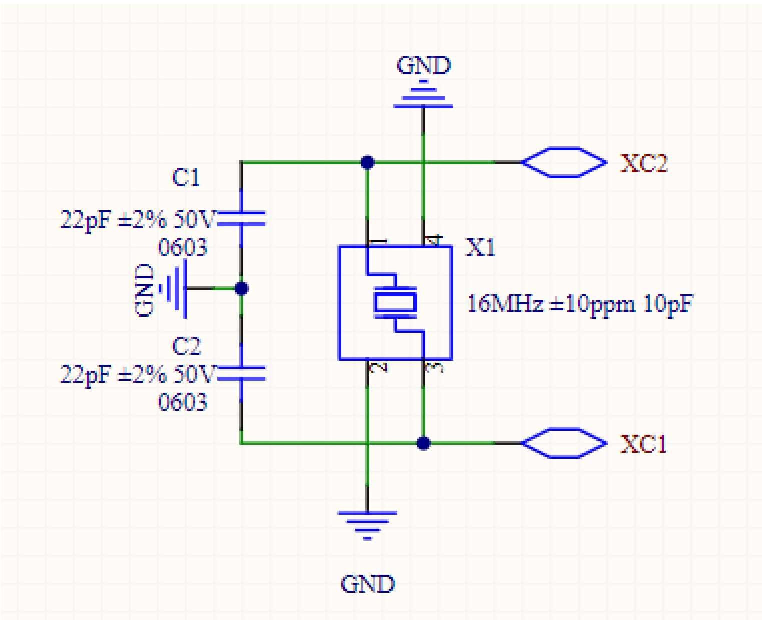


Figure 2-30 Transceiver V2 oscillator part snippets

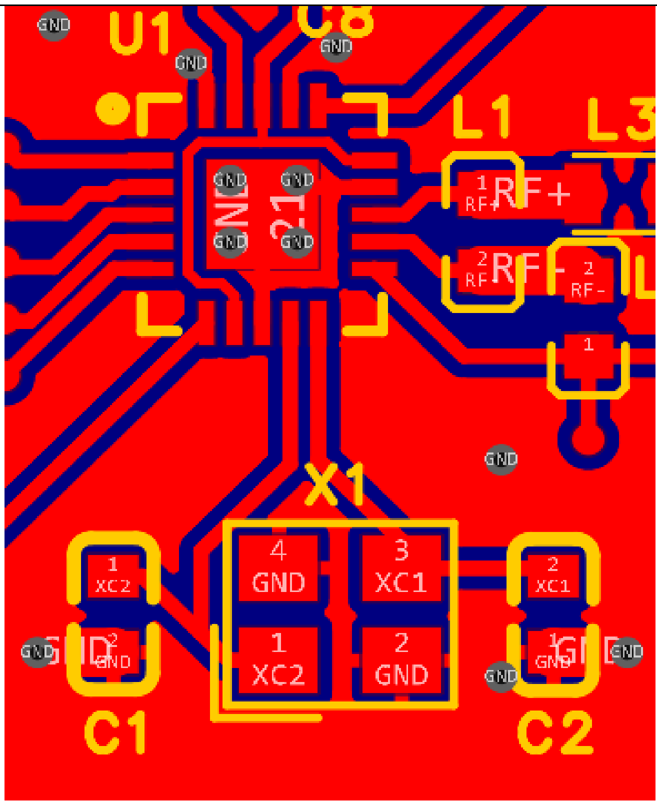


Figure 2-31 Transceiver V2 oscillator PCB part snippets

Power Amplifier Section

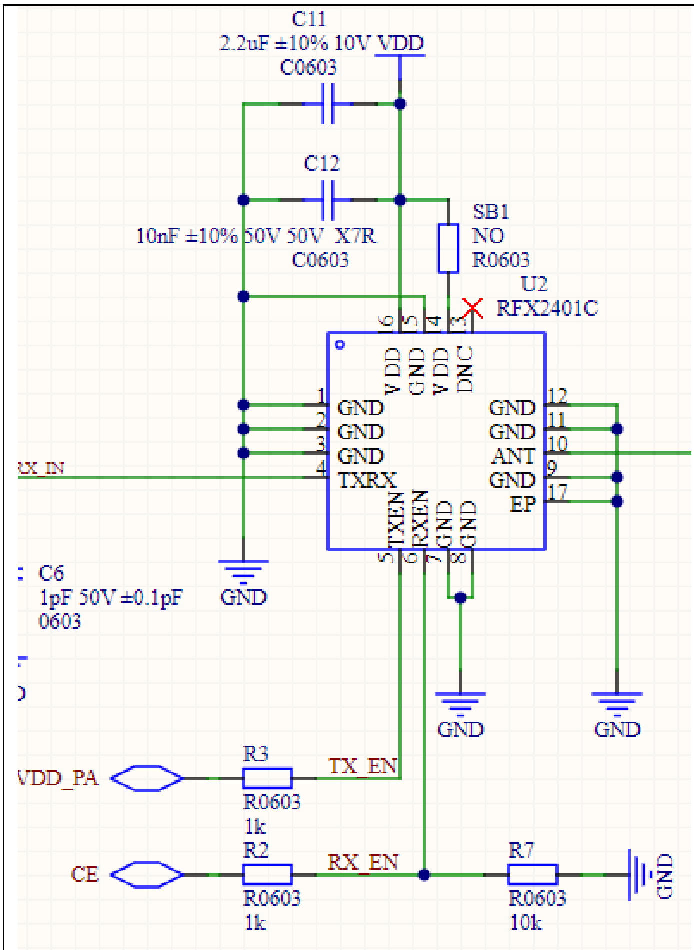


Figure 2-32 TT&C transceiver V2 power amplifier part snippets

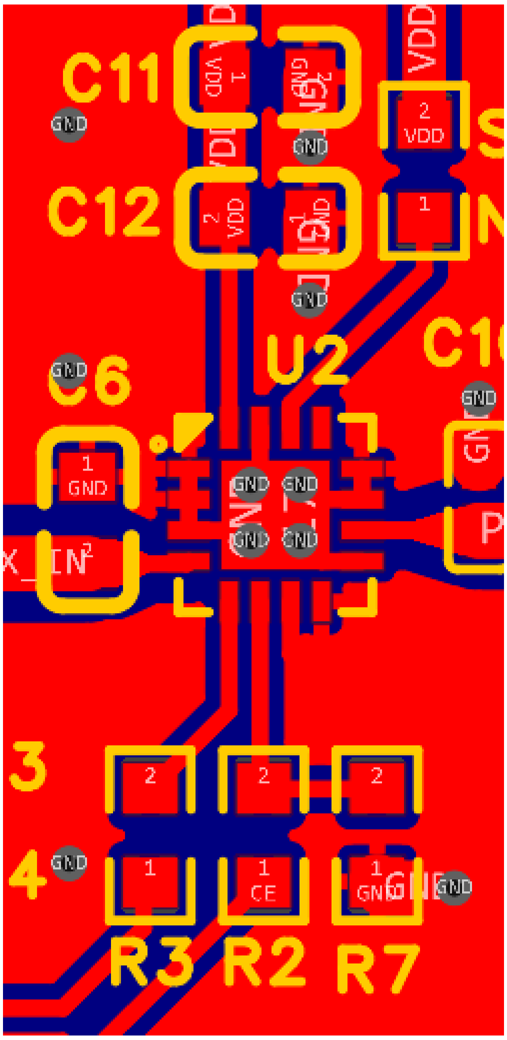


Figure 2-33 TT&C transceiver V2 power amplifier PCB part snippets

Harmonics Filter Section

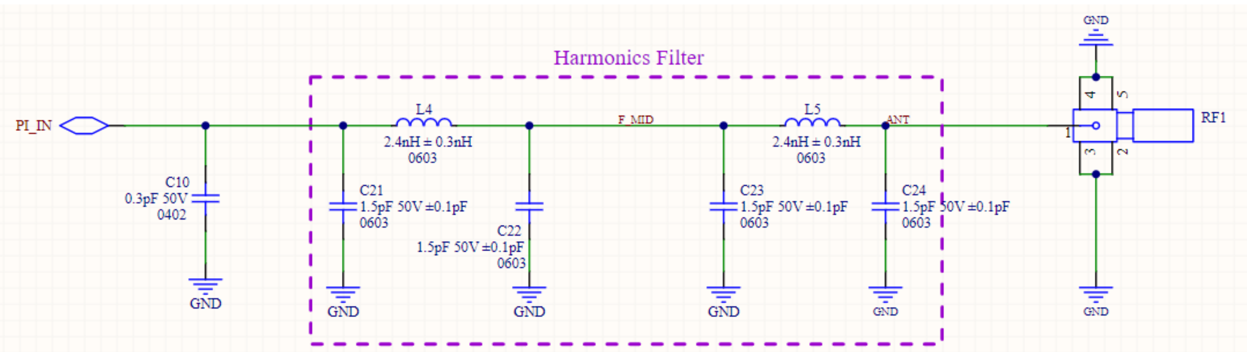


Figure 2-34 Transceiver V2 harmonics filter part snippets

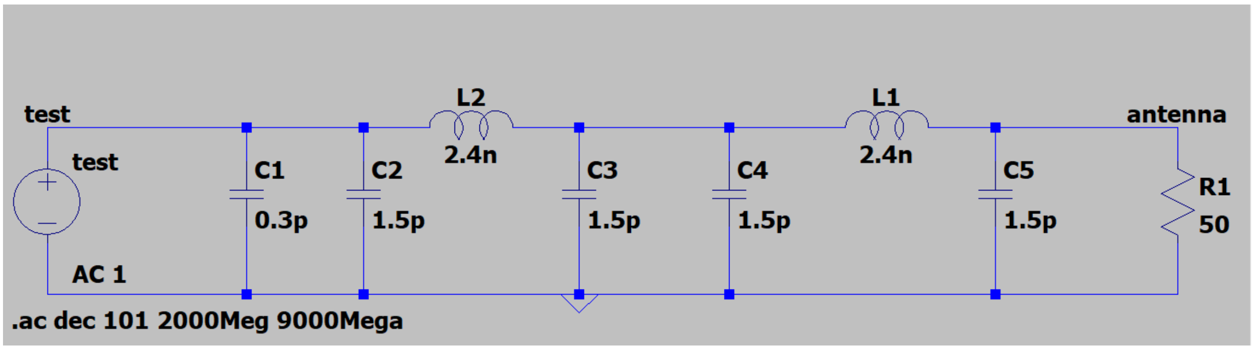


Figure 2-35 Harmonics filter LTspice simulation

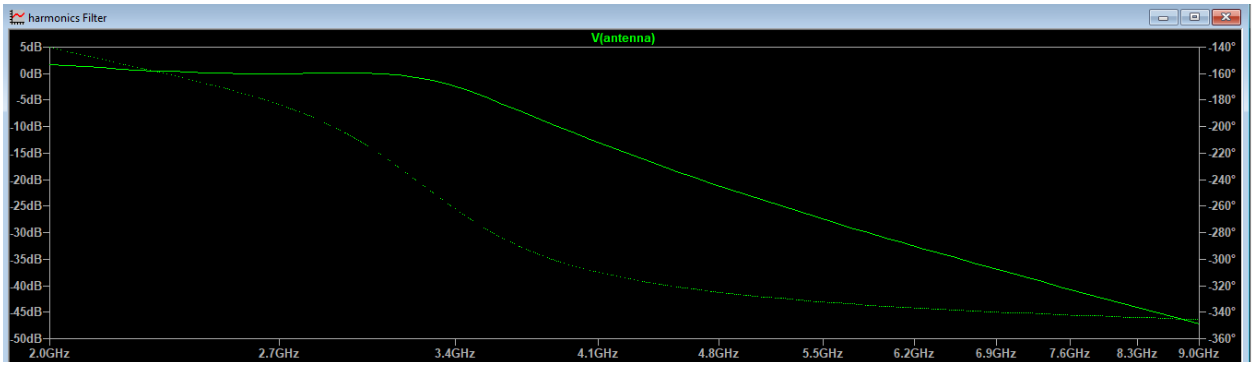


Figure 2-36 Harmonics filter results (output power)

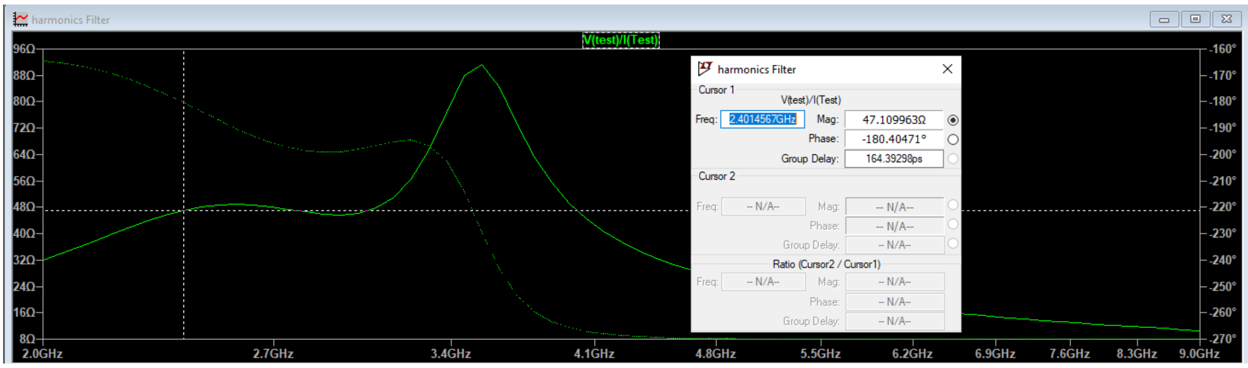


Figure 2-37 Harmonics filter results (impedance)

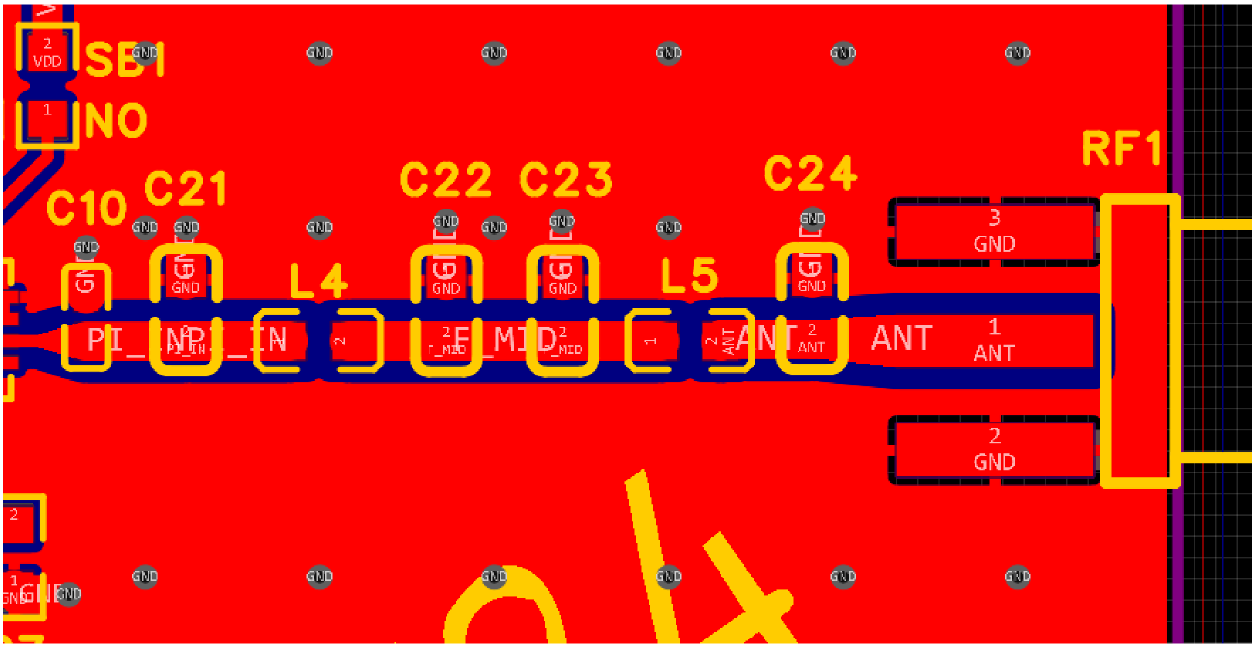


Figure 2-38 Transceiver V2 harmonics filter PCB part snippets

Trace width = 0.9 mm.

10.2.6.6 TT&C Software Design (on STM32 microcontroller)

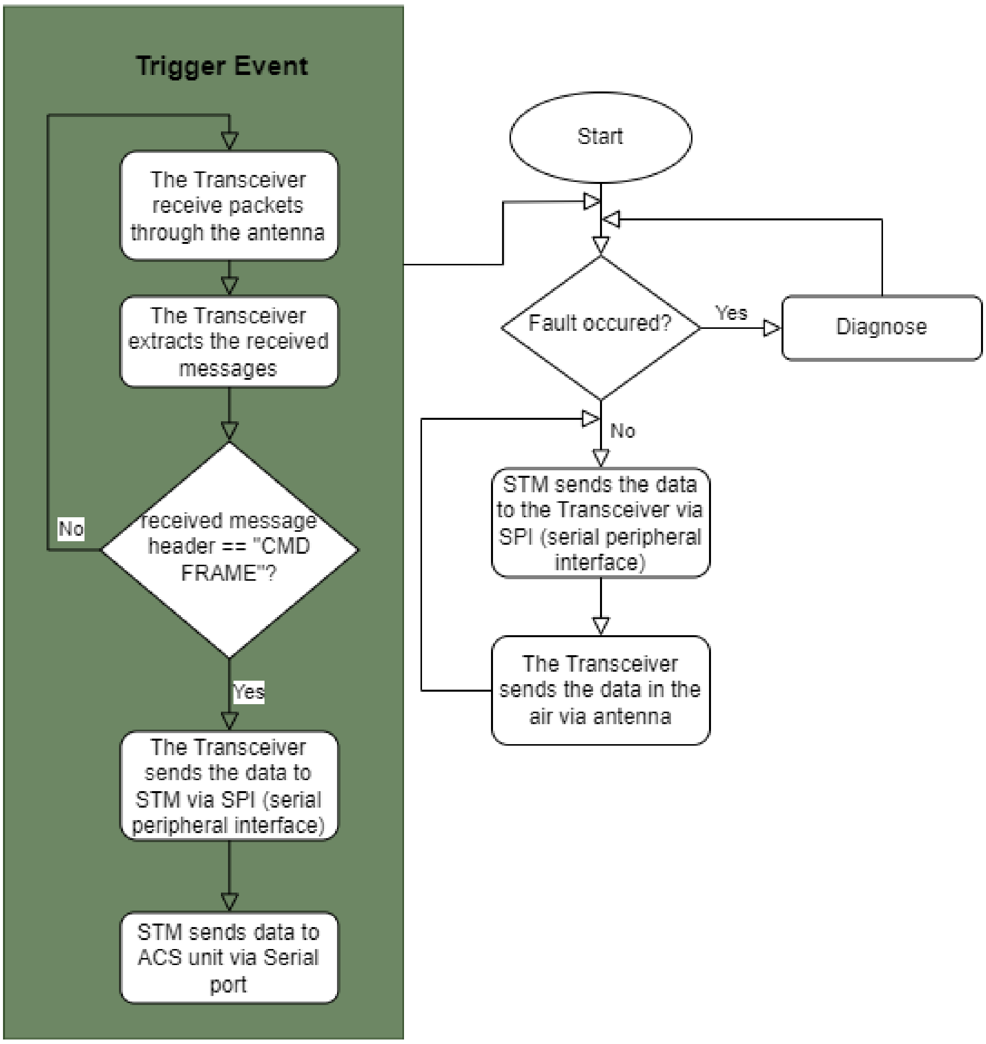


Figure 2-39 Satellite transceiver flowchart

The satellite needs to communicate with the ground station via the transceiver. The transceiver is responsible to collect all the data from the plugged STM controllers (units) and send it to the ground station via antenna. In order for the transceiver to send the data, it needs first to get the data from the STM. The STM sends the data to the transceiver via SPI (serial peripheral interface). Lastly, the transceiver sends the data in the air via antenna in order for the data to be fetched by the transceiver of the ground station. The transceiver has a trigger event that occurs depending if the GUI on the ground station sent any command to the satellite. The transceiver receives the message and extract it. It then checks if the received message is "CMD FRAME", then the transceiver sends the message to the STM via SPI (serial peripheral interface). Lastly, the STM sends the message to the ACS unit via Serial port.

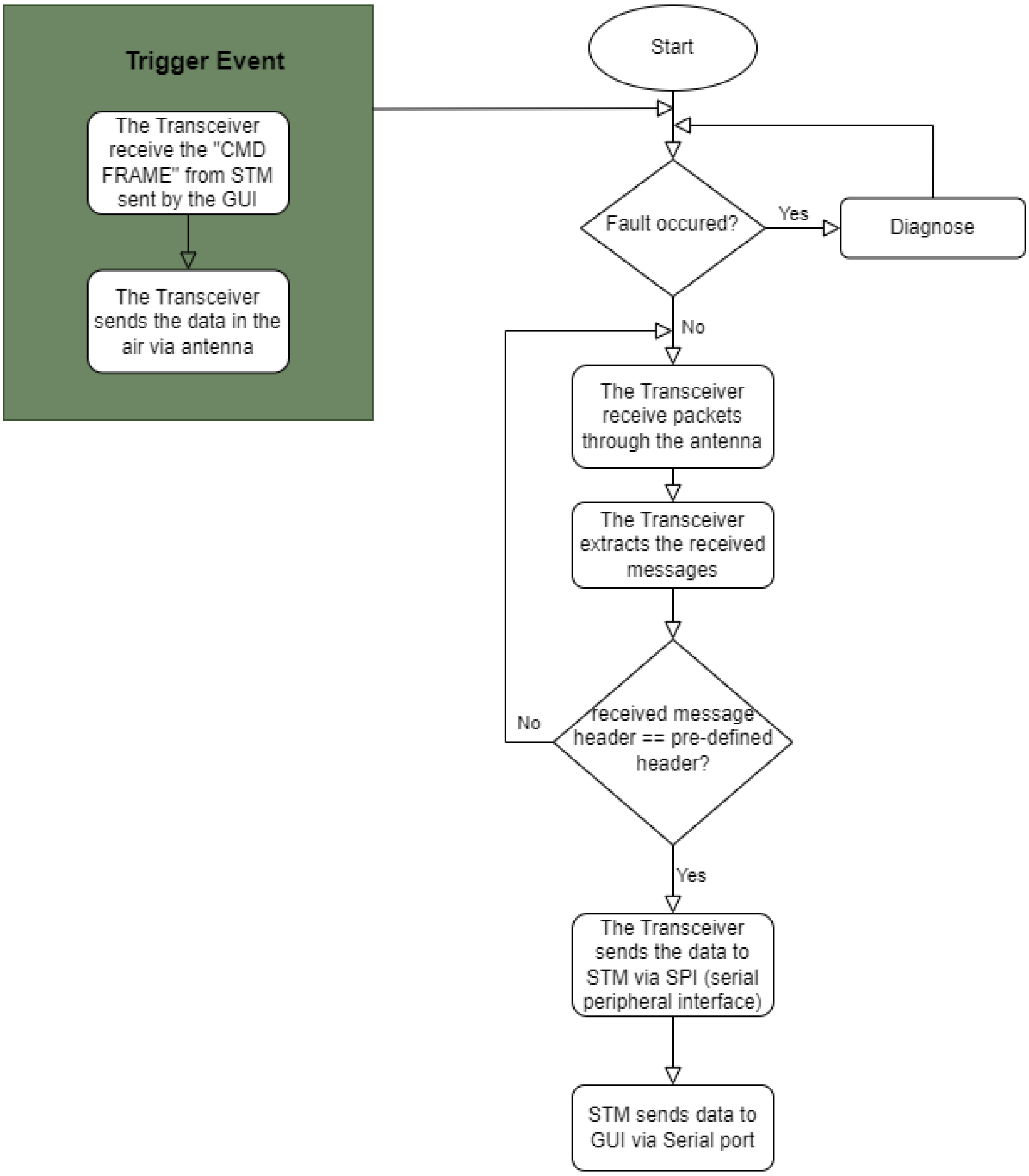


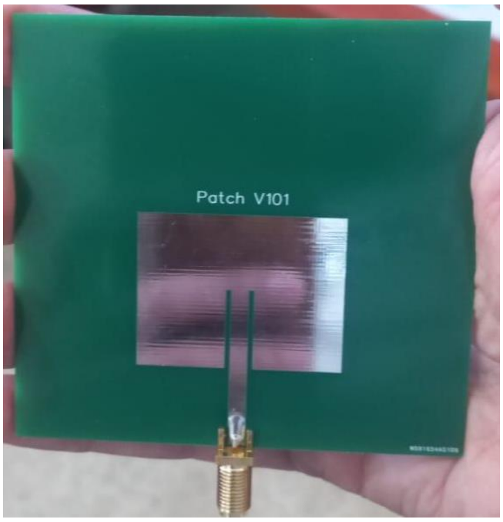
Figure 2-40 Ground-station transceiver flowchart

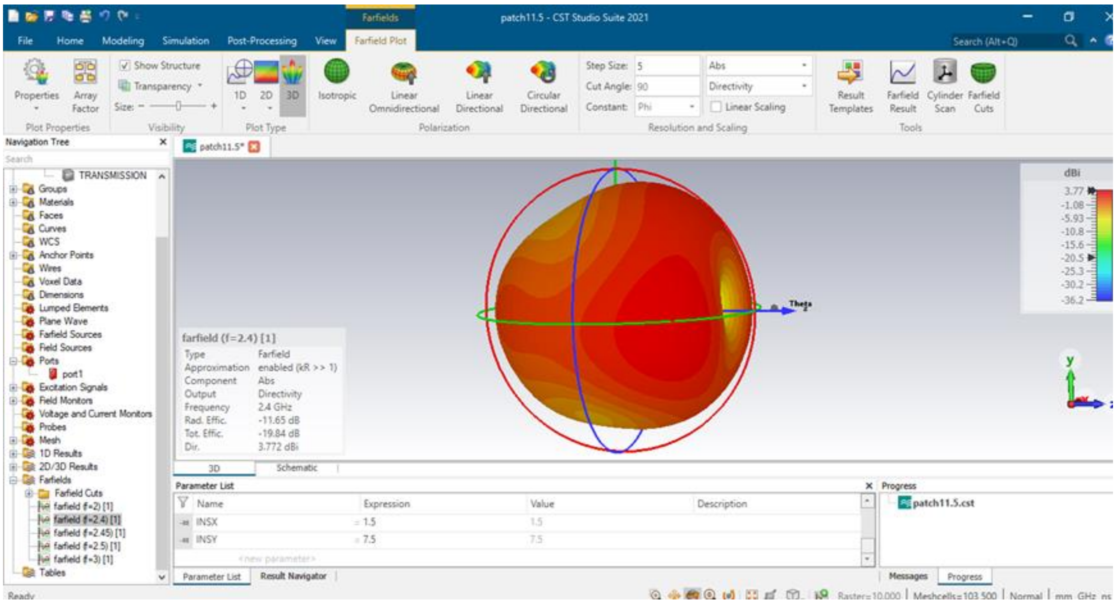
The ground station needs to communicate with the satellite via the transceiver. The transceiver is responsible to collect all the received data from the plugged STM controller and send it to the Graphical User Interface (GUI) via Serial port. In order for the transceiver to send the data, it needs first to get the data from the STM. The STM sends the data to the transceiver via SPI (serial peripheral interface). Lastly, the transceiver sends the data to the GUI via Serial port. The transceiver has a trigger event that occurs depending if the GUI sent any command to the satellite. The transceiver receives the message and send it to the STM via SPI. Lastly, the STM sends the message to the satellite via antenna.

10.2.7 Antenna

The antenna design can be accessed from:

https://www.aecenar.com/index.php/companies/as-comsat/rf-2-4ghz-tranceiver-unit-prototype/rf-sys-implementation/antenna-design-rf





10.2.8 On-Board Computer

10.2.8.1 Introduction

The On-Board Computer (OBC) is responsible for collecting the data from the connected units such as Power Unit and Attitude Control System (ACS) Unit then send the collected data to the Transceiver Unit. It also collects other frames from the transceiver, such as CMD_Frame, then forward the data to its corresponding unit. The system uses a raspberry pi as the OBC, and it is coded in python.

Figure 2-41 Raspberry Pi picture

10.2.8.2 OBC Software Design

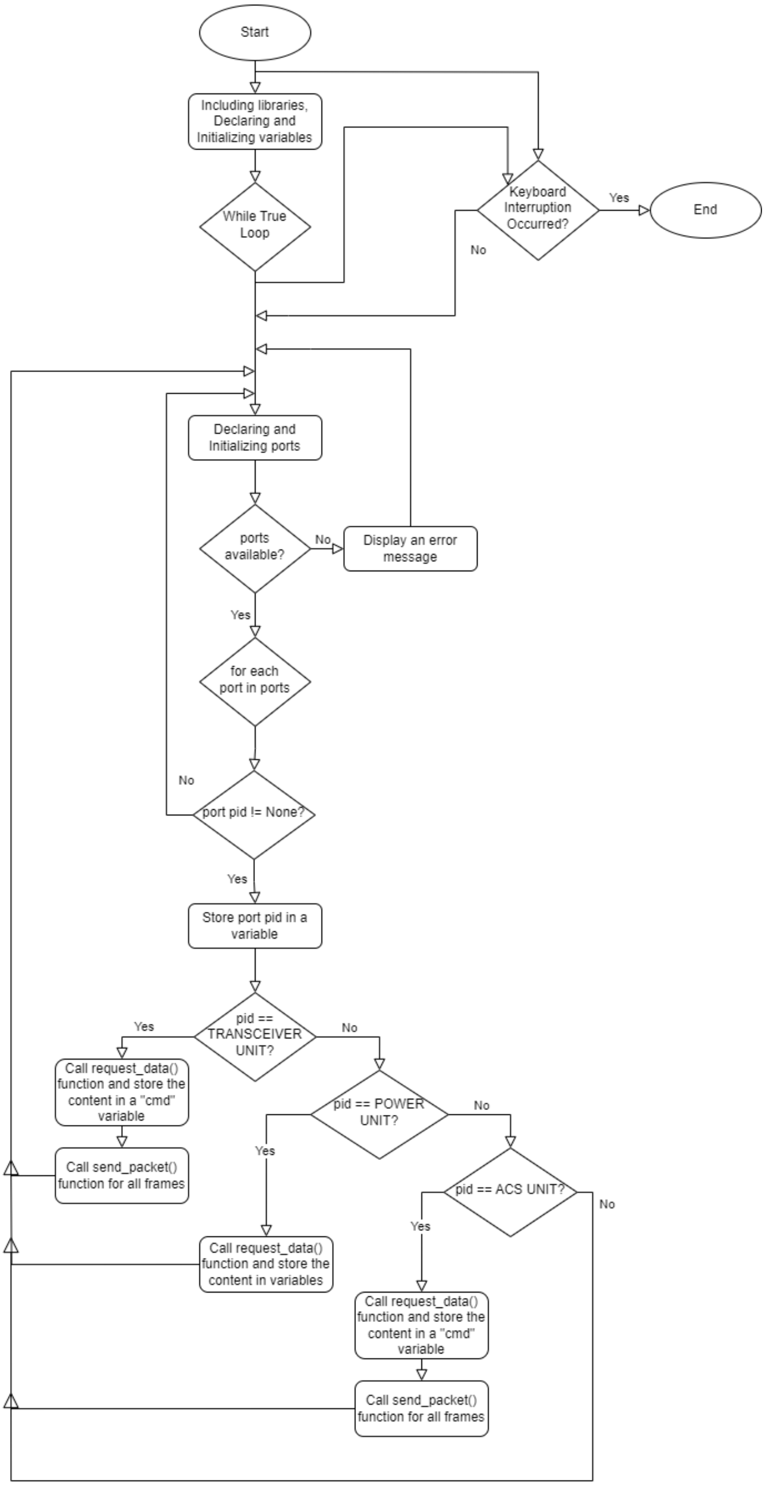


Figure 2-42 OBC Flowchart

The flowchart above explains how the flow process of the following code will work.

```
import serial.tools.list_ports
import serial
import time

PIDs = {
    'TRANSCEIVER UNIT': b'\xFC',
```

```
    'POWER UNIT':       b'\xFD',
    'ACS UNIT':         b'\xF0'
}

FRAMES = {
    'WHO AM I':                 b'\xFF',
    'VER FRAME':                b'\x17',

    'ACS MEASUREMENT FRAME':    b'\x11',
    'ACCELERATION FRAME':       b'\x12',
    'ANG RATE FRAME':           b'\x13',
    'ANGLES FRAME':             b'\x14',
    'POSITION FRAME':           b'\x15',
    'STATUS FRAME':             b'\x16',
    'ACS DIAG FRAME':           b'\x18',

    'POWER MEASUREMENT FRAME':  b'\x20',
    'POWER STATUS FRAME':       b'\x21',
    'POWER DIAG FRAME':         b'\x22',

    'CMD FRAME':                b'\x30',
    'OBC DIAG FRAME':           b'\x31'
}

def request_data(port, request):
    try:
        ser = serial.Serial(port.device, baudrate=115200, timeout=0.001)  # Adjust
baudrate and timeout as needed
        ser.reset_input_buffer()

        ser.write(request)
        response = ser.read(11)  # Read 11 bytes of response
        ser.close()

        if response:
            return response
        else:
            return None

    except serial.SerialException as e:
        print(f"Error opening {port.device}: {e}")
        return None

def send_packet(port, packet):
    try:
        ser = serial.Serial(port.device, baudrate=115200, timeout=1)

        if packet != None:
            ser.write(packet)
```

```
        ser.close()

    except TypeError:
        print("Errorrr")

    except serial.SerialException:
        print("Error Writing to COM port")

def main():
    pid = None

    cmd = trans_ver = bytes(0)
    power_meas = power_stat = power_ver = power_diag = bytes(0)
    acs_meas = raw_acceleration = raw_ang_rate = angles = positions = acs_status
=  acs_version = acs_diag = bytes(0)
    obc_diag = bytes(0)

    try:
        while True:
            ports = list(serial.tools.list_ports.comports())
            new_connected_targets = {}

            now = time.time()
            if ports:
                for port in ports:

                    if port.pid != None:
                        pid = port.pid & 0xFF
                        pid = pid.to_bytes(1)

                        if pid == PIDs['TRANSCEIVER UNIT']:
                            new_connected_targets['TRANSCEIVER UNIT'] = port.device

                            cmd = request_data(port, FRAMES['CMD FRAME'])

                            send_packet(port, power_meas)
                            send_packet(port, power_stat)
                            send_packet(port,power_ver)
                            send_packet(port,power_diag)

                            send_packet(port,acs_meas)
                            send_packet(port,raw_acceleration)
                            send_packet(port,raw_ang_rate)
                            send_packet(port,angles)
                            send_packet(port,positions)
                            send_packet(port,acs_status)
                            send_packet(port,acs_diag)

                        elif pid == PIDs['POWER UNIT']:
```

```
                            new_connected_targets['POWER UNIT'] = port.device

                            power_meas = request_data(port, FRAMES['POWER
MEASUREMENT FRAME'])
                            power_stat = request_data(port, FRAMES['POWER STATUS
FRAME'])
                            power_ver = request_data(port, FRAMES['VER FRAME'])
                            power_diag = request_data(port, FRAMES['POWER DIAG
FRAME'])


                            if power_meas != None:
                                v_batt = (power_meas[2] << 8) | power_meas[3]
                                v_batt = v_batt * (3.3 / 4096)
                                v_batt /= 0.0945

                        elif pid == PIDs['ACS UNIT']:
                            new_connected_targets['ACS UNIT'] = port.device

                            acs_meas = request_data(port, FRAMES['ACS MEASUREMENT
FRAME'])
                            raw_acceleration = request_data(port,
FRAMES['ACCELERATION FRAME'])
                            raw_ang_rate = request_data(port, FRAMES['ANG RATE
FRAME'])
                            angles = request_data(port, FRAMES['ANGLES FRAME'])
                            positions = request_data(port, FRAMES['POSITION
FRAME'])
                            acs_status = request_data(port, FRAMES['STATUS FRAME'])
                            acs_diag = request_data(port, FRAMES['ACS DIAG FRAME'])

                            send_packet(port, cmd)

                                  if raw_acceleration != None:
                               a_res = ((raw_acceleration[8] << 8) |
raw_acceleration[9]) / 100000000.00

                               x_acc = (((raw_acceleration[2] << 8) |
raw_acceleration[3]))

                               y_acc = (((raw_acceleration[4] << 8) |
raw_acceleration[5]))

                               z_acc = (((raw_acceleration[6] << 8) |
raw_acceleration[7]))


                               if x_acc & 0x8000:
                                     x_acc = x_acc - 65536

                               if y_acc & 0x8000:
                                     y_acc = y_acc - 65536
```

```
                              if z_acc & 0x8000:
                                    z_acc = z_acc - 65536

                              x_acc *= a_res
                              y_acc *= a_res
                              z_acc *= a_res

            else:
                print("No connected serial ports found.")

    except KeyboardInterrupt:
        print("exit")


if __name__ == "__main__":
    main()
```

Before starting with the program, the code should import the following libraries:

- serial.tools.list_ports
- serial

Then, the code should contain the PID frames for each unit it will communicating with. The PIDs are the following:

```
PIDs = {
    'TRANSCEIVER UNIT': b'\xFC',
    'POWER UNIT':       b'\xFD',
    'ACS UNIT':         b'\xF0'
}
```

Then, the code should contain all the frame headers that it needs to collect. The Frames are the following:

```
FRAMES = {
    'WHO AM I':                 b'\xFF',
    'VER FRAME':                b'\x17',

    'ACS MEASUREMENT FRAME':    b'\x11',
    'ACCELERATION FRAME':       b'\x12',
    'ANG RATE FRAME':           b'\x13',
    'ANGLES FRAME':             b'\x14',
    'POSITION FRAME':           b'\x15',
    'STATUS FRAME':             b'\x16',
    'ACS DIAG FRAME':           b'\x18',

    'POWER MEASUREMENT FRAME':  b'\x20',
    'POWER STATUS FRAME':       b'\x21',
    'POWER DIAG FRAME':         b'\x22',

```

```
    'CMD FRAME':                b'\x30',
    'OBC DIAG FRAME':           b'\x31'
}
```

After including all these information, the code will start with the "main()" function. The function will start by initializing all the variables that it needs in order to store the data from the requested frames. The following variables need to be initialized:

```
pid = None

    cmd = trans_ver = bytes(0)
    power_meas = power_stat = power_ver = power_diag = bytes(0)
    acs_meas = raw_acceleration = raw_ang_rate = angles = positions = acs_status =
acs_version = acs_diag = bytes(0)
    obc_diag = bytes(0)
```

The code then enters into an infinite loop that will keep executing the program until it detects a "KeyboardInterrupt" exception. The code will initialize and store the ports that are connected by using the imported library into the "ports" variable.

```
ports = list(serial.tools.list_ports.comports())
```

The code then checks if "ports" variable contains any value and is not empty. If the variable is empty, then it will skip the iteration and keeps looping until "ports" store a value. Then the code loops for each "port" that is in "ports" variable in order to check on each port and if it's the required port needed. Then the code check if the "port pid" is not "None", it will store the value in "pid" variable and convert it to a byte.

```
while True:
            ports = list(serial.tools.list_ports.comports())
            new_connected_targets = {}

            now = time.time()
            if ports:
                for port in ports:

                    if port.pid != None:
                        pid = port.pid & 0xFF
                        pid = pid.to_bytes(1)
```

After taking the pid of the port, the code check the pid if it is equal to one of the specified PIDs at the beginning of the code. If it is the "TRANSCEIVER UNIT", it will call the function "request_data()" for the "CMD FRAME", that is responsible for requesting data frames from a given port and return the value of the frame as an array of bytes, and store them in a variable "cmd". It will also call the function "send_packet()", that is responsible to send the given data to the given port, and send all the frames "power_meas, power_stat, power_ver, power_diag, acs_meas, raw_acceleration, raw_ang_rate, angles, positions, acs_status, acs_diag" to the TRANSCEIVER UNIT.

```
def request_data(port, request):
    try:
```

```
        ser = serial.Serial(port.device, baudrate=115200, timeout=0.001)  # Adjust
baudrate and timeout as needed
        ser.reset_input_buffer()

        ser.write(request)
        response = ser.read(11)  # Read 11 bytes of response
        ser.close()

        if response:
            #if response[0].to_bytes(1) == request:
            return response
        else:
            return None

    except serial.SerialException as e:
        print(f"Error opening {port.device}: {e}")
        return None

def send_packet(port, packet):
    try:
        ser = serial.Serial(port.device, baudrate=115200, timeout=1)

        if packet != None:
            ser.write(packet)
        ser.close()

    except TypeError:
        print("Errorrr")

    except serial.SerialException:
        print("Error Writing to COM port")


if port.pid != None:
                        pid = port.pid & 0xFF
                        pid = pid.to_bytes(1)

                        #print(PIDs.items())
                        if pid == PIDs['TRANSCEIVER UNIT']:
                            new_connected_targets['TRANSCEIVER UNIT'] = port.device

                            cmd = request_data(port, FRAMES['CMD FRAME'])

                            send_packet(port, power_meas)
                            send_packet(port, power_stat)
                            send_packet(port,power_ver)
                            send_packet(port,power_diag)

                            send_packet(port,acs_meas)
```

```
                            send_packet(port,raw_acceleration)
                            send_packet(port,raw_ang_rate)
                            send_packet(port,angles)
                            send_packet(port,positions)
                            send_packet(port,acs_status)
                            send_packet(port,acs_diag)
```

If it is the "POWER UNIT", it will call the function "request_data()" to request 4 headers, "POWER MEASUREMENT FRAME, POWER STATUS FRAME, VER FRAME, POWER DIAG FRAME", and store them in a variables. It will also check for the "power_meas" frame if it is not equal to None, then it will calculate a formula for the "v_batt" variable to make sure that the transceiver receives the accurate value of the battery voltage.

```
elif pid == PIDs['POWER UNIT']:
                            new_connected_targets['POWER UNIT'] = port.device

                            power_meas = request_data(port, FRAMES['POWER 
MEASUREMENT FRAME'])
                            power_stat = request_data(port, FRAMES['POWER STATUS 
FRAME'])
                            power_ver = request_data(port, FRAMES['VER FRAME'])
                            power_diag = request_data(port, FRAMES['POWER DIAG 
FRAME'])

                            if power_meas != None:
                                v_batt = (power_meas[2] << 8) | power_meas[3]
                                v_batt = v_batt * (3.3 / 4096)
                                v_batt /= 0.0945
```

If it is the "ACS UNIT", it will call the function "request_data()" to request 7 headers, "ACS MEASUREMENT FRAME, ACCELERATION FRAME, ANG RATE FRAME, ANGLES FRAME, POSITION FRAME, STATUS FRAME, ACS DIAG FRAME", and store them in a variables. It will also call the "send_packet()" function to send the "cmd" variable to the ACS unit. It will also use a formula to convert the "raw" data received from the Accelerometer into accurate data. It will also check if the values received after conversion are negative value by checking the first bit of the data (signed value: 1 for negative value, 0 for positive value) using the command "&8000" which can be represented as "1000000000000000". This method will check if the first bit is 1 (means negative), it will subtract 65536 which is the maximum number a 16 bit integer can reach to convert the value accordingly.

```
elif pid == PIDs['ACS UNIT']:
                            new_connected_targets['ACS UNIT'] = port.device

                            acs_meas = request_data(port, FRAMES['ACS MEASUREMENT 
FRAME'])
                            raw_acceleration = request_data(port, 
FRAMES['ACCELERATION FRAME'])
```

```
                            raw_ang_rate = request_data(port, FRAMES['ANG RATE 
FRAME'])
                            angles = request_data(port, FRAMES['ANGLES FRAME'])
                            positions = request_data(port, FRAMES['POSITION 
FRAME'])
                            acs_status = request_data(port, FRAMES['STATUS FRAME'])
                            acs_diag = request_data(port, FRAMES['ACS DIAG FRAME'])

                            send_packet(port, cmd)

                            if raw_acceleration != None:
                              a_res = ((raw_acceleration[8] << 8) | 
raw_acceleration[9]) / 100000000.00

                              x_acc = (((raw_acceleration[2] << 8) | 
raw_acceleration[3]))
                              y_acc = (((raw_acceleration[4] << 8) | 
raw_acceleration[5]))
                              z_acc = (((raw_acceleration[6] << 8) | 
raw_acceleration[7]))

                              if x_acc & 0x8000:
                                    x_acc = x_acc - 65536

                              if y_acc & 0x8000:
                                    y_acc = y_acc - 65536

                              if z_acc & 0x8000:
                                    z_acc = z_acc - 65536

                              x_acc *= a_res
                              y_acc *= a_res
                              z_acc *= a_res
```

10.2.9 Payload Transmitter

Same as TT&C transceiver

10.2.10 Power Management Unit (PMU)

10.2.10.1 Introduction

The PMU is designed to charge the onboard batteries via solar panels. In addition, it is responsible for reporting the power status of the system such as: charging state, overvoltage, undervoltage, current consumption, and voltage levels.

10.2.10.2 Bill of Material

10.2.10.3 PMU Hardware Design

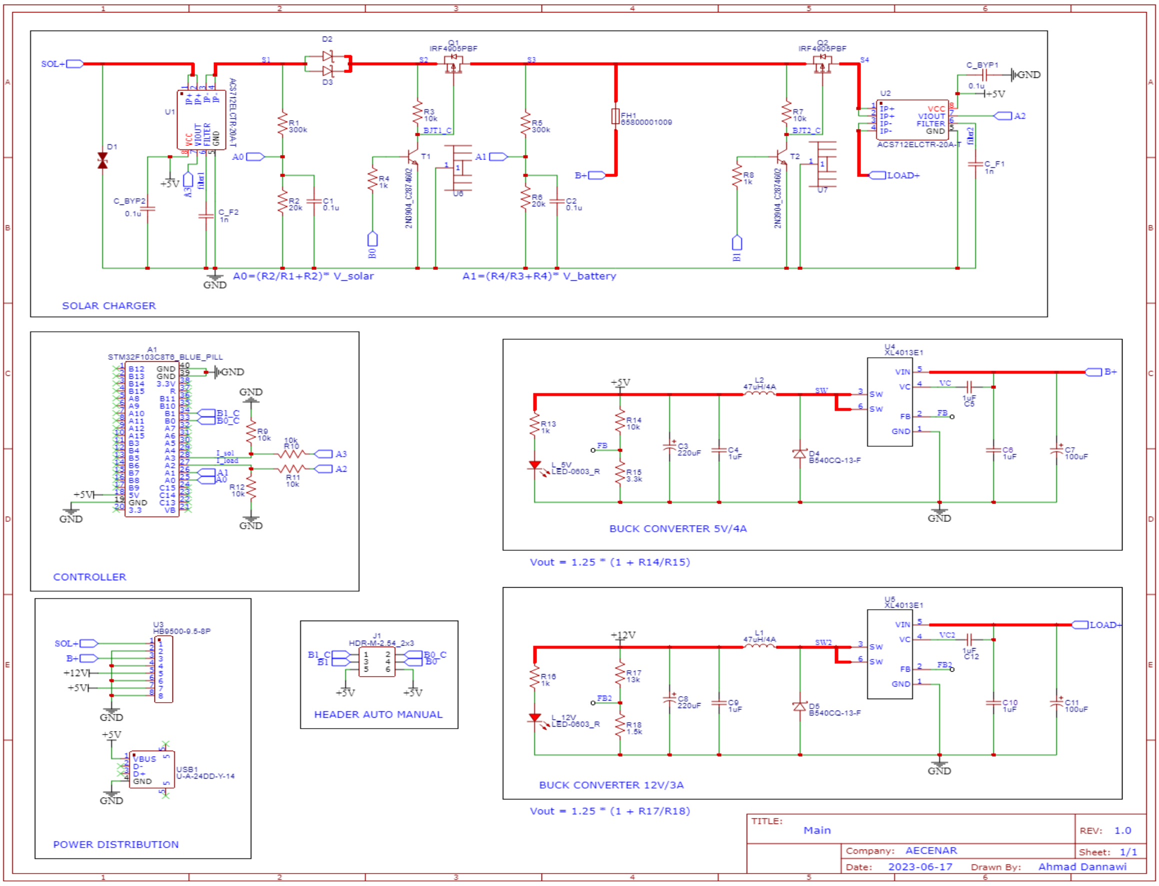


Figure 2-43 PMU schematic diagram

Refer to this to access the power management unit development https://docs.google.com/document/d/10I53u1CAf4NAXECpuC_PeLMeE_lChAot/edit#heading=h.gjdgxs

10.2.10.4 PMU Software Design

Refer to this to access the power management unit software development https://docs.google.com/document/d/10I53u1CAf4NAXECpuC_PeLMeE_lChAot/edit#heading=h.gjdgxs

10.3 TT&C Ground Station

10.3.1 Introduction

This chapter will introduce the steps of developing the graphical user interface (GUI) and explain how the classes in the GUI interact with each other. A sequence diagram will be used to show how the data is being transferred between the forms and the use case diagram will illustrate the actions that can be done by the user. Moreover, each function in the code will be explained.

10.3.2 GUI Sequence Diagram

Figure 3-1 below shows the time sequence of the GUI behavior.

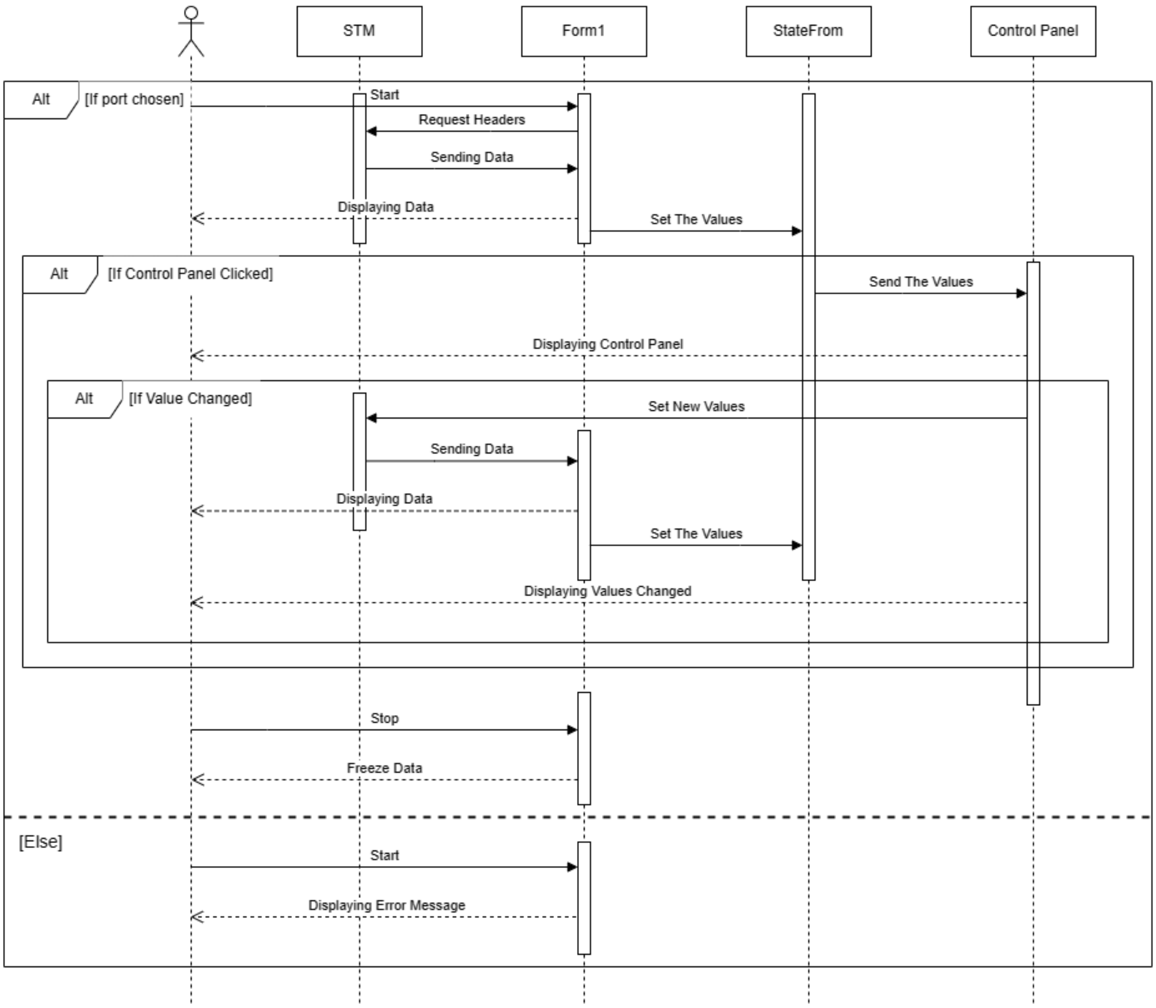


Figure 3-1- System Sequence Diagram

The user will have to choose a port from the combo box and click the start button. Then the GUI will request headers from the STM that will send the frames to be displayed in the text boxes in the first form of the GUI. The values of Phi, Theta, Psi angles and Altitude will be set in the Stateform to be used in the Control Panel. In the control panel, the user can view the angles of the satellite and change them. When a value gets changed it will be sent directly to the STM. The other steps will be repeated as previous in order to view the updated values on both forms.

The stop button will freeze the values that are being displayed on the form. If an invalid port was chosen, an error message will be displayed to the user.

10.3.3 GUI Use Cases

In Figure 3-2, all the actions that can be performed by the user in the GUI are represented.

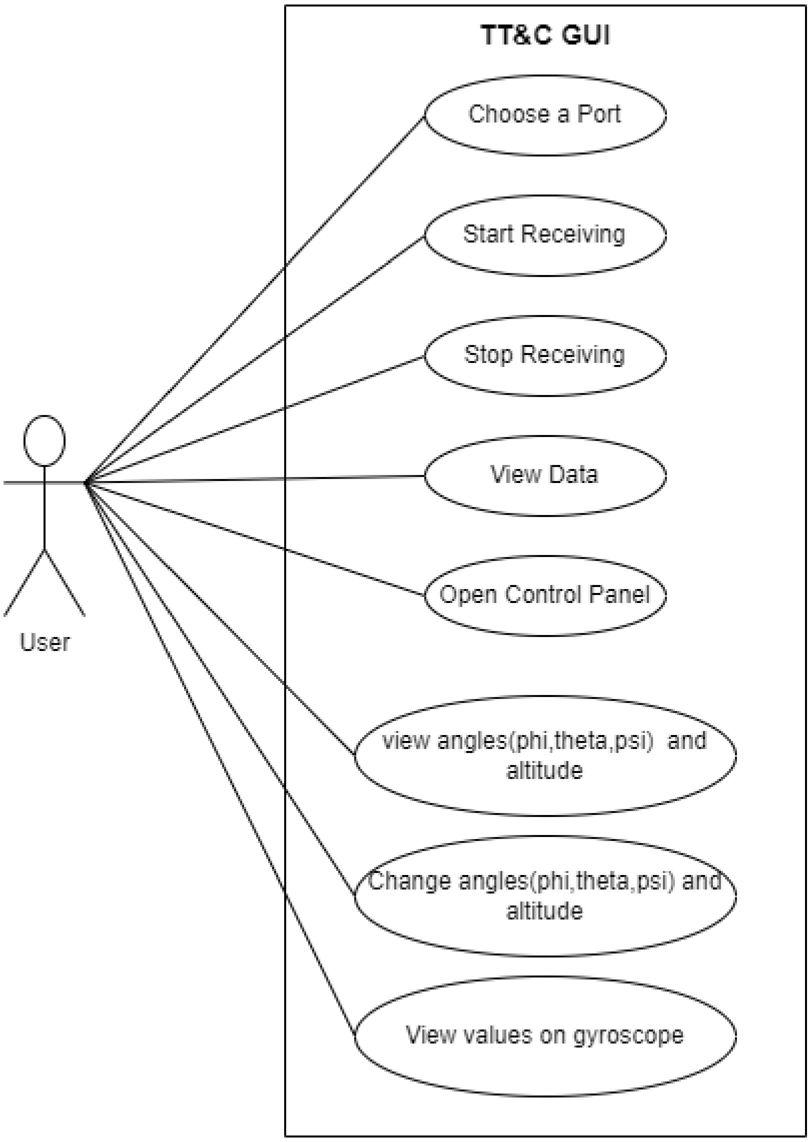


Figure 3-2- System Use Case Diagram

10.3.4 Interface between GUI and Ground Station TT&C Transceiver

The transceiver of the ground station is similar to the TT&C Transceiver. Ground Station (GS) Transceiver is used to transmit control data from the GUI to the satellite transceiver and to deliver received data from TT&C transceiver to be displayed on the GUI Forms.

The data received by the GS transceiver consists of frames with the structure as shown in Table 3-1 below.

Frame Name	ID	Length	Content								
			Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9
Power Meas. Frame	0x20	9	v_batt	v_batt	v_solar	v_solar	I_load	I_load	I_solar	I_solar	SOC
Power Status Frame	0x21	1									
Power Diag. Frame	0x22	1									
Measurement Frame	0x11	6	Temp	Temp	Pressure	Pressure	Sun	Sun			
Acceleration Frame	0x12	6	x_acc	x_acc	y_acc	y_acc	z_acc	z_acc			
Angular Rate Frame	0x13	6	phi_dot	phi_dot	theta_dot	theta_dot	psi_dot	psi_dot			
Angles Frame	0x14	6	phi	phi	theta	theta	psi	psi			
Position Frame	0x15	8	x	x	x	y	y	y	z	z	z
Status Frame	0x16	1	RF[4:5] V4[3:4] V3[2:3] V2[1:2] V1[0:1]								
ACS Diag. Frame	0x18	1	SPI[0:1]								
CMD Frame	0x30	9	Phi Set Point	Phi Set Point	Theta Set Point	Theta Set Point	Psi Set Point	Psi Set Point	Altitude Set Point	Altitude Set Point	V4[3:4] V3[2:3] V2[1:2] V1[0:1]
OBC Diag. Frame	0x31	1	ACS[2:3] PWR[1:2] TR[0:1]								
Who Am I Frame	0xFF										
Version Frame	0x17	2	sw_ver								

Table 3-1- Schedule Table

Each frame consists of 11 bytes and contains specific data about the state of the satellite. All the frames in the table are being received by the GS transceiver and displayed in the GUI except for the CMD Frame (ID: 0x30). This frame is the one being transmitted from the control panel of the graphical user interface to the GST and then to the TT&C transceiver.

10.3.5 GUI Design

10.3.5.1 Requirements

Satellite to Ground Station:

1. Battery Voltage.
2. Solar Voltage.
3. Solar Current.
4. Load Current.
5. State of Charge (SOC).
6. Temperature.
7. Sun Exposure.
8. Pressure.
9. Software Version.
10. Cube SAT ID.
11. Radio Status.
12. Valves Status (4 Valves).
13. Acceleration.
14. Position.
15. Angular Rate.
16. Angles.

Ground Station to Satellite:

1. Position Change.
2. Manual Control.

10.3.5.2 Building the Interfaces

<u>Form 1:</u> AS-COMSAT_1 TT&C:

This form will allow the user to choose a port and navigate through the different forms of the GUI.

To do that:

- Add a combo box and change its name to “AvailablePorts”.
- Add a button and set its name and Text to “Start”.
- Add a button and set its name and Text to “Stop”.
- Add a button and set its name to “powerUnitBtn” and Text to “Power Unit”.
- Add a button and set its name to “acsUnitBtn” and Text to “ACS Control Unit”.
- Add a button and set its name to “futureBtn” and Text to “Control”.
- Add from the toolbox “serialPort” hidden component.
- In the properties of the serialPort --> ReceivedBytesThreshold = 2. (2 Bytes will be received for the value at a time).
- Add a picture box to generate the graphics of the “Satellite to PC” illustration.
- Add from the toolbox “timer1” hidden component and set its name to “pictureTimer” and interval to 500 ms.
- Add another timer call it "data_request_timer" with interval 5 ms.

The form should look like Figure 3-3 below.

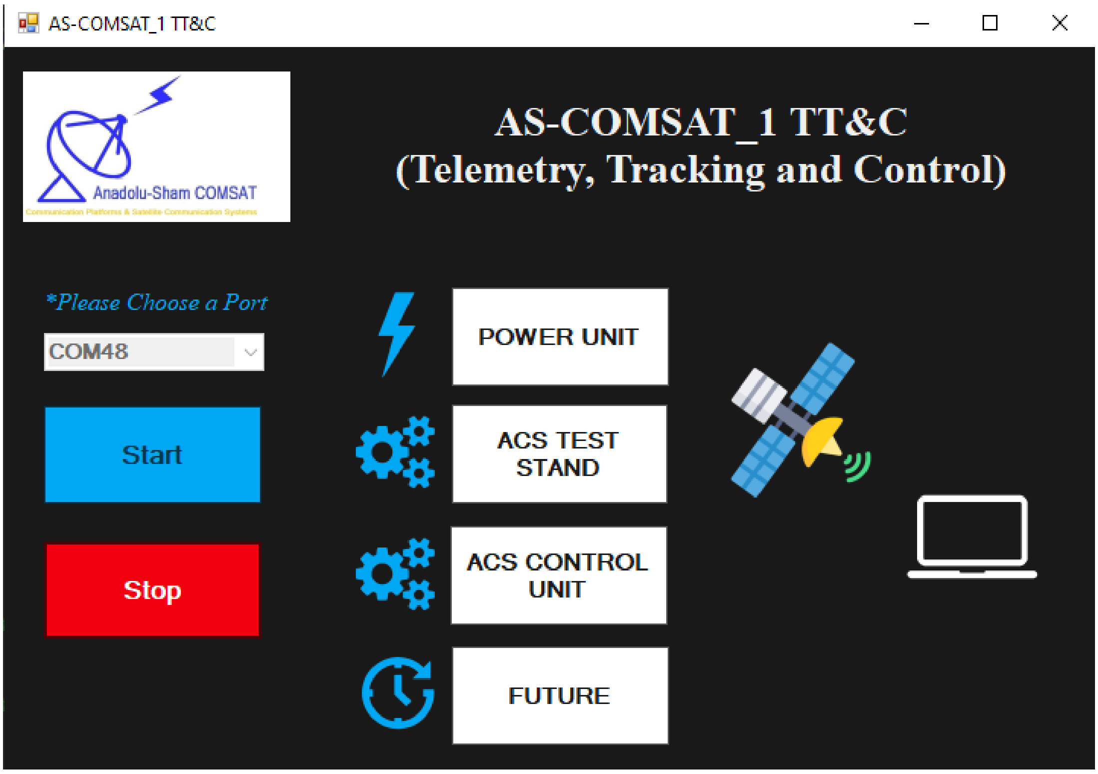


Figure 3-3- Main Form

Form 2: Power Unit

This form contains tall the power variables related to the satellite. To build it:

- Place 5 textboxes to display the 5 power values (Battery voltage, solar voltage, solar current, load current, state of charge).
- Place 2 textboxes to display cubeSAT ID, software version.

All the values in this form will be taken from the stateForm. It should look like Figure 3-4.

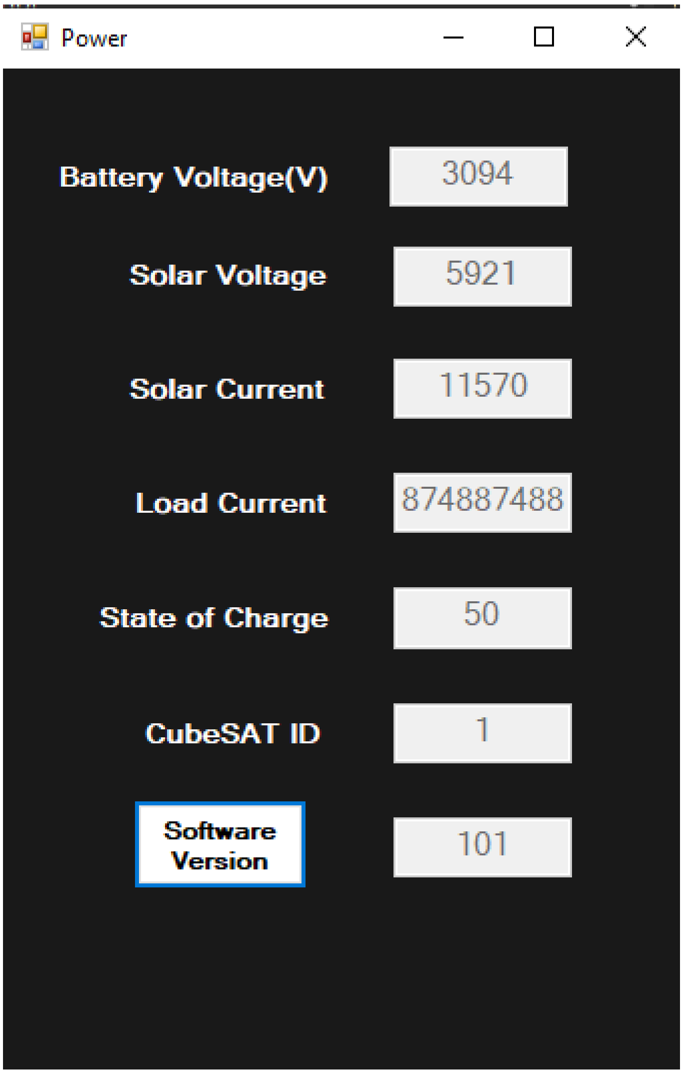


Figure 3-4- Power Unit Form

Form 3: ACS Control Unit

This form allows the user to view angles Phi, Theta and Psi and the altitude graphically. It also contains the states of the valves. All the mentioned variables can be modified by the user in this form. And the new values will be sent to the STM.

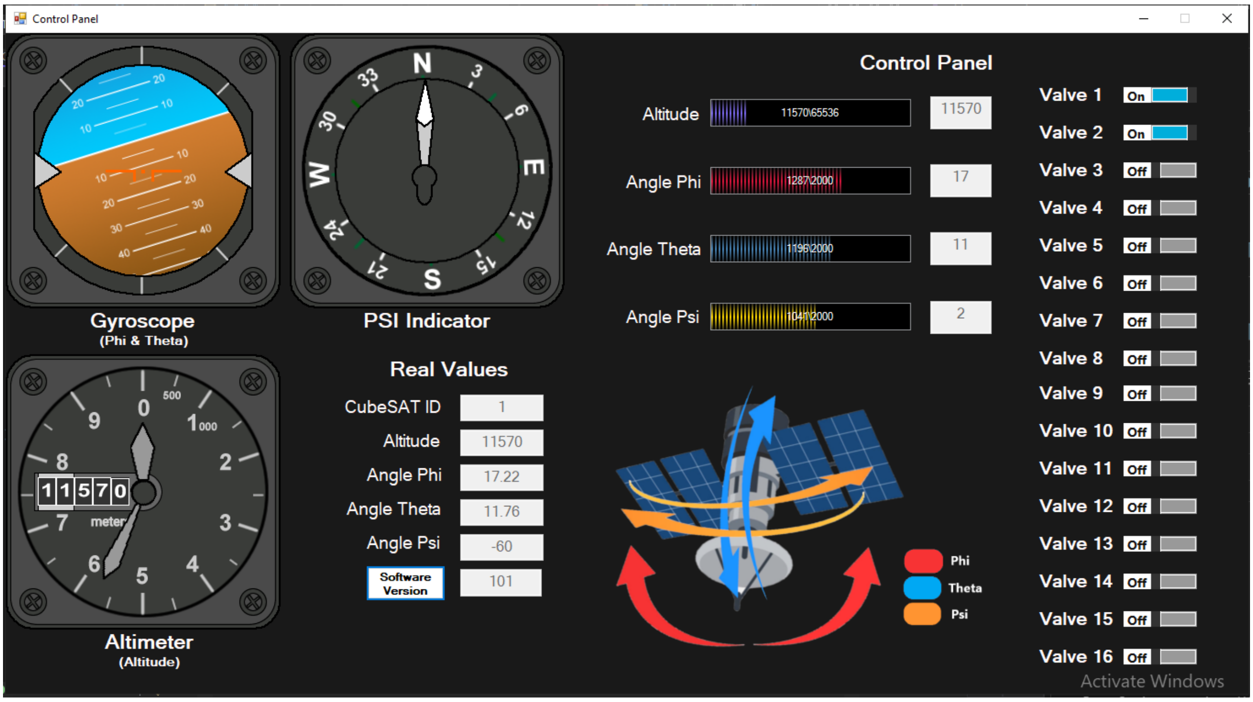


Figure 3-5- Control Panel Form

The meters represented in Figure 3-5 are drawn and manipulated by code. To change any of the angles, the user can scroll the slider left or right, the data will be sent to the STM to be sent back to the GUI and seen in real time on the meters. To change the value of the valves, the switch could be toggled on or off.

Building this form requires textboxes, toggles and sliders. The Software version button displays the version on request. Moreover, the outlines of the meters are pre-drawn pictures which are then called and controlled in the code.

Form 4: Control

This form contains the rest of the control panel variables and the state of the valves.

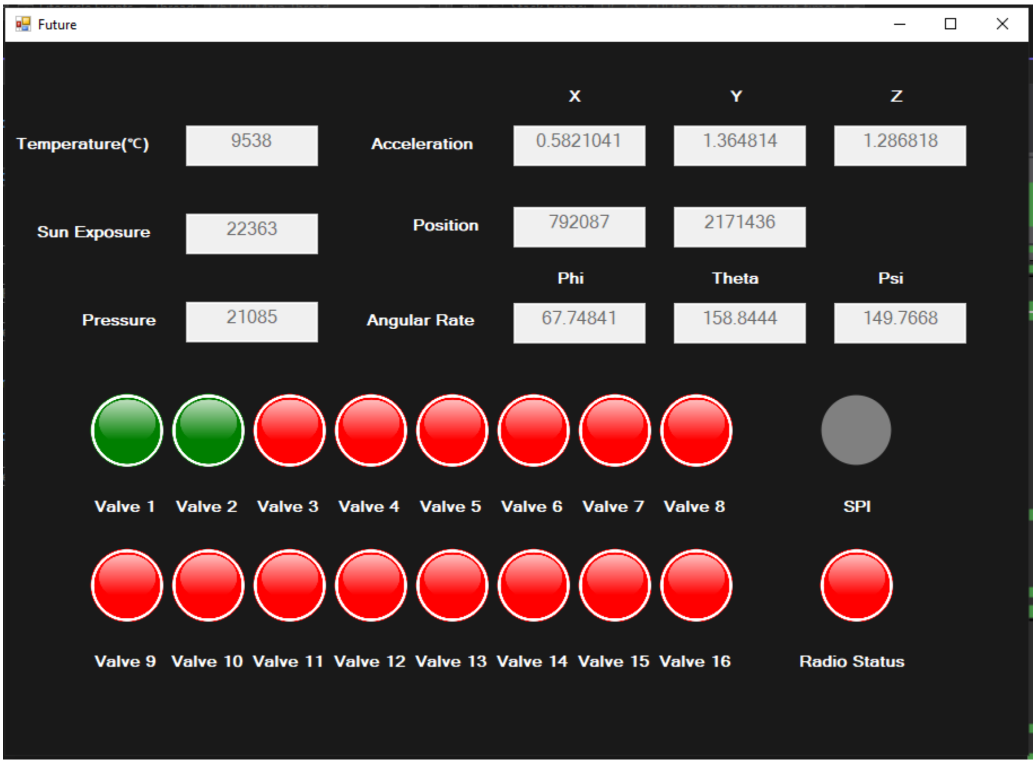


Figure 3-6- Control

<u>Form 5:</u> Propulsion System:

In the Refresh_Timer_Tick event, the following controls are updated: Pressure1_textBox, Pressure2_textBox, Pressure3_textBox, Flow_Meter1_textBox, Flow_Meter2_textBox, Valve_Opening_VM1_textBox, and Valve_Opening_VM2_textBox. Additionally, within this event, if the valve is turned open, the CMD_ON_OFF_Valve1_button's text is set to "CMD ON" and its background color is changed to green; otherwise, it is set to red. In the CMD_ON_OFF_Valve1_button_Click event, clicking the button toggles the valve status between "CMD_ON" and "CMD_OFF".

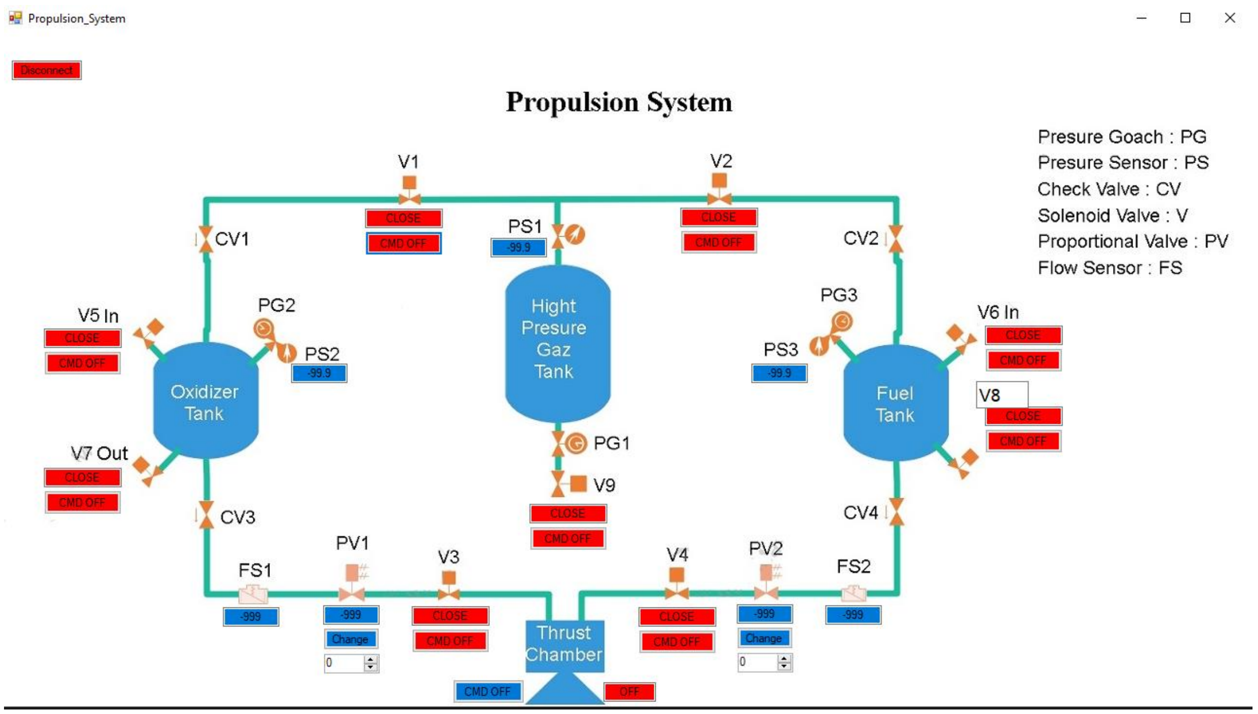


Figure 3-7- Propulsion System

In the latest update (20-02-2024), we added valve codes 1 to 9 into the STATUS_FRAME to keep track of them better. We also changed the C code to work better with C# code. These changes make it easier for different parts of the system to communicate with each other. We tested the changes to make sure they work well and don't cause any problems.

After opening the "acs_control_unit" and the "propulsion_system," all valves are initially set to the "OFF" position. When a valve, such as valve 1, 5, or 9, is opened in the "acs_control_unit," it should also be opened in the "propulsion_system." Similarly, when a valve is opened in the "propulsion_system," it should be reflected as open in the "acs_control_unit." This ensures synchronization between the two systems, providing consistent valve status updates across both components. Overall, these updates make the system more efficient and reliable.

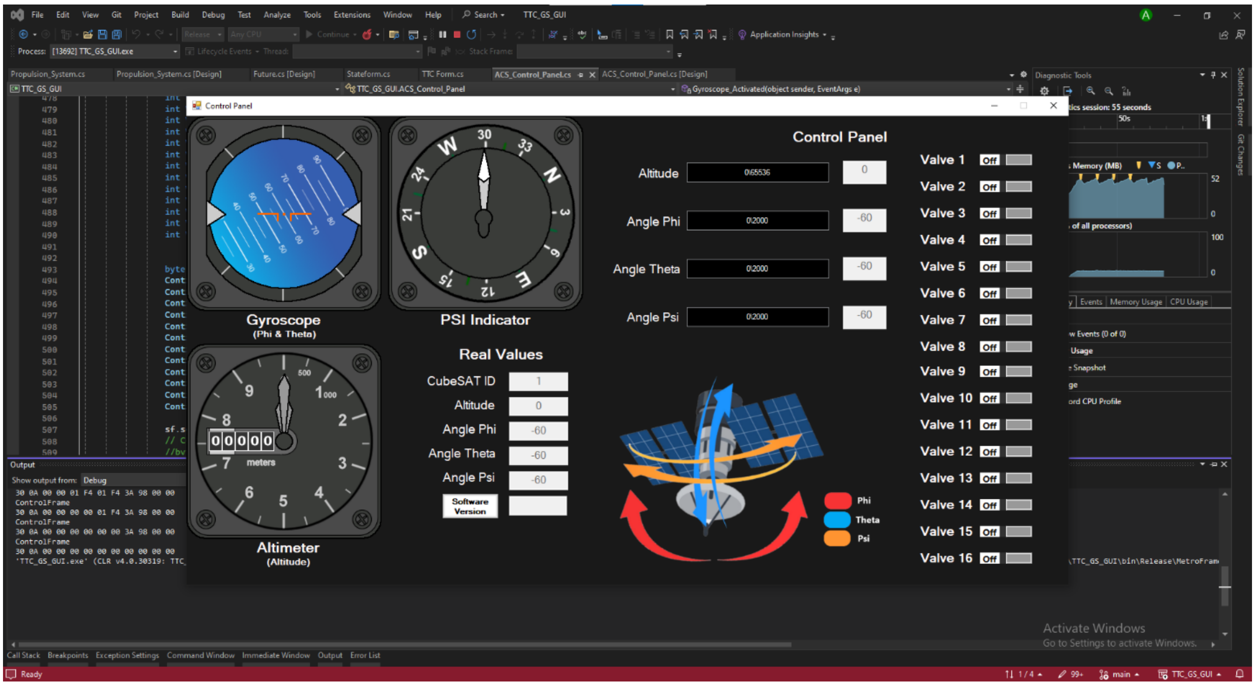


Figure 3-8- ACS Control Unit Valve OFF

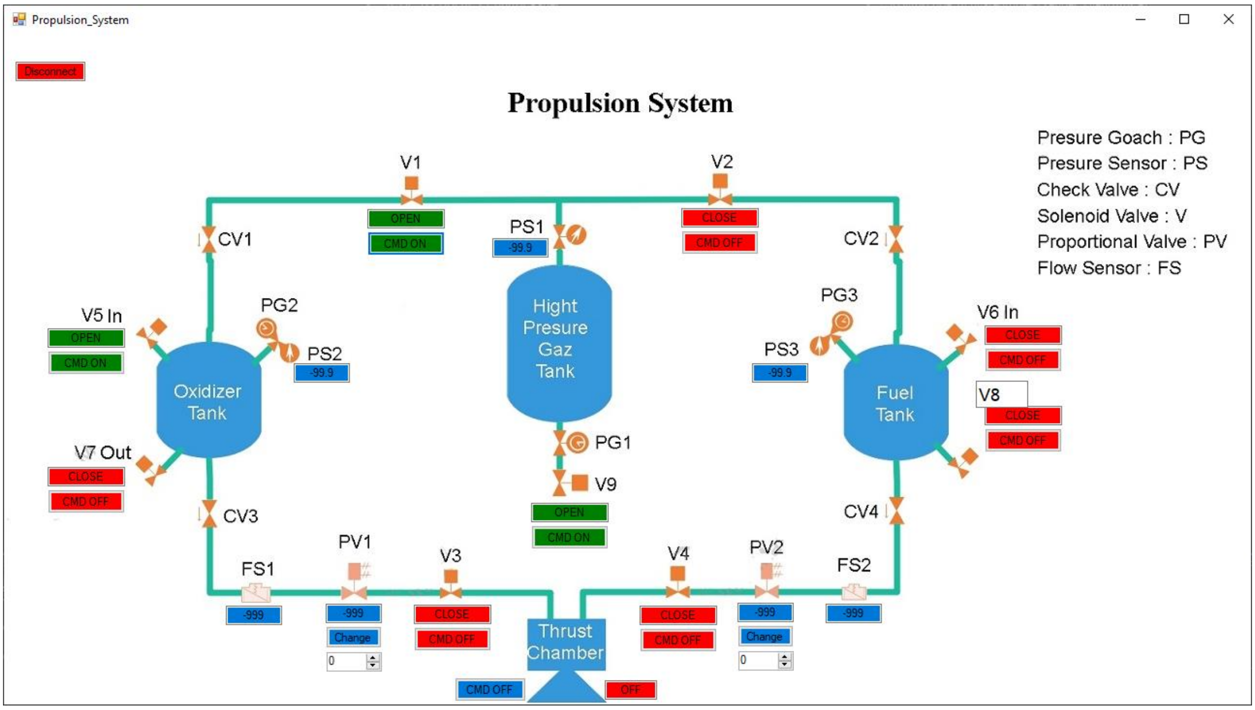


Figure 3-9- ACS Test Stand ON

10.3.6 GS Transceiver Functions

- **CDC_Receive_FS:** when the STM receives the headers requested by the GUI, it fills the data in the buffer according to the request and calls the "CDC_Transmit_FS" function.
- **CDC_Transmit_FS:** this function transmits the data back to the GUI.

10.3.7 GUI Functions

The code consists of 4 forms and a stateForm class.

The role and behavior of each of these forms will be explained below.

<u>TTC_Form:</u>

- **Enum Frames_ID_t:** variable containing the frames IDs matching with the transceiver IDs.
- **ttcForm:** initialize the components of the Form and calls an instance of the stateForm class.
- **ttcForm_Load:** the first function that gets activated when the app starts.
  - Calls "InitializePortWatcher" function.
  - Calls "PopulateAvailablePorts" function.
  - Controls the "Satellite to PC" illustration.
  - Starts the "data_request_timer", which is the timer used to time the requests to the STM.
  - Disables all the buttons except the start button.
- **<u>Port Connectivity Check Region:</u>** the functions found in this region are to automatically gather COM ports connected to the GUI.
  - **InitializePortWatcher:** it monitors the serial port of the PC to check when a new device is connected.
  - **PortWatcher_EventArrived:** when a port is connected it adds it to the comboBox by calling the "PopulateAvailablePorts" function.
  - **PopulateAvailablePorts:** Collects all the COM ports and adds them to the comboBox.
- **<u>Serial Data Region:</u>** this region is to receive and organize the data sent by the STM.
  - **serialPort_DataReceived:** generates an array of bytes to hold the received data and calls ProcessReceivedData function.
  - **ProcessReceivedData:** this function separates the data according to each frame ID. It stores each value in a variable and sends them to the stateForm.
- **<u>Start_Stop_Buttons Region:</u>**
  - **Start_Click:** this function gets triggered when the Start button is clicked.
    - It takes the selected port in the comboBox.
    - Enables all the buttons and disables itself.

    - It opens the serial port according the comboBox selection.
    - Calls serialPort_DataReceived function to start receiving data.
  - **Stop_Click:** this function gets triggered when the Stop button is clicked.
    - Disables all the buttons and enables the Start button.
    - Stops the data_request_timer
    - Disconnects the serial port.
- **ShowErrorMessageAsync** and **error_Handler** functions are responsible for the error messages that appear when an error occurs.
- **New Form Openings Region:** the functions in this region open the different forms of the GUI.
  - **powerUnitBtn_Click:** gets triggered when the "Power" button is clicked. It opens the power form as a thread.
  - **acsUnitBtn_Click:** gets triggered when the "ACS Control Unit" button is clicked. It opens the ACS_Control_Panel as a thread.
  - **futureBtn_Click:** gets triggered when the "Control" button is clicked. It opens the future form as a thread.

Power Form:

- **Power_Load:** starts the powerTimer when the form is loaded.
- **Soft_Ver_Click:** triggered when "Software Version" button is clicked and gets the software version from the stateForm to be displayed in the textbox.
- **powerTimer_Tick:** gets triggered on each time the timer duration elapse to get the power values from the stateForm and display them in textboxes.

ACS_Control_Panel:

- The pictures of the meters are called at the top of this form each in a variable.
- **ACS_Control_Panel:** initializes the components of this form, set the style of the drawings to be done later.
- **Control_Panel_Load:** this function gets all the values that will be used in the form (phi, theta, psi, altitude, and valves status) from the stateForm and place them in the corresponding components.
- **OnPaint:** this is a built in function that paints the graphics on the form by using the images declared at the top and controls the images through a specific scale provided for each value. It also calls the "ProcessAngles" function.

- **ProcessAngles:** this function controls the Pitch, Roll, and Yaw angles orientation and sets the values in the textboxes.
- **Drawing Functions Region:**
    - **ScrollCounter:** to control the altitude displayed number.
    - **RotateImage:** to control the arrows in altitude and the circular motion of the Psi meter.
    - **InterpolPhyToAngle:** specify the limits of each angle.
    - **RotateAndTranslate:** control the phi/theta meter.
- **Control Slider Value Changing Region:**
    - **ControlAltitude_ValueChanged, ControlPhi_ValueChanged, ControlTheta_ValueChanged, ControlPsi_ValueChanged** all these functions have similar functionalities. They get triggered when the corresponding slider value changes and call the "ControllingData" function.
    - **ControllingData:** this function is responsible for instantly sending the new data from the sliders or toggles to the STM.

Propulsion System:

- Refresh_Timer_Tick:
- Pressure1_textBox, Pressure2_textBox, Pressure3_textBox
- Flow_Meter1_textBox, Flow_Meter2_textBox
- Valve_Opening_VM1_textBox, Valve_Opening_VM2_textBox
- if "VALVE turn " open ==> CMD_ON_OFF_Valve1_button. Text = "CMD ON" ==> the background of the button set to green, else set to red;
- CMD_ON_OFF_Valve1_button_Click: when we click the button, we can change the status of the valve on to "CMD_ON" or "CMD_OFF".

Future Form: this form has a similar functionality as the Power Form.

10.3.8 GUI Code

https://aecenar.com/index.php/downloads/summary/13-temo-space-communication/1568-ttc-gs-gui-v-200224-acs-test-stand-gui-integrated-in-tt-c-gui-full-connected-version

10.4 APPENDIX

Table 4-1 Utilized software packages for designing, simulating, coding, and ordering PCBs

Tool	Description	Website
EasyEDA Pro	Design schematics and PCBs	https://easyeda.com/
LTspice	Simulate RF and other electronic circuits.	https://www.analog.com/en/design-center/design-tools-and-calculators/ltspice-simulator.html
STM32CubeIDE (1.12.1)	Write the code for STM32 microcontrollers	https://www.st.com/en/development-tools/stm32cubeide.html
STM32 Programmer (2.14)	Program the STM32 microcontrollers	https://www.st.com/en/development-tools/stm32cubeprog.html
Smith Chart (V4.1)	Design impedance matching networks	https://smith1.software.informer.com/4.1/
JLCPCB	To order PCBs	https://jlcpcb.com/
LCSC	To order PCB parts	https://www.lcsc.com/

10.5 STM32 IDE configuration and Code Upload

First, we create a new STM32 Project

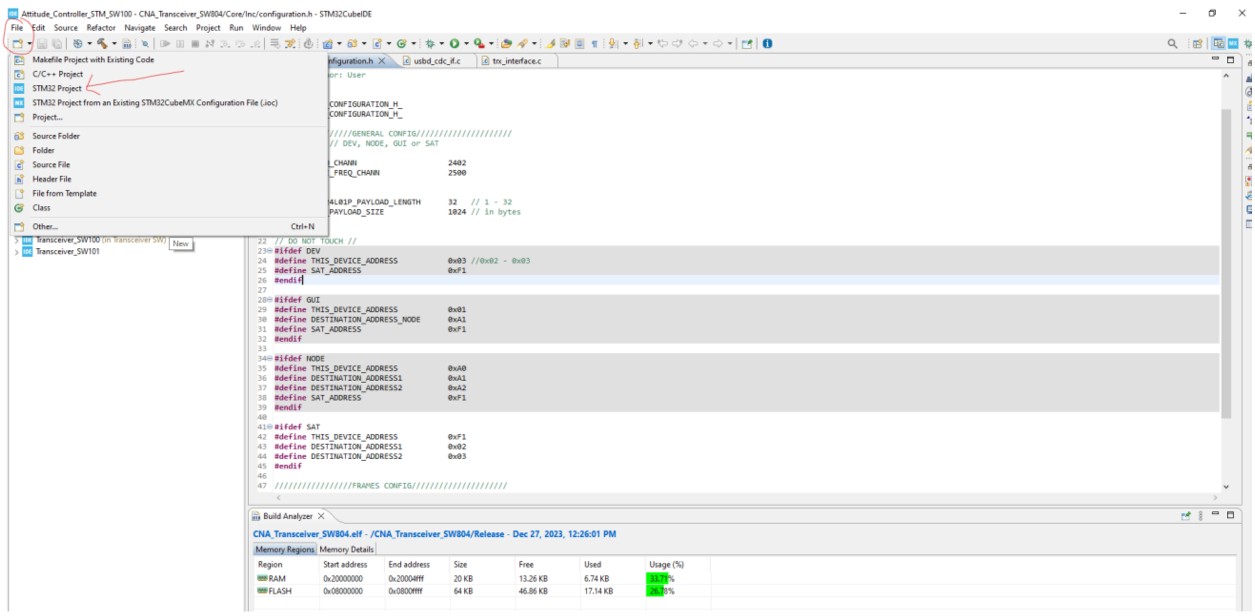


Figure 5-1- Create new STM32 Project

Next, we choose our STM model and click next button

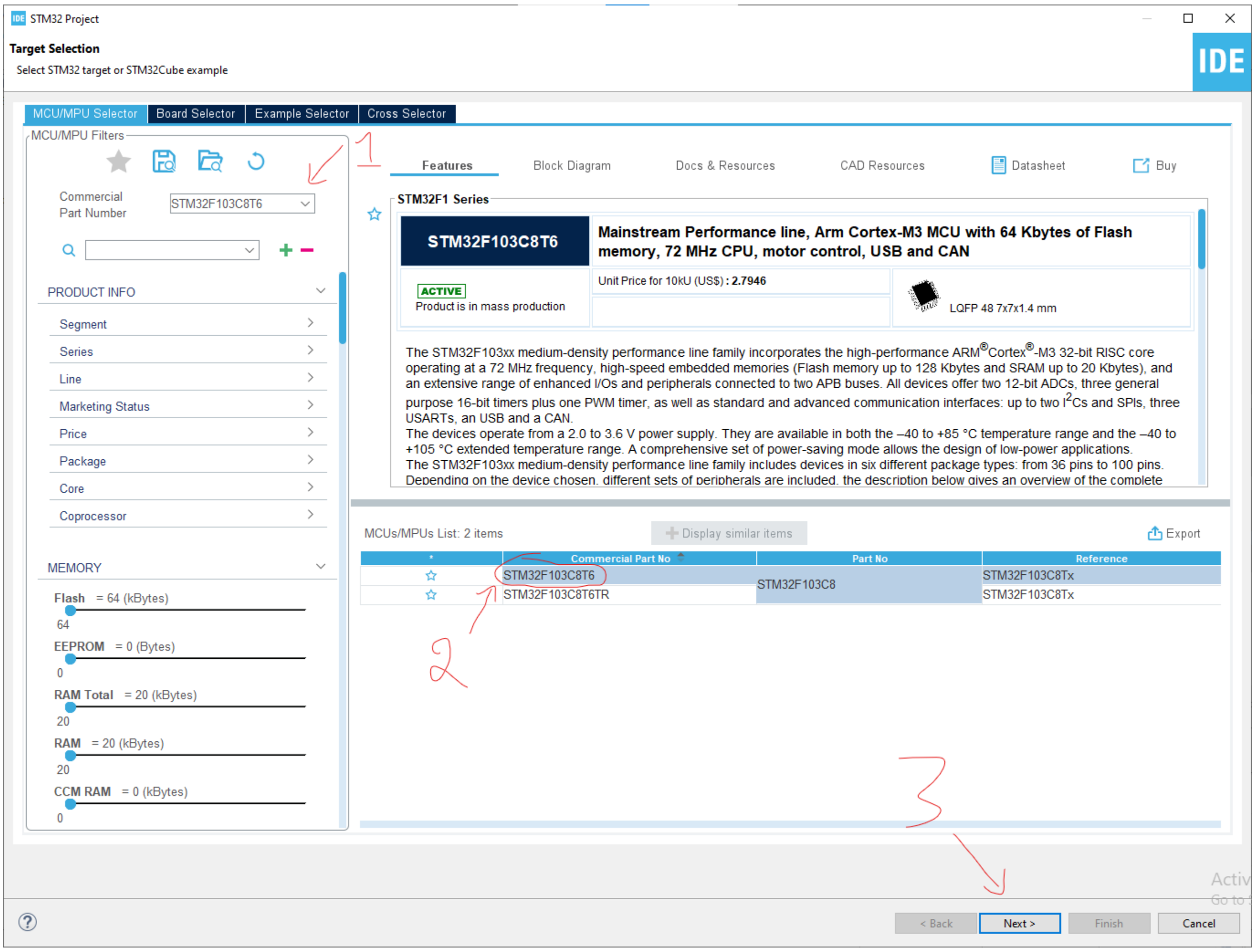


Figure 5-2- Choose STM model

Lastly, we give our project a name and click Finish

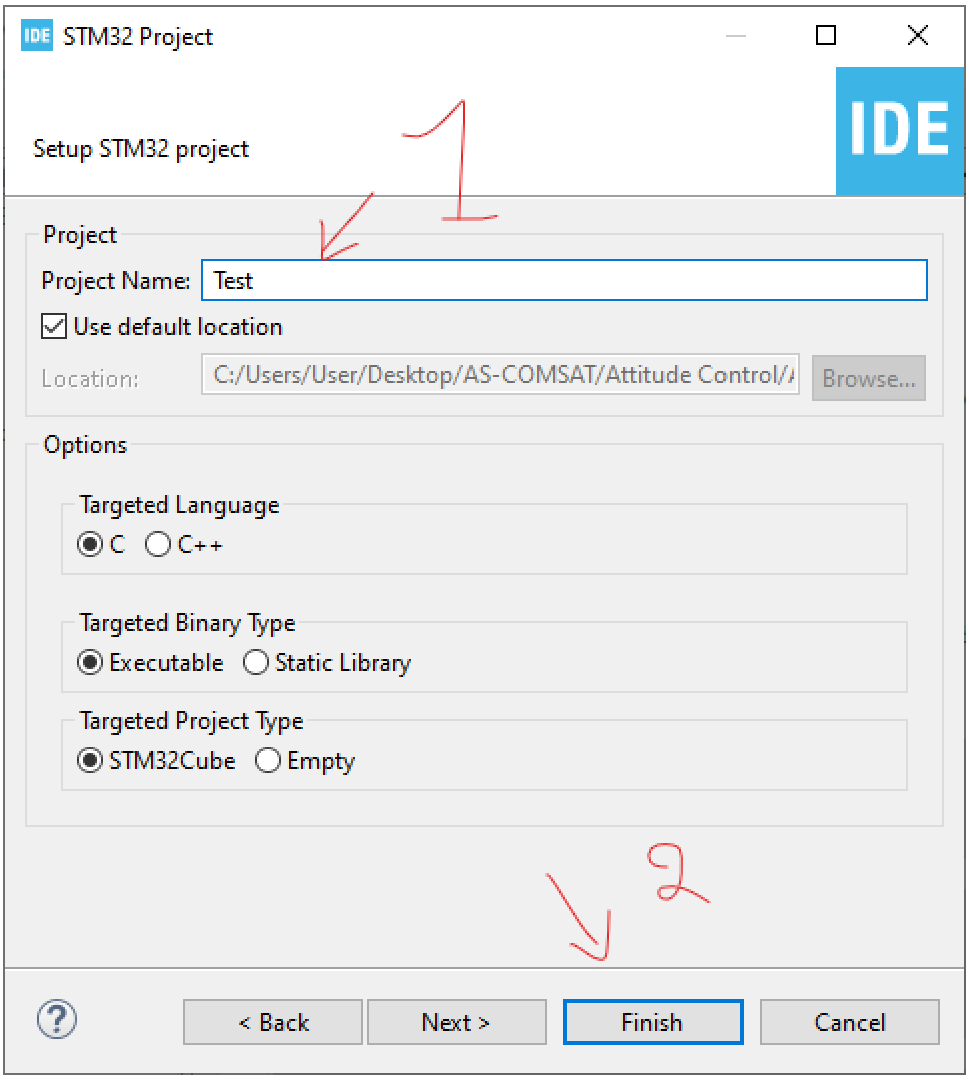


Figure 5-3- Project Name and Creation

When we reach this page, first we need to configure the RCC configuration as shown in the figure below.

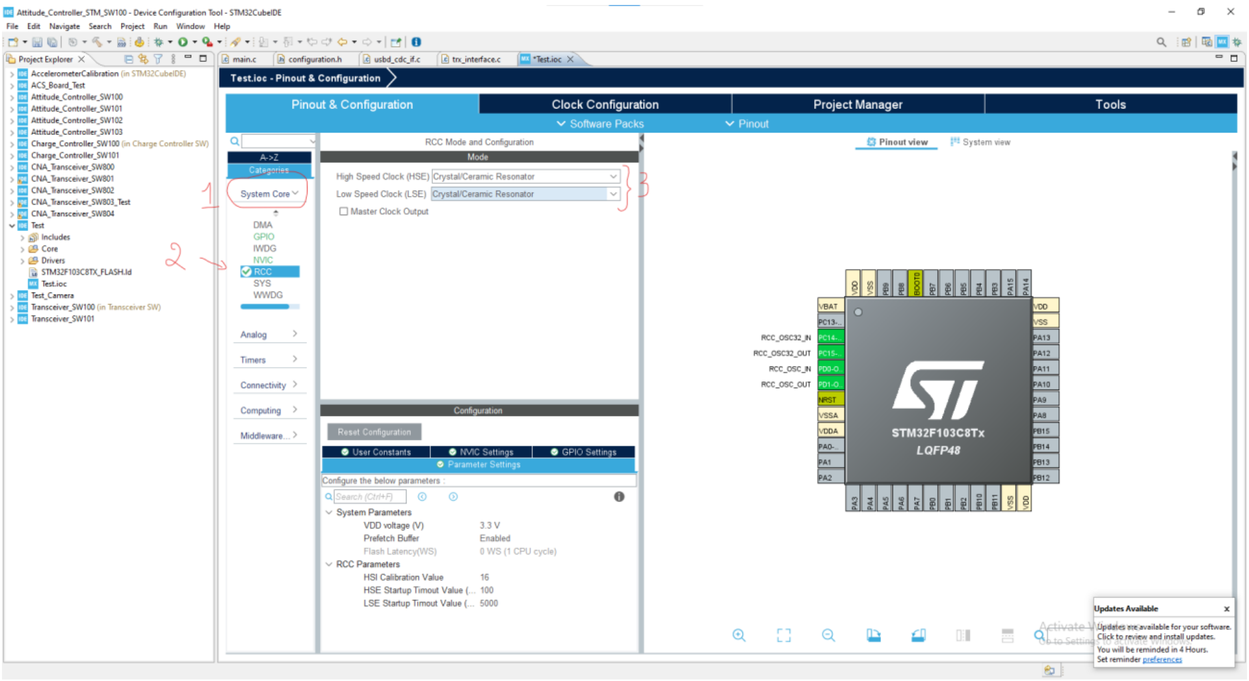


Figure 5-4- RCC configuration

Next, we need to configure the USB configuration as shown in figure 5-5 and 5-6 below

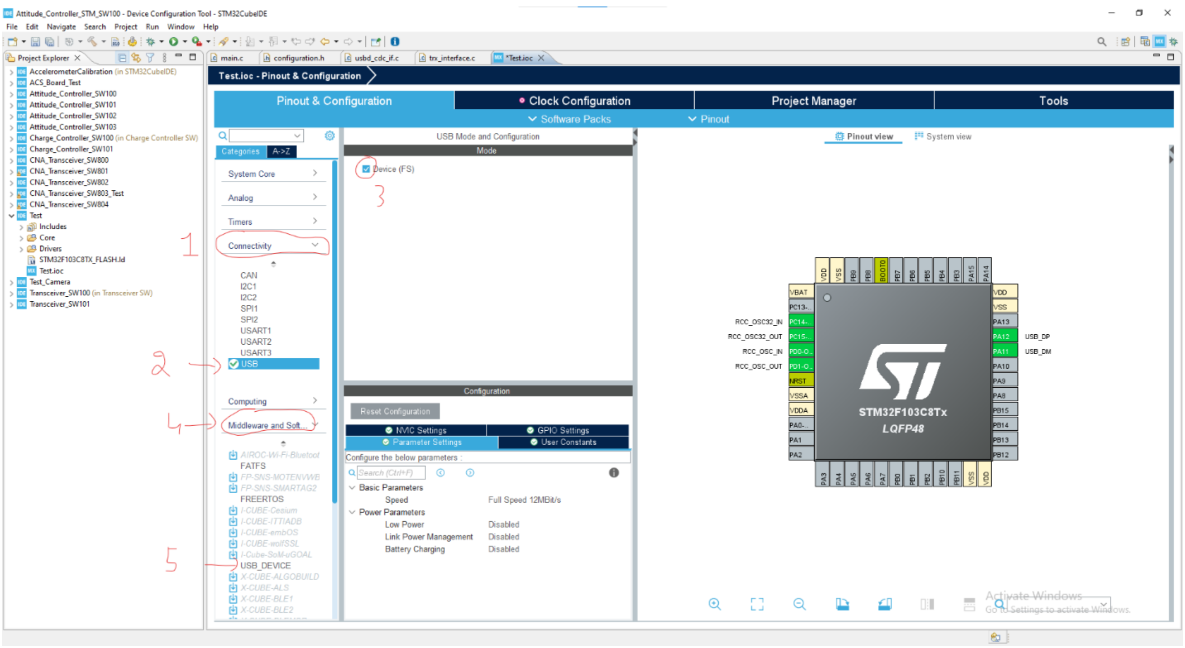


Figure 5-5- USB Configuration

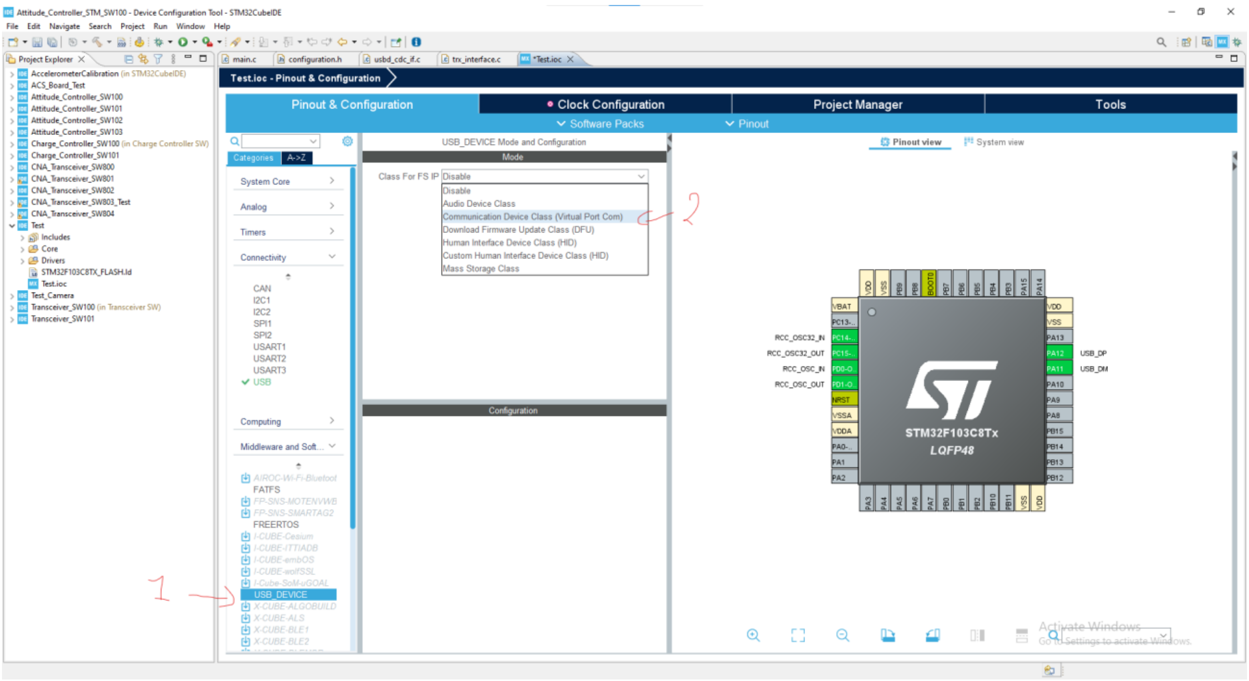


Figure 5-6- USB Configuration

Lastly, we need to change the clock configuration and put it to 72 MHz as shown in the figure below

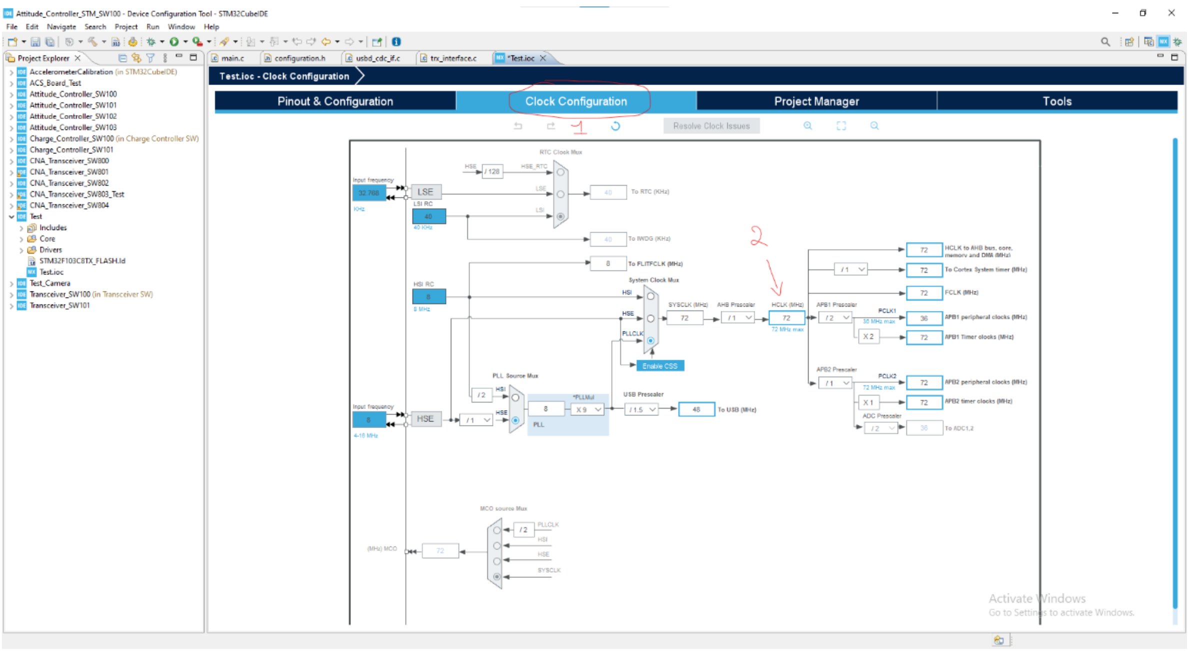


Figure 5-7- Clock Configuration

After finishing with the configuration, we need to “release” the code.

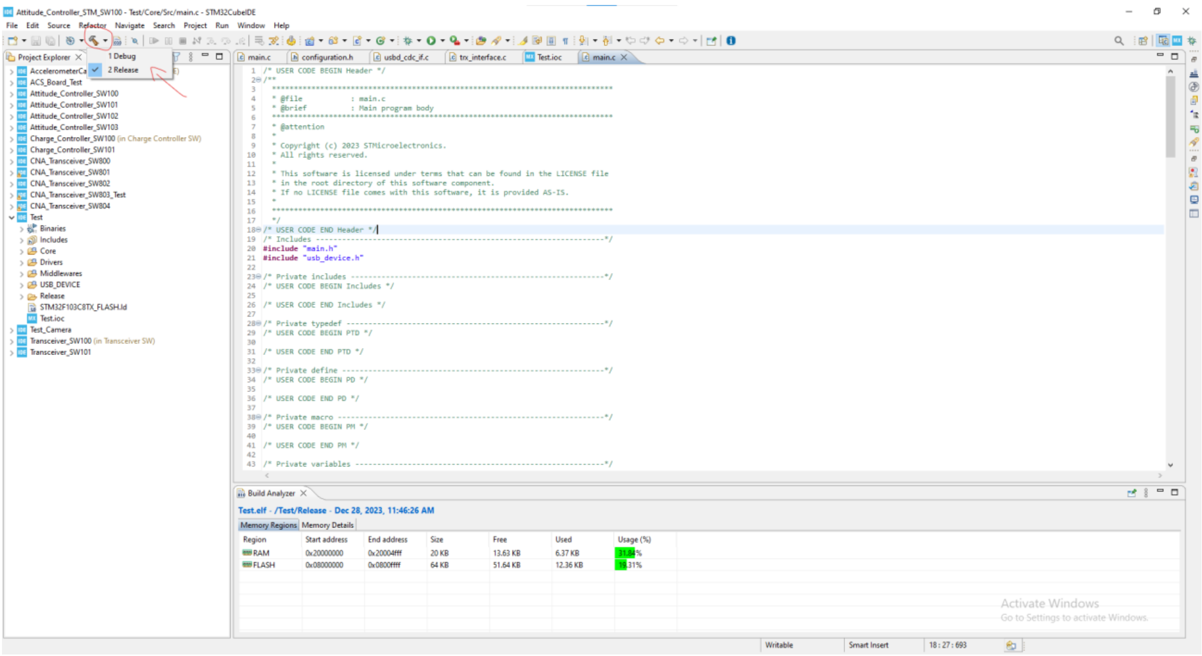


Figure 5-8- Code Release

After releasing the code, we open "STM32CubeProgrammer" and then we need to choose the file that we have released.

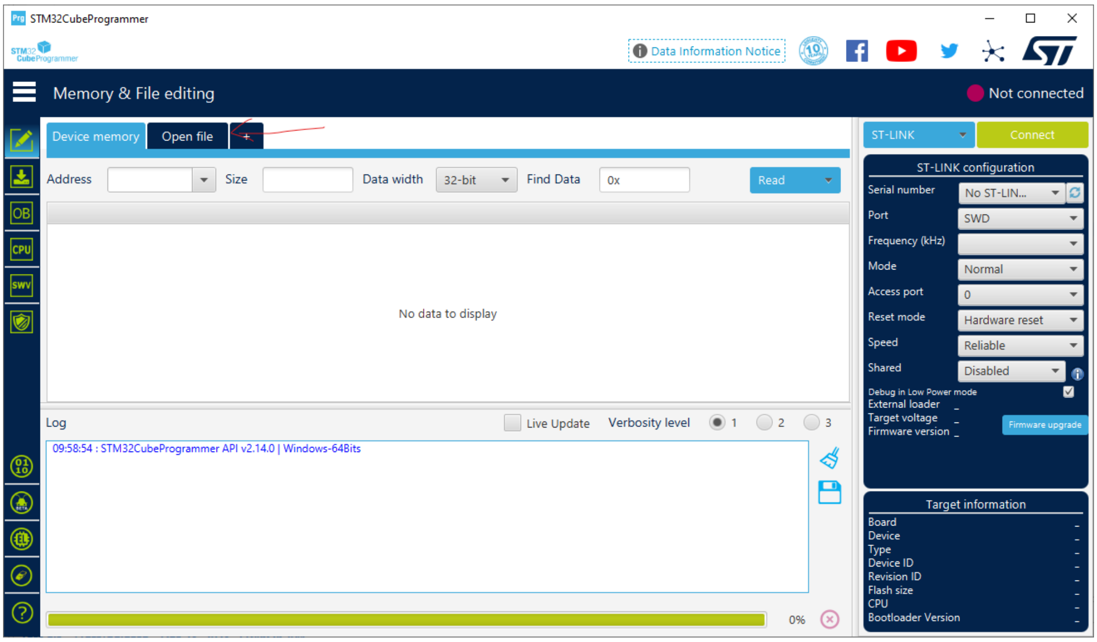


Figure 5-9- Choosing file from Programmer

Then we choose the code elf file from where we saved it in the "Release" folder. Here is a sample directory path of where the file is saved: Test\Release\fileName.elf

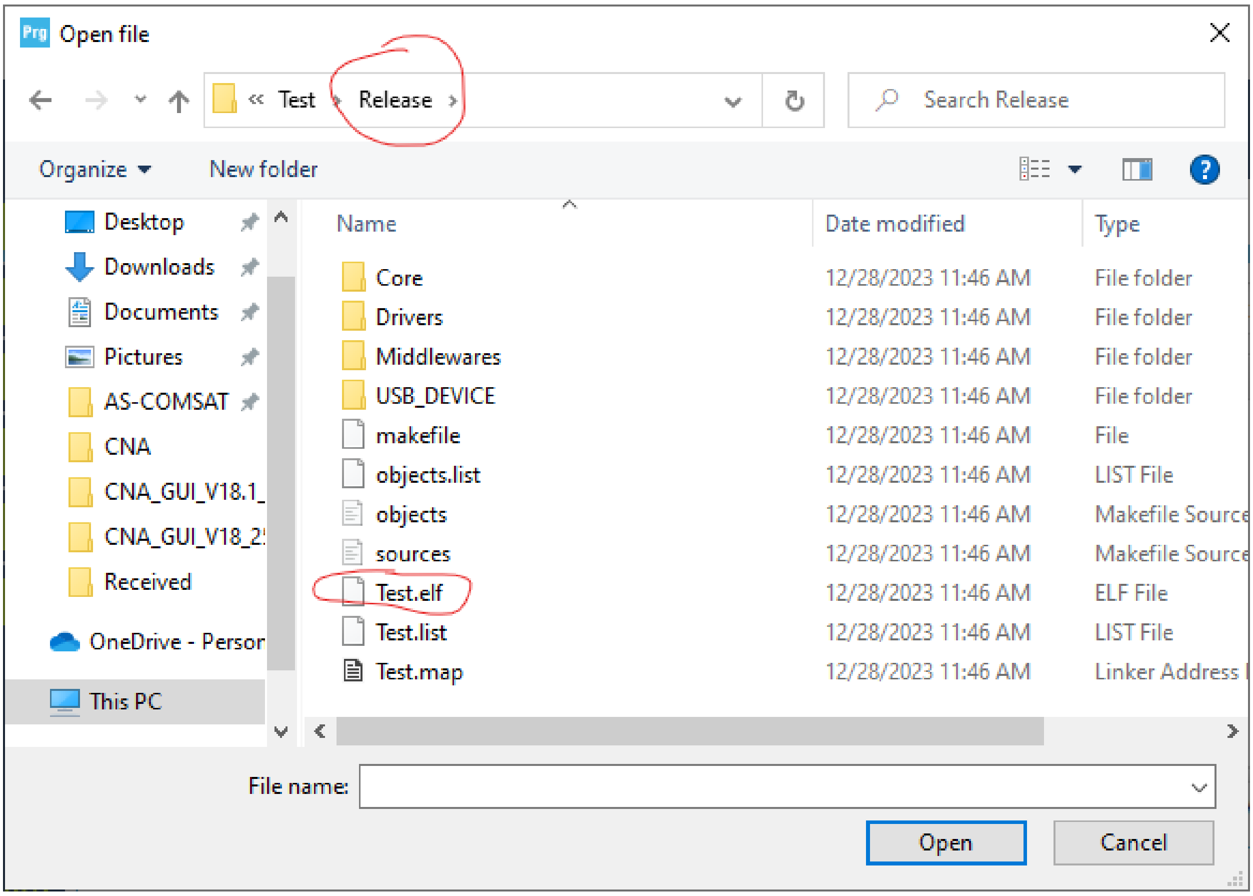


Figure 5-10- Choosing elf file for Programmer

We need to make sure to connect the "ST-Link" USB programmer to the PC without connecting a USB serial port.

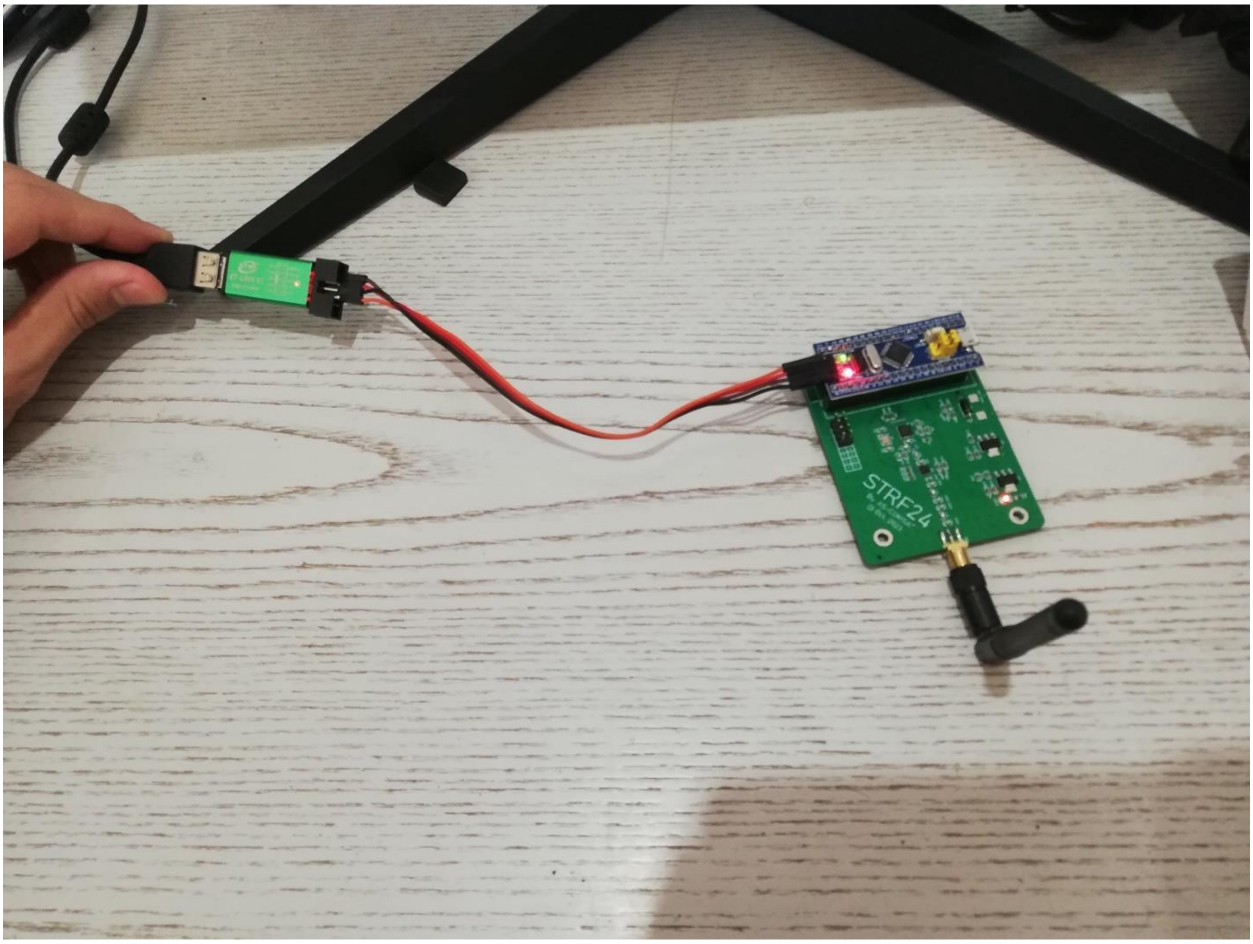


Figure 5-11- Connecting ST-Link programmer USB

Next, we click connect and then we download the file to the STM32.

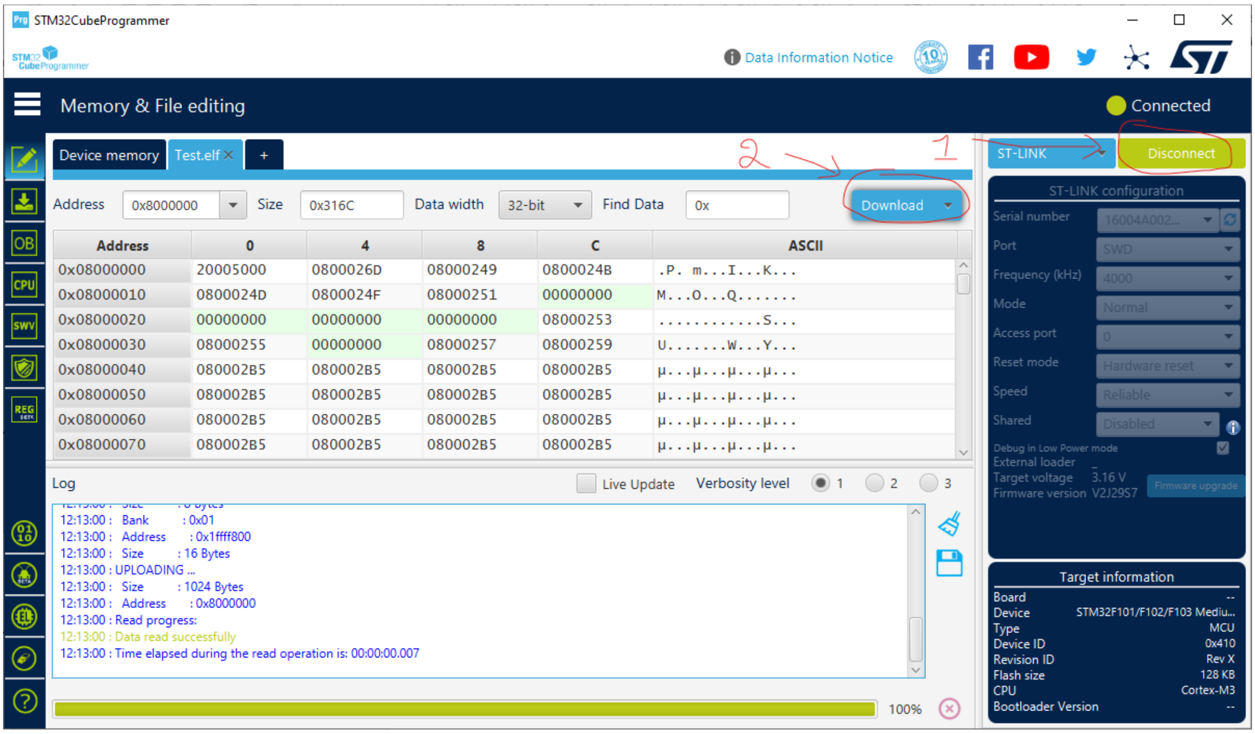


Figure 5-12- Downloading the code on the STM32

Lastly, we need to click the reset button found on the STM32 to refresh the code and we can plug a USB serial port after finishing all the steps.

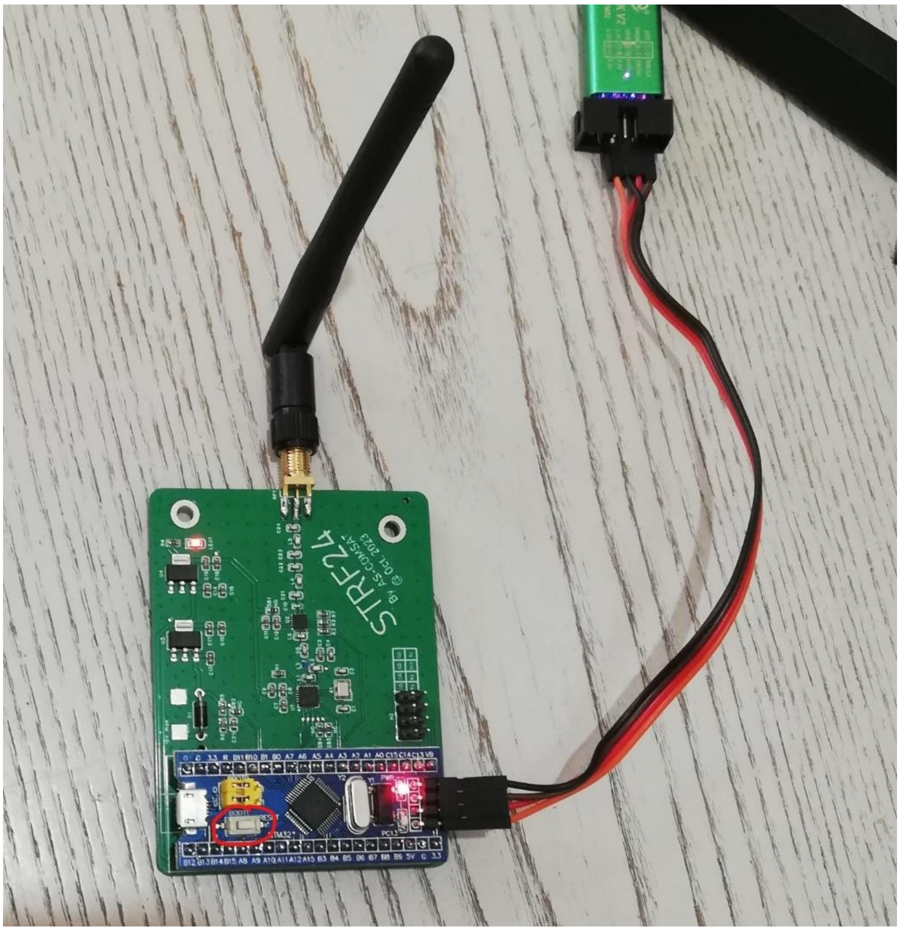


Figure 5-13- Resetting the STM

11 Further Issues to Satellite Development 2024

11.1 AS-COMSAT_1 TT&C (Telemetry, Tracking & Control) System

https://aecenar.com/index.php/companies/as-comsat/as-comsat-platforms-devices/ics-iap-sat/as-comsat-ground-segment

11.1.1 TT&C Testrig

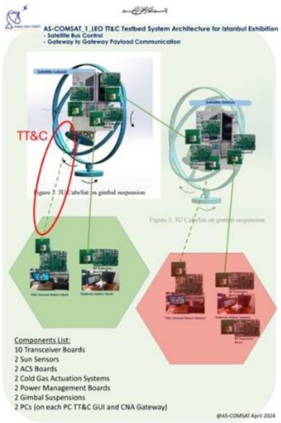
AECENAR TECDA
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region

بسم الله الرحمن الرحيم

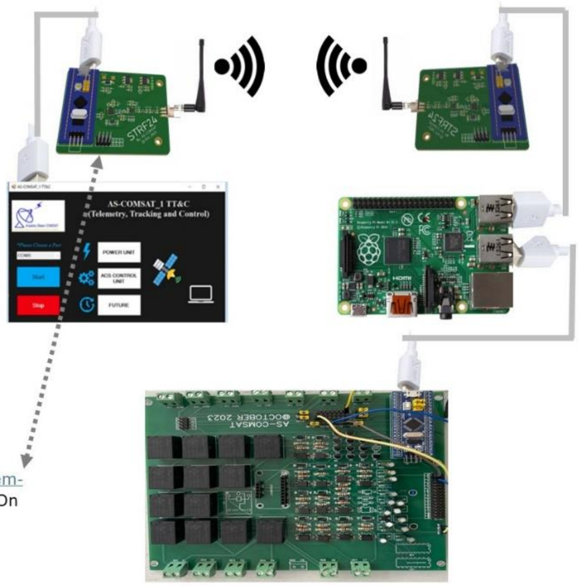
Anadolu-Sham COMSAT

AS-COMSAT_1 Telemetry, Tracking & Control (TT&C) System

TT&C part on the ground station: Sending Guidance data to the satellite ACS board from ground station and receiving position status data from the satellite
TT&C part on the satellite: Sending navigation information to the ground and receiving guidance information from the ground



The Telemetry, Tracking, and Control (TT&C) transceiver on the satellite is responsible for transmitting back to the ground station the telemetry and tracking data such as battery health, solar panels health, position updates, acceleration, etc. It is also responsible for receiving navigation commands from the ground station.



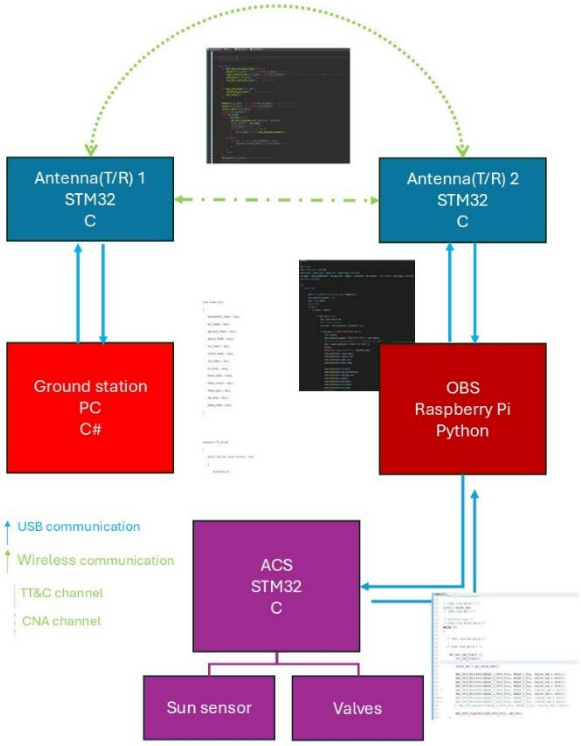
AS-COMSAT developed transceiver boards (https://aecenar.com/index.php/companies/as-comsat/as-comsat-platforms-devices/rf-2-4ghz-tranceiver-unit-prototype/rf-2-4ghz-system-design-microchip) are responsible for providing this communication. On the transceiver board there is a STM32 chip.

Protocol 1:
Sending Guidance data to the satellite ACS board from ground station

1. ground station sends data (CMD) to the antenna (T/R) 1 via USB
2. antenna (T/R) 1 will receive the data from ground station via USB
3. antenna (T/R) 1 will send this data to the second antenna via TT&C channel
4. antenna (T/R) 2 will receive this data from the first antenna via TT&C channel
5. antenna (T/R) 2 will send this data to the OBS via USB
6. OBS will receive the data from the antenna (T/R) 2 via USB
7. OBS will send the data to the ACS via USB
8. ACS will receive the data from OBS via USB
9. ACS will take actions depending on those data

Protocol 2:
Transmitting back telemetry and tracking data from satellite to the ground station

1. Ground station will send data to the antenna (T/R) 1 that requests telemetry and tracking data
2. antenna (T/R) 1 will receive the data from ground station via USB
3. antenna (T/R) 1 will send this data to the second antenna via TT&C channel
4. antenna (T/R) 2 will receive this data from the first antenna via TT&C channel
5. antenna (T/R) 2 will send this data to the OBS via USB
6. OBS will receive the data from the antenna (T/R) 2 via USB
7. OBS will request the data from the ACS system via USB
8. ACS will receive the request from OBS via USB
9. ACS will send the required data to the OBS via USB
10. OBS will receive the data from the ACS via USB
11. OBS will send the data to the antenna (T/R) 2 via USB
12. antenna (T/R) 2 will receive this data from the OBS via USB
13. antenna (T/R) 2 will send this data to the first antenna via TT&C channel
14. antenna (T/R)1 will receive this data from the second antenna via TT&C channel
15. antenna (T/R) 1 will send this data to the Ground station via USB
16. Ground station will receive the data from antenna (T/R) 1 via USB



Ahmad Ismail Achaji
10/July(07)/2024

AECENAR TECDA
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region

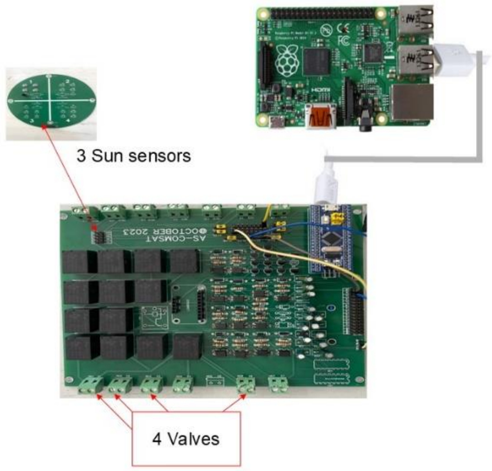
بسم الله الرحمن الرحيم

Anadolu-Sham COMSAT

AS-COMSAT ACS & OBC Design

The attitude control system is responsible for receiving the high-level navigation commands and stabilize the satellite at a specific orientation. The On-Board Computer (OBC) is responsible for receiving the data from the Attitude Control System (ACS) Unit then do some calculations and send the updated status of the valve to the ACS to command. The system uses a raspberry pi as the OBC, and it is coded in python.

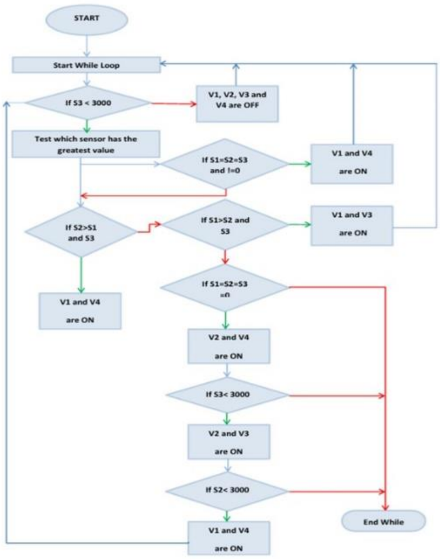
Sun Sensor, Analog signal, goes to the STM32 on the ACS Actuator board and then immediately to the Raspberry Pi. On the Raspberry Pi the python ACS Algorithms determines, which valves shall open. Then Raspberry Pi sends an appropriate signal to the ACS Actuator Board, which is obviously a list contains zeros or one indicates the status of the four valves. Then the ACS Actuator Boards sends a signal to the valves.

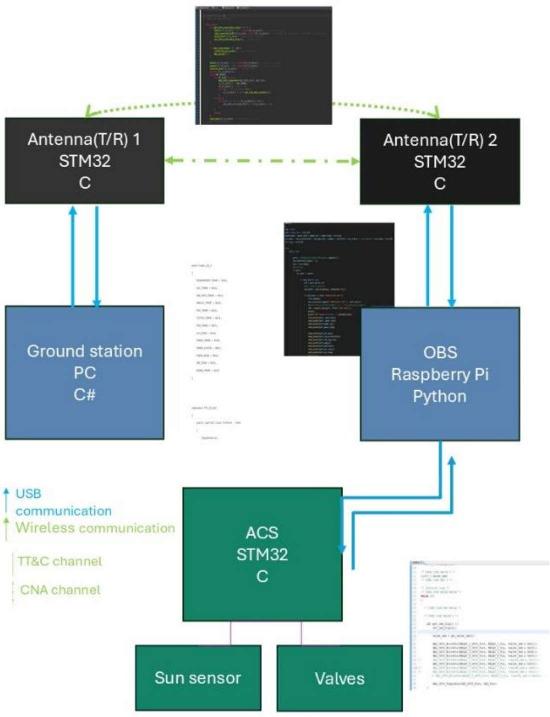


This action of opening or closing the valve is happened through relays on the ACS board.

The Raspberry Pi is connected to the stm32 through a USB cable.

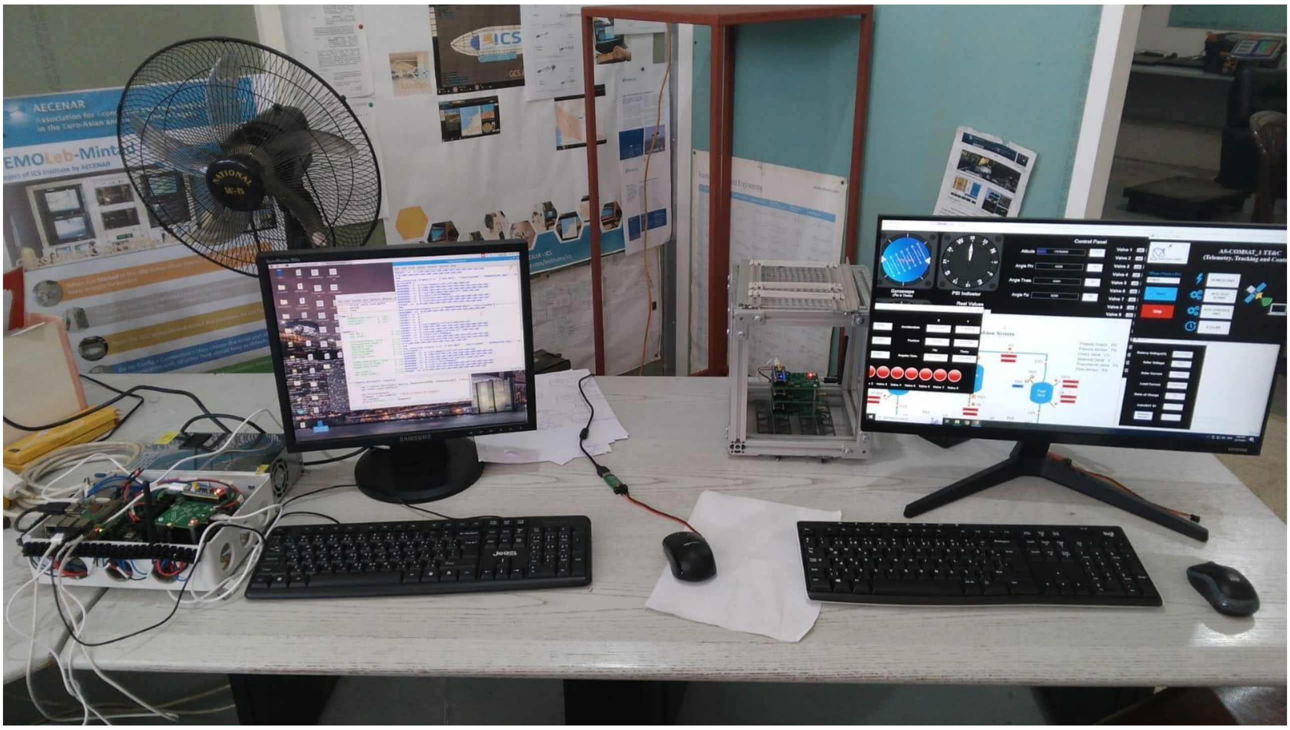
The STM32 should update the Raspberry Pi by the status of the valves when it sends a request.



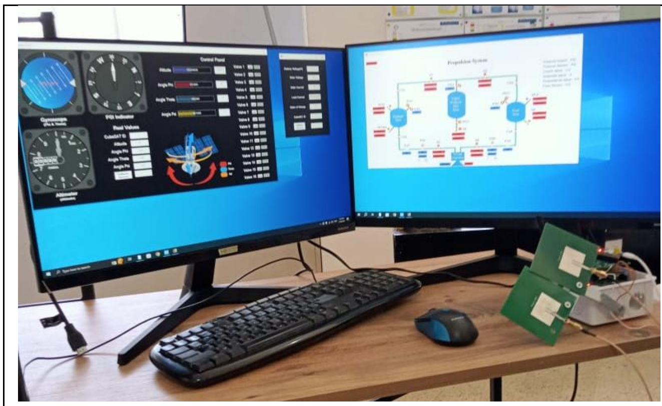


Layla Taleb @aecenar_as_comsat/august 2024

11.1.2 TT&C Testbed



TT&C GUI + Satellite OBC python programme_270224

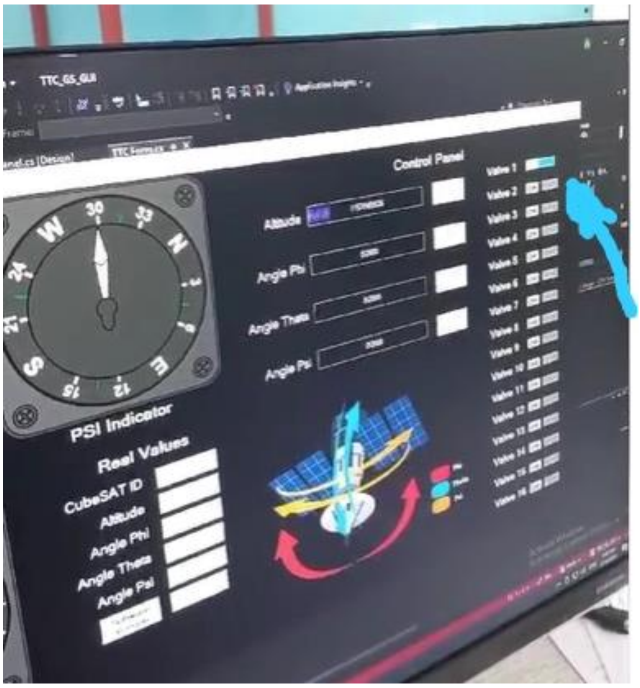
11.1.3 Monitoring values of TT&C Ground Station Transceiver STM32 C Code

In the following we see a film series, where Eng. Raja Mourad explains how to monitor values on the transceiver STM32 of the TT&C ground station while executing:

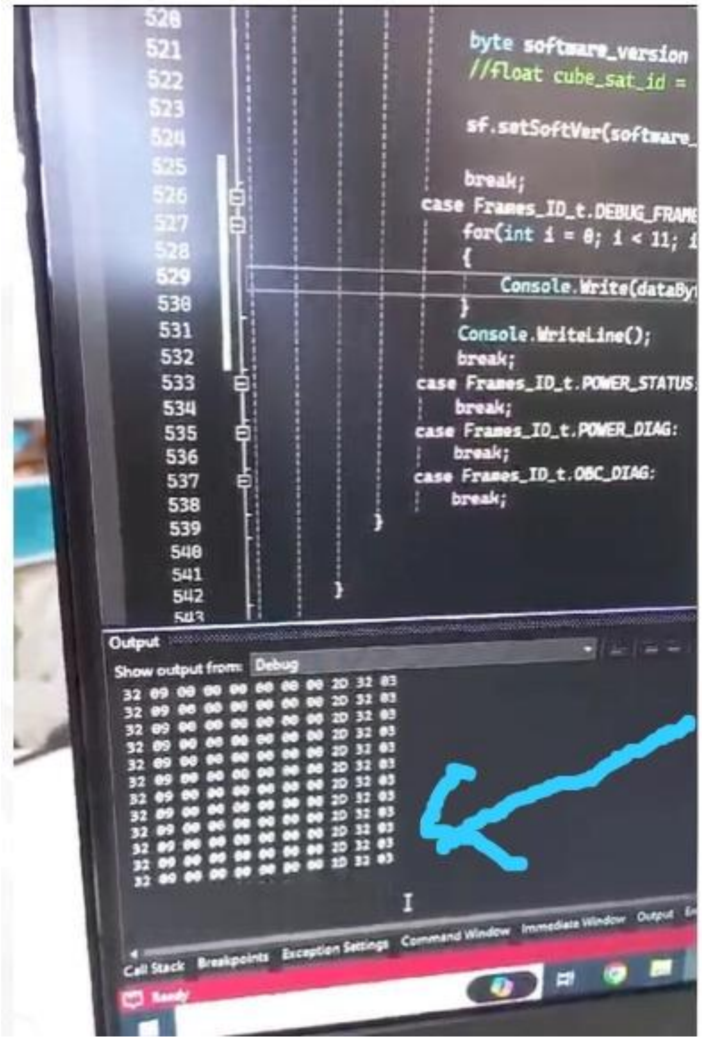
Monitoring values of TT&C Ground station Transceiver STM32 C Code - part 1 (mp4)

Monitoring values of TT&C Ground station Transceiver STM32 C Code - part 2 (mp4)

Action: Change valve status command button:



last frame is for the changed valve status command button:



11.2 ACS

TEMO Soft-, Hardware & Consulting e.K.

AS-COMSAT

hi enterprises

Green Chemistr

Reports of AECE

Start-Up Compa

AS-COMSAT SW&HW Repository

AS-COMSAT Planning&Controlling

AS-COMSAT Platforms&De

AS-COMSAT Projects

DevOps CI/CI Development (HW, GUI anc SW)

AS-COMSAT CNA+LEO-Sa

Launch Issue and other sup India and Rus

AS-COMSAT_1 (LEO Communicati

RF 2.4GHz Tı Prototype

ICS Emergen System (ECS based)

AS-COMSAT Ambulance ((

AIS Specifica Cases

RF 144 MHz Unit Prototyp

AS-COMSAT_1 (LEO Satellite) System Architecture

AS-COMSAT_1 ACS (Design&Realization&Testing)

AS-COMSAT Management

AS-COMSAT Satellite - Stru Integration

AS-COMSAT

AS-COMSAT Computer (OI

AS-COMSAT_1 Launching

AS-COMSAT_1 LEO Satellite Concepts

AS-COMSAT_1 LEO to GEO Orbit Change

AS-COMSAT_1 ACS Board STM32 SW

ACS Board - Ver. 0524

AS-COMSAT_1 ACS Sun Sensor

AS-COMSAT_1 ACS Teststand (Requirements&Design&Realization)

ion-testing

- All Pages

```
def OutofSolarSensorToPercentage():
    """
    DependOfTypeOutputofSolarSensor

    """
    return
def PercentageToDecimal(perc):
    """This methode transform the pourcentage to a nombre decimal 8 bits """
    if(perc<0 or perc>100):
        raise (ValueError)
    else:
        decimal_value= (perc/100)*255
    return decimal_value
def DegreToTime(degree):
    """ this methode have the formula of valve
        she transforme the degree to time needed to open a valve
        exp: turn the valve for 10 sec the satelite turn 5 degree

    Args:
        degree (float): the degree of rotation
    """
    return

def RotateRight(time):
    """this method control the valve needed to open if i need rotate to right """
    """the following methodes is the same"""
    return 0
def RotateLeft(time):
    return 0
def RotateBas(time):
    return 0
def RotateHaut(time):
    return 0
```

```python
def newPercentageBeforRotate():
    """
    this methode prend the new ouptput of the solar sensor
    """
    return 0
def DecimalValueToCondition(decimal_value):
    """this methode find the sens of rotation needed to rotate the satelite
    the value 80% is not true is the condition when the satelite is in the true position
    """
    ConditionNeeded= PercentageToDecimal(80)
    if(decimal_value<80):
      temp=decimal_value
      RotateRight(DegreToTime(1))
      if newPercentageBeforRotate()>temp :
        temp2=newPercentageBeforRotate()
        while(newPercentageBeforRotate()>=temp2):
            temp2=newPercentageBeforRotate()
            RotateRight(1)
      elif newPercentageBeforRotate()<temp:
        RotateLeft(DegreToTime(2))
        temp2=newPercentageBeforRotate()
        while(newPercentageBeforRotate()>=temp2) :
            temp2=newPercentageBeforRotate()
            RotateLeft(1)
        """complet the same methode of all or optimised method if you have the data complet"""
        """this case if i have 1 solar sensor in the satelite if i have anothers the programe is very easy
           because one sensor in haut give u the
           the sens of rotation in haut or in bas
           and one in the left give u the sens left or right
           compare the value of the sensor in the center whith the other sensor in the left or right ...and the sensor in haut or bas
           3 sensor very good but if the nomber of sensor augment the resolution augment

        """
```

```python
def OutofSolarSensorToPercentage():
    """
    DependOfTypeOutputofSolarSensor


    """
    return
def PercentageToDecimal(perc):
    """This methode transform the pourcentage to a nombre decimal 8 bits """
    if(perc<0 or perc>100):
        raise (ValueError)
    else:
        decimal_value= (perc/100)*255
    return decimal_value
def DegreToTime(degree):
    """ this methode have the formula of valve
        she transforme the degree to time needed to open a valve
        exp: turn the valve for 10 sec the satelite turn 5 degree


    Args:
        degree (float): the degree of rotation
    """
    return
```

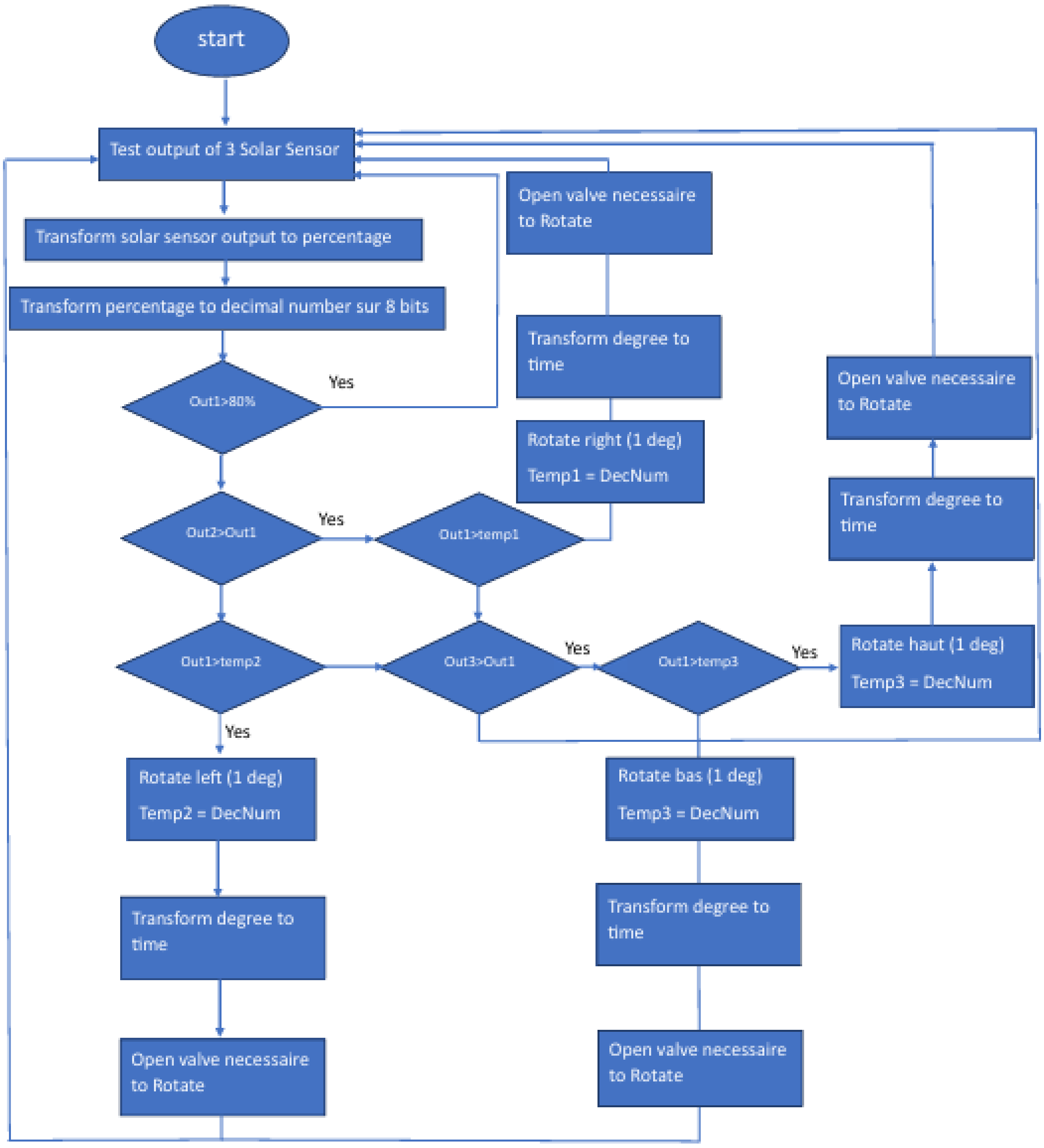
```
def RotateRight(time):
    """this method control the valve needed to open if i need rotate to right """
    """the following methodes is the same"""
    return 0
def RotateLeft(time):
    return 0
def RotateBas(time):
    return 0
def RotateHaut(time):
    return 0
def newPercentageBeforRotate():
    """
    this methode prend the new ouptput of the solar sensor
    """
    return 0
def DecimalValueToCondition(decimal_value):
    """this methode find the sens of rotation needed to rotate the satelite
    the value 80% is not true is the condition when the satelite is in the true position
    """
    ConditionNeeded= PercentageToDecimal(80)
    if(decimal_value<80):
      temp=decimal_value
      RotateRight(DegreToTime(1))
      if newPercentageBeforRotate()>temp :
        temp2=newPercentageBeforRotate()
        while(newPercentageBeforRotate()>=temp2):
            temp2=newPercentageBeforRotate()
            RotateRight(1)
      elif newPercentageBeforRotate()<temp:
        RotateLeft(DegreToTime(2))
        temp2=newPercentageBeforRotate()
        while(newPercentageBeforRotate()>=temp2) :
            temp2=newPercentageBeforRotate()
            RotateLeft(1)
        """complet the same methode of all or optimised method if you have the data complet"""
        """this case if i have 1 solar sensor in the satelite if i have anothers the programe is very easy
           because one sensor in haut give u the
           the sens of rotation in haut or in bas
           and one in the left give u the sens left or right
```

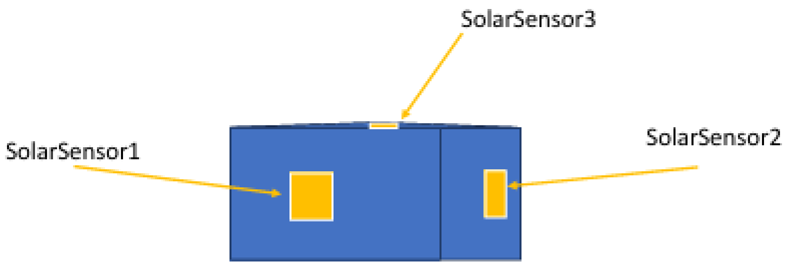
compare the value of the sensor in the center whith the other sensor in the left or right ...and the sensor in haut or bas

3 sensor very good but if the nomber of sensor augment the resolution augment

''''''

three solar sensor





11.3 AS-COMSAT_1 ACS Teststand (Requirements, Design & Realization)

https://aecenar.com/index.php/companies/as-comsat/as-comsat-platforms-devices/ics-iap-sat/as-comsat-1-acs-design-realization-testing/as-comsat-1-acs-teststand-requirements

11.3.1 Requirements for ACS Teststand

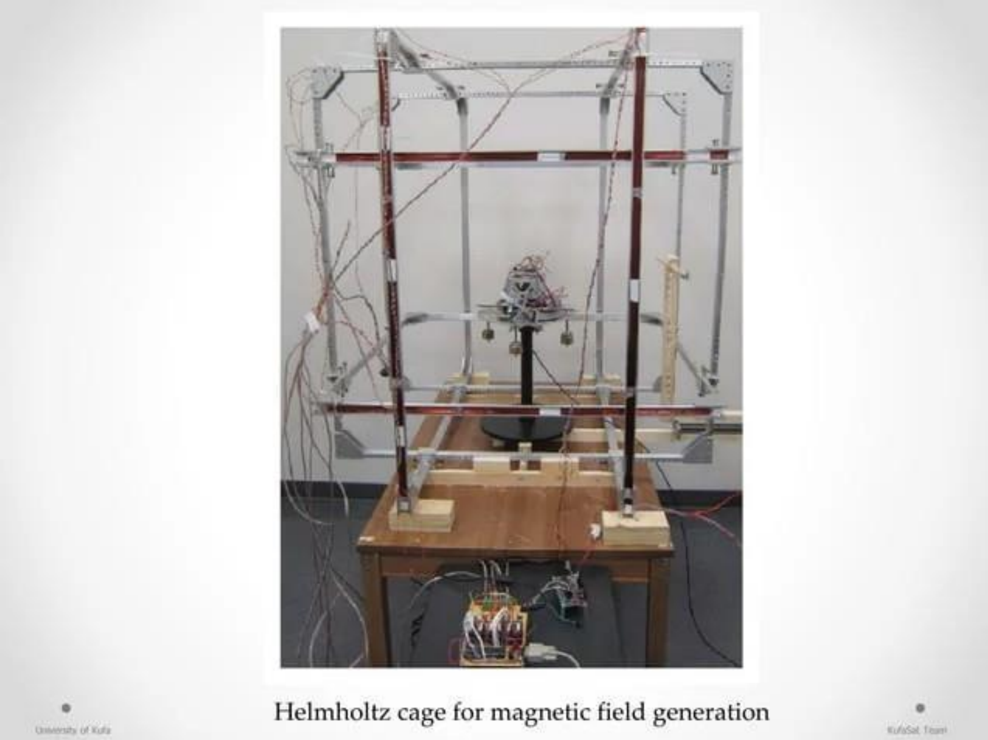
11.3.1.1 Teststand System

(here: Magnetorquer as Actuator)

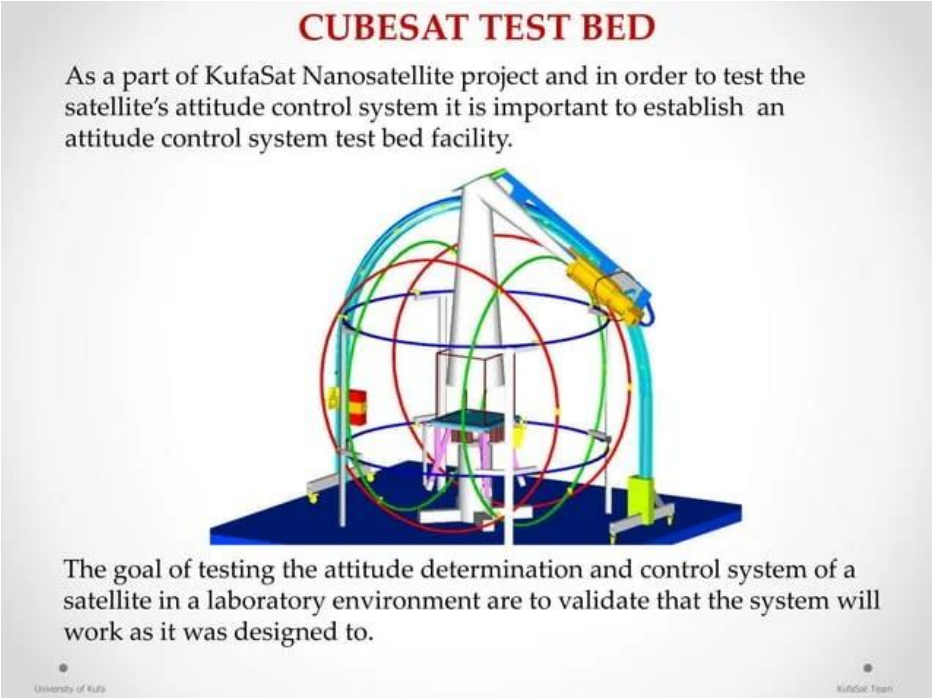
References:

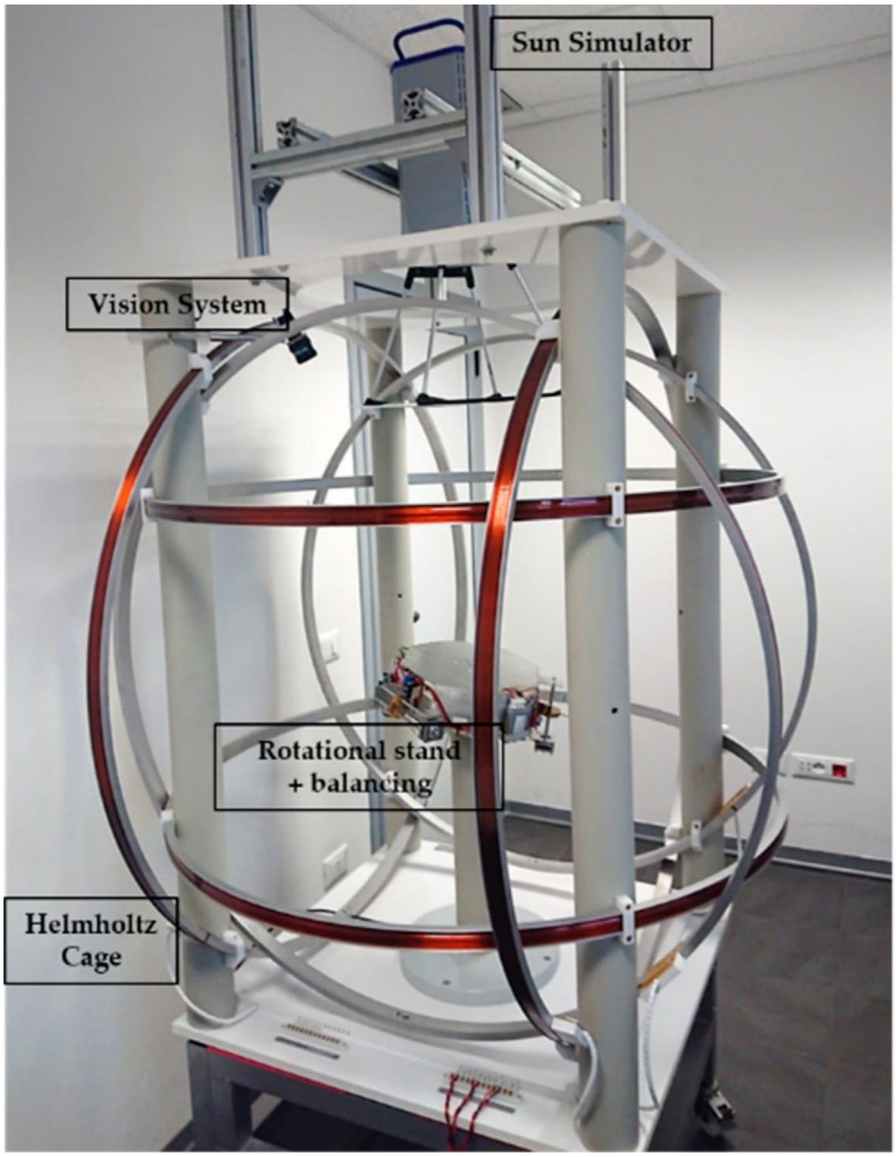
https://www.astrofein.com/en/products/acs-test-stand/

https://www.slideshare.net/mchessab/cube-sat-test-bed


Helmholtz cage for magnetic field generation
University of Kufa
KufaSat Team


The term **Helmholtz cage** refers to a three orthogonal axes Helmholtz coil system for controlling magnetic fields
University of Kufa
KufaSat Team


CUBESAT TEST BED
As a part of KufaSat Nanosatellite project and in order to test the satellite's attitude control system it is important to establish an attitude control system test bed facility.
The goal of testing the attitude determination and control system of a satellite in a laboratory environment are to validate that the system will work as it was designed to.


Sun Simulator
Vision System
Rotational stand
+ balancing
Helmholtz
Cage

11.3.1.2 Actuator System (Cold Gas)

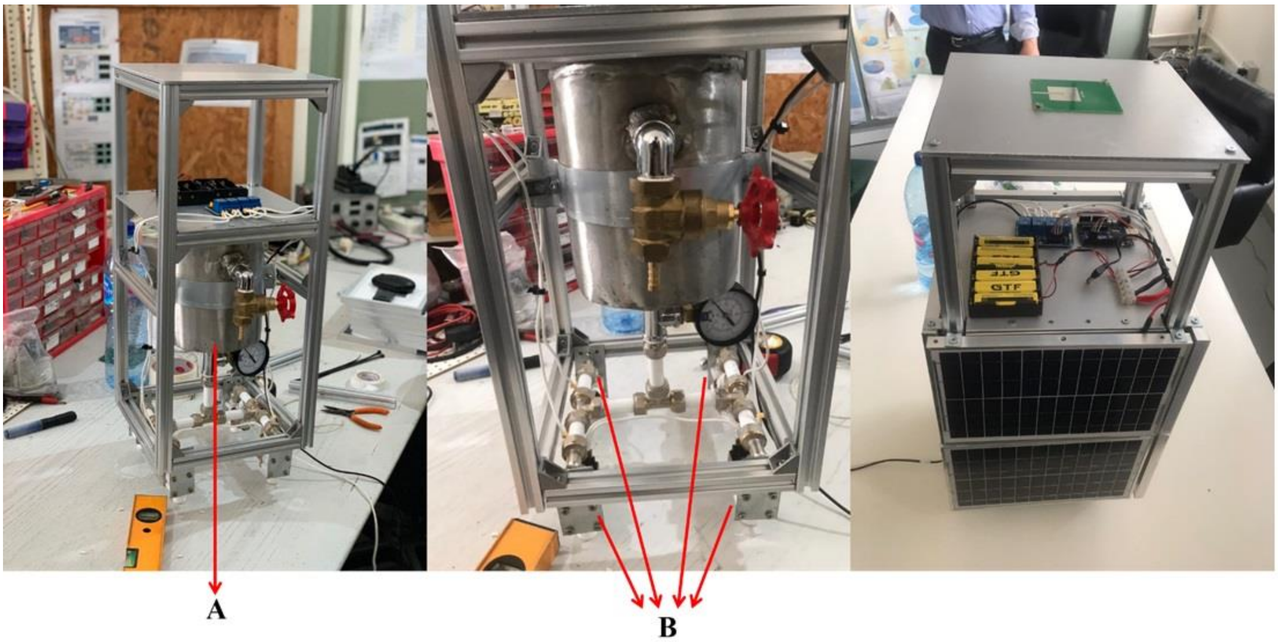


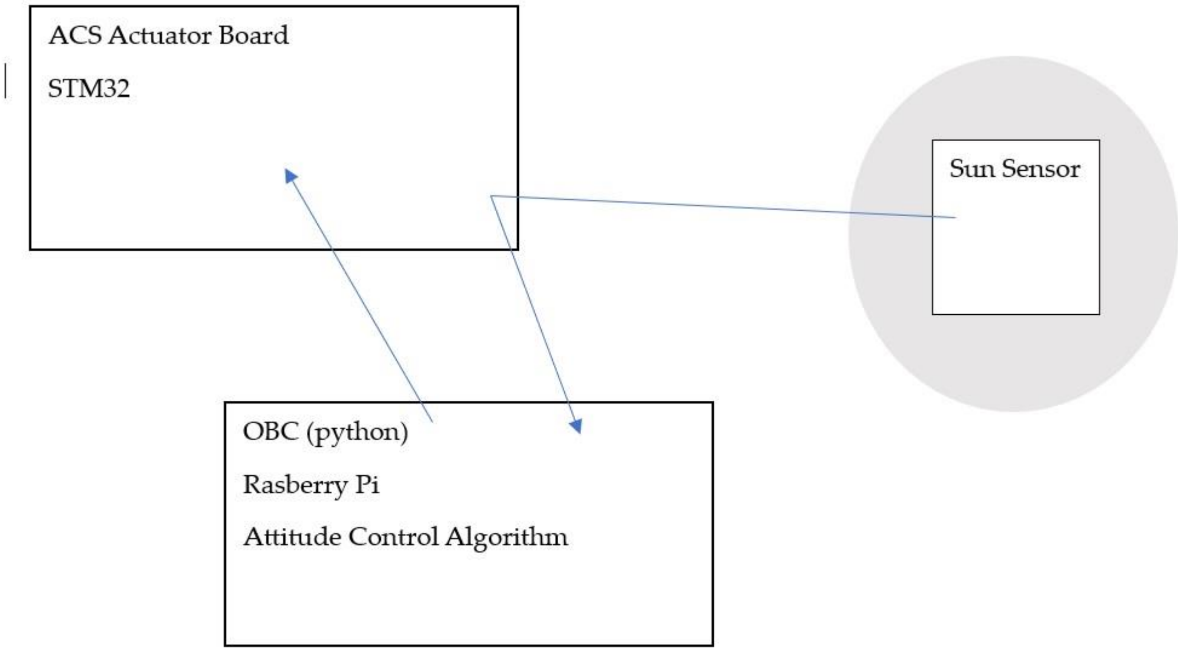
Figure 2-3 Mechanical realization of satellite system V1(cold gas propulsion unit)
)

- A: Pressurized gas container: used to store a high-pressure gas such as nitrogen to generate the propulsion for the Attitude Control System (ACS).
- B: Solenoid valves used to distribute the pressure and generate thrust for the CubeSat.

11.3.2 Design of ACS Teststand

Responsible: Layla Taleb

11.3.2.1 Block Diagram (System Design)



Explanation:

Sun Sensor Analog signal goes to the STM32 of the ACS Actuator board and then immediately to the Raspberry Pi. On the Raspberry Pi the python ACS Algorithms determines, which valves shall open. Then Raspberry Pi sends an appropriate signal to the ACS Actuator Board. Then the ACS Actuator Boards sends a signal to the valves.

The Raspberry Pi is connected to the stm32 through a USB cable.

The Raspberry Pi should update the stm32 by the status of the valves when it sends a request.

STM32 Program:

```
/* USER CODE BEGIN 2 */
  ADC_ChannelConfTypeDef sConfig = { 0 };
  uint32_t adc_values[3];


  HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin, 0);
/* USER CODE END 2 */

/* Infinite loop */
/* USER CODE BEGIN WHILE */
  while (1) {

      // Loop through ADC channels for PA0 (ADC_CHANNEL_0), PA1 (ADC_CHANNEL_1), and PA2 (ADC_CHANNEL_2)
      for (int i = 0; i < 3; i++) {
          sConfig.Channel = ADC_CHANNEL_0 + i; // Increment channels (0 -> PA0, 1 -> PA1, 2 -> PA2)
          sConfig.Rank = 1;
          sConfig.SamplingTime = ADC_SAMPLETIME_13CYCLES_5;
          HAL_ADC_ConfigChannel(&hadc1, &sConfig);

          HAL_ADC_Start(&hadc1);
          HAL_ADC_PollForConversion(&hadc1, 100);
          adc_values[i] = HAL_ADC_GetValue(&hadc1); // Store each value in the array

          HAL_Delay(5);
      }

  /* USER CODE END WHILE */
```

The while loop presented above allows us to read constantly the inputs from channels 0,1 and 2 which are connected to the sun sensor and stores each of them in the array.

```
/* USER CODE BEGIN 4 */
void get_s_values (uint8_t *retArr)
{
  memcpy(s_values, retArr, 6);
}
/* USER CODE END 4 */
```

We now need to send these values to the Raspberry Pi, we can write them here but this will end by sending them constantly.

```
static int8_t CDC_Receive_FS(uint8_t *Buf, uint32_t *Len) {
    /* USER CODE BEGIN 6 */
    USBD_CDC_SetRxBuffer(&hUsbDeviceFS, &Buf[0]);
    USBD_CDC_ReceivePacket(&hUsbDeviceFS);

    memset(tx_buffer, '\0', 11);  // clear the tx buffer
    memset(rx_buffer, '\0', 11);  // clear the rx buffer
    uint8_t len = (uint8_t) *Len;

    memcpy(rx_buffer, Buf, len);  // copy the data to the buffer
    memset(Buf, '\0', len);   // clear the Buf also



    switch (rx_buffer[0]) {

    case 0x32:
        tx_buffer[0] = 0x32;
        tx_buffer[1] = 6;
        get_s_values(test);
        for (int i = 0; i < 6; i++) {
            tx_buffer[i + 2] = test[i];
        }

        break;

    }

    CDC_Transmit_FS(tx_buffer, sizeof(tx_buffer));
    return (USBD_OK);
    /* USER CODE END 6 */
}
```

The method above is written in the file "usbd_cdc_if.c".

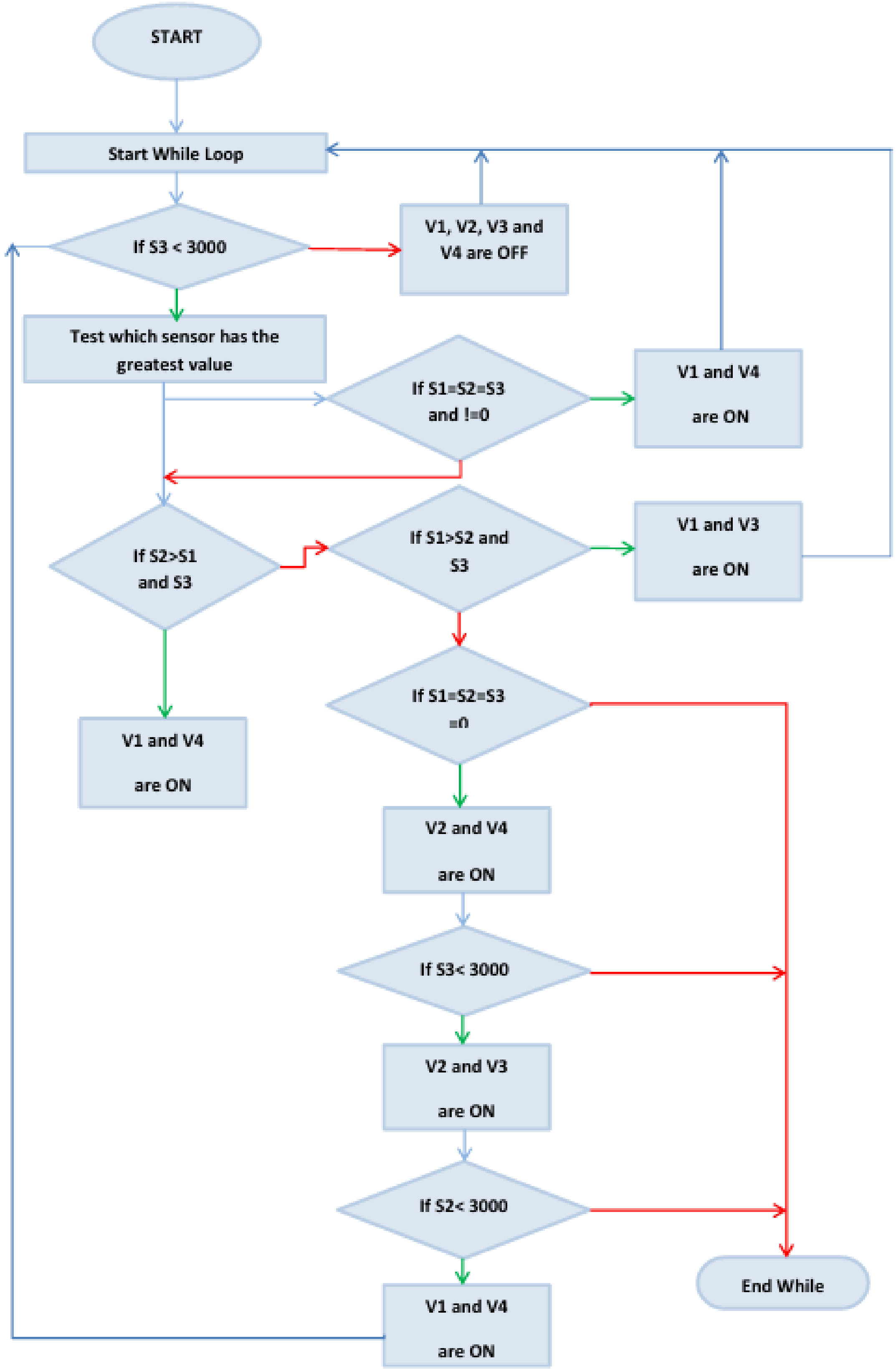
This method ensures that the stm32 won't send the data of the sun sensor to Raspberry Pi until it sends a request carrying with the address which is 0x32.

Python Code:

```
FRAMES = {
'WHO AM I': b'\xFF',
'VER FRAME': b'\x17',


'ACS MEASUREMENT FRAME': b'\x11',
'ACCELERATION FRAME': b'\x12',
'ANG RATE FRAME': b'\x13',
'ANGLES FRAME': b'\x14',
'POSITION FRAME': b'\x15',
'STATUS FRAME': b'\x16',
'ACS DIAG FRAME': b'\x18',

'POWER MEASUREMENT FRAME': b'\x20',
'POWER STATUS FRAME': b'\x21',
'POWER DIAG FRAME': b'\x22',


'CMD FRAME': b'\x30',
'OBC DIAG FRAME': b'\x31',
```

```
'Sun Sensor': b'\x32'
}

elif pid == PIDs['ACS UNIT']:
new_connected_targets['ACS UNIT'] = port.device

acs_meas = request_data(port, FRAMES['ACS MEASUREMENT FRAME'])
raw_acceleration = request_data(port, FRAMES['ACCELERATION FRAME'])
raw_ang_rate = request_data(port, FRAMES['ANG RATE FRAME'])
angles = request_data(port, FRAMES['ANGLES FRAME'])
positions = request_data(port, FRAMES['POSITION FRAME'])
acs_status = request_data(port, FRAMES['STATUS FRAME'])
acs_diag = request_data(port, FRAMES['ACS DIAG FRAME'])
sun_sensor = request_data(port, FRAMES['Sun Sensor'])

send_packet(port, cmd)
```

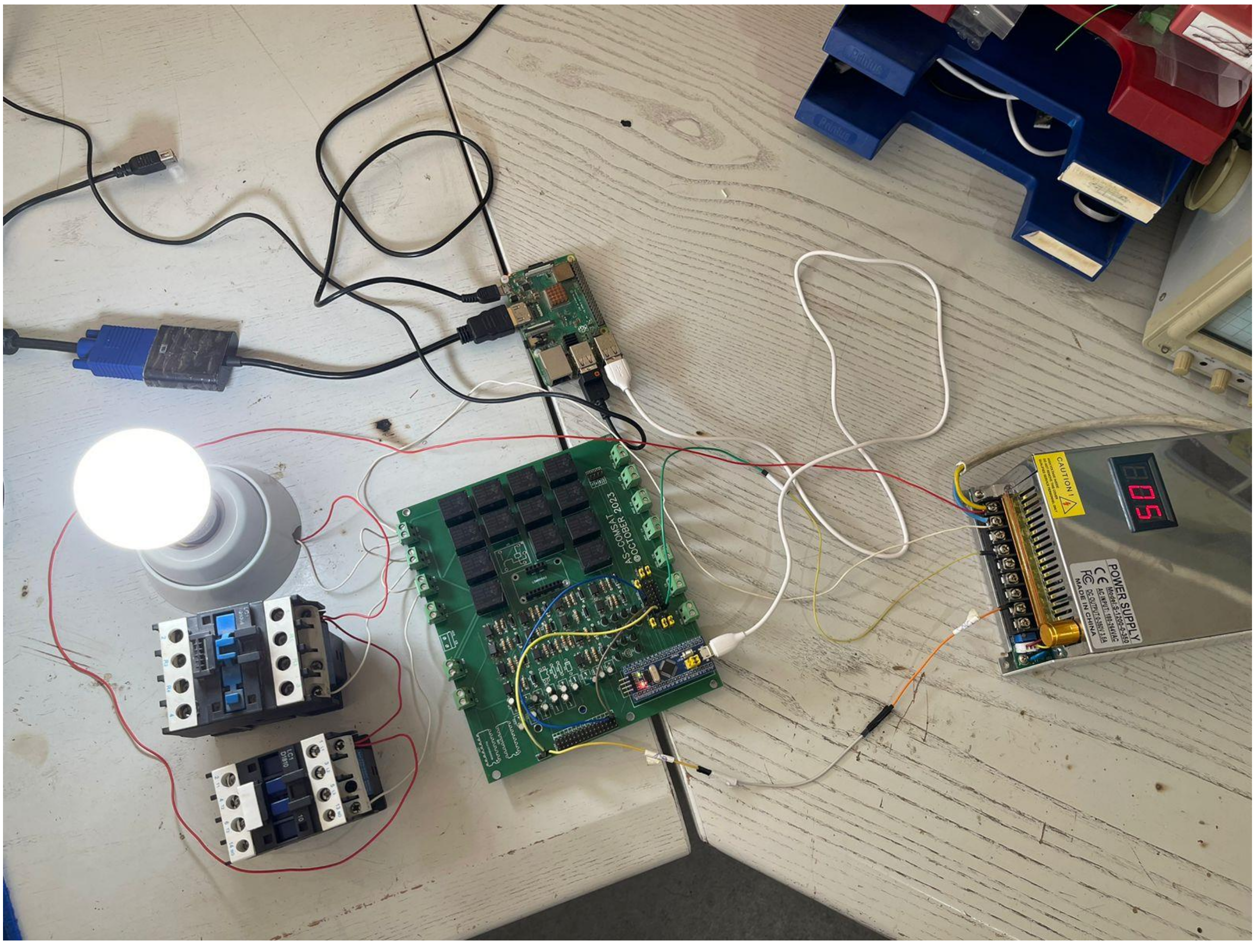
This part is responsible of sending the request from the Raspberry Pi to STM32 on the ACS board.



The Flow chart above represents the algorithm of the motion of the satellite by setting the valves. (S3 should get the highest value)

ACS&OBC Poster [pptx file]

See also the videos on [AS-COMSAT_1 On-Board-Computer (OBC) (aecenar.com)](https://aecenar.com/index.php/companies/as-comsat/as-comsat-platforms-devices/ics-iap-sat/obc-block-diagram) (https://aecenar.com/index.php/companies/as-comsat/as-comsat-platforms-devices/ics-iap-sat/obc-block-diagram):



11.3.3 Realization of ACS Teststand

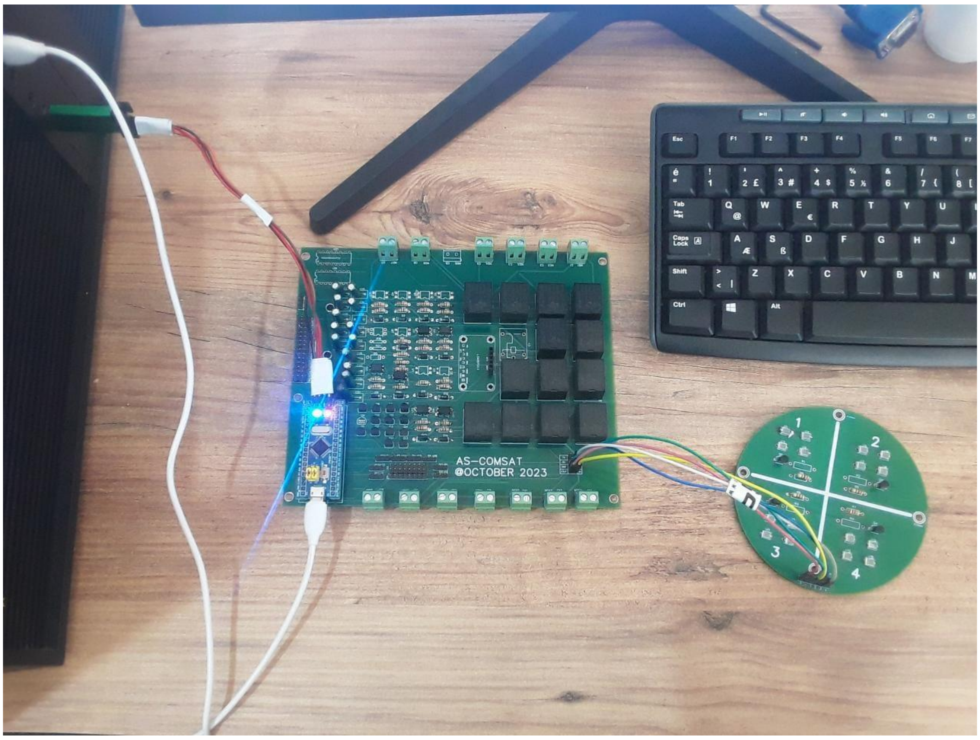
ACS Teststand Realization 2.4.24 (film) (https://aecenar.com/index.php/downloads/send/13-temo-space-communication/1723-as-comsat-1-acsteststand-realization-film)

11.3.3.1 ACS Actuator Board

11.3.3.2 OBC

11.3.3.3 Sun Sensor

Connecting Sun-Sensor directly to ACS Actuator Board



Connecting Sun Sensor to On-Board-Computer (OBC)

Responsible: Layla Taleb

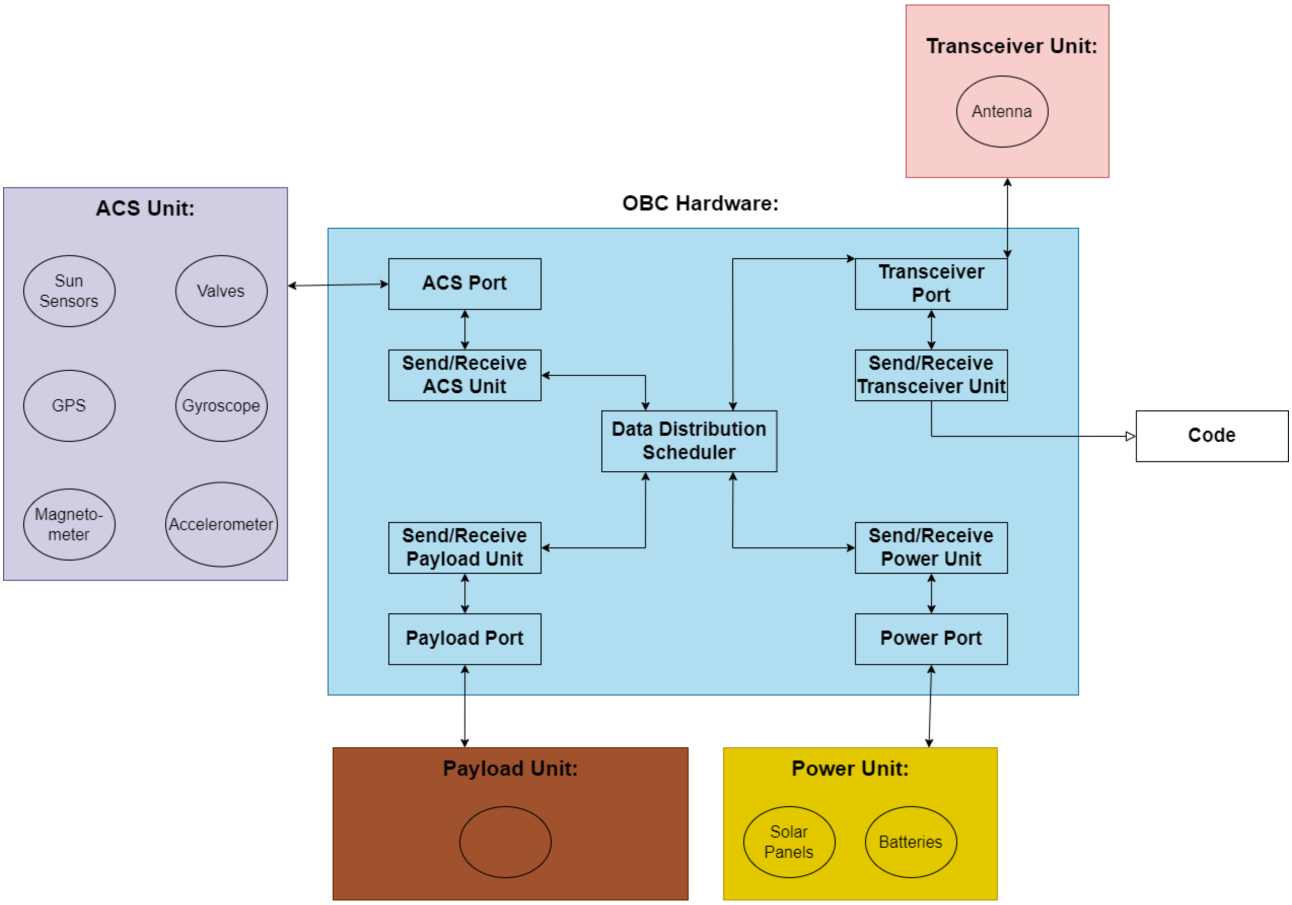
needs Analog Digital Converter (ADC)

11.3.3.4 On-Board Computer

AS-COMSAT_1 On-Board-Computer (OBC) (https://aecenar.com/index.php/companies/as-comsat/as-comsat-platforms-devices/ics-iap-sat/obc-block-diagram)

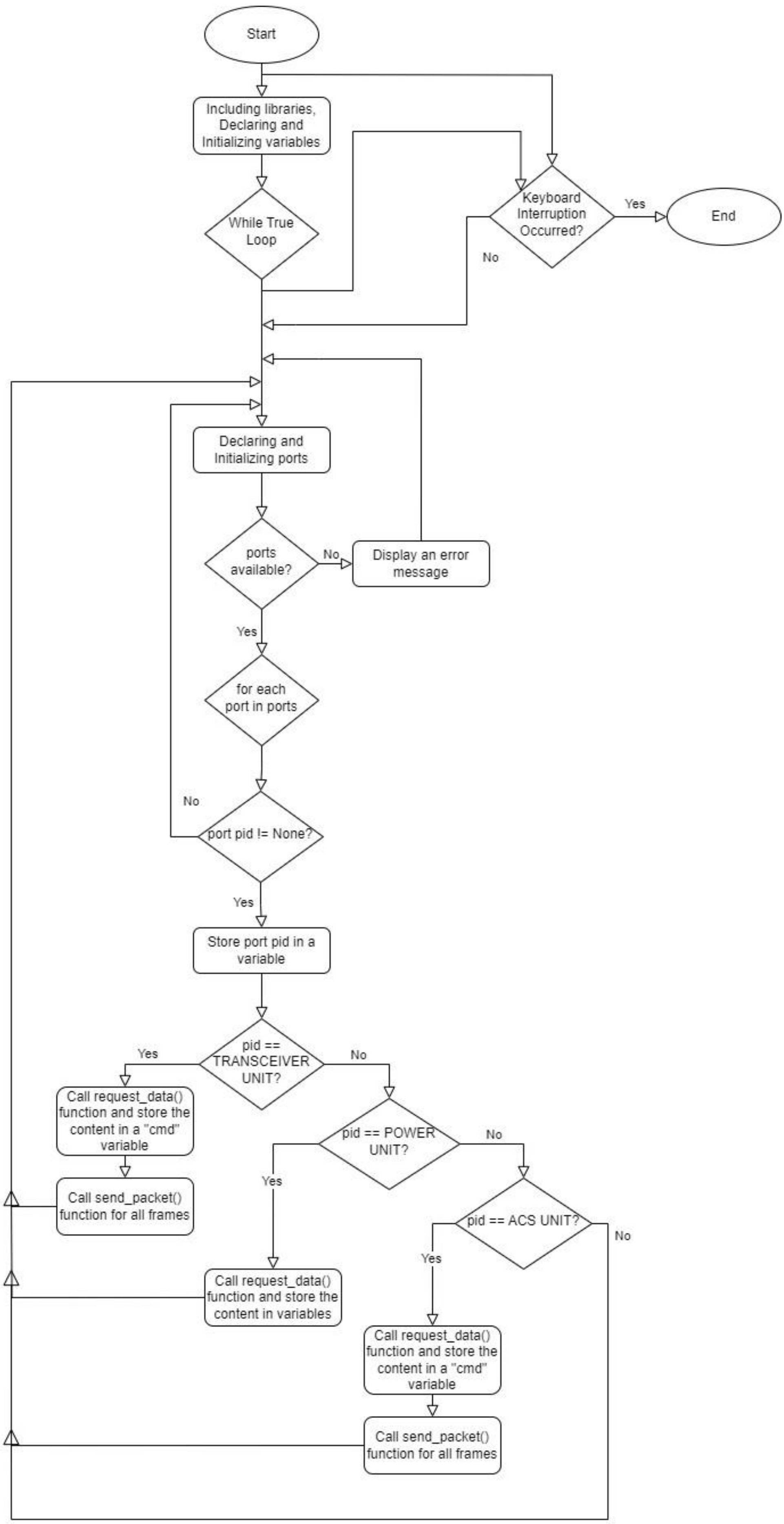
The On-Board Computer (OBC) is responsible for collecting the data from the connected units such as the Power Unit and Attitude Control System (ACS) Unit then sends the collected data to the Transceiver Unit. It also collects other frames from the transceiver, such as CMD_Frame, then forward the data to its corresponding unit. The system uses a raspberry pi as the OBC, and it is coded in python.

OBC Block Diagram

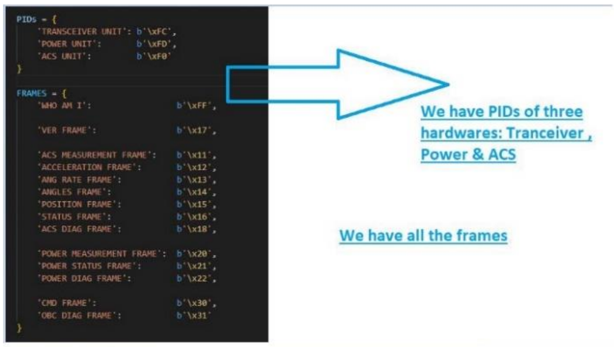


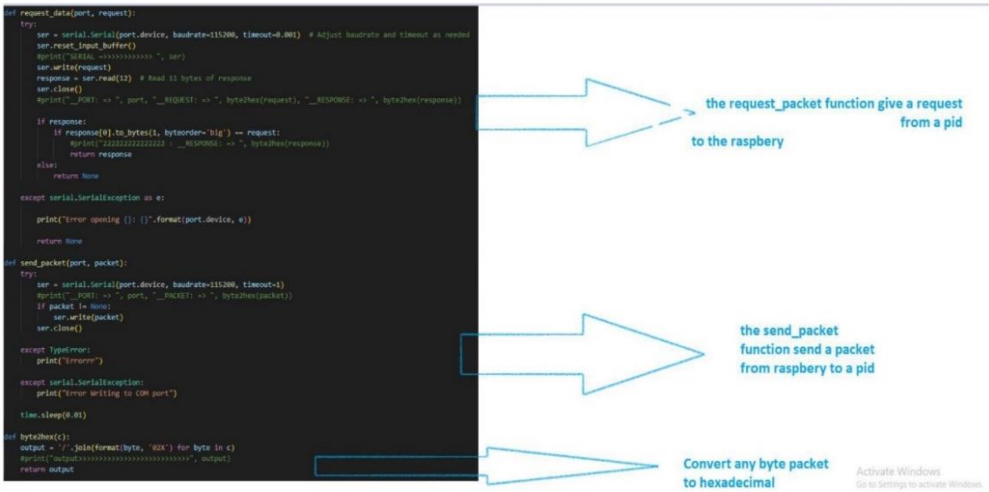
Download OBC Diagram File: OBC Block Diagram.drawio

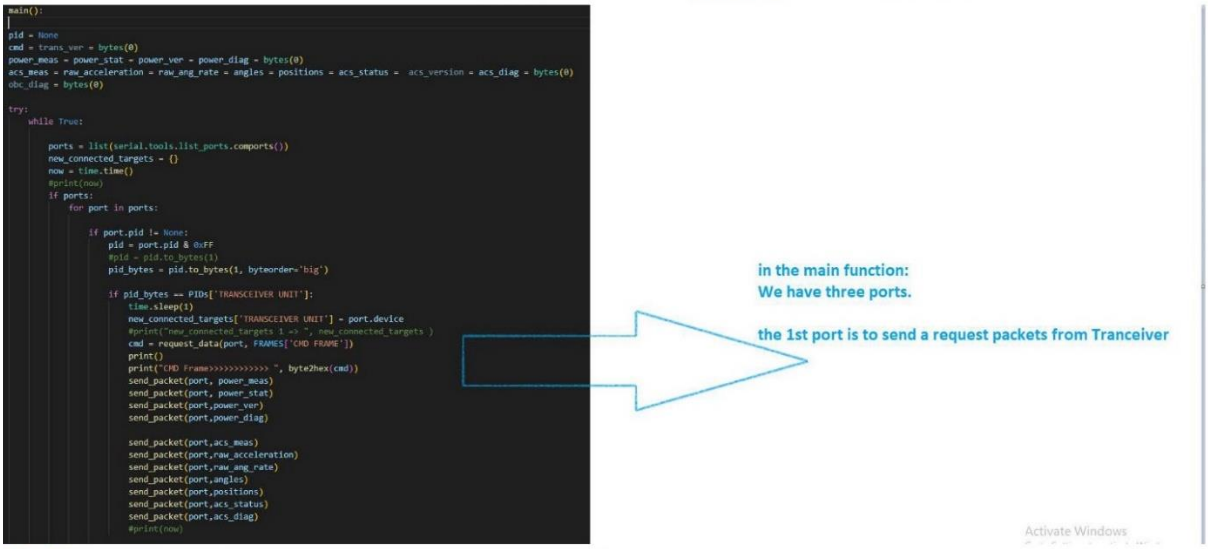
OBC Software Design

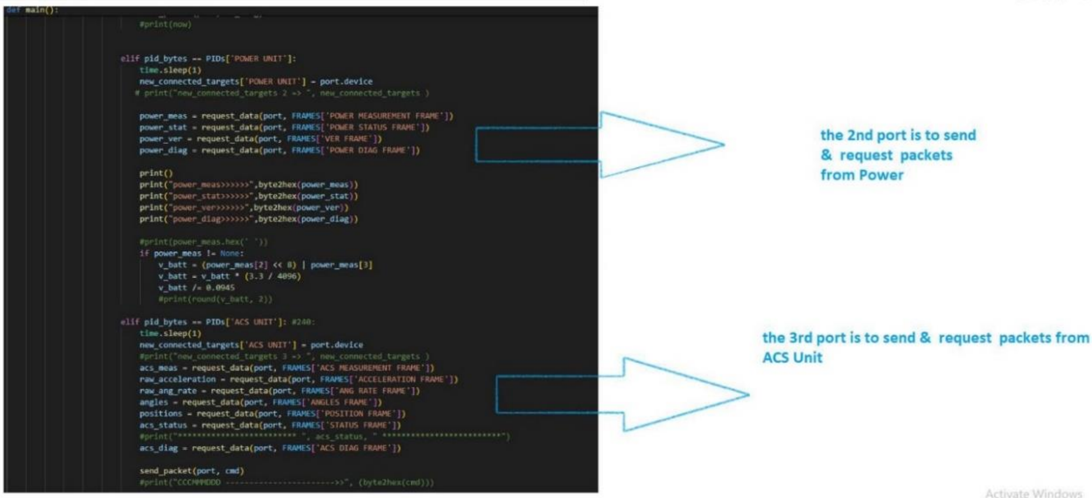


OBC to ACS Board Communication and Signal Handling (18.3.24)









The flowchart above explains how the flow process of the following code will work.

OBC Software Code (python)

OBC Rasberry (python)	OBC_Raspberry (python) Ver. 061223 OBC Raspberry (python) Ver. until April 24

Communication between OBC and ACS board

We need to test our communication between the OBC (raspberry py) and ACS board (STM32), we are using a USB as a communication protocol between these two.

Algorithm of operation for ACS board (STM32)

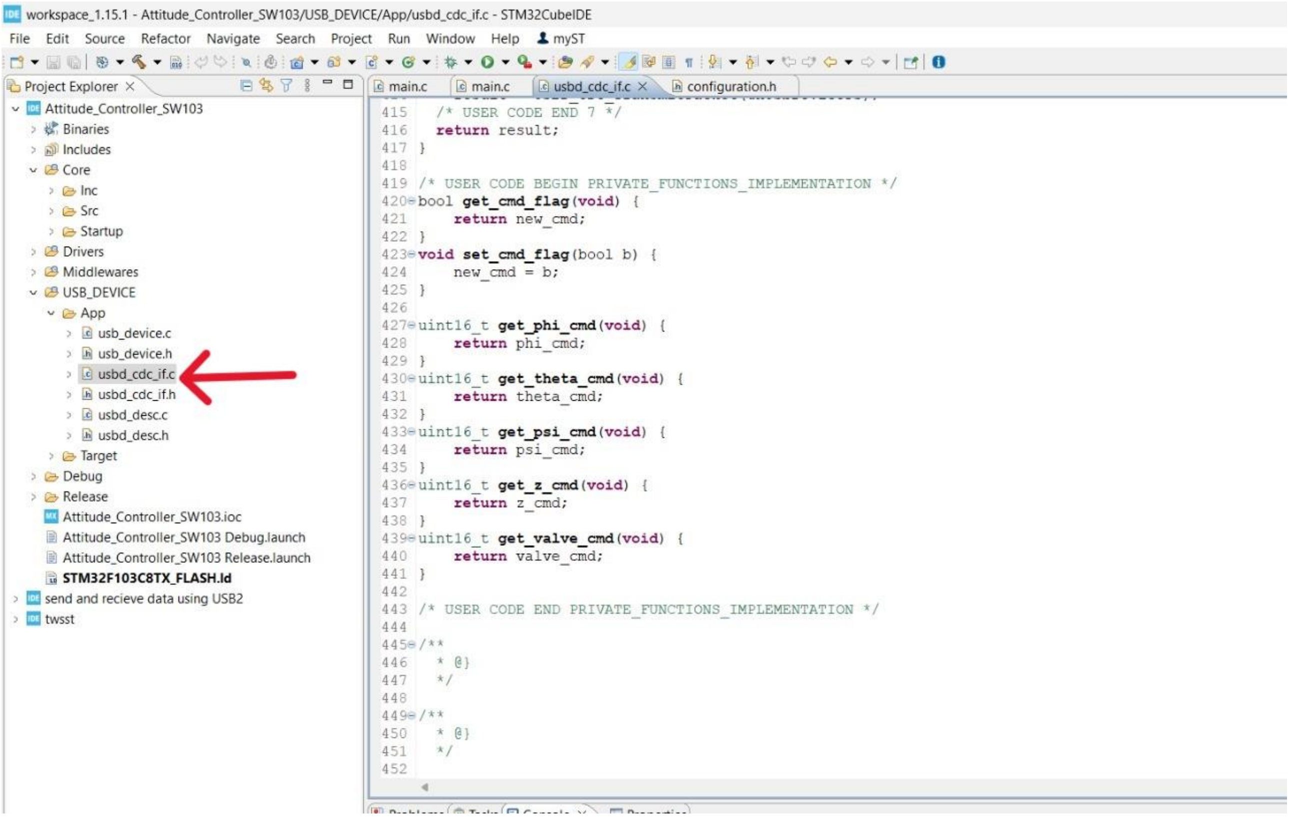


Figure 1: the path of the communication code

the code that is responsible for transmitting and receiving data can be reached by following Figure 1.

```
/* Private typedef -----------------------------------------------------------*/
/* Private define ------------------------------------------------------------*/
/* Private macro -------------------------------------------------------------*/

/* USER CODE BEGIN PV */
uint8_t rx_buffer[FRAME_LEN];
uint8_t tx_buffer[FRAME_LEN];
uint8_t cmd_data[FRAME_LEN - 2];

bool new_cmd = false;

uint16_t phi_cmd, theta_cmd, psi_cmd, z_cmd, valve_cmd;
/* Private variables ---------------------------------------------------------*/

```

Figure 2: defining some variables and their bits capacity

As we can see in Figure 2 We start by defining three variables with "unit8_t" which means each one of these variables is 8-bits (0000 0000) and we've created for each one an array of "FRAME_LEN" rows (cmd array length simply is the value of "FRAME_LEN" -2).

we define a variable called "new_cmd" as a bool format (0 or 1) and an initial value 0 (false) to simply use later to track if there is a new command or not.

finally, we've defined many variables as "unit16_t" which means each one of these variables is 16-bits (0000 0000 0000 0000).

"FRAME_LEN" is simply a variable that we've defined in another place ("configuration.h") (we've defined him here as 12). You can access the configuration of "FRAME_LEN" and many other variables by following the path in Figure 3.

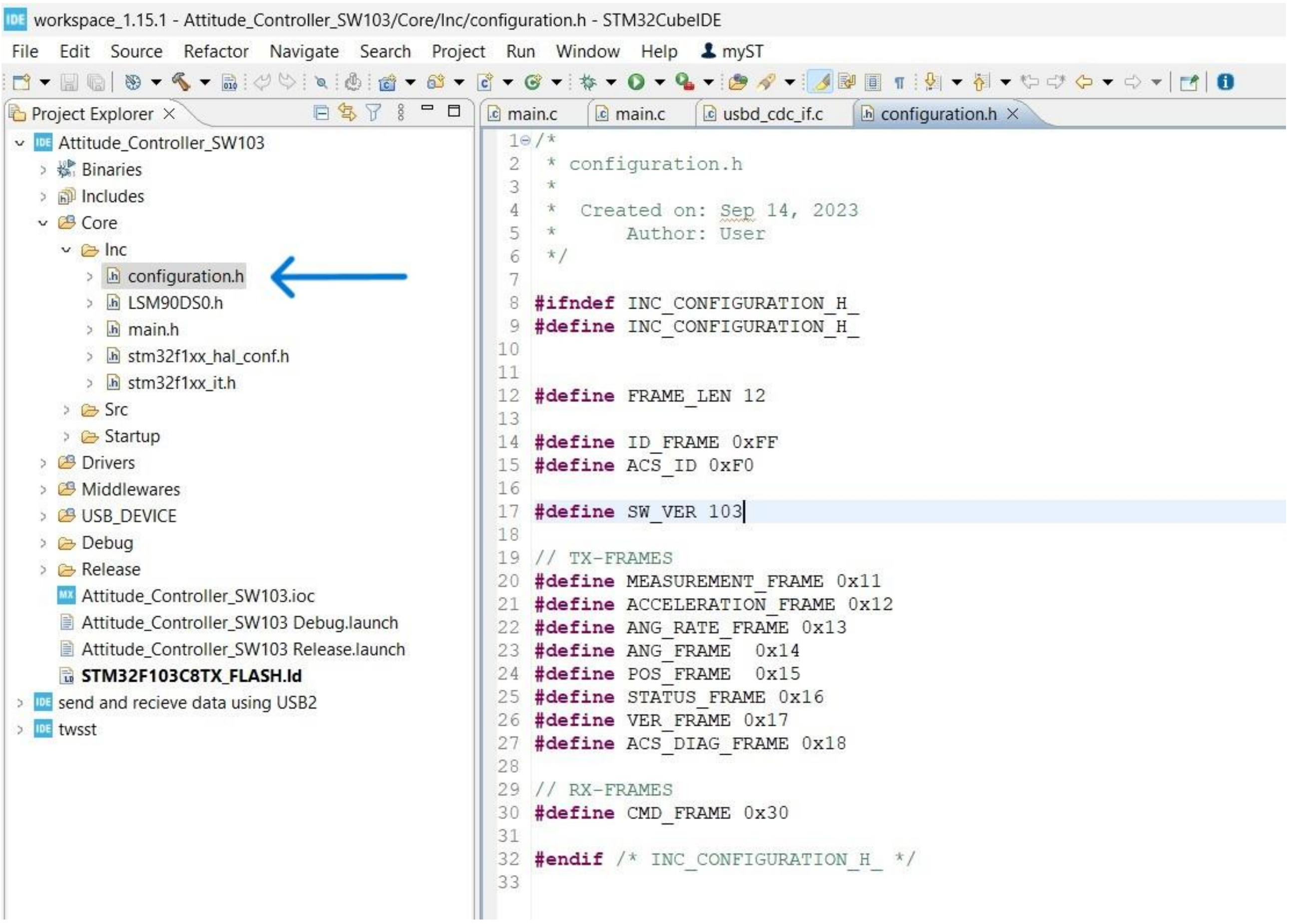


Figure 3: configurations of variables

After we know the value of the "FRAME_LEN" we can know how each of our main three variables (Figure2) will look (just to facilitate understanding), we will have 8bit each array:

1. rx_buffer [] = [00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000]
2. tx_buffer [] = [00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000]
3. cmd_data []= [00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000, 00000000]

Assume that rx_buffer [12] = [00000001, 00000010, 00001001, 00100100, 00010000, 10000000, 00011100, 00000000, 00000000, 00000000, 00000000, 00000000]

In this example rx_buffer [0] =[00000001], rx_buffer [3]= [00100100], rx_buffer [5]= [10000000], the same concept for the other variables (now if we mention them later the reader can imagine them easily).

Most of the time we will use the hexadecimal format instead of the binary to facilitate writing in this case how our variables will look like:

1. rx_buffer [12] = [0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00]
2. tx_buffer [12] = [0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00]
3. cmd_data [12-2]= [0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00]

```
main.c   main.c   usbd_cdc_if.c   configuration.h
  * @retval Result of the operation: USBD_OK if all operations are OK else USBD_FAIL
  */
static int8_t CDC_Receive_FS(uint8_t* Buf, uint32_t *Len)
{
  /* USER CODE BEGIN 6 */
    USBD_CDC_SetRxBuffer(&hUsbDeviceFS, &Buf[0]);
    USBD_CDC_ReceivePacket(&hUsbDeviceFS);
    memset(tx_buffer, '\0', FRAME_LEN);  // clear the tx buffer

    memset(rx_buffer, '\0', FRAME_LEN);  // clear the rx buffer
    uint8_t len = (uint8_t) *Len;

    memcpy(rx_buffer, Buf, len);  // copy the data to the buffer
    memset(Buf, '\0', len);   // clear the Buf also

    switch (rx_buffer[0]) {

    case MEASUREMENT_FRAME:
        tx_buffer[0] = MEASUREMENT_FRAME;
        tx_buffer[1] = 6;
        tx_buffer[2] = 2654 >> 8;
        tx_buffer[3] = 2654;
        tx_buffer[4] = 123;
        tx_buffer[5] = 122;
        tx_buffer[6] = 0xFF;
        tx_buffer[7] = 0xFF;
        break;

```

Figure 4: transmitting and receiving algorithm

The function in Figure 4 is the one that is responsible for transmitting and receiving data. the first 10 lines of this function are self-explanatory (check the comment in green for more details in Figure 4). The main thing here is that we will receive our data and this data will be saved at rx_buffer. according to the data that we will receive we will later transmit the required data.

we specify the first byte of our received and transmitted data as the ID (to know simply this data where it should go exactly). according to Figure 3, we can see the ID of our variables.

what the switch and case function do is simply a comparison between the "rx_buffer[0]" (ID send it to STM32) and each one of my frames, if ("rx_buffer[0]" == 0x11("MEASUREMENT_FRAME")) start this case if we start this case we won't look anymore to other cases, if not go to other case and do the same comparison (it's like using else if).

For example, if the OBC sends a signal that has ID=0x11 (00010001), we will enter the measurement case and fill the tx buffer with the required data (usually those data came from sensors).

According to Figure 4 our tx_buffer will look like that: tx_buffer[] = [0x11 ("MEASUREMENT_FRAME"),0x06, 0x14B, 0xA5E, 0x7B, 0x7A, 0xFF, 0xFF]. (about those values in the middle I am not sure yet if they are decimal, but they look like it). (2654>>8 we will explain this later on In the article).

```
    case VER_FRAME:
        tx_buffer[0] = VER_FRAME;
        tx_buffer[1] = 1;
        tx_buffer[2] = SW_VER;
        break;

    }
    CDC_Transmit_FS(tx_buffer, FRAME_LEN);
    return (USBD_OK);
  /* USER CODE END 6 */
}
```

Figure 5: sending data function (from stm32 to OBC)

After we fill the tx_buffer with the required data we need now to send this data to the Raspberry py, to do so we can see in Figure 5 we have a function that called "CDC_Transmit_FS" This function sends the information that is located in the tx_buffer to the OBC system through USB.

now all cases instead of CMD_Frame are responsible for sending data to the OBC system and they work the same way as the "MEASUREMENT_FRAME" (the OBC system requests data by sending the requirement ID and STM32 sends this data for him).

```
        break;

    case CMD_FRAME:
        new_cmd = true;
        phi_cmd =   (rx_buffer[2] << 8) | rx_buffer[3];
        theta_cmd = (rx_buffer[4] << 8) | rx_buffer[5];
        psi_cmd =   (rx_buffer[6] << 8) | rx_buffer[7];
        z_cmd =     (rx_buffer[8] << 8) | rx_buffer[9];
        valve_cmd = (rx_buffer[10] << 8 | rx_buffer[11]);
        break;

    case VER_FRAME:
        tx_buffer[0] = VER_FRAME;
        tx_buffer[1] = 1;
```

Figure 6: CMD frame

To start the CMD_frame it will be the same process as before (like the other frames), but the difference here is that this frame is the only one that receives data, we use this frame to apply a manual control to our system.

As we can see in Figure 6, we first change the variable new_cmd to true to indicate that there is a new commend.

All the variables here can store up to 16 bits, what we've done in the code that's we've taken two arrays from the rx_buffer and merged them to have a 16-bit.

let's take an example to understand how exactly each function worked here. assume that rx_buffer [] = [00110000 (0x30), 00000010, 00001001, 00100100, 00010000, 10000000, 00011100, 11000000, 01001001, 10000001, 00110000, 00001000]. (the first array represents the ID of the CMD_FRAME)

1. Extract Values from rx_buffer: rx_buffer[2]: 00001001 - rx_buffer[3]: 00100100

2. Left Shift Operation: rx_buffer[2] << 8 : 00001001 << 8 → 00001001 00000000

3. Bitwise OR Operation (or is like summation): rx_buffer[2] << 8 | rx_buffer[3]: 00001001 00000000 OR 00000000 00100100 -----------------> 00001001 00100100

- Result: phi_cmd = 00001001 00100100 (binary) / 0x0924 (hex) / 2340 (decimal)

the same concept goes for other variables of the CMD frame.

```c

/* USER CODE BEGIN PRIVATE_FUNCTIONS_IMPLEMENTATION */
bool get_cmd_flag(void) {
    return new_cmd;
}
void set_cmd_flag(bool b) {
    new_cmd = b;
}

uint16_t get_phi_cmd(void) {
    return phi_cmd;
}
uint16_t get_theta_cmd(void) {
    return theta_cmd;
}
uint16_t get_psi_cmd(void) {
    return psi_cmd;
}
uint16_t get_z_cmd(void) {
    return z_cmd;
}
uint16_t get_valve_cmd(void) {
    return valve_cmd;
}

/* USER CODE END PRIVATE_FUNCTIONS_IMPLEMENTATION */

/**
  * @}
```

Figure 7: return commends functions

According to Figure 7, we've added many functions to get the CMD value for each variable (phi, theta, psi,....).

"get_cmd_flag" is the function that tells you whether there is a new command or not.

st_cmd_flag allows you to change the value of a new_cmd we use this to do a reset after finishing applying the commend.

```c
  /* Infinite loop */
  /* USER CODE BEGIN WHILE */
  while (1)
  {

    /* USER CODE END WHILE */

    /* USER CODE BEGIN 3 */

      if (get_cmd_flag() ){
          set_cmd_flag(0);

          valve_cmd = get_valve_cmd();

          HAL_GPIO_WritePin(RELAY_1_GPIO_Port, RELAY_1_Pin, (valve_cmd & 0x01));
          HAL_GPIO_WritePin(RELAY_2_GPIO_Port, RELAY_2_Pin, (valve_cmd & 0x02));
          HAL_GPIO_WritePin(RELAY_3_GPIO_Port, RELAY_3_Pin, (valve_cmd & 0x04));
          HAL_GPIO_WritePin(RELAY_4_GPIO_Port, RELAY_4_Pin, (valve_cmd & 0x08));
//        HAL_GPIO_WritePin(RELAY_5_GPIO_Port, RELAY_5_Pin, !(valve_cmd & 0x10));
          HAL_GPIO_WritePin(RELAY_6_GPIO_Port, RELAY_6_Pin, (valve_cmd & 0x20));
//        HAL_GPIO_WritePin(RELAY_7_GPIO_Port, RELAY_7_Pin, !(valve_cmd & 0x40));
//        HAL_GPIO_WritePin(RELAY_8_GPIO_Port, RELAY_8_Pin, !(valve_cmd & 0x80));
         // HAL_GPIO_WritePin(RELAY_9_GPIO_Port, RELAY_9_Pin, !(valve_cmd & 0x80));

          HAL_GPIO_TogglePin(LED_GPIO_Port, LED_Pin);
      }
```

Figure 8: Main function

As we can see in Figure 8 we start with the condition to know I there is a CMD or not ("get_cmd_flag"), if there is we will reset the CMD flag (set_cmd_flag(0)) which means we've entered no command mode (to be ready to receive a new command again), after that we've requested for the CMD data that are related to the valves (get_valve_cmd) and save it in the valve_cmd.

we will turn each relay ON and OFF according to the valve_cmd. to understand how (valve_cmd & 0x01) work we will take an example.

assume that valve cmd = 00000000 01001001, '&' work as multiplication and the 0x01 is represented in binary in this form: 00000001.

00000000 01001001 AND 00000000 00000001 --------> 00000000 00000001 which means 1 so this relay will turn on.

Relay 2: (0x02: 00000010), 01001001 AND 00000010--------> 00000000 which means 0 so this relay will turn OFF.

the same goes for all other relays.

relay	hexa nb	binary nb
relay 1	0x01	00000000 00000001
relay 2	0x02	00000000 00000010
relay 3	0x04	00000000 00000100
relay 4	0x08	00000000 00001000
relay 5	0x10	00000000 00010000
relay 6	0x20	00000000 00100000

relay 7	0x40	00000000 01000000
relay 8	0x80	00000000 10000000
relay 9	0x100	00000001 00000000

Table 1: the nb of each relay and its representation in binary and hexadecimal

According to Figure 8 and Table 1, we know that each bit from our 16-bit represents a relay if it's 0 it will turn off, and if 1 it will turn on. the LSB bit represents relay 1 and the MSB represents relay 16.

valve cmd = (relay 16)MSB 00000000 01001001 LSB(relay 1) those 16-bit represents the relays that will turn on and off, in this example relays: 1,4, and 7 will turn on.

communication test :

because we are working at a small scale our OBC won't take command from the ground station so we will add the command manual inside the code, figure 9 will show us this command.

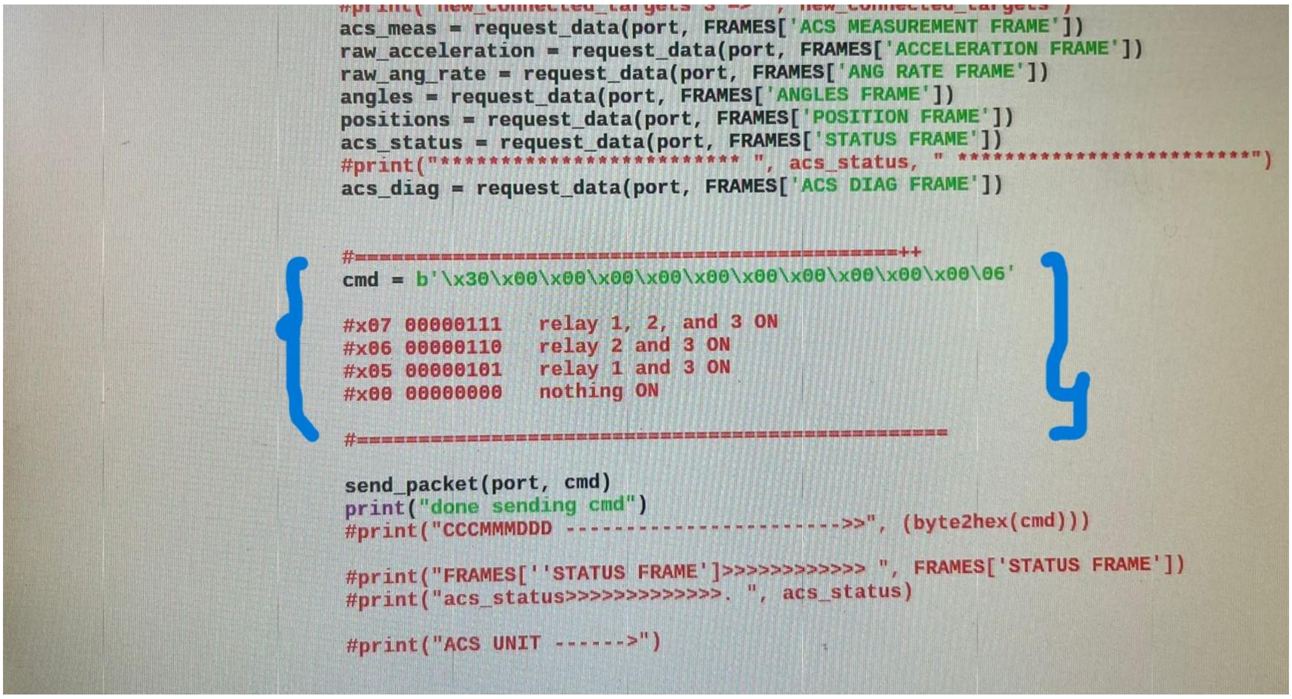


Figure 9: Raspberry Py manual command

In Figure 9 we have the first byte x30, which is the ID ("CMD frame"), we have an explanation inside Figure 9 to tell us if we change the final byte what will happen.

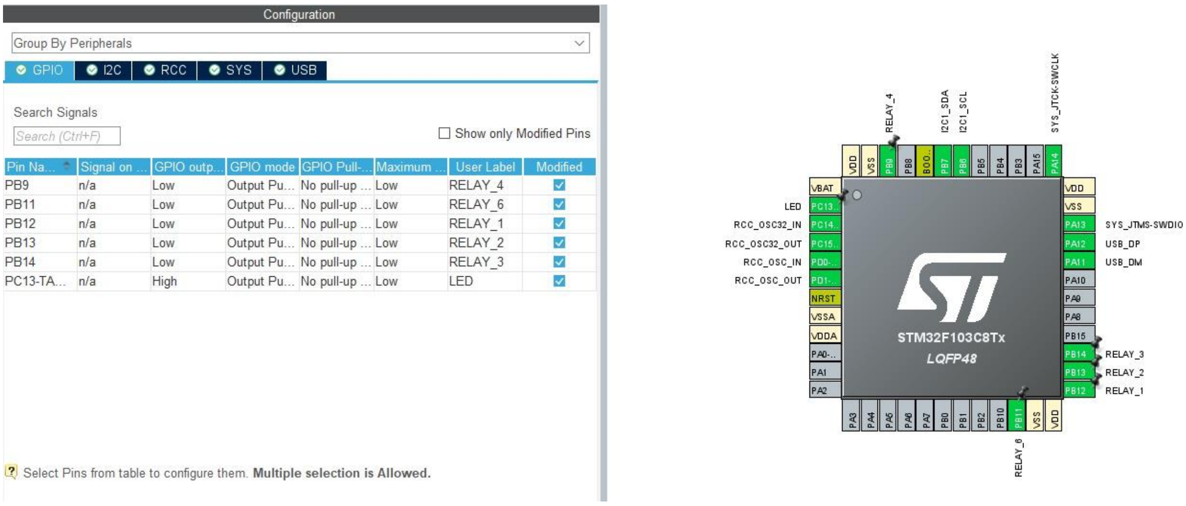


Figure 10: STM32 pin configuration

For this test we will only control 3 relays, as we can see in Figure 10 we've configured the required relays and set them initially low. We've defined the LED to let it blink when we've received comments (for visualization) (The LED pin is high because when we set it high = the LED is OFF and vice versa).

Now we are ready to apply our test. We will apply 4 tests each one turning on and off different relays, we will be moving in the same order as the comment in Figure 9, and Table 2 will show us more details about our test.

TEST	**relay 1**	**relay 2**	**relay 3**	**cmd hexa representation**	**binary representation (last 2 byte)**
1	ON	ON	ON	b'\x30\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x**07**'	00000000 00000**111**
2	**OFF**	ON	ON	b'\x30\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x**06**'	00000000 00000**11**0
3	ON	**OFF**	ON	b'\x30\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x**05**'	00000000 00000**1**0**1**
4	**OFF**	**OFF**	**OFF**	b'\x30\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x**00**'	00000000 00000000

Table 2: tests details

Test 1,2,3, and 4 results are represented in Figures 11,12,13 and 14

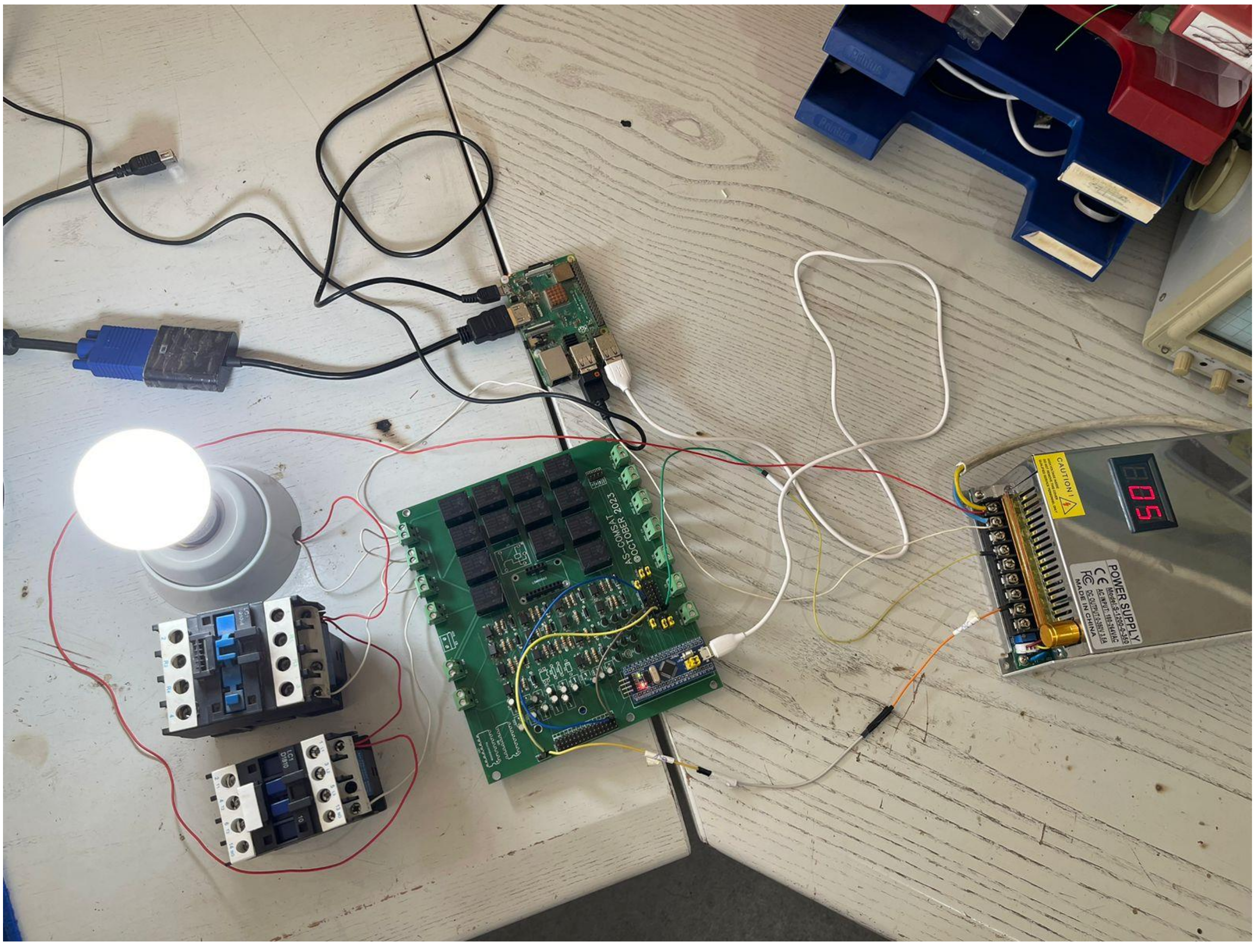


Figure 11: Result of test 1

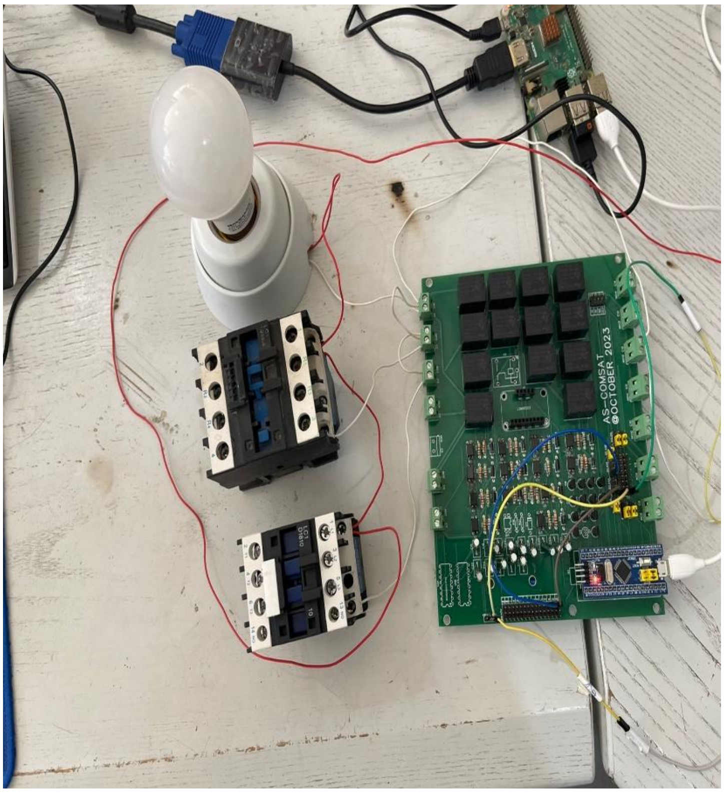


Figure 12: Result of test 2

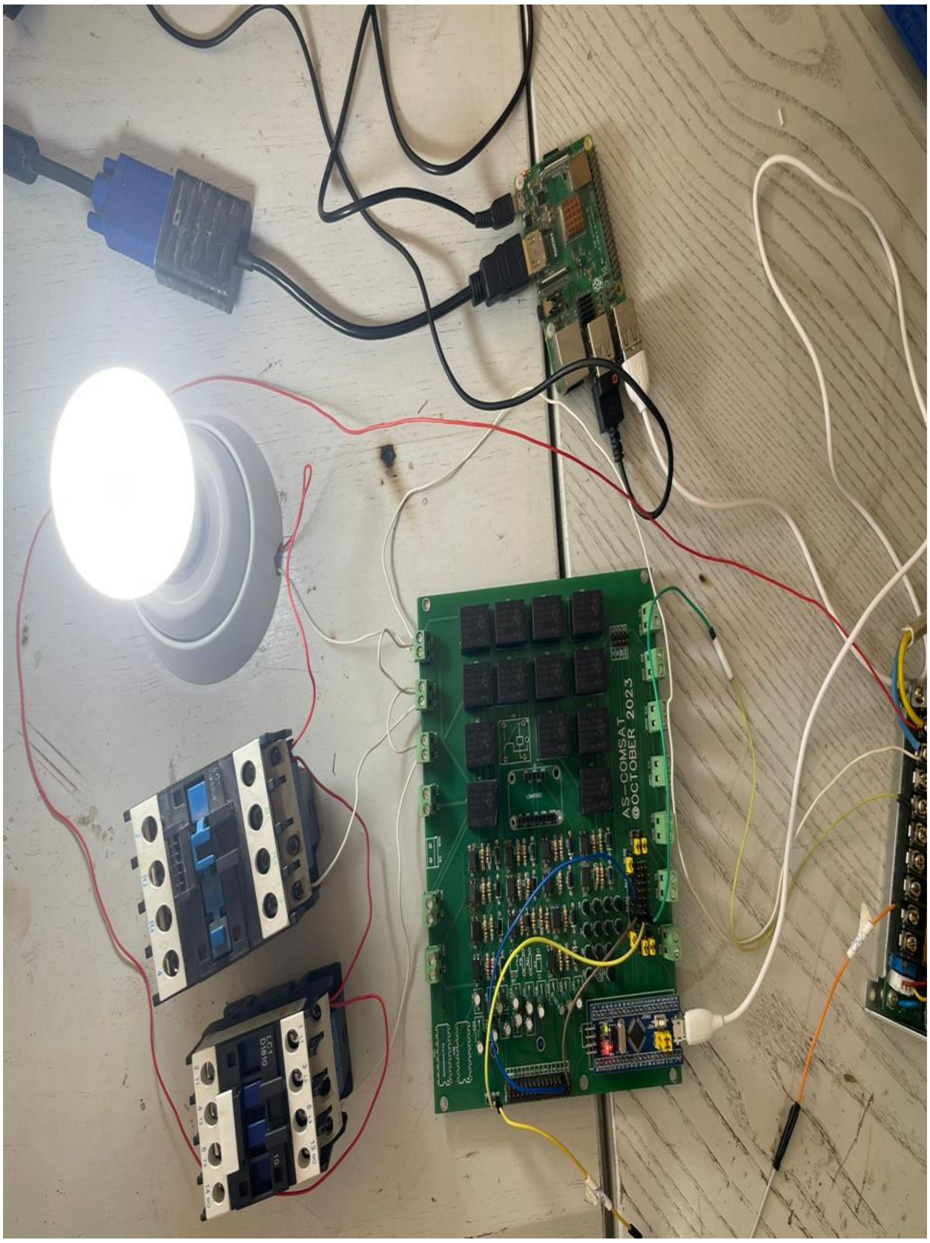


Figure 13: Result of test 3

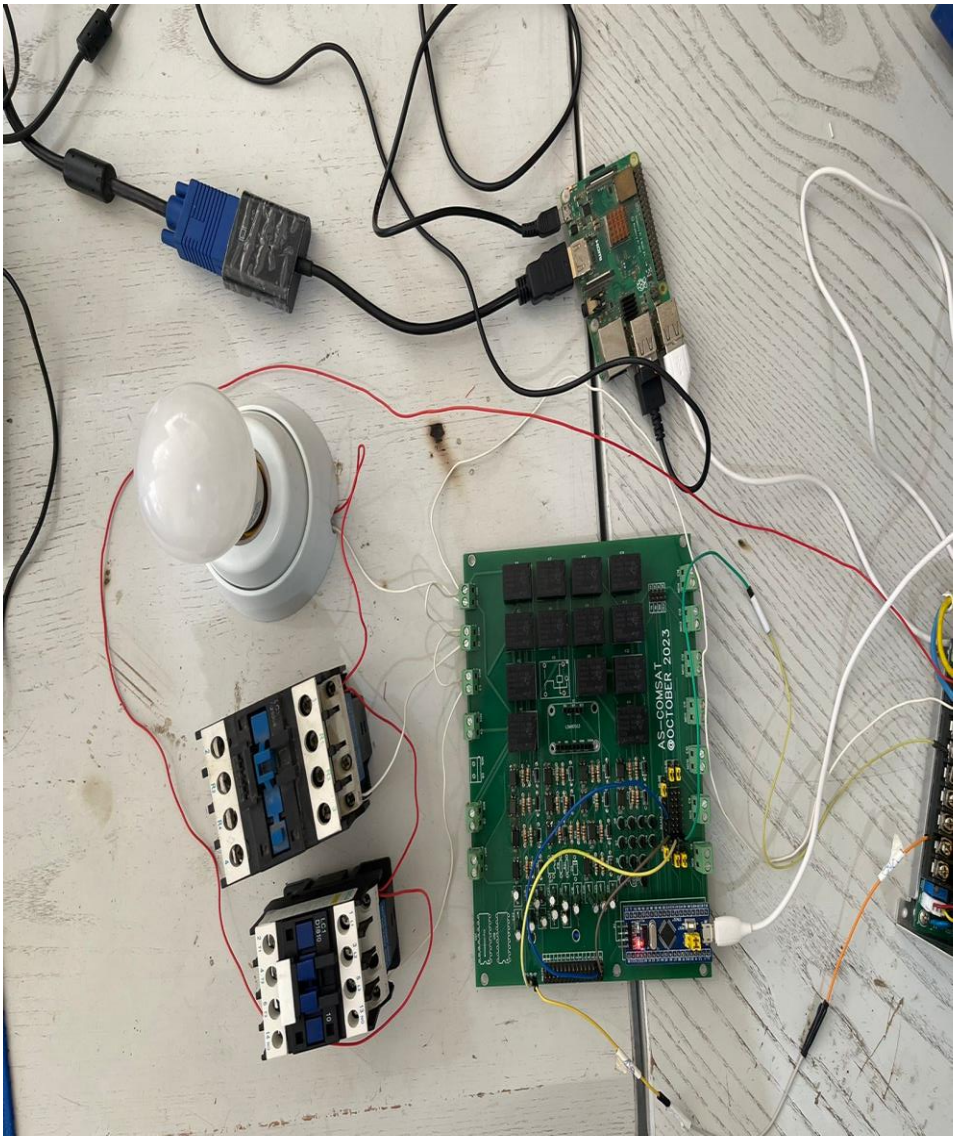


Figure 14: Result of test 4

videos of each test and their result: communication_Tests_Results 40624

STM 32 code (used in this test): Attitude_Controller_communication_test_SW103 40624

raspberry py code (used in this test): OBC_Raspberry_communication_test (python) 40624

2.9.24:

https://aecenar.com/index.php/downloads/send/16-ics/1748-as-comsat-acs-obc-poster290824-pptx (Poster pptx file)

AECENAR TECDA
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region

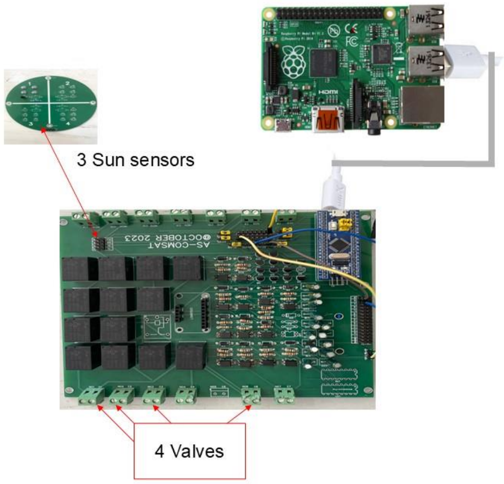
بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

Anadolu-Sham COMSAT

AS-COMSAT ACS & OBC Design

The attitude control system is responsible for receiving the high-level navigation commands and stabilize the satellite at a specific orientation. The On-Board Computer (OBC) is responsible for receiving the data from the Attitude Control System (ACS) Unit then do some calculations and send the updated status of the valve to the ACS to command. The system uses a raspberry pi as the OBC, and it is coded in python.

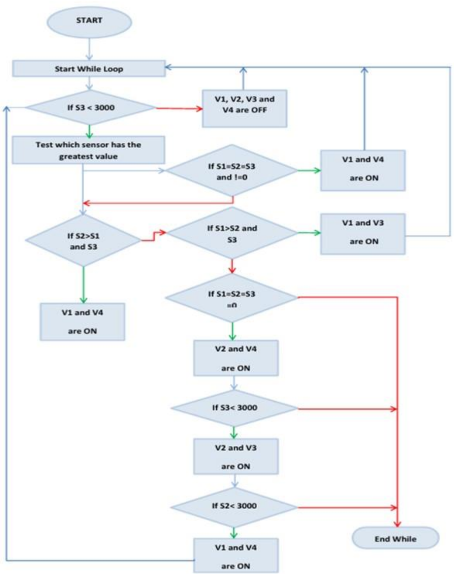
Sun Sensor, Analog signal, goes to the STM32 on the ACS Actuator board and then immediately to the Raspberry Pi. On the Raspberry Pi the python ACS Algorithms determines, which valves shall open. Then Raspberry Pi sends an appropriate signal to the ACS Actuator Board, which is obviously a list contains zeros or one indicates the status of the four valves. Then the ACS Actuator Boards sends a signal to the valves.

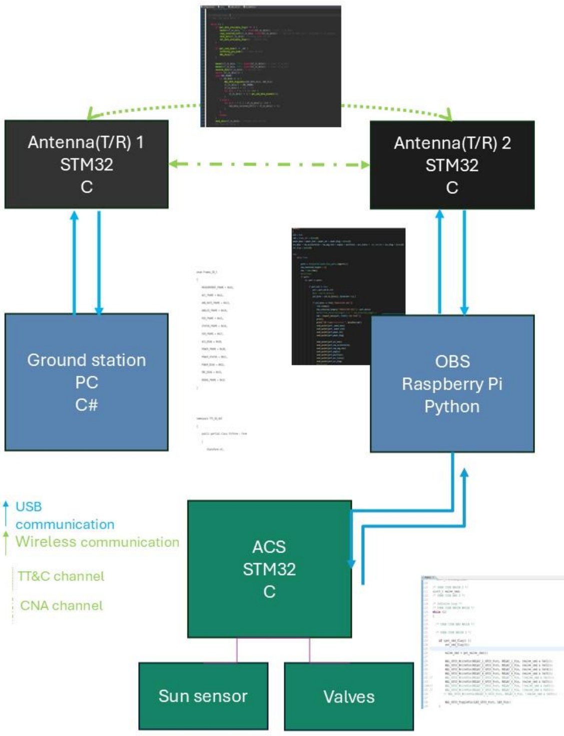


This action of opening or closing the valve is happened through relays on the ACS board.

The Raspberry Pi is connected to the stm32 through a USB cable.

The STM32 should update the Raspberry Pi by the status of the valves when it sends a request.





Layla Taleb @aecenar_as_comsat/august 2024

12 AS-COMSAT_1 LEO to GEO Orbit Change Testrig

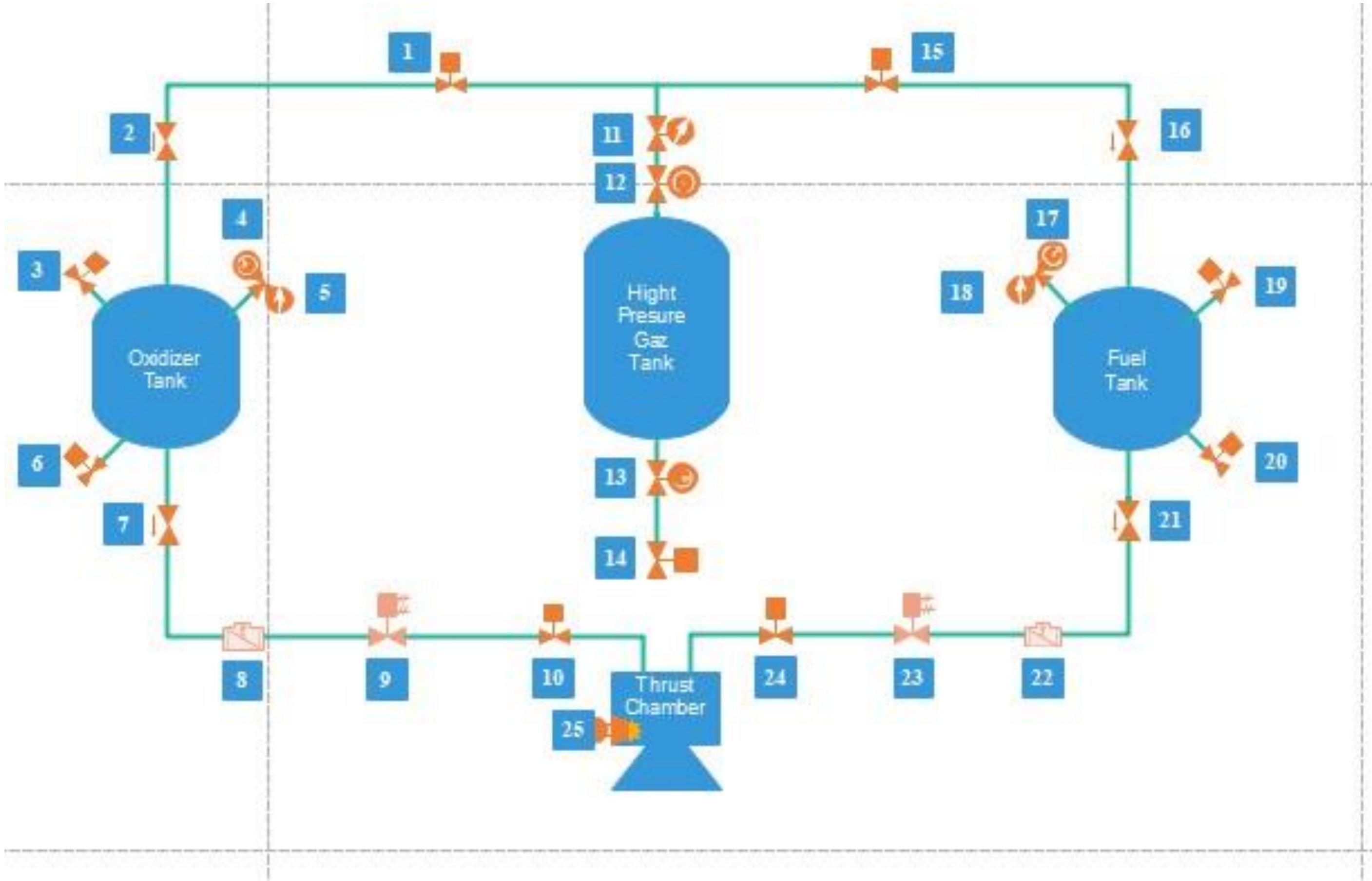
[AS-COMSAT_1 LEO to GEO Orbit Change Teststand](https://aecenar.com/index.php/companies/as-comsat/as-comsat-platforms-devices/ics-iap-sat/as-comsat-1-leo-to-geo-orbit-change-module/as-comsat-1-attitude-control-system-testrig) (https://aecenar.com/index.php/companies/as-comsat/as-comsat-platforms-devices/ics-iap-sat/as-comsat-1-leo-to-geo-orbit-change-module/as-comsat-1-attitude-control-system-testrig)

12.1 System Design & Mechanical Realization & Testing

12.1.1 Propulsion System

Picture 1: Testrig For Satellite Propulsion

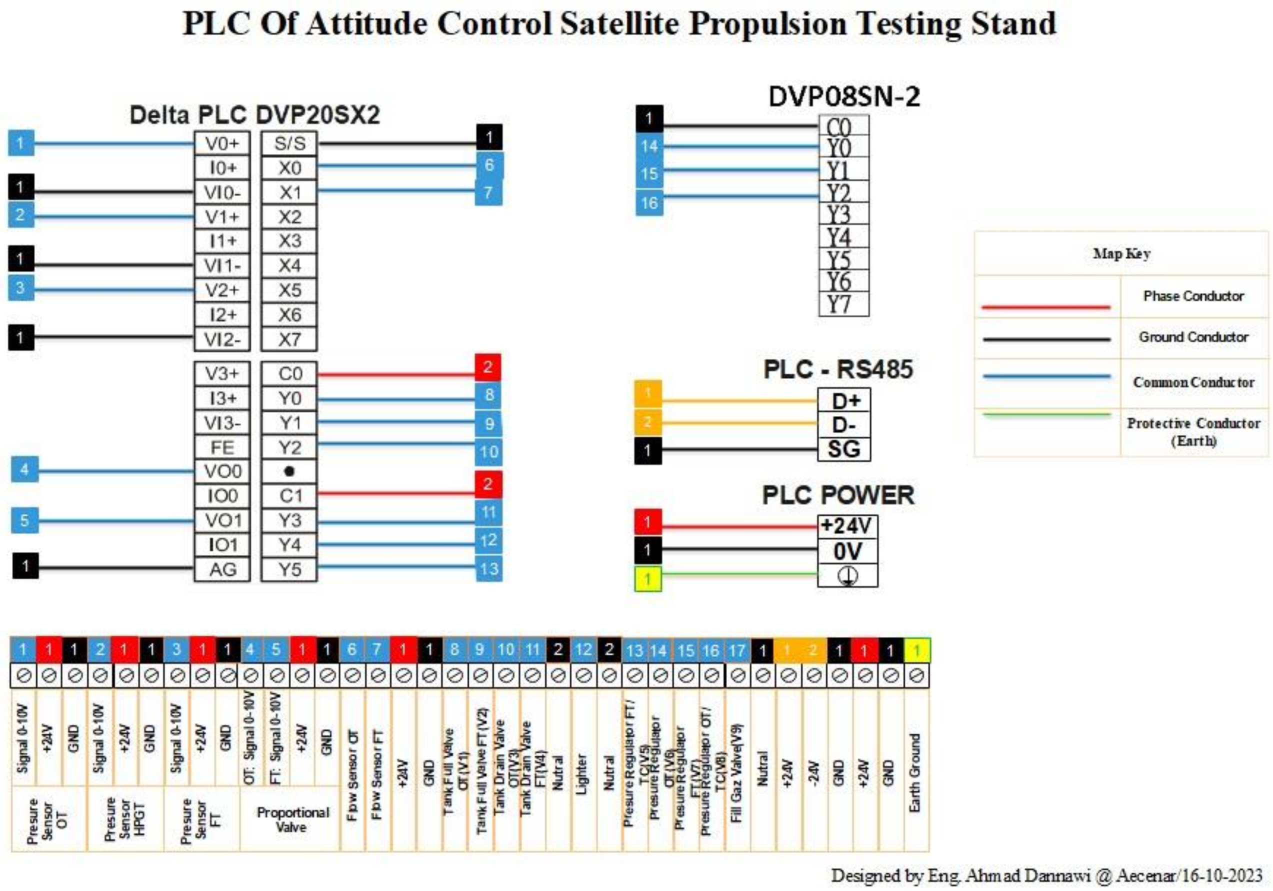
12.1.2 Propulsion System Design



Number	Name
1	Presure Regulator For The Oxidizer Tank
2	Check Valve The Input Of The Oxidizer Tank
3	Tank Full Valve For The Oxidizer Tank
4	Presure Gonch For The Oxidizer Tank
5	Presure Sensor For The Oxidizer Tank
6	Tank Drain Valve For The Oxidizer Tank
7	Check Valve Output For The Oxidizer Tank
8	Flow Sensor For The Oxidizer Tank
9	Propotional Valve For The Oxidizer Tank
10	Presure Regulator Between Oxidizer Tank And Thrust Chamber
11	Presure Sensor Hight Presure For The Gaz Tank
12	Presure Gonch For Hight Presure The Output Of The Gaz Tank
13	Presure Gonch For Hight Presure The Input Of The Gaz Tank
14	Fill Gaz Valve
15	Presure Regulator For The Fuel Tank
16	Check Valve The Input Of The Fuel Tank
17	Presure Gonch For The Fuel Tank
18	Presure Sensor For The Fuel Tank
19	Tank Full Valve For The Fuel Tank
20	Tank Drain Valve For The Fuel Tank
21	Check Valve The Output Of The Fuel Tank
22	Flow Sensor For The Fuel Tank
23	Propotional Valve For The Fuel Tank
24	Presure Regulator Between Fuel Tank And Thrust Chamber
25	Lighter

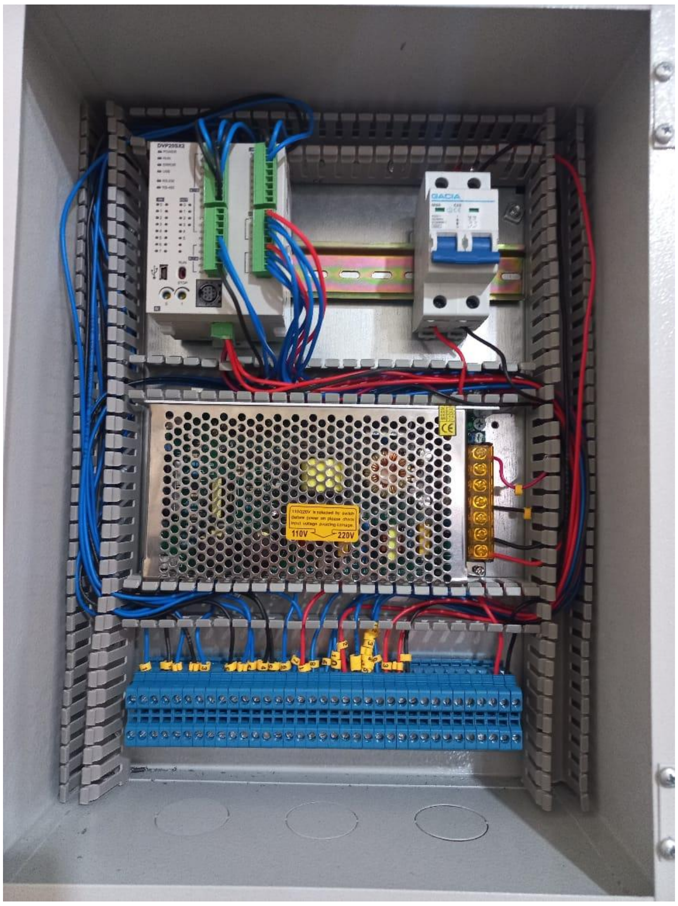
Picture 2: System of All Sensor's for the Testrig For Satellite Propulsion

12.1.3 Panel Wiring Diagram



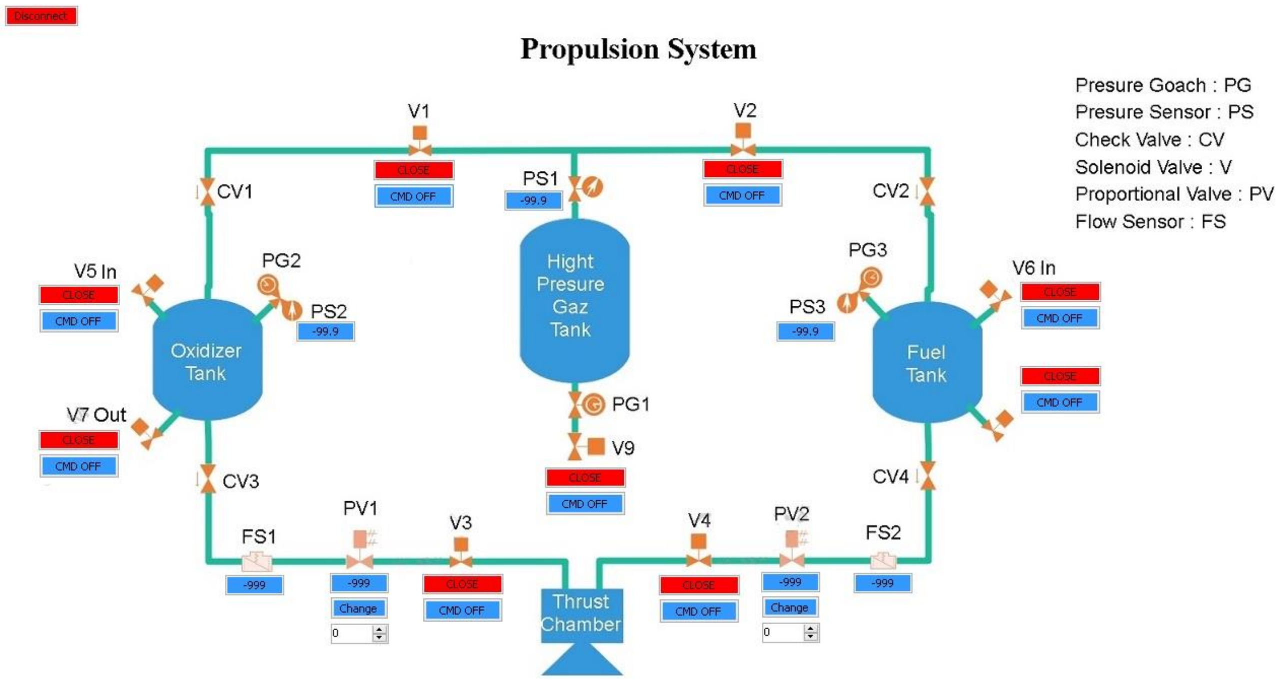
Picture 3: Connection of PLC of attitude control satellite propulsion teststand

12.1.4 PLC Panel



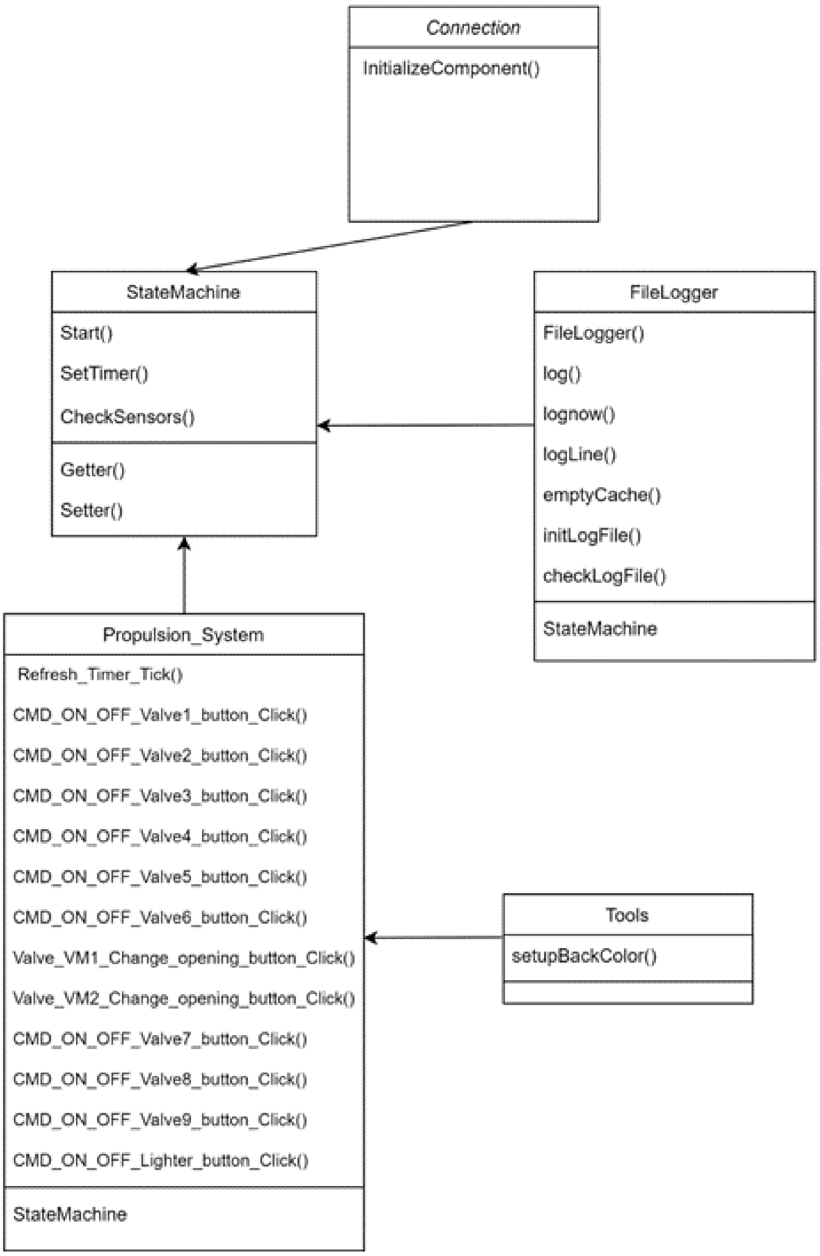
Picture 4: PLC Panel of teststand

12.1.5 Graphical User Interface



Picture 5: GUI of Teststand by PLC.

12.1.5.1 ACS PLC Teststand GUI (Diagrams and Code)



12.1.6 Modbus Address - PLC Points interface

Testing Stand Propulsion	Current addresses inside (GUI)	Modbus-addresses (PLC)	Physical Address (PLC)	Remarque
Presure sensore 1	4095	4096	D0	// Presure Sensor For The Oxidizer Tank
Presure sensore 2	4097	4098	D2	// Presure Sensor For The Fuel Tank
Presure sensore 3	4099	4100	D4	// Presure Sensor Hight Presure For The Gaz Tank
Proportional Valve 1	4101	4102	D6	// Propotional Valve For The Oxidizer Tank
Proportional Valve 2	4103	4104	D8	// Propotional Valve For The Fuel Tank
Flow Meter 1	4109	4110	D14	// Flow Sensor For The Oxidizer Tank
Flow Meter 2	4115	4116	D20	// Flow Sensor For The Fuel Tank
ON/OFF Valve 1	2047	2048	M0	// Presure Regulator For The Oxidizer Tank
ON/OFF Valve 2	2048	2049	M1	// Presure Regulator For The Fuel Tank
ON/OFF Valve 3	1282	1283	Y2	// Presure Regulator Between Oxidizer Tank And Thrust Chamber
ON/OFF Valve 4	1283	1284	Y3	// Presure Regulator Between Fuel Tank And Thrust Chamber
ON/OFF Valve 5	1284	1285	Y4	// Tank Full Valve For The Oxidizer Tank
ON/OFF Valve 6	1284	1285	Y4	// Tank Full Valve For The Fuel Tank
ON/OFF Valve 7	1284	1285	Y4	// Tank Drain Valve For The Oxidizer Tank
ON/OFF Valve 8	1284	1285	Y4	// Tank Drain Valve For The Fuel Tank
ON/OFF Valve 9	1285	1286	Y5	// Fill Gaz Valve

Picture 6: table of modbus

12.1.6.1 Testrig of the propulsion stand system at 30/11/2023

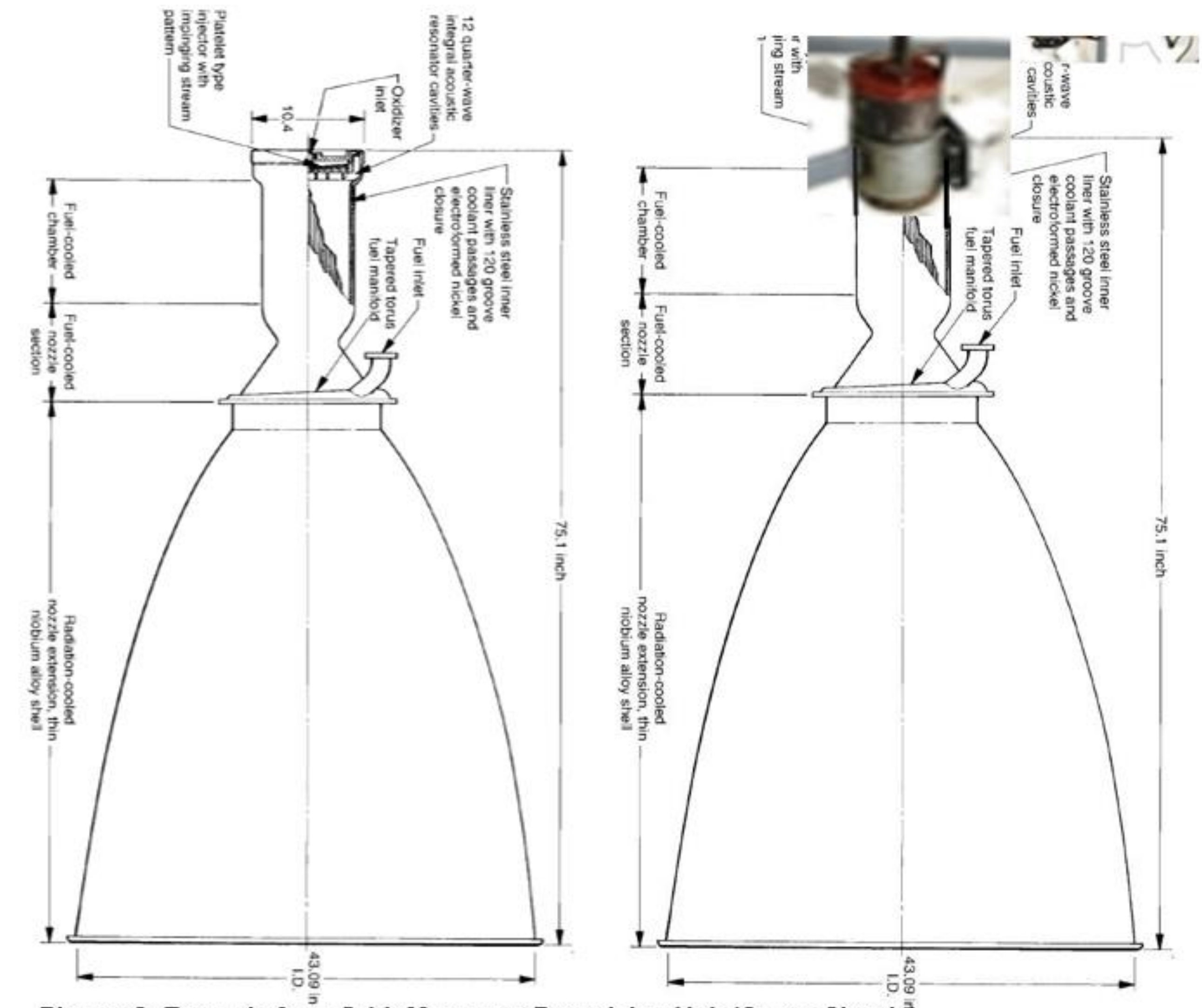
[Testing of the propulsion system (mp4 file).]

Picture 7: This figure show the important and unimportant enrichment point.

[Graphical User Interface code (C#)]

[PLC Code]

[Modbus Address - PLC Points interface]

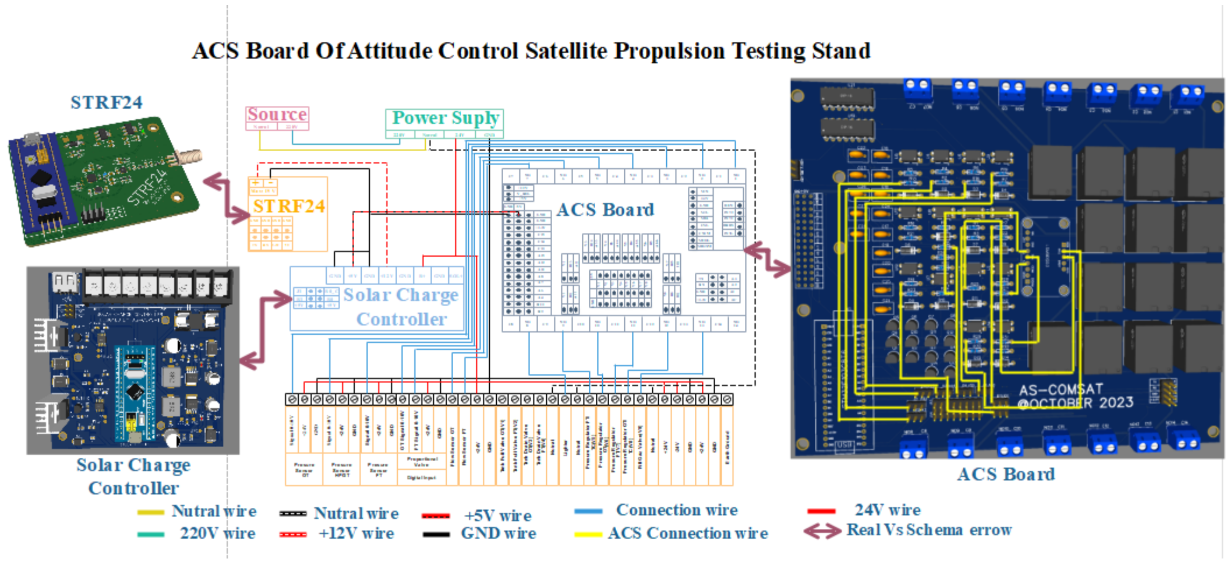


Picture 8: Example for a Orbit Maneuver Propulsion Unit (Space Shuttle).

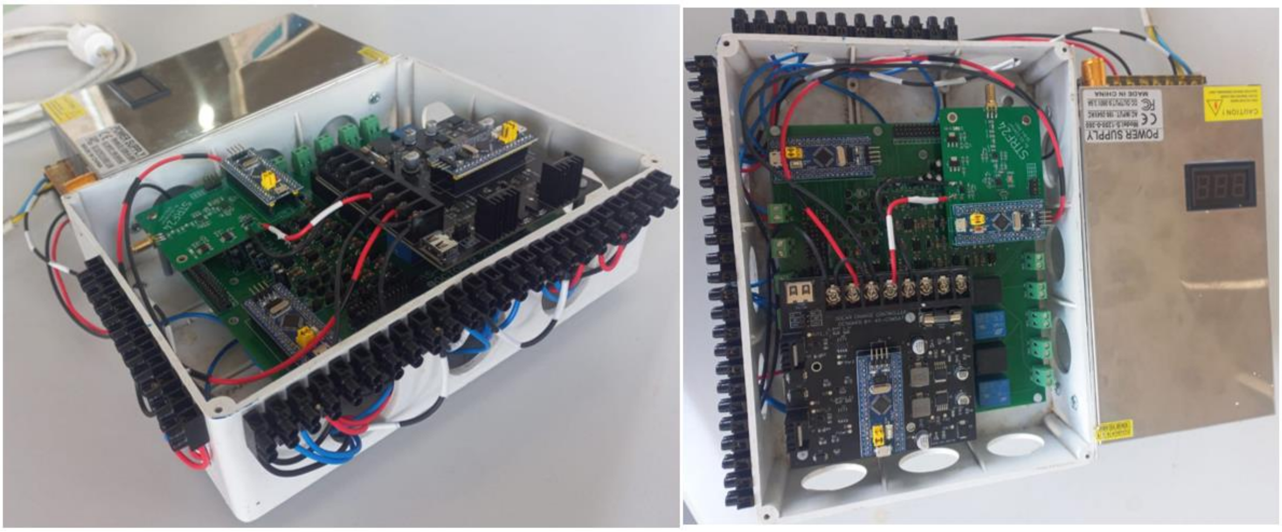
Simplified half-section of one of the two thrust chambers of the orbital maneuvering engines used on the Space Shuttle vehicle. Each develops a vacuum thrust of 6000 lbf (26,689 N) and delivers a minimum vacuum specific impulse of 310 sec, using nitrogen tetroxide and monomethyl hydrazine propellants at a nominal mixture ratio of 1.65 and a nominal chamber pressure of 128 psia. It is designed for 100 flight missions, a service life of 10 years, and a minimum of 500 starts. These engines provide the thrust for final orbit attainment, orbit circularization, orbit transfer, rendezvous, and deorbit maneuvers. The nozzle area ratio of 55:1. (Courtesy of Aerojet Propulsion Company.)

Picture 9: Fuel and oxidizer inlet to combustion chamber

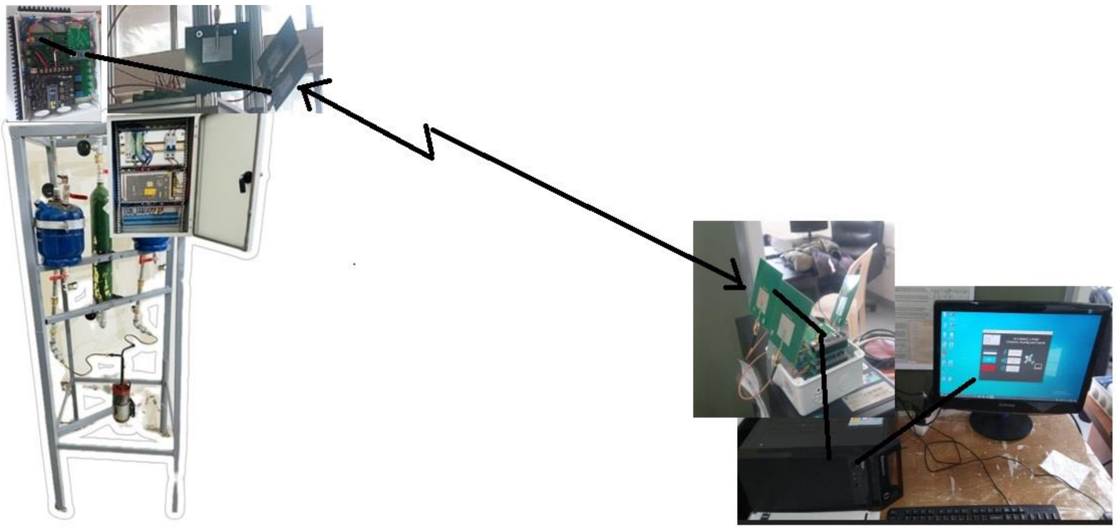
Picture 10 : Gazifier and Connecting Lines



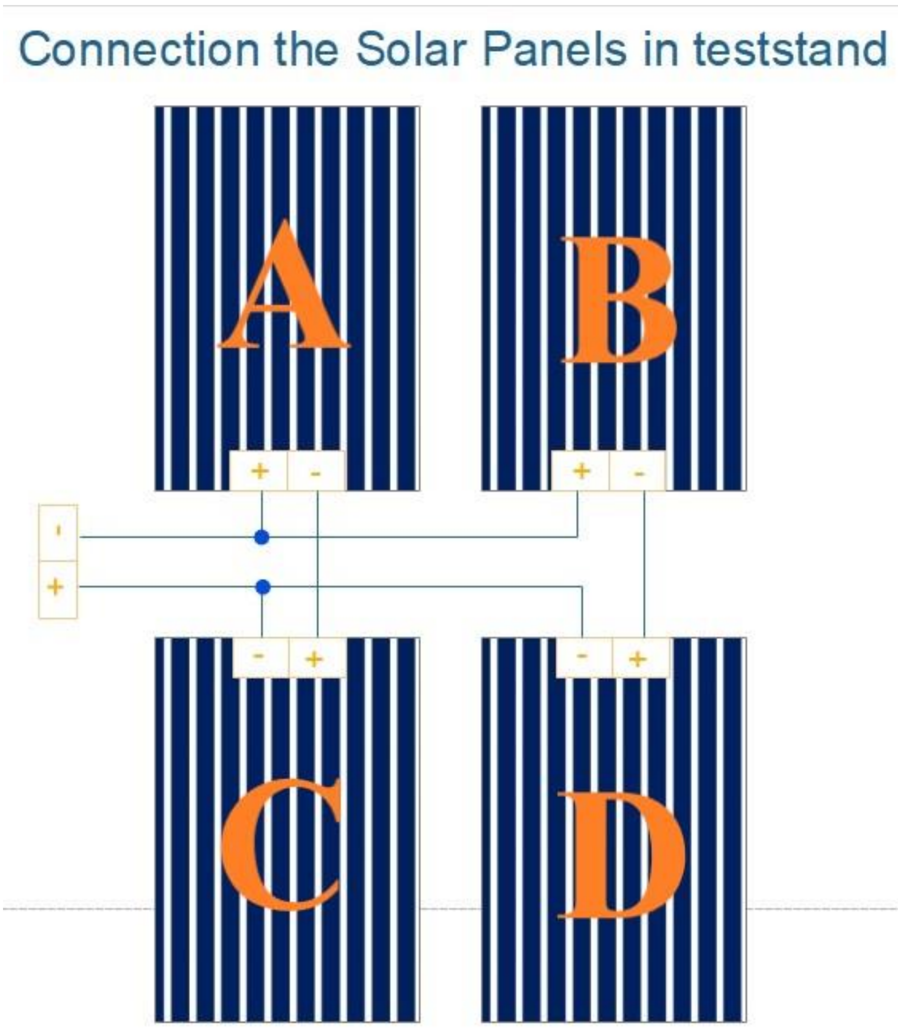
Picture 11: Connection of ACS Board for Satellite Attitude Control System (ACS) Teststand



Picture 12: ACS Panel of teststand

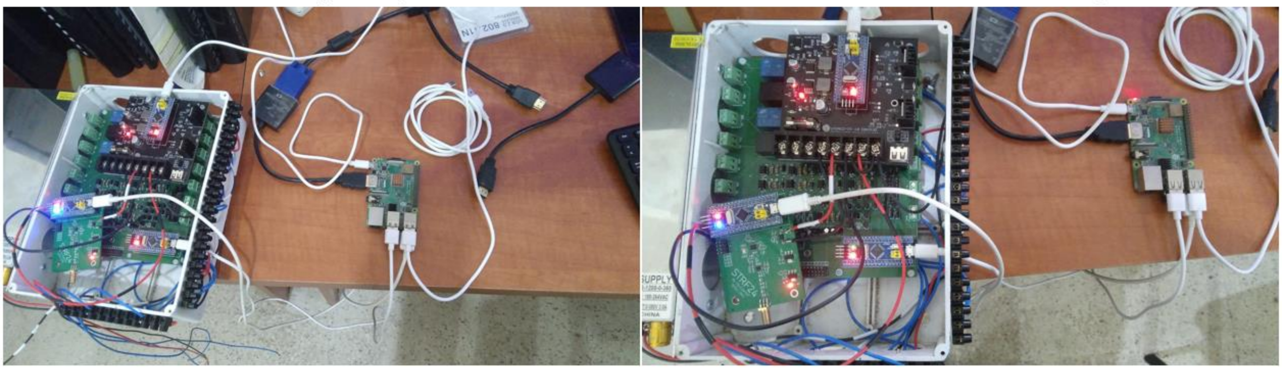


Picture 13: ACS Teststand controlled by TT&C GUI

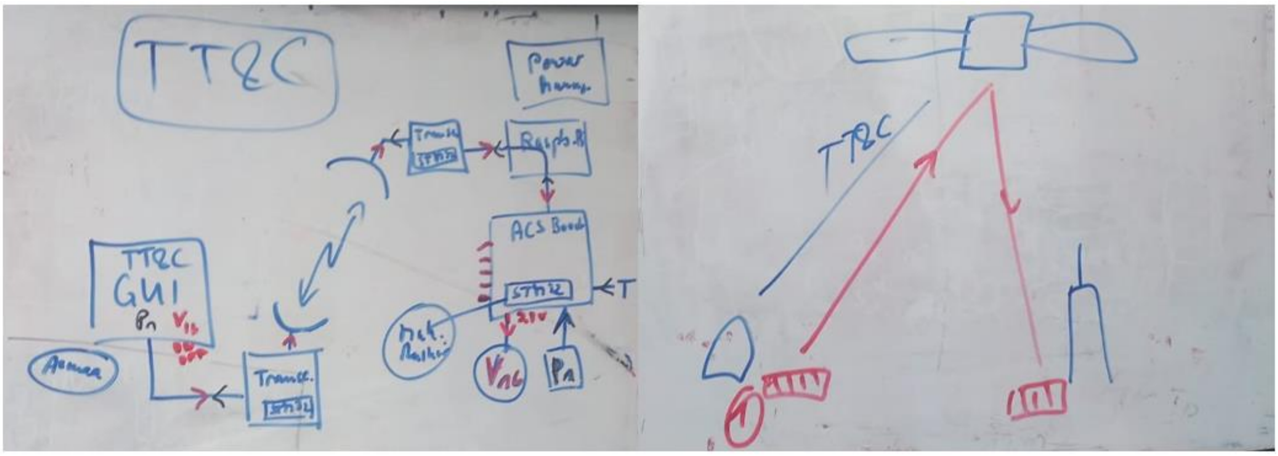


Picture 14: Connection the Solar Panels in Teststand, connected A and C in series, and B and D in series, and (A,C) with (B,D) in parallel.

Connected the Raspberry with ACS Board, Transvers Board, Power Charge Board

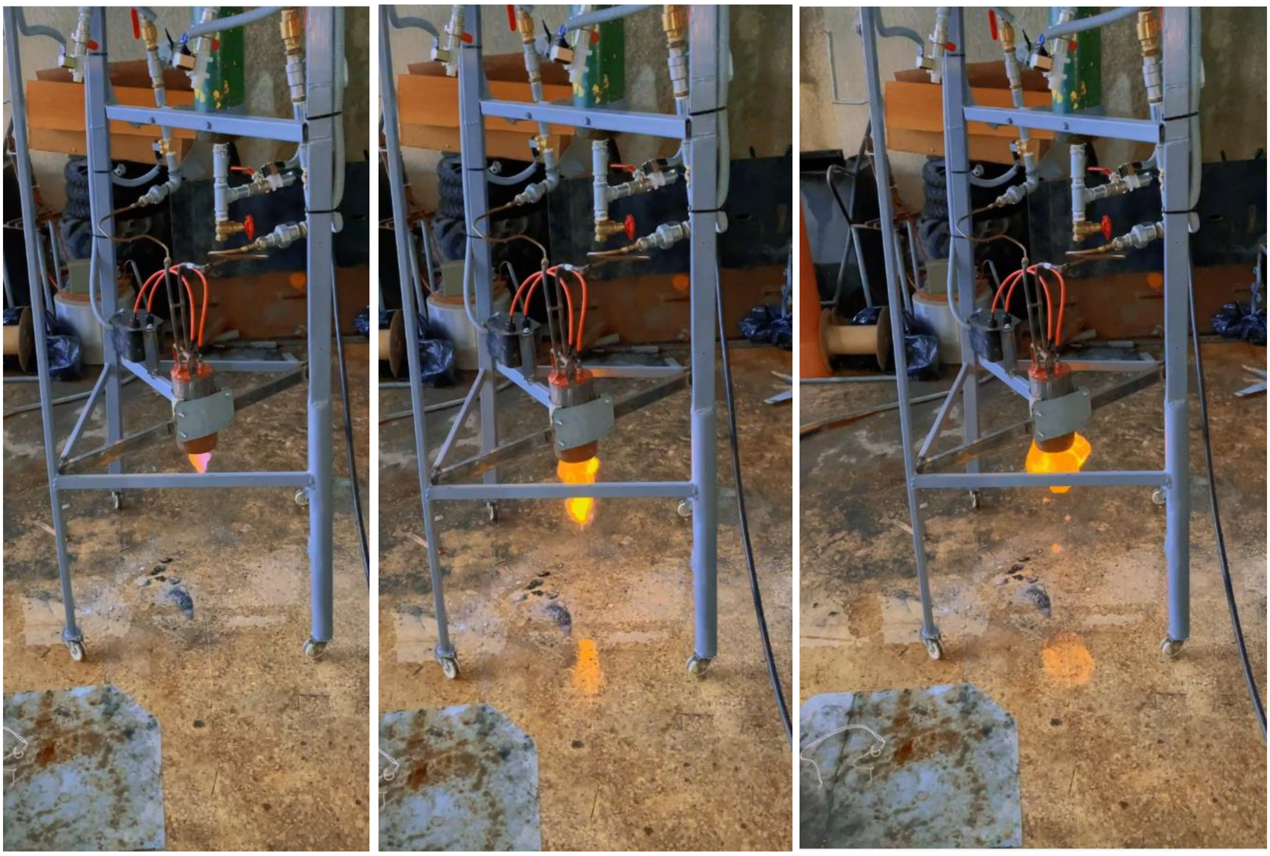


Picture 15: Connected the Raspberry with ACS Board, Transvers Board, Power Charge Board.



Picture 16: Drawing showing remote control method.

12.2 Test with Ethanol Dec 23



AS-COMSAT_OrbitCh
angeTestrig2023-12-2

13 Design of a LOX/LCH4 Thruster Engine for Satellite Transfer from LEO to GEO (Master Thesis 2024)

بسم الله الرحمن الرحيم

Project N° :
88/2497
89/2498
2023 / 2024

FINAL YEAR PROJECT

Submitted in fulfillment of the requirements for the

ENGINEERING DEGREE FROM THE LEBANESE UNIVERSITY
BRANCH 1

Major : Mechanical Engineering in Energy

Done by

Jana Hammoud & Roukaya Al Samad

Design of a LOX/LCH4 Thruster Engine with a Pressure Feed System for Satellite Transfer from LEO to GEO, with Regenerative Cooling and Heat Recovery using RPA

Supervisor:

Dr. Fadi Taychouri

Defended on December --, 2024 in front of the jury:

Dr. Fadi Taychouri	President
Dr. Mohammad Abdel-Wahab	Member
Eng. Loubna Khaled	Member

13.1 Presentation of Master Thesis on 8.12.24

Introduction

- Background
- Classifications Of Propulsive Devices
- Liquid Propellant Thrusters
- Types of Liquid-Propellant Thruster Engines
- Components of Liquid Propellant Thruster Engines
- Propellant Feed System
- Thrust Chambers: Combustion Chambers and Nozzles

3

Background

- Satellites play a crucial role in communication, navigation, and weather forecasting.

- Geostationary Earth Orbit (GEO):
  - ➢ 35,786 km above the equator.
  - ➢ ideal for uninterrupted coverage.

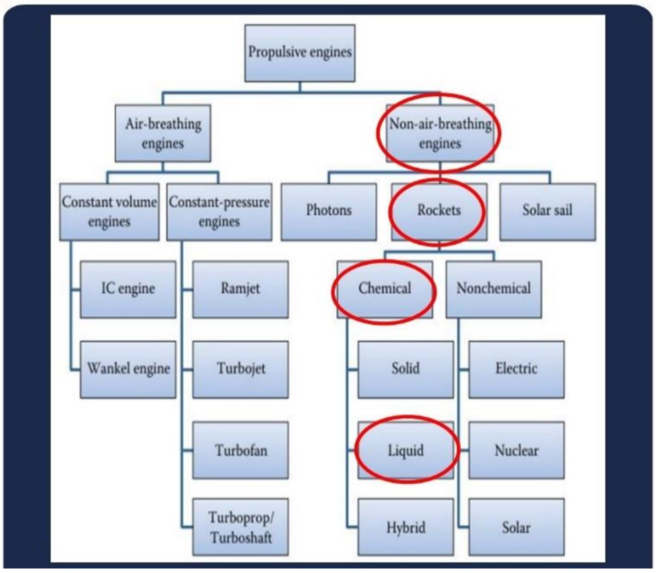
Challenges of direct GEO Transfer

- High energy demands .

✓ Solution: Transfer via Low Earth Orbit (LEO) (160–2,000 km above the Earth's surface), which is energy-efficient for deployment.

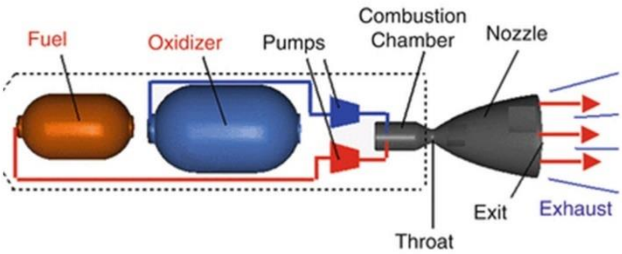
4

Classifications Of Propulsive Devices



5

Liquid Propellant Thrusters

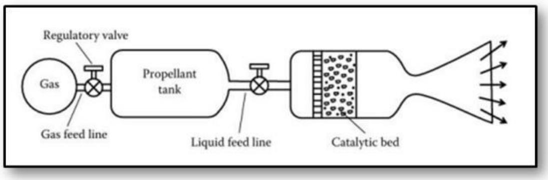


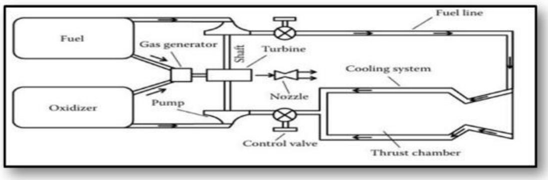
6

Types of Liquid-Propellant Thruster Engines

01 Monopropellant Engines

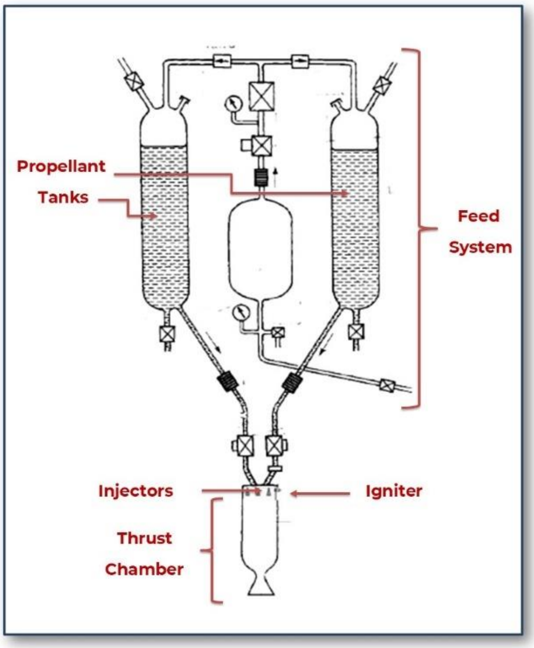
02 Bipropellant Engines





7

Components of Liquid Propellant Thruster Engines

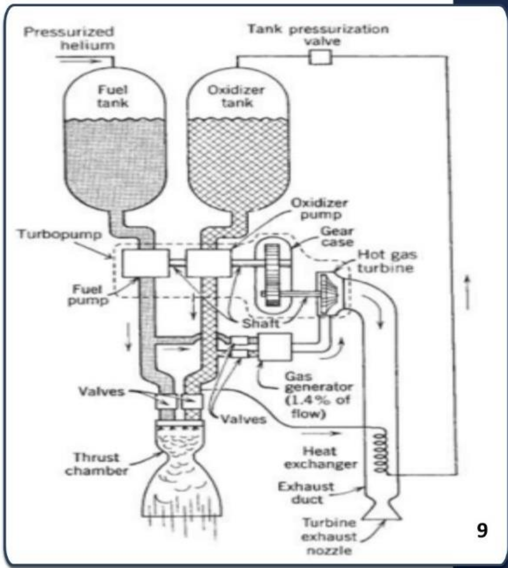


8

Propellant Feed System

01 Pressure-Fed Systems

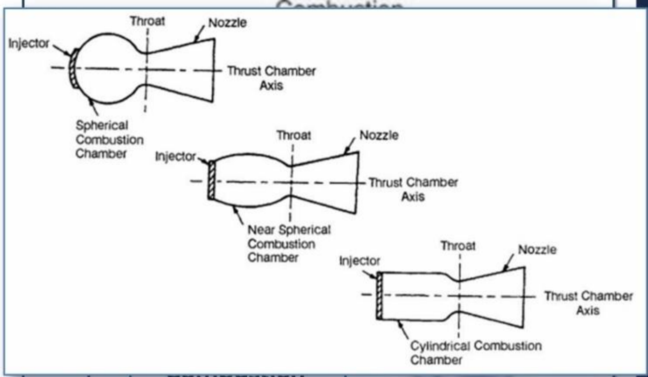
02 Pump-Fed Systems



9

Thrust Chambers: Combustion Chambers and Nozzles

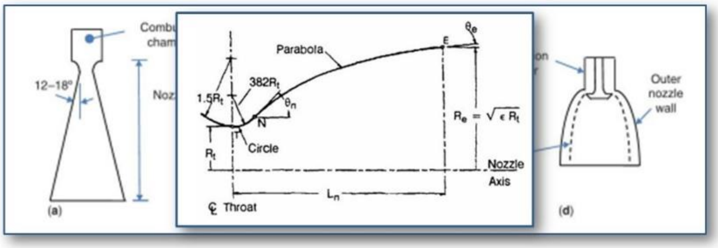
Combustion Chambers



10

Thrust Chambers: Combustion Chambers and Nozzles

- Combustion Chambers
- Nozzles

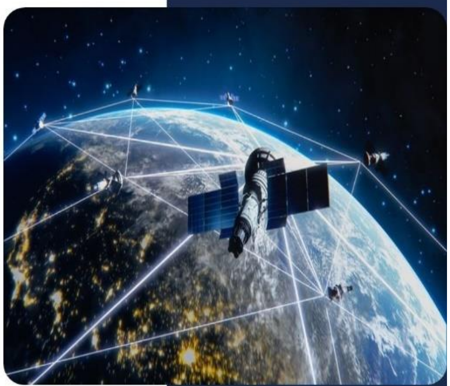


- Conical Nozzles
- Contoured (Bell) Nozzles
- Annular nozzles (plug)
- Annular nozzles (expansion–deflection)

11

Mission Requirements

- **Payload Mass Specifications : 50 kg**
- **Δv Requirement : 4 km/s.**
- **Thrust Force : 5 kN.**
- **Chamber Pressure: 300 psia (approximately 20.7 bar).**



12

Design Objectives:

- **High Performance**
  Achieve high specific impulse and efficient propellant use to minimize system mass
- **Reliability**
  Use a pressure-fed system to ensure consistent operation.
- **Thermal Management**
  Use regenerative cooling to manage the extreme thermal loads in the combustion chamber and nozzle.
- **Scalability and Feasibility**
  Ensure the design is practical for implementation and testing within the constraints of the mission and the resources available.

13

Performance Calculations and Design Methodology

- **01 . Stoichiometric and Optimal Mixture Ratios**
- **02. NASA CEA Simulations at Optimal mixture Ratio**
- **03. Interpretation of NASA CEA Results**
- **04. Mass Flow Calculations**
- **05. Geometry Optimization**
- **06. Design Validation and Testing Preparation**

14

Performance Calculations and Design Methodology

01 . Stoichiometric and Optimal Mixture Ratios

1.1 . Stoichiometric Mixture Ratio

The balanced chemical equation for the combustion of methane is:

$$CH_4 + 2O_2 \rightarrow CO_2 + 2H_2O$$

- Molar mass of CH4 (methane) : 12.01 g/mol+ 4.032 g/mol= 16.042 g/mol
- Molar mass of O2 (oxygen) : : 16.00 g/mol x 2 = 32.00 g/mol

To calculate the stoichiometric mass ratio, the following equation is used:

$$Stoichiometric\ Mass\ Ratio(O/F) = \frac{Mass\ of\ O_2}{Mass\ of\ CH_4} = \frac{2\times 32.00\ g/mol}{16.04\ g/mol} \approx 4$$

15

Performance Calculations and Design Methodology

01 . Stoichiometric and Optimal Mixture Ratios

1.2. Optimal Mixture Ratio

- **Inputs for NASA CEA Simulations:**
  - Fuel: Liquid Methane (LCH_4)
  - Oxidizer: Liquid Oxygen (LOX)
  - Chamber Pressure (Pc): 300 psia
  - Chamber-to-Exit Pressure Ratio (Pc/Pe): 30,000,000,000 to simulate vacuum conditions
  - Mixture Ratios: O/F mass ratios ranging from 2.0 to 4.6 in increments of 0.04
  - Area Ratio (Ae/At): Limited to a practical maximum of 100 for manufacturability

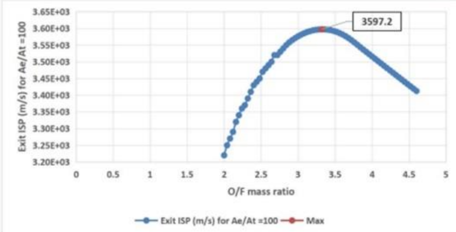


Figure 3.1 Specific Impulse (Isp) vs. O/F Mass Ratio at 300 psia Chamber Pressure in Vacuum Conditions (NASA CEA Results)

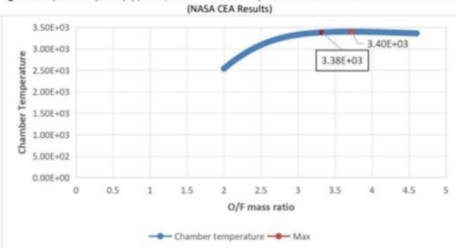


Figure 3.2 Chamber Temperature vs. O/F Mass Ratio at 300 psia Chamber Pressure in Vacuum Conditions (NASA CEA Results)

Performance Calculations and Design Methodology

02. NASA CEA Simulations at Optimal mixture Ratio

Station	Entropy (s) (J/(kg·K))	Gibbs Energy (G) (J/kg)	Internal Energy(u) (J/kg)	Molecular Wt. (M) (mw) (g/mol)	Temperature (t) (K)	Pc/Pe
Chamber	1.2733E+01	-4.4689E+04	-2.9314E+03	2.1121E+01	3.3840E+03	1.0000E+00
Throat	1.2733E+01	-4.3437E+04	-3.5577E+03	2.1414E+01	3.2304E+03	1.7241E+00
Exit	1.2733E+01	-1.3550E+04	-1.1171E+04	2.3120E+01	1.9164E+02	3.0000E+10
Exit at Ae/At = 100	1.2733E+01	-2.6756E+04	-8.5973E+03	2.3101E+01	1.4676E+03	1.2410E+03

c

17

Performance Calculations and Design Methodology

03. Interpretation of NASA CEA Results

- **Performance Parameters:**
  - The thrust coefficient (Cf)
  - Effective exhaust velocity (c) :

    c = Ve + Ae (Pe – Pa)
  - and specific impulse (Isp):

    Isp = $\frac{c}{g} = \frac{3597.2\ m/s}{9.81\ m/s^2} = 366.687\ s.$
  - Characteristic velocity (C*),

Station	Thrust Coefficient (Cf)	Effective exhaust velocity (c), m/s))	Characteristic velocity (C*, m/s)
Chamber	0.0000E+00	0.0000E+00	--
Throat	6.5121E-01	1.1867E+03	1822.3
Exit	2.3933E+00	4.3613E+03	1822.3
Exit at Ae/At = 100	1.9740E+00	3.5972E+03	1822.3

18

Performance Calculations and Design Methodology

03. Interpretation of NASA CEA Results

- **Thermodynamic Properties:**
  - Chamber temperature peaks at 3384 K
  - Density drops as gases expand

Station	Temperature (t) (K)	Density (ρ) (kg/m³)
Chamber	3.3840E+03	1.5527E+00
Throat	3.2304E+03	9.5647E-01
Exit	1.9164E+02	1.1255E-09
Exit at Ae/At = 100	1.4676E+03	3.1554E-03

19

Performance Calculations and Design Methodology

03. Interpretation of NASA CEA Results

- **Flow and Expansion Characteristics:**
  - The Mach number transitions from 1 at the throat to 4.5072 for Ae/At = 100

Station	Mach Number
Chamber	0.0000E+00
Throat	1.0000E+00
Exit	1.7621E+01
Exit at Ae/At = 100	4.5072E+00

a

20

Performance Calculations and Design Methodology

04. Mass Flow Calculations

$$\dot{m} = \frac{F}{c} = \frac{5000\ N}{3597.2\ m/s} = 1.39\ kg/s$$

- Total Mass Flow Rate: 1.39 kg/s.
- Fuel Mass Flow Rate: $\dot{m}_{CH_4} = \frac{\dot{m}}{1+MR} = 0.3217\ kg/s$
- Oxidizer Mass Flow Rate: $\dot{m}_{O_2} = \frac{\dot{m}}{1+1/MR} = 1.0682\ kg/s$

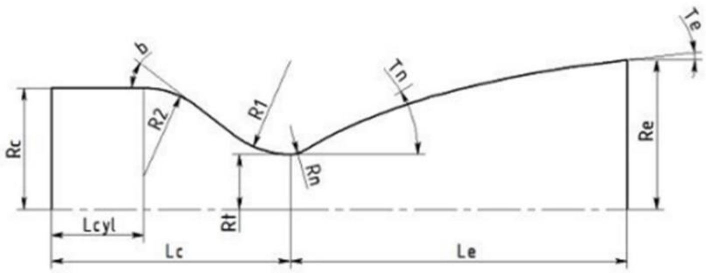
21

Performance Calculations and Design Methodology

05. Geometry Optimization

01 Nozzle Geometry

02 Combustion Chamber Geometry



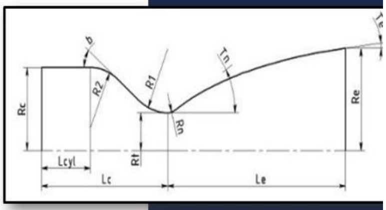
22

Performance Calculations and Design Methodology

05. Geometry Optimization

01 Nozzle Geometry

1.1. Throat Area and Diameter



- Using the equation for characteristic velocity (c*) :

$$c^* = \frac{P_c \times A_t}{\dot{m}}$$

- Rearranging for At = $\frac{\dot{m} \times c^*}{P_c} = 1.2246 \times 10^{-3} m^2$.

- The throat diameter (Dt) is then : $D_t = \sqrt{\frac{4 A_t}{\pi}} = 3.9486\ cm$

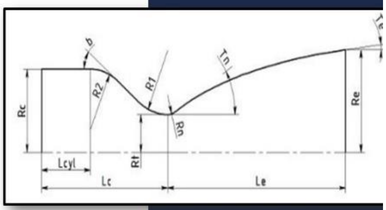
23

Performance Calculations and Design Methodology

05. Geometry Optimization

01 Nozzle Geometry

1.2. Exit Area and Diameter



- The exit area Ae can be calculated as follows:

Ae = 100 x At = 0.12246 m^2.

- The exit diameter De is then calculated as:.

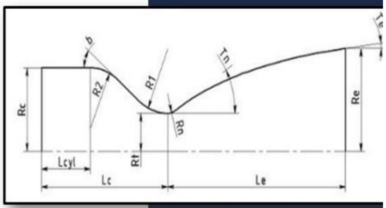
$$D_e = \sqrt{\frac{4 A_e}{\pi}} = 39.486\ cm$$

24

Performance Calculations and Design Methodology

05. Geometry Optimization

02 Combustion Chamber Geometry



2.1 . Combustion Chamber Area and Diameter

- Assuming a contraction ratio Ac/At = 4:

  Ac = 4 x At = 4.8983 x 10^{-3} m^2.

$$Dc = \sqrt{\frac{4\,Ac}{\pi}} = 7.8973\ cm.$$

2.2. Combustion Chamber Volume Vc = $L^* \times At$

- Assuming L^*= 0.8 m , so Vc = 9.7966 x 10^{-4} m^3

2.3. Combustion Chamber Length

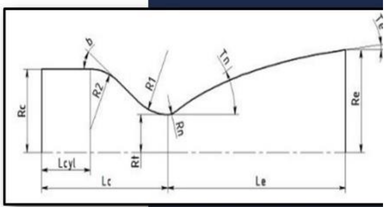
$$L_c = \frac{V_c}{A_c} = \frac{9.7966 \times 10^{-4}}{4.8983 \times 10^{-3}} = 0.2\ m = 20\ cm.$$

25

Performance Calculations and Design Methodology

05. Geometry Optimization

02 Combustion Chamber Geometry



2.4. Propellant Stay Time in the Combustion Chamber

$$t_s = \frac{V_c}{\dot{m} \times v}$$ where: $$v = \frac{RT}{P_c}$$

$R = \frac{R_u}{MW} = \frac{8314}{21.121}$= 393.64 J/kg.K

So : $$v = \frac{393..64\ \times 338402}{2.068 \times 10^6} = 0.644\ m^3/kg$$

Then we obtained : $$t_s = \frac{9.7966\ \times 10^{-4}}{1.39 \times 00.644} = 1.09\ ms.$$

26

Performance Calculations and Design Methodology

06. Design Validation and Testing Preparation

6.1 . RPA Results Interpretation

6.2. CAD Modeling

Rocket Propulsion Analysis

27

6.1 . RPA Results Interpretation

Chamber Performance

Thermodynamic properties | Performance | Altitude performance | Throttled performance

Theoretical (ideal) performance (O/F=3.320)

Parameter	Sea level	Sea level (flow sep.)	Optimum expansion	Vacuum	Unit
Characteristic velocity			1822.05		m/s
Effective exhaust velocity	-5289.98	337.45	3596.82	3743.73	m/s
Specific impulse (by mass)	-5289.98	337.45	3596.82	3743.73	N-s/kg
Specific impulse (by weight)	-539.43	34.41	366.77	381.75	s
Thrust coefficient	-2.9033	0.1852	1.9741	2.0547	

Estimated delivered performance (O/F=3.320)

Reaction efficiency: 0.9711

Nozzle efficiency: 0.9777

Overall efficiency: 0.9495

Parameter	Sea level	Sea level (flow sep.)	Optimum expansion	Vacuum	Unit
Characteristic velocity			1769.44		m/s
Effective exhaust velocity	-5479.12	168.74	3407.68	3554.59	m/s
Specific impulse (by mass)	-5479.12	168.74	3407.68	3554.59	N-s/kg
Specific impulse (by weight)	-558.71	17.21	347.49	362.47	s
Thrust coefficient	-3.0965	0.0954	1.9258	2.0089	

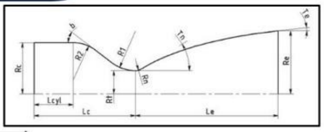
Ambient condition for optimum expansion: H=27.87 km, p=0.016 atm

Table 3-4 performance parameters by RPA tool

28

Performance Calculations and Design Methodology

06. Design Validation and Testing Preparation



6.1 . RPA Results Interpretation

Geometry of thrust chamber with parabolic nozzle			
Dc	78.77 mm	Le	474.39 mm
b	30 deg	Te	10.52 deg
R2	58.73 mm	De	393.87 mm
R1	29.54 mm	Ae/At	100
L*	800 mm	Le/Dt	12.04
Lc	221.62 mm	Le/c15 relative to length of cone nozzle	71.61 %
Lcyl	163.86 mm	Mass	9.06 kg
Dt	39.39 mm	Divergence efficiency	0.98417
Rn	7.52 mm	Drag efficiency	0.99157
Tn	38.84 deg	Thrust coefficient	2.00888 (vac)

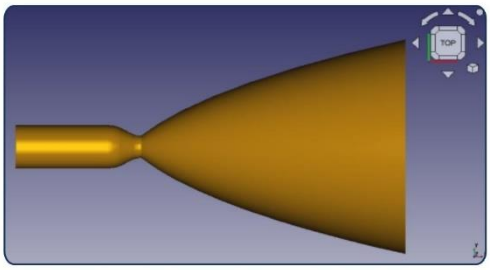
Table 3-2 Geometry of Thrust Chamber with Parabolic Nozzle by RPA tool

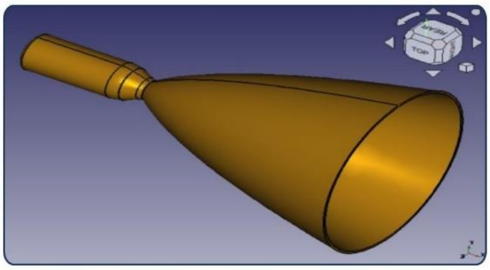
29

Performance Calculations and Design Methodology

06. Design Validation and Testing Preparation

6.2. CAD Modeling

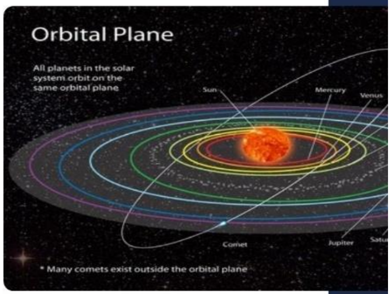




30

Fuel Requirements for Orbit Change

1. Key Mission and Design Parameters



- Δv = 4000 m/s.
- Payload mass = 50 kg,
- Dry mass of the spacecraft (including the structure, tanks, and engine) = 60 kg ,
- Total mass without propellant of 110 kg.
- The specific impulse of the engine = 366.69 seconds,
- The oxidizer-to-fuel mixture ratio =3.32,
- and the densities of the propellants are:
  - Liquid Oxygen (LOX): 1141 kg/m³.
  - Liquid Methane (LCH_4): 422.62 kg/m³.

31

Fuel Requirements for Orbit Change

2. Propellant Mass Calculations

- Rearranged Tsiolkovsky rocket equation : $m_i = m_f \times e^{\frac{\Delta v}{I_{sp} \times g_0}}$

$$m_i = 100 \times e^{\frac{4000}{36669 \times 9.81}} \approx 334.44\ kg$$

- The total propellant mass : $M_p = m_i - m_f = 334.44\ kg\ - 110\ kg = 224.44\ kg$

3. Oxidizer and Fuel Mass Proportions

$$Oxidizer\ Mass = M_{LO_2} = \frac{M_p \times MR}{1 + MR} = \frac{224.44 \times 3.32}{1 + 3.32} = 172.48\ kg$$

$$Fuel\ Mass = M_{LCH_4} = \frac{M_p}{1 + MR} = \frac{224.44}{1 + 3.32} = 51.95\ kg$$

32

Fuel Requirements for Orbit Change

4. Propellant Volume Calculations

- Liquid Oxygen (LOX) Volume : $V_{LO_2} = \frac{M_{LO_2}}{\rho_{LO_2}} = \frac{172.48\ kg}{1.141\ kg/l} \approx 151.17\ l.$
- Liquid Methane (LCH4) Volume : $V_{LCH_4} = \frac{M_{LCH_4}}{\rho_{LCH_4}} = \frac{51.95\ kg}{0.422\ kg/l} \approx 123.11\ l.$

5. Burn Time Calculations

- The burn time, or the duration the engine operates to achieve the required Δv.

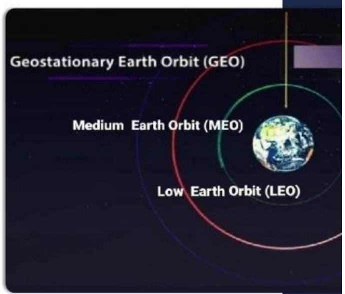
$$t_b = \frac{M_p}{\dot{m}} = \frac{224.44\ kg}{1.39\ kg/s} \approx 161.47\ s = 2\ minutes\ 41.47\ seconds.$$

33

Fuel Requirements for Orbit Change

6. Mission-Level and Engine Performance Insights

- **From a mission perspective:**
  - Burn time aligns with Δv requirements and ensures sufficient propellant for the transfer.
  - Provides adequate time for precise and controlled orbit transfer execution.
- **From an engine performance perspective:**
  - Validates the thruster's ability to maintain required thrust and mass flow rates
  - Ensures combustion stability with a 1.09 ms propellant stay time (shorter than burn time)
  - Chamber pressure (300 psia) and mass flow rates confirm efficient propellant utilization.



34

Thermal Analysis of LOX/LCH_4 Thrust Chamber with Regenerative Cooling

1 . The regenerative cooling system in our design serves two main purposes:

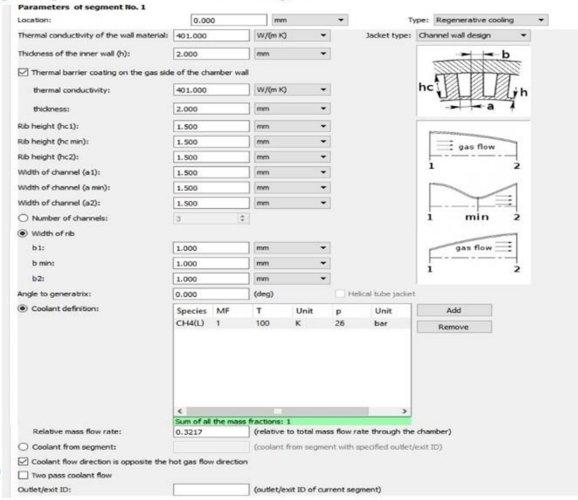
1. **Wall Protection:** It prevents the combustion chamber wall, made of copper, from melting by maintaining its temperature below the melting point of copper at 20.7 bar, which is 1357.94 K.

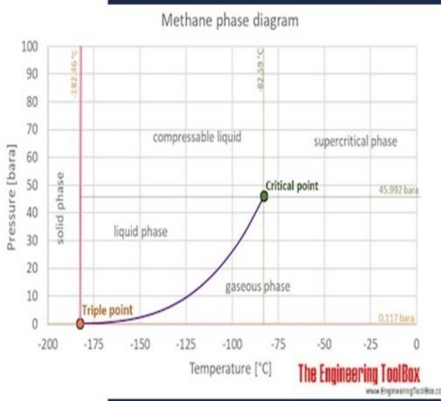
2. **Coolant Preheating:** It preheats the liquid methane before entering the combustion chamber, ensuring a full phase transition to gas, which enhances combustion efficiency.

35

Thermal Analysis of LOX/LCH_4 Thrust Chamber with Regenerative Cooling

2. ey Inputs for the Analysis

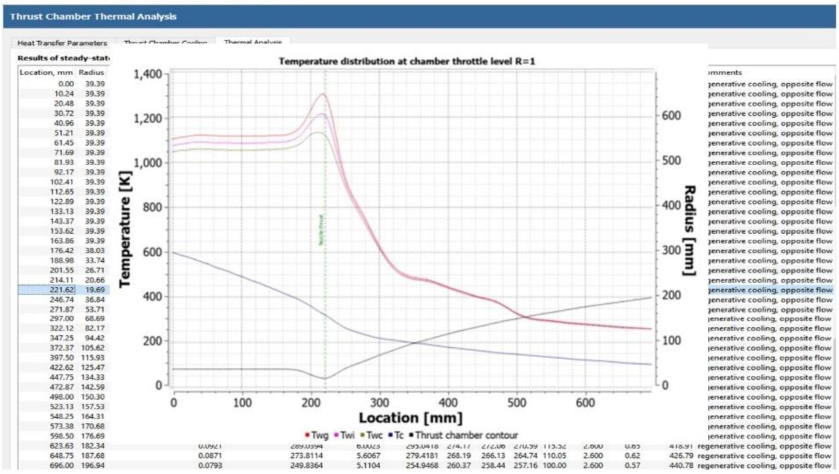




36

Thermal Analysis of LOX/LCH_4 Thrust Chamber with Regenerative Cooling

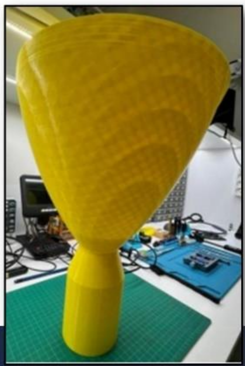
3. Results and Observations



37

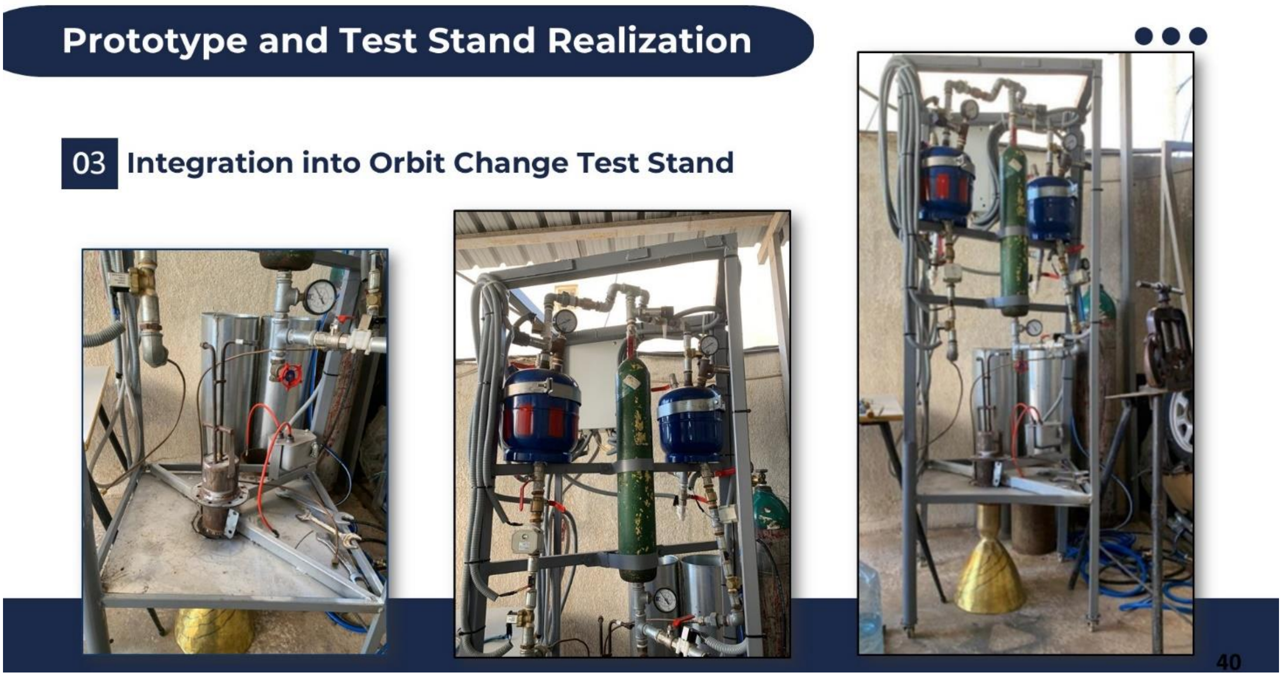
Prototype and Test Stand Realization

01 **Plastic Model of Combustion Chamber and Nozzle from 3D Printer**



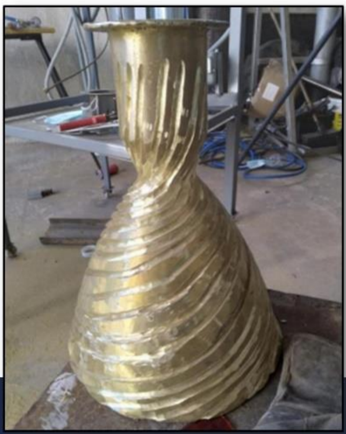
38


Prototype and Test Stand Realization
02 Combustion Chamber and Nozzle After Melting and Forming Copper
39


Prototype and Test Stand Realization
03 Integration into Orbit Change Test Stand
40

Prototype and Test Stand Realization

04 Realization of Pipes Layout

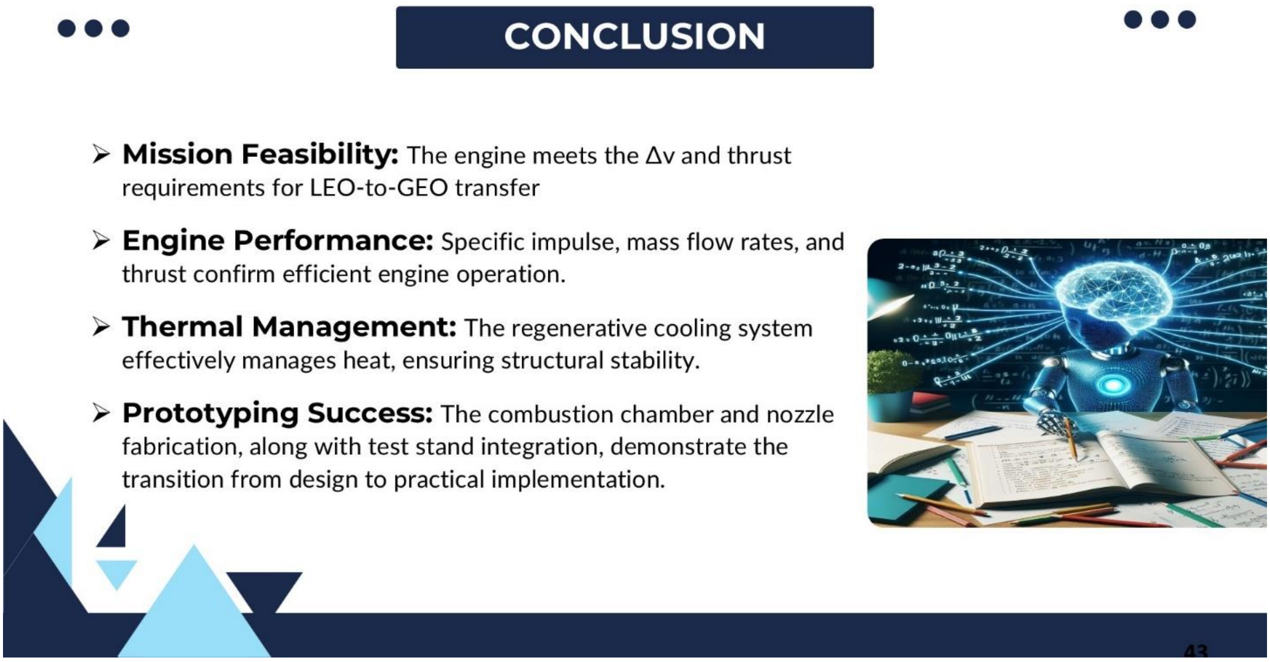



41

Prototype and Test Stand Realization

05 Integration of Closeout Wall


42



13.2 Abstract

This thesis explores the design and analysis of a LOX/LCH_4 thruster engine with a pressure feed system, tailored to efficiently transfer a spacecraft from Low Earth Orbit (LEO) to Geostationary Orbit (GEO). Blending innovation with practical feasibility, the study delves into advanced computational tools to optimize engine performance and sustainability.

The first phase involves calculating key performance parameters using NASA CEA and RPA software to determine the optimal oxidizer-to-fuel mixture ratio and evaluate specific impulse, effective exhaust velocity, and combustion characteristics. These findings guide the design of critical engine components, including the thrust chamber, combustion chamber, and nozzle, ensuring optimal operation under vacuum conditions for a target thrust force of 5 kN.

The second phase focuses on propellant calculations for orbital transfer, employing the Tsiolkovsky rocket equation to estimate fuel requirements based on spacecraft mass and mission Δv. Thermal management strategies are explored through RPA simulations to develop effective cooling and heat recovery systems, enhancing engine reliability and efficiency.

This research bridges the gap between theoretical propulsion models and practical design, contributing to the development of efficient and reliable liquid propulsion systems for small satellite missions. By integrating advanced simulation tools and focusing on performance optimization, the study paves the way for future advancements in sustainable and cost-effective space exploration.

Keywords: LOX/LCH_4 thruster engine, pressure feed system, NASA CEA, RPA, orbital transfer, chemical propulsion, heat recovery, space exploration.

13.3 Résumé

Cette thèse explore la conception et l'analyse d'un moteur-fusée LOX/LCH_4 avec un système d'alimentation par pression, conçu pour transférer efficacement un engin spatial de l'orbite terrestre basse (LEO) à l'orbite géostationnaire (GEO). Alliant innovation et faisabilité pratique, l'étude s'appuie sur des outils informatiques avancés pour optimiser les performances du moteur et sa durabilité.

La première phase consiste à calculer les principaux paramètres de performance à l'aide des logiciels NASA CEA et RPA afin de déterminer le rapport optimal oxydant/carburant et d'évaluer l'impulsion spécifique, la vitesse d'éjection effective et les caractéristiques de combustion. Ces résultats orientent la conception des composants clés du moteur, notamment la chambre de poussée, la chambre de combustion et la tuyère, garantissant un fonctionnement optimal dans des conditions de vide pour une poussée cible de 5 kN.

La deuxième phase se concentre sur les calculs de propergol pour le transfert orbital, en utilisant l'équation de Tsiolkovski pour estimer les besoins en carburant en fonction de la masse de l'engin spatial et du Δv de la mission. Les stratégies de gestion thermique sont étudiées à l'aide de simulations avec RPA pour développer des systèmes de refroidissement et de récupération de chaleur efficaces, améliorant ainsi la fiabilité et l'efficacité du moteur.

Cette recherche comble le fossé entre les modèles théoriques de propulsion et la conception pratique, en contribuant au développement de systèmes de propulsion liquide efficaces et fiables pour les missions de petits satellites. En intégrant des outils de simulation avancés

et en mettant l'accent sur l'optimisation des performances, cette étude ouvre la voie à de futures avancées dans l'exploration spatiale durable et rentable.

Mots-clés: moteur-fusée LOX/LCH_4, système d'alimentation par pression, NASA CEA, RPA, transfert orbital, propulsion chimique, récupération de chaleur, exploration spatiale.

13.4 Abbreviations

LOX	Liquid Oxygen
LCH_4	Liquid Methane
LEO	Low Earth Orbit
GEO	Geostationary Orbit
CEA	Chemical Equilibrium with Applications
NASA	National Aeronautics and Space Administration
RPA	Rocket Propulsion Analysis
CAE	Computer-Aided Engineering (tools)

13.5 List of Parameters

MR	Oxidizer-to-Fuel Ratio
P_C	Combustion Chamber Pressure
p_e	Exit Pressure
p_a	Ambient Pressure
Isp	Specific Impulse
C_f	Thrust Coefficient
γ	specific heat ratio of gas
a	Sonic Velocity
h	Enthalpy
s	Entropy
G	Gibbs Energy
u	Internal Energy
C_p	Specific Heat at Constant Pressure
T	temperature
c	Effective Exhaust Velocity
V_e	Exit velocity

g	gravitational acceleration
$\dot{m}$	Total Propellant Mass Flow Rate
$\dot{m}_{CH4}$	Fuel Mass Flow Rate
$\dot{m}_{O2}$	Oxidizer Mass Flow Rate
F	Thrust Force
C^*	Characteristic velocity
A_t	Nozzle Throat Area
D_t	Nozzle Throat Diameter
A_e	Nozzle Exit Area
D_e	Nozzle Exit Diameter
A_C	Combustion Chamber Area
D_C	Combustion Chamber Diameter
V_C	Combustion Chamber Volume
L_C	Combustion Chamber Length
L^*	Characteristic length of the Combustion Chamber
ε	Nozzle Expansion Ratio
t_S	Propellant Stay Time
v	specific volume of propellant
R	specific gas constant
R_U	universal gas constant
MW	Molecular Weight
Δv	velocity change
ρ_{LO2}	Liquid Oxygen Density
ρ_{LCH4}	Liquid Methane Density
m_i	Initial mass (spacecraft mass including fuel)
m_f	Final mass (spacecraft mass without fuel)
M_{LO2}	Oxidizer Mass
M_{LCH4}	Fuel Mass
V_{LO2}	Liquid Oxygen Volume
V_{LCH4}	Liquid Methane Volume
t_b	Burn Time
hc1, hc min, hc2	Rib Height
a1, a min, a2	Channel Width
b1, b min, b2	Rib Width
T_{wg}	Temperature of chamber wall on its hot gas side
T_{wi}	Temperature between the thermal barrier coating layer and chamber wall

T_{wc}	Temperature of chamber wall on its cooler side
T_c	Temperature of the coolant
p_c	Pressure of the coolant
w_c	Velocity of the coolant
ρ	Density of the coolant
q_{conv}	Convective Heat Flux
q_{rad}	Radiative Heat Flux
q_{total}	Total Heat Flux
h	Heat Transfer Coefficient
T_{gas}	combustion gas temperature
T_{wall}	inner wall temperature
ϵ	Emissivity
σ	Stefan-Boltzmann constant
T_m	Melting temperature at standard pressure
Δv	Change in specific volume during melting
ΔH_f	Latent heat of fusion
T_m'	Melting temperature at combustion chamber pressure

13.6 List of Tables and Figures

13.7 Introduction

13.7.1 Overview of the Thruster Design Process

The goal of this thesis is to design a LOX/LCH4 thruster engine with a pressure feed system for transferring spacecraft from Low Earth Orbit (LEO) to Geostationary Earth Orbit (GEO), with an integrated heat recovery system. The process of designing this thruster engine requires a comprehensive approach that takes into account performance, efficiency, and reliability, while adhering to mission-specific constraints such as weight, size, and thermal management. This chapter introduces the methodology followed in this project, covering the mission requirements, design constraints, and key performance metrics used in the development of the engine.

The design approach is broken down into several key stages, from identifying mission objectives to finalizing the components and conducting structural and thermal analyses using computational tools like RPA.

13.7.2 Methodology for Thruster Development

The development of the LOX/LCH_4 thruster engine follows a structured methodology tailored to meet the mission requirements of transferring a spacecraft from Low Earth Orbit (LEO) to Geostationary Earth Orbit (GEO). The design process balances performance optimization with constraints on weight, size, and operational efficiency.

13.7.2.1 Design Requirements and Constraints

The development of the LOX/LCH4 thruster engine is driven by specific mission requirements, particularly the need to provide efficient thrust while keeping the system compact and reliable. This section highlights the major design requirements and the operational constraints that shape the thruster engine's development.

Mission Requirements

The mission objective is to transfer a spacecraft from Low Earth Orbit (LEO) to Geostationary Earth Orbit (GEO) using a LOX/LCH_4 propulsion system. To achieve this, the design must fulfill the following key requirements:

1. Payload and Dry Mass Specifications:

- Payload mass: 50 kg
- Dry mass of the spacecraft: 60 kg

2. Δv Requirement:

- The orbit transfer maneuver requires achieving the necessary Δv, accounting for gravitational and orbital mechanics considerations.

3. Thrust Force:

- The propulsion system should generate sufficient thrust to efficiently perform the orbit transfer. An estimated thrust of 5 kN is considered as a preliminary target.

4. Propellant Selection and Mixture Ratio:

- Propellants: Liquid Oxygen (LOX) and Liquid Methane (LCH_4) are chosen for their performance characteristics, storability, and environmental compatibility.
- The oxidizer-to-fuel ratio will be optimized in later calculations to achieve maximum engine performance.

5. Structural and Thermal Design:

- The engine must be capable of withstanding high-pressure combustion and extreme thermal conditions, necessitating robust materials and an efficient cooling system.

6. System Integration:

- The propulsion system design must ensure seamless integration with the spacecraft, adhering to constraints on mass, volume, and structural compatibility.

These mission requirements establish the foundation for the propulsion system's design and performance objectives, guiding subsequent calculations and optimization efforts.

Performance Metrics

Key performance metrics are defined to evaluate the effectiveness of the thruster engine:

- Specific Impulse (Isp): This is a key measure of fuel efficiency, which the engine design aims to maximize while maintaining consistent combustion.
- Thrust-to-Weight Ratio: A high thrust-to-weight ratio is targeted to provide optimal performance without adding unnecessary mass to the spacecraft.

Physical and Operational Constraints

Key constraints influencing the design include:

- Weight and Size: Minimizing engine mass and volume to fit within spacecraft constraints.
- Pressure and Temperature: The engine operates at high chamber pressures, estimated at 300 psia (approximately 20.7 bar), and temperatures exceeding 3000 K, requiring robust materials and precision engineering.
- Thermal Management: Incorporating reliable cooling systems to handle extreme heat and improve efficiency through heat recovery.

Design Challenges

The engine design faces several challenges:

- Thermal Stress: High temperatures in the combustion chamber and nozzle require advanced cooling techniques and heat-resistant materials.
- Pressure Stability: Ensuring stable chamber pressure without pumps poses a challenge to achieving consistent thrust.
- Efficient Combustion: The thruster must ensure complete and efficient combustion of LOX and LCH4 to maximize performance while minimizing fuel consumption.

13.7.2.2 Combustion Chamber Design

The combustion chamber is a crucial component responsible for igniting and sustaining the combustion process. Its design must ensure efficient combustion while managing extreme temperature and pressure conditions. The following aspects are considered in the design:

- Chamber Geometry: The combustion chamber's geometry, is optimized for fuel mixing and complete combustion.
- Material Selection: Materials like Inconel or copper alloys are selected for their strength under high thermal and mechanical stress.
- Cooling System: Regenerative cooling ensures thermal stability.

13.7.2.3 Nozzle Design

The nozzle transforms thermal energy into kinetic energy to produce thrust:

- Geometry: Optimized throat and exit diameters maximize exhaust velocity.
- Materials: High-temperature-resistant materials ensure durability.
- Cooling Requirements: Regenerative cooling is integrated for thermal management.

13.7.2.4 Pressure Feed System

A pressure feed system is chosen for its simplicity and reliability compared to pump-fed systems. Key considerations include:

- Feed System Overview: The pressure feed system eliminates the need for complex pumps, reducing system mass and complexity.
- Tank Design: Propellant tanks are designed to withstand high pressures and provide steady flow into the combustion chamber.
- Pressurization Gas: Helium is used for pressurizing the tanks due to its inert nature and light weight.

13.7.2.5 Cooling and Heat Recovery System

Effective cooling and heat recovery are vital to ensure long engine life and improve efficiency:

- Cooling Techniques: Regenerative cooling is implemented, where the fuel is used to cool the combustion chamber walls before injection.
- Heat Recovery: The system recovers heat from the combustion process to preheat the fuel, improving the overall efficiency of the engine.

13.7.2.6 Structural and Thermal Analysis

To ensure the engine's reliability under operational conditions, structural and thermal analyses are essential:

- RPA: Computational analysis tools like RPA are used to simulate thermal stresses, fluid dynamics, and heat transfer in the thruster engine.

13.8 Fundamentals of Thruster Propulsion

13.8.1 LEO to GEO Transfer

13.8.1.1 Low Earth Orbit (LEO)

Low Earth Orbit (LEO) refers to an orbit around the Earth with an altitude typically ranging from 160 km to 2,000 km. This region is commonly used for various satellite missions, including communications, Earth observation, and human spaceflight, due to its proximity to the Earth’s surface, which allows for lower launch costs and shorter orbital periods.

For the spacecraft in this project, LEO represents the point of departure, from which the thruster engine will provide the necessary delta-v to execute the transfer to GEO, fulfilling the mission's orbital transfer objectives.

13.8.1.2 Geostationary Earth Orbit (GEO)

The geostationary orbit (GEO) is a circular orbit located approximately 35,786 kilometers (22,236 miles) above the Earth's equator, where a satellite's orbital period matches the Earth's 24-hour rotation. This synchronization results in the satellite appearing stationary to an observer on the ground, continuously covering the same geographical area. This stable positioning is invaluable for communication satellites, weather monitoring, and other applications that require uninterrupted coverage of specific regions.

***Figure 13.1** illustration of Orbital Altitudes: Low Earth Orbit (LEO), Medium Earth Orbit (MEO), and Geostationary Earth Orbit (GEO)*

Importance of the Geostationary Orbit

- **Continuous Coverage:** GEO satellites provide uninterrupted communication and monitoring over large regions, which is crucial for services like television broadcasting, telecommunications, and emergency responses.
- **Simplified Ground Equipment:** Since satellites in GEO remain fixed relative to Earth's surface, ground-based antennas can stay pointed in a single direction, eliminating the need for complex tracking systems.
- **Improved Signal Quality:** The stable position of GEO satellites ensures stronger and more reliable signal transmission, making them ideal for high-demand applications like satellite internet and real-time communication.

13.8.1.3 Challenges of Direct GEO Transfers

Placing a satellite directly into GEO is challenging due to the high altitude and the energy required. Direct transport to GEO is rarely practiced for the following reasons:

- **High Energy Requirements:** The altitude of GEO demands significant energy to overcome Earth's gravity and achieve the necessary orbital speed.
- **Launch Vehicle Limitations:** Most thrusters are designed to place payloads in low Earth orbit (LEO) or medium Earth orbit (MEO), making direct launches to GEO complex and expensive.
- **Gravity Losses:** A direct ascent to GEO would result in greater gravity losses, making the mission less efficient.

13.8.1.4 LEO to GEO Transfer Process

Given these challenges, satellites are typically launched into low Earth orbit (LEO) first. Satellites in LEO travel at high speeds, orbiting the Earth approximately every 90 to 120 minutes. This proximity allows for high-resolution imaging and low-latency communication but requires multiple satellites to cover a larger area due to their limited field of view.

Here's how the transfer process works:

- **Launch to LEO:** The satellite is initially placed in LEO, an altitude of around 160 to 2,000 kilometers, where launching is easier and requires less energy.
- **Transfer to GTO:** From LEO, the satellite is moved into a geostationary transfer orbit (GTO), an elliptical orbit with an apogee at GEO altitude. This is done using the thruster's upper stage or a separate propulsion system.
- **Final Maneuver to GEO:** At the apogee of GTO, the satellite’s propulsion system is fired to circularize the orbit, completing the transition to GEO.

13.8.2 Classifications Of Propulsive Devices

In the past century, a range of propulsive devices were developed for modern aircraft and spacecraft, classified into air-breathing and non-air-breathing engines (see Figure 2.2). Air-breathing engines use ambient air as an oxidizer and include gas turbines (turbojet, turbofan, turboprop), ramjets, and specialized scramjets. Non-air-breathing engines, or thruster engines, carry their oxidizer onboard and are categorized into chemical (solid, liquid, hybrid) and non-chemical (solar, electric, nuclear) types. This report focuses on chemical propulsion, with an emphasis on liquid propellant thruster engines.

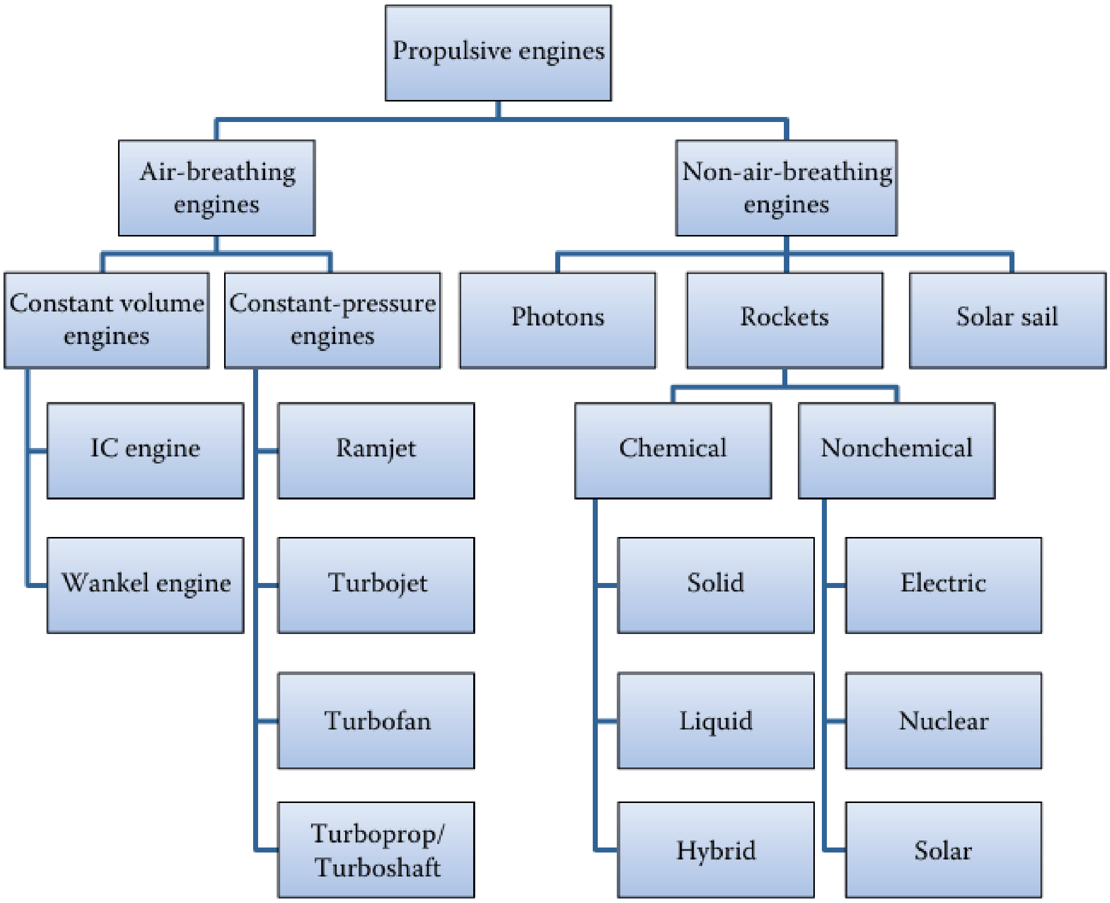


***Figure 13.2** Classification of propulsive*

13.8.3 Liquid Propellant Thrusters

Liquid propellant thruster engines have been extensively researched since the late 1930s and remain the most widely used propulsion system for space launch systems. They provide higher thrust and specific impulse due to the high chemical energy stored in liquid propellants. These engines use a liquid oxidizer and liquid fuel, which are fed into a combustion chamber either through pressurized tanks or pumps. The chemical reaction between the propellants generates hot gases that are ejected through a nozzle at high velocity, producing the necessary thrust to propel the vehicle. As the technology has matured, current developments focus on achieving design flexibility, simplicity, and reliability, all while maintaining high performance.

13.8.3.1 Components Of Liquid Propellant Thruster Engines

A typical liquid propellant thruster engine consists of several key components: the thrust chamber (which includes the combustion chamber and nozzle), injector, igniter, propellant tanks, propellant feed system, and cooling system (see Figure 2.3). The combustion chamber houses injectors that atomize the liquid propellants, mix them, and ignite them, resulting in the production of high-temperature, high-pressure gases. These gases are then expanded through a convergent-divergent nozzle to produce thrust.

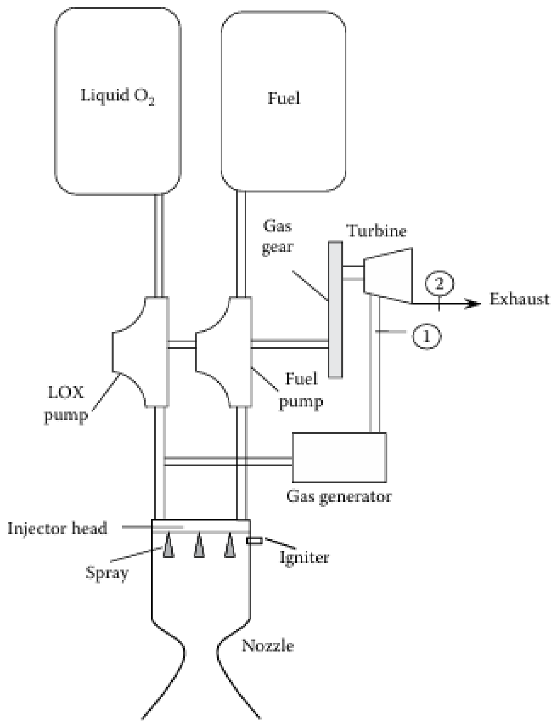


Figure 13.3 A typical liquid-propellant Thruster engine.

The propellant feed system, which may use high-pressure tanks or turbo-pumps, ensures the proper delivery of fuel and oxidizer to the combustion chamber. Efficient combustion requires the propellants to be vaporized and mixed thoroughly, which is achieved by feeding them at high pressure into the injectors. The ignition system provides the initial energy to start combustion, although hypergolic propellants ignite on contact and do not need an igniter. Finally, a cooling system is employed to manage the high temperatures in the combustion chamber and nozzle.

13.8.3.2 Types of Liquid-Propellant Thruster Engines

Since the development of liquid-propellant thruster engines in 1926, several variations have emerged. These engines can be broadly classified into two main types based on the number of liquid propellants used:(1) monopropellant thruster engine and (2) bipropellant thruster engine.

Monopropellant Thruster Engines

In a monopropellant thruster engine, a single liquid propellant is used, which decomposes with the help of a catalyst to produce hot gases. These gases are expanded through a nozzle to generate thrust. The key advantage of this system is its simplicity, as it eliminates the need for an oxidizer, making the overall engine design more straightforward. However, monopropellant engines are generally limited to applications requiring low thrust and short-duration burns.

A schematic of a typical monopropellant thruster engine is shown in Figure 2.4 , in which liquid propellant is injected into a catalyst bed and decomposes into high-pressure and high-temperature gas. This gas is then expanded through a convergent-divergent nozzle to create the required thrust. Monopropellants are usually chemicals that decompose easily in an exothermic reaction, releasing hot gas.

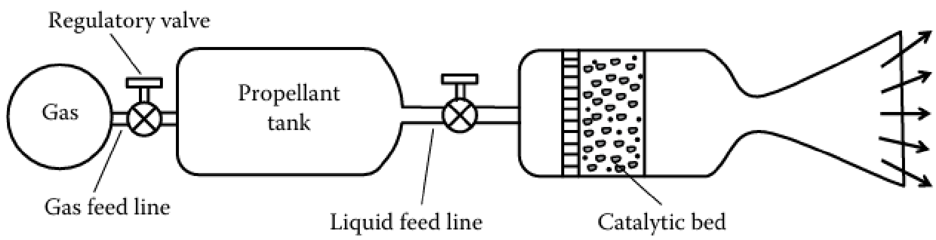


Figure 13.4 *Schematic of monopropellant LPR engine.*

Bipropellant Thruster Engines

Bipropellant thruster engines utilize two separate liquid propellants: one as fuel and the other as an oxidizer. This configuration, as illustrated in Figure 2.5, offers significant advantages over monopropellant engines, including higher specific impulse (performance), the ability to restart, variable thrust, and broader operational versatility. These engines are widely used in applications such as launch vehicles and missiles due to their superior performance.

A typical bipropellant thruster engine consists of has thrust chamber, injection system, cooling system, propellant feed sys tem, nozzle, and so on. The liquid fuel and oxidizer are atomized into fine sprays, mixed, vaporized, and ignited to produce high-temperature, high-pressure gases. These gases are then expanded through a convergent–divergent nozzle to generate thrust.

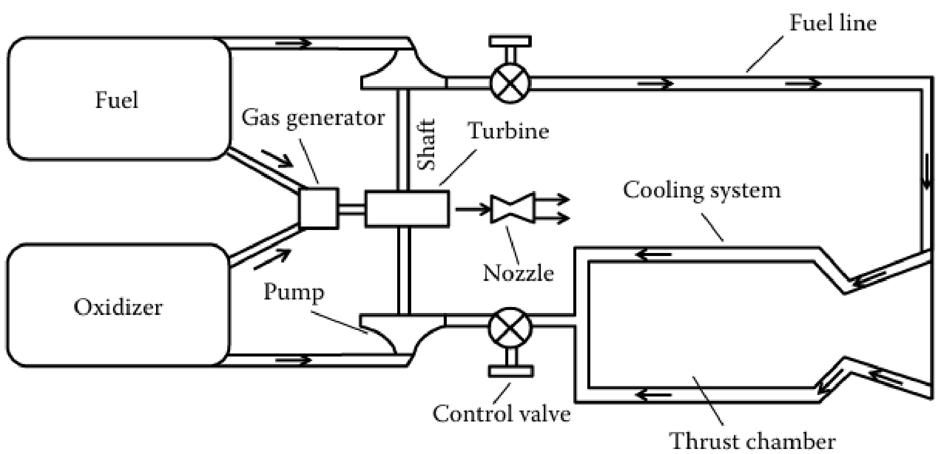


Figure 13.5 *Schematic of bipropellant LPR engine.*

Based on ignition methods, bipropellant engines are divided into two categories:

➢ **Hypergolic Engines**

In hypergolic engines, the fuel and oxidizer ignite spontaneously upon contact without the need for external ignition. Common hypergolic propellants include:

Fuels: Aniline, Triethylamine, Hydrazine, MMDH (Monomethylhydrazine), UDMH (Unsymmetrical Dimethylhydrazine)

Oxidizers: White fuming nitric acid (with nitrogen tetroxide), Red fuming nitric acid (with higher nitrogen tetroxide content)

These propellants can typically be stored at normal pressure and temperature, but some combinations, like liquid fluorine and liquid hydrogen, require cryogenic storage.

➢ **Non-Hypergolic Engines**

Non-hypergolic engines require external ignition energy to initiate combustion. These engines use fuels such as kerosene, hydrocarbons, alcohol, methane, and liquid hydrogen. Liquid methane (LCH4) is increasingly being favored in modern designs, especially when paired with liquid oxygen (LOX), due to its higher specific impulse, cleaner combustion, and cryogenic properties, which are easier to manage compared to liquid hydrogen.

13.8.3.3 Propellant Feed System

The propellant feed system plays a critical role in liquid thruster engines by delivering propellants from storage tanks to the combustion chamber at the correct flow rate and pressure. The feed system has two principal functions:

1. To increase the pressure of the propellants.

2. To supply them at design mass flow rates to the thrust chamber(s).

The energy for these functions is provided either by high-pressure gas, centrifugal pumps, or a combination of both. The choice of a specific feed system is governed by the thruster application, mission duration, number and type of thrust chambers, past experience, and general design requirements such as simplicity, ease of manufacture, cost-efficiency, and minimum inert mass.

Feed systems consist of components such as piping, valves, provisions for filling, draining, filters, and control devices to manage propellant flow. Depending on how the propellants are pressurized and fed into the thrust chamber, the feed system is classified as pressure-fed or pump-fed as shown in Figure 2.6.

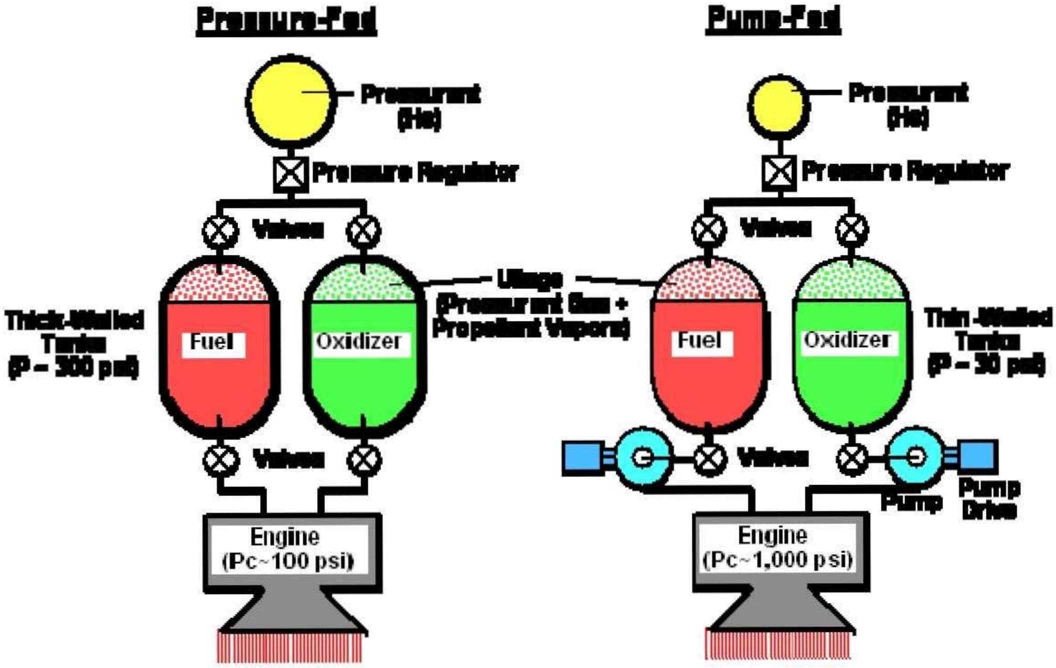


Figure 13.6 *Pressure-Fed and Pump-Fed Liquid Propulsion*

Pressure Feed System

The pressure feed system uses high-pressure gas to force the propellants from the tanks into the combustion chamber. This type of system is suitable for low-thrust applications with low chamber pressures and relatively low total impulse, such as attitude control systems or small upper stages.

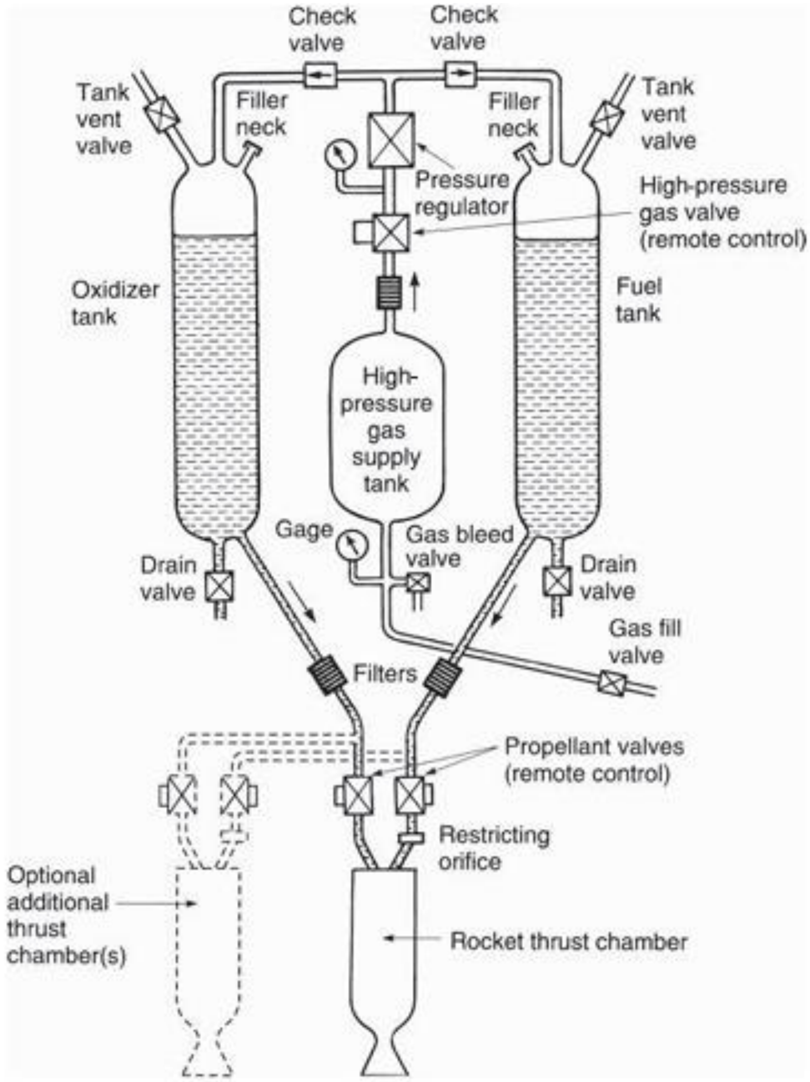


Figure 13.7 *Schematic Flow Diagram of a Liquid Propellant Thruster Engine with a Gas Pressure Feed System.*

The system typically includes the following components: (see Figure 2.7)

- Pressurized gas tank: Stores the high-pressure gas.
- Pressurant gas or other expulsion devices (e.g., helium gas) to provide the energy for the feed system
- Valves to control the pressure and flow and filters.
- Ducting or piping to transfer fluids to the combustion chamber.
- Thrust chamber: Converts the propellants' energy into thrust.
- Gas pressure regulator: Manages the pressurant gas flow to maintain consistent pressure.
- Propellant tanks: Hold the fuel and oxidizer.
- Propellant valves and feed lines: Direct propellant flow into the thrust chamber.

These systems are generally classified according to the source of the pressurant gas, which determines how the propellants are expelled from the tanks. There are two common configurations:

- **Monopropellant systems**, where a single propellant, such as hydrazine, flows through a catalyst bed and expands in a nozzle to generate thrust.
- **Bipropellant systems**, which use separate oxidizer and fuel tanks, both requiring a pressurization system to expel the propellants into the feed lines and ultimately to the thrust chamber.

One of the advantages of pressure-fed systems is their simplicity and reliability. These systems are often used in applications with low to moderate engine performance requirements, such as orbital maneuvering, reaction control, and small upper-stage propulsion. However, they tend to be heavier due to the need for thick-walled pressurized tanks.

Stored-gas pressurant systems are widely used, where gases such as helium (due to its low molecular weight) are stored at high pressures (up to 270 atm) and then supplied to the propellant tanks at regulated pressures. Factors that influence the selection of pressurization gases include mission requirements, cost, weight, reliability, and compatibility with tank materials.

The pressure-fed system is favored when system simplicity and low cost are prioritized over performance. Nonetheless, these systems are typically heavier because of the need for thick-walled, pressurized propellant tanks.

Turbopump Feed System

The turbo-pump feed system is favored for high-thrust, long-duration thruster engine systems with high specific impulse, as the gas pressure feed system is unsuitable for such applications. Turbo-pump systems are typically used in boosters, sustainers of space vehicles, long-range missile systems, and aircraft performance augmentation due to their advantages over gas pressure feed systems, such as:

- Flexibility: Easier control of pump speed allows better operational flexibility.
- High Pressure: The system can achieve stable pressures as high as 6–8 MPa.
- Compact Design: Smaller volume requirements, even for higher-thrust engines.
- High Power-to-Weight Ratio: Ranges from 15 to 50 kW/kg, making it more efficient.

A simplified turbo-pump system consists of several key components: (see Figure 2.8)

- Propellant tanks
- Inlet and discharge ducts
- Pumps: Pressurize the propellants.

- Turbine: Powers the pumps.
- Speed reduction gearbox: Transfers torque from the turbine to the pump at a reduced rotational speed.
- Gas generator: Produces hot gases by burning a small portion (1%-5%) of the total propellant flow, which powers the turbine.
- Heat exchanger and nozzle: Expands the hot gases generated by the gas generator.

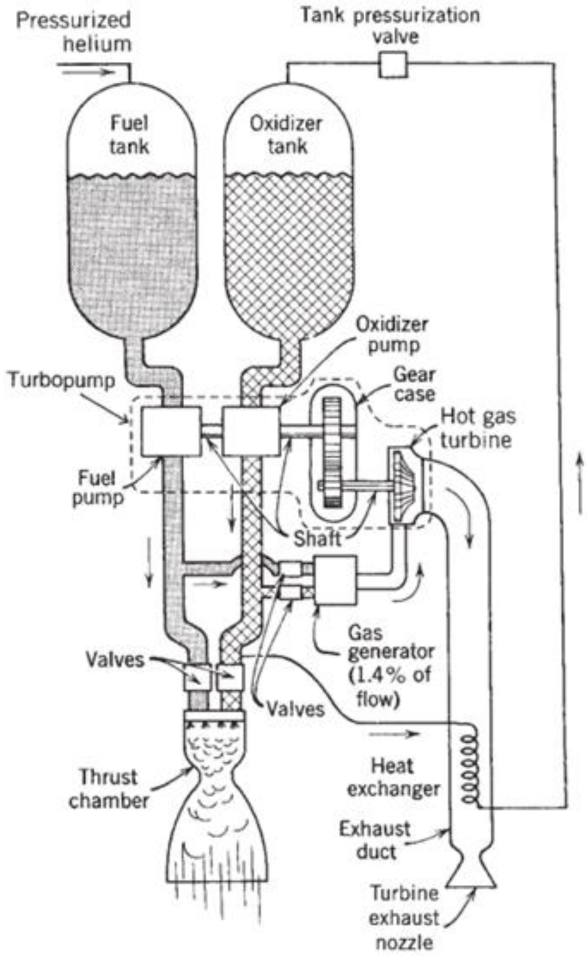


Figure 13.8 Simplified Schematic Diagram of a Liquid Propellant Thruster Engine with a Turbopump Feed System and Separate Gas Generator.

In this system, propellants are pressurized by pumps driven by turbines. The turbines are powered by hot gases, which are created in the gas generator by combusting a small fraction of the propellant. This setup allows the turbo-pump system to deliver high performance, making it ideal for demanding missions where weight and engine efficiency are critical.

Turbo-pump feed systems are classified based on the configuration of the turbine-pump drive and the exhaust gas discharge modes. The main types include:

1. Monopropellant Cycle
2. Bipropellant Cycle
3. Expander Cycle

4. Staged Combustion Cycle

13.8.4 Liquid Propellants

Propellants, the working substances of thruster engines, constitute the fluid that undergoes chemical and thermodynamic changes. The term "liquid propellant" encompasses all the various propellants stored as liquids.

Liquid propellants consist of a liquid fuel, a liquid oxidizer, and sometimes liquid additives. Examples of liquid fuels include hydrocarbons, liquid hydrogen, and alcohols, while common oxidizers are liquid oxygen, nitric acid, and liquid fluorine.

Liquid propellants can be classified based on several factors such as fuel-oxidizer arrangement, energy content, ignitability, and storability. Broadly, they are categorized into:

13.8.4.1 Monopropellants

Monopropellants are further divided into:

- **Simple Monopropellants:** The fuel and oxidizer are part of the same molecule, such as in methyl nitrate (CH_3NO_3), which decomposes into CH_3O and NO_2.
- **Composite Monopropellants:** A mixture of fuel and oxidizer, such as nitric acid and amyl acetate, which undergo exothermic reactions.

13.8.4.2 Bipropellants

Bipropellants can be further classified based on ignitability into:

- **Hypergolic Propellants**
- **Non-Hypergolic Propellants**

13.8.4.3 Energy Classification

Liquid propellants are also categorized by energy content, which typically correlates with their specific impulse. They are divided into:

- **Low-Energy Propellants:** These have lower specific impulses and are used for simpler applications.
- **Medium-Energy Propellants:** Offering a balance between performance and complexity.
- **High-Energy Propellants:** Known for delivering high specific impulses, such as hydrogen-fluorine (H_2/F_2) and liquid hydrogen-liquid oxygen (LH_2/LOX).

The choice of liquid propellants depends on the mission requirements, such as performance, storability, and safety, making them highly versatile for various thruster propulsion applications.

13.8.4.4 Why LOX / LCH4 ?

The development of LOX/LCH4 propulsion technology has a rich history, dating back to the 1970s. Efforts during this period focused on creating technologies to store and utilize cryogenic propellants in space as a non-toxic alternative to traditional propellants, aiming to achieve high-performance spacecraft. Numerous programs explored non-toxic propellants suitable for spacecraft thrusters, including liquid hydrogen and liquid oxygen. However, using liquid hydrogen for propulsion proved to be complicated due to challenges in long-term storage and the necessity for engine pumps. The high boil-off rate of hydrogen, the large volume of storage tanks, and the complexity of redundant pump systems led to increased spacecraft mass and costs, which counteracted the advantages of high specific impulse offered by the H2/O2 combination. While LO2/LH2 performs well for short-duration upper stages, denser propellants are more favorable for smaller, long-duration spacecraft.

In the 1980s, liquid methane emerged as a superior hydrocarbon fuel for in-space applications, thanks to its clean-burning properties, non-sooting characteristics, compatibility with LO2, and ability to be pressure-fed. LOX/LCH4 outperforms hydrazine and electric propulsion systems in thrust capability and is recognized as a green propellant ideal for deep space missions and planetary landers, offering increased cargo capacity and high thrust along with substantial delta-V capabilities.

Methane/oxygen thruster engines offer potentially significant life cycle mission advantages compared to traditional thruster propellants used in the United States today. Figure 2.9 shows that liquid methane (LCH4) and liquid oxygen (LOX) propulsion is very competitive based on bulk density impulse compared to current booster and in-space propellant combinations traditionally used today.

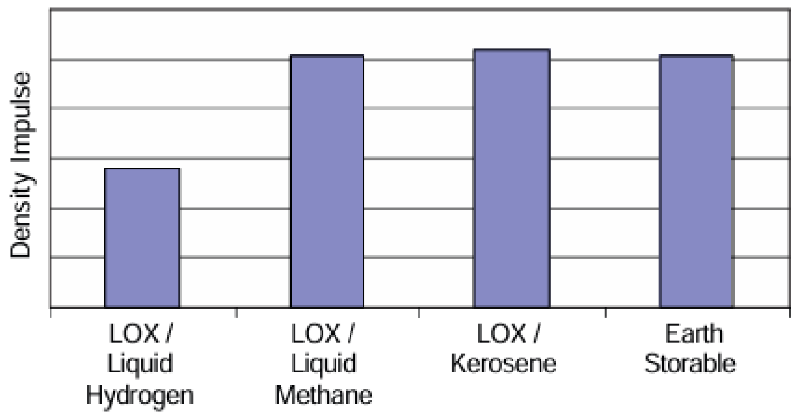


Figure 13.9 *Density impulse Comparisons show methane comparable with traditional propellants.*

The combination of LOX and LCH4 can significantly reduce spacecraft mass due to its higher specific impulse (Isp) when used with composite propellant tanks and cold-stored gaseous helium (GHe) pressurant. LOX/LCH4 is storable in space without requiring heaters like earth-storable propellants or active cooling like LH2. In certain environments, LOX and methane can be stored indefinitely at temperatures ranging from 90 to 120 K in deep space or for months in other orbits.

Additionally, LOX/LCH4 offers key reliability advantages for spacecraft propulsion. LOX is compatible with many materials, while methane is compatible with nearly all materials, and both are non-corrosive. The pressure-fed capability of LOX/LCH4 enhances reliability compared to LOX/LH2, which typically requires pump-fed systems.

These clean-burning, high vapor pressure propellants do not contaminate sensitive optics or damage surfaces, which is beneficial for platforms such as the International Space Station. Ground operations see significant improvements with LOX/LCH4, as these propellants are non-toxic and low-cost, facilitating rapid loading, testing, reusability, and clean turnaround operations for spacecraft and subsystems. Automated loading of spacecraft can occur concurrently with the launch vehicle, similar to LOX/LH2 upper stages, but without the complications of LH2 air liquefaction or toxic propellant issues. There is no need for hazardous propellant pre-loading at offsite facilities, allowing inert spacecraft to be transported and integrated with the launch vehicle easily. For launch pad operations, mobile propellant storage tanks, transfer lines, and cryogenic fluid couplings are necessary. The high vapor pressure of both propellants significantly enhances reusability, as they can be easily and safely inerted with GN2 and vented.

13.8.5 Propellant Tanks and Pressurization

In liquid bipropellant thruster engine systems, the propellants are stored in separate oxidizer and fuel tanks within the vehicle. Monopropellant thruster engines, by definition, use only one propellant tank. Typically, one or more high-pressure auxiliary gas tanks are included, which are used to pressurize the propellant tanks. However, there are tank pressurization methods that utilize heated gas from the engine, eliminating the need for additional heavy, high-pressure gas storage tanks.

The arrangement of tanks can vary, and the design, shape, and placement of the tanks can influence the vehicle's center of gravity. Typical arrangements are illustrated in Figure 2.10. Since propellant tanks must also fly, their mass is a significant factor, and they are often highly stressed. Common tank materials include aluminum, stainless steel, titanium, alloy steels, and fiber-reinforced plastics with a thin, impervious metal liner to prevent leakage through the porous walls of the fiber-reinforced material.

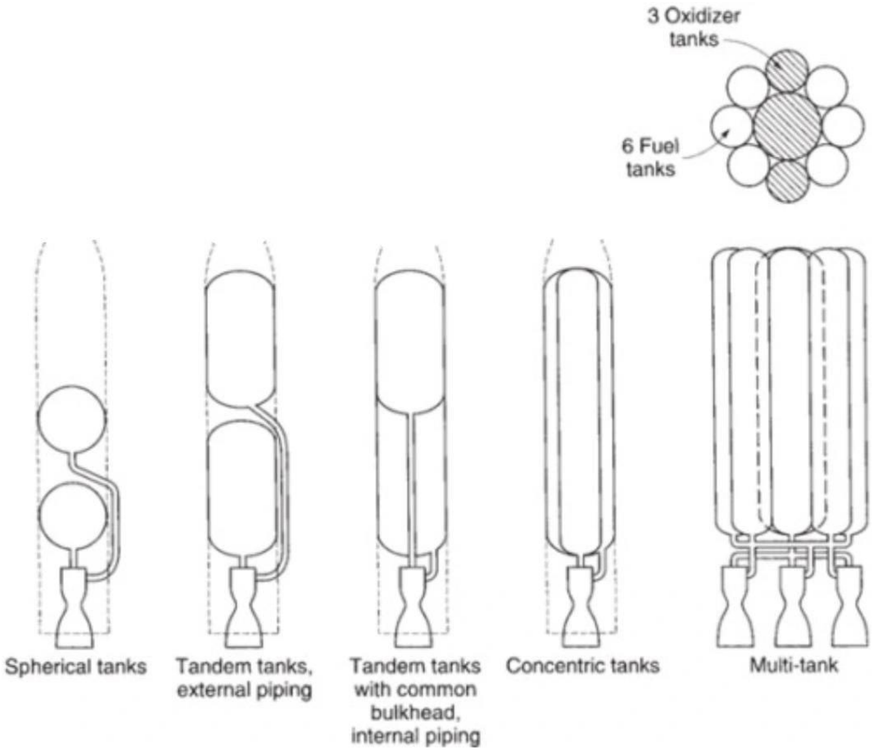


***Figure 13.10** Simplified Sketches of Typical Tank Arrangements for Large Turbopump-Fed Liquid Bipropellant Thruster Engines.*

The optimal shape for both propellant and gas pressurizing tanks is spherical, as this shape results in the least mass for a given volume. Small spherical tanks are often used in reaction control engine systems, where they can be easily integrated with other vehicle equipment. However, larger spherical tanks, which are necessary for primary propulsion systems, do not efficiently fill the available space in the vehicle. Therefore, larger tanks are often integrated into the vehicle's fuselage or wings, typically cylindrical with hemispherical or ellipsoidal ends, but they can also be irregular in shape.

There are several categories of tanks in liquid propellant propulsion systems, with relevant pressure values generally falling into the following:

- **Pressurized feed systems:** Propellant tanks in these systems typically operate at average pressures between 1.3 and 9 MPa (200 to 1800 psi). These tanks have thick walls and are relatively heavy.
- **High-pressure stored gas tanks:** Used to expel propellants, these tanks need to withstand pressures between 6.9 and 69 MPa (1000 to 10,000 psi). They are usually spherical to minimize inert mass, and several small spherical tanks may be connected together. In some vehicles, these smaller high-pressure gas tanks are placed inside the liquid propellant tanks.
- **Turbopump feed systems:** The propellant tanks in these systems must be pressurized slightly (to prevent pump cavitation) to average pressures between 0.07 and 0.34 MPa

(10 to 50 psi). These low pressures allow for thinner tank walls, resulting in lower inert tank mass for turbopump feed systems.

13.8.5.1 Tank Pressurization

As previously mentioned, the objective of feed systems is to move propellants under pressure from propellant tanks to the thrust chamber(s). The tank pressurization system is the part of the feed system that provides the propellant expellant gas.

Types of Tank Pressurization Systems:

- **Pressurized Gas Feed System:** A relatively high-pressure gas displaces the propellants from the tanks.
- **Pumped Feed System:** The main energy for feeding the propellants comes from one or more pumps. This system requires lower gas pressures in the tanks to move the propellants to the pump inlet, helping to avoid pump cavitation.

Sources of Pressurizing Gas Used in Tank Pressurization Systems:

- **High-Pressure Inert Gases:** Commonly used gases include helium, nitrogen, and air, stored at ambient temperature. When gases expand adiabatically, their temperature drops.
- **Heated High-Pressure Inert Gases:** Typically heated to between 200 and 800 °F (93 to 427 °C), which reduces the amount of required gas and thus the inert mass of the pressurizing system. Examples include gases heated by a heat exchanger using warm exhaust from a gas generator or turbine.
- **Gases Created by Chemical Reactions:** These can be derived from either liquid bipropellants, monopropellants, or solid propellants, resulting in "warm gas." The term "warm gas" typically refers to gases between 400 and 1600 °F (204 to 871 °C), distinguishing it from the "hot gas" in the main combustion chamber (4000 to 6000 °F or 2204 to 3319 °C). Chemically generated warm gases usually result in lighter tank pressurization systems compared to heated inert gas systems.
- **Evaporated Flow of Cryogenic Liquid Propellant:** A small portion of cryogenic liquid propellant, usually liquid hydrogen or liquid oxygen, can be evaporated by applying heat from a thrust chamber cooling jacket or from turbine exhaust gases, using part or all of this evaporated flow for tank pressurization.
- **Direct Injection of Hypergolic Fuel:** A small stream of hypergolic fuel can be injected into the main oxidizer tank and a small flow of hypergolic oxidizer into the fuel tank, though this has seen limited success.
- **Self-Pressurization of Cryogenic Propellants by Evaporation:** This method is feasible but can be difficult to control, with limited experience in this area.

Information Required for Designing and Analyzing a Pressurization System:

To design or analyze any pressurization system, it is necessary to have relevant information about the tank and the engine, which can include:

- Basic Engine Parameters: Such as propellant flow, thrust, duration, and pulse width.
- Propellant Tank Volume and Percent Ullage of Tank Volume.
- Storage Temperature Range.
- Properties of the Propellant and Pressurizing Gas.
- Propellant Tank Pressure and Gas Tank Pressure.
- Amount of Unavailable Residual Propellant.

13.8.6 Thrust Chambers

The thrust chamber is a critical component of a thruster engine, responsible for converting the chemical energy stored in the propellants into the kinetic energy needed to generate thrust. It comprises three main parts: an injector, a combustion chamber, and a nozzle.

In simple terms, the fuel and oxidizer are combined in the combustion chamber, creating high-temperature and high-pressure gases. These gases are then expanded through the nozzle, converting pressure and temperature into velocity. At the nozzle throat, the flow becomes choked thermodynamically, reaching sonic velocity. The combustion gases continue expanding at supersonic speeds through the diverging section of the nozzle. This process exchanges internal energy for kinetic energy, generating the momentum thrust required for propulsion.

This section will cover key elements associated with thrust chamber design, organized as follows: a comprehensive description of the thrust chamber is presented, detailing the underlying design concept and its applications. Then, various nozzle types used with the thrust chamber are explained.

As illustrated in Figure 2.11, the thrust chamber with an integral nozzle operates as follows: the propellants enter through the injector and undergo a series of complex physical and chemical processes such as atomization, vaporization, mixing, reaction, and expansion. The combustion chamber contains the high-pressure, high-temperature combustion gases and ensures stable combustion throughout the process. These gases are then expanded through the nozzle, with the diverging nozzle section (downstream of the throat) typically forming an integral part of the combustion chamber hardware. In many cases, a separate nozzle extension is added to further expand the gases and increase thrust.

For a liquid bi-propellant thruster engine, the combustion process can be summarized as follows:

- **Propellant Injection and Atomization:** the fuel and the oxidiser are injected into the combustion chamber at the correct oxidizer/fuel mixture ratio (O/F) and atomized into fine droplets.
- **Vaporization and Mixing:** These droplets vaporize as they absorb heat from the surrounding gases. Throughout this process, the droplets' size and velocity change, leading to rapid mixing and further heating of the vaporized propellants. The reaction between these vaporized propellants significantly increases the gaseous mass flow rate within the combustion chamber.
- **Combustion Process:** The gas-phase reactions, driven by high-speed diffusion of reactive molecules and atoms, continue as the gases flow toward the chamber throat. Combustion is generally completed upstream of the throat, ensuring that all droplets have vaporized. However, certain conditions may cause shock waves or pressure oscillations, leading to "combustion instability," which can produce destructive vibrations and heat flux. Thus, ensuring stable combustion is a significant aspect of thruster design and development.
- **Gas Acceleration and Ejection:** As the combustion products move toward and through the throat, they accelerate to sonic speeds and then to supersonic velocities within the expanding nozzle section, ultimately being expelled to generate thrust.

The principal components of a thrust chamber are the injector including the propellant inlets and distributing manifolds, the ignition device (which is necessary in case of a thruster engine burning non-hypergolic propellants), the combustion chamber, the converging portion of the nozzle between the inlet plane and the throat, and the diverging portion of the nozzle between the throat and the exit plane.

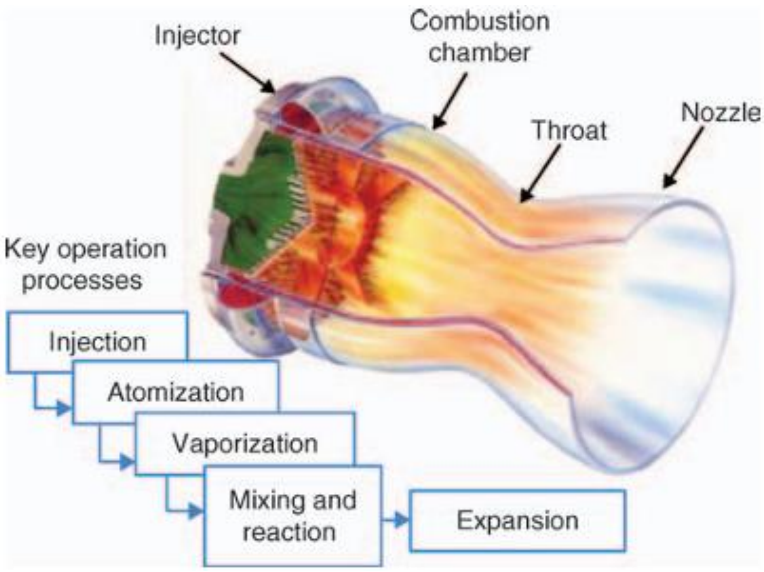


Figure 13.11 *Thrust Chamber with integral nozzle and key operation processes (courtesy of Pratt & Whitney Thrusterdyne.)*

13.8.6.1 Injectors

The injector is the part of a thruster engine in which the liquid fuel and liquid oxidizer are admitted into the combustion chamber, broken into particles or droplets to increase the contact surface areas, mixed, and vaporized before reacting in the combustion process. The injector terminates with a perforated plate marking the start of the combustion chamber.

Types of Injectors

Several types of injectors have been devised for thruster engines. They can be broadly divided into two categories: (1) nonimpinging and (2) impinging. Among nonimpinging types of injectors, three types, namely, (a) shower head injector, (b) coaxial injector, and (c) swirl atomizers, are used in liquid-propellant thruster engines. The impinging injectors are broadly classified into two: (a) unlike-impinging and (b) like-impinging injectors. All these injectors are discussed in detail in the following.

1. **Nonimpinging Injectors:**

a. **Shower-head Injector:** This type of non-impinging injector is one of the earliest designs, in which fuel and oxidizer are ejected perpendicularly from the injector face, resembling the flow from a water shower (see Figure 2.12a(A)). The axial streams of fuel and oxidizer create spray cones or sheets, which interact to promote atomization and mixing through turbulence and diffusion. However, this design tends to produce inefficient atomization and incomplete mixing, necessitating a longer combustion chamber for complete combustion. Despite these drawbacks, the shower-head injector is effective in cooling the combustion chamber walls, as the axial flow helps prevent heat transfer from the combustion zone to the walls. Additionally, it allows for easy throttling of the engine by adjusting the spray cone or sheet width using axially movable sleeves, without causing significant injection pressure drops. This concept was successfully used in the lunar excursion module, demonstrating throttling capabilities with a flow rate range of over 10:1 without substantially altering the mixture ratio.

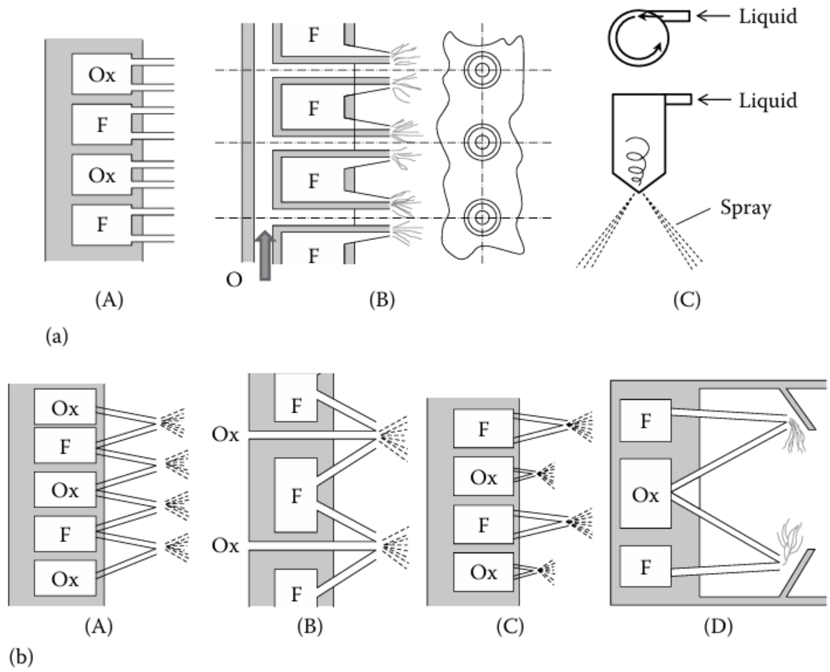


***Figure 13.12** Types of injector elements: (a) nonimpinging: (A) shower head, (B) co-axial injector, (C) swirl injector and (b) impinging: (A) unlike doublet, (B) unlike triplet, (C) like doublet, (D) splash plate (Ox, oxidizer; F, Fuel).*

b. **Coaxial Injector:** This is the most commonly used non-impinging injector, particularly suitable for non-hypergolic propellants and preferred in semicryogenic liquid-propellant thruster engines. It was first developed by NASA during the early experiments with cryogenic liquid-propellant engines. A typical coaxial injector, as shown in Figure 2.12a(B), consists of two concentric tubes with a recessed length. Generally, the liquid propellant (often liquid oxygen) is injected through the central tube at a relatively low velocity (below 30 m/s), while the gaseous fuel flows through the outer tube at a much higher velocity (over 300 m/s). The slower velocity of the liquid propellant allows for reduced injection speed into the recess area, while the high-velocity gaseous fuel shears the liquid surface into ligaments and then fine droplets, promoting better atomization and mixing. This results in a high-performance, stable injector, which is widely used in semicryogenic thruster engines that utilize gaseous fuel and liquid oxygen. The fuel surrounding the oxidizer helps prevent combustion instability and shields the combustion zone, reducing heat transfer to the combustion chamber walls. However, the performance of the coaxial injector declines significantly when used with two liquid streams, as it becomes difficult to achieve the optimal momentum flux ratio necessary for effective atomization.

c. **Swirl Injector:** In a swirl injector, the liquid propellant is injected tangentially into the injector chamber, causing the formation of a hollow conical sheet with a cone angle ranging from 40° to 100°, as shown in Figure 2.12a(C). This liquid sheet breaks into ligaments and eventually into fine droplets. As the swirl component increases, the cone angle becomes larger, resulting in a more uniform distribution of the droplets. This type of injector is commonly used for non-hypergolic bipropellants that require rapid vaporization and thorough mixing of fuel and oxidizer in the gas phase for efficient combustion. The swirling motion promotes enhanced atomization and facilitates the mixing needed for successful ignition and combustion.

2. Impinging Injectors

In impinging injectors, two or more streams of propellant jets collide with each other, causing the bulk liquid jet or sheet to break up into a fine spray. These injectors are widely used in thruster engines due to their superior performance, simplicity, and lower cost. While impinging injectors are generally preferred for non-hypergolic propellants, they have also been successfully used with hypergolic propellants.

Impinging injectors are categorized into two main types: (1) unlike-impinging injectors, where different propellant streams (fuel and oxidizer) impinge, and (2) like-impinging injectors, where streams of the same propellant impinge.

a. **Unlike-impinging injector:** In this case, two/three different liquid streams impinge on each other when they are issued from two/three angled orifices. Several types of unlike-impinging injectors have been developed for thruster engine applications. Some of them are (a) unlike-impinging doublets, (b) unlike-impinging triplets, which are discussed here:

- **Unlike-Impinging Doublets:** In an unlike-impinging doublet injector, two streams of fuel and oxidizer collide, creating a fan-shaped spray of the mixed liquids, as depicted in Figure 2.12b(A). The impact at the impinging point generates waves that convert the bulk liquid jet or sheet into ligaments, which then fragment into smaller droplets, enhancing atomization and the distribution of fuel and oxidizer in the combustion chamber. The disintegration of the jets results in spherical waves that propagate outward, influencing the extent of the atomization process. Key parameters affecting this process include jet diameter, momentum, injection pressure drop, chamber pressure, and the angle of impingement. When the streams are different, they form a two-dimensional fan-shaped spray in a plane, assuming no chemical reactions occur in the liquid phase. However, various factors, such as momentum mismatches and stream-diameter discrepancies, can distort the spray shape and size, leading to poor atomization and mixing. In hypergolic propellants, rapid chemical reactions can occur simultaneously with atomization, causing reactive-stream separation and affecting performance. Hypergolic propellants have short ignition delays, producing gases before the complete hydrodynamic impact, which can separate the reacting surfaces. Similar effects may occur in non-hypergolic systems, like liquid kerosene and liquid oxygen, particularly under high pressure. Additionally, combustion during the atomization of hypergolic propellants can alter the mixing and mass distribution of the injected spray. The presence of hot gas cross-flow and increased turbulence near the injector can lead to radial winds that deform the spray pattern, stripping away the rapidly atomizing portions of the injected propellants.

- **Unlike-Impinging Triplets:** Unlike-impinging triplet injectors address the issue of distorted spray fans that can occur in doublet injectors due to mismatches in stream size and momentum between fuel and oxidizer streams. This distortion can lead to poor atomization and mixing. In a triplet injector, a symmetrical axial central stream of one propellant is surrounded by two symmetrical impinging streams of another propellant, as shown in Figure 2.12b(B). The triplet configuration can consist of two fuel streams impinging on a single oxidizer stream (F-Ox-F) or two oxidizer streams impinging on a single fuel stream (Ox-F-Ox). The latter configuration is often preferred, as it provides a larger oxidizer area, which is beneficial for mixing, especially in fuel-lean conditions. However, care must be taken to avoid oxidizer-rich streaks near the combustion wall. The primary advantage of the unlike triplet injector is its enhanced mixing capabilities, resulting in higher combustion efficiency compared to doublet injectors. However, it is also more susceptible to combustion instability issues. Various combinations of unlike streams can be utilized to produce the spray, but this increases complexity. Although designs involving quadlets, pentads, or hexads can yield improved mixing, they often result in poorer mass distribution and are rarely used in practice due to their tendency toward instability. Despite these challenges, multiple-impinging injectors with a higher number of streams are advantageous in applications requiring high propellant flow rates.

b. Like-impinging injectors: In this case, two/three or more same liq uid streams are impinged on each other when they are issued from their respective angled orifices. This kind of injectors is also known as self-impinging injectors. Several types of like-impinging injectors have been developed for thruster engine applications. Some of them, namely, (a) like-impinging doublet and (b) like-impinging triplets, are in the following:

- **Like-Impinging Doublets:** In a like-impinging doublet injector, two streams of the same propellant collide, forming a fan-shaped spray of droplets, as shown in Figure 2.12b(C). The impact generates waves along the spray fan's two-dimensional surface, dissipating energy and converting the bulk liquid jet into ligaments that fragment into smaller droplets. Unlike unlike-impinging doublets, there is no mixing between the streams since they consist of the same liquid. The degree of mixing is influenced by the orientation of the initial fans for secondary impingements and the overlapping of sprays. Like-impinging injectors are typically used in thruster engines with non-hypergolic liquid propellants, as they mitigate the reactive demixing issues seen in unlike-impinging injectors, ensuring higher combustion stability. While they provide lower mixing levels than unlike-impinging doublets, effective design improvements can enhance combustion efficiency

- **Like-Impinging Triplets:** To overcome the issue of undesired shifts in the impinging point caused by mismatches in stream size and momentum, like-impinging triplet injectors allow three identical propellant streams to collide at a single point, as illustrated in Figure 2.12. These triplets typically produce a narrower spray fan with larger droplets compared to doublet injectors, which may result in overall performance losses. Additionally, smaller orifices are required to accommodate more triplet injectors within the same manifold surface area. Similar challenges can arise with other multiple-stream self-impinging injectors, such as quadlets and pentads, potentially complicating design and performance.

3. Other Types of Injectors

Several other types of injectors have been tried during the development of thruster engines. Two of them, namely, (1) splash plate and (2) premixing injectors, shown in Figure 2.12, are discussed:

a. Splash Plate Injector: The splash plate injector utilizes the principle of impingement in conjunction with a splash plate, as shown in Figure 2.12b(D). This design promotes the breaking of liquid jets or sheets, facilitating better mixing of the propellants in their liquid state. By directing the liquid streams against the splash plate, the injector minimizes the misalignment issues associated with the impinging points seen in doublet configurations. This feature enhances performance across a wide range of operating conditions and has been successfully applied to certain storable propellants.

b. Premixing Injector: In a premixing injector, the liquid fuel and oxidizer are mixed prior to being injected into the combustion chamber. The dimensions of the premixing chamber are influenced by the reaction time and the residence time of the propellant streams. It is crucial to avoid explosions of the premixed propellants within the chamber, particularly because flame can travel back from the combustion chamber, leading to dangerous conditions—especially under high-pressure and high-mass-flux scenarios. To mitigate this risk, swirls may be introduced into the liquid stream. Due to these safety concerns, premixing injectors are uncommon in thruster engines, although they can be used in the combustion chambers of other types of engines. While they have been tested with non-hypergolic propellants, they can cause excessive thermal loads on the injector structure due to potential precombustion in the premixing chamber. Consequently, the use of premixing injectors is typically reserved for addressing specific injection challenges.

13.8.6.2 Combustion Chambers

A liquid-thruster combustion chamber is designed to convert propellants into high-temperature, high-pressure gas through combustion, releasing the chemical energy of the propellant and increasing the internal energy of the gas. Traditionally, combustion chambers have been of tubular construction and are a critical part of the thrust chamber, where nearly all of the propellant burning takes place. The chamber serves as an envelope to retain the propellants long enough to ensure complete mixing and combustion, referred to as the "stay time."

Historically, three geometric shapes have been used in combustion chamber design: spherical, near-spherical, and cylindrical.Three geometrical shapes that have been used in combustion-chamber design are shown in Figure 2.13. A spherical combustion chamber has, in comparison with a cylindrical one of the same volume, a smaller mass and a smaller surface to be cooled. In addition, for the same pressure and for the same strength of the materials used, the walls of a spherical combustion chamber can be less thick than those of a cylindrical combustion chamber. On the other hand, a spherical combustion chamber is more difficult to manufacture and offers a lower performance than is the case with a cylindrical combustion chamber. However, despite these advantages, spherical chambers are more difficult to manufacture and generally provide poorer performance in other respects, leading to the more frequent use of cylindrical chambers.

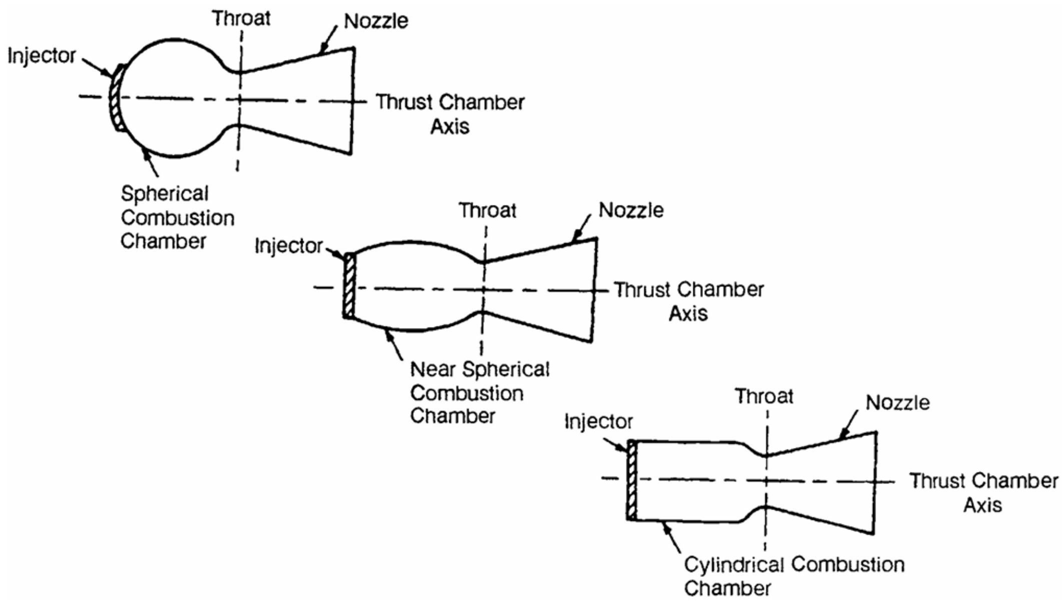


Figure 13.13 *Frequently used geometrical shapes for combustion chambers.*

13.8.6.3 Nozzles

The primary function of a thruster nozzle is to efficiently convert the thermal energy of combustion gases into kinetic energy, producing high exhaust velocities. Typically, thruster nozzles are converging-diverging (De Laval) designs, where gas is accelerated to sonic speed at the throat and to supersonic speed in the diverging section. The nozzle increases thrust by converting high-pressure, high-temperature gases into velocity.

Maximum thrust is achieved when the exhaust gas pressure at the nozzle exit matches the ambient pressure, a condition called optimum nozzle expansion.

When the exit pressure differs from the ambient pressure, two scenarios occur:

- **Over-expansion** (exit pressure less than ambient), leading to shocks and pressure adjustments through repeating compression and expansion waves (Mach disks).
- **Under-expansion** (exit pressure greater than ambient), where the flow expands and compresses in a series of Prandtl-Meyer expansions and compressions to reach pressure equilibrium.

Nozzle Shape

Most thruster nozzles are of the converging-diverging De Laval design. In the converging section, where gas flow velocity is relatively low, a smooth and well-rounded contour results in minimal energy losses. However, the diverging section's shape is critical due to the high flow velocities involved. The optimal nozzle shape for a specific expansion area ratio is guided by key design goals:

- Ensuring uniform, parallel, and axial gas flow at the exit for maximum momentum.
- Minimizing flow separation and turbulence losses.
- Keeping the nozzle as short as possible to reduce weight, space, friction losses, and cooling needs.
- Facilitating ease of manufacturing.

To avoid shock waves or turbulence, abrupt changes in the nozzle wall contour should be avoided. While the nozzle throat represents the minimum cross-sectional area, it is typically designed with a smooth, rounded contour. Only the nozzle exit features a sharp edge to prevent over-expansion and flow separation.

Various nozzle shapes have been studied, tested, and used in liquid thruster engines. The most common types include conical, contoured, plug (aerospike), and expansion-deflection nozzles and are shown in Figure1-14(a to d).

a. Conical nozzle:

Conical nozzles are simple to design and manufacture with flexibility in resizing. Thus, the conical nozzle was widely used in early experimental liquid thruster engines. This simple geometry, however, comes with the penalty of decreased performance due to the radial component of the exhaust gas velocity. Optimum-divergence cone half angles are generally between 12◦ and 18◦ as shown in Figure1-14a in terms of performance and nozzle size. Small divergence angles result in low divergence losses, but increase the noz zle length and weight. In contrast, large divergence angles increase the divergence losses, but result in lower length and weight.

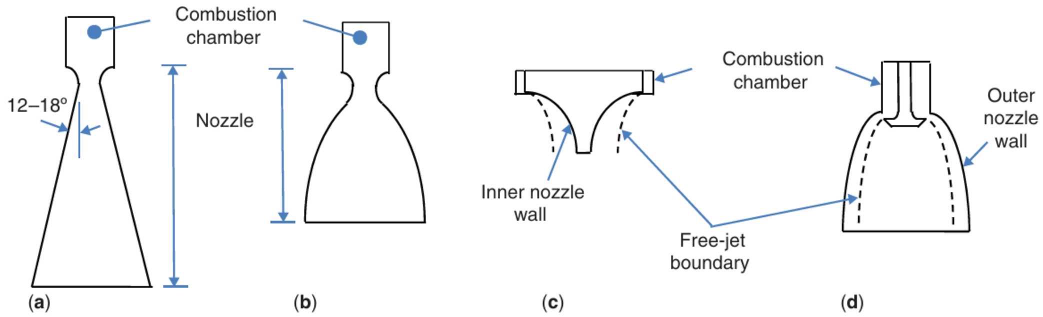


Figure 13.14 Most common nozzle shapes: (a) conical; (b) contoured; (c) plug; (d) expansion–deflection.

b. Contoured Nozzle:

Contoured nozzles offer superior performance and reduced length compared to conical nozzles, making them the preferred choice in modern liquid thruster engines. The most common configuration is the bell-shaped nozzle as shown in Figure 2.14b, which features an initial section with a high expansion angle (30°–60°) immediately after the throat. This is followed by a gradual reduction in the contour slope, resulting in a near-zero divergence angle at the nozzle exit.

The large divergence angles near the throat are made possible by the rapid acceleration of flow, which prevents flow separation when the nozzle contour is smooth. The sudden expansion behind the throat generates weak expansion waves, while the reversal of the contour slope creates compression waves as the flow is redirected. The expansion waves from the throat region counteract the compression effects, minimizing nozzle losses.

A nearly uniform distribution of exit velocity can be achieved with minimal divergence loss, but the length of an ideal bell-shaped nozzle is typically too long. Proper contour shaping, however, can reduce the nozzle length by 10–25% compared to a 15° conical nozzle. Bell-shaped nozzles are usually designed using Rao's parabolic approximation, derived from the method of characteristics.

Similar to thrust chamber design, the nozzle can be cooled using various techniques: (i) regenerative cooling, (ii) film cooling, (iii) transpiration cooling, (iv) ablative cooling, and (v) radiative cooling. The choice of cooling method depends on careful consideration of design trade-offs.

- **Parabolic approximation of the bell nozzle:**

The parabolic approximation, proposed by G. V. R. Rao, is a convenient method for designing near-optimum-thrust bell nozzles. The design configuration of a parabolic approximation bell nozzle is shown in Figure 2.15. In this design, the nozzle contour just upstream of the throat is a circular arc with a radius of 1.5 times the throat radius Rt. The divergent section consists of a circular entrance section with a radius of 0.382 Rt from the throat to point N, and a parabolic contour from N to the exit. The design process requires key data, such as the throat diameter, nozzle length, expansion area ratio, and the initial and exit wall angles. By selecting appropriate inputs, an optimal nozzle contour can be approximated accurately, with minimal influence from the specific heat ratio of the propellant.

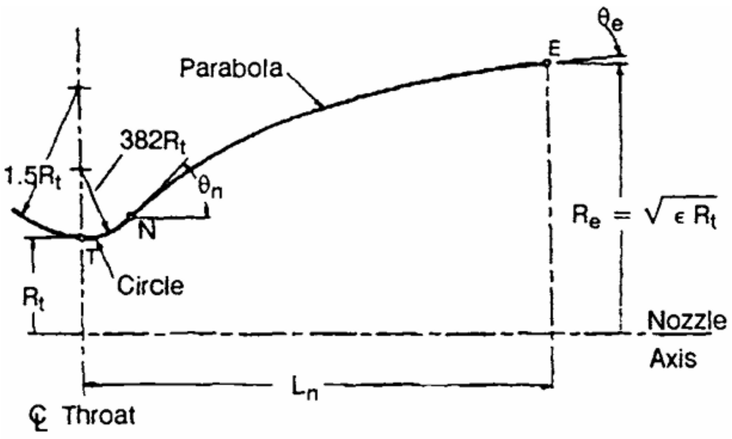


Figure 13.15 Parabolic approximation of bell nozzle.

c. Annular nozzles (plug and expansion–deflection) :

Comparedtoaconicalorbell-shapednozzle,annularnozzles are more complex to design and operate. There are two basic types of annular nozzles: (i) plug (or commonly known as aerospike) nozzle and (ii) expansion–deflection nozzle. As shown in Figure 2.14c, the plug nozzle has the outer surface of the annular flowasafree-jet boundary, whichisself-adjusted by ambient pressure. Similarly, the expansion–deflection nozzle has the inner free-jet boundary with the outer nozzle wall contour as shown in Figure 2.14d. Because of the altitude compensation characteristics with the free-jet boundary, annular nozzles are not subject to flow separation losses from over-expansion at low altitude as in the case of a conventional nozzle. Also, it allows a short nozzle design, potentially reducing interstage structural weight. However, annular nozzles require relatively high cooling require ments, heavier structural construction, and manufacturing complexity.

13.8.7 Computer-Aided Engineering (CAE) Tools

In designing and optimizing the LOX/LCH4 thruster engine, various Computer-Aided Engineering (CAE) tools are employed to model, simulate, and analyze key aspects of the engine's performance. These tools provide essential insights into combustion processes, thermal management, and fluid dynamics, allowing for effective design validation and optimization. This section highlights the main CAE tools used in the project.

13.8.7.1 Rocket Propulsion Analysis (RPA[23])

RP Software+Engineering UG offers Rocket Propulsion Analysis (RPA), a powerful software tool used for preliminary analysis and design of liquid Thruster engines. RPA is widely used in the industry due to its ability to calculate engine performance based on input parameters such as propellant type, chamber pressure, and mixture ratio. Key features of RPA include:

- **Thrust and Specific Impulse Calculation:** RPA provides detailed calculations of thrust, specific impulse (Isp), and propellant flow rates for various engine configurations.
- **Performance Optimization:** The software allows for optimization of key parameters such as expansion ratio and chamber pressure to achieve the desired performance levels.
- **Combustion Analysis:** RPA can simulate combustion conditions, predicting temperature, pressure, and gas composition in the combustion chamber and nozzle.

RPA serves as an essential tool for the initial design phase, providing a solid foundation for more detailed simulations using other CAE tools.

13.8.7.2 NASA CEA

CEA (Chemical Equilibrium with Applications) is a NASA software tool used to calculate the thermodynamic properties of combustion products and predict chemical equilibrium for a wide range of Thruster engine conditions. Key capabilities of CEA include:

Equilibrium Calculations: CEA calculates the composition, temperature, and pressure of combustion gases at equilibrium, providing valuable data for understanding the behavior of the propellants under different operating conditions.

Performance Estimation: The software is used to estimate important engine parameters such as specific impulse, thrust, and exhaust velocity based on the propellant chemistry and engine design.

Propellant Mixture Ratio Optimization: CEA helps optimize the LOX/LCH4 mixture ratio for maximum combustion efficiency and performance.

CEA NASA is particularly useful for validating combustion models and ensuring that the chemical reactions inside the engine are correctly represented.

The combination of RPA and CEA NASA provides a comprehensive toolkit for the design, simulation, and optimization of the LOX/LCH4 thruster engine. These tools allow for accurate performance predictions, detailed combustion analysis, and efficient design

[23] See https://rocket-propulsion.com/index.htm

optimization, ensuring the engine meets all mission requirements while operating efficiently and reliably.

13.9 Thruster Calculation

13.9.1 Calculation of Stoichiometric Mixture Ratio for LOX/LCH4

To determine the stoichiometric mixture ratio for the propellant combination of liquid oxygen (LOX) and liquid methane (LCH_4), the following steps are followed:

- **Step 1: Chemical Equation**

The combustion reaction for liquid methane with liquid oxygen can be expressed as:

$$CH4 + O2 \rightarrow CO2 + H2O \tag{13-1}$$

- **Step 2: Balancing the Equation**

The balanced chemical equation for the combustion of methane is:

$$CH4 + 2O2 \rightarrow CO2 + 2H2O \tag{13-2}$$

From this equation, it is evident that one mole of methane (CH4) reacts with two moles of oxygen (O2).

- **Step 3: Molar Mass Calculations**

The molar masses of the reactants are calculated as follows:

➢ **Molar mass of CH4 (methane):**

Carbon (C): 12.01 g/mol

Hydrogen (H): 1.008 g/mol×4=4.032 g/mol

Total: 12.01+4.032=16.042 g/mol

➢ **Molar mass of O2 (oxygen):**

Oxygen (O): 16.00 g/mol×2=32.00 g/mol

- **Step 4: Calculating the Stoichiometric Mixture Ratio**

The stoichiometric mixture ratio (O/F) can be determined using the formula:

$$O/F = \frac{moles\ of\ O_2}{moles\ of\ CH_4} \tag{13-3}$$

From the balanced equation, we find:

$$O/F = \frac{2\ moles\ of\ O_2}{1\ moles\ of\ CH_4} = 2 \tag{13-4}$$

- **Step 5: Mass Ratio Calculation**

To calculate the mass ratio, the following equation is used:

$$\text{Stoichiometric } Mass\ Ratio\ (O/F) = \frac{Mass\ of\ O_2}{Mass\ of\ CH_4} = \frac{2\times 32.00\ g/mol}{16.042\ g/mol} = \frac{64.00}{16.042} \approx 4 \tag{13-5}$$

This indicates that the stoichiometric mixture ratio by mass of LOX to LCH_4 is approximately 4. Thus, for every 1 kg of methane, about 4 kg of liquid oxygen is required for complete combustion.

- **Step 6: Adjusting for Practical Applications**

While the calculated value of 4 serves as a theoretical baseline, it is typically adjusted for practical applications, where the commonly cited stoichiometric mixture ratio falls around 3.2 to 3.5. This adjustment accounts for various factors such as:

Real-World Engine Efficiencies: Actual Thruster engines may not achieve perfect stoichiometric ratios due to design limitations or optimization for performance.

Combustion Characteristics: The combustion behavior can differ based on mixture preparation, injector design, and combustion chamber conditions.

Hydrogen Addition: In some designs, a small amount of hydrogen may be introduced to enhance combustion, effectively reducing the oxygen requirements.

13.9.2 Optimal Mixture Ratio Analysis

As previously discussed, the optimal mixture ratio for the LOX/LCH_4 propellant system may differ from the stoichiometric ratio, depending on performance and thermal constraints. This section utilizes NASA CEA calculations to determine the most efficient operational parameters for the engine by identifying the mixture ratio that maximizes specific impulse (Isp) while considering chamber temperature limits.

13.9.2.1 Parameters for the Analysis

NASA CEA will be used to conduct simulations for each mixture ratio within the specified parameters. The inputs for each simulation will include:

- **Fuel:** Liquid Methane (LCH_4)
- **Oxidizer:** Liquid Oxygen (LOX)
- **Chamber Pressure (Pc):** The simulations will be conducted at a chamber pressure of 300 psia, which represents typical operational conditions for many Thruster engines.
- **Chamber to exit pressure ratio:** The chamber-to-exit pressure ratio (Pc/Pe) will be set to 30,000,000,000 to simulate vacuum conditions, where the external pressure is nearly zero. In practical terms, we approximate the exit pressure Pe as 10^{-8} psia to reflect the near-vacuum environment of space.
- **Mixture Ratios:** The analysis will explore O/F mass ratios ranging from 2.0 to 4.6, with increments of 0.04. This range enables a comprehensive examination of how different LOX-to-LCH_4 proportions influence specific impulse and chamber temperature.
- **Area Ratio (A_e/A_t):** To ensure a feasible thruster design, the area ratio was limited to a maximum of 100, as the optimal area ratio from initial calculations was impractically

large for a vacuum environment. By setting $A_e/A_t = 100$, the design is adjusted to more realistic dimensions, balancing theoretical efficiency with manufacturable limits.

The simulations will yield specific impulse (Isp) and chamber temperature values for each mixture ratio, allowing for the identification of the optimal fuel and oxidizer proportion to maximize efficiency.

13.9.2.2 Analysis Based on Specific Impulse and Chamber Temperature

To identify the optimal mixture ratio for the LOX/LCH_4 propulsion system, both the specific impulse (Isp) and the chamber temperature as functions of the oxidizer-to-fuel (O/F) mass ratio were analyzed. The goal was to find a balance between maximizing performance (high Isp) and maintaining manageable thermal conditions (chamber temperature) to enhance engine durability and efficiency. The graphs below illustrate these variations.

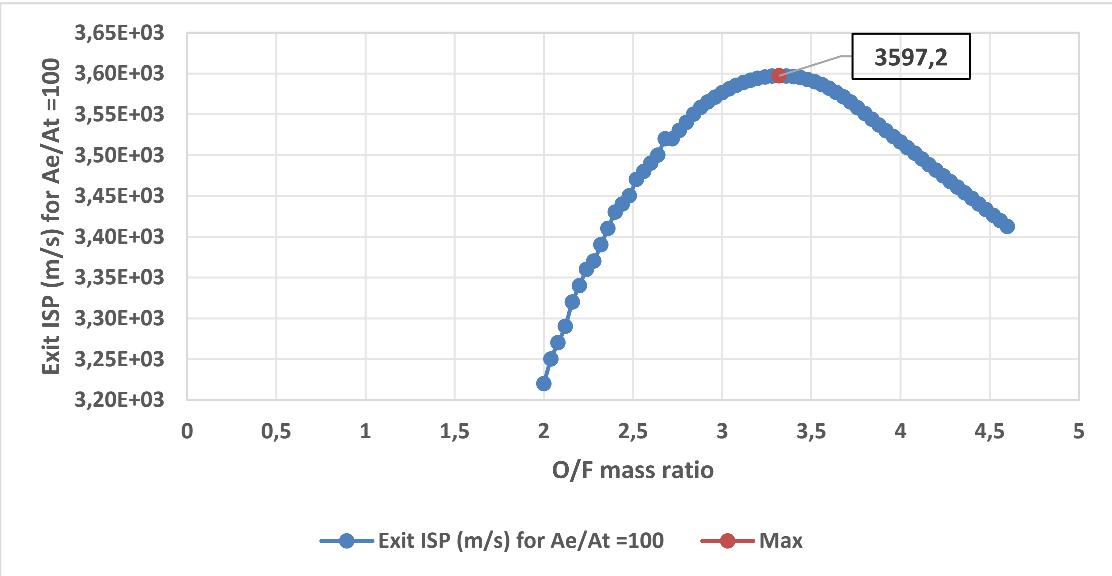


Figure 13.16 *Specific Impulse (Isp) vs. O/F Mass Ratio at 300 psia Chamber Pressure in Vacuum Conditions (NASA CEA Results)*

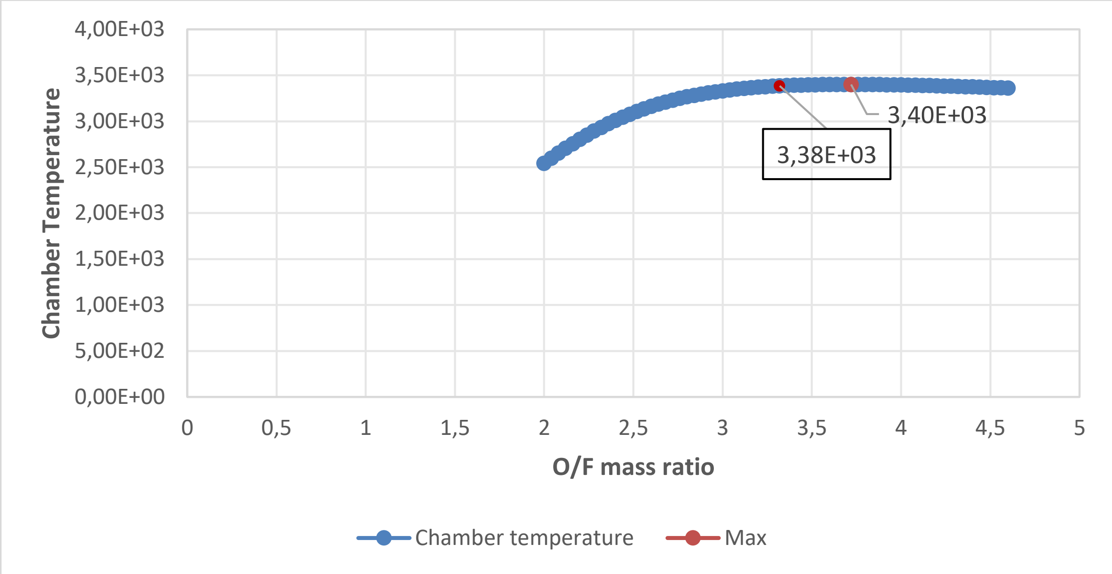


Figure 13.17 *Chamber Temperature vs. O/F Mass Ratio at 300 psia Chamber Pressure in Vacuum Conditions (NASA CEA Results)*

Specific Impulse and Chamber Temperature Comparison

The specific impulse curve, as shown in Figure 6.1, peaks at an O/F ratio of approximately 3.32, where the exit Isp reaches around 3597.2 m/s. This mixture ratio offers the highest efficiency in terms of thrust production for the LOX/LCH_4 system, making it an attractive option from a performance standpoint.

In contrast, Figure 6.2 reveals that the chamber temperature continues to rise with the O/F ratio, reaching a maximum at an O/F ratio of around 3.72, where the temperature is approximately 3400 K. However, at the O/F ratio that maximizes specific impulse (3.32), the chamber temperature is slightly lower than this peak, around 3380 K.

Determining the Optimal Mixture Ratio

This discrepancy between the O/F ratios for maximum Isp and maximum chamber temperature highlights an important trade-off. Operating at the O/F ratio of 3.32 achieves the highest specific impulse, thereby maximizing thrust efficiency. However, the chamber temperature at this ratio is still very close to the maximum observed temperature, which could impose significant thermal stress on the combustion chamber and require advanced cooling solutions.

The optimal mixture ratio for the LOX/LCH_4 system is identified as MR = 3.32, where specific impulse is maximized, and the chamber temperature remains near, but not at, the peak level. This selection provides a balance between maximizing performance and maintaining thermal conditions that support engine durability.

13.9.3 Performance Parameters of the Thruster at Optimal Mixture Ratio Analysis

In this section, we analyze key performance parameters for the LOX/LCH_4 propulsion system at the optimal mixture ratio of 3.32, as determined in Section 3.2. Using NASA CEA data, we explore the performance characteristics across three main regions of the rocket nozzle: the chamber, throat, and exit. This analysis includes values for specific impulse, thrust coefficient, and other thermodynamic properties critical to understanding the engine’s efficiency and thermal management requirements.

13.9.3.1 Definitions of NASA CEA Output Parameters

- **Thrust Coefficient (Cf):** A dimensionless parameter that measures the efficiency of a rocket nozzle, defined as the thrust produced per unit pressure in the combustion chamber.
- **Mach Number:** A dimensionless quantity representing the ratio of the speed of the flow to the speed of sound in that medium, indicating whether the flow is subsonic, sonic, or supersonic.
- **Gamma (γ):** The specific heat ratio of the gas, defined as the ratio of specific heat at constant pressure (Cp) to specific heat at constant volume (Cv), influencing the thermodynamic behavior of the flow.
- **Cross-section Area Ratio (Ae/At):** The ratio of the exit area (Ae) of the nozzle to the throat area (At), used to evaluate the expansion and acceleration of the exhaust gases through the nozzle.
- **Pressure (P):** The static pressure of the gas at various points in the propulsion system, measured in bar, indicating the energy available for propulsion.
- **Sonic Velocity (a):** The speed at which pressure waves travel through the gas, dependent on the gas properties and temperature, critical for determining flow characteristics in the nozzle.
- **Density (ϱ):** The mass per unit volume of the gas, measured in kg/m^3, affecting both the mass flow rate and the thrust generated by the engine.
- **Enthalpy (h):** The total heat content of the gas, measured in J/kg, representing the energy available for doing work during expansion.
- **Entropy (s):** A measure of the disorder or randomness of the gas, expressed in J/(kg·K), indicating the irreversibility of processes and the energy unavailable for work.
- **Gibbs Energy (G):** The energy associated with a system that can perform work at constant temperature and pressure, measured in J/kg, useful for evaluating the feasibility of chemical reactions.
- **Internal Energy (u):** The total energy contained within the gas due to its temperature and state, expressed in J/kg, influencing the thermodynamic efficiency of the engine.
- **Molecular Weight (M):** The mass of a molecule of gas, expressed in g/mol, affecting the gas properties and performance of the propulsion system.

- **Temperature (T):** The absolute temperature of the gas in Kelvin (K), which influences the thermodynamic properties and performance characteristics of the engine.

13.9.3.2 Performance Parameters from NASA CEA

The data obtained from the NASA CEA simulations provide critical insights into the performance parameters for each station (Chamber, Throat, Exit and Exit at Ae/At = 100) of the LOX/LCH$_4$ propulsion system at the optimal mixture ratio of 3.32. The results are summarized in Tables 3.1.a, 3.1.b and 3.1.c.

Station	Thrust Coefficient (Cf)	Mach Number	Gamma (γ)	(Ae/At)	Pressure (P, bar)
Chamber	0.0000E+00	0.0000E+00	1.1258E+00	0.0000E+00	2.0684E+01
Throat	6.5121E-01	1.0000E+00	1.1228E+00	1.0000E+00	1.1997E+01
Exit	2.3933E+00	1.7621E+01	1.0000E+00	2.3123E+08	6.8947E-10
Exit at Ae/At = 100	1.9740E+00	4.5072E+00	1.2059E+00	1.0000E+02	1.6667E-02

a

Station	Sonic Velocity (m/s)	Specific Heat (Cp, kJ/(kg·K))	Specific Impulse (Isp, m/s))	Density (ρ) (kg/m³)	Enthalpy (h) (J/kg)
Chamber	1.2246E+03	8.2199E+00	0.0000E+00	1.5527E+00	-1.5993E+03
Throat	1.1867E+03	7.9788E+00	1.1867E+03	9.5647E-01	-2.3034E+03
Exit	2.4750E+02	1.2656E+06	4.3613E+03	1.1255E-09	-1.1110E+04
Exit at Ae/At = 100	7.9809E+02	2.1095E+00	3.5972E+03	3.1554E-03	-8.0691E+03

b

Station	Entropy (s) (J/(kg·K))	Gibbs Energy (G) (J/kg)	Internal Energy(u) (J/kg)	Molecular Wt. (M) (mw) (g/mol)	Temperature (t) (K)	Pc/Pe
Chamber	1.2733E+01	-4.4689E+04	-2.9314E+03	2.1121E+01	3.3840E+03	1.0000E+00
Throat	1.2733E+01	-4.3437E+04	-3.5577E+03	2.1414E+01	3.2304E+03	1.7241E+00

Exit	*1.2733E+01*	*-1.3550E+04*	*-1.1171E+04*	*2.3120E+01*	*1.9164E+02*	3.0000E+10
Exit at Ae/At = 100	*1.2733E+01*	*-2.6756E+04*	*-8.5973E+03*	*2.3101E+01*	*1.4676E+03*	1.2410E+03

c

Table 13.1 *performance parameters for each station (Chamber, Throat, Exit and Exit at Ae/At = 100) at the optimal mixture ratio (NASA CEA Results)*

13.9.3.3 Interpretation of NASA CEA Results

The NASA CEA results provide detailed insights into the performance parameters and thermodynamic properties of the LOX/LCH$_4$ thruster engine at various critical points: the chamber, throat, and exit, along with an additional exit condition using a limited area ratio (Ae/At = 100). These parameters are crucial for assessing the engine's efficiency, thermal behavior, and feasibility of design in vacuum conditions. The following is an interpretation of key findings based on these results:

Operating Conditions and Inputs**:**

- The chamber pressure is set to 300 psia with an extremely high chamber-to-exit pressure ratio of 3×10^{10}, simulating an ideal vacuum.
- An oxidizer-to-fuel ratio (O/F) of 3.32 is used, close to the specific impulse peak, optimizing performance.
- The analysis considers liquid methane (CH_4) as the fuel and liquid oxygen (O_2) as the oxidizer, with full combustion assumed.

Performance Parameters:

- **Thrust Coefficient (Cf):** The thrust coefficient shows a significant increase from the throat to the exit, with a maximum value of 2.3933 at the exit. This indicates the high efficiency of the nozzle in converting combustion energy into thrust. However, at an area ratio of 100, the exit thrust coefficient drops slightly to 1.9740. This more practical area ratio still allows for efficient thrust production while maintaining structural feasibility.
- **Effective Exhaust Velocity (c):** Effective exhaust velocity is the speed at which exhaust gases effectively exit the nozzle, considering both the exit velocity v_e and pressure differential at the exit Ae(pe−pa).

$$c = V_e + A_e(P_e - P_a) \tag{13-6}$$

In vacuum conditions where pe=pa=0, c=v_e. for this engine, the effective exhaust velocity c at the vacuum exit is 4361.3 m/s, representing the highest achievable efficiency. With Ae/At limited to 100, c slightly decreases to 3597.2 m/s, providing a more practical value without a severe loss of performance. This compromise between maximum Isp and realistic area ratios helps balance performance with feasible nozzle dimensions.

- **Specific Impulse (I_{sp}):** Specific impulse measures the thrust generated per unit weight flow rate of the propellant, typically in seconds. Since NASA CEA provides c, the specific impulse can be derived by dividing c by gravitational acceleration g≈9.81 m/s^2, yielding:

$$Isp = \frac{c}{g} = \frac{3597.2\ m/s}{9.81\ m/s^2} = 366.687\ s \tag{13-7}$$

- **C*, (m/s):** Characteristic velocity, indicating the performance of the engine in terms of combustion efficiency. C* remains constant across all stations at 1822.3 m/s, indicating stable combustion efficiency.

Thermodynamic Properties:

- **Temperature (T):** The temperature in the chamber reaches 3384 K, aligning with the high energy release of the combustion process. At the vacuum exit, the temperature falls to 191.64 K, while at Ae/At = 100, the exit temperature is 1467.6 K, a more reasonable temperature for material durability and nozzle design.
- **Density (ϱ):** Density decreases from 1.5527 kg/m^3 in the chamber to near vacuum levels at the exit, due to gas expansion. For Ae/At = 100, density is 0.00315 kg/m^3, indicating a lower but manageable expansion for real-world applications.
- **Sonic Velocity (a):** The sonic velocity decreases from 1224.6 m/s in the chamber to 247.5 m/s at the vacuum exit, reflecting cooling and expansion of the exhaust gases. For Ae/At = 100, the sonic velocity remains high at 798.09 m/s, allowing for a controlled supersonic flow within feasible design limits. It reflects the thermal energy being converted into kinetic energy as the gas expands, accelerating the flow to supersonic speeds downstream of the throat.
- **Specific Heat (Cp):** The specific heat Cp decreases progressively from 8.22 kJ/(kg·K) in the chamber to 7.98 kJ/(kg·K) at the throat and 2.11 kJ/(kg·K) at the exit. This reduction reflects the energy transformation as gases expand and cool. High Cp in the chamber supports efficient energy release during combustion, while the lower Cp at the exit indicates successful energy conversion to achieve high exhaust velocity.
- **The enthalpy** and **internal energy** values provide insight into the energy available for work at each station, with both values dropping significantly as the exhaust gases expand.
- **The entropy** remains constant across stations, suggesting minimal irreversibility in the expansion process, while **Gibbs energy** values reflect the thermodynamic feasibility of reactions at each stage.

Flow and Expansion Characteristics:

- **Mach Number:** In the chamber, the Mach number is zero, as expected in subsonic regions. It reaches unity at the throat, marking the transition to supersonic flow, and increases dramatically to 17.621 at the ideal exit, demonstrating high gas acceleration under vacuum conditions. For an area ratio of 100, the exit Mach number is reduced to 4.5072, providing a controlled supersonic expansion that remains manageable for structural and thermal stability.
- **Gamma (γ):** The specific heat ratio (γ) remains relatively stable across stations but decreases slightly at the exit, which indicates changes in gas properties at low pressures. This variation reflects shifts in specific heat capacities during the expansion process.

Molecular Properties:

- **Molecular Weight (M):** The molecular weight slightly increases from the chamber to the exit, with values ranging from 21.121 to 23.101 g/mol. This gradual increase reflects changes in gas composition due to varying pressure conditions during expansion.

Design Feasibility:

- **Cross-section Area Ratio (Ae/At):** An area ratio of 2.3123×10^{8} is required for ideal vacuum expansion, which is unfeasible in practical applications. Limiting the area ratio to 100 provides significant expansion while keeping the design achievable.
- **Pressure (P):** Pressure decreases dramatically from 20.684 bar in the chamber to nearly zero at the exit under ideal vacuum. With Ae/At = 100, the exit pressure is 0.01667 bar, providing a feasible low-pressure environment for significant expansion without compromising nozzle integrity, facilitating the acceleration of exhaust gases to high velocities.

13.9.4 Thrust and Mass Flow Calculations

13.9.4.1 Total Mass Flow Rate:

To determine the total mass flow rate $\dot{m}$, we use the relationship between thrust F, effective exhaust velocity c, and total mass flow rate as follows:

$$\dot{m} = \frac{F}{c} \tag{13-8}$$

Given an assumed thrust force of 5 kN and the effective exhaust velocity c derived from NASA CEA as 3597.2 m/s, we obtain $\dot{m}$ = 1.39 Kg/s

This parameter is critical for determining the specific fuel and oxidizer flow rates required to sustain the desired thrust.

13.9.4.2 Fuel and Oxidizer Mass Flow Rates:

The mass flow rates for fuel $\dot{m}_{CH4}$ and oxidizer $\dot{m}_{O2}$ are derived based on the mixture ratio O/F MR=3.32. Using the relationships:

$$\dot{m}_{CH_4} = \frac{\dot{m}}{1+MR} = \frac{1.39}{1+3.32} = 0.3217\ Kg/s \tag{13-9}$$

$$\dot{m}_{O_2} = \frac{\dot{m}}{1+{}^{1}/_{MR}} = \frac{1.39}{1+{}^{1}/_{3.32}} = 1.0682\ Kg/s \tag{13-10}$$

These calculations provide the basis for configuring the propellant feed system, ensuring accurate delivery of both fuel and oxidizer to meet thrust requirements while maintaining the optimal mixture ratio.

13.9.5 Nozzle Geometry Calculations

13.9.5.1 Throat Area and Diameter

The throat area At is a key design parameter that supports the nozzle's ability to choke the flow and achieve critical conditions. It is calculated using the equation for characteristic velocity (c*):

$$c^* = \frac{P_c \times A_t}{\dot{m}} \tag{13-11}$$

where c* is provided by NASA CEA, and Pc is the chamber pressure assumed as 300 psia = 2.06843 x 10^6 Pa.

Rearranging for At:

$$A_t = \frac{\dot{m} \times c^*}{P_c} = \frac{1.39\ Kg/s \times 1822.3\ m/s}{2.06843\ x\ 10^6\ Pa} = 1.2246 \times 10^{-3}\ m^2 \tag{13-12}$$

The throat diameter (Dt) is then:

$$D_t = \sqrt{\frac{4\ A_t}{\pi}} = \sqrt{\frac{4 \times 1.2246 \times 10^{-3}}{\pi}} = 0.039486\ m = 3.9486\ cm \tag{13-13}$$

13.9.5.2 Exit Area and Diameter

Using the area ratio Ae/At = 100, the exit area Ae can be calculated as follows:

$$A_e = 100 \times A_t = 100 \times 1.2246 \times 10^{-3} = 0.12246\ m^2 \tag{13-14}$$

The exit diameter De is then calculated as:

$$D_e = \sqrt{\frac{4\ A_e}{\pi}} = \sqrt{\frac{4 \times 0.12246}{\pi}} = 0.39486\ m = 39.486\ cm \tag{13-15}$$

13.9.6 Combustion Chamber Geometry

13.9.6.1 Combustion Chamber Area and Diameter

Assuming a contraction ratio Ac/At = 4, the combustion chamber area Ac is calculated as:

$$A_c = 4 \times A_t = 4 \times 1.2246 \times 10^{-3} = 4.8983 \times 10^{-3}\ m^2 \tag{13-16}$$

the chamber diameter Dc is then calculated as:

$$D_c = \sqrt{\frac{4\ A_c}{\pi}} = \sqrt{\frac{4 \times 4.8983 \times 10^{-3}}{\pi}} = 0.078973\ m = 7.8973\ cm \tag{13-17}$$

13.9.6.2 Combustion Chamber Volume

The volume Vc of the combustion chamber is essential for maintaining a stable combustion process. It is calculated using the characteristic length L* and throat area At as:

$$V_C = L^* \times A_t \tag{13-18}$$

The characteristic length, L*, is a design parameter in rocket engine combustion chambers that represents the effective length required for efficient mixing and combustion of propellants. In practical terms, L* is an empirical measure used to ensure that the combustion chamber has sufficient volume and residence time for the propellants to mix, ignite, and burn completely before reaching the nozzle throat. Typical values of L* vary depending on the propellant combination and combustion chamber design, generally ranging from 0.8 to 1.6 meters for liquid-propellant thruster engines.

With an assumed characteristic length L*=0.8m, we obtain:

$$V_C = 0.8 \times 1.2246 \times 10^{-3} = 9.7966 \times 10^{-4}\ m^3$$

13.9.6.3 Combustion Chamber Length

Using the calculated chamber volume and area, the combustion chamber length Lc is derived as:

$$L_C = \frac{V_C}{A_C} = \frac{9.7966 \times 10^{-4}}{4.8983 \times 10^{-3}} = 0.2\ m = 20\ cm \tag{13-19}$$

13.9.6.4 Propellant Stay Time in the Combustion Chamber

The propellant stay time ts in the combustion chamber is a crucial parameter for ensuring proper combustion. It is determined by the relationship:

$$t_s = \frac{V_C}{\dot{m} \times v} \tag{13-20}$$

Where:

v is the specific volume of the propellant, calculated using combustion chamber pressure, chamber temperature, and molecular weight obtained from NASA CEA.

The specific volume v is derived from the ideal gas law:

$$v = \frac{RT}{P_C} \qquad (13\text{-}21)$$

Where:

- R is the specific gas constant, calculated using the universal gas constant Ru=8314 J/(kmol.K) and the molecular weight MW=21.121 kg/kmol from NASA CEA:

$$R = \frac{R_u}{MW} = \frac{8314}{21.121} = 393.64\ J/kg.K \qquad (13\text{-}22)$$

- T=3384.02 K is the chamber temperature from NASA CEA.
- P_C=300 psia=2.068×10^6 Pa is the chamber pressure.

Now, substituting values into the specific volume equation:

$$v = \frac{393.64 \times 3384.02}{2.068 \times 10^6} = 0.644\ m^3/kg \qquad (13\text{-}23)$$

substituting values into the propellant stay time ts equation:

$$t_s = \frac{9.7966 \times 10^{-4}}{1.39 \times 0.644} = 1.09 \times 10^{-3} s = 1.09\,ms \quad (13\text{-}24)$$

This value demonstrates that the propellant remains in the chamber for an adequate amount of time to allow for complete mixing and combustion.

13.9.7 RPA Results Interpretation

To confirm the validity of the design parameters obtained from NASA CEA, we used the Rocket Propulsion Analysis (RPA) software. The RPA provided additional design parameters and confirmed the consistency of the key performance metrics, such as thrust, specific impulse, mass flow rates, and chamber geometry. The results from RPA align closely with those calculated using NASA CEA, providing confidence in the accuracy and reliability of the design.

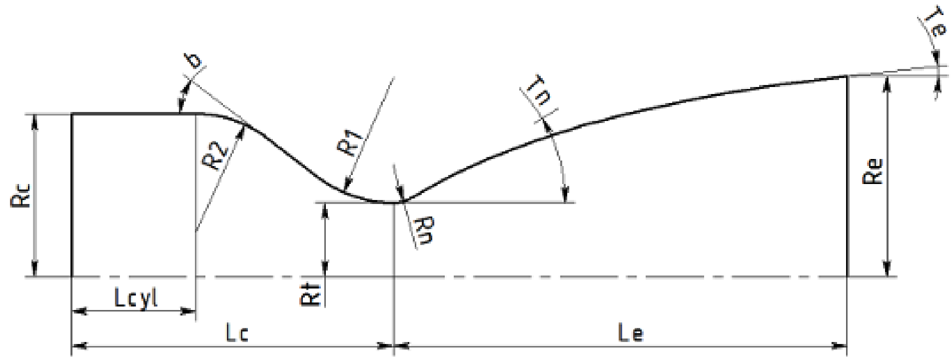


Figure 13.18 *Design parameters by RPA tool*

Thrust and mass flow rates			
Chamber thrust (vac)		5 kN	
Specific impulse (vac)		362.46736 s	
Chamber thrust (opt)		4.79335 N	
Specific impulse (opt)		347.48677 s	
Total mass flow rate		1.40663 kg/s	
Oxidizer mass flow rate		1.08102 kg/s	
Fuel mass flow rate		0.32561 kg/s	
Geometry of thrust chamber with parabolic nozzle			
Dc	78.77 mm	Le	474.39 mm
b	30 deg	Te	10.52 deg
R2	58.73 mm	De	393.87 mm
R1	29.54 mm	Ae/At	100

L*	800 mm	Le/Dt	12.04
Lc	221.62 mm	Le/c15 relative to length of cone nozzle with Te=15 deg	71.61 %
Lcyl	163.86 mm	Mass	9.06 kg
Dt	39.39 mm	Divergence efficiency	0.98417
Rn	7.52 mm	Drag efficiency	0.99157
Tn	38.84 deg	Thrust coefficient	2.00888 (vac)

Table 13-2 *Geometry of Thrust Chamber with Parabolic Nozzle by RPA tool*

These geometric parameters from RPA closely match those calculated through NASA CEA, further verifying the consistency and correctness of the design.

The data obtained from RPA thus confirms the performance predictions and supports the validity of the parameters used for the LOX/LCH_4 thruster engine design. This alignment enhances the confidence in our engine's expected behavior and performance during orbital transfer missions.

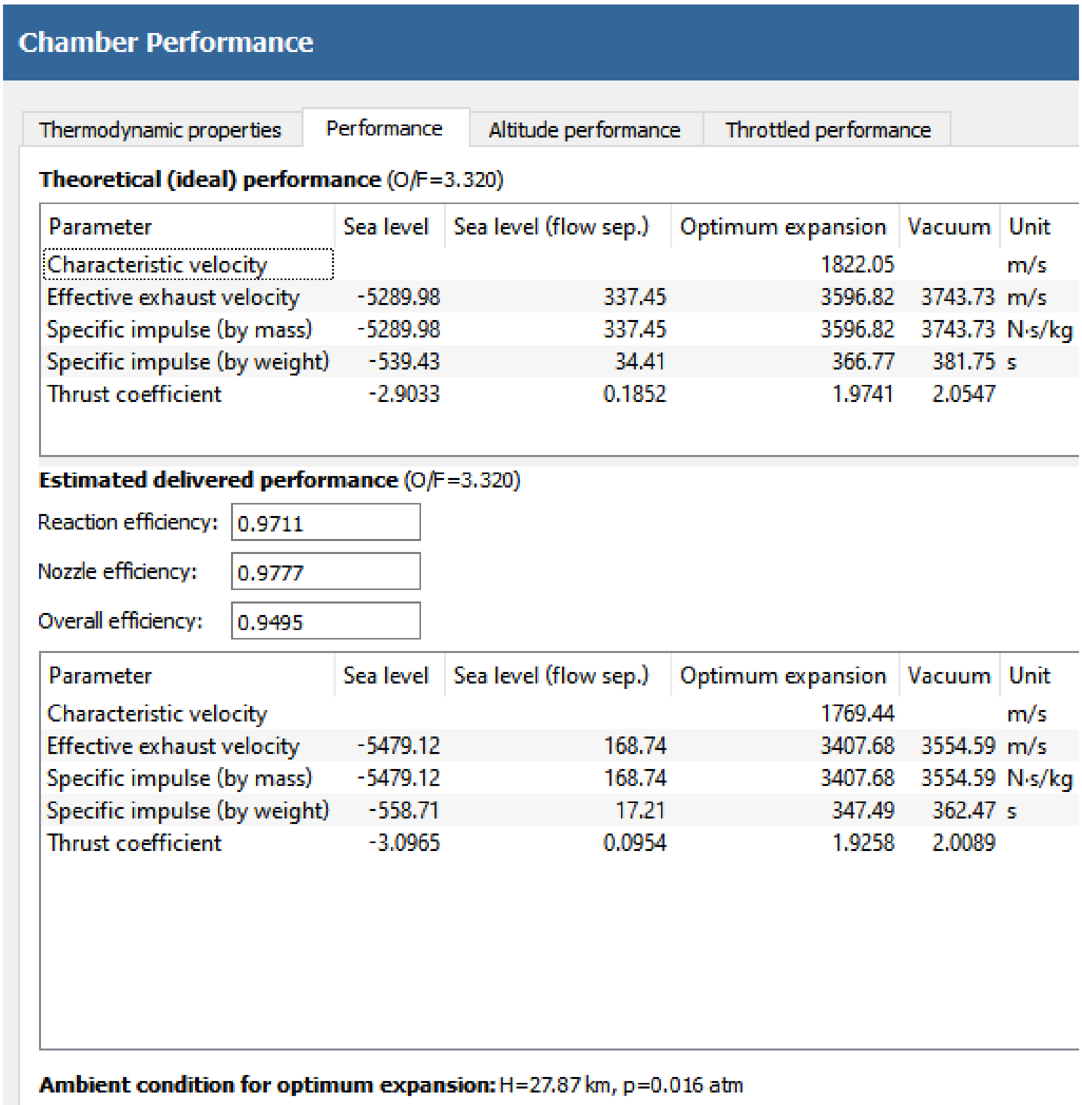
Chamber Performance

Thermodynamic properties | Performance | Altitude performance | Throttled performance

Thermodynamic properties (O/F=3.320)

Parameter	Injector	Nozzle inlet	Nozzle throat	Nozzle exit	Unit
Pressure	2.0684	2.0168	1.1848	0.0016	MPa
Temperature	3384.0145	3378.6817	3228.9549	1468.1145	K
Enthalpy	-1599.2878	-1616.3146	-2303.1804	-8067.8605	kJ/kg
Entropy	12.7334	12.7383	12.7383	12.7383	kJ/(kg·K)
Internal energy	-2931.4055	-2946.1026	-3557.1059	-8596.2680	kJ/kg
Specific heat (p=const)	8.2198	8.2300	7.9934	2.1096	kJ/(kg·K)
Specific heat (V=const)	7.0007	7.0106	6.8594	1.7495	kJ/(kg·K)
Gamma	1.1741	1.1739	1.1653	1.2059	
Isentropic exponent	1.1258	1.1257	1.1227	1.2059	
Gas constant	0.3937	0.3936	0.3883	0.3599	kJ/(kg·K)
Molecular weight (M)	21.1215	21.1251	21.4104	23.1007	
Molecular weight (MW)	0.02112	0.02113	0.02141	0.0231	
Density	1.5527	1.5166	0.9449	0.0031	kg/m^3
Sonic velocity	1224.6454	1223.4965	1186.5014	798.2384	m/s
Velocity	0.0000	184.5360	1186.5014	3596.8244	m/s
Mach number	0.0000	0.1508	1.0000	4.5060	
Area ratio	4.0000	4.0000	1.0000	100.0000	
Mass flux	279.8706	279.8706	1121.1271	11.2163	$kg/(m^2 \cdot s)$
Mass flux (relative)	1.353e-04	1.388e-04			kg/(N·s)
Viscosity	0.0001087	0.0001086	0.0001055	6.063e-05	kg/(m·s)
Conductivity, frozen	0.3793	0.3788	0.3631	0.1694	W/(m·K)
Specific heat (p=const), frozen	2.335	2.335	2.324	2.012	kJ/(kg·K)
Prandtl number, frozen	0.6692	0.6693	0.6749	0.7201	
Conductivity, effective	1.75	1.751	1.629	0.1708	W/(m·K)
Specific heat (p=const), effective	8.22	8.23	7.993	2.017	kJ/(kg·K)

***Table 13-3** Thermodynamics properties by RPA tool*



Chamber Performance

Thermodynamic properties | Performance | Altitude performance | Throttled performance

Theoretical (ideal) performance (O/F=3.320)

Parameter	Sea level	Sea level (flow sep.)	Optimum expansion	Vacuum	Unit
Characteristic velocity			1822.05		m/s
Effective exhaust velocity	-5289.98	337.45	3596.82	3743.73	m/s
Specific impulse (by mass)	-5289.98	337.45	3596.82	3743.73	N·s/kg
Specific impulse (by weight)	-539.43	34.41	366.77	381.75	s
Thrust coefficient	-2.9033	0.1852	1.9741	2.0547	

Estimated delivered performance (O/F=3.320)

Reaction efficiency: 0.9711

Nozzle efficiency: 0.9777

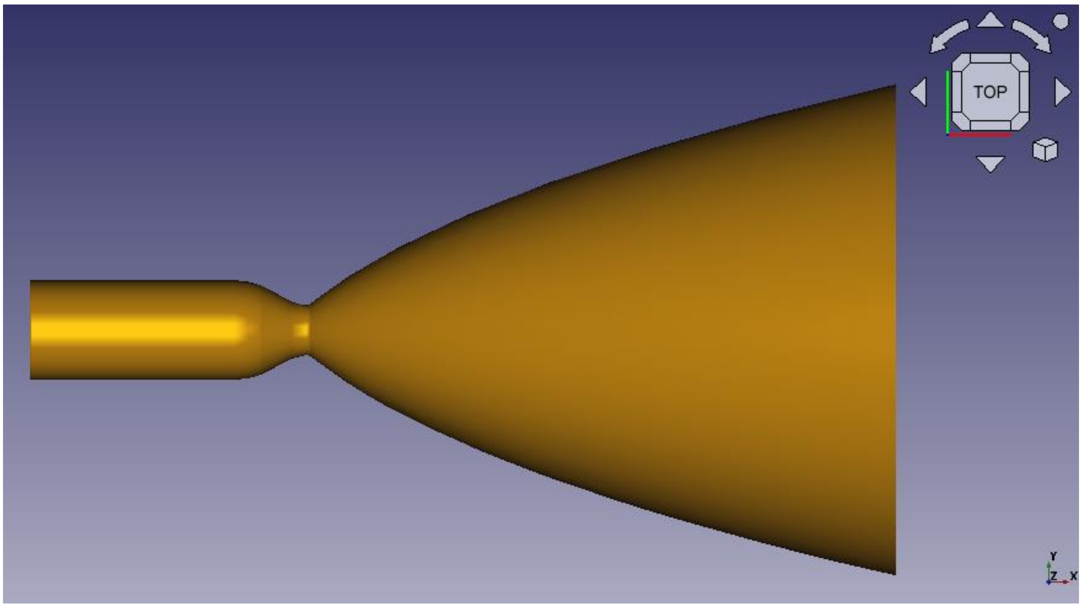
Overall efficiency: 0.9495

Parameter	Sea level	Sea level (flow sep.)	Optimum expansion	Vacuum	Unit
Characteristic velocity			1769.44		m/s
Effective exhaust velocity	-5479.12	168.74	3407.68	3554.59	m/s
Specific impulse (by mass)	-5479.12	168.74	3407.68	3554.59	N·s/kg
Specific impulse (by weight)	-558.71	17.21	347.49	362.47	s
Thrust coefficient	-3.0965	0.0954	1.9258	2.0089	

Ambient condition for optimum expansion: H=27.87 km, p=0.016 atm

Table 13-4 performance parameters by RPA tool

13.9.8 Thrust chamber Design using FreeCAD



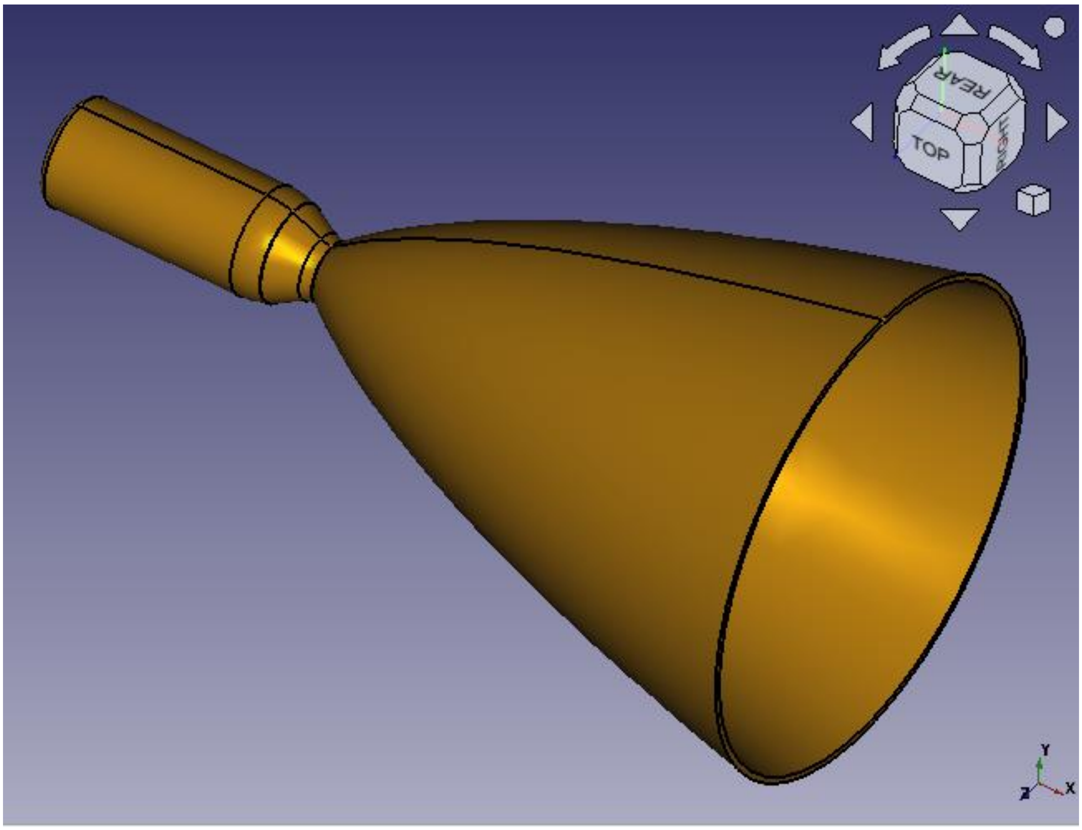


Figure 13.19 *Thrust Chamber Design using FreeCAD*

13.10 Fuel Requirements for Orbit Change

This chapter outlines the calculations required to determine the propellant mass and volumes for a spacecraft transferring from Low Earth Orbit (LEO) to Geostationary Earth Orbit (GEO). The analysis leverages the Tsiolkovsky rocket equation, using key parameters such as the required velocity change (Δv), specific impulse (Isp), and spacecraft masses. The chapter also includes calculations of the oxidizer and fuel mass proportions based on the mixture ratio and their respective volumes using density values. Finally, the burn time (tb) is computed to evaluate the operational feasibility of the LOX/LCH_4 thruster engine for this mission

13.10.1 Project Requirements

- Δv (velocity change required): 4000 m/s
- Payload mass: 50 kg

13.10.2 Key Parameters

- Specific Impulse (Isp): 366.69 seconds (according to design in chapter 3)
- Dry mass (structure + tanks + engine): 60 kg (estimation based on design in chapter 3)
- Total mass without propellant (dry mass + payload mass): 110 kg
- Oxidizer/Fuel Mixture Ratio: MR = 3.32 (mass ratio of LOX to LCH4)

- Densities:
  - Liquid Oxygen (LOX): $\rho_{LO_2} = 1.141\ kg/\ell$
  - Liquid Methane (LCH4): $\rho_{LCH_4} = 0.422\ kg/\ell$

13.10.3 Tsiolkovsky Equation

The rocket equation, also known as the **Tsiolkovsky rocket equation**, is used to relate the change in velocity (Δv) to the ratio of the initial mass (m_i) and final mass (m_f) of the spacecraft. The equation is as follows:

$$\Delta v = I_{sp} \times g_0 \times ln\left(\frac{m_i}{m_f}\right) \tag{13-25}$$

Where:

g_0 = 9.81 m/s² (standard gravitational acceleration)

m_i= initial mass (spacecraft mass including fuel)

m_f = final mass (spacecraft mass without fuel, i.e., dry mass + payload mass)

13.10.4 Propellant mass Calculations

Rearranging the rocket equation to solve for the initial mass m_i:

$$m_i = m_f \times e^{\frac{\Delta v}{I_{sp} \times g_0}} \tag{13-26}$$

Substituting the known values:

$$m_i = 110 \times e^{\frac{4000}{366.69 \times 9.81}} \approx 334.44\ kg \tag{13-27}$$

The total propellant mass is the difference between the initial mass and the final mass:

$$Propellant\ Mass = M_P = m_i - m_f = 334.44\ kg - 110\ kg = 224.44\ kg \tag{13-28}$$

Since the oxidizer-to-fuel mixture ratio is 3.32, the masses of liquid oxygen (LOX) and liquid methane (LCH4) can be calculated as follows:

$$Oxidizer\ Mass = \frac{M_P \times MR}{1 + MR} \tag{13-29}$$

$$\text{Oxidizer Mass} = M_{LO_2} = \frac{224.44 \times 3.32}{1 + 3.32} = 172.48\ \text{kg}$$

$$Fuel\ Mass = \frac{M_P}{1 + MR} \tag{13-30}$$

$$\text{Fuel Mass} = M_{LCH_4} = \frac{224.44}{1 + 3.32} = 51.95\ \text{kg}$$

13.10.5 Propellant Volume Calculations

To convert the propellant masses to volume, we use their respective densities:

- **Liquid Oxygen (LOX) Volume:**

$$V_{LO_2} = \frac{M_{LO_2}}{\rho_{LO_2}} = \frac{172.48\ kg}{1.141\ kg/\ell} \approx 151.17\ell \tag{13-31}$$

- **Liquid Methane (LCH4) Volume:**

$$V_{LCH_4} = \frac{M_{LCH_4}}{\rho_{LCH_4}} = \frac{51.95\ kg}{0.422\ kg/\ell} \approx 123.11\ \ell \tag{13-32}$$

13.10.6 Burn Time

The burn time tb is a critical parameter in evaluating the feasibility of the orbit transfer mission that connects mission requirements and engine performance. It reflects the duration the thruster must operate to achieve the required Δv for the transfer from LEO-to-GEO while ensuring efficient use of propellants.

The burn time is calculated using the relation:

$$t_b = \frac{M_P}{\dot{m}} = \frac{224.44\ kg}{1.39\ kg/s} = 161.47\ s = 2\ minutes\ 41.47s \quad (13\text{-}33)$$

13.10.6.1 Mission-Level Analysis

From a mission perspective:

- The burn time aligns with the Δv requirement, confirming that the engine can sustain operation for the duration needed to complete the orbit transfer maneuver.
- The calculated tb ensures sufficient propellant is available, eliminating the risk of fuel depletion.
- This duration allows precise and controlled execution of the orbit transfer, meeting the mission's overall objectives.

13.10.6.2 Engine Performance Analysis

From an engine performance perspective:

- The burn time validates the thruster's ability to maintain the required thrust and mass flow over the specified duration.
- Combustion stability is confirmed by the propellant stay time (ts) (1.09 ms), which is significantly shorter than tb. This indicates that the combustion chamber design ensures complete mixing and combustion within the chamber before expulsion through the nozzle.
- The chamber pressure of 300 psia and the designed mass flow rate align with the calculated burn time, verifying the engine's efficiency in utilizing the propellants.

13.10.7 Conclusion

The calculations presented in this chapter validate the feasibility of the LEO-to-GEO orbit transfer using the LOX/LCH_4 thruster engine. The propellant mass of 224.44 kg meets the mission's Δv requirement, with oxidizer and fuel volumes well within practical storage limits. A burn time of approximately 161.47 seconds confirms that the engine can sustain the required operation to execute the maneuver, while maintaining efficiency and stability in combustion. These results demonstrate the engine's capability to fulfill mission objectives and provide a solid foundation for subsequent design and operational considerations.

13.11 Thermal Analysis of LOX/LCH4 Thrust Chamber with Regenerative Cooling

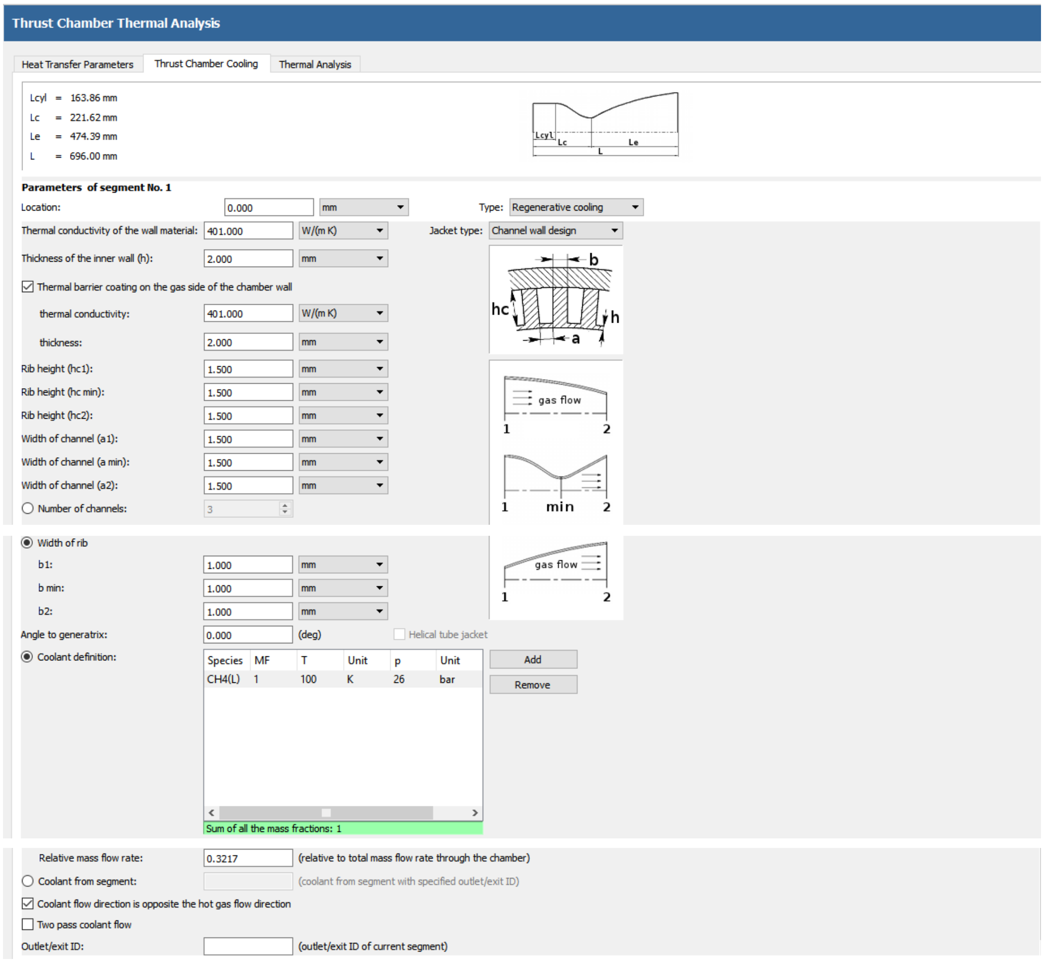
This section details the thermal analysis of the LOX/LCH4 thruster engine's thrust chamber, focusing on the implementation of a regenerative cooling system. The design was analyzed using RPA software, incorporating various critical parameters such as wall material, coolant properties, and channel geometry to evaluate the thermal performance and ensure the structural integrity of the thrust chamber under operational conditions.

13.11.1 Methodology and Inputs

Regenerative cooling is the most widely used method of cooling a thrust chamber and is accomplished by flowing high-velocity coolant over the back side of the chamber hot gas wall to convectively cool the hot liner. The coolant with the heat input from cooling the liner is then usually discharged into the injector and utilized as a propellant.

The regenerative cooling system employs liquid methane (LCH_4) as the coolant due to its favorable thermodynamic properties.

Configuration of the regenerative cooling segment includes the following parameters:



The regenerative cooling approach ensures efficient heat transfer between the chamber wall and the coolant, leveraging the high thermal conductivity of copper to maintain structural and thermal integrity.

The selection of initial pressure and temperature for the cooling system is based on operational requirements and thermodynamic considerations:

- **Initial Pressure (26 bar):**

The combustion chamber pressure is set at 300 psia (approximately 20.68 bar). To ensure sufficient pressure for liquid methane at the injector inlet, the pressure of the coolant at the cooling pipe outlet is set at 22 bar. Assuming a pressure drop of 2 bar in the injector, this configuration ensures proper flow into the combustion chamber.

A pressure drop of approximately 4 bar is expected across the cooling pipes. Consequently, the initial pressure of liquid methane at the cooling pipe inlet is set at 26 bar.

Since the methane is stored at atmospheric pressure (1 bar) in the fuel tank, it must be pressurized to 26 bar using helium before entering the cooling pipes.

- **Initial Temperature (100 K):**

Using the methane phase diagram (Figure 5.1), the boiling point of methane at 26 bar is approximately 173 K, and the freezing point is approximately 91 K. To avoid phase change within the cooling pipes, the initial temperature of the liquid methane at the inlet is chosen to be higher than the freezing point. The selected temperature of 100 K ensures that the coolant remains in liquid form throughout the cooling process.

As the coolant absorbs heat in the cooling pipes, its temperature will increase to above the boiling point (173 K) before entering the combustion chamber, ensuring efficient heat absorption and proper injection as a gaseous propellant.

This careful selection of pressure and temperature parameters ensures the stability, efficiency, and reliability of the regenerative cooling system under the specified operational conditions.

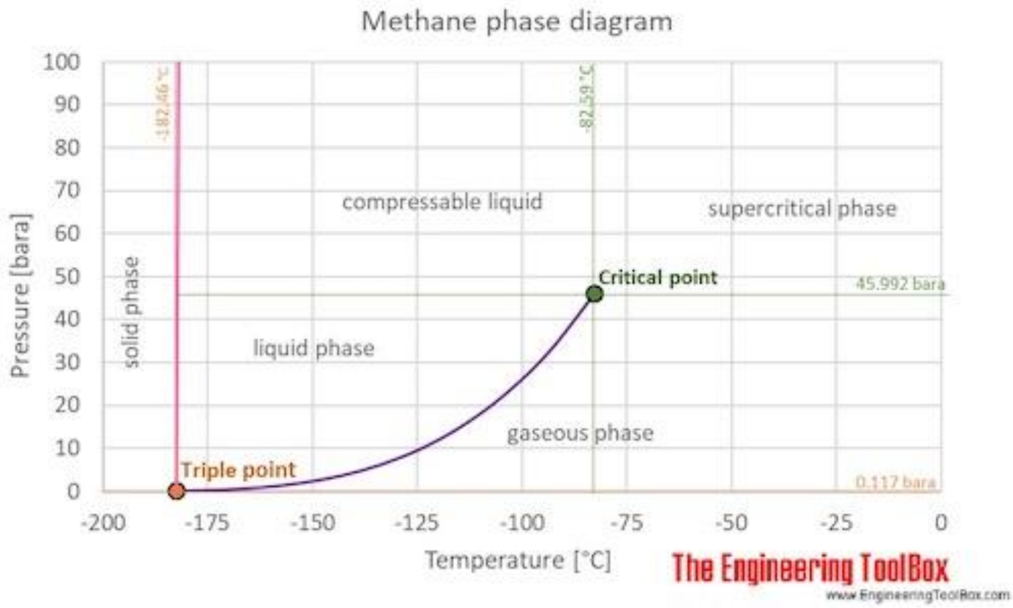


***Figure 13.20** Methane phase diagram*

13.11.2 Results and Observations

The results of the RPA thermal analysis include critical thermal and flow parameters distributed along the thrust chamber. Key metrics are summarized in Table 5-1, providing insights into the chamber's thermal performance and the efficiency of the regenerative cooling system.

13.11.2.1 Key Parameters

- Location (mm): The axial distance along the cooling channel.
- Radius (mm): The radial position of the point of analysis.
- Convective Heat Transfer Coefficient (W/m^2-K): Indicates the efficiency of heat transfer between the coolant and the wall.
- Convective Heat Flux (kW/m^2): Heat flux attributed to convection from the gas-side wall to the coolant.
- Radiative Heat Flux (kW/m^2): Heat flux due to radiation from the combustion gases to the chamber wall.
- Total Heat Flux (kW/m^2): Combined heat flux from convection and radiation.
- T_{wg}, (K): the temperature of chamber wall on its hot gas side.
- Twi, (K): the temperature between the thermal barrier coating layer and chamber wall (if coating is available).
- T_{wc}, (K): the temperature of chamber wall on its cooler side.
- p_c (MPa), T_c, (K), w_c (m/s), ϱ (kg/m^3): the pressure, temperature, velocity, and density of the coolant correspondingly (if applicable).

Thrust Chamber Thermal Analysis

Heat Transfer Parameters | Thrust Chamber Cooling | Thermal Analysis

Results of steady-state thermal analysis at chamber throttle level R=1.000

Location, mm	Radius	Conv. heat coeff., kW/m^2-K	Conv. heat flux, kW/m^2	Rad. heat flux	Total heat flux	Twg, K	Twi, K	Twc, K	Tc, K	pc, MPa	wc, m/s	rho, kg/m^3	Comments
0.00	39.39	2.2204	5419.0424	369.5441	5788.5865	1101.65	1072.79	1043.92	598.46	2.296	199.62	7.33	regenerative cooling, opposite flow
10.24	39.39	2.2177	5397.4519	501.8724	5899.3243	1108.39	1078.96	1049.54	588.46	2.309	194.87	7.51	regenerative cooling, opposite flow
20.48	39.39	2.2157	5381.6192	604.4595	5986.0787	1113.37	1083.52	1053.66	578.17	2.321	190.23	7.70	regenerative cooling, opposite flow
30.72	39.39	2.2144	5371.1883	677.9497	6049.1379	1116.63	1086.46	1056.29	567.63	2.333	185.72	7.88	regenerative cooling, opposite flow
40.96	39.39	2.2138	5366.4461	722.1986	6088.6447	1118.13	1087.76	1057.39	556.88	2.345	181.33	8.07	regenerative cooling, opposite flow
51.21	39.39	2.2140	5367.8666	736.5216	6104.3882	1117.66	1087.18	1056.74	545.94	2.356	176.81	8.28	regenerative cooling, opposite flow
61.45	39.39	2.2146	5372.4771	736.7734	6109.2505	1116.22	1085.74	1055.27	534.84	2.367	172.01	8.51	regenerative cooling, opposite flow
71.69	39.39	2.2150	5375.7877	736.7342	6112.5219	1115.14	1084.61	1054.12	523.59	2.378	167.39	8.75	regenerative cooling, opposite flow
81.93	39.39	2.2152	5377.9539	737.1293	6115.0832	1114.46	1083.92	1053.42	512.18	2.388	162.95	8.99	regenerative cooling, opposite flow
92.17	39.39	2.2154	5378.8946	737.2670	6116.1616	1114.19	1083.66	1053.16	500.61	2.399	158.68	9.23	regenerative cooling, opposite flow
102.41	39.39	2.2155	5380.1238	737.4304	6117.5542	1113.80	1083.26	1052.74	488.87	2.408	153.76	9.52	regenerative cooling, opposite flow
112.65	39.39	2.2155	5380.0498	737.5160	6117.5658	1113.83	1083.30	1052.79	476.97	2.418	149.02	9.83	regenerative cooling, opposite flow
122.89	39.39	2.2152	5378.0220	737.0217	6115.0437	1114.40	1083.82	1053.32	464.89	2.427	144.50	10.13	regenerative cooling, opposite flow
133.13	39.39	2.2148	5374.6115	736.8000	6111.4115	1115.54	1085.03	1054.55	452.63	2.436	140.17	10.45	regenerative cooling, opposite flow
143.37	39.39	2.2144	5371.5847	736.3127	6107.8974	1116.45	1085.92	1055.46	440.19	2.444	135.20	10.83	regenerative cooling, opposite flow
153.62	39.39	2.2139	5367.3373	737.3523	6104.6897	1117.84	1087.38	1056.94	427.58	2.452	130.28	11.24	regenerative cooling, opposite flow
163.86	39.39	2.2130	5360.7789	736.1863	6096.9652	1119.83	1089.35	1058.94	414.78	2.460	125.64	11.65	regenerative cooling, opposite flow
176.42	38.03	2.3594	5685.3289	737.0024	6422.3313	1132.49	1100.44	1068.41	398.58	2.470	125.23	12.20	regenerative cooling, opposite flow
188.98	33.74	2.9393	6981.5572	723.6018	7705.1590	1166.55	1128.00	1089.57	380.43	2.482	136.00	13.00	regenerative cooling, opposite flow
201.55	26.71	4.5075	10393.4782	686.1074	11079.5855	1234.64	1179.33	1124.07	358.26	2.499	170.05	14.03	regenerative cooling, opposite flow
214.11	20.66	7.1421	15996.4388	509.8045	16506.2433	1294.21	1211.84	1129.52	333.32	2.529	218.43	15.62	regenerative cooling, opposite flow
221.62	19.69	7.6722	17115.1256	479.9472	17595.0728	1294.96	1207.18	1119.43	318.49	2.554	218.01	16.81	regenerative cooling, opposite flow
246.74	36.84	2.1600	5380.4762	86.7848	5467.2610	981.34	954.11	926.85	269.79	2.584	72.01	22.05	regenerative cooling, opposite flow
271.87	53.71	1.0235	2721.7401	51.4268	2773.1669	795.81	782.29	768.46	239.07	2.593	37.12	27.45	regenerative cooling, opposite flow
271.87	53.71	1.0235	2721.7401	51.4268	2773.1669	795.81	782.29	768.46	239.07	2.593	37.12	27.45	regenerative cooling, opposite flow
297.00	68.69	0.6275	1755.6752	36.2274	1791.9026	647.83	639.77	630.83	219.62	2.596	21.39	36.15	regenerative cooling, opposite flow
322.12	82.17	0.4405	1280.0616	26.9494	1307.0109	533.14	526.43	519.91	207.33	2.597	12.81	49.60	regenerative cooling, opposite flow
347.25	94.42	0.3318	976.4498	21.2432	997.6930	491.11	485.56	480.58	197.67	2.598	8.04	67.97	regenerative cooling, opposite flow
372.37	105.62	0.2648	782.4703	17.2756	799.7459	475.14	470.86	466.87	188.24	2.599	5.89	82.33	regenerative cooling, opposite flow
397.50	115.93	0.2188	651.7133	14.5079	666.2212	448.48	444.95	441.62	178.37	2.599	4.80	91.45	regenerative cooling, opposite flow
422.62	125.47	0.1872	561.8975	12.4208	574.3183	422.72	419.66	416.80	168.92	2.599	4.11	98.38	regenerative cooling, opposite flow
447.75	134.33	0.1639	495.3010	10.8938	506.1948	399.27	396.56	394.03	160.08	2.599	3.61	104.15	regenerative cooling, opposite flow
472.87	142.59	0.1461	444.5900	9.6953	454.2853	376.99	374.63	372.36	151.94	2.599	3.23	109.23	regenerative cooling, opposite flow
498.00	150.30	0.1333	411.1724	8.7764	419.9488	333.21	331.15	329.06	144.74	2.600	1.29	258.40	regenerative cooling, opposite flow
523.13	157.53	0.1224	380.7931	8.0227	388.8157	304.96	302.45	300.51	138.45	2.600	0.83	382.57	regenerative cooling, opposite flow
548.25	164.31	0.1127	351.6565	7.3625	359.0190	295.85	293.47	291.68	132.55	2.600	0.78	392.54	regenerative cooling, opposite flow
573.38	170.68	0.1047	327.3311	6.8147	334.1458	287.77	285.49	283.83	126.77	2.600	0.73	401.88	regenerative cooling, opposite flow
598.50	176.69	0.0979	306.7629	6.3980	313.1610	280.33	278.15	276.59	121.09	2.600	0.69	410.70	regenerative cooling, opposite flow
623.63	182.34	0.0921	289.0394	6.0023	295.0418	274.17	272.06	270.59	115.52	2.600	0.65	418.91	regenerative cooling, opposite flow
648.75	187.68	0.0871	273.8114	5.6067	279.4181	268.19	266.13	264.74	110.05	2.600	0.62	426.79	regenerative cooling, opposite flow
696.00	196.94	0.0793	249.8364	5.1104	254.9468	260.37	258.44	257.16	100.00	2.600	0.57	440.78	regenerative cooling, opposite flow

Table 13-5 *Thermal Analysis Results from RPA for Regenerative Cooling with Direct Flow*

13.11.2.2 Observations and Design Implications

The results presented in Table 5-1 highlight several significant trends and design implications for the engine's thermal management:

Interpretation of Heat Transfer Coefficient and Heat Flux

In thruster engine thermal analysis, the heat transfer coefficient (h) and heat flux (q) are critical parameters that characterize the cooling and heat dissipation performance:

- **Heat Transfer Coefficient (h):**

Represents the efficiency of heat exchange between the hot gases inside the combustion chamber and the chamber wall.

Higher values indicate more effective heat transfer, which may require robust cooling mechanisms to prevent material failure.

In the presented results in table 5-1, h remains relatively constant along the regenerative cooling flow path, with slight variations indicating the influence of flow properties and chamber geometry.

- **Convective Heat Flux (q_{conv}):**

Quantifies the heat transferred per unit area from the hot gases to the wall via convection.

Calculated as:

$$q_{Conv} = h \,.(T_{gas} - T_{wall}) \qquad (13\text{-}34)$$

where T_{gas} is the combustion gas temperature, and T_{wall} is the inner wall temperature.

High q_{conv} values near the throat region highlight the need for enhanced cooling as gas velocity and temperature peak in this area.

- **Radiative Heat Flux (q_{rad}):**

Results from thermal radiation emitted by the hot gases, described by:

$$q_{rad} = \varepsilon . \sigma . T_{gas}^4 \tag{13-35}$$

where ϵ is the emissivity, and σ is the Stefan-Boltzmann constant.

Comparatively smaller than q_{conv} but still contributes to the overall heat load.

- **Total Heat Flux (q_{total}):**

Summation of convective and radiative components:

$$q_{total}=q_{conv}+q_{rad} \tag{13-36}$$

Represents the overall thermal stress on the cooling system and chamber walls. Regions with high q_{total} require sufficient cooling capacity to maintain structural integrity.

Observations

The heat transfer coefficient decreases slightly along the flow direction, corresponding to reduced gas temperatures and changing flow characteristics. (Figure 5.2)

Convective heat flux values are highest near the throat (17,115 kW/m²) due to the combination of high temperatures and gas velocities.

Radiative heat flux becomes significant where gas temperatures remain elevated, particularly in the combustion zone.

The total heat flux peaks in the throat area and gradually diminishes downstream, corresponding to decreasing temperatures and expanding flow in the nozzle.

These results underline the importance of an optimized regenerative cooling design, especially near the throat and combustion chamber regions, where thermal loads are most severe.

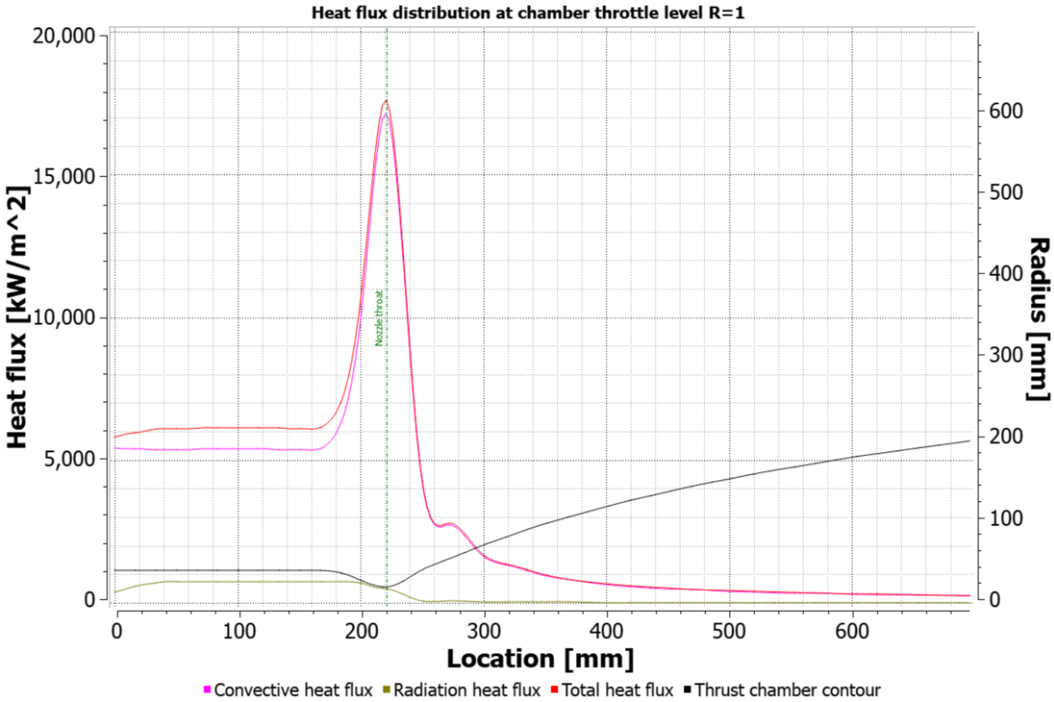


Figure 13.21 RPA-Generated Variation of Convective, Radiation, and Total Heat Flux with Axial Location and Radius, Emphasizing the Significant Flux at the Throat

Coolant Temperature

In the coolant system design for the thruster, it is observed that the coolant reaches its boiling point before entering the combustion chamber. As shown in Figure 5-3 and detailed in Table 5-1, the coolant temperature (Tc) increases within the coolant pipes, ultimately achieving its boiling point. This phenomenon is particularly advantageous as it ensures that the coolant undergoes a phase change to gas prior to entering the combustion chamber. The transition to gas enhances heat transfer efficiency, as the latent heat of vaporization facilitates effective cooling. This thermal behavior is beneficial for maintaining optimal operating conditions for the thruster and preventing overheating of critical components.

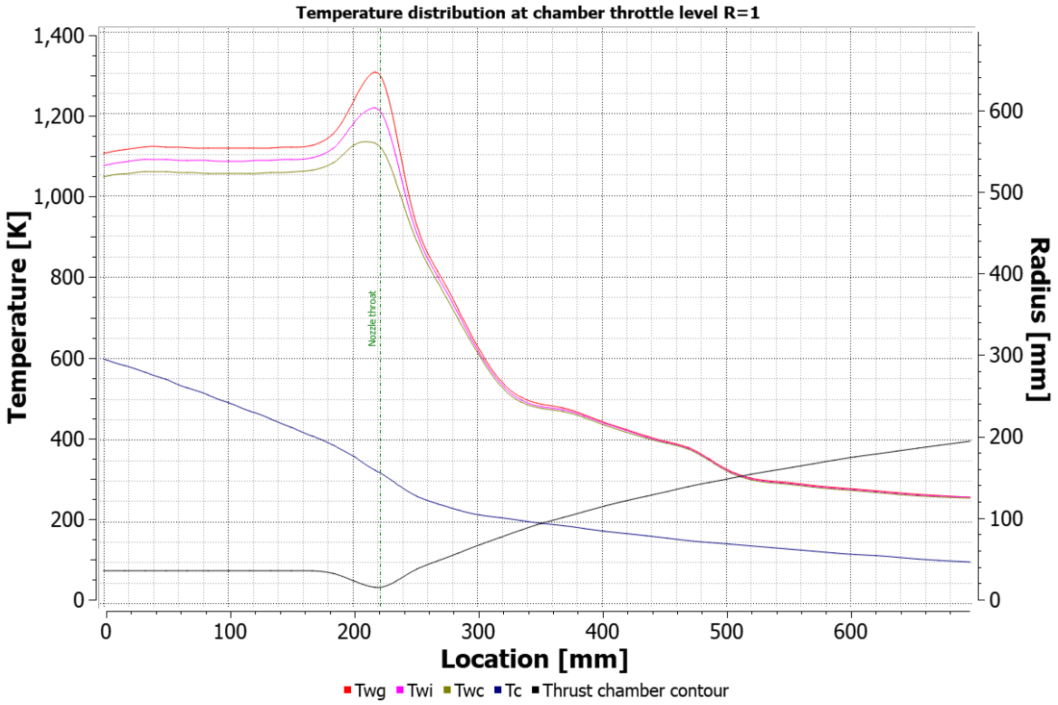


Figure 13.22 *RPA-Generated Variation of T_{wg}, T_{wi}, T_{wc} and T_C with Radius and Axial Location in the Thrust Chamber*

Wall Temperatures:

- The gas-side wall temperature (T_{wg}) is highest near the throat, reaching a peak of approximately 1,294 K. This observation reflects the intense thermal loading at this location due to high flow velocities and reduced cross-sectional area.
- The coolant-side wall temperature (T_{wc}) is significantly lower than gas-side wall temperature, reflecting the effectiveness of the cooling mechanism. Figure 5-3 depicts the temperature distribution across the radius and length, highlighting the critical thermal gradient at the throat region.

Structural Integrity and Material Selection

The structural integrity of the wall materials must accommodate elevated temperatures observed in the combustion chamber and nozzle. This necessitates the selection of materials with high melting points, excellent thermal conductivity, and robust mechanical properties under extreme conditions. In this context, copper and its alloys are commonly preferred due to their favorable properties, including high thermal conductivity and adequate melting points.

Melting Point of Copper at Elevated Pressure

The melting point of copper (Cu) increases slightly under elevated pressures, such as the 300 psia (approximately 20.7 bar) operational conditions in the combustion chamber. This behavior can be analyzed using the Clausius-Clapeyron relation, which estimates the shift in melting point with pressure:

$$\frac{dT}{dP} = \frac{T_m \, . \Delta v}{\Delta H_f} \quad (13\text{-}37)$$

Where:

T_m: Melting temperature in kelvin at standard pressure (1357.77 K).

Δv: Change in specific volume during melting ($v_{liquid} - v_{solid} = 1.31 \times 10^{-5} m^3/kg$).

ΔH_f: Latent heat of fusion (13.05kJ/mol, equivalent to 205 kJ/kg for copper).

dT/dP: Rate of change of melting temperature with pressure.

- **Standard Melting Point:**

At 1 atm pressure, the melting point of copper is 1357.77 K (1084.62°C).

- **Effect of Pressure:**

  For the given conditions:

$$\frac{dT}{dP} = \frac{1357.77 \times 1.31 \times 10^{-5}}{205 \times 10^{3}} = 8.7 \times 10^{-8} \, K/_{Pa} \quad (13\text{-}38)$$

With a pressure increase of ΔP=19.7bar = 197x 10^4 Pa.

$\Delta T_m \approx 0.171$k

- **Resulting Melting Point:**

The new melting point under 300 psia pressure becomes:

$$T_m' = 0.171 + 1357.77 = 1357.94\, K \quad (13\text{-}39)$$

Comparison with Wall Temperatures

The calculated melting point of copper under 300 psia, approximately 1357.94 K, is significantly higher than the highest observed inner wall temperature near the throat region (1294.96 K). This confirms that copper-based materials can maintain their structural integrity under the operating conditions.

Implications for Design

Material Suitability: Copper and its alloys are suitable for the combustion chamber and nozzle due to their ability to withstand extreme thermal conditions without reaching their melting point.

Thermal Management: High thermal conductivity of copper facilitates efficient heat dissipation, reducing thermal gradients and stress across the wall.

Safety Margin: The melting point margin ensures operational reliability and durability, even in the most thermally critical regions.

In conclusion, the selection of copper or its high-temperature alloys, potentially enhanced with thermal barrier coatings, provides a robust solution to manage the thermal and structural demands of the thruster engine.

These observations underscore the importance of detailed thermal analysis in guiding the design and optimization of the thruster's regenerative cooling system. The data provides a foundation for selecting materials, optimizing cooling channel geometry, and ensuring the overall thermal reliability of the propulsion system.

13.12 Prototype and Test Stand Realization

This chapter presents the process of prototyping and test stand realization for the LOX/LCH_4 thruster engine. It includes visual documentation of the combustion chamber and nozzle manufacturing stages, the integration into the orbit change test stand, examples of commercially available components, and considerations for regenerative cooling.

13.12.1 Plastic Model of Combustion Chamber and Nozzle from 3D Printer

The initial stage involved fabricating a plastic model of the combustion chamber and nozzle using a 3D printer. This model served as the basis for creating a sand mold to be used in the copper casting process (Figures 6.1).

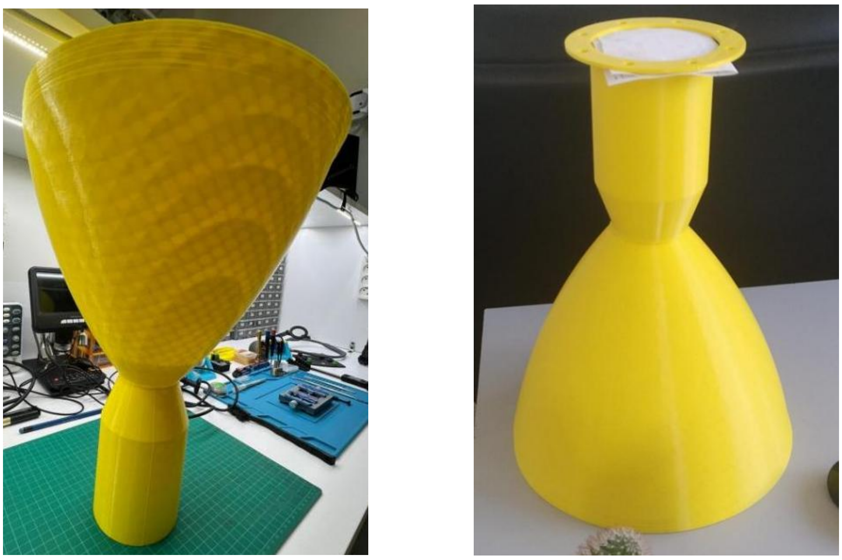


Figure 13.23 Plastic Model of the Combustion Chamber and Nozzle Fabricated at AECENAR

13.12.2 Combustion Chamber and Nozzle After Melting and Forming Copper

Following the sand mold preparation, the combustion chamber and nozzle were cast in copper. This process ensured the high thermal conductivity required for effective heat dissipation during operation.(Figures 6.2)

Figure 13.24 *Copper-Formed Combustion Chamber and Nozzle Fabricated at AECENAR*

13.12.3 Integration into Orbit Change Test Stand

The copper combustion chamber and nozzle were subsequently integrated into the orbit change test stand. This setup forms the foundation for testing the thruster engine under controlled conditions (Figure 6.3).

Figure 13.25 *Integration of the Copper Thrust Chamber into the Orbit Change Test Stand at AECENAR*

13.12.3.1 Examples of Commercially Available Components

In the context of designing and realizing the test stand, understanding commercially available hardware helps to evaluate practical implementation possibilities. One such example is an oxidizer tank designed for 45 liters of LOX, photographed in Germany (Figure 6.4).

While this tank does not meet the specific requirements of our design, it provides insight into critical parameters for component selection, including:

- Capacity: 45 liters of LOX, equivalent to approximately 90 kg.
- Wall Thickness: Approximately 10 cm, designed to insulate -193°C storage conditions.
- Cost:
    - Empty tank: 6000 EUR
    - Filling cost: 670 EUR
- Manufacturer: Linde, commonly used for medical applications.

Including real-world examples such as this strengthens our understanding of available solutions and provides a foundation for future improvements to our design.

***Figure 13.26** Oxidizer Tank Designed for 45 Liters of LOX, Photographed in Germany*

13.12.4 Regenerative Cooling

13.12.4.1 Structure of Regenerative Cooling System

The regenerative cooling system is essential for managing the intense thermal environment within the combustion chamber and nozzle. It is typically structured as a three-layer "sandwich" arrangement, each layer playing a distinct role in heat management.

- **Innermost Layer – Hot Gas Wall:**
  The innermost layer is the hot gas wall, which is directly exposed to the combustion gases. This wall absorbs the extreme heat generated by the combustion process and is

subjected to high thermal stresses. The material selected for this layer must possess high thermal conductivity to transfer heat effectively while withstanding the harsh conditions of the combustion chamber (Figure 6.2)

- **Middle Layer – Cooling Channels:** Surrounding the hot gas wall is the cooling channel layer, which is discussed in Section 6.4.2. This layer contains the channels through which the coolant (typically the propellants) flows. The coolant absorbs heat from the hot gas wall as it passes through these channels, ensuring that the combustion chamber and nozzle remain within safe temperature limits. The design of these channels is critical for efficient heat transfer and for preventing hot spots that could damage the chamber (Fgures 6.5).
- **Outermost Layer – Closeout Wall:** The outermost layer is the closeout wall, which seals the cooling system and keeps the coolant within the channels. This layer also protects the structural integrity of the engine by providing an outer casing that prevents coolant leakage and maintains pressure within the cooling system. The closeout wall is designed to withstand external stresses and environmental conditions while supporting the efficient operation of the cooling system.

13.12.4.2 Realization of pipes layout

The design and layout of the regenerative cooling pipes are critical to ensuring the structural integrity and thermal efficiency of the combustion chamber and nozzle. This section highlights the integration of cooling pipes, including the incorporation of inlets and outlets for the propellants—fuel and oxidizer—designed to pass through the cooling system. The proper arrangement of these pipes allows for optimal heat dissipation, preventing excessive thermal stress on the engine components.

Figure 13.27 *Realization of Pipes Layout for Regenerative Cooling*

Figure 13.28 *Thrust Chamber After Integration of Propellant Inlets and Outlets for Regenerative Cooling*

13.12.4.3 Outermost Layer – Closeout Wall

The closeout wall is the outermost layer of the regenerative cooling system, serving to seal the cooling channels and contain the coolant. It is designed to withstand external stresses, pressure buildup, and environmental factors, ensuring the integrity of the cooling system. Made from high-strength, thermally resistant materials, the closeout wall prevents coolant leakage and protects the system from mechanical vibrations and temperature extremes. Figures in this section illustrate the integration of the closeout wall.

Figure 13.29 *Integration of the Closeout Wall into the Regenerative Cooling System at AECENAR*

13.13 Conclusion

In conclusion, this thesis has made significant contributions to the design, analysis, and realization of a LOX/LCH_4 thruster engine for spacecraft propulsion, with a focus on the transfer from Low Earth Orbit (LEO) to Geostationary Earth Orbit (GEO). Through the use of advanced simulation tools such as NASA CEA and RPA, we optimized critical design parameters, including the stoichiometric and optimal mixture ratios, nozzle geometry, combustion chamber dimensions, and fuel requirements, ensuring the propulsion system meets mission objectives.

The successful design of the combustion chamber and nozzle, coupled with the integration of regenerative cooling, demonstrated the feasibility of the system under expected thermal and structural conditions. The comprehensive calculations for thrust, mass flow rates, and burn time, supported by a prototype test stand, underscore the practical potential for real-world implementation.

This work not only advances our understanding of LOX/LCH_4 propulsion systems but also contributes to the broader field of space propulsion technology. The results presented provide a solid foundation for further optimization and experimental validation of the design. Future research should focus on the experimental testing of the propulsion system to confirm its performance and address any challenges encountered during implementation.

Moreover, the insights gained from this study have broader implications for space exploration, where efficient propulsion systems are critical for reducing mission costs and improving the reliability of space travel. As the space industry continues to evolve, the findings of this thesis align with efforts to enhance propulsion systems, making space missions more sustainable and cost-effective.

In summary, this thesis provides a comprehensive approach to the design and optimization of LOX/LCH_4 thruster engines, contributing valuable knowledge to the field of space propulsion and laying the groundwork for future advancements in spacecraft propulsion systems.

13.14 Bibliography

[1] M. J. Casiano, J. R. Hulka und V. Yang, „Liquid-Propellant Rocket Engine Throttling: A Comprehensive Review," *Journal of Propulsion and Power,* pp. 897-923, 09/2010.

[2] D. Krejci und P. Lozano, „Space Propulsion Technology for Small Spacecraft," Chicago, IL, USA, 3/2018.

[3] D. K. Huzel und D. H. Huang, Modern Engineering for Design of Liquid-Propellant Rocket Engines, Washington DC: American Institute of Aeronautics and Astronautics, 1992-01-01, p. 467.

[4] G. P. Sutton und O. Biblarz, Rocket Propulsion Elements, 9th Hrsg., Hoboken, New Jersey: John Wiley & Sons, 2017, p. 791.

[5] D. P. Mishra, Fundamentals of rocket propulsion, Boca Raton: CRC Press, Taylor & Francis Group, CRC Press is an imprint of the Taylor & Francis Group, an informa business, 2017, p. 461.

[6] A. De Iaco Veris, Fundamental Concepts of Liquid-Propellant Rocket Engines, Cham,Switzerland: Springer International Publishing, 2021, p. 755.

[7] J. J. Pocha, An Introduction to Mission Design for Geostationary Satellites, 1st Hrsg., Dordrecht: Springer Netherlands, 1987, p. 237.

[8] D. Harrje, LIQUID PROPELLANT ROCKET . COMBUSTION INSTABILITY, Washington, D.C: National Aeronautics And Space Administration, 1972, p. 657.

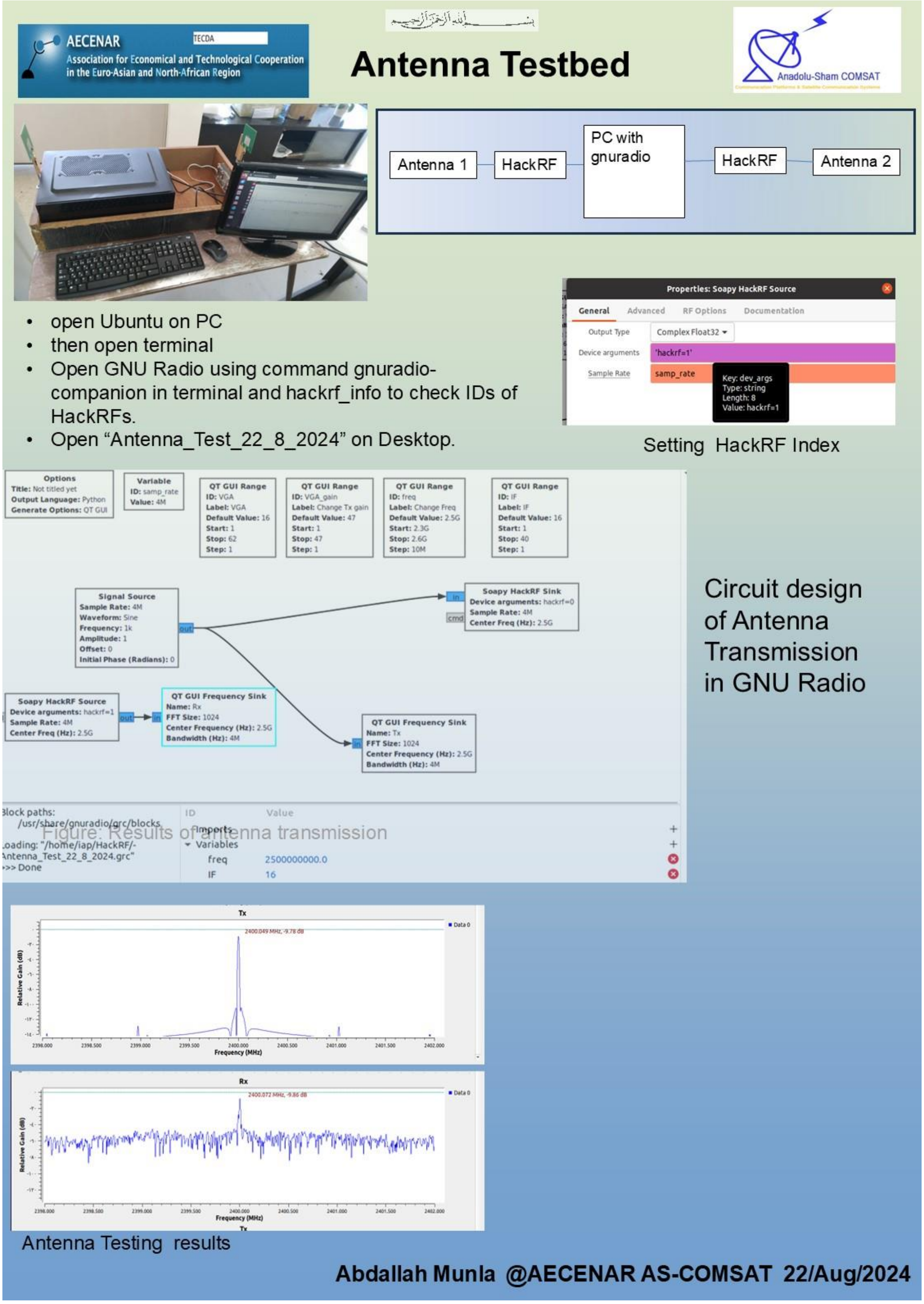
[9] E. A. Hurlbert, M. J. Atwell, J. C. Melcher und R. L. Morehead, „Integrated Pressure-Fed Liquid Oxygen / Methane Propulsion Systems - Morpheus Experience, MARE, and Future Applications," in *52nd AIAA/SAE/ASEE Joint Propulsion Conference.*, Salt Lake City, Utah, USA., 2016-07-25.

[10] J. L. Cannon, „Liquid Propulsion: Propellant Feed System Design," in *Encyclopedia of Aerospace Engineering*, L. R. Blockley und W. Shyy, Hrsg., Chichester, UK, Wiley, 2010-12-15, p. 34.

14 2.5 GHz Transceiver Development

See Satellite Development 2023, Section TT&C

15 Antenna Testbed



بسم الله الرحمن الرحيم

Antenna Testing and Sending&Receiving Testing with gnu radio and HackRF

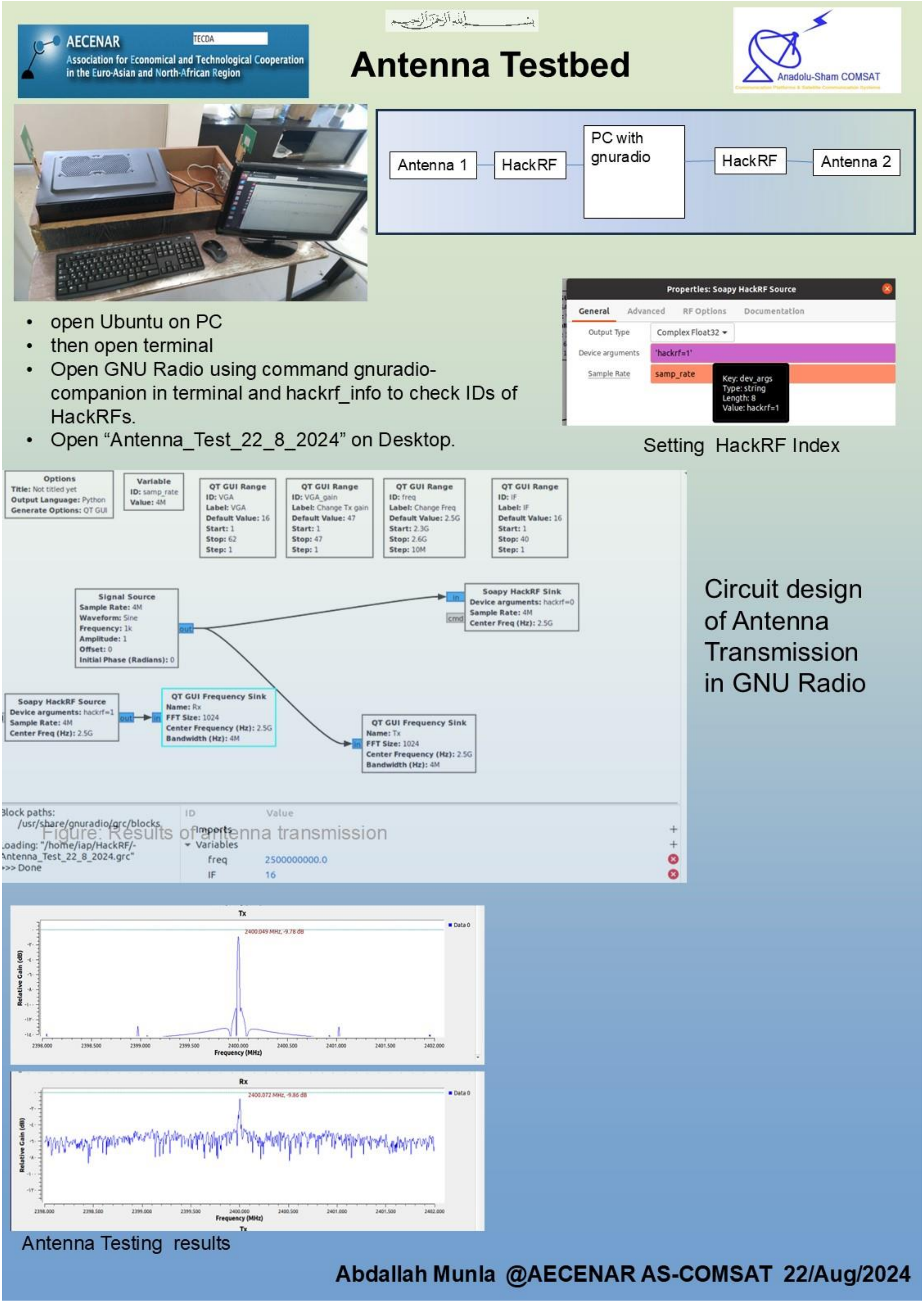
Authors:

Abdullah Munla, Ahmad Ali

Last Update: 26.08.2024 11:40

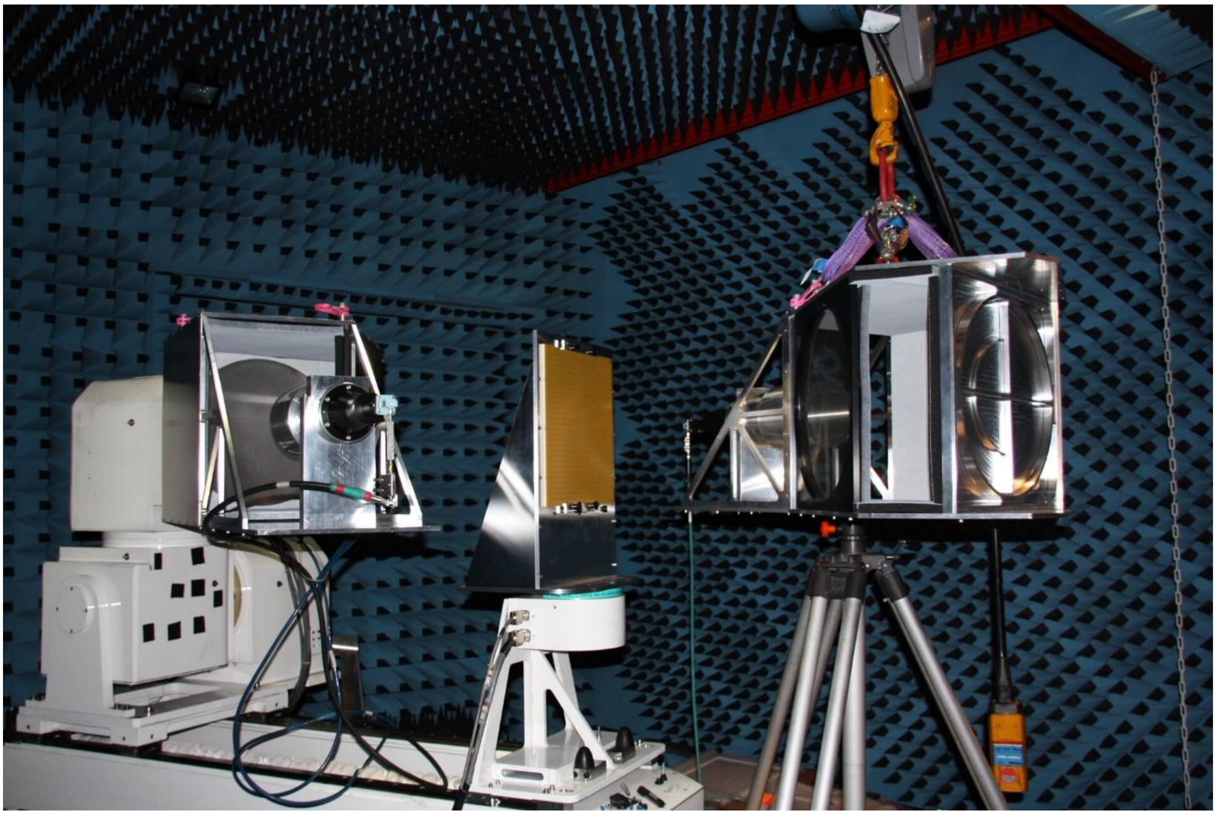
Table of Contents

15.1 Summary



AECENAR
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region
Antenna Testbed
Anadolu-Sham COMSAT
Antenna 1
HackRF
PC with gnuradio
HackRF
Antenna 2
Properties: Soapy HackRF Source
General
Advanced
RF Options
Documentation
Output Type
Complex Float32
Device arguments
'hackrf=1'
Sample Rate
samp_rate
Key: dev_args
Type: string
Length: 8
Value: hackrf=1
• open Ubuntu on PC
• then open terminal
• Open GNU Radio using command gnuradio-companion in terminal and hackrf_info to check IDs of HackRFs.
• Open "Antenna_Test_22_8_2024" on Desktop.
Setting HackRF Index
Options
Variable
QT GUI Range
Signal Source
Soapy HackRF Sink
Soapy HackRF Source
QT GUI Frequency Sink
Circuit design of Antenna Transmission in GNU Radio
Figure: Results of antenna transmission
Tx
Rx
Relative Gain (dB)
Frequency (MHz)
Antenna Testing results
Abdallah Munla @AECENAR AS-COMSAT 22/Aug/2024

15.2 Basics: Antenna testing labs



From: Antenna measurement - Wikipedia

Antenna measurement techniques refers to the testing of antennas to ensure that the antenna meets specifications or simply to characterize it. Typical parameters of antennas are gain, bandwidth, radiation pattern, beamwidth, polarization, and impedance.

The antenna pattern is the response of the antenna to a plane wave incident from a given direction or the relative power density of the wave transmitted by the antenna in a given direction. For a reciprocal antenna, these two patterns are identical. A multitude of antenna pattern measurement techniques have been developed. The first technique developed was the far-field range, where the antenna under test (AUT) is placed in the far-field of a range antenna. Due to the size required to create a far-field range for large antennas, near-field techniques were developed, which allow the measurement of the field on a distance close to the antenna (typically 3 to 10 times its wavelength). This measurement is then predicted to be the same at infinity. A third common method is the compact range, which uses a reflector to create a field near the AUT that looks approximately like a plane-wave.

15.2.1 Typical setup for characterizing the RF properties of microwave materials.

Step 1: Sample Preparation

- The material to be characterized is prepared in a suitable form, such as a thin sheet or a cylindrical rod.
- The sample dimensions and properties (thickness, dielectric constant, etc.) are recorded.

Step 2: Fixture Setup

- The appropriate fixture is selected based on the sample type and measurement frequency.
- The sample is placed accurately within the fixture, ensuring proper contact and alignment.

Step 3: Calibration

- The measurement system is calibrated using known standards to establish a reference point.
- This step corrects for system errors and ensures accurate measurements.

Step 4: Data Acquisition

- The measurement instrument (likely a network analyzer) is used to collect data about the sample's interaction with the RF signal.
- This data includes parameters like reflection coefficient, transmission coefficient, and phase information.

Step 5: Data Processing

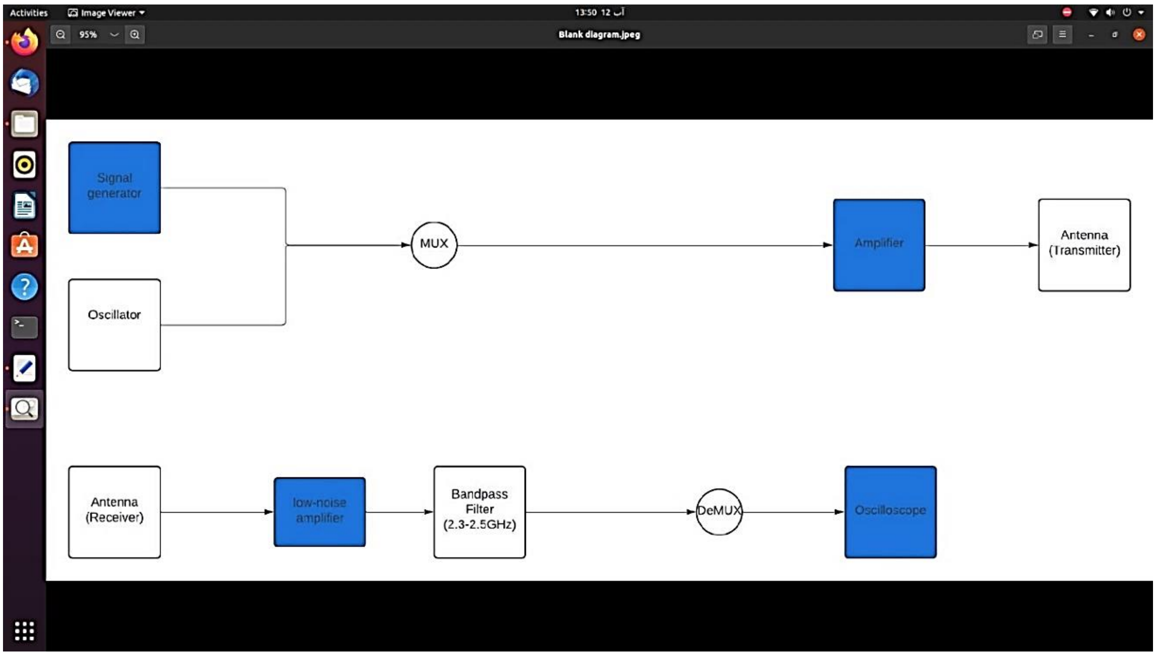
- The collected data is processed using specialized software to extract the desired material properties.
- These properties typically include complex permittivity and permeability.

Step 6: Analysis and Interpretation

- The calculated material properties are analyzed to understand the material's behavior at different frequencies.
- The results are compared to theoretical models or previous data for validation.

Note: The specific steps and equipment used can vary depending on the material, measurement frequency, and desired accuracy.

15.2.2 System Design (Block Diagram)



15.2.3 Sending Device in Testing Lab

Network Analyzer: This is the primary instrument used for data acquisition in this context. It generates a known RF signal, transmits it through the material sample, and measures the resulting wave.

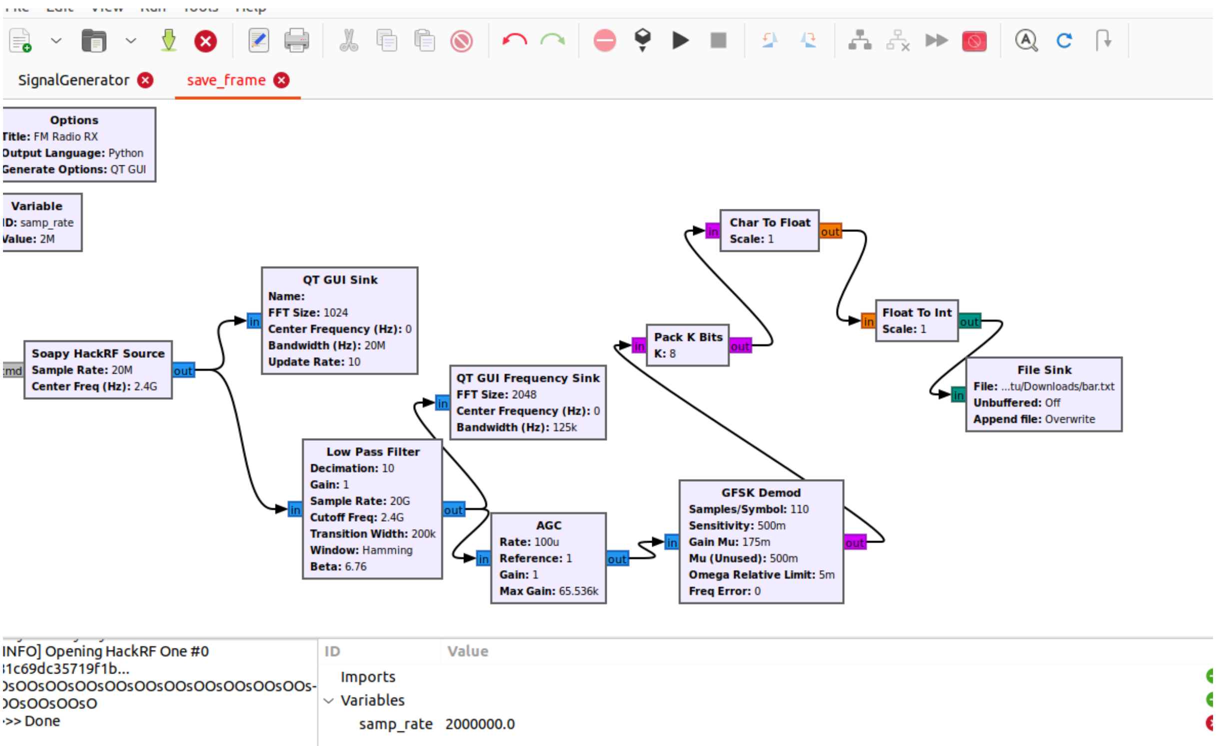
15.2.4 Receiver Device in Testing Lab

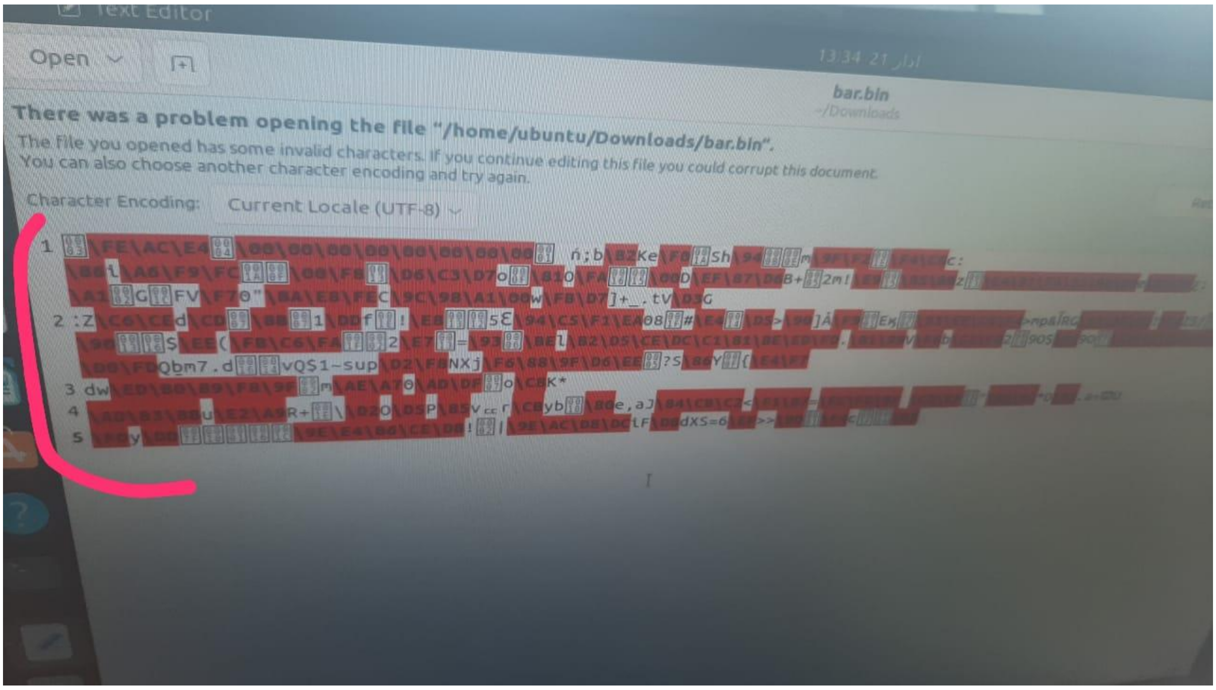
- The network analyzer sweeps through a range of frequencies, measuring the amplitude and phase of the transmitted and reflected waves at each frequency point.
- This data is typically represented in complex numbers, capturing both magnitude and phase information.
- **Measurement Techniques:** Various techniques can be employed, including:
    - **Transmission measurements:** Measuring the signal that passes through the material.
    - **Reflection measurements:** Measuring the signal reflected from the material's surface.
    - **Scattering parameters (S-parameters):** Characterizing the wave scattering properties of the material.

15.3 For receiving bits

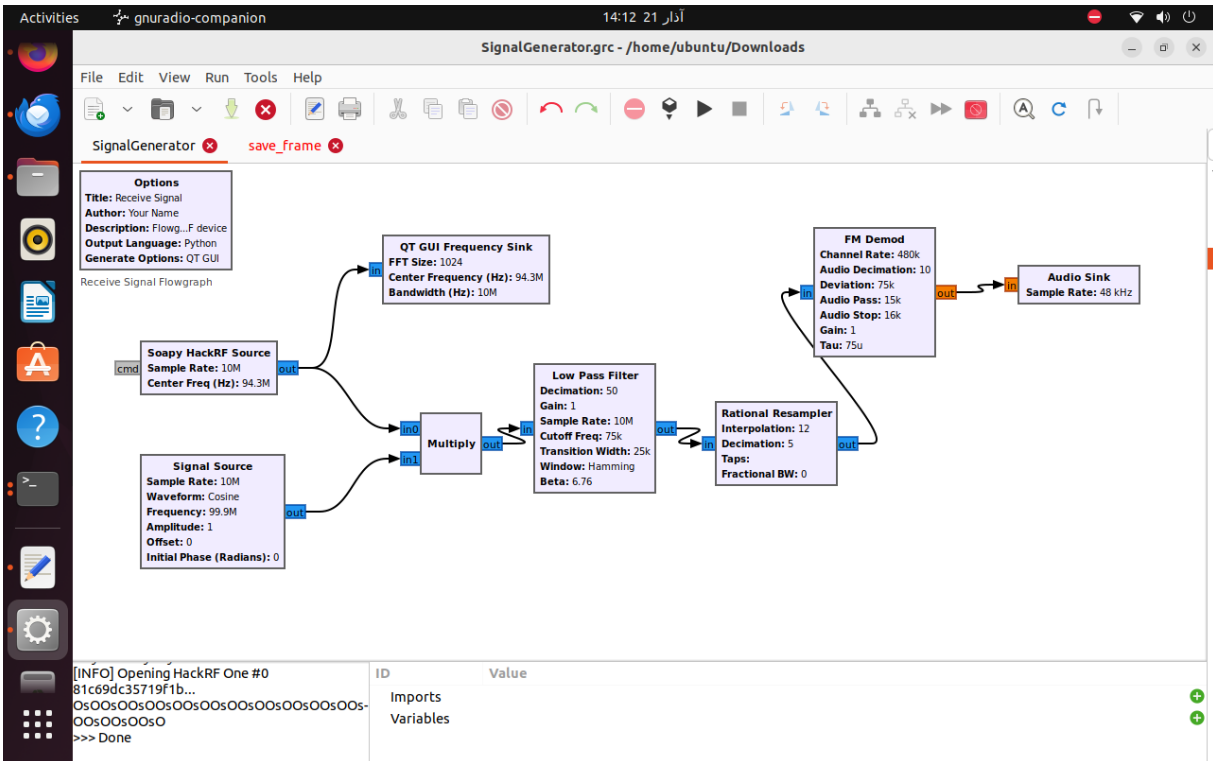
15.3.1 Test 21.3.24, 1 p.m.:

TT&C GUI: Click VALVE ON -> TT&C Transceiver -> TT&C GroundStation Antenna ----------> 360°C Antenna -> HackRF -> GnuRadio on ubuntu

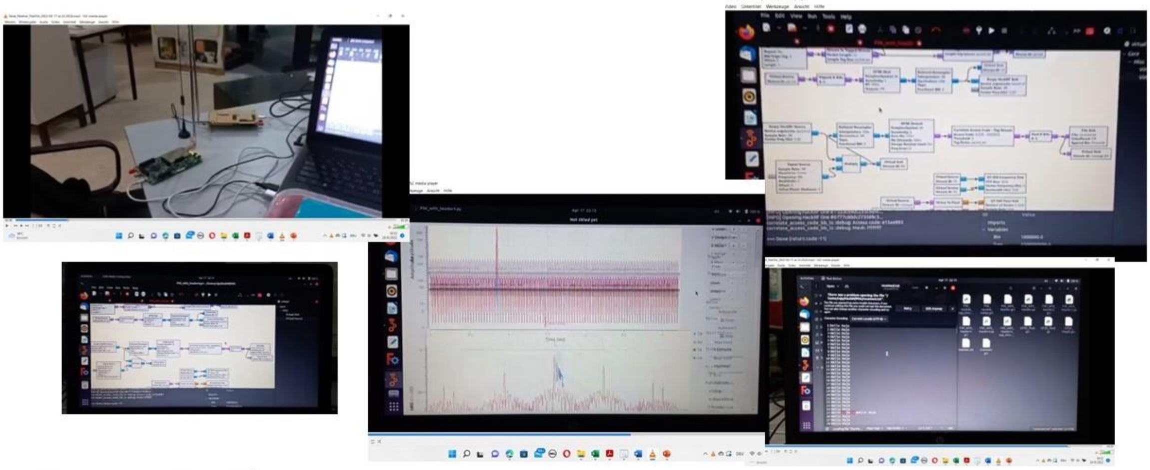




15.4 For receiving continous audio signal (radio station on 94.3 MHz):



Able to send and to receive data with HackRF and gnuradio



Demonstration Film

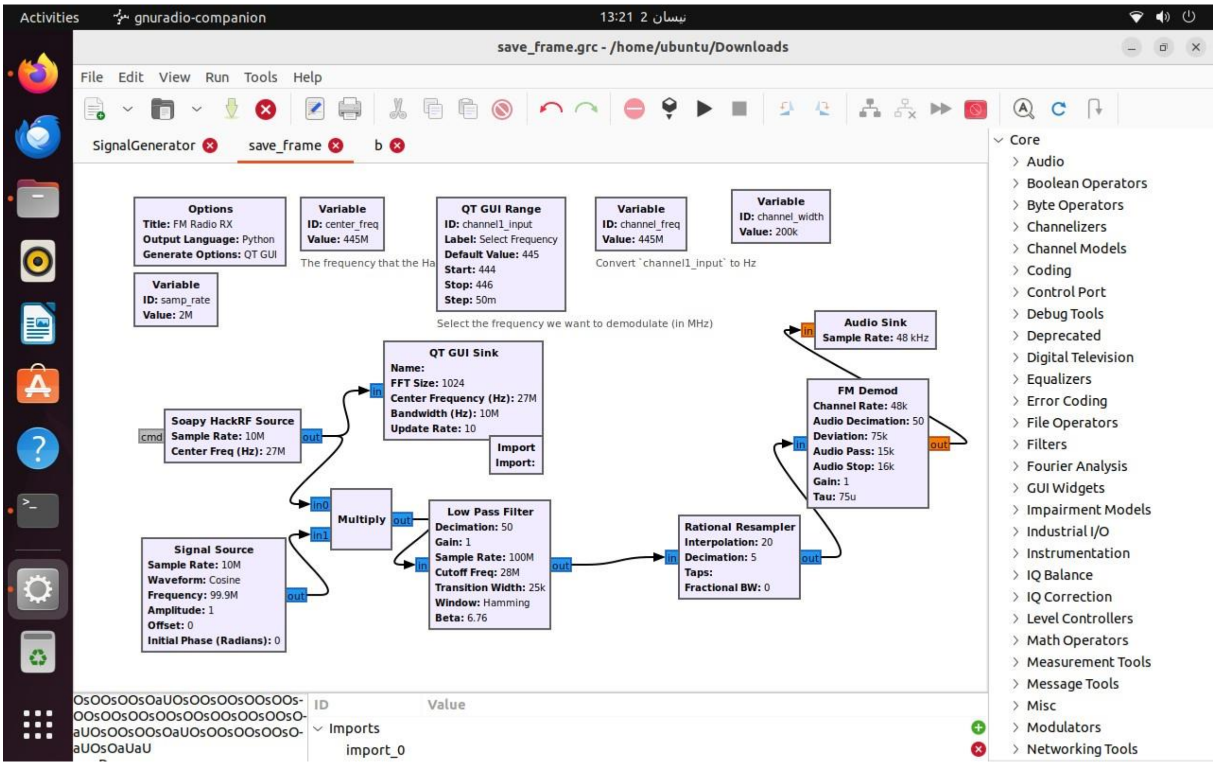
<u>Demonstration Film: Send and Receive a Text File with Gnuradio and HackRF</u>

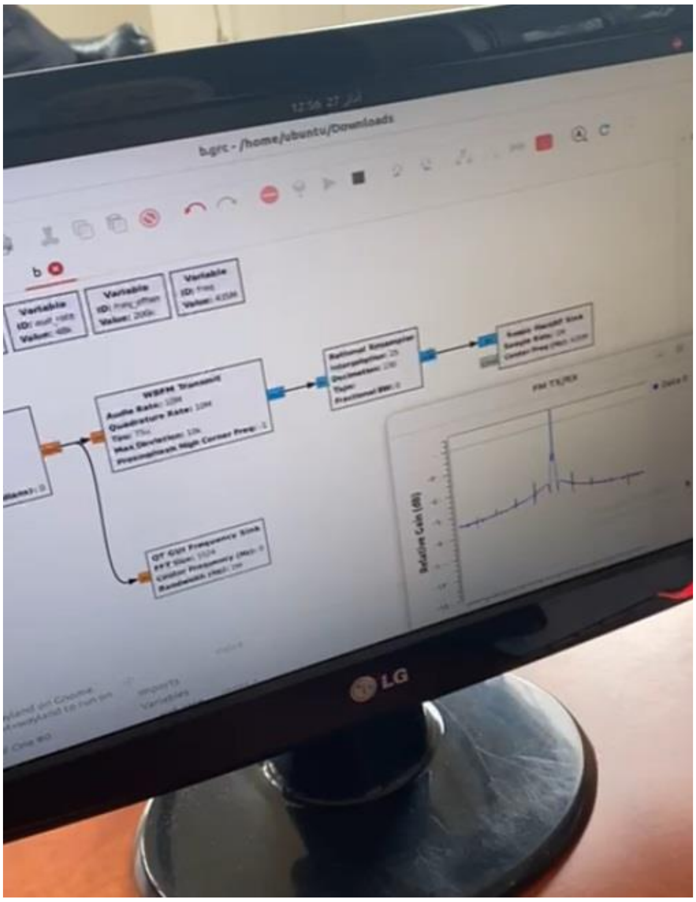
how to open gunuradio in ubuntu: 1. open terminal 2. type gnuradio-companion

15.5 2 PCS, on each is ubuntu and gnuradio

2.4.24, 1 p.m. + 22.5.24 (Halima, Abdullah Munla, Ahmad Ali)

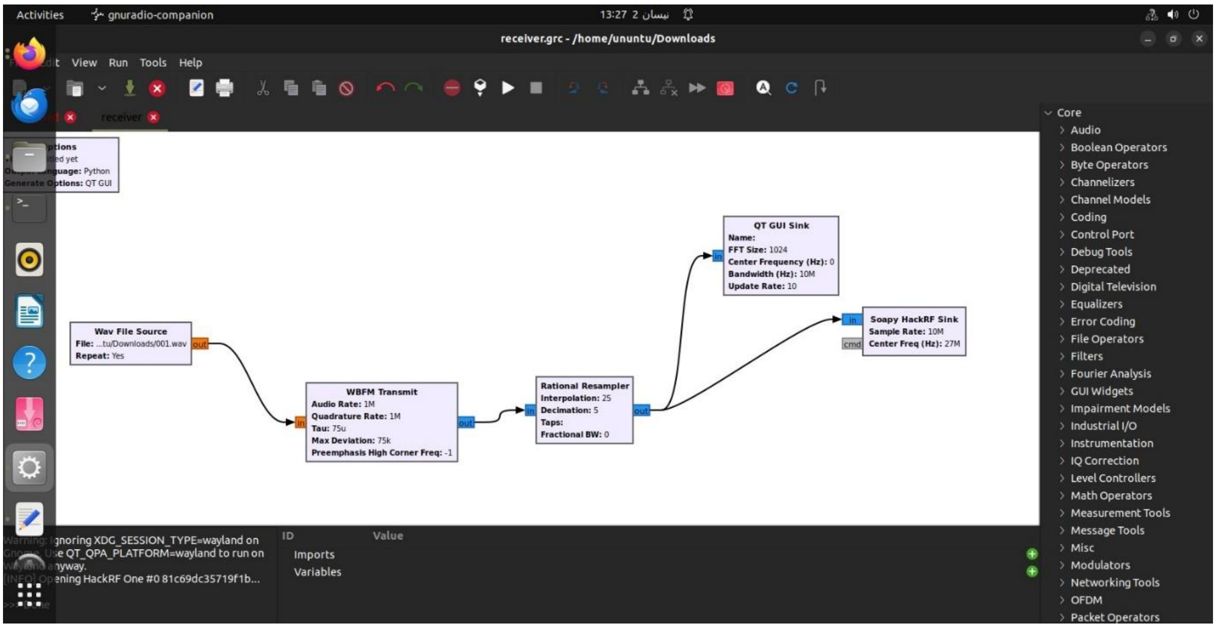
15.5.1 Sender



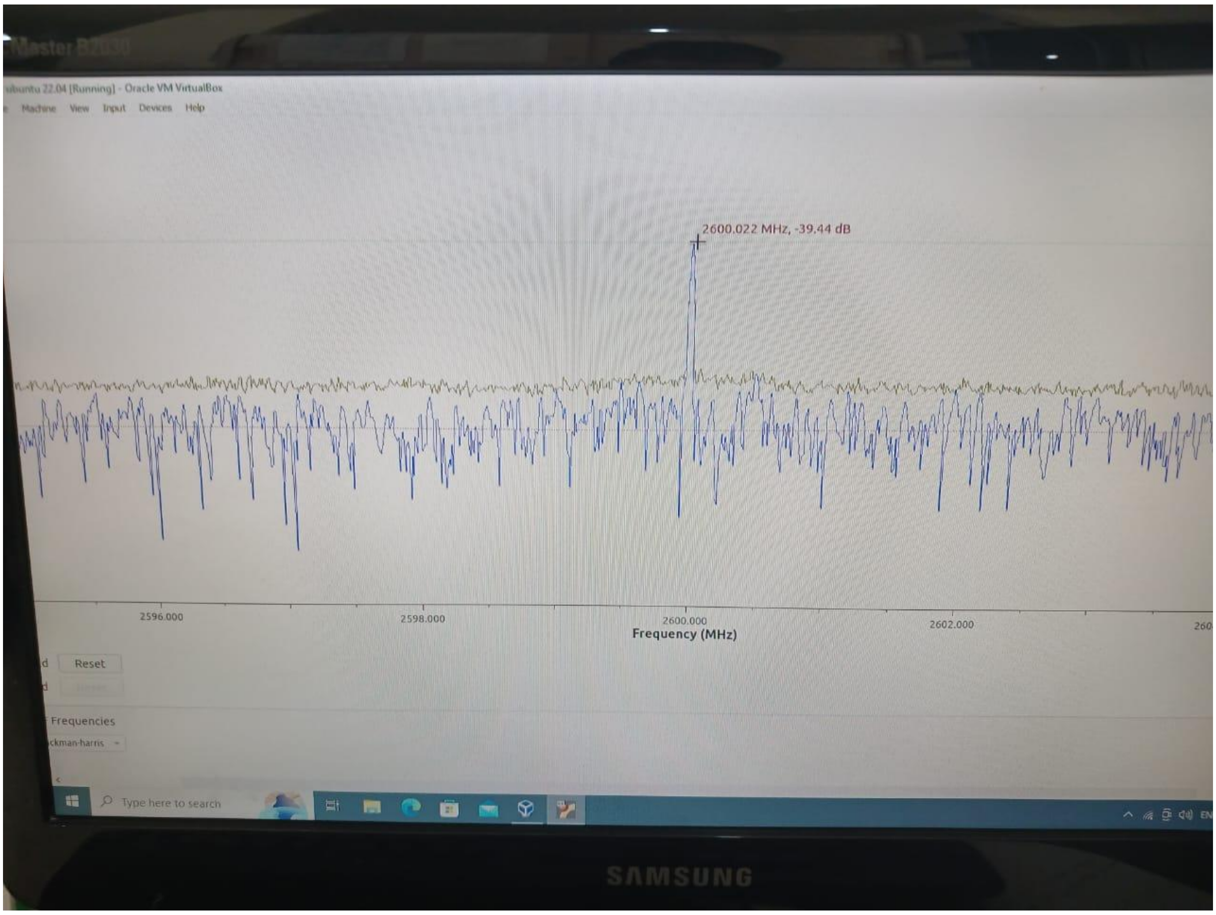


15.5.1.1 Sender Results:

15.5.2 Receiver



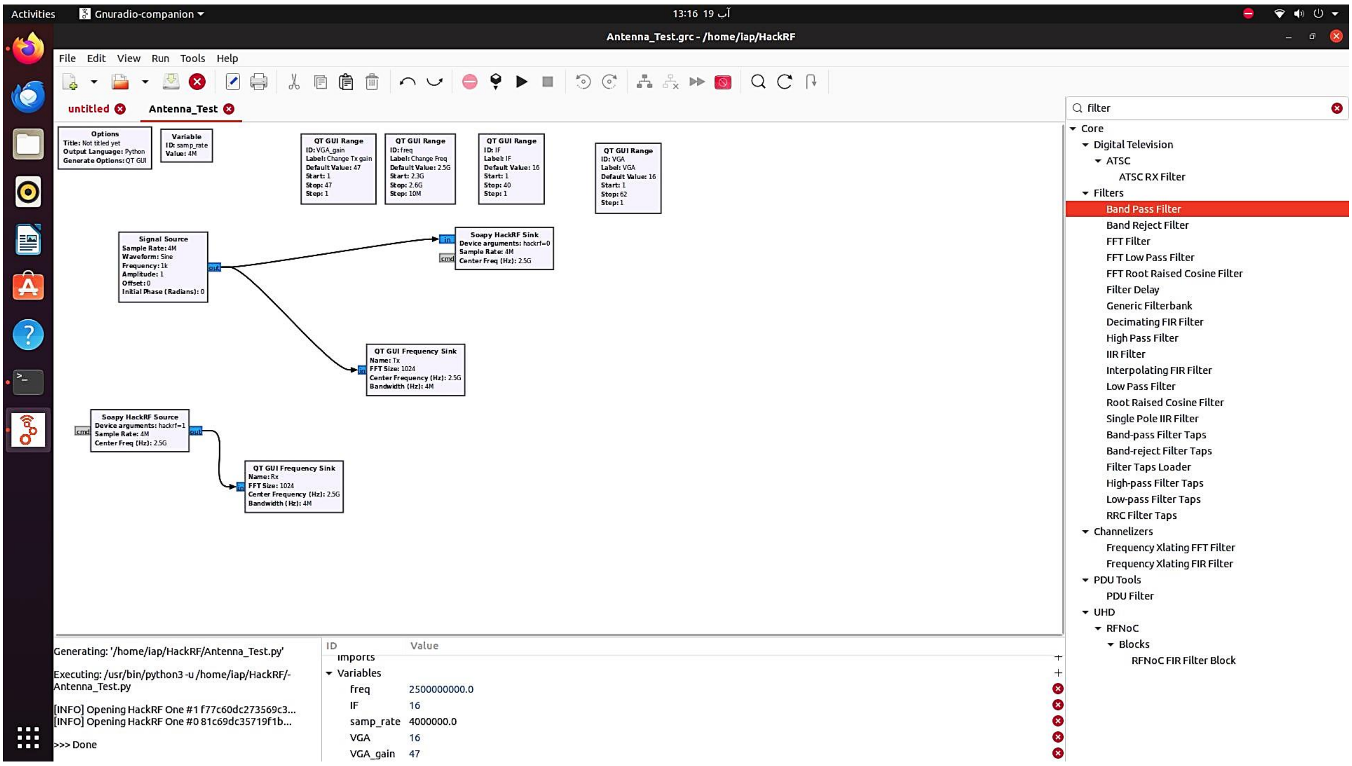
15.5.2.1 Receiver Results:



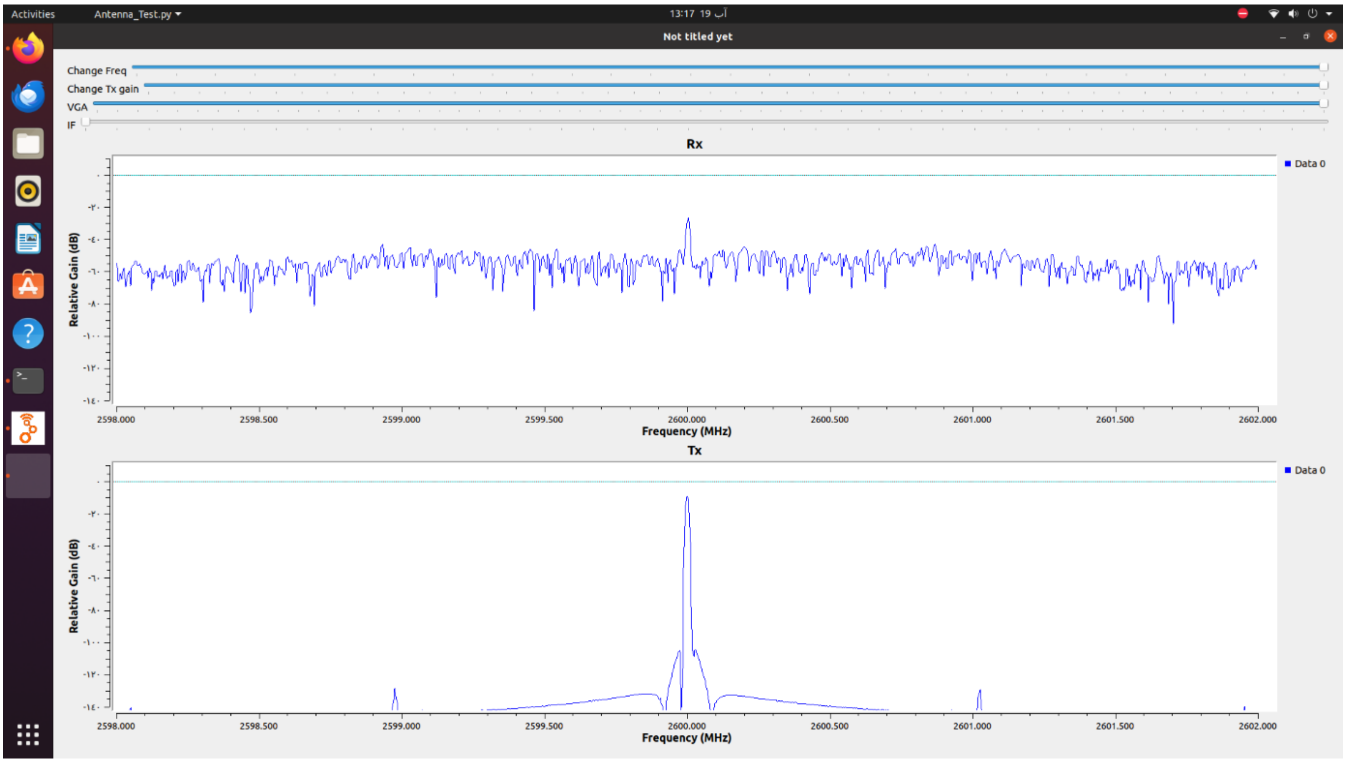
15.6 Graphical User Interface in Receiver Device in AS-COMSAT Antenna Testing Lab with GNU Radio

15.6.1 Sending and Receiving with HackRF and gnuradio

Simple circuit for sender and receiver



- Sending a sinusoidal wave through patch antenna
- soapy being the Tx
- source being Rx
- IF gain is the intermediate frequency which increases signal strength
- VGA gain is Variable Gain Amplifier. *Always increase VGA to the maximum for increased radiation. Note: These results are when both antennas are facing each other (angle 0)*


Activities
Antenna_Test.py
Not titled yet
Change Freq
Change Tx gain
VGA
IF
Rx
Data 0
Relative Gain (dB)
2598.000
2598.500
2599.000
2599.500
2600.000
2600.500
2601.000
2601.500
2602.000
Frequency (MHz)
Tx
Data 0
Relative Gain (dB)
2598.000
2598.500
2599.000
2599.500
2600.000
2600.500
2601.000
2601.500
2602.000
Frequency (MHz)

15.7 AS-COMSAT Antenna Teststand - Test Specification and Tests

15.7.1 Test Specification

Test Nr.	Test angle	Steps	Expected result	Result
1	0 Degrees	set the antennas facing each other Set VGA to max and IF to 12th slide.	Gain must be at maximum	
2	0 with iron plate	set the antennas facing each other Set VGA to max and IF to 12th slide. Place the iron plate very close to the antenna	Gain must decrease	
3	45	Set the antennas at 45 degrees keep VGA and IF like before. remove iron plate	Gain must decrease slightly.	
4	90	Set the antennas perpendicular to each other keep VGA and IF like before.	Gain must decrease more	

5	180	Set the antennas facing opposite to each other keep VGA and IF like before.	Gain must decrease drastically	

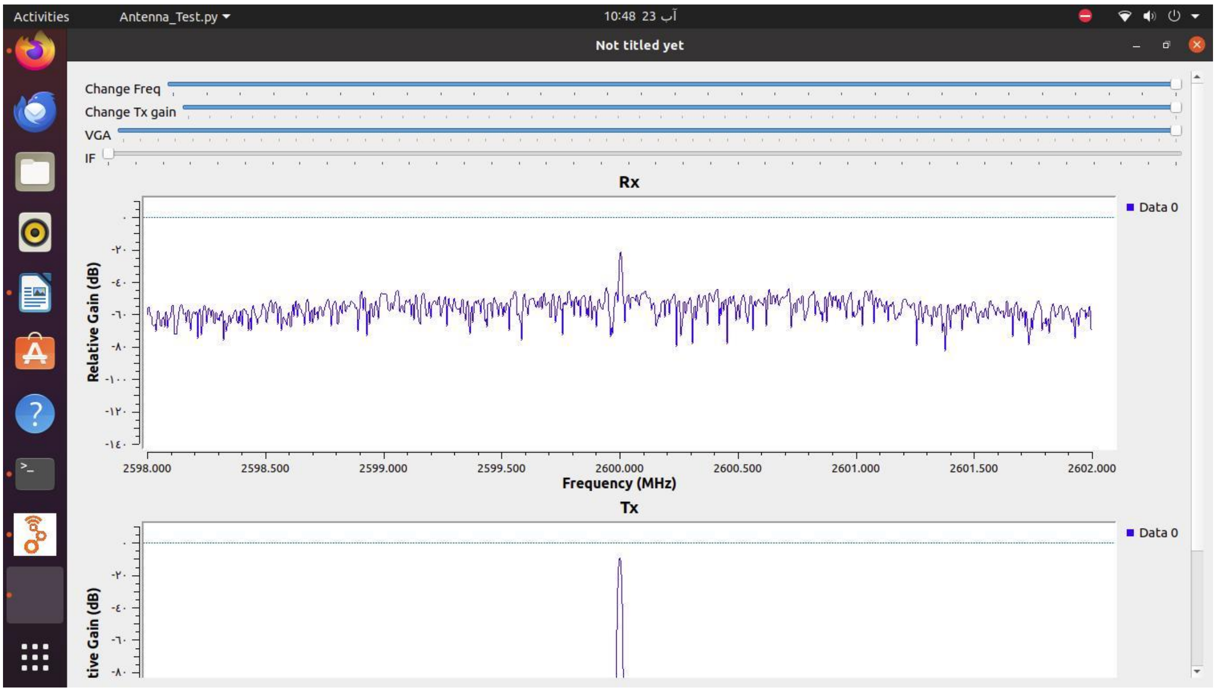
15.7.2 Tests

15.7.2.1 Test 23/08/2024

Test Nr.	Test variable	Steps	Expected result	Real result
1	0 Degrees	set the antennas facing each other Set VGA to max and IF to 12th slide.	Gain must be at maximum	gain is -9.8 dB (Maximum Gain)
2	0 with iron plate	set the antennas facing each other Set VGA to max and IF to 12th slide. Place the iron plate very close to the antenna	Gain must decrease	No signal recieved Gain is -42 dB
3	45	Set the antennas at 45 degrees keep VGA and IF like before. remove iron plate	Gain must decrease slightly.	Gain remained the same at -9.8 dB
4	90	Set the antennas perpendicular to each other keep VGA and IF like before.	Gain must decrease more	-15 dB decreased slightly
5	180	Set the antennas facing opposite to each other keep VGA and IF like before.	Gain must decrease drastically	Gain is at -30 dB Decreased so much

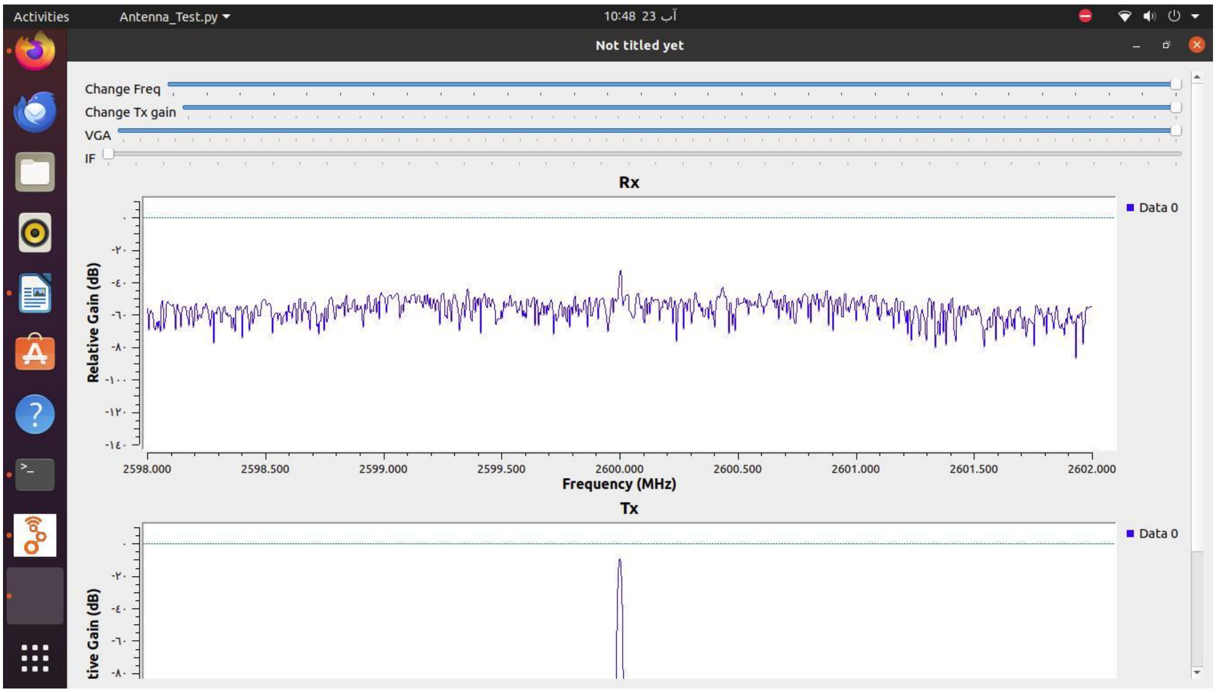
15.7.2.2 Test1 23/08/2024 - Without placing iron plate

Gain was -19.8 dB



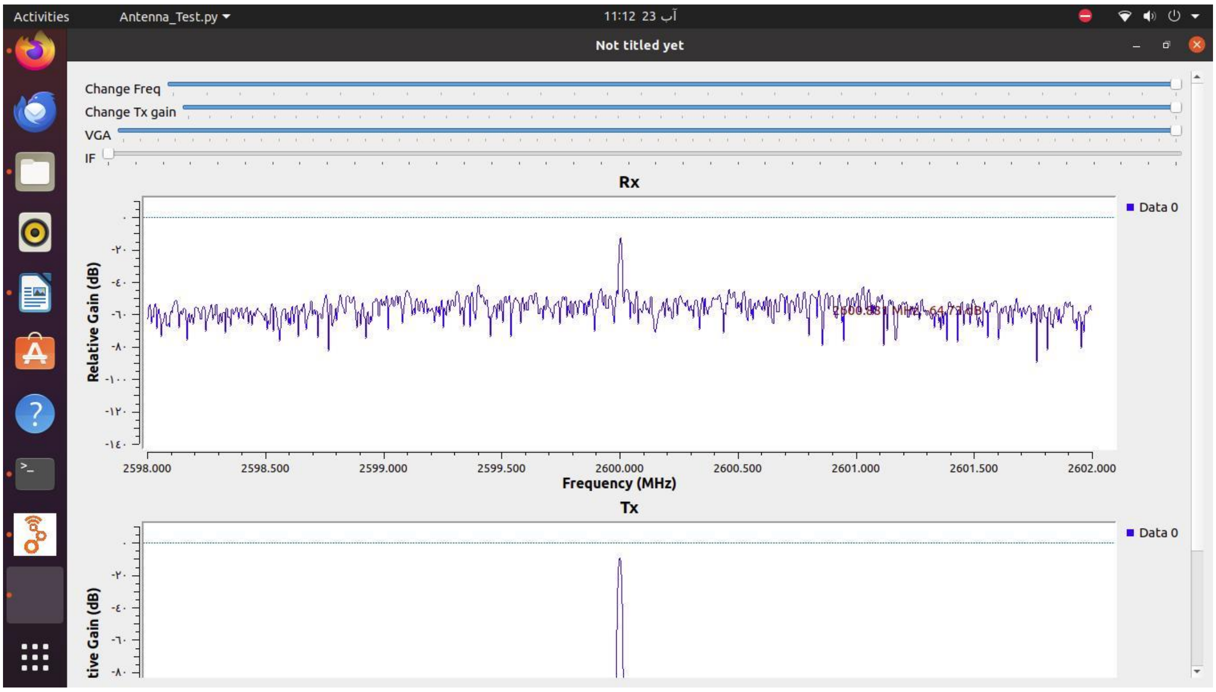
15.7.2.3 Test1 23/08/2024 - After placing iron plate

Gain got down to -42 dB



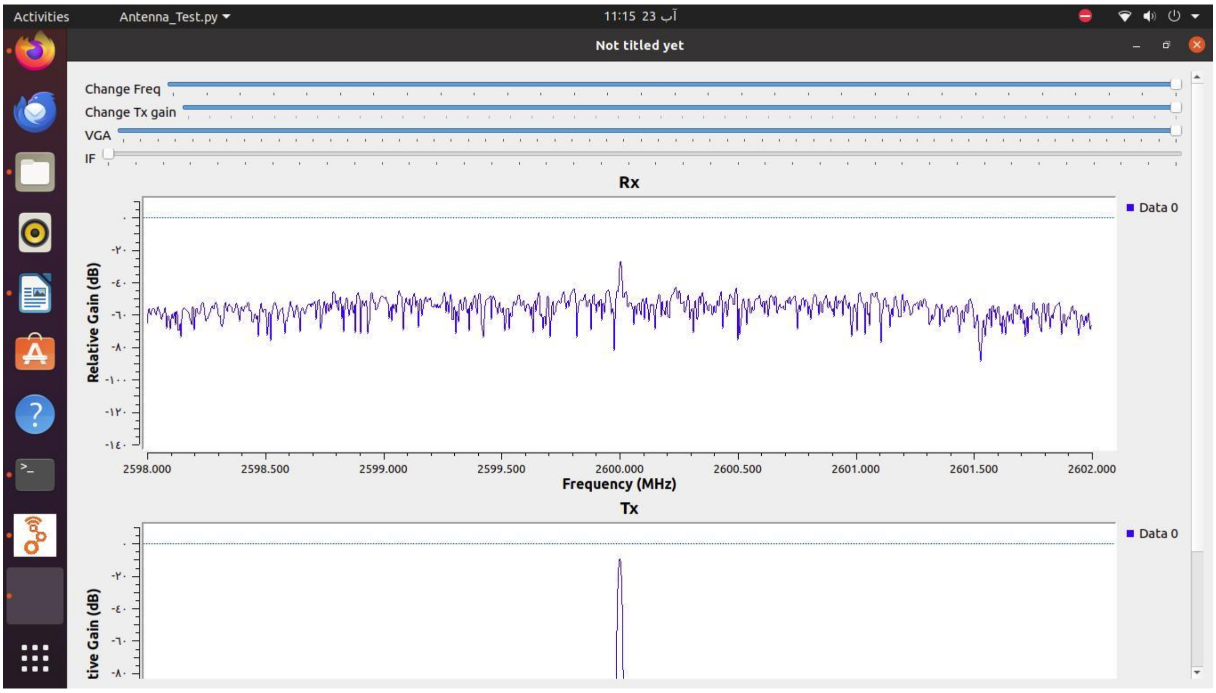
15.7.2.4 At 90 degrees:

Gain at -12 dB



15.7.2.5 At 180 degrees:

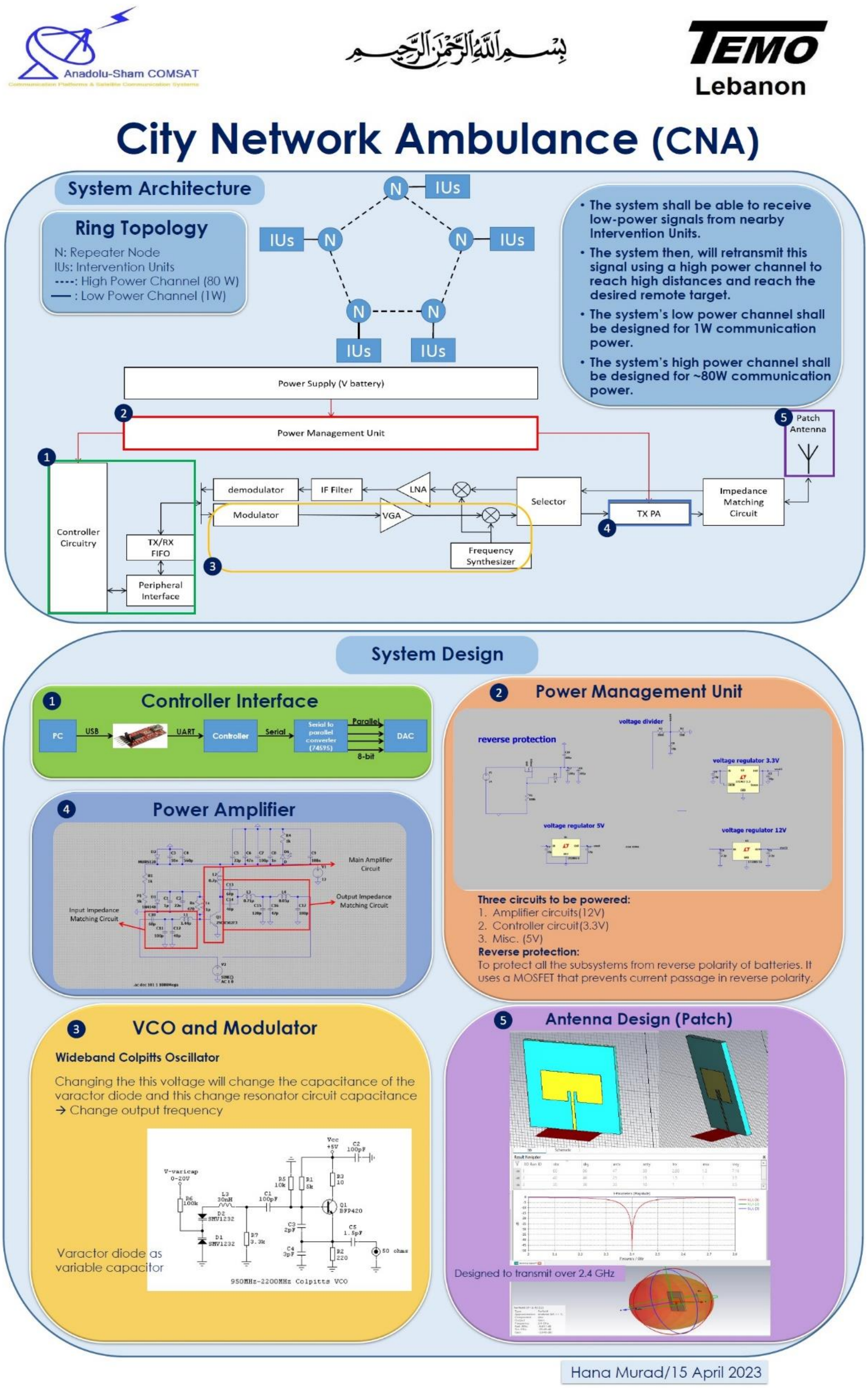
Gain at -30 dB



15.8 Literature

https://en.wikipedia.org/wiki/Antenna_measurement

16 City Network Ambulance (CNA), Baseline 31.12.23



AECENAR
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

Anadolu-Sham COMSAT

City Network Ambulance (CNA) System Architecture & Functionality

CNA System Architecture

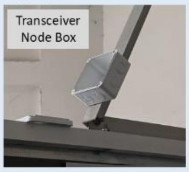
We have a "Star Topology" like architecture representing a prototype scheme for a network and its connectivity.

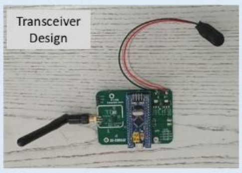
The way the network works is by using a central gateway to control the flow of the transmitting and receiving of data.

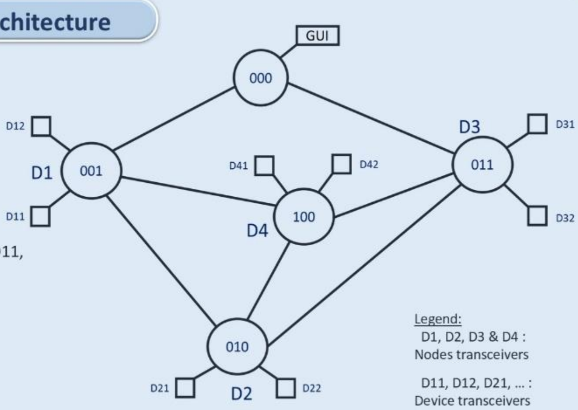
The central gateway will be a Graphical User Interface (GUI) connected on one of the transceivers.

The CNA GUI will serve as the gateway between all the Transceivers.

Each Node (D1, D2, D3, D4) will broadcast a request to all the connected transceivers (D11, D12, D21, …) on the network and will store all the devices into an Address Table.







Frame Architecture

The Frame will be distributed into 4 parts:

* First Part contains the "Source Device Address" which consists of 4 Bytes (2 Bytes for the Node Source Address and 2 Bytes for Device Source Address).

Source Device	Destination Device	Gateway Address	Packet
4 Bytes	4 Bytes	2 Bytes	1014 Bytes

* Second Part contains the "Destination Device Address" which consists of 4 Bytes (2 Bytes for the Node Destination Address and 2 Bytes for Device Destination Address).

* Third Part contains the "Gateway Address" which will be the address of the GUI (0000) and it consist of 2 Bytes.

* Last Part contains the "Packet" that will be sent across the network and it consists of 1014 Bytes.

Functionality

There will be 3 possibilities on how the data will be sent across the network and how the architecture will handle it:

- Case 1: GUI To Node:

- The Frame will contain the Device Source Address which is 0000 001 that is the standard address related to the central GUI, and the Destination Address that the user specifies according to the address that he wants to communicate with.

- The Frame will also contain the Gateway Address which will always be the address of the GUI so each packet can reach the GUI and the GUI will transmit the data to the network to reach the specified address.

- The last piece of the message will contain the packet that will be sent as a message to the other user.

Case 1

0000	001	0010	011	0000	Packet
Node Source Address	Device Source Address	Node Destination Address	Device Destination Address		

- Case 2: Node To GUI:

- The Frame will contain the Device Source Address which is the address of the device related to the network, and the Destination Address that is the central gateway in this case which is 0000 001.

- The Frame will also contain the Gateway Address which will always be the address of the GUI so each packet can reach the GUI and the GUI will transmit the data to the network to reach the specified address.

- The last piece of the message will contain the packet that will be sent as a message to central GUI.

Case 2

0010	001	0000	001	0000	Packet
Node Source Address	Device Source Address	Node Destination Address	Device Destination Address		

- Case 3: Node To Node:

- The Frame will contain the Device Source Address which is the address of the device related to the network, and the Destination Address that the user specifies according to the address that he wants to communicate with.

- The Frame will also contain the Gateway Address which will always be the address of the GUI so each packet can reach the GUI and the GUI will transmit the data to the network to reach the specified address.

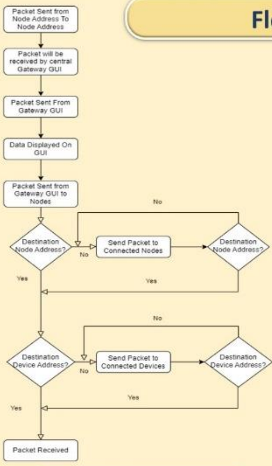
- The last piece of the message will contain the packet that will be sent as a message to the other user.

Case 3

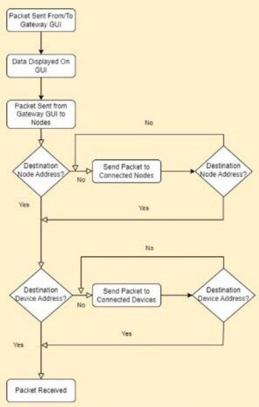
0010	001	0011	100	0000	Packet
Node Source Address	Device Source Address	Node Destination Address	Device Destination Address		

In all cases, the Frame will pass through the central gateway (GUI) so the information will be displayed and the GUI will be responsible for transmitting the data again to the network

Flow Chart



Node To Node Flow Chart



GUI to Node or Node to Node Flow Chart

Riyad Murad @AS-COMSAT – 06 OCT 2023

بسم الله الرحمن الرحيم

Anadolu-Sham COMSAT

Communication Platforms & Satellite Communication Systems

AS-COMSAT CNA

City Network Ambulance System Development - Report 2023

Written By:

Raja Mourad

Riyad Mourad

Ahmad Awwad

Ahmad Dannawi

Hana Mourad

Asmaa Dhaybi

Editor: Samir Mourad

Last Update: 08.01.2024 14:39

Table of Contents

16.1 Introduction

This document aims to explain the development process of the City Network Ambulance system designed by AS-COMSAT.

16.2 System Overview

The CNA system is divided into 2 categories:

1. Gateway Ground Station system: which is a Graphical User Interface (GUI) that acts like a central communication device between the users and the destination nodes. It receives and displays the messages from the users and redirect them to the destination intended. It also sends messages via the nodes.
2. System Design: CNA system consists of five transceiver nodes. Users will connect to the nodes via the user device (GUI). If the user wants to send a message to a certain node, the transceiver connected to the user's device (including user GUI) will transmit the packet to the nearest node. The process will be repeated until the packet reaches the transceiver of the Gateway Ground Station which is connected to it serially. The GUI will then process the packet, display the message, and redirect the packet to the node the user intended to send the packet to.

Figure 2-1 illustrates the process of sending a packet from the user's device to the GUI.

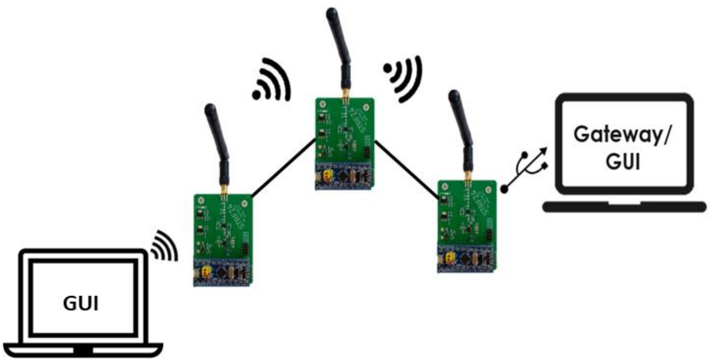


Figure 2-1- Message Sending Process

The poster shown in Figure 2-2 summarizes the overview of the system.

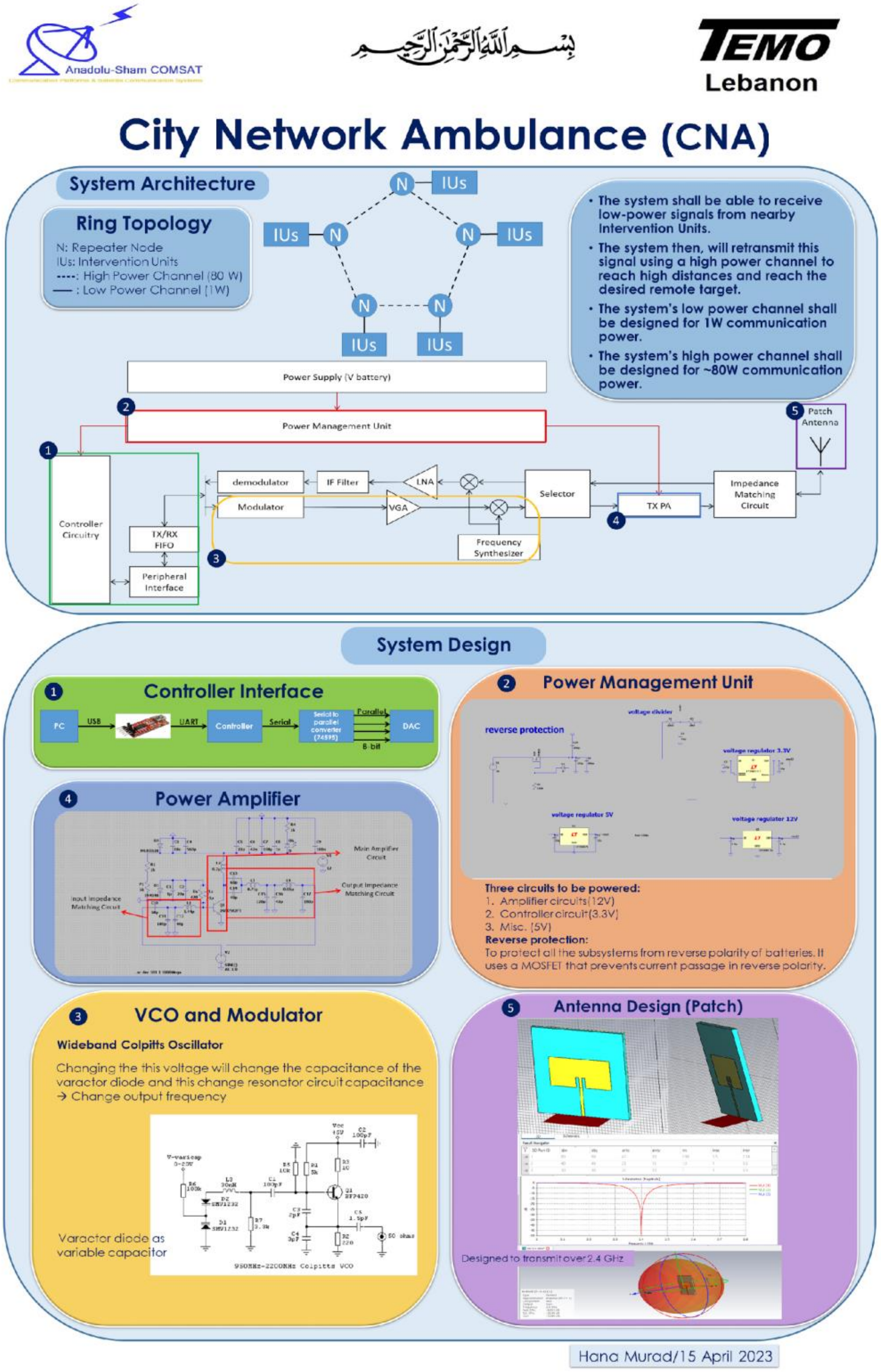
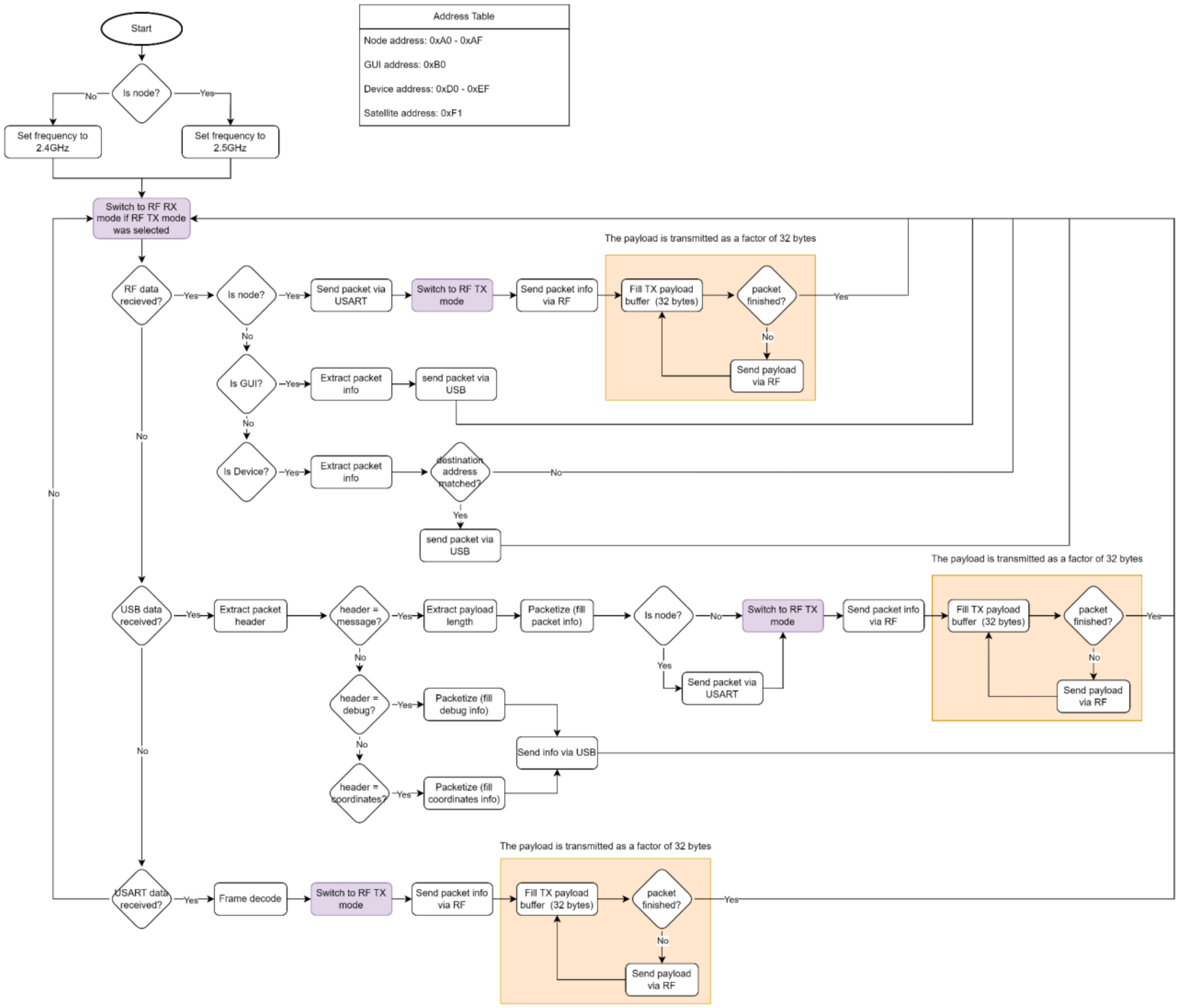


Figure 2-2- CNA System Overview

16.3 System Design and Configuration

The following flowchart is used to design the transceiver software on the STM32 for each node, device, satellite or GUI.

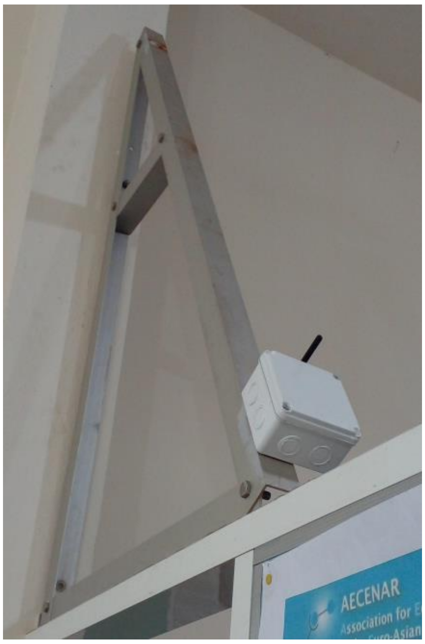


16.3.1 Introduction

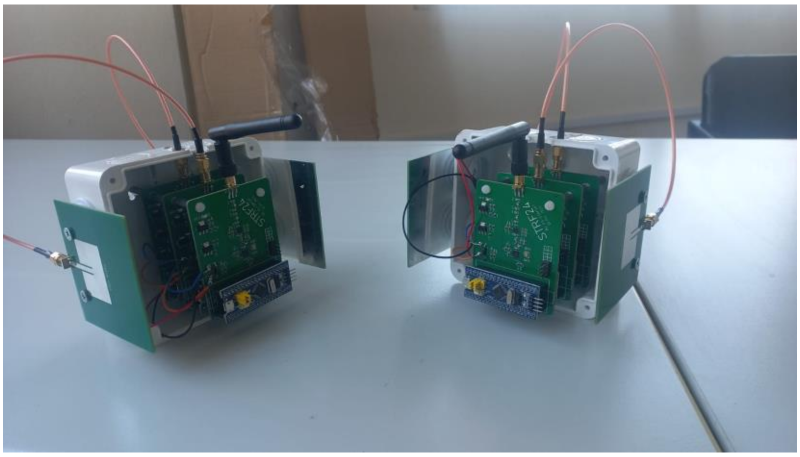
Chapter 3 tackles the design part of the system.

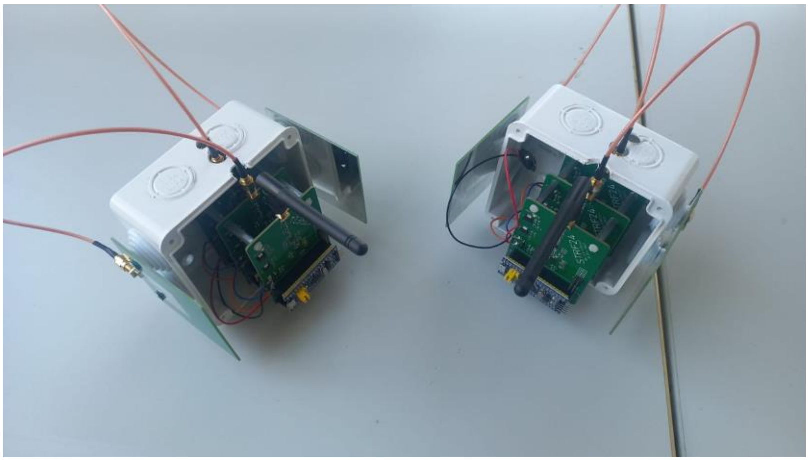
16.3.2 Mechanical Realization

This section shows where the nodes where placed.

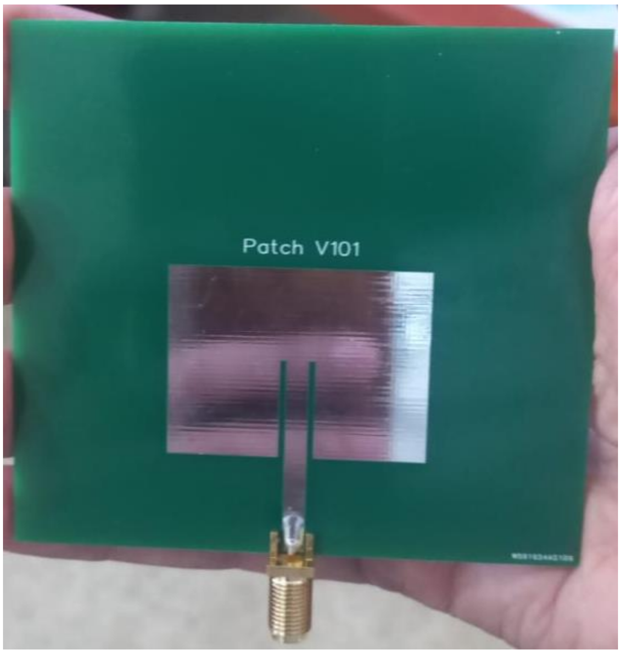


16.3.3 CNA Transceiver





16.3.4 Antenna



16.3.5 Software Design (STM32 Low Level SW Architecture)

This section shows how the transceivers will work and how the communication between nodes will occur until it reaches the user. The following figure will show how the Transceiver nodes will work.

STM32 code (version 802)

https://drive.google.com/drive/folders/1FW6pGfZCofv_EuiDp9o6juZRKiaZba55?hl=en

STM32 code (version 803)

https://aecenar.com/index.php/downloads/send/16-ics/1487-cna-transceiver-sw803

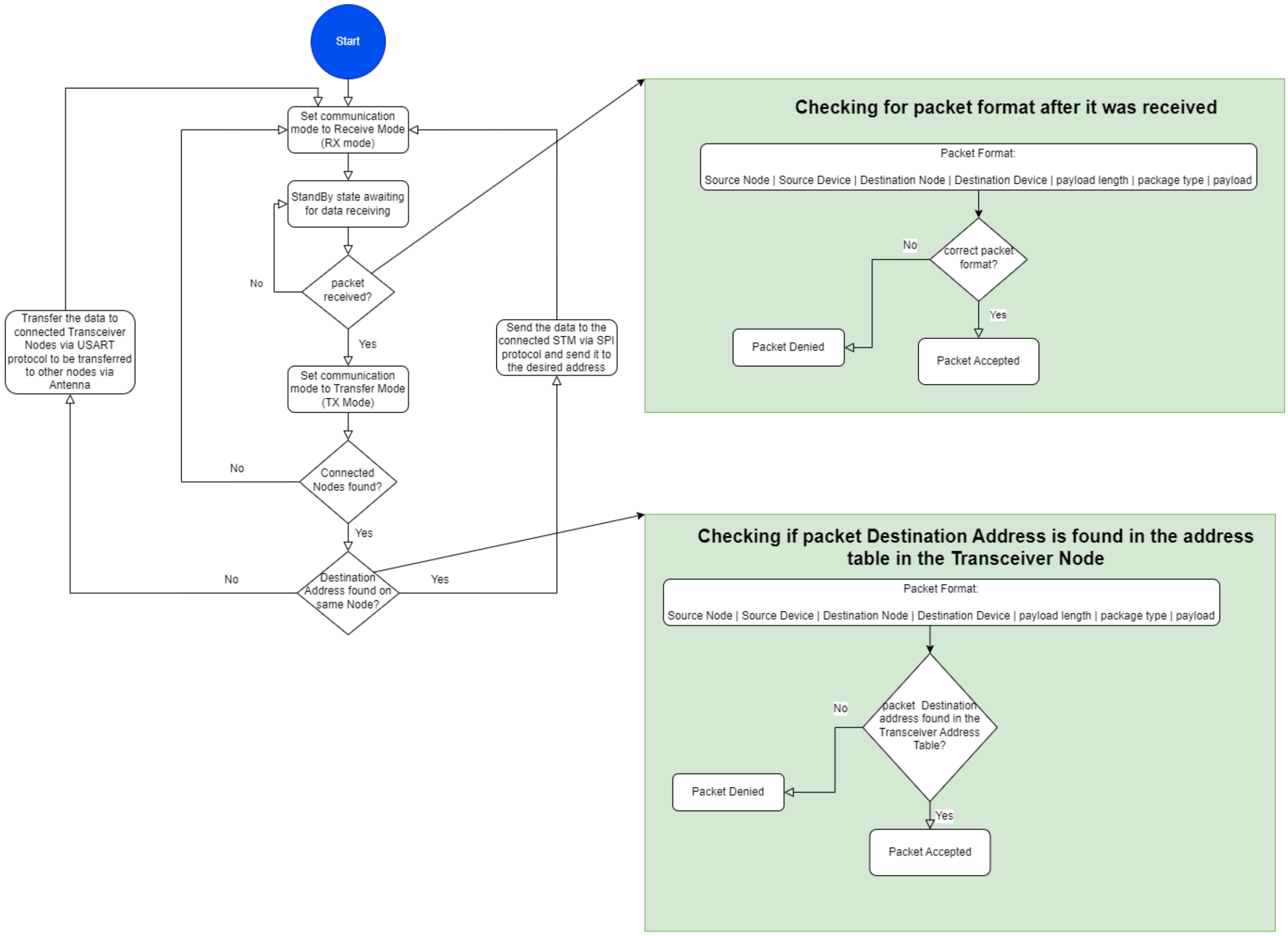


Figure 3.5 - 1

First, all the transceivers will act as a "Receiver" node (RX communication mode) and wait to receive data. The transceiver will wait in a "StandBy" state waiting for any actions. Second, the transceiver checks if it received any packets or not. If it received a packet, it then checks if the format of the packet is correct and is not missing any information. If the packet format is not correct, then the packet is denied and the transceiver returns to the StandBy state. If the packet format is correct, the packet is accepted and the communication mode is changed to "Transmitter mode" (TX communication mode). Lastly, it checks for the destination address of the packet if it is found in the Address Table of the transceiver, it sends the packet to the connected STM via SPI protocol to be redirected to the corresponding device address. If the address is not found in the Address table, the packet is then transferred to connected transceivers via USART protocol to be transferred to other nodes via Antenna.

The next figure, will show how the user will react with the message.

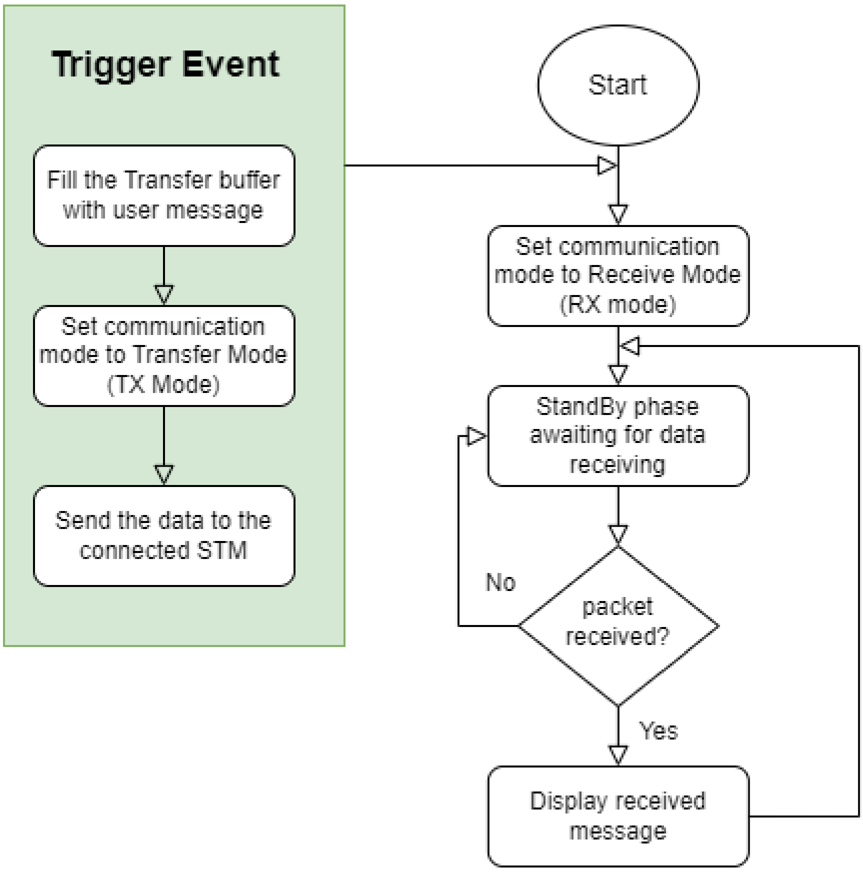


Figure 3.5 - 2

The user transceiver communication mode is also set to "Receiver mode" (RX mode) and it also waits in a "StandBy" state waiting for any actions. If the packet is received, it displays the message to the user. The user can also trigger an event where he can send a packet to other users. The packet fills the TX_buffer with the message. It then set the communication mode to Transfer Mode "TX mode". Lastly, it sends the data to the STM to be forwarded to the network via Antenna.

16.4 Gateway Ground Station

16.4.1 Introduction

This chapter will introduce the steps of developing the graphical user interface (GUI) and explain how the classes in the GUI interact with each other. A sequence diagram will be used to show how the data is being transferred between the forms and the use case diagram will illustrate the actions that can be done by the user. Moreover, each function in the code will be explained.

16.4.2 System Architecture and Functionality

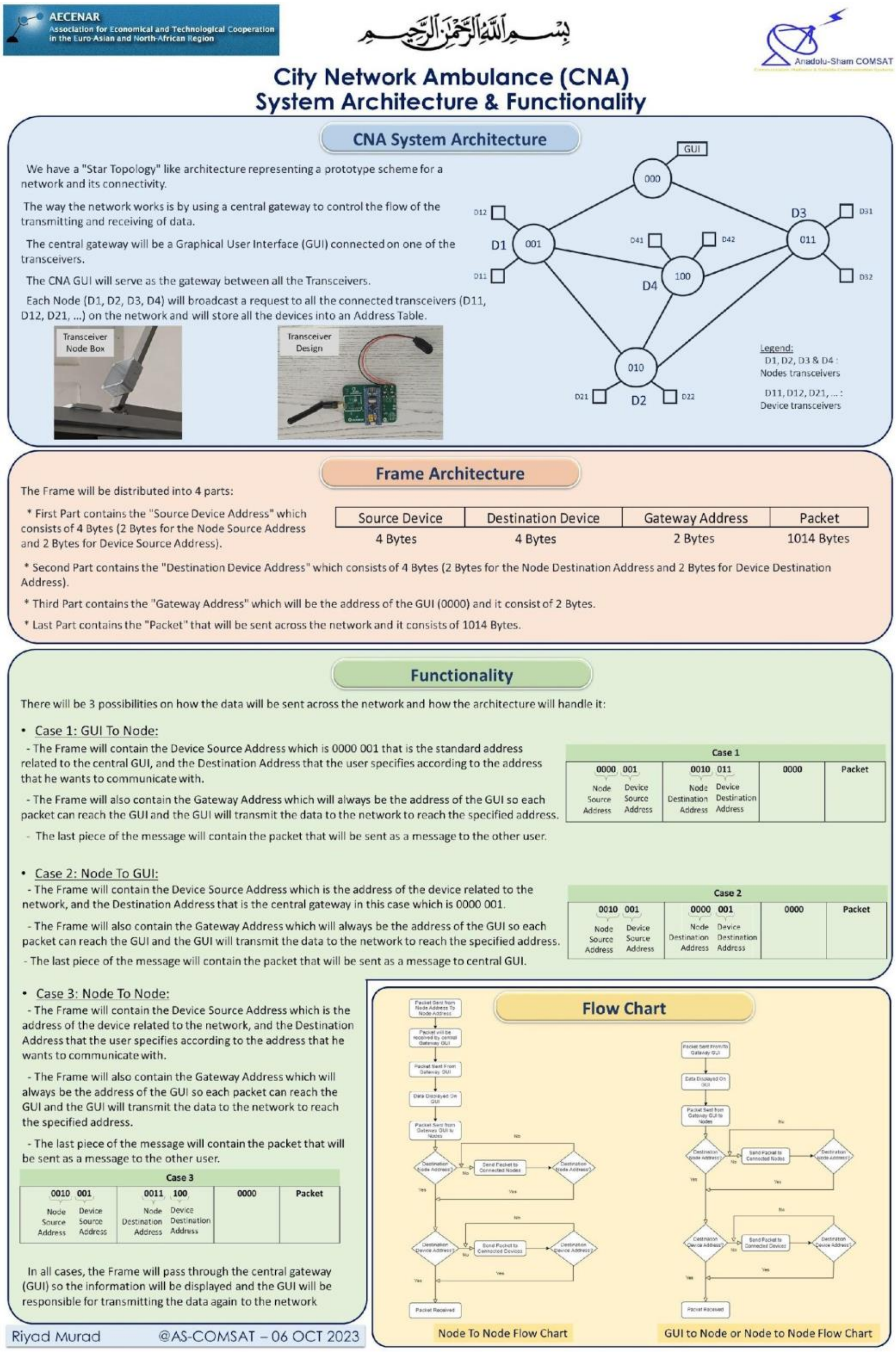


Figure 4-1- CNA System Architecture and Functionality

16.4.3 GUI Overview

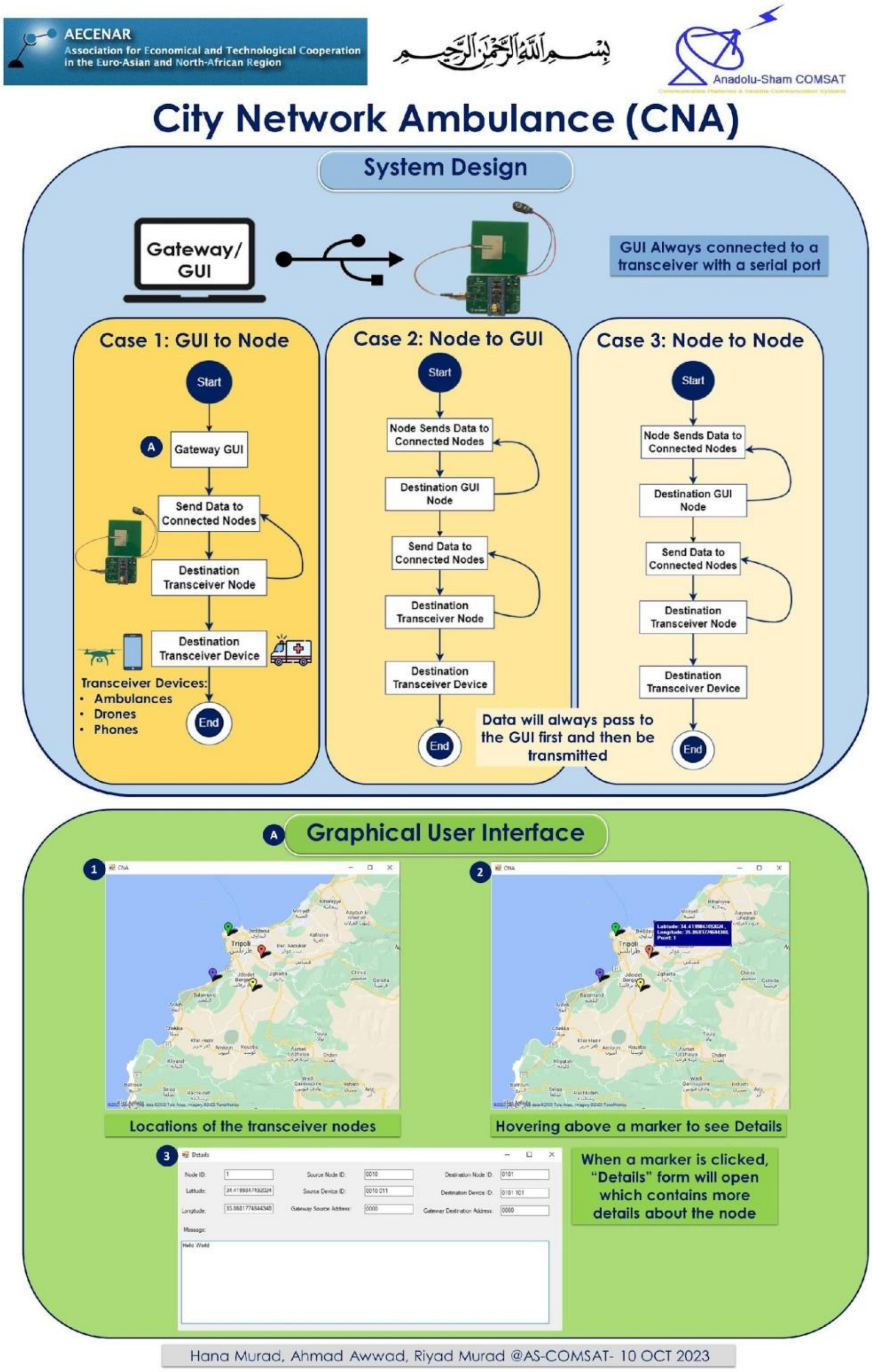


Figure 4-2- CNA GUI Overview

16.4.4 Customer Software Requirements

Table 4-1- Customer Software Requirements Table

ID	Title	Description	Safety Level	Verification Criteria
Req.0001	Software App	Visual Studio IDE.	-	-

Req.0002	Coding language	C# Language.	-	-
Req.0003	Extension	Should include Windows Form App(.NET Framework extension).	-	-
Req.0004	Extension	Should include GMAP API.	ASIL-D	Open a new C# Windows Form application on visual studio > On the right side of the screen, in the solution explorer tab, press right click on the project name > Manage NuGet Packages > Installed > You can find "GMap" APIs.
Req.0005	Compatibility	Should include a compatible software with the operating system on which it will be running, preferably Windows.	ASIL-D	-
Req.0006	Hardware	Should connect an STM32 to the device for the GUI to work	ASIL-D	STM node Transceiver should be connected to your device via Serial Cable.
Req.0007	User Interface	Should include a map to display the coordinates of the node transceivers.	ASIL-D	Coordinates should be displayed on the map
Req.0008	User Interface	Should include a Label to display if the required transceiver is connected.	ASIL-D	When the specified transceiver port is connected, the label will display that the transceiver is connected and display it's information.
Req.0009	User Interface	Should display the frames received.	ASIL-D	Data received should all appear in text boxes and on the map.
Req.0010	User Interface	Should send frames to node transceivers.	ASIL-D	Check if the frame reached the node by writing a "Console" statement to display the frame.

16.4.5 GUI Sequence Diagram

Figure 4-3 below shows the time sequence of the GUI behavior.

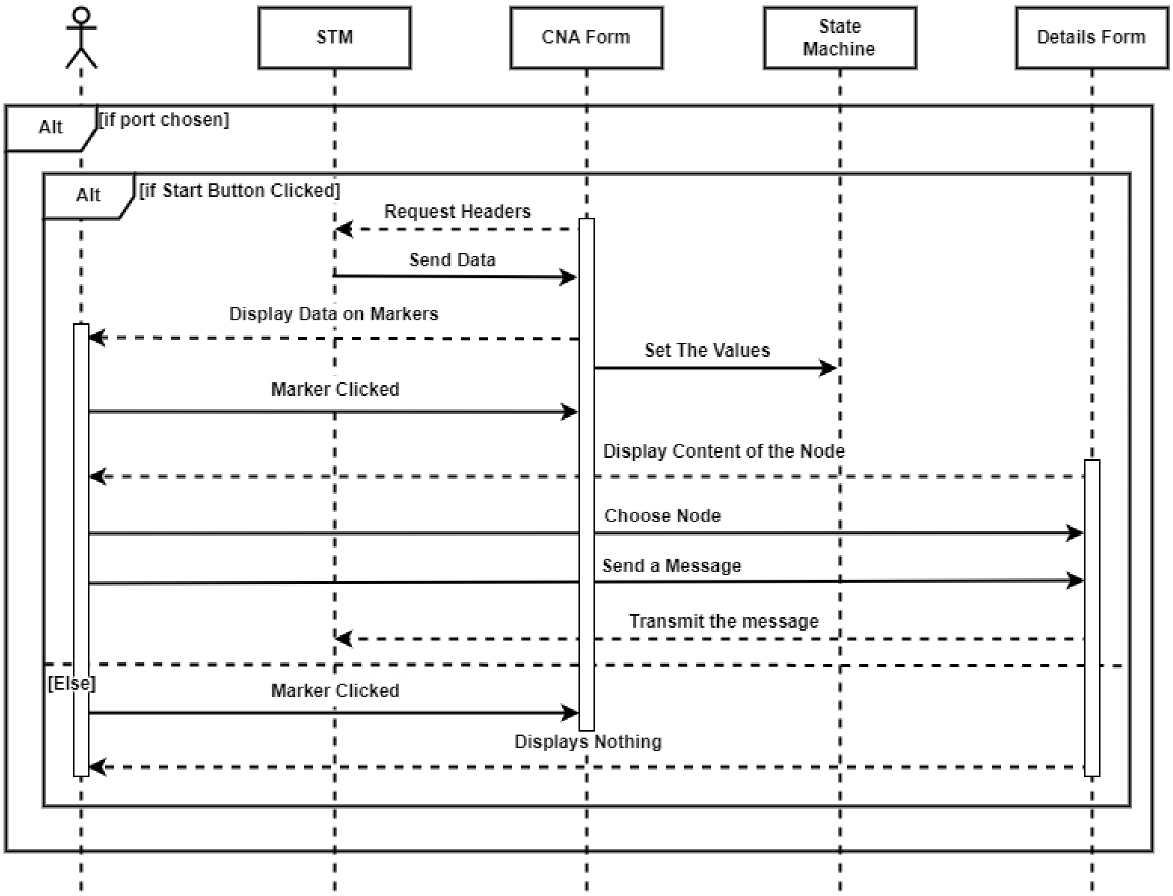


Figure 4-3- System Sequence Diagram

The user chooses a port from the combo box and clicks the start button. The port will be opened and the GUI will request headers from the STM. In turn, the STM will send the frames to be received by the GUI and stored in variables and send them to the StateMachine. When the user clicks the marker they will be redirected to the "Details" form. The form contains the coordinates and messages sent to this node. The user can also send messages to the other nodes from this form. If the start button was not clicked, the Details form will not contain any information as the port was not opened.

16.4.6 GUI Use Cases

In Figure 4-4, all the actions that can be performed by the user in the GUI are represented.

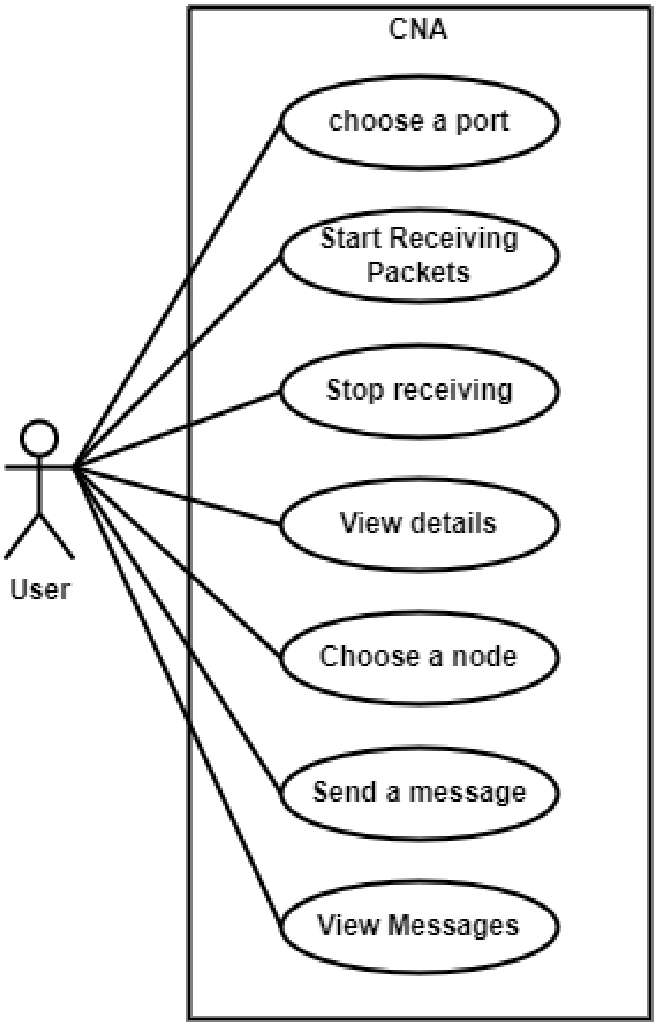


Figure 4-4- System Use Case Diagram

16.4.7 Gateway Ground Station/Transceiver Interface

The transceiver of the Gateway Ground Station is similar to the TT&C Transceiver. Gateway Ground Station (GS) Transceiver is used to transmit messages from the GUI to the chosen node and to deliver received messages from TT&C transceiver of the nodes to be displayed on the GUI Form.

The data received by the GS transceiver consists of frames with the structure as shown in Table 4-2 below.

Table 4-2- Schedule Table

Frame Name	**ID**	**Length**	**Content**		
			Byte 2	**Byte 3**	**Byte 4 till end**
ID_FRAME	0x1A	2	Device Address	-	-
Message Frame	0x60	32	Source Device	Destination Device	Message
Command Frame	0x70	32	Source Device	Destination Device	Message

Debug Frame	0x80	3	trx_status	battery_voltage	-

Frame Name	ID	Length	Content	
			Byte 2 to 5	Byte 6 to 9
Coordinates Frame	0x50	9	Latitude	Longitude

The Command frame is the one sent from the GUI to Destination Device.

16.4.8 GUI Design

16.4.8.1 System Requirements

Transceiver Nodes to Gateway:

1. Node ID
2. Node Latitude
3. Node Longitude
4. Source Node ID
5. Source Device ID
6. Destination Node ID
7. Destination Device ID
8. Gateway Address
9. Message Packet

Gateway to Nodes:

1. Message
2. Destination Node
3. Destination Device

16.4.8.2 Building the Interfaces (Developers Manual)

Form 1: Main Form:

This form will allow the user to choose a port to start receiving data from the nodes and shows the position of the nodes on the map.

To do that:

1. To insert the Google Map API, a library should be installed:
   a. Go to Manage NuGet Packages.
   b. In the search box of the browse tab type gmap.
   c. Select the file "Gmap.Net.Windows"
   d. A gmap control will be added to the toolbox
   e. Drag and drop the control to insert the map inside the slicer.
   f. Place the anchor property on both the map so it adjusts to the changes of the interface (expand, reduce).
   g. Anchor the map from the four directions (top, right, bottom, left).

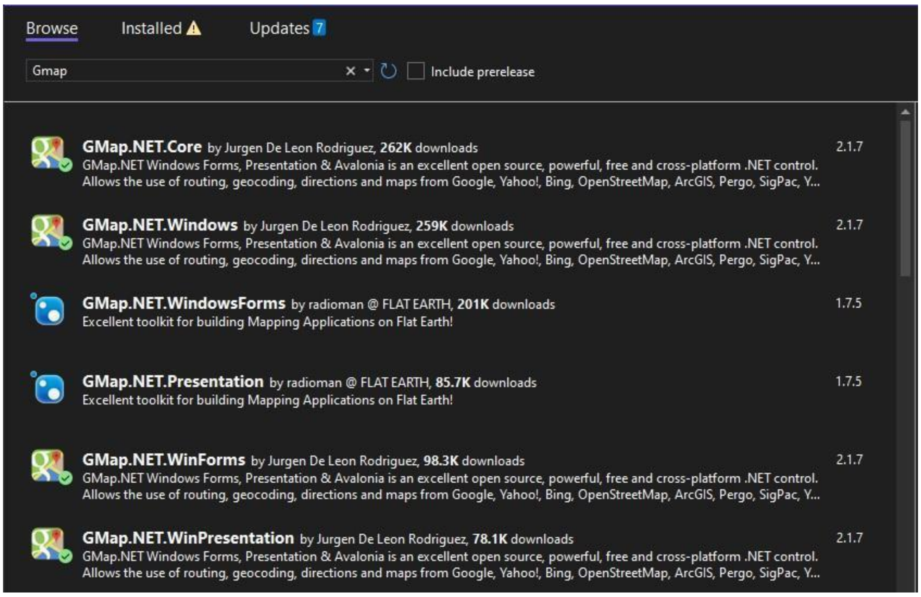


Figure 4-5- GMap.Net.Windows Library

2. Click the map to change its name to “map” in the properties.
3. Add a combo box and change its name to “AvailablePorts”.
4. Add a button and set its name and Text to “Start”.
5. Add a button and set its name and Text to “Stop”.

The form should look like Figure 4-6 below.

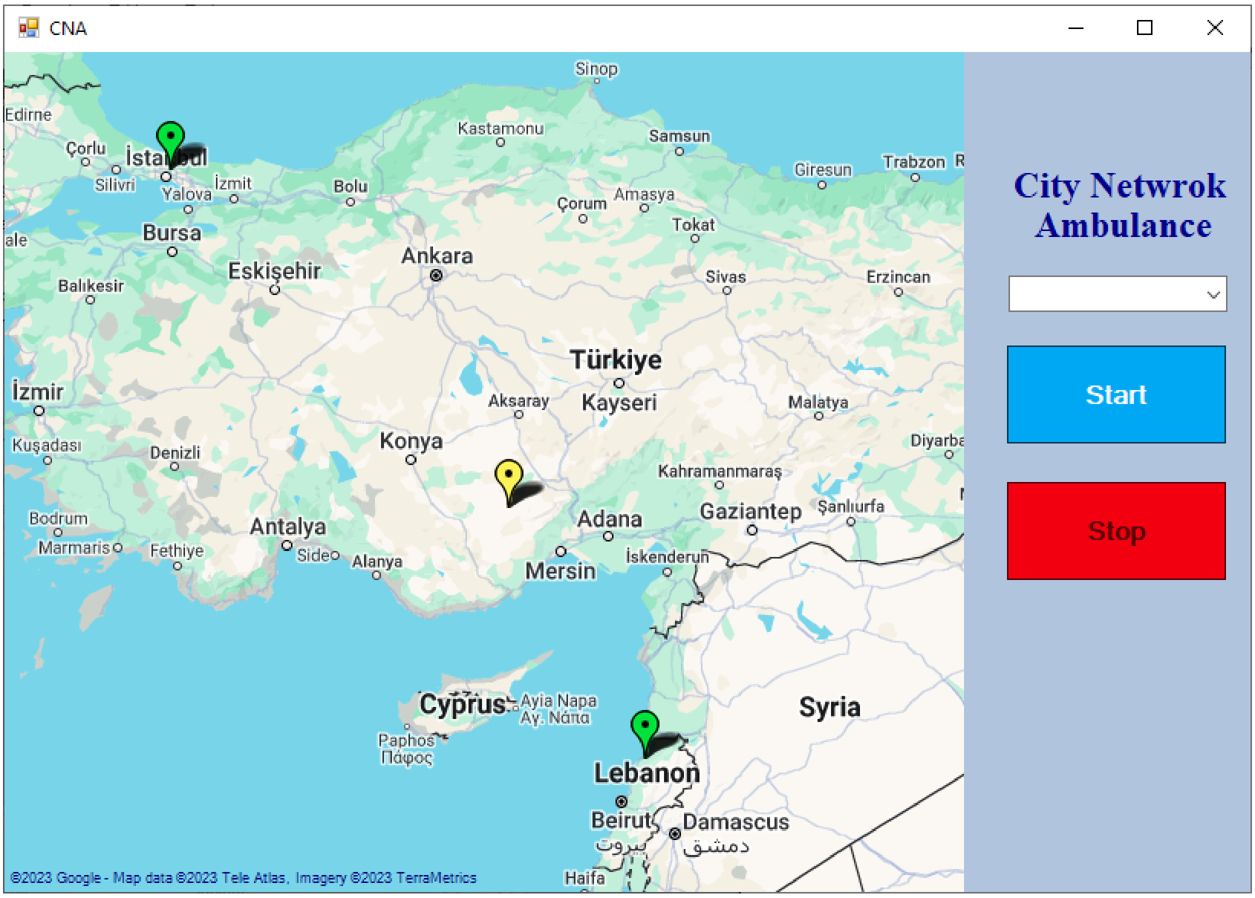


Figure 4-6- Main Form

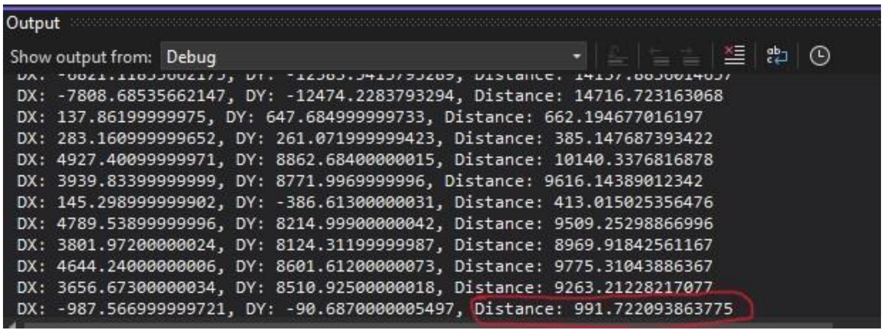


Figure 4-7- Main Form

If the distance between the markers is less than 1 Kilometer, then a line will be drawn between them. The “Device” marker is in green and the “Satellite” marker is yellow.

<u>Form 2:</u> Details

This form contains more details about each node. It displays the messages received by this node and contains a section to send data to other nodes.

1. Place a text box to display the values of each of the variables mentioned in the requirements section.
2. Place 2 text boxes
   a. Set Enabled = False.
   b. Set Multiline = True.

c. Set Scrollbars = Vertical.

3. Place a third text box with multiline = true.
4. Place a "Send" button, and set its name to msgSendBtn.

The interface should look like Figure 4-8 below.

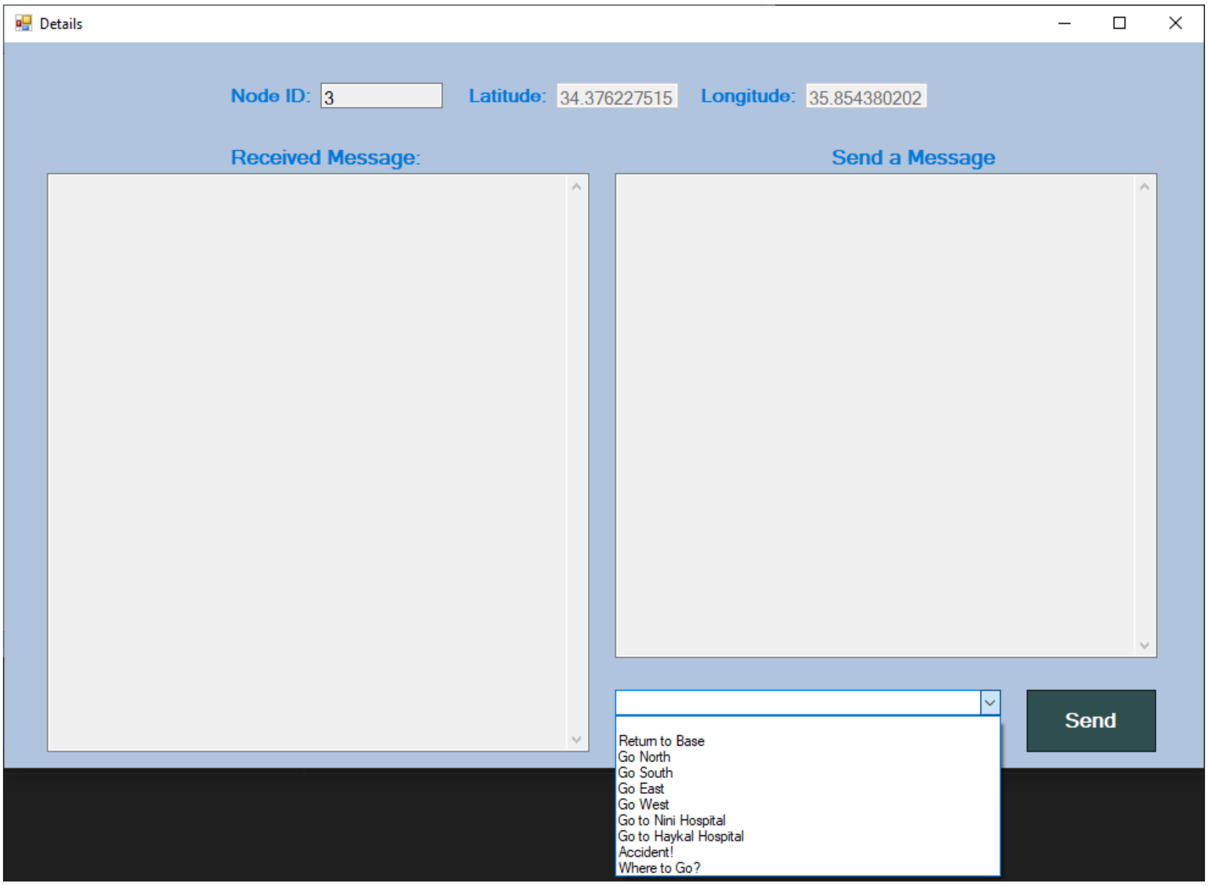


Figure 4-8- Details Interface

16.4.9 GS Transceiver Functions

- **CDC_Receive_FS:** when the STM receives the headers requested by the GUI, it fills the data in the buffer according to the request and calls the "CDC_Transmit_FS" function.
- **CDC_Transmit_FS:** this function transmits the data back to the GUI.

16.4.10 GUI Functions

The code consists of 2 forms and a StateMachine class.

The role and behavior of each form will be explained below.

<u>**CNA_Main_Form:**</u>

- **Enum Frames_ID_t:** variable containing the frames IDs matching with the transceiver IDs.

- **Main_Form:** initialize the components of the Form and calls an instance of the StateMachine class.
- **Main_Form_Load:** the first function that gets activated when the app starts.
    - Calls "InitializePortWatcher" function.
    - Calls "PopulateAvailablePorts" function.
    - Disables all the "Stop" button.
- **Map_Load:** identify the path of the file containing the coordinates. It then displays a marker on the map to represent each coordinate with some details above it.
- **<u>Port Connectivity Check Region:</u>** the functions found in this region are to automatically gather COM ports connected to the GUI.
    - **InitializePortWatcher:** it monitors the serial port of the PC to check when a new device is connected.
    - **PortWatcher_EventArrived:** when a port is connected it adds it to the comboBox by calling the "PopulateAvailablePorts" function.
    - **PopulateAvailablePorts:** Collects all the COM ports and adds them to the comboBox.
- **<u>Serial Data Region:</u>** this region is to receive and organize the data sent by the STM.
    - **serialPort_DataReceived:** generates an array of bytes to hold the received data and calls ProcessReceivedData function.
    - **ProcessReceivedData:** this function separates the data according to each frame ID. It stores each value in a variable and sends them to the StateMachine.
    - It generates a file that holds the messages for each node according to the node name.
- **<u>Stop/Start_Buttons:</u>**
    - **Start_Click:** this function gets triggered when the Start button is clicked.
        - It takes the selected port in the comboBox.
        - Enables all the stop button and disables itself.
        - It opens the serial port according to the comboBox selection.
        - Calls serialPort_DataReceived function to start receiving data.
    - **Stop_Click:** this function gets triggered when the Stop button is clicked.
        - Enables the Start button.
        - Stops the data_request_timer

- Disconnects the serial port.

- **Data_request_timer_Tick:** gets triggered on each time the timer interval elapses and send the header requests to the STM.
- **ShowErrorMessageAsync** and **error_Handler** functions are responsible for the error messages that appear when an error occurs.
- **Map_Region:** the functions in this region control the map markers and open the details form.
  - **map_OnMarkerClick:** gets triggered when a Marker is clicked. It opens the Details form and sends the selected node ID with the constructor of the form.

Main Form Code:

```
enum Frames_ID_t
{
    ID_FRAME = 0x1A,
    COORDINATES_FRAME = 0X50,
    MESSAGE_FRAME = 0x60,
    CMD_FRAME = 0x70,
    DEBUG_FRAME = 0x80
}
namespace CNA_GUI
{
    public partial class Main_Form : Form
    {
        private ManagementEventWatcher portWatcher;
        StateMachine sm;
        GMapOverlay markers = new GMapOverlay("markers");
        public Main_Form(StateMachine sm)
        {
            InitializeComponent();
            map.DragButton = MouseButtons.Left;
            map.MapProvider = GMapProviders.GoogleMap;
            this.sm = sm;
        }
        #region Main Form/Map Region
        private void Main_Form_Load(object sender, EventArgs e)
        {
            // Initialize the ManagementEventWatcher
            InitializePortWatcher();
```

```
            PopulateAvailablePorts();
            Stop.Enabled = false;
        }
        int SatAddress = 1;
        int deviceCount = 2;
        int nodeCount = 1;

        private void map_Load(object sender, EventArgs e)
        {
            map.ShowCenter = false;
            map.MinZoom = 0;     // Minimum zoom level
            map.MaxZoom = 18;    // Maximum zoom level
            map.Zoom = 5;     // Current zoom level
            string relativePath = @"..\..\..\mapCoordinates.txt";
            string filePath = Path.GetFullPath(relativePath);
            //Lists to hold the file contents
            List<string> lines = new List<string>();
            //Reading from the file
            lines = File.ReadAllLines(filePath).ToList();
            //Array to hold list content
            string[] items;
            //loop to extract the list content
            foreach (string line in lines)
            {
                items = line.Split(',');
                location l = new location(items[1], items[2]);
                if (items[0] == "S")
                {
                    sm.addCoordinatesItem("S");
                }
                else if (items[0] == "D")
                {
                    sm.addCoordinatesItem("D");
                }
                else if (items[0] == "N")
                {
                    sm.addCoordinatesItem("N");
                }
                sm.addLatitudeItem(double.Parse(l.latitude));
```

```
                sm.addLongitudeItem(double.Parse(l.longitude));
            }
            //Loop to insert locations on the map
            for (int i = 0; i < sm.getLatitudes().Count; i++)
            {
                //mapping positions
                map.Position = new PointLatLng(sm.latitudes[i], sm.longitudes[i]);
                //Markers
                PointLatLng point = new PointLatLng(sm.latitudes[i],
sm.longitudes[i]);
                //Cover map with Overlay
                map.Overlays.Add(markers);
                if (sm.getCoordinatesType()[i] == "S")
                {
                    GMapMarker marker = new GMarkerGoogle(point,
GMarkerGoogleType.yellow_dot);
                    marker.Tag = SatAddress + ";" + sm.getCoordinatesType()[i];
                    //sm.nodes.Add("Node " + nodeCount);
                    marker.ToolTipText = $"Latitude: {sm.latitudes[i]} ,
\nLongitude:  {sm.longitudes[i]}, \nSatellite: {SatAddress}";
                    var toolTip = new GMapToolTip(marker);
                    toolTip.Fill = new SolidBrush(Color.DarkBlue);
                    toolTip.Foreground = new SolidBrush(Color.White);
                    toolTip.Offset = new System.Drawing.Point(10, -50);
                    toolTip.Stroke = new Pen(new SolidBrush(Color.PaleGoldenrod));
                    marker.ToolTip = toolTip;
                    markers.Markers.Add(marker);
                    SatAddress++;
                }
                else if (sm.getCoordinatesType()[i] == "D")
                {
                    GMapMarker marker = new GMarkerGoogle(point,
GMarkerGoogleType.green_dot);//GMarkerGoogleType.green_dot
                    marker.Tag = deviceCount + ";" + sm.getCoordinatesType()[i];
                    marker.ToolTipText = $"Latitude: {sm.latitudes[i]} ,
\nLongitude:  {sm.longitudes[i]}, \nDevice: {deviceCount}";
                    var toolTip = new GMapToolTip(marker);
                    toolTip.Fill = new SolidBrush(Color.DarkBlue);
                    toolTip.Foreground = new SolidBrush(Color.White);
                    toolTip.Offset = new System.Drawing.Point(10, -50);
                    toolTip.Stroke = new Pen(new SolidBrush(Color.PaleGoldenrod));
```

```
                    marker.ToolTip = toolTip;
                    markers.Markers.Add(marker);
                    deviceCount++;
                }
                else if (sm.getCoordinatesType()[i] == "N")
                {
                    GMapMarker marker = new GMarkerGoogle(point,
GMarkerGoogleType.red_dot);
                    marker.Tag = SatAddress + ";" + sm.getCoordinatesType()[i];

                    marker.ToolTipText = $"Latitude: {sm.latitudes[i]} ,
\nLongitude:  {sm.longitudes[i]}, \nNode: {nodeCount}";
                    var toolTip = new GMapToolTip(marker);
                    toolTip.Fill = new SolidBrush(Color.DarkBlue);
                    toolTip.Foreground = new SolidBrush(Color.White);
                    toolTip.Offset = new System.Drawing.Point(10, -50);
                    toolTip.Stroke = new Pen(new SolidBrush(Color.PaleGoldenrod));
                    marker.ToolTip = toolTip;
                    markers.Markers.Add(marker);
                    nodeCount++;
                }
            }
        }
        private void DrawPolyline(int startIndex, int endIndex)
        {
            PointLatLng startPoint = new PointLatLng(sm.latitudes[startIndex],
sm.longitudes[startIndex]);
            PointLatLng endPoint = new PointLatLng(sm.latitudes[endIndex],
sm.longitudes[endIndex]);

            List<PointLatLng> points = new List<PointLatLng>();
            points.Add(startPoint);
            points.Add(endPoint);
            GMapRoute route = new GMapRoute(points, "Route");
            route.Stroke = new Pen(Color.Red, 2); // Adjust the color and width as
needed
            GMapOverlay polylineOverlay = new GMapOverlay("polyline");
            polylineOverlay.Routes.Add(route);
            map.Overlays.Add(polylineOverlay);
        }
        private int FindNearestNIndex(int deviceIndex)
        {
```

```
            int nearestNIndex = -1;
            double minDistance = double.MaxValue;

            for (int j = 0; j < sm.getLatitudes().Count; j++)
            {
                if (sm.getCoordinatesType()[j] == "N")
                {
                    double deltaX = (sm.getLatitudes()[deviceIndex] -
sm.getLatitudes()[j]) * 111000;
                    double deltaY = (sm.getLongitudes()[deviceIndex] -
sm.getLongitudes()[j]) * 111000;
                    // Calculate distance between points using the provided formula
                    double distance = Math.Sqrt(deltaX * deltaX + deltaY *
deltaY);
                    // Check if distance is less than the current minimum distance
                    if (distance < minDistance)
                    {
                        minDistance = distance;
                        nearestNIndex = j;
                    }
                }
            }
            return nearestNIndex;
        }

        private void DrawPolyline(PointLatLng startPoint, PointLatLng endPoint)
        {
            List<PointLatLng> points = new List<PointLatLng>();
            points.Add(startPoint);
            points.Add(endPoint);

            GMapRoute route = new GMapRoute(points, "Route");
            route.Stroke = new Pen(Color.Red, 2); // Adjust the color and width as
needed
            GMapOverlay polylineOverlay = new GMapOverlay("polyline");
            polylineOverlay.Routes.Add(route);
            map.Overlays.Add(polylineOverlay);
        }
        #endregion
        #region Port Connectivity Check
        private void InitializePortWatcher()
```

```
            {
            // Create a WMI query to monitor for changes in Win32_SerialPort
            WqlEventQuery query = new WqlEventQuery("SELECT * FROM 
Win32_DeviceChangeEvent WHERE EventType = 2 OR EventType = 3");
            // Initialize the ManagementEventWatcher with the query
            portWatcher = new ManagementEventWatcher(query);
            // Set up an event handler for when a new device is connected
            portWatcher.EventArrived += PortWatcher_EventArrived;
            // Start monitoring
            portWatcher.Start();
        }
        private void PortWatcher_EventArrived(object sender, EventArrivedEventArgs 
e)
        {
            int eventType = Convert.ToInt32(e.NewEvent["EventType"]);
            // Check if it's a device arrival (2) or removal (3) event
            if (eventType == 2 || eventType == 3)
            {
                // You can refresh the ComboBox here
                this.Invoke((MethodInvoker)delegate
                {
                    PopulateAvailablePorts();
                });
            }
        }
        private void PopulateAvailablePorts()
        {
            // Clear the existing items in the ComboBox
            AvailablePorts.Items.Clear();
            // Create a ManagementObjectSearcher to query the Win32_SerialPort class
            ManagementObjectSearcher searcher = new 
ManagementObjectSearcher("SELECT * FROM Win32_SerialPort");
            try
            {
                // Execute the query and get the collection of COM ports
                ManagementObjectCollection portCollection = searcher.Get();
                // Loop through each COM port and display its properties
                foreach (ManagementObject obj in portCollection)
                {
                    string description = obj["Description"] as string;
                    if (description.Contains("USB Serial Device"))
```

```
                    {
                        if (obj["PnPDeviceID"].ToString().Contains("PID_0000"))
                        {
                            AvailablePorts.Items.Add(obj["DeviceID"]);
                        }
                    }
                }
            }
            catch (ManagementException ex)
            {
                Console.WriteLine("WMI Query Error: " + ex.Message);
            }
            catch (Exception ex)
            {
                Console.WriteLine("Error: " + ex.Message);
            }

            AvailablePorts.Text = "";
            AvailablePorts.ValueMember = null;
        }
        #endregion
        #region Serial Data Region
        private void serialPort_DataReceived(object sender,
SerialDataReceivedEventArgs e)
        {
            byte[] rx_buffer = new byte[32];
            try
            {
                sm.serialPort.Read(rx_buffer, 0, rx_buffer.Length);
                Console.WriteLine("RX Buff: ");
                for (int i = 0; i < rx_buffer.Length; i++)
                {
                    Console.Write(rx_buffer[i].ToString("X2") + " ");
                }
                Console.WriteLine();
                try
                {
                    ProcessReceivedData(rx_buffer);
                }
                catch (System.IndexOutOfRangeException)
                {
```

```
                    // Reset rx_buffer to an array of 32 bytes with empty data
                    Array.Clear(rx_buffer, 0, rx_buffer.Length);
                }
            }
            catch (IOException ex)
            {
                Console.WriteLine(ex.ToString());
            }
        }

        private void ProcessReceivedData(byte[] rx_buff)
        {
            byte frameID = rx_buff[0]; //Packet Type
            switch ((Frames_ID_t)frameID)
            {
                case Frames_ID_t.ID_FRAME:
                    sm.setIDFrame(rx_buff[1]);
                    break;
                case Frames_ID_t.COORDINATES_FRAME:
                    float lati1 = (float)(rx_buff[1] << 24 | rx_buff[2] << 16 |
rx_buff[3] << 8 | rx_buff[4]);
                    float longi1 = (float)(rx_buff[5] << 24 | rx_buff[6] << 16 |
rx_buff[7] << 8 | rx_buff[8]);
                    double lat1 = lati1 / 1000000;
                    double long1 = longi1 / 1000000;
                    sm.addLongitudeItem(long1);
                    sm.addLatitudeItem(lat1);
                    break;
                case Frames_ID_t.MESSAGE_FRAME:
                    float midDevice = (float)(rx_buff[1]);
                    float destinationDevice = (float)(rx_buff[2]);
                    float sourceDevice = (float)(rx_buff[3]);
                    int messageStartIndex = 4;
                    string message = Encoding.UTF8.GetString(rx_buff,
messageStartIndex, rx_buff.Length - messageStartIndex);
                    string relativePath = @"..\..\..\messages\Received\";

                  string fileName = $"Chat {sm.getIDFrame()}_{sourceDevice}.txt";
                    //string fileName = $"Destination Device {sm.getIDFrame()}.txt";
                    string filePath = Path.GetFullPath(relativePath + fileName);
                    StreamWriter txt = new StreamWriter(filePath, true);
```

```
                    string content =
$"{DateTime.Now.ToString("yyyy/MM/dd_HH:mm:ss")} : {message}\n";
                    txt.WriteLine(content);
                    txt.Close();
                    string[] fileContents = File.ReadAllLines(filePath);

                    string concatenatedContents = string.Join(Environment.NewLine,
fileContents); // Join the lines into a single string
                    sm.addNodeMessageItem(concatenatedContents);
                    sm.latitudes.Clear();
                    sm.longitudes.Clear();
                    break;
                case Frames_ID_t.CMD_FRAME:
                    break;
                case Frames_ID_t.DEBUG_FRAME:
                    Console.WriteLine("RX Buff 1: " + rx_buff[1]);
                    Console.WriteLine("RX Buff 2: " + rx_buff[2]);
                    break;
            }
        }
        #endregion
        #region Map Functions
        private void map_OnMarkerClick(GMapMarker item, MouseEventArgs e)
        {
            if (!sm.getIsDetailsOpened())
            {
                if (sm.getStartClicked() && sm.getIsConnected())
                {
                    string[] tagSplit = item.Tag.ToString().Split(';');
                    string nodeID = tagSplit[0];
                    double lat = item.Position.Lat;
                    double lng = item.Position.Lng;
                    // Check if the coordinateType is "D"
                    if (tagSplit[1] == "D" && ((byte)sm.getIDFrame()) !=
byte.Parse(nodeID))
                    {
                        sm.setIsDetailsOpened(true);
                        Details dt = new Details(sm, int.Parse(nodeID), lat, lng);
                        dt.Show();
                    }
```

```
                }
            }
        }
        private void map_OnMarkerEnter(GMapMarker item)
        {
            Cursor.Current = Cursors.Hand;
        }
        private void map_OnMarkerLeave(GMapMarker item)
        {
            Cursor.Current = Cursors.Default;
        }
        #endregion
        #region Error_Region
        private async Task ShowErrorMessageAsync(string message)
        {
            await Task.Yield();
            MessageBox.Show(message, "Error", MessageBoxButtons.OK,
MessageBoxIcon.Error);
        }
        private bool isErrorDisplayed = false;
        public void error_Handler(string message)
        {
            if (!isErrorDisplayed == true)
            {
                MessageBox.Show("Port Disconnected", message,
MessageBoxButtons.OK, MessageBoxIcon.Error);
                sm.setIsConnected(false);
                sm.serialPort.Close();
                this.Refresh();
                isErrorDisplayed = false;
            }
            else
            {
                sm.setIsConnected(false);
                sm.serialPort.Close();
                isErrorDisplayed = false;

                this.Refresh();
            }
        }
        #endregion
```

```
        #region Start/Stop Buttons
        private async void Start_Click(object sender, EventArgs e)
        {
            sm.setStartClicked(true);
            sm.setIsConnected(true);
            string sp = AvailablePorts.SelectedItem as string;
            Start.Enabled = false;
            Stop.Enabled = true;
            if (sm.serialPort != null && sm.serialPort.IsOpen == true)
            {
                sm.serialPort.DataReceived -= serialPort_DataReceived;
                sm.setContinueReading(false);
                sm.serialPort.Close();
                sm.setIsConnected(false);
            }
            try
            {
                sm.serialPort = new SerialPort(sp, 115200);
                // Set the data received event handler
                sm.serialPort.DataReceived += serialPort_DataReceived;
                sm.serialPort.Open();
                sm.setIsConnected(true);
                sm.setTimerState(true);
                byte[] request_header = new byte[1];
                request_header[0] = (byte)Frames_ID_t.ID_FRAME;
                sm.serialPort.Write(request_header, 0, 1);
            }
            catch (Exception ex)
            {
                await ShowErrorMessageAsync(ex.Message);
                Start.Enabled = true;
                Stop.Enabled = false;
                AvailablePorts.Enabled = true;
                AvailablePorts.Text = "";
            }
        }
        private void Stop_Click(object sender, EventArgs e)
        {
            if (sm.getStartClicked() == true)
            {
```

```
                Start.Enabled = true;
                Stop.Enabled = false;
            }
            if (sm.getIsConnected() == true)
            {
                sm.setIsConnected(false);
                sm.setContinueReading(false);
                sm.serialPort.DiscardInBuffer();
                data_request_timer.Stop();
                sm.setTimerState(false);
                AvailablePorts.Enabled = true;
            }
        }
        #endregion
    }
}
```

Details Form:

- **Details:** initializes the components of the form and gets the attributes in the constructor.
- **Details_Load:** this function fill each of the components with the corresponding data.
- **Details_FormClosed:** raises a flag that the form was closed so another form could be opened.
- **msgSendBtn_Click:** gets triggered when the send button is clicked.
  - It calls "ControllingData" function.
  - Displays the content of the files in the directory in the message box.

- **ControllingData:** This function is responsible for sending data from GUI to STM. It creates a frame to be filled with all the data required to be send.

  The frame contains:

  - The Message Frame ID.
  - Packet Length.
  - GUI ID.
  - Source Device ID.
  - Destination Node ID.

- Destination Device ID.

- Payload Length (got from the elected item from the comoBox called "contentBox").

- The Message.

It then sends this frame through the serial port and adds the message to the created text file of the node.

- **refreshSentMessageBox:** This function automatically refreshes the content of the text boxes "display" by retrieving the corresponding text files from the "sent" folder according to the nodes that were changed.
- **refreshReceivedMessageBox:** This function automatically refreshes the content of the text boxes "messagesTxt" by retrieving the corresponding text files from the "Received" folder according to the nodes that were changed.

Details Code:

```
namespace CNA_GUI
{
    public partial class Details : Form
    {
        StateMachine sm;
        public Details(StateMachine sm)
        {
            InitializeComponent();
            this.sm = sm;
        }
        public Details(StateMachine sm, int nodeID, double latitude, double longitude)
        {
            InitializeComponent();
            Details_Load(sm, nodeID, latitude, longitude);
        }
        #region Details Load Region
        int ndID;
        private void Details_Load(StateMachine sm, int nodeID, double latitude, double
longitude)
        {
            if (sm.getIsDetailsOpened() == true)
```

```
            {
                try
                {
                    this.sm = sm;
                    ndID = nodeID;
                    this.nodeIDTxt.Text = nodeID.ToString();
                    this.latitudeTxt.Text = latitude.ToString();
                    this.longitudeTxt.Text = longitude.ToString();
                    received_message_timer.Start();
                    string relativePath = @"..\..\..\messages\Received\";
                    string fileName = $"Destination Device {nodeID}.txt";
                    string filePath = Path.GetFullPath(relativePath + fileName);
                    string[] fileContents = File.ReadAllLines(filePath);
                    string concatenatedContents = string.Join(Environment.NewLine,
fileContents); // Join the lines into a single string
                    messageTxt.Text = concatenatedContents;
//////////////////////////////////////////////////////////////////////////////////
                    string relativePath1 = @"..\..\..\messages\Sent\";
                    string fileName1 = $"Source Device {ndID}.txt";
                    string filePath1 = Path.GetFullPath(relativePath1 + fileName1);
                    string[] fileContents1 = File.ReadAllLines(filePath1);
                    List<string> selectedNodeMessages = new List<string>();
                    foreach (string line in fileContents1)
                    {
                        if (line.StartsWith($"To: {ndID}"))
                        {
                            int commaIndex = line.IndexOf(",");
                            if (commaIndex != -1 && commaIndex + 2 < line.Length)
                            {
                                string messagePart = line.Substring(commaIndex + 2);
                                selectedNodeMessages.Add(messagePart);
                            }
                        }
                    }
                    string concatenatedContents1 = string.Join(Environment.NewLine,
selectedNodeMessages);
                    display.Text = concatenatedContents1;
                }
                catch (System.IO.FileNotFoundException)
                {
                    this.sm = sm;
```

```
                ndID = nodeID;
                this.nodeIDTxt.Text = nodeID.ToString();
                this.latitudeTxt.Text = latitude.ToString();
                this.longitudeTxt.Text = longitude.ToString();
                string relativePath1 = @"..\..\..\messages\Sent\";
                string fileName1 = $"Source Device {ndID}.txt";
                string filePath1 = Path.GetFullPath(relativePath1 + fileName1);
                StreamWriter txt = new StreamWriter(filePath1, true);
                txt.Close();
            }
        }
    }
    private void Details_FormClosed(object sender, FormClosedEventArgs e)
    {
        if (sm.getIsDetailsOpened() == true)
        {
            sm.setIsDetailsOpened(false);
            received_message_timer.Stop();
        }
    }
    #endregion
    #region Message Control Region
    private void msgSendBtn_Click(object sender, EventArgs e)
    {
        sm.setUserMessage(contentBox.SelectedItem.ToString());

        Invalidate();
        Refresh();
        ControllingData();
        refreshSentMessageBox();
        Thread.Sleep(100);
        //refreshReceivedMessageBox();
    }
    public void ControllingData()
    {
        try
        {
            string message = sm.getUserMessage();
            byte[] ControlFrame = new byte[32];
            ControlFrame[0] = (byte)sm.GetMESSAGE_FRAME();
            //ControlFrame[1] = sm.getIDFrame(); //Source ID
```

```
                ControlFrame[1] = sm.getIDFrame(); //Source ID
                ControlFrame[2] = (byte)ndID; //Destination Device
                // Convert the string to bytes and copy them to ControlFrame
                byte[] messageBytes = Encoding.UTF8.GetBytes(message);
                Array.Copy(messageBytes, 0, ControlFrame, 3, messageBytes.Length);
                sm.serialPort.Write(ControlFrame, 0, ControlFrame.Length);
                contentBox.SelectedIndex = -1; // Clears the selected item
/////////////////////////////////////////////////////////////////////////////////////
                string relativePath = @"..\..\..\messages\Sent\";
                string fileName = $"Source Device {ndID}.txt";
                string filePath = Path.GetFullPath(relativePath + fileName);
                StreamWriter txt = new StreamWriter(filePath, true);
                string content = $"To: {ndID}, Message: {sm.getUserMessage()}";
                txt.WriteLine(content);
                txt.Close();
                string[] fileContents = File.ReadAllLines(filePath);


                string concatenatedContents = string.Join(Environment.NewLine,
fileContents); // Join the lines into a single string
                sm.addNodeMessageItem(concatenatedContents);
                this.Invalidate();
                this.Refresh();
            }
            catch (System.NullReferenceException)
            {
                sm.setInvalidOperationExceptionCatched(true);
                this.Close();
            }
            catch (System.InvalidOperationException)
            {
                sm.setInvalidOperationExceptionCatched(true);
                this.Close();
            }
        }
        public void refreshSentMessageBox()
        {
            try
            {
                string relativePath1 = @"..\..\..\messages\Sent\";
                string fileName1 = $"Source Device {ndID}.txt";
                string filePath1 = Path.GetFullPath(relativePath1 + fileName1);
```

```
                string[] fileContents1 = File.ReadAllLines(filePath1);
                List<string> selectedNodeMessages = new List<string>();
                foreach (string line in fileContents1)
                {
                    if (line.StartsWith($"To: {ndID}"))
                    {
                        int commaIndex = line.IndexOf(",");
                        if (commaIndex != -1 && commaIndex + 2 < line.Length)
                        {
                            string messagePart = line.Substring(commaIndex + 2);
                            selectedNodeMessages.Add(messagePart);
                        }
                    }
                }
                string concatenatedContents1 = string.Join(Environment.NewLine,
selectedNodeMessages);
                display.Text = concatenatedContents1;
                Invalidate();
                Refresh();
            }
            catch (System.IO.FileNotFoundException)
            {
                string relativePath1 = @"..\..\..\messages\Sent\";
                string fileName1 = $"Source Device {ndID}.txt";
                string filePath1 = Path.GetFullPath(relativePath1 + fileName1);
                StreamWriter txt = new StreamWriter(filePath1, true);
                txt.Close();
            }
        }
        private long lastReadPosition = 0; // Add this variable at the class level
        public void refreshReceivedMessageBox()
        {
            try
            {
                string relativePath = @"..\..\..\messages\Received\";
                //string fileName = $"Destination Device {ndID}.txt";
                string fileName = $"Destination Device {sm.getIDFrame()}.txt";
                string filePath = Path.GetFullPath(relativePath + fileName);
                string[] fileContents = File.ReadAllLines(filePath);
                /*
```

```
                                using (FileStream fs = new FileStream(filePath,
FileMode.Open, FileAccess.Read, FileShare.ReadWrite))
                                {
                                    fs.Seek(lastReadPosition, SeekOrigin.Begin);
                                    using (StreamReader sr = new StreamReader(fs))
                                    {
                                        while (!sr.EndOfStream)
                                        {
                                            string line = sr.ReadLine();
                                            messageTxt.AppendText(line +
Environment.NewLine);
                                        }
                                        lastReadPosition = fs.Position;
                                    }
                                }
                */
                using (FileStream fs = new FileStream(filePath, FileMode.Open,
FileAccess.Read, FileShare.ReadWrite))
                {
                    fs.Seek(lastReadPosition, SeekOrigin.Begin);
                    using (StreamReader sr = new StreamReader(fs))
                    {
                        bool isFirstMessage = true;
                        while (!sr.EndOfStream)
                        {
                            string line = sr.ReadLine();
                            if (!isFirstMessage)
                            {
                                messageTxt.AppendText(Environment.NewLine);
                            }
                            messageTxt.AppendText(line + "\n");
                            isFirstMessage = false;
                        }
                        lastReadPosition = fs.Position;
                    }
                }
                Invalidate();
                Refresh();
            }
            catch (System.IO.FileNotFoundException)
            {
                string relativePath1 = @"..\..\..\messages\Received\";
```

```
                string fileName1 = $"Destination Device {ndID}.txt";
                string filePath1 = Path.GetFullPath(relativePath1 + fileName1);
                StreamWriter txt = new StreamWriter(filePath1, true);
                txt.Close();
            }
        }
        #endregion
        private void received_message_timer_Tick(object sender, EventArgs e)
        {
            refreshReceivedMessageBox();
        }
    }
}
```

StateMachine Class:

This class contains all the getters and setters in the application, and it is used to store the data to easily transfer it from one form to the other.

```
namespace CNA_GUI
{
    public class StateMachine
    {
        #region Variables

        public StateMachine() { }

        public SerialPort serialPort;

        public static bool IsDetailsOpened = false;
        public static bool isConnected = false;
        public double latitude, longitude;
        public byte[] coordinatesFrame;
        public List<double> latitudes = new List<double>();
        public List<double> longitudes = new List<double>();
        public List<string> coordinatesType = new List<string>();
        public List<int> nodeIDs = new List<int>();
        public List<string> fileMessage = new List<string>();
        public List<string> nodes = new List<string>();
        public List<string> tempNodes = new List<string>();
        public double dstNodesLength;
```

```
        public string userMsg;
        public byte IDFrame;
        //public int FRAME_LEN = 1024;
        public int FRAME_LEN = 1024;

        public double sourceNode, sourceDevice, destinationNode, destinationDevice,
gatewayAddress, payloadLength, pktType;
        byte[] packet;
        public double message;

        private static bool continueReading = true;

        public bool timerState;
        public static bool StartClicked = false;
        public static bool invalidOperationExceptionCatched = false;
        public enum Frames_ID_t
        {
            COORDINATES_FRAME = 0X50,
            MESSAGE_FRAME = 0x60,
            CMD_FRAME = 0x70,
            DEBUG_FRAME = 0X80
        }
        #endregion

        #region Getters

        public string getUserMessage()
        {
            return userMsg;
        }

        public bool getIsDetailsOpened()
        {
            return IsDetailsOpened;
        }

        public bool getIsConnected()
        {
            return isConnected;
        }
```

```
        public List<double> getLatitudes()
        {
            return latitudes;
        }

        public List<double> getLongitudes()
        {
            return longitudes;
        }

        public byte getIDFrame()
        {
            return IDFrame;
        }

        public bool getStartClicked()
        {
            return StartClicked;
        }

        public List<int> getNodeID()
        {
            return nodeIDs;
        }

        public List<string> getCoordinatesType()
        {
            return coordinatesType;
        }

        public Frames_ID_t GetMESSAGE_FRAME()
        {
            return Frames_ID_t.MESSAGE_FRAME;
        }

        #endregion

        //#################################### Setters
################################

        #region Setters
```

```
        public void setUserMessage(string message)
        {
            userMsg = message;
        }
        public void setDstNodesLength(double dnl)
        {
            dstNodesLength = dnl;
        }
        public void setIsDetailsOpened(bool isDO)
        {
            IsDetailsOpened = isDO;
        }

        public void setIsConnected(bool isC)
        {
            isConnected = isC;
        }

        public void setSourceDevice(double sourceDevice)
        {
            this.sourceDevice = sourceDevice;
        }

        public void setDestinationDevice(double destinationDevice)
        {
            this.destinationDevice = destinationDevice;
        }

        public void setPayloadLength(double payloadLength)
        {
            this.payloadLength = payloadLength;
        }

        public void setTimerState(bool state)
        {
            timerState = state;
        }

        public void setStartClicked(bool state)
        {
```

```
            StartClicked = state;
        }

        public void setInvalidOperationExceptionCatched(bool iOEC)
        {
            invalidOperationExceptionCatched = iOEC;
        }

        public void setContinueReading(bool con)
        {
            continueReading = con;
        }

        #endregion

        #region Additional Functions

        public void addLatitudeItem(double item)
        {
            latitudes.Add(item);
        }

        public void addLongitudeItem(double item)
        {
            longitudes.Add(item);
        }

        public void addCoordinatesItem(string item)
        {
            coordinatesType.Add(item);
        }

        public void setIDFrame(byte id)
        {
            IDFrame = id;
        }

        public void addNodeIDItem(int item)
        {
            nodeIDs.Add(item);
        }
```

```
        public void addNodeMessageItem(string item)
        {
            fileMessage.Add(item);
        }

        #endregion
    }
}
```

16.4.11 GUI Code

Link to download GUI: https://aecenar.com/index.php/downloads/send/16-ics/1492-cna-gui-v19-010124-fully-functional

16.5 Names and Approxiations

Gateway Ground Station: Gateway, where all messages of the system are displayed

16.6 User Manual

16.6.1 GUI Manual:

The programmer in charge of coding the GUI employed the Visual Studio installer and utilized the C# language. To download the program, visit this link: https://visualstudio.microsoft.com/downloads/ .

After writing the code, run the application by clicking the "Start" button. Before doing so, ensure that the solution configuration is set to "Release." Upon running the application, the main window displayed is depicted in Figure 6-1 below.

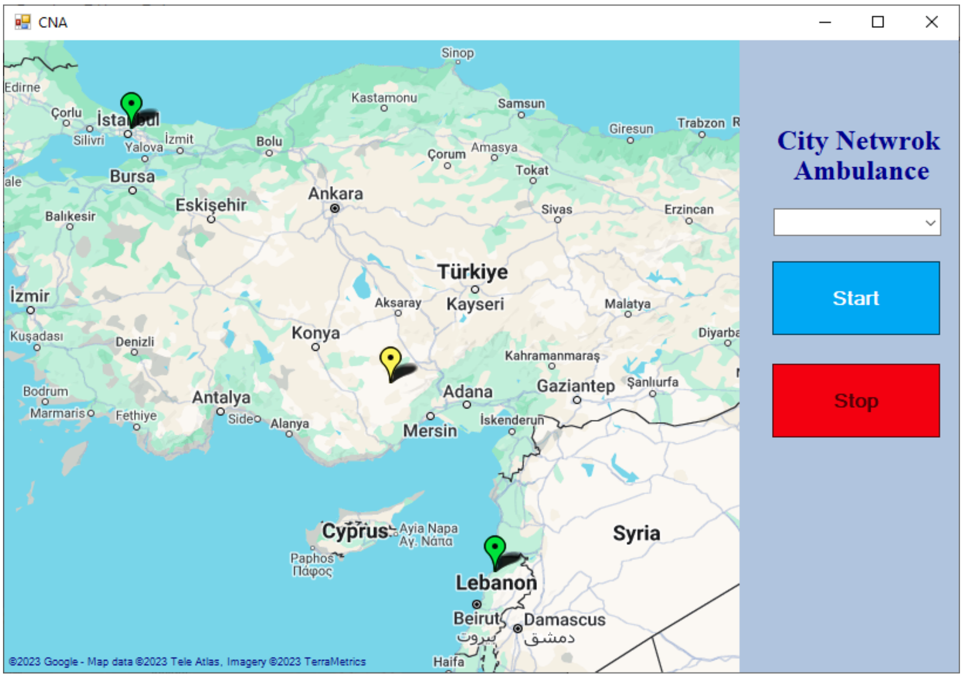


Figure 6-1- Main MAP

we choose the port that is connected to the PC (the combo box will filter and only show the Transceiver ports connected) as shown in the figure 6-2 below.

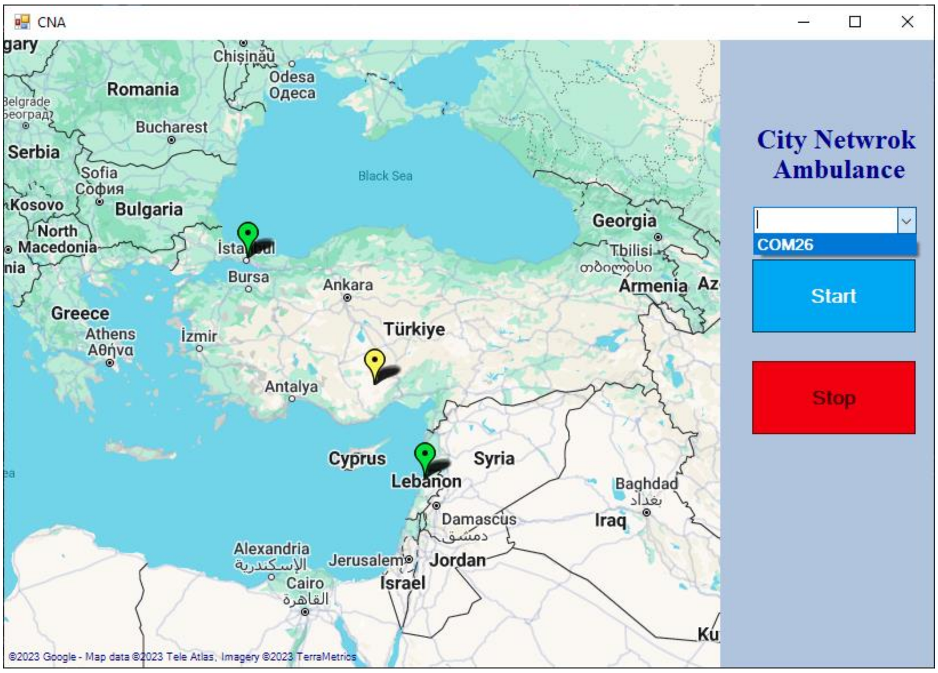


Figure 6-2- Main MAP with Port

When we press start, we will be able to click on the markers that are of type "Device". When we hover over the marker, the marker description will pop up as shown in the figure 6-3.

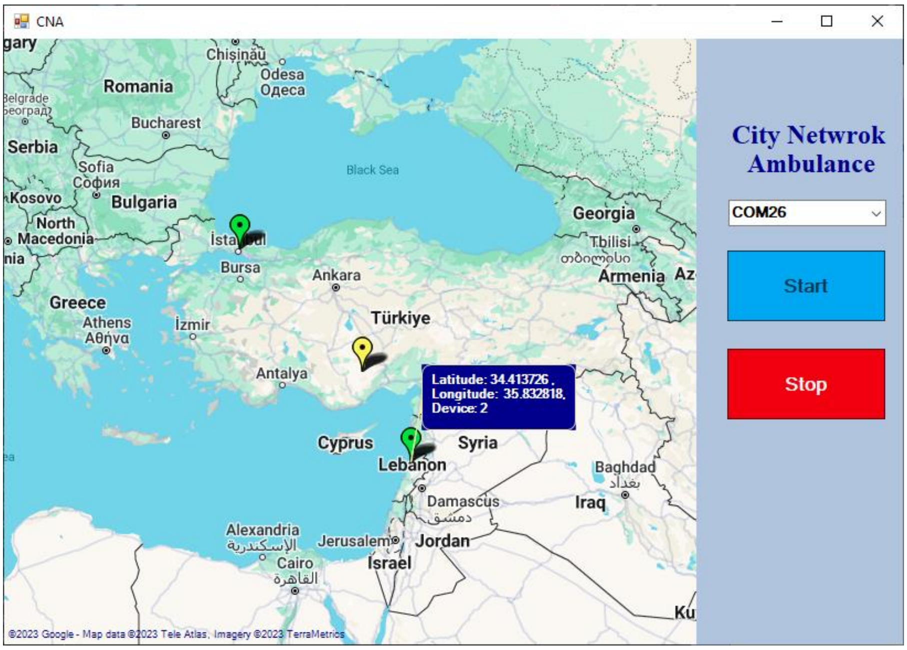


Figure 6-3- Marker Description

When we click on the marker, the details form will open and we can send messages as shown in the figure 6-4 below.

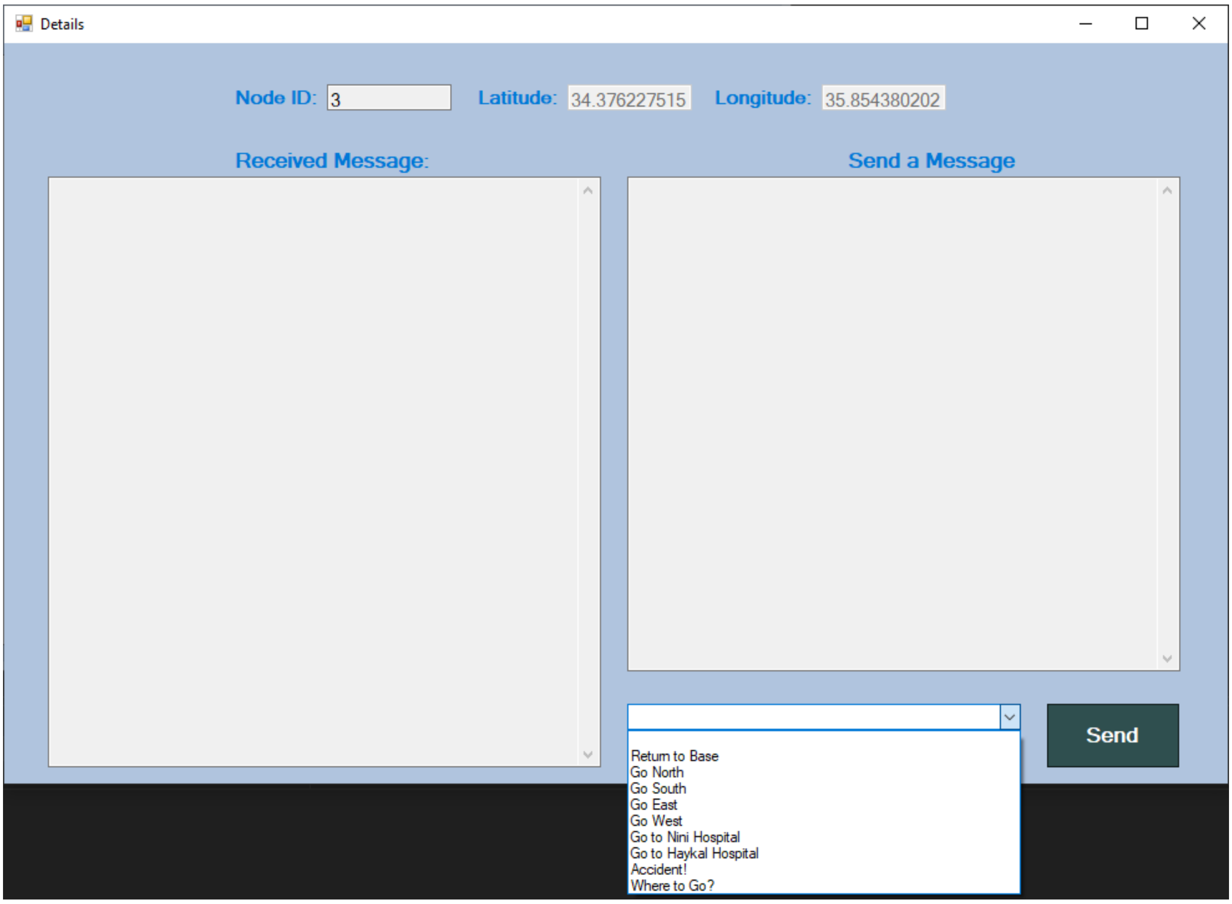


Figure 6-4- Details form and messages

After we send the message, it will be displayed in the "Send a Message" section as shown in the figure below.

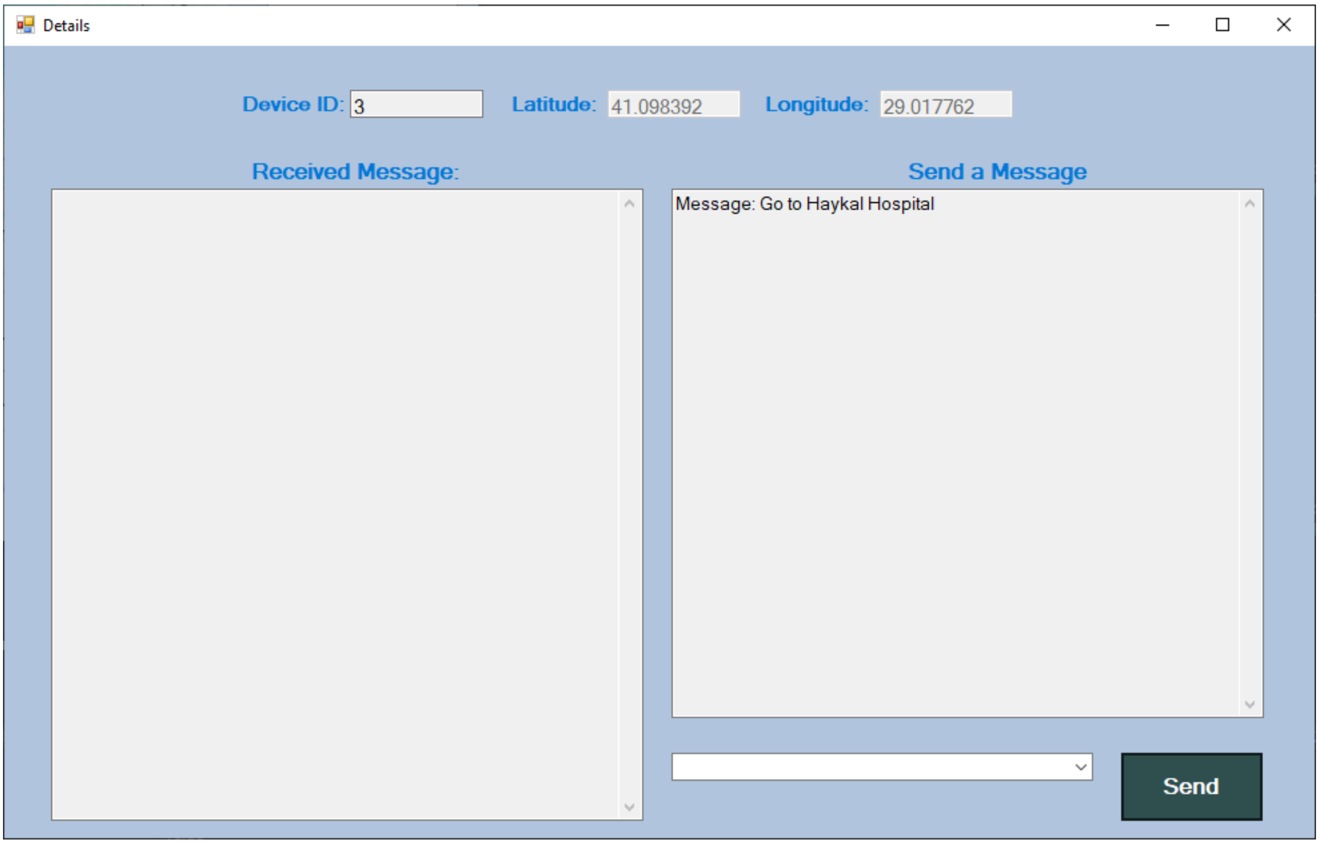


Figure 6-5- Details Sending Message

- **<u>STM32 Manual:</u>**

If we want to change the code configuration for each device type (GUI, NODE, SAT, or DEV), it can be done using the STM32cubeIDE. All we need to do is to change the definition of the device from the "configuration.h" class. We can see in the figure below where we can find the configuration class. We open the project then open the folder "Core", after that we open the "Inc" folder and open the "configuration.h" class.

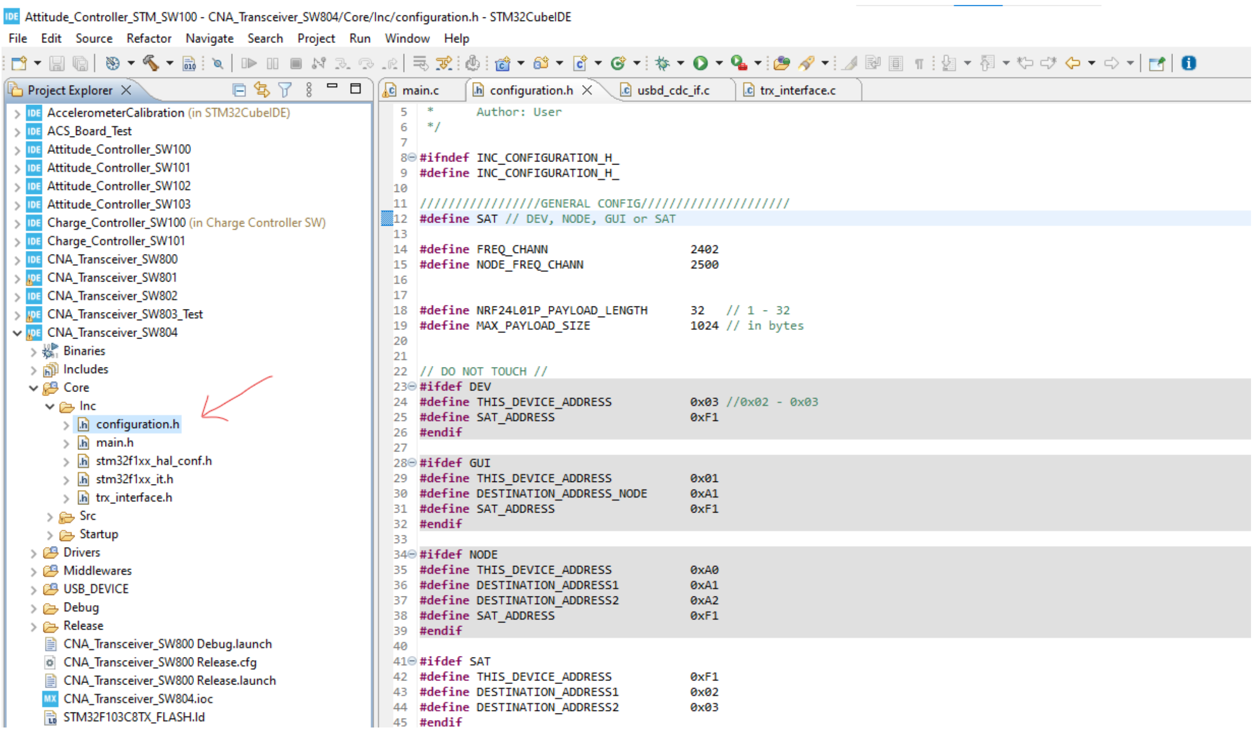


Figure 6-6- configuration.h class

After opening the "configuration.h" class, we only change the define type of the transceiver. We can see in the figure below where we should change the type and we can see in the comment next to it what are the different transceiver types we can name (DEV, NODE, GUI or SAT) where it is marked by the red arrow. If the transceiver type is "DEV", we should change the variable "THIS_DEVICE_ADDRESS" based also on the comment written next to the code line where it is marked by the blue arrow.

P.S: The device name should be CAPITAL case or else it would not work.

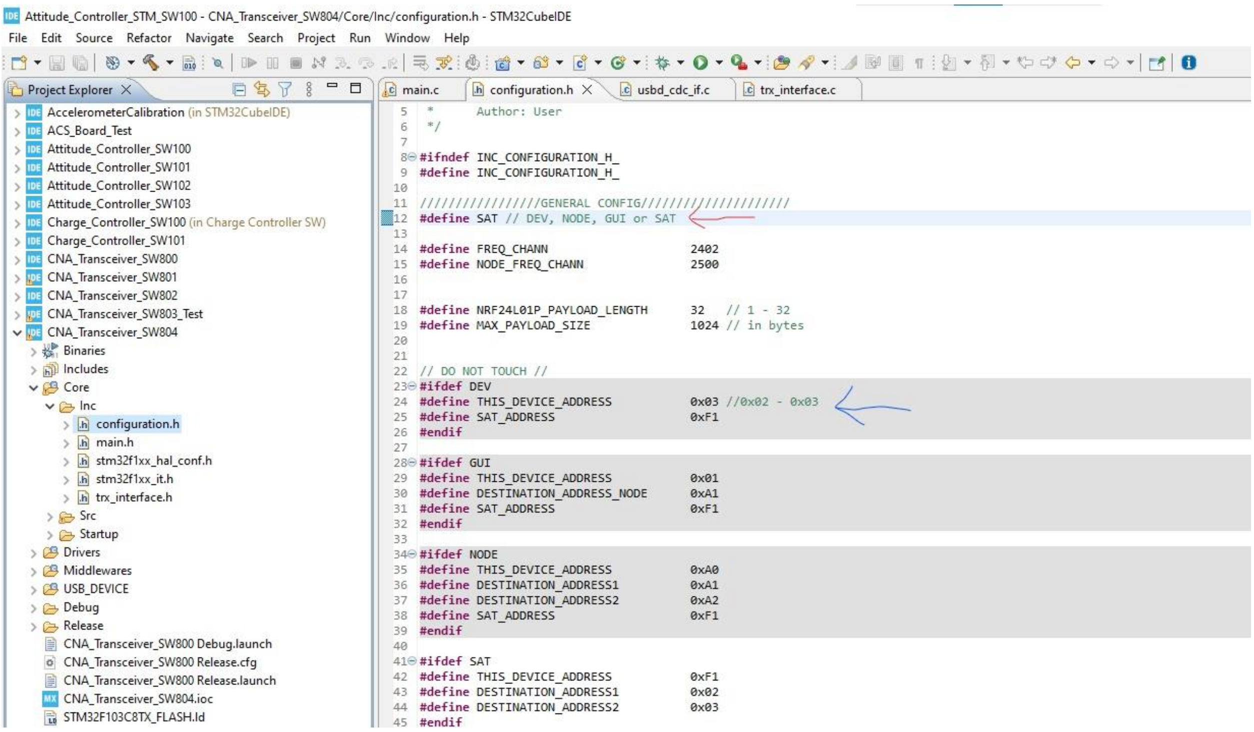


Figure 6-7- Device Type and DEV address

16.7 STM32 IDE configuration and Code Upload

First, we create a new STM32 Project

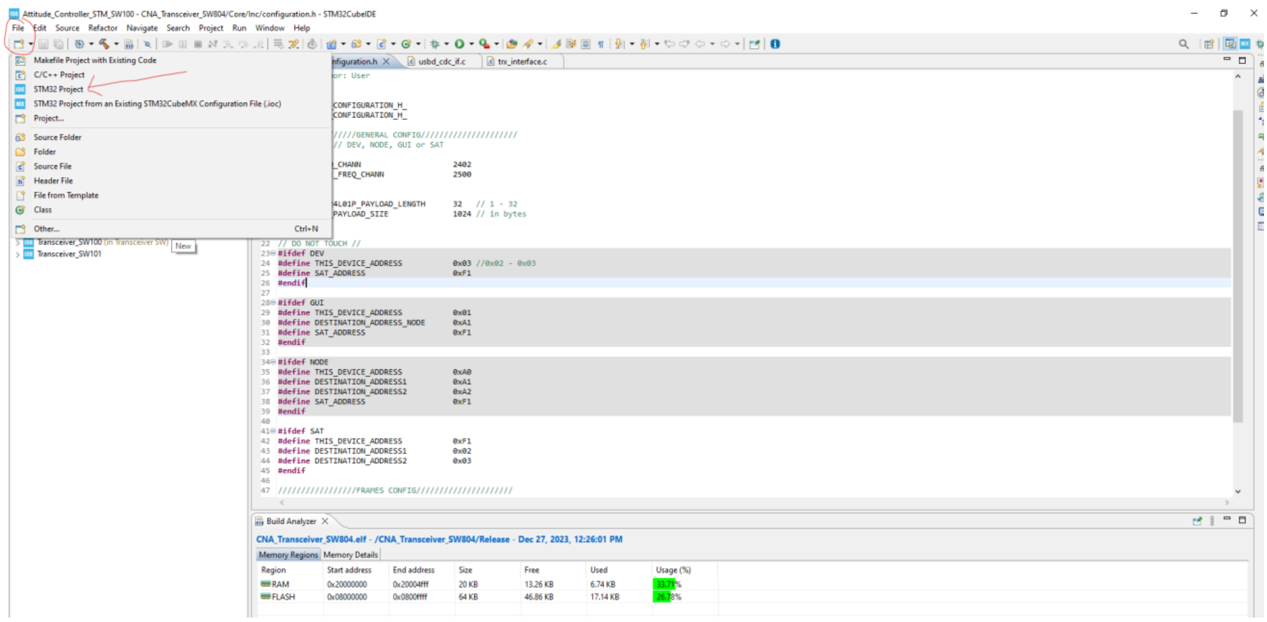


Figure 7-1- Create new STM32 Project

Next, we choose our STM model and click next button

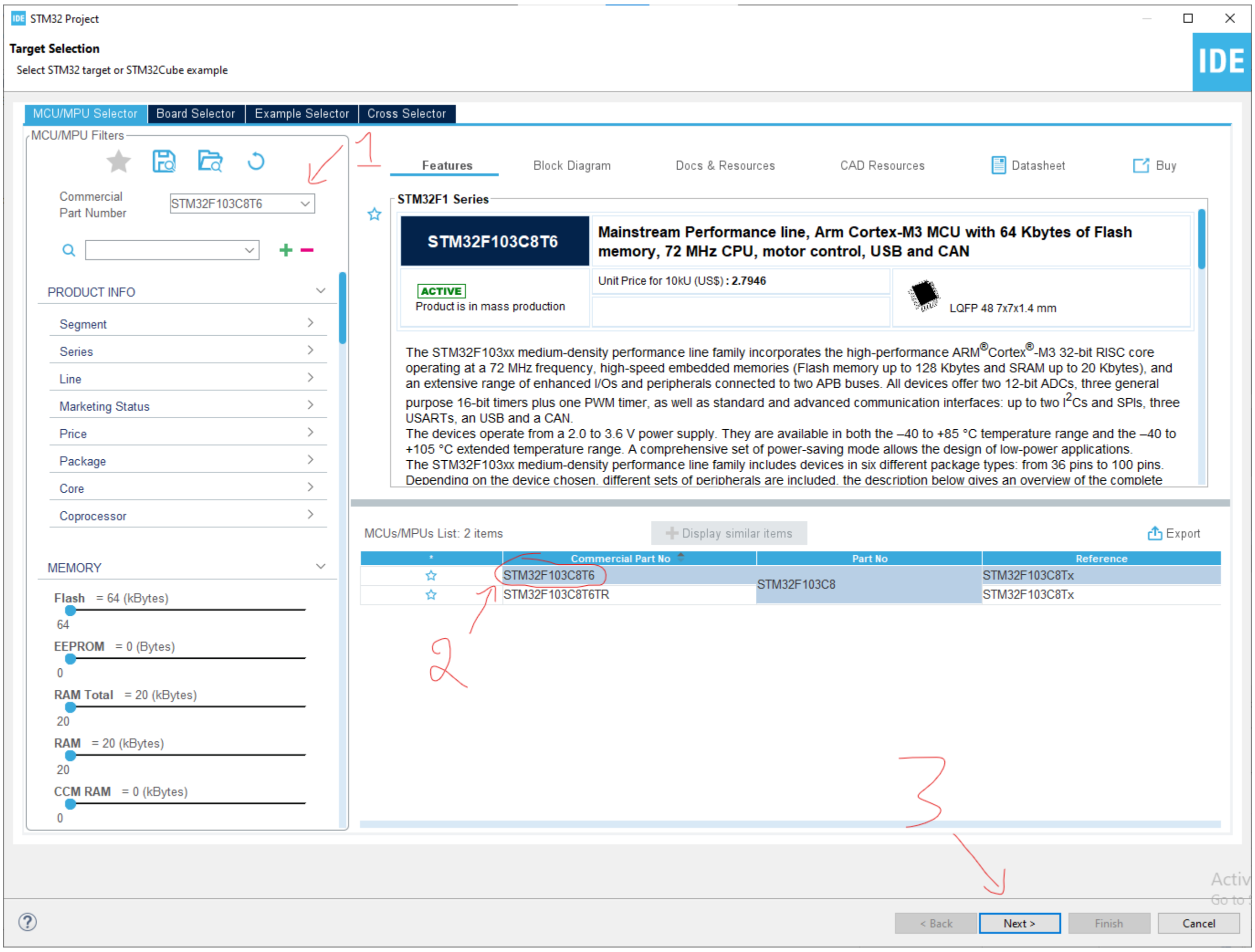


Figure 7-2- Choose STM model

Lastly, we give our project a name and click Finish

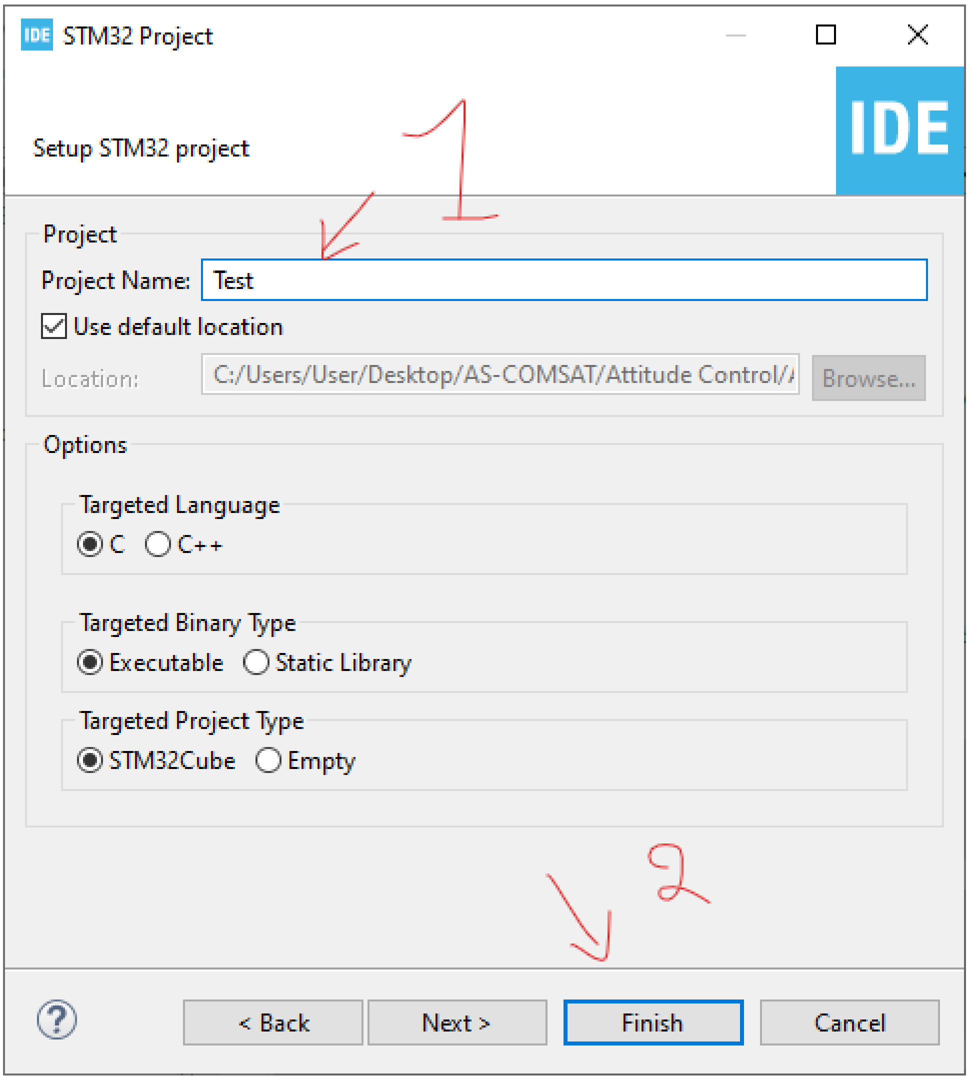


Figure 7-3- Project Name and Creation

When we reach this page, first we need to configure the RCC configuration as shown in the figure below.

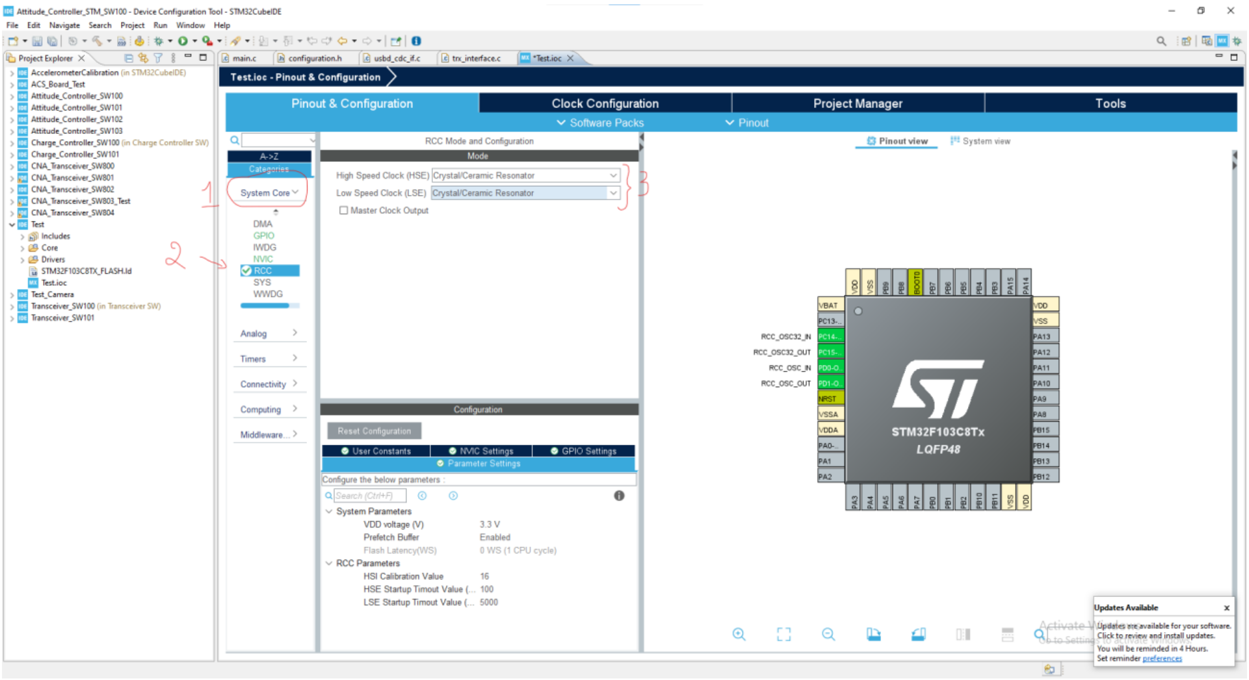


Figure 7-4- RCC configuration

Next, we need to configure the USB configuration as shown in figure 7-5 and 7-6 below

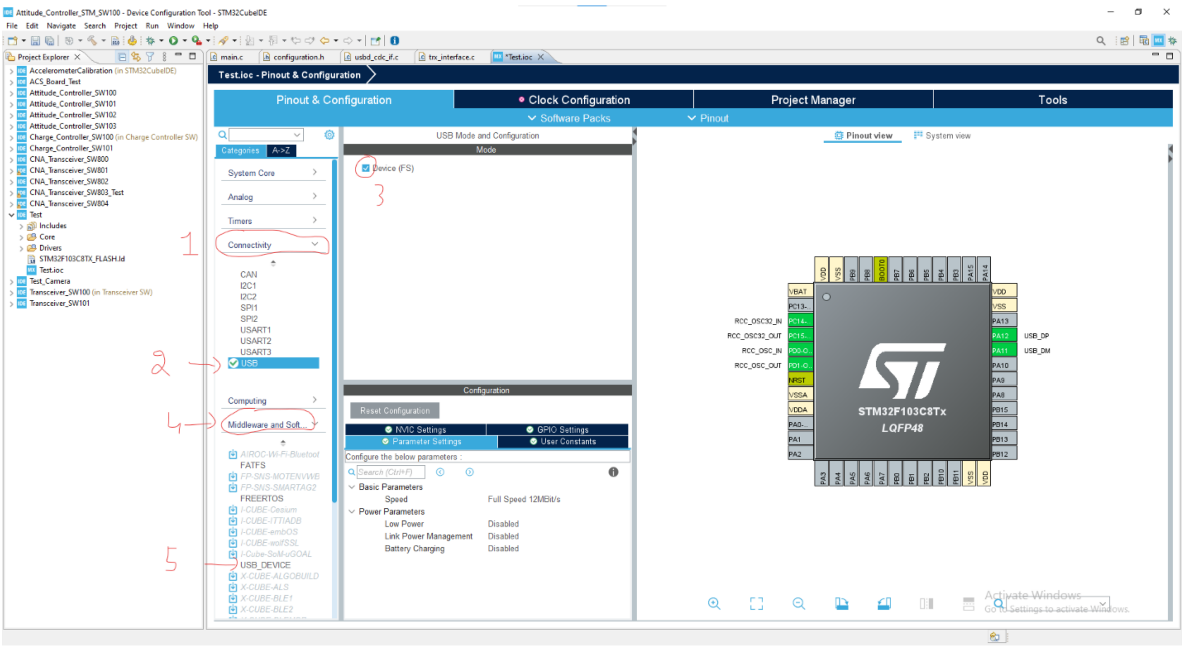


Figure 7-5- USB Configuration

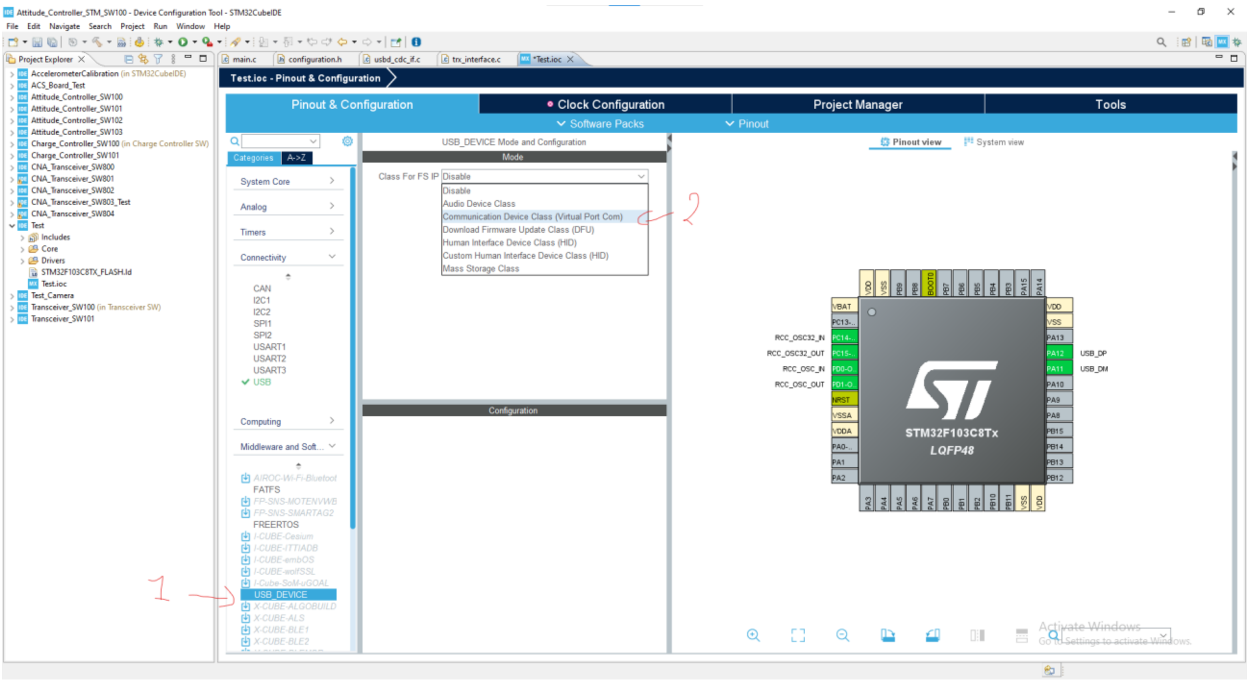


Figure 7-6- USB Configuration

Lastly, we need to change the clock configuration and put it to 72 MHz as shown in the figure below

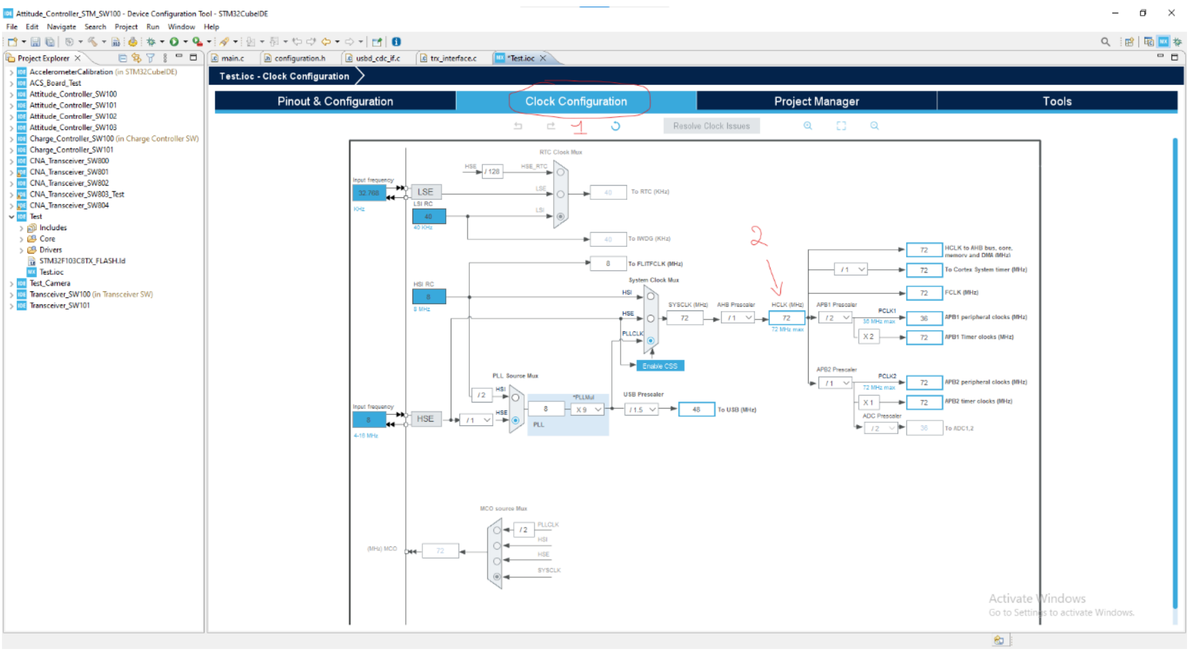


Figure 7-7- Clock Configuration

After finishing with the configuration, we need to “release” the code.

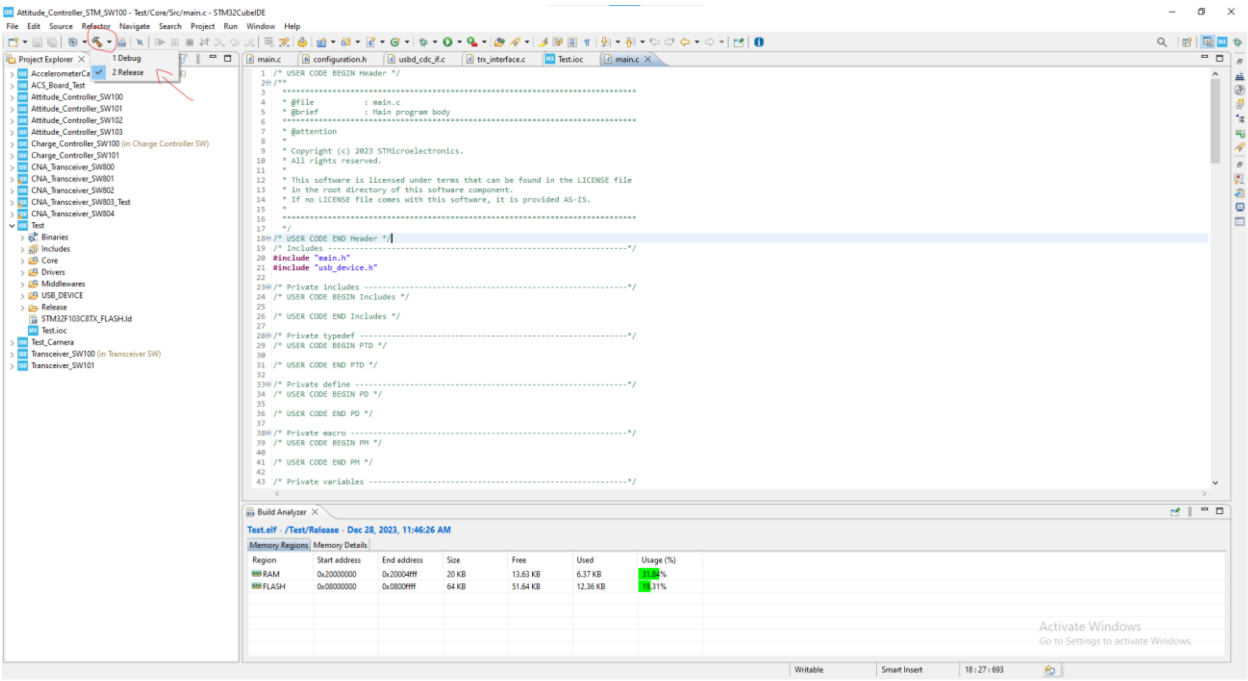


Figure 7-8- Code Release

After releasing the code, we open “STM32CubeProgrammer” and then we need to choose the file that we have released.

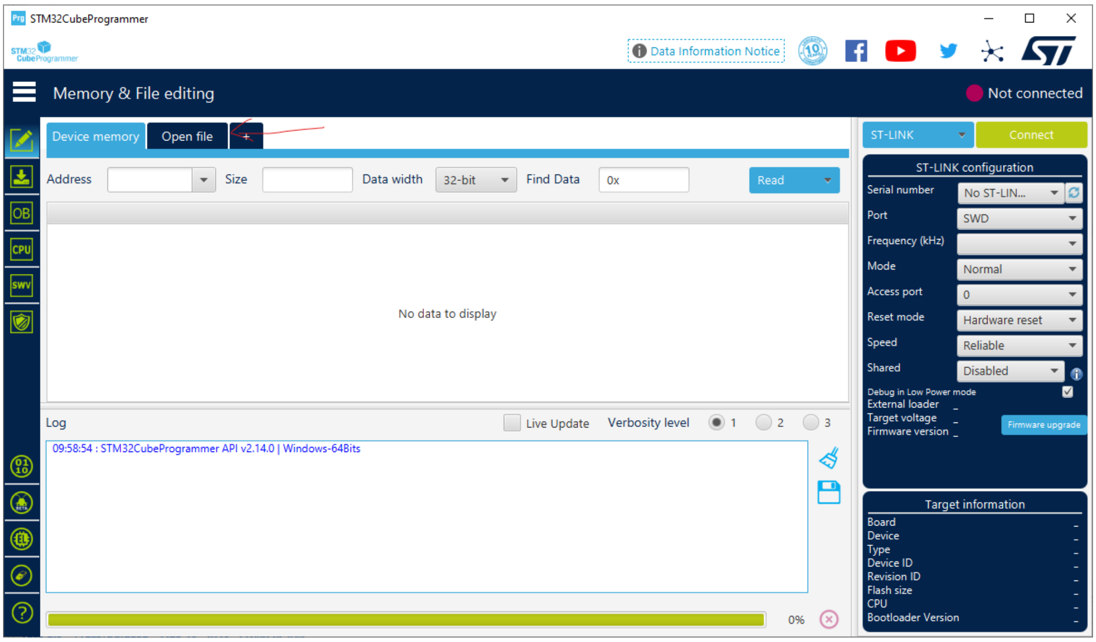


Figure 7-9- Choosing file from Programmer

Then we choose the code elf file from where we saved it in the “Release” folder. Here is a sample directory path of where the file is saved: Test\Release\fileName.elf

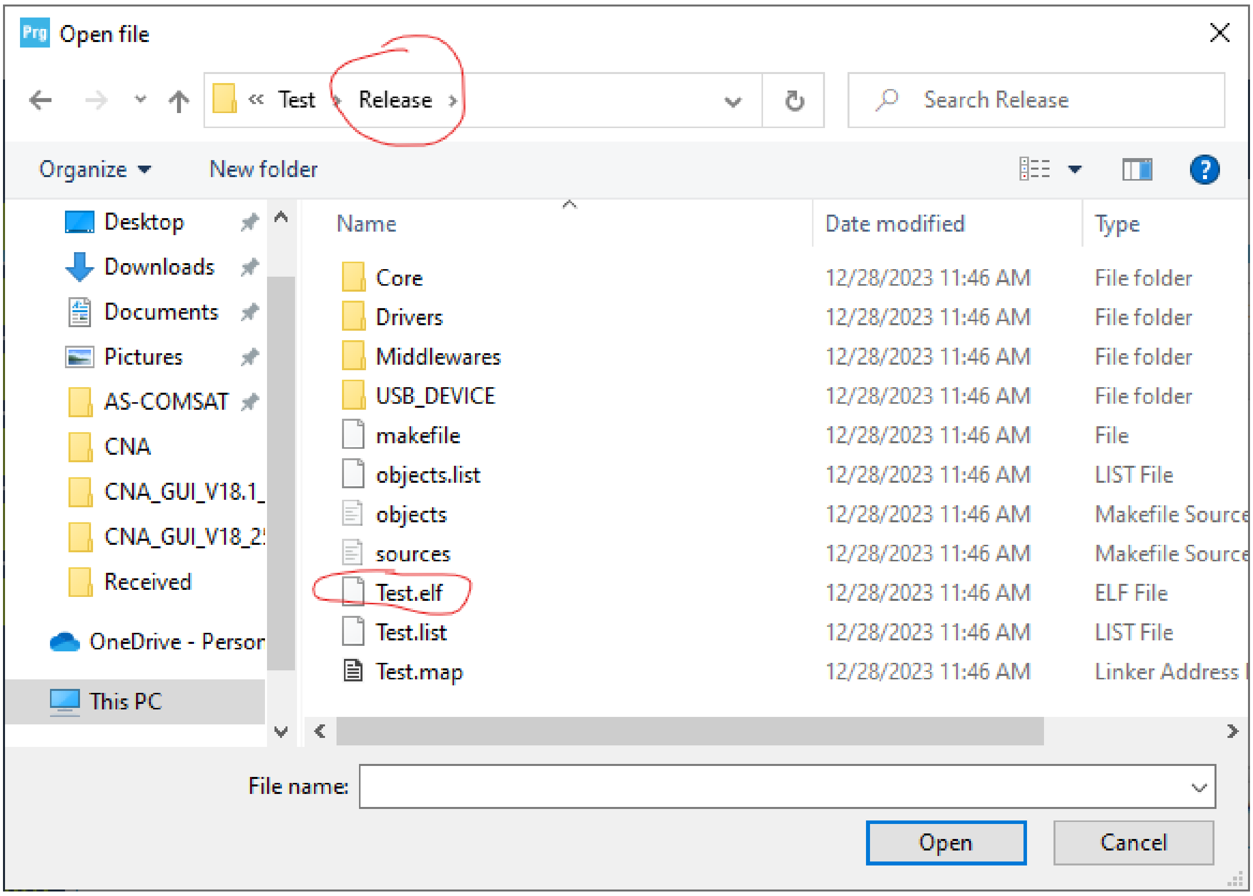


Figure 7-10- Choosing elf file for Programmer

We need to make sure to connect the "ST-Link" USB programmer to the PC without connecting a USB serial port.

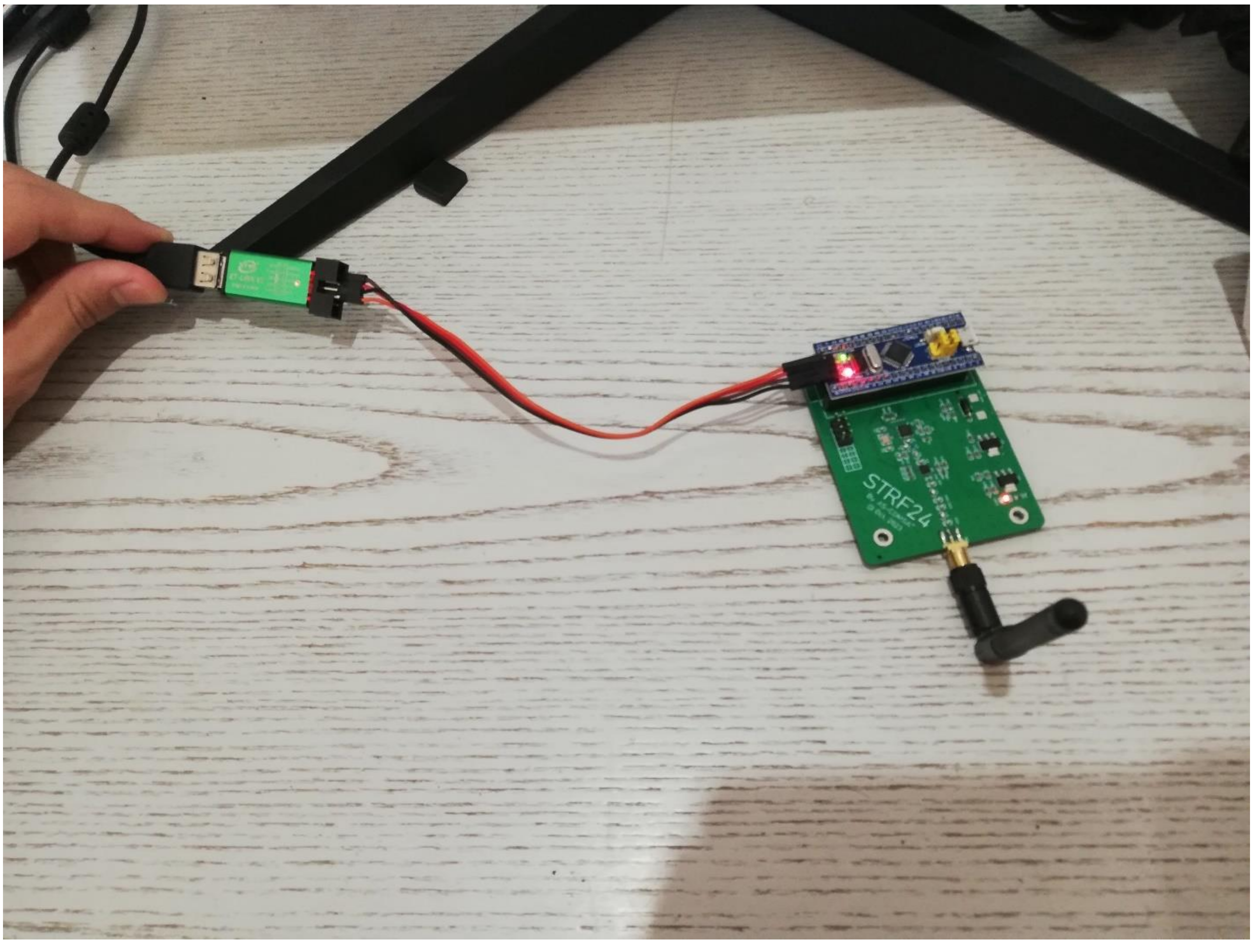


Figure 7-11- Connecting ST-Link programmer USB

Next, we click connect and then we download the file to the STM32.

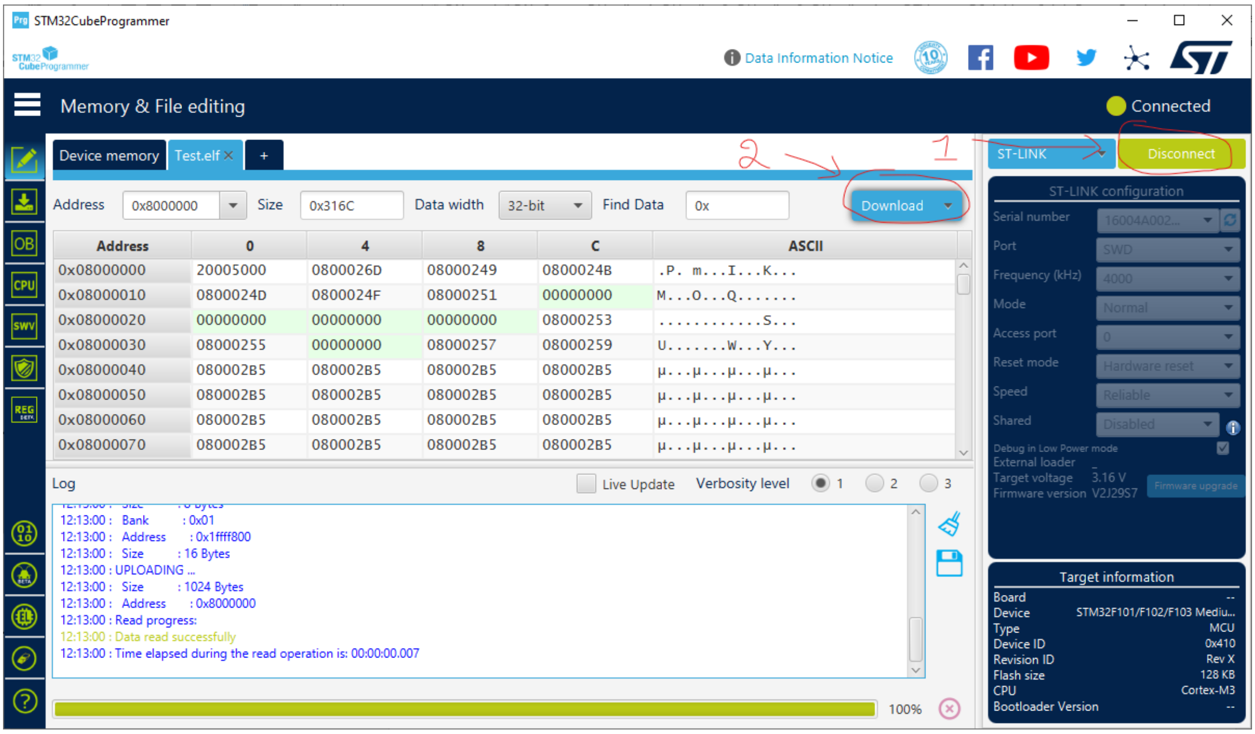


Figure 7-12- Downloading the code on the STM32

Lastly, we need to click the reset button found on the STM32 to refresh the code and we can plug a USB serial port after finishing all the steps.

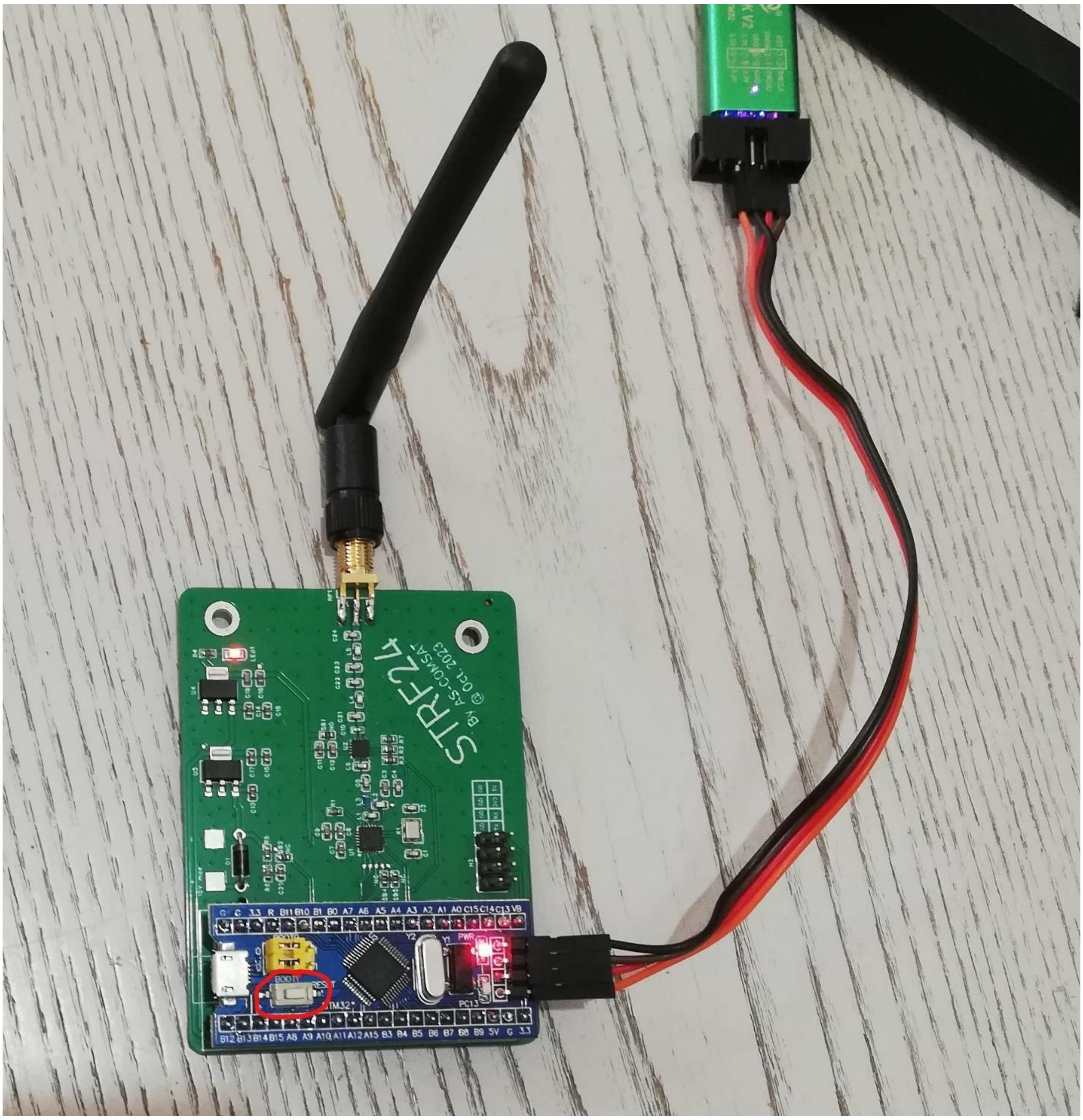


Figure 7-13- Resetting the STM

16.8 Testing

16.8.1 Test Configuration 2 Jan 2024 (circular antennas instead of steered antennas for fist testing, no TT&C)

As previously discussed, our primary focus during testing is to elucidate and evaluate the development process of the City Network Ambulance system meticulously designed by AS-COMSAT. In this section, we aim to demonstrate the testing of the communication device between users and destination nodes.

During testing, we simulate scenarios such as an ambulance user in Lebanon sending a message to a user device in Turkey. This process involves routing the message through the local gateway in Lebanon, transmitting it via the satellite in Lebanon, forwarding it to the satellite in Turkey, and ultimately delivering the message to the user device in Turkey so the Table 8-1 below show the connection between the transceiver. This comprehensive testing ensures the seamless communication between users and destination nodes in the City Network Ambulance system designed by AS-COMSAT.

Table 8-1- Connection between Transceiver

Transceiver	Connected to [0]	Connected to [1]
Ambulance user Lebanon	Gateway node	-
Gateway Node	Ambulance user	Satellite Lebanon
Satellite Lebanon	Gateway Node	Satellite Turkey
Satellite Turkey	Satellite Lebanon	Device User Turkey
Device User Turkey	Satellite Turkey	-

As we show in the figure 8-1 below, to relay a message from an ambulance user in Lebanon to a recipient in Turkey, the process begins by opening the graphical user interface (GUI). Connecting to the serial port is the next step, followed by clicking "Start" to initiate communication. By selecting the marker labeled "Device 3" within the GUI, representing the user in Turkey, we designate the communication target. Subsequently, a concise message is composed and sent through the GUI, ensuring successful delivery.

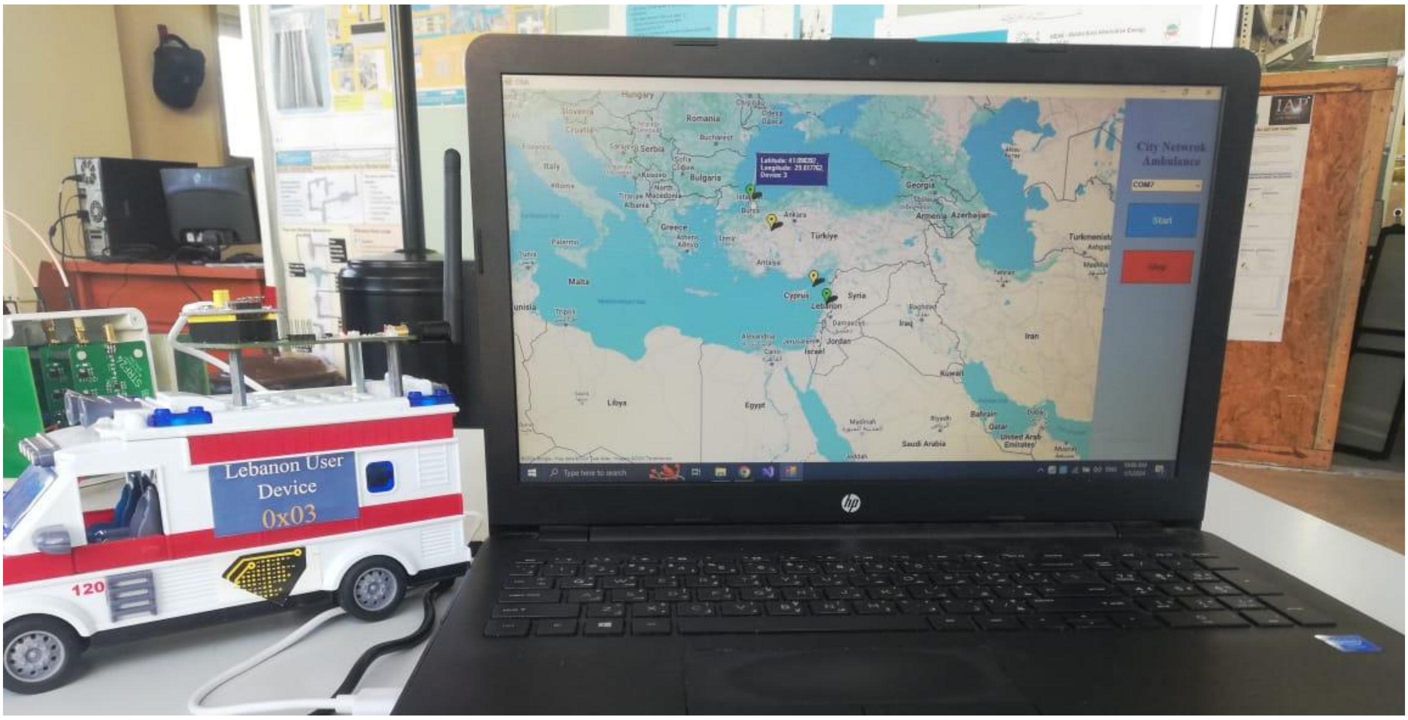


Figure 8-1- Lebanon User

Here in this figure 8-2, we show the message send such as "Go South, and Go East".

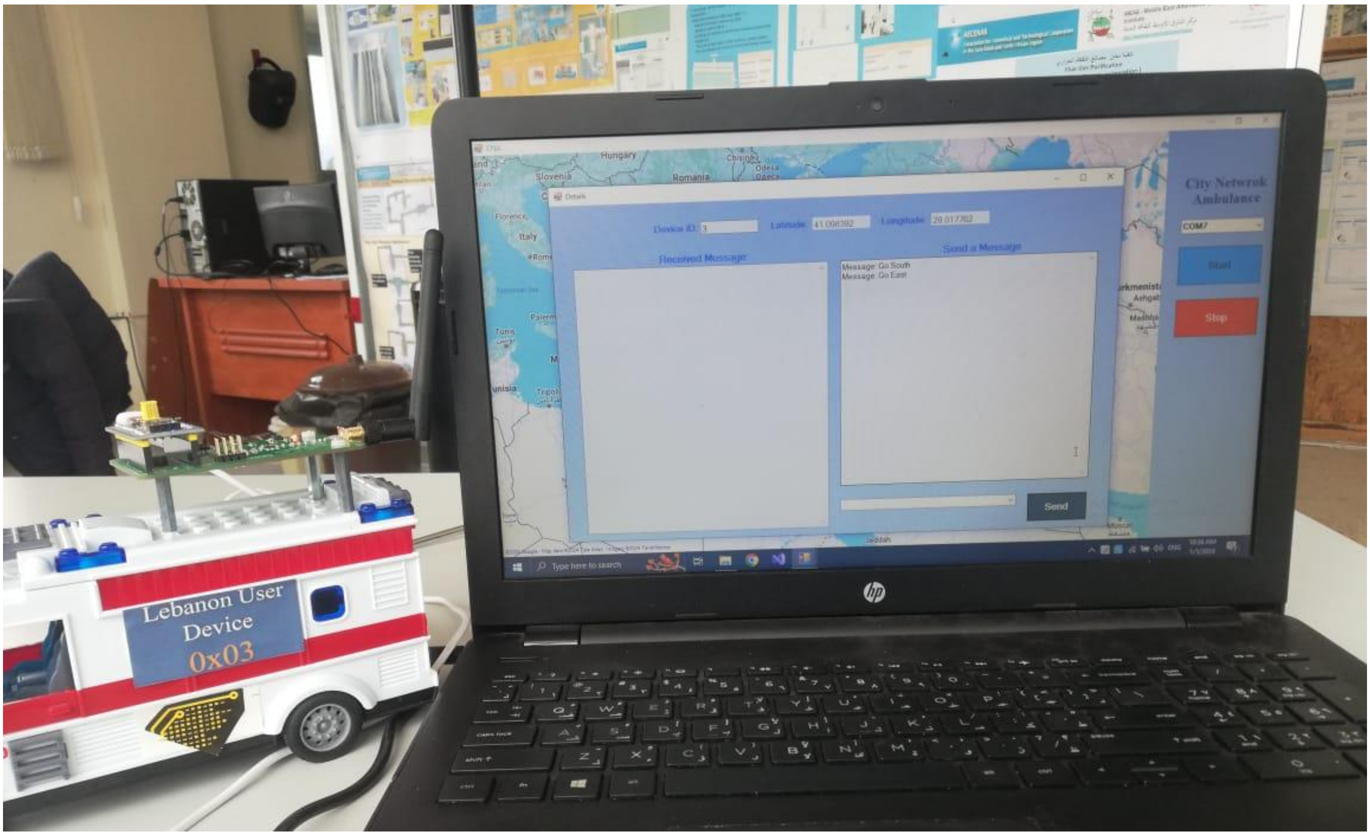


Figure 8-2- Message Send by Lebanon User

The message first passed through a gateway located in Lebanon.

Figure 8-3- Gateway in Lebanon

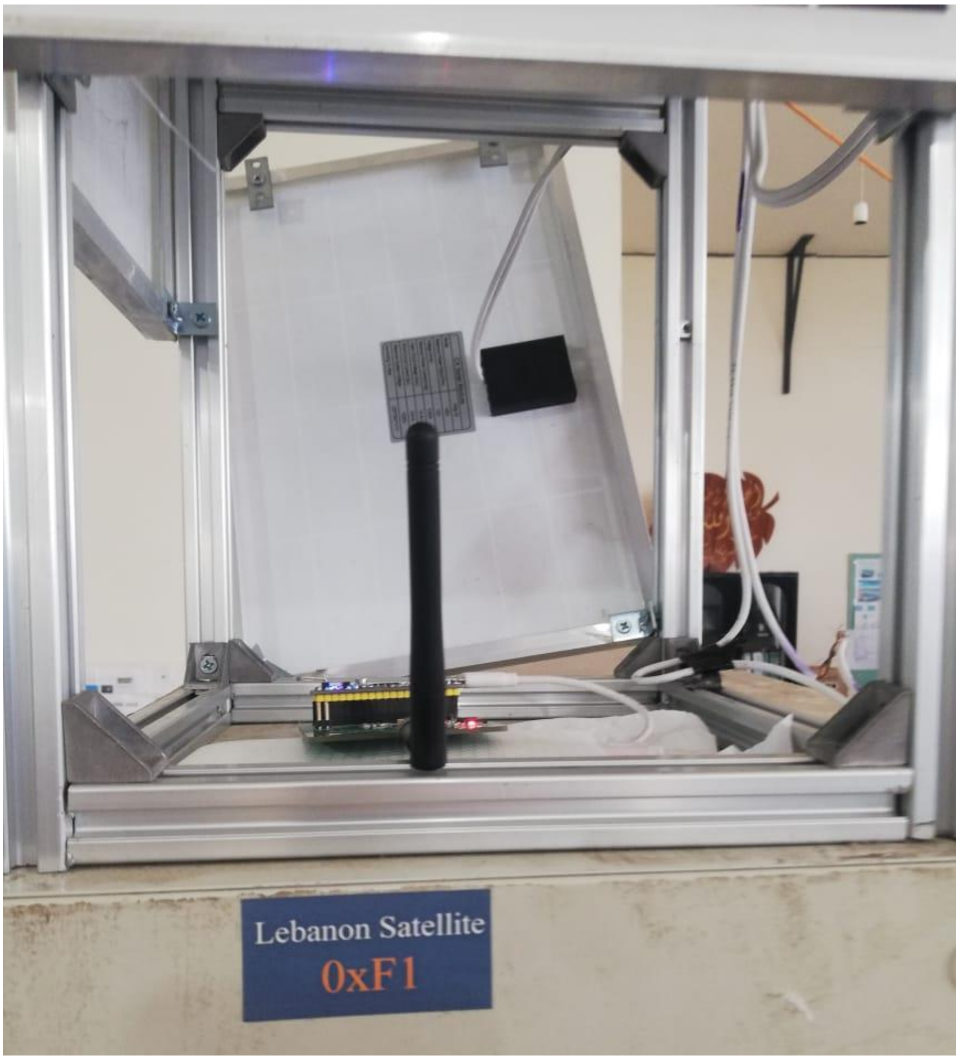


Figure 8-4- Satellite Lebanon

In this scenario, the message sent by the ambulance user in Lebanon was initially received by the "Satellite Turkey." However, before reaching its destination, the message first passed through a gateway located in Lebanon. The gateway in Lebanon processed the message and subsequently forwarded it to the "Satellite Lebanon."

Figure 8-5- Satellite Turkey

From there, the message was relayed to the destination, the "Satellite Turkey," completing the communication flow.

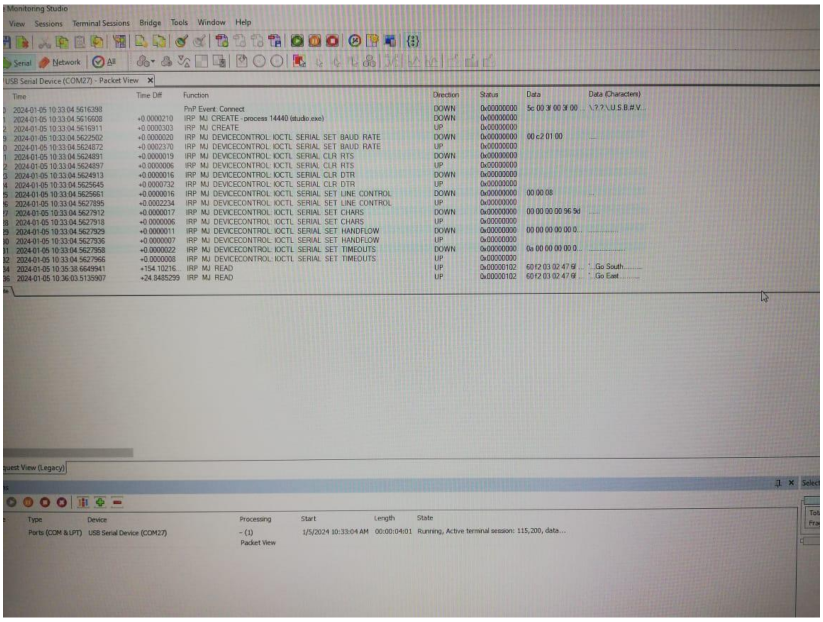


Figure 8-6- Message Send to Satellite Turkey

In this figure 8-6 above, we show that the message that was send in received to the “Satellite Turkey “.

Figure 8-7- Turkey User

Finally, the message arrived at the user in Turkey as show in the figure 8-8, and it was successfully received, as depicted in the graphical user interface (GUI) on the "User Turkey" display.

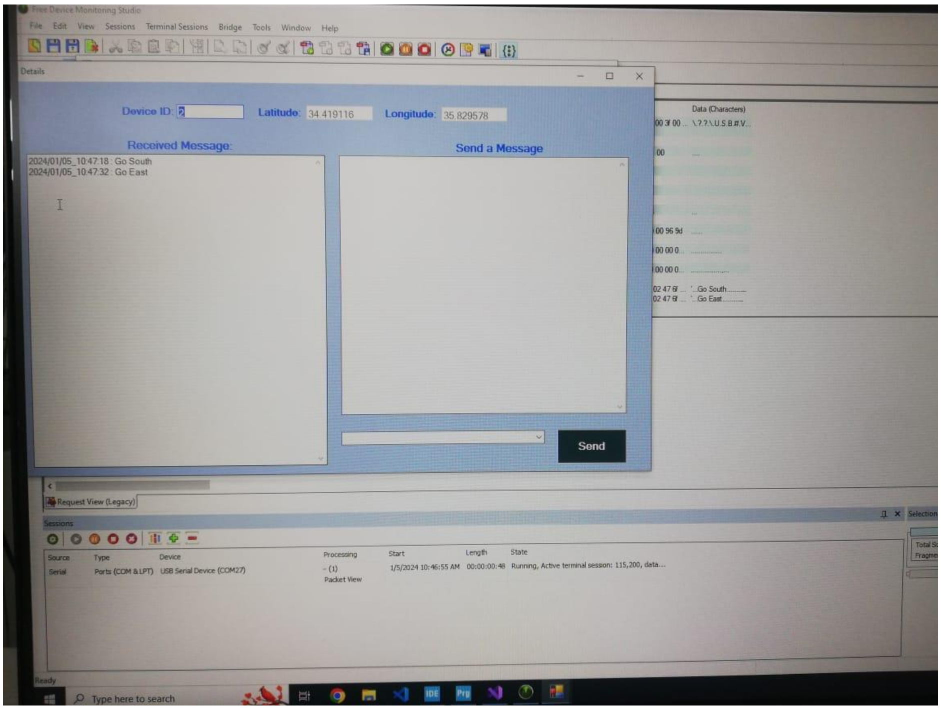


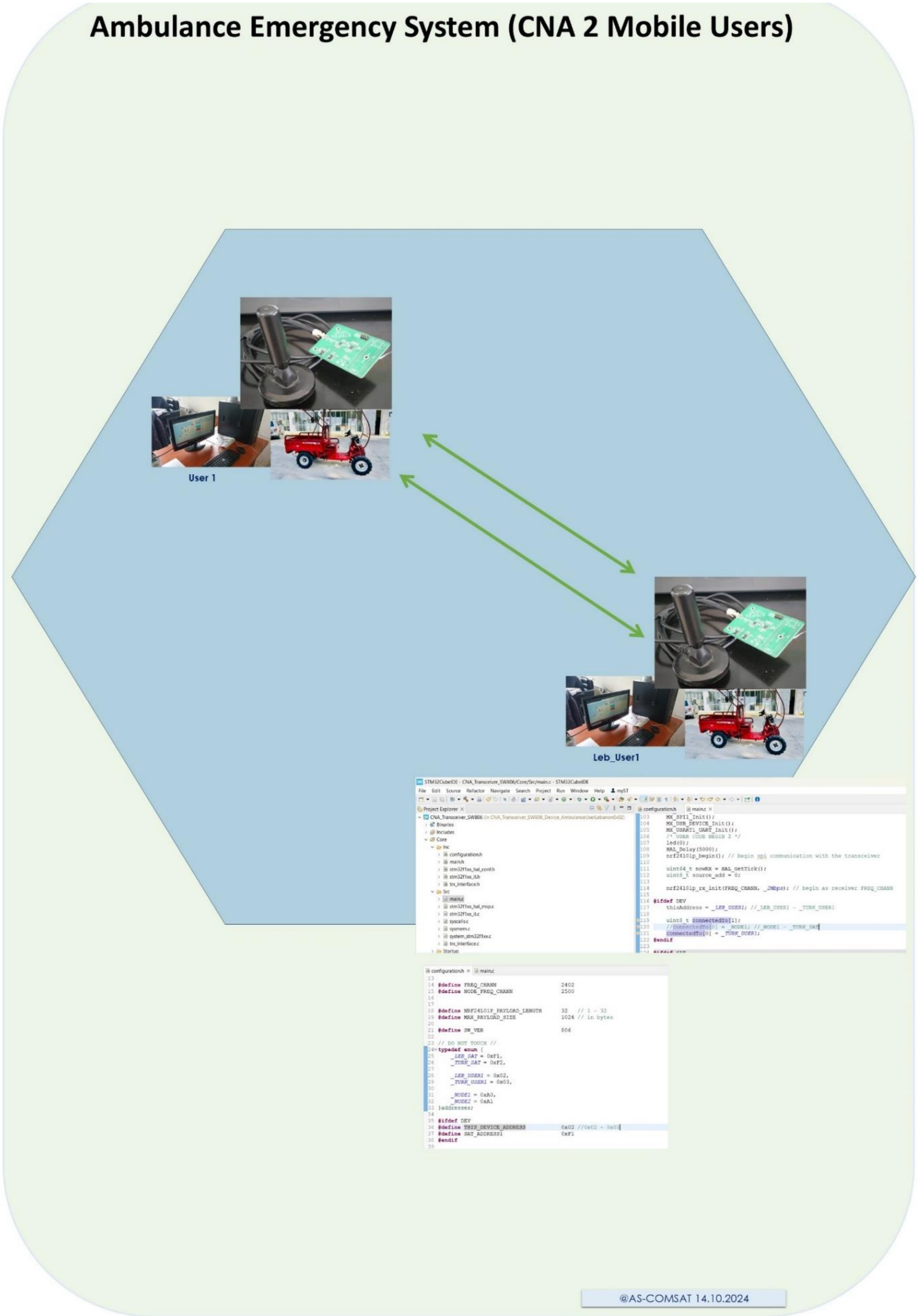
Figure 8-8- Turkey User GUI

16.8.2 Test Configuration 8 Jan 2024

See Testing CNA 8 Jan 2024 (aecenar.com) (https://aecenar.com/index.php/companies/as-comsat/cna-leo-sat-testbed/testing-cna-8-jan-2024)

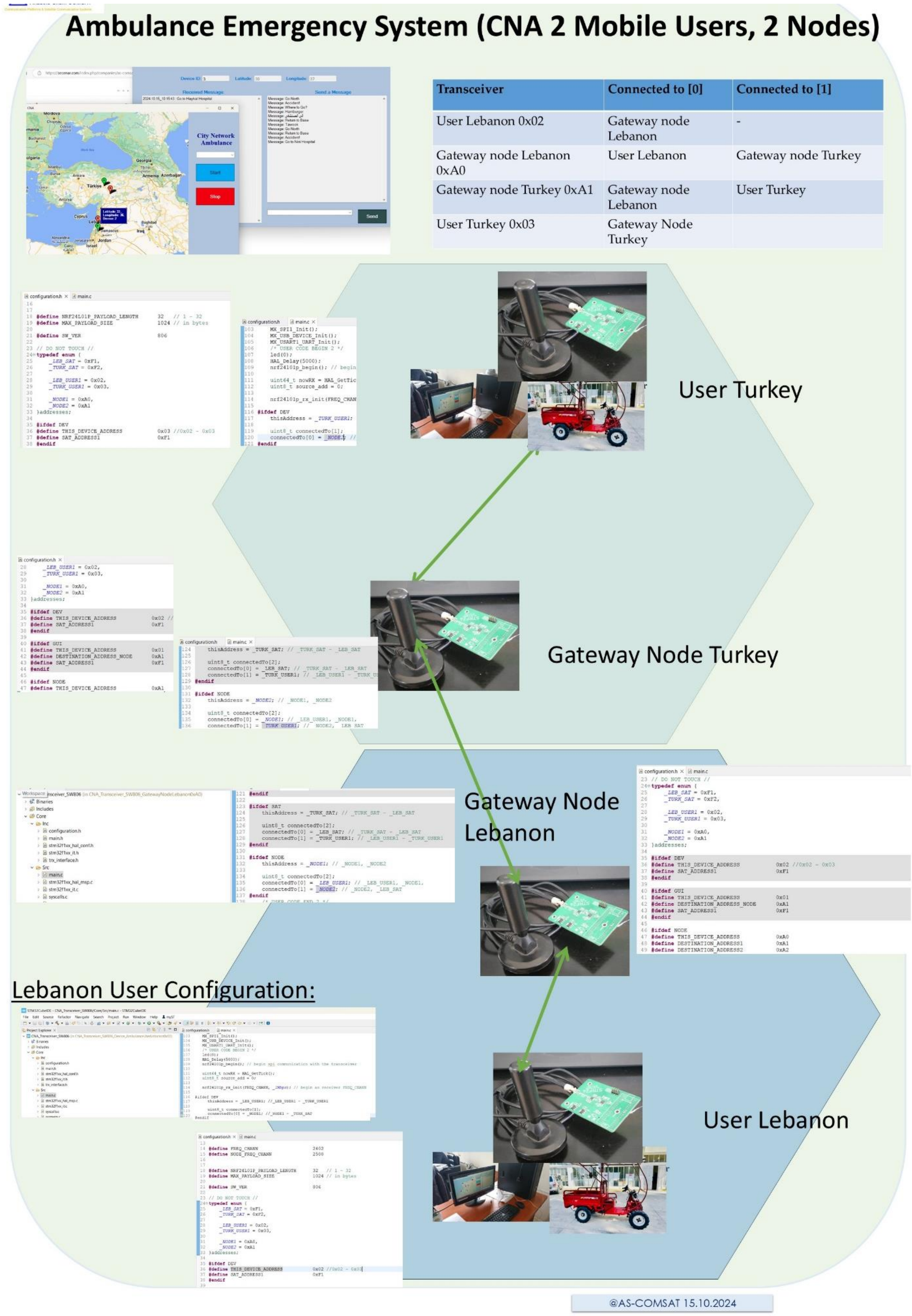
17 CNA (Baseline 31.12.23 Version) with different configurations

17.1 CNA with mobile 2 users



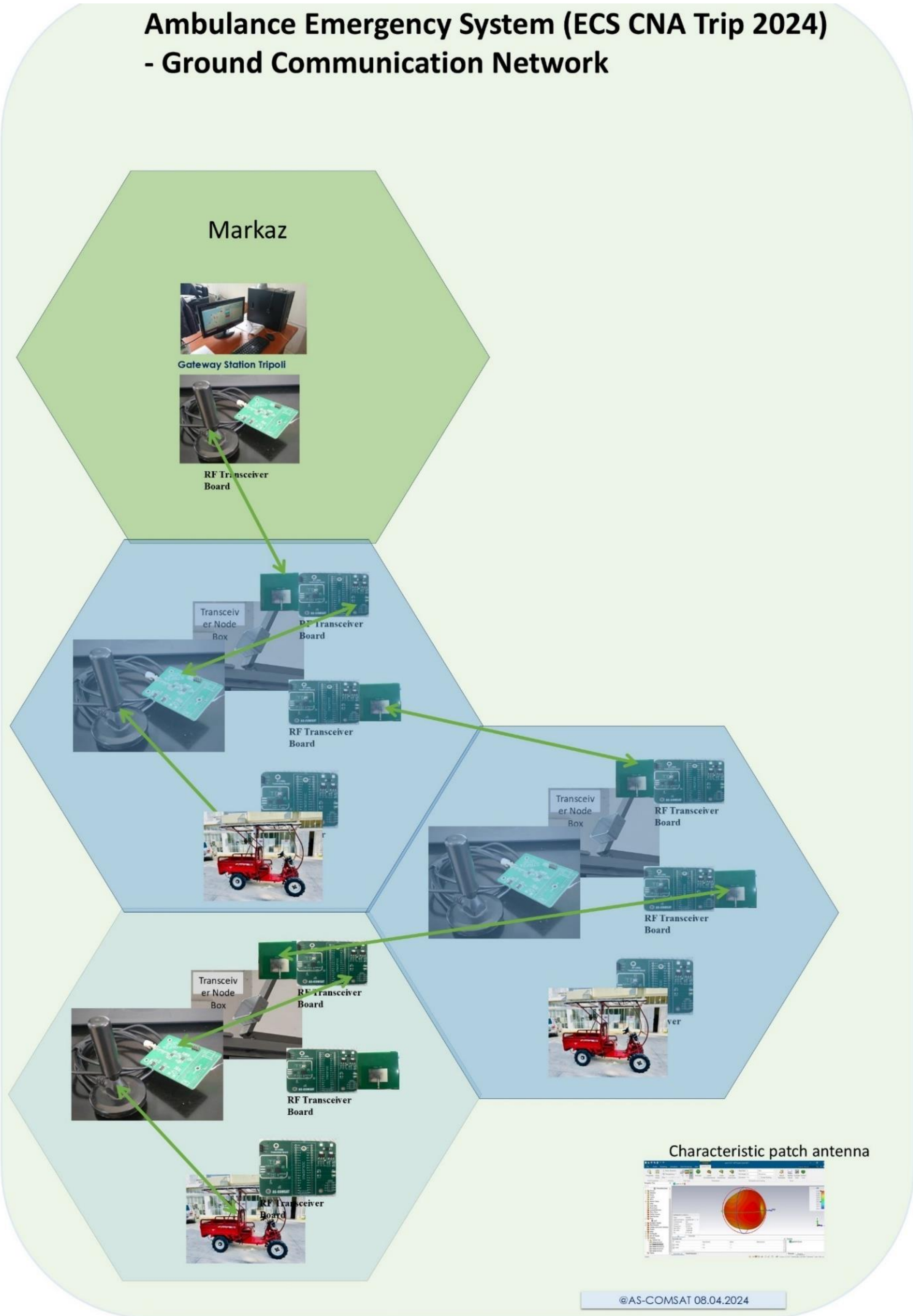
AS-COMSAT CNA 2 mobile users (GUI Code and STM Code for each of the two transceiver cards)

17.2 CNA with 2 nodes and 2 mobile users



[C# GUI Code + STM32 C Code for all of the four transceivers]() (zip file 98 MB)

18 (CNA Version Sep 24, obsolete version!!!!!) CNA with 1 Gateway, 3 nodes, and n fixed users



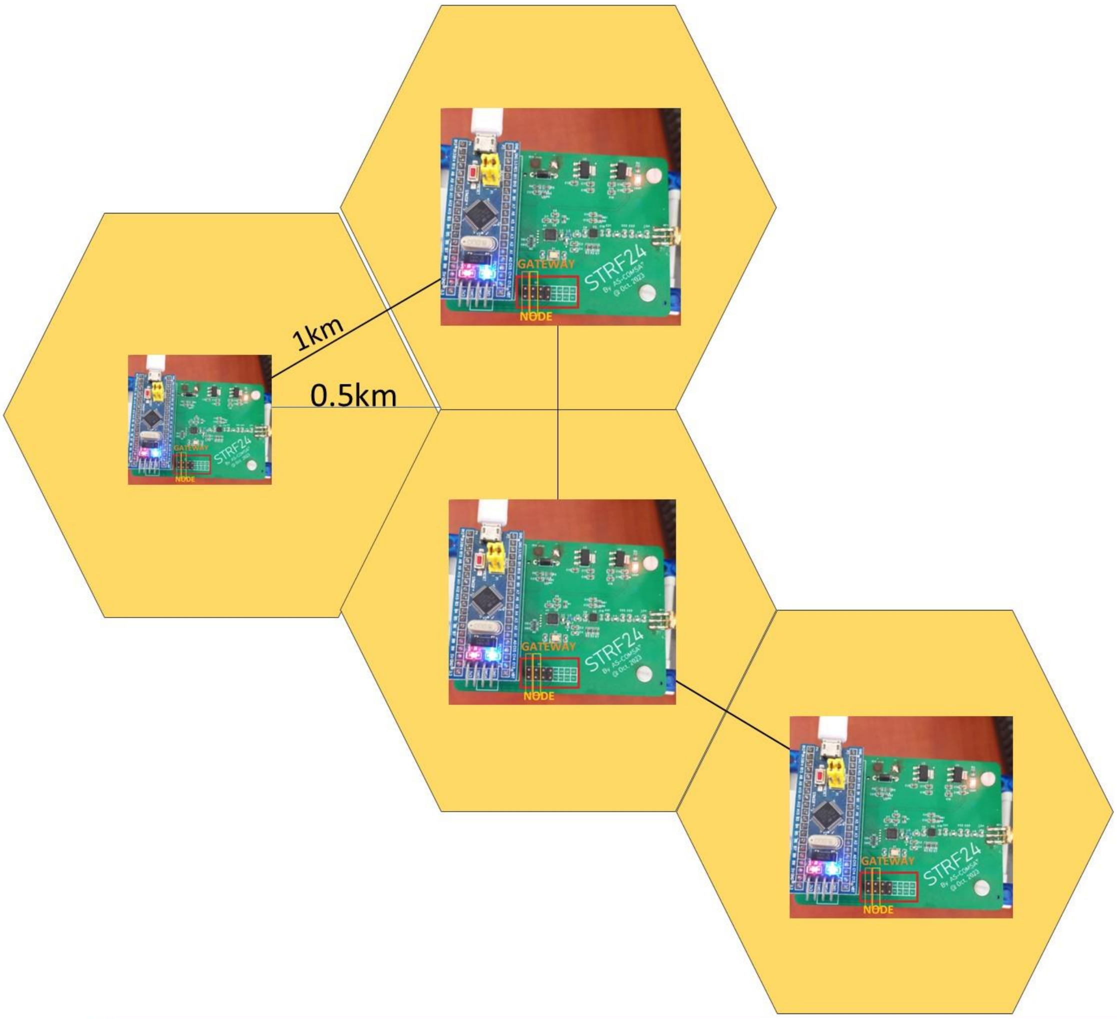
ECS_CNA_Trip_24 Testbed Poster (pptx)

AECENAR
Association for Economical and Technological Cooperation in the Euro-Asian and North-African Region

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

Anadolu-Sham COMSAT

System Design of CNA Communication Node



Nodes
[MReq10] There shall be 1 Gateway, max. 10 users and 5x6=30 nodes.
[MReq20] In the Gateway the communication of all users shall be protocolled.

Nodes Configuration
[MReq30] There shall be covered an area of 5x5 square kilometers. This can be done by a system of 5x6 nodes

YSM @AS-COMSAT – 03 Oct 2024

Poster CNA Trip Requirements (pptx)

Remark: The STM32 SW used here is the version CNA_STM32 (C) Ver. 260924 (fixed users full functional)

18.1 Gateway (Markaz)

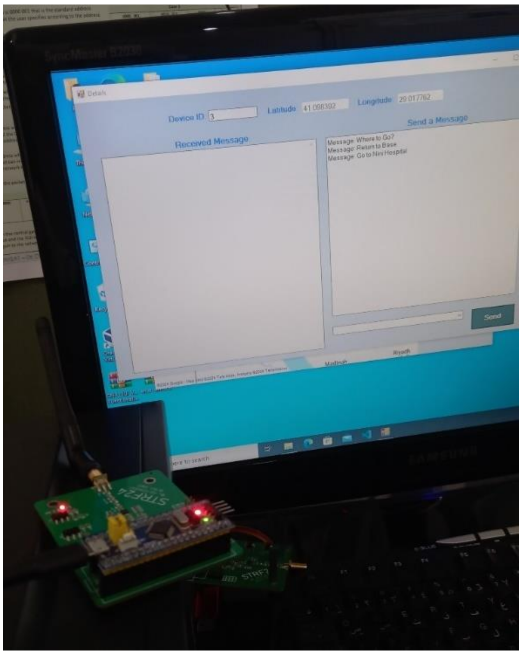
The gateway has a CNA GUI. All communication between mobile users is logged at the gateway to be supervised in case of trouble. This gateway can be considered as the center of operations(all nodes forward their incoming data to this gateway).

18.1.1 Gateway Hardware (Transceiver Board with STM32 + PC (with GUI))

+ PC (with GUI)

The STM is programmed in C language using STM32CubeIDE to send and receive data to the STM at the next node. It is set at receiving state (rx_mode)as a default state, checking incoming data at a fixed time interval. Once the GUI forwards data to be sent, the STM mode gets set to transmitting state (tx_mode).

18.2 Gateway GUI



The STM code: [CNA_Test_Node_V1_020924 (zip folder)]

18.3 SW on STM32 (Transceiver Board embedded SW)

The STM has 2 types of communications (protocols): A USB connection that receives data packets from the GUI through the USB connection, then stores it in rx_buffer. And a SPI protocol that establishes the connection within the STM board to receive data packets by air, storing them in a buffer, then forwarding them to the USB buffer.

```
 *
 ******************************************************************************
 */
/* USER CODE END Header */
/* Includes ------------------------------------------------------------------*/
#include "main.h"
#include "usb_device.h"

/* Private includes ----------------------------------------------------------*/
/* USER CODE BEGIN Includes */
#include "stdio.h"
#include "trx_interface.h"
#include "configuration.h"
#include <stdbool.h>
#include "trx_interface.h"
#include "ttc_app.h"
#include "usbd_cdc_if.h"
#include "string.h"
/* USER CODE END Includes */

/* Private typedef -----------------------------------------------------------*/
/* USER CODE BEGIN PTD */

/* USER CODE END PTD */

```

At the beginning of the "main.c" file, we state our includes as the header files (main.h, configuration.h, usbd_cdc_if.h, ...) and the necessary libraries.

```c

/* Private macro -----------------------------------------------------------*/
/* USER CODE BEGIN PM */

/* USER CODE END PM */

/* Private variables -------------------------------------------------------*/
SPI_HandleTypeDef hspi1;

/* USER CODE BEGIN PV */
uint8_t tx_pld[NRF24L01P_PAYLOAD_LENGTH];
uint8_t rx_pld[NRF24L01P_PAYLOAD_LENGTH];

bool new_rx_data = false;

// addresses for Node 1
uint64_t tx_address = 0x1111111111;
uint64_t rx_address_user = 0xAAAAAAAAAA; // address on pipe 0, connected to Node 2

// addresses for Node 2
//uint64_t tx_address = 0xAAAAAAAAAA;
//uint64_t rx_address_user = 0x1111111111; // address on pipe 0, connected to Node 1


/* USER CODE END PV */

/* Private function prototypes ---------------------------------------------*/
void SystemClock_Config(void);
static void MX_GPIO_Init(void);
static void MX_SPI1_Init(void);
/* USER CODE BEGIN PFP */

/* USER CODE END PFP */

/* Private user code -------------------------------------------------------*/
/* USER CODE BEGIN 0 */
```

Whatever is written outside of the USER snippets is auto-generated by the software. The first field we changed was the "USER PV", as we added private variables such as "rx_pld" and "tx_pld", 2 arrays of type int where we'll store the upcoming and to_be_forwarded data respectively. A variable of boolean type "new_rx_data" that serves as a flag set as false by default, meaning there is no incoming data yet. As well as 2 int variables, one for the transmitting address, and the other for receiving.

```c
int main(void) {

    /* Configure the system clock */
    SystemClock_Config();

    /* USER CODE BEGIN SysInit */

    /* USER CODE END SysInit */

    /* Initialize all configured peripherals */
    MX_GPIO_Init();
    MX_SPI1_Init();
    MX_USB_DEVICE_Init();
    /* USER CODE BEGIN 2 */
    nrf24l01p_begin(); // begin SPI communication with the TRX

    nrf24l01p_rx_init(2405, _2Mbps); // begin as RX --> default TRX address = E7E7E7E7E7

    HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin, 0);
    HAL_Delay(3000);

    nrf24l01p_set_tx_address(tx_address);
    nrf24l01p_set_rx_address(0, rx_address_user);
    //nrf24l01p_set_rx_address(0, rx_address_user);

    //nrf24l01p_set_tx_address(tx_address);
    //nrf24l01p_set_rx_address(0, rx_address0);
    //nrf24l01p_set_rx_address(1, rx_address1);
    //nrf24l01p_set_rx_address(2, rx_address2);

    /* USER CODE END 2 */
```

The STM uses the "nrf24l01p" to sent and receive packets by air(radio channel) using its own buffers. So to display the data on the user's GUI, we created functions that handle moving the content of the buffers to the STM's own buffers then send them over USB. This operation isn't as straightforward as it sounds, and it should be treated carefully step by step. First we initiate the communication between the STM and nrf24l01p using the SPI protocol, through the function "nrf24l01p_begin()", set it to receiving mode using "nrf24l01p_rx_init()" that takes the radio channel and frequency as inputs. The "HAL_GPIO_WritePin()" and "HAL_Delay()" functions are simply to turn on the STM's LED to make it easier to detect if the STM started to work or not. And finally the last 2 functions are to set the Tx_address as well the Rx_address. But the difference is that the STM has 6 channels so it can listen up to 6 connected devices, each on its own channel and Rx_address, that's why it takes 2 inputs: channel_number and rx_address.

```
    while (1) {
        /* USER CODE END WHILE */

        /* USER CODE BEGIN 3 */
#ifdef GUI

        nrf24l01p_ptx_mode(); // switch to TX mode
        memset(tx_pld, 0, sizeof(tx_pld)); // clear TX payload buffer to load new data

        //get_cmd_data(tx_pld);
        get_cna_message(tx_pld);
        //CDC_Transmit_FS(tx_pld, sizeof(tx_pld));

        // send the packet by air
        nrf24l01p_write_tx_fifo(tx_pld);
        nrf24l01p_flush_tx_fifo();

        // if the TRX is not set to RX, set it to RX
        if (nrf24l01p_get_comm_mode() != _RX) {
            nrf24l01p_prx_mode();
        }

        // wait for a response within 1 second
        uint64_t time = HAL_GetTick();
        while (!new_rx_data && (HAL_GetTick() - time) < 1000) {

        }

        if (new_rx_data) {
            new_rx_data = false; // Reset reception flag

            switch (rx_pld[0]) {
                case CMD_FRAME:
                    // Send the received message to the USB for GUI display
                    CDC_Transmit_FS(rx_pld, sizeof(rx_pld));
```

We start a while loop that's always true, this is to ensure that the STM is always functioning and ready to transceive data. In this loop we define two different cases: "GUI"(what's written in ifdef section) and "SAT"(what's written in else section). The GUI section, when data are incoming by radio (indicated by the first byte of the package), we switch the mode of the STM to Tx mode using the function "nrf24l01p_ptx_mode()", clear the

For further information on STM SW, see: index.php?option=com_content&view=article&id=698

18.4 Communication Node (Transceiver Node Box)

Each communication node has an antenna and two microstrip patch antennas for transferring the data, and a STM32F103C8T6 to process them. The antenna receives signals from users within a radius (<1000m) then the node forwards those signals to their appropriate receivers, whether them being users within the same node or users in another node by projecting the signals to the next node using the microstrip patch antennas.

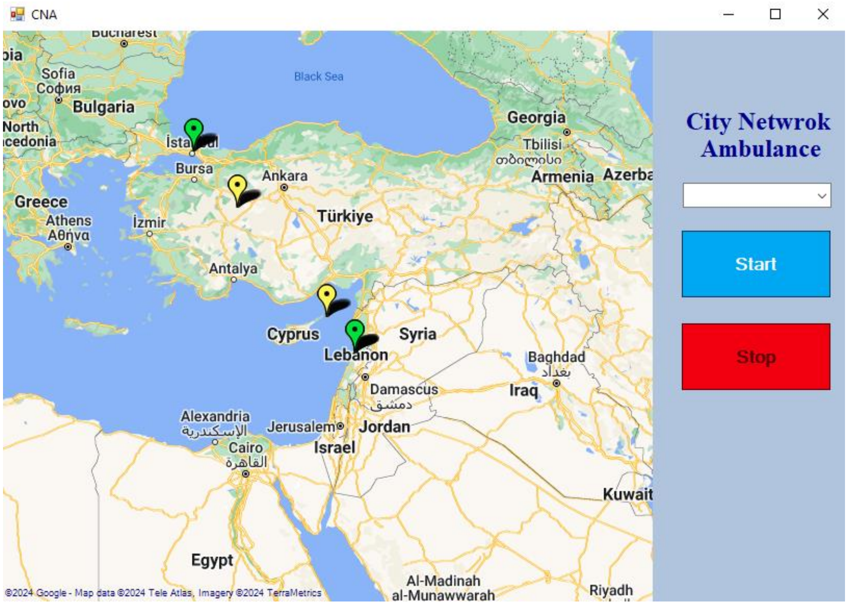
By giving each node a transmitting address (Tx address) over 5 bytes, this node can listen up to 6 different connected devices through 6 different receiving addresses (Rx address 0...6) whereas those connected devices can be mobile users and nodes.

18.5 Mobile User (Ambulance Vehicle)

Each mobile user has a CNA GUI, actually running on laptop (same program as Gateway). There shall be developed an application running with Android mobile phone which is connected to the AS-COMSAT transceiver board.

18.5.1 3.2 Mobile user's GUI

Link to download the VS GUI code: CNA_GUI(zip) The GUI has two forms and a state machine that handles the getters and setters:



The 1st form, called "CNA", displays a map locating the connected users (indicated by green markers) and the nodes (indicated by yellow markers). The user selects the port first from the drop list (com27 for example), then presses the "start" button to establish the connection. Once the connection starts, the user can choose one of the green markers (other users) which will open the second form "Details.cs".

```
namespace CNA_GUI
{
    public partial class Main_Form : Form
    {
        // Create a ManagementEventWatcher to monitor for serial port changes
        private ManagementEventWatcher portWatcher;

        StateMachine sm;

        //Initialize marker to store the coordinates of the nodes and display them on the map
        GMapOverlay markers = new GMapOverlay("markers");

        public Main_Form(StateMachine sm)
        {
            InitializeComponent();
            map.DragButton = MouseButtons.Left;
            map.MapProvider = GMapProviders.GoogleMap;
            this.sm = sm;
        }

        #region Main Form/Map Region

        private void Main_Form_Load(object sender, EventArgs e)
        {
            // Initialize the ManagementEventWatcher
            InitializePortWatcher();

            PopulateAvailablePorts();

            Stop.Enabled = false;
        }

        int SatAddress = 1;
        int deviceCount = 2;
        int nodeCount = 1;
```

We call the "StateMachine.cs" as "sm", which contains getters and setters allowing the first frame to get the longitude and latitude coordinates of the destined device then forwarding them to the second form. Initializing markers on the map to store the coordinates of the nodes and display them on the map. The main_form shall take the stateMachine as an input and display a map provided by GoogleMaps.

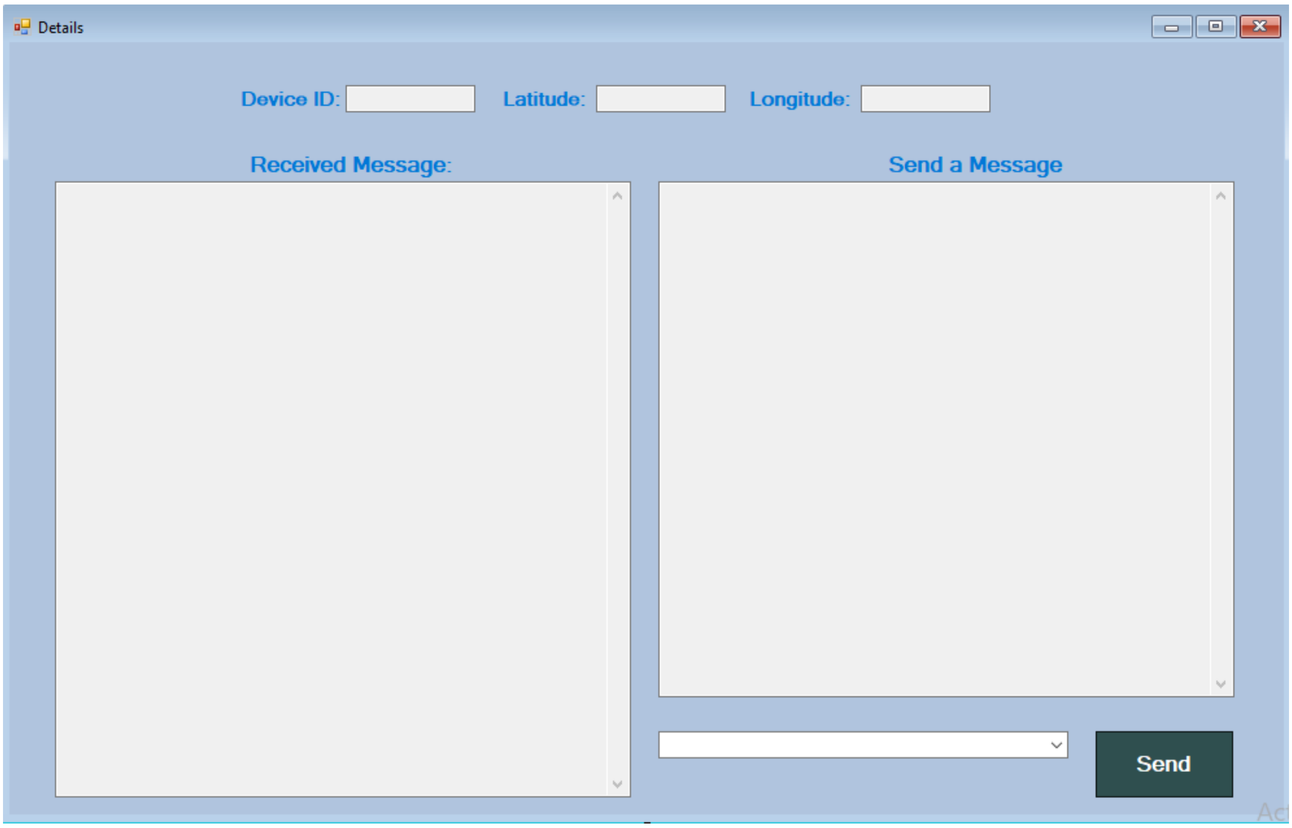
When this form is loaded in the beginning, the "stop" button is disabled, the "InitializePortWatcher()" and "PopulateAvailablePorts()" functions called. Variables as "SatAddress", "deviceCount", and "nodeCount" are declared outside of the main_form to be used by multiple functions later.

```
private void map_Load(object sender, EventArgs e)
{

    map.ShowCenter = false;
    map.MinZoom = 0;    // Minimum zoom level
    map.MaxZoom = 18;   // Maximum zoom level
    map.Zoom = 5;    // Current zoom level

    string relativePath = @"..\..\..\mapCoordinates.txt";
    string filePath = Path.GetFullPath(relativePath);

    //Lists to hold the file contents
    List<string> lines = new List<string>();

    //Reading from the file
    lines = File.ReadAllLines(filePath).ToList();

    //Array to hold list content
    string[] items;

    //loop to extract the list content
    foreach (string line in lines)
    {
        items = line.Split(',');

        location l = new location(items[1], items[2]);

        if (items[0] == "S")
        {
            sm.addCoordinatesItem("S");
        }
        else if (items[0] == "D")
        {
            sm.addCoordinatesItem("D");
```

Adding the map comes with several properties to adjust, such as the minimum value as well as the maximum value of zooming. The markers previously mentioned, have fixed coordinates which are saved in a text file "mapCoordinates.txt". This file is read, then had its content saved in an array of list content "items". Each line in the text file is written as "X, xx.xxx, xx.xxx" where X is either "S" or "D" (user or node), and xx.xxx are the coordinates, latitude and longitude of the device. These contents are then displayed properly on the map as markers, as written below.

```
//Loop to insert locations on the map
for (int i = 0; i < sm.getLatitudes().Count; i++)
{

    //mapping positions
    map.Position = new PointLatLng(sm.latitudes[i], sm.longitudes[i]);

    //Markers
    PointLatLng point = new PointLatLng(sm.latitudes[i], sm.longitudes[i]);

    //Cover map with Overlay
    map.Overlays.Add(markers);

    if (sm.getCoordinatesType()[i] == "S")
    {
        GMapMarker marker = new GMarkerGoogle(point, GMarkerGoogleType.yellow_dot);
        marker.Tag = SatAddress + ";" + sm.getCoordinatesType()[i];
        //sm.nodes.Add("Node " + nodeCount);

        marker.ToolTipText = $"Latitude: {sm.latitudes[i]} , \nLongitude:  {sm.longitudes[i]}, \nSatellite: {SatAddress}";
        var toolTip = new GMapToolTip(marker);
        toolTip.Fill = new SolidBrush(Color.DarkBlue);
        toolTip.Foreground = new SolidBrush(Color.White);
        toolTip.Offset = new System.Drawing.Point(10, -50);
        toolTip.Stroke = new Pen(new SolidBrush(Color.PaleGoldenrod));
        marker.ToolTip = toolTip;
        markers.Markers.Add(marker);
        SatAddress++;

    }
    else if (sm.getCoordinatesType()[i] == "D")
    {
```

```
}
else if (sm.getCoordinatesType()[i] == "D")
{
    GMapMarker marker = new GMarkerGoogle(point, GMarkerGoogleType.green_dot);//GMarkerGoogleType.green_dot
    marker.Tag = deviceCount + ";" + sm.getCoordinatesType()[i];

    marker.ToolTipText = $"Latitude: {sm.latitudes[i]} , \nLongitude:  {sm.longitudes[i]}, \nDevice: {deviceCount}";
    var toolTip = new GMapToolTip(marker);
    toolTip.Fill = new SolidBrush(Color.DarkBlue);
    toolTip.Foreground = new SolidBrush(Color.White);
    toolTip.Offset = new System.Drawing.Point(10, -50);
    toolTip.Stroke = new Pen(new SolidBrush(Color.PaleGoldenrod));
    marker.ToolTip = toolTip;
    markers.Markers.Add(marker);
    deviceCount++;
}
else if (sm.getCoordinatesType()[i] == "N")
{
    GMapMarker marker = new GMarkerGoogle(point, GMarkerGoogleType.red_dot);
    marker.Tag = SatAddress + ";" + sm.getCoordinatesType()[i];

    marker.ToolTipText = $"Latitude: {sm.latitudes[i]} , \nLongitude:  {sm.longitudes[i]}, \nNode: {nodeCount}";
    var toolTip = new GMapToolTip(marker);
    toolTip.Fill = new SolidBrush(Color.DarkBlue);
    toolTip.Foreground = new SolidBrush(Color.White);
    toolTip.Offset = new System.Drawing.Point(10, -50);
    toolTip.Stroke = new Pen(new SolidBrush(Color.PaleGoldenrod));
    marker.ToolTip = toolTip;
    markers.Markers.Add(marker);
    nodeCount++;
```

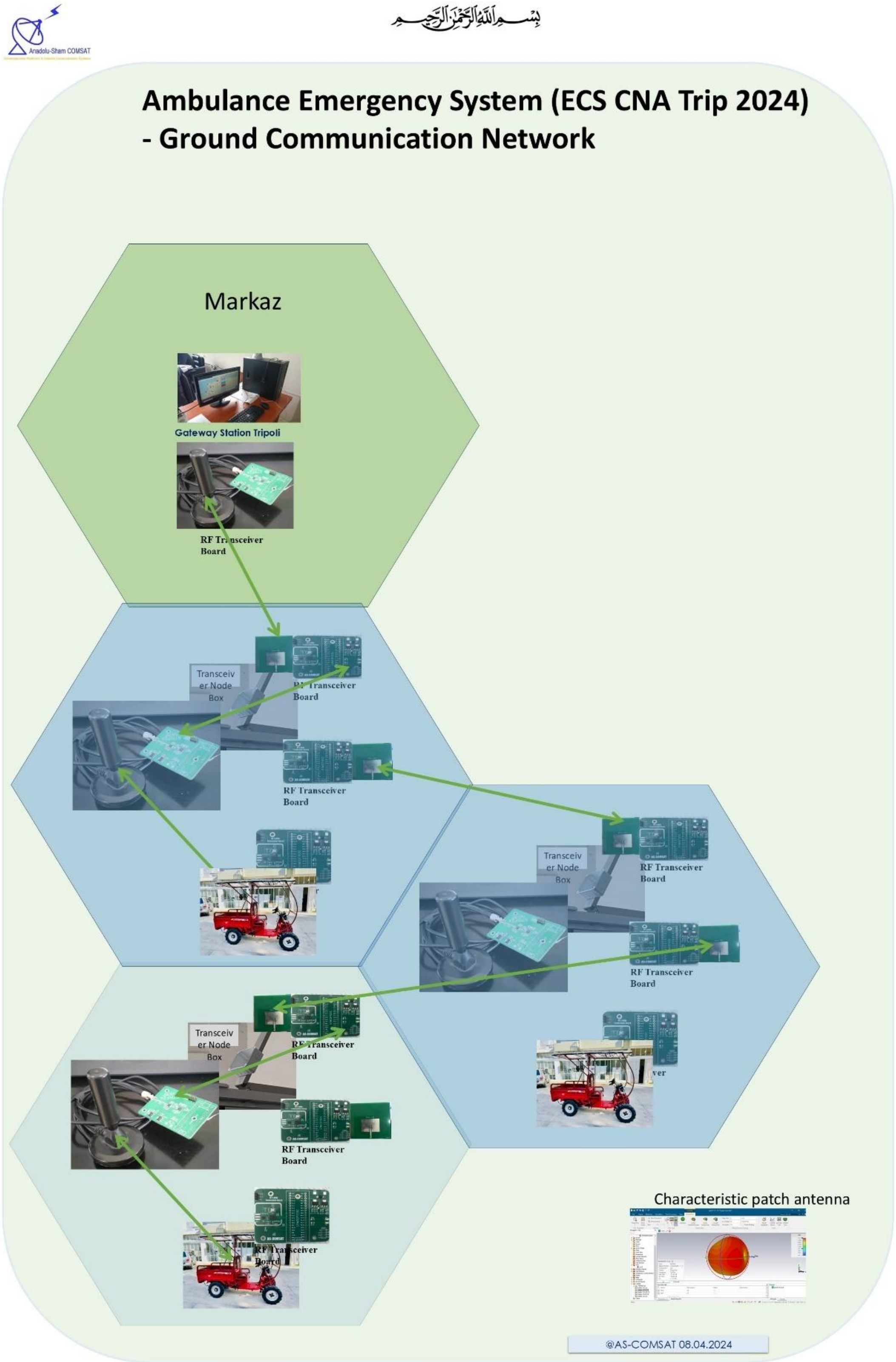


The 2nd form, called "Details", displays the received messages of this user in the "Received Message" box, while also displaying the device ID as well as the coordination in which the device is located. It also allows this user send specific messages (command messages) from the drop list then pressing send to forward this message to the near node. This message is displayed in the "Send a Message" box for more visualization

18.6 (CNA Version Sep 24, obsolete version!!!!!) Users Guide - CNA with 1 Gateway, 3 nodes, and n fixed users

18.6.1 Installation and first steps

18.6.1.1 Overview

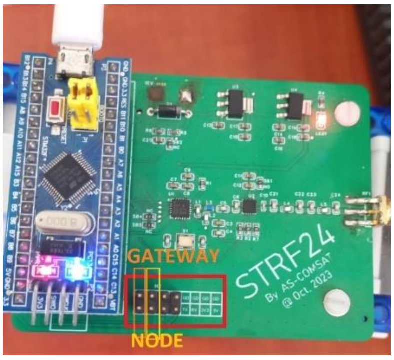


18.6.1.2 Gateway PC

Needed HW

1 Windows PC, 1 AS-COMSAT Transceiver Board (with installed STM32 SW: CNA_STM32 (C) Ver. 260924 (fixed users full functional)), 1 router antenna (360 degrees), 1 small USB Cable (as for old Android mobile phones)

HW Configuration



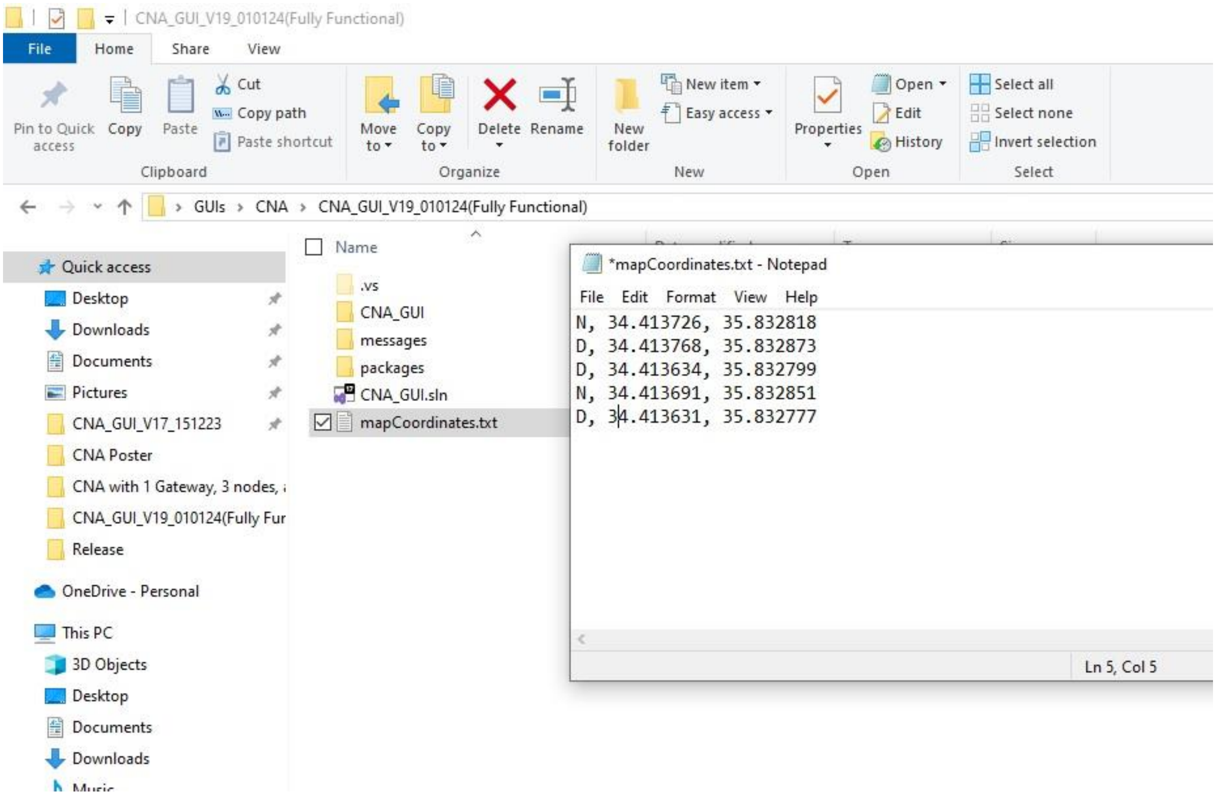
Connect the two left pins with a cable or a dumper (as shown in above picture for "Gateway")

Needed SW

CNA GUI SW: CNA GUI C# Ver. 010124 (Fully Functional)

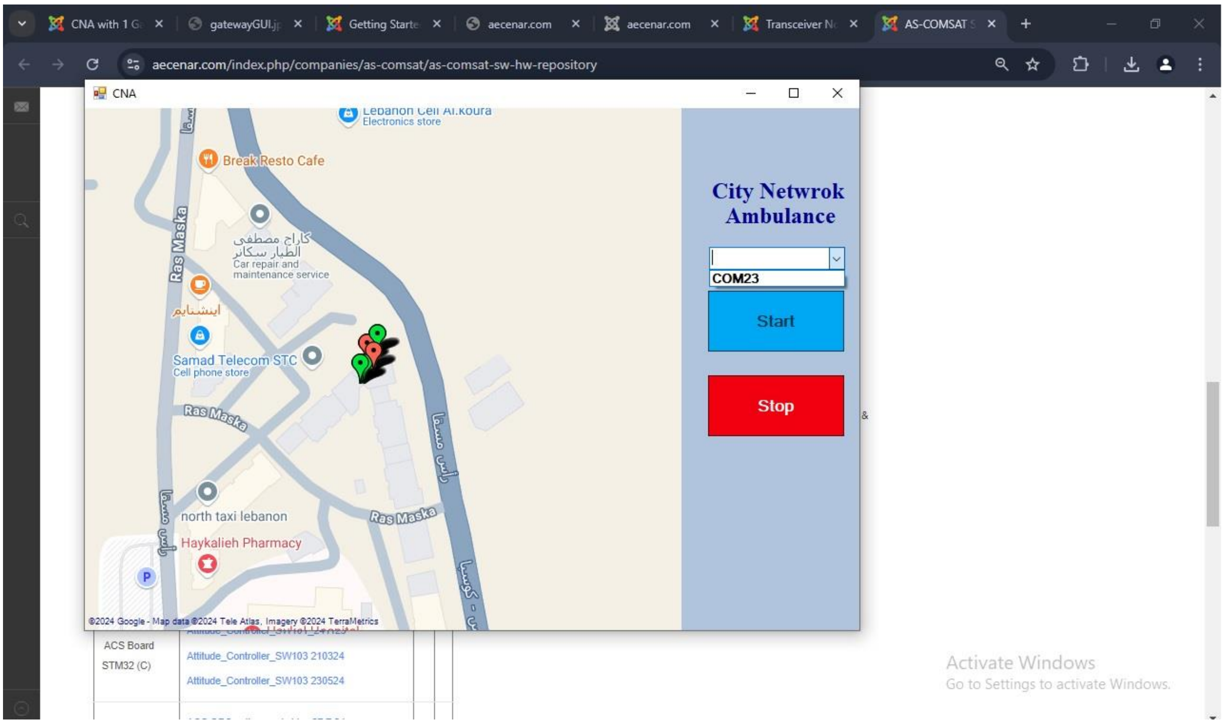
Starting the CNA GUI SW

1. Configure the locations of node (N) and communication partners (gateway and users) (D) by putting the coordinates into the .txt file



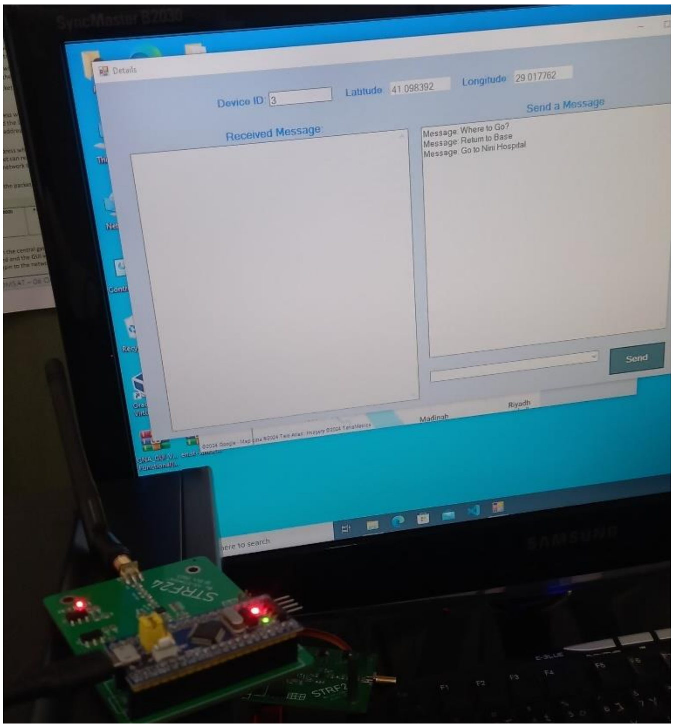
For this example the location file is here: https://aecenar.com/index.php/downloads/send/13-temo-space-communication/1772-250924cna-1gateway-3nodes-nusers-mapcoordinates-example1

2. Start the CNA GUI application file from the release folder. You will see the window below:



3. Choose a COM port. As you only plugged in one transceiver card, there will only one possibility.

4. Click on one of the above locations, with which you want to communicate. (You are automatically Device 1). Then the below picture appears. You can send a message by choosing a message from the scroll bar (at bottom right side). The message you send will appear in the right window. The messages of you communication partner will appear in the left window.

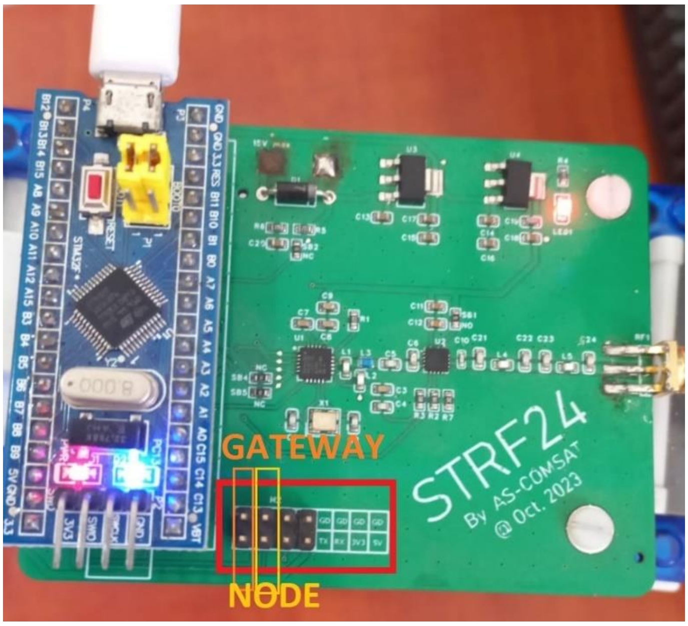


18.6.1.3 User 1

Needed HW

A laptop or PC, rest same as Gateway

HW Configuration



Do not connect any pins.

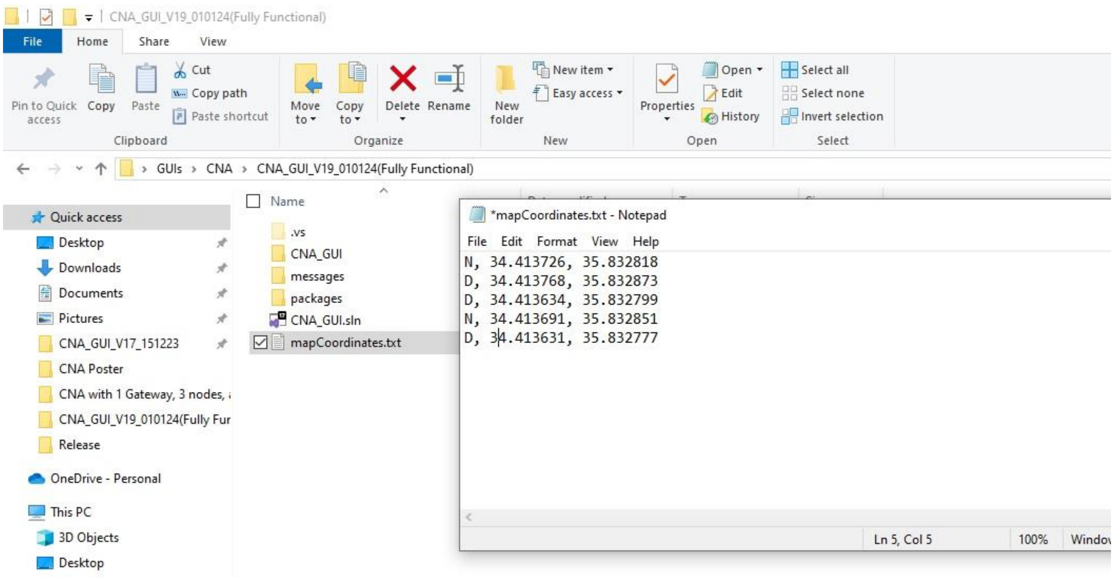
Needed SW

Same as Gateway

Put for this example the location file

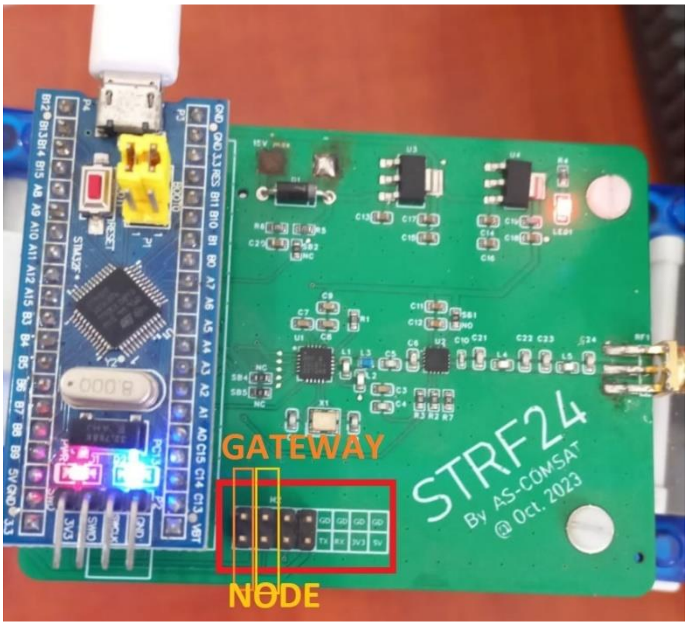
https://aecenar.com/index.php/downloads/send/13-temo-space-communication/1772-250924cna-1gateway-3nodes-nusers-mapcoordinates-example1

into the folder



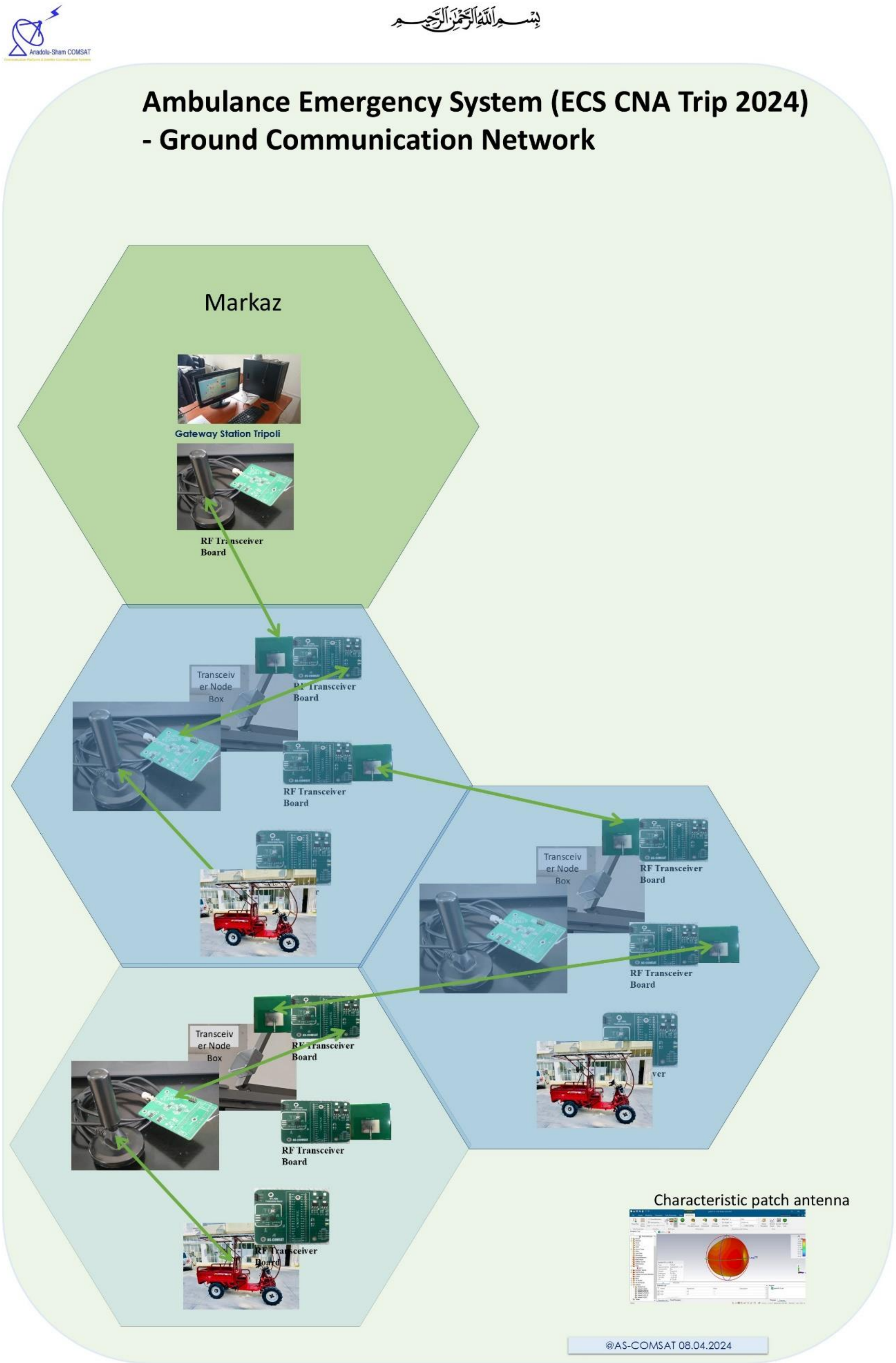
18.6.1.4 Remark 1.10.24

Actually the STM32 SW CNA_STM32 (C) Ver. 260924 (fixed users full functional)) is only functional if two user (no jumper connected as shown in the following picture



18.7 (CNA Version Sep 24, obsolete version!!!!!) Developers Guide - CNA with 1 Gateway, 3 nodes, and n fixed users

18.7.1 System Overview



18.7.2 Actual (Sep 26) main features

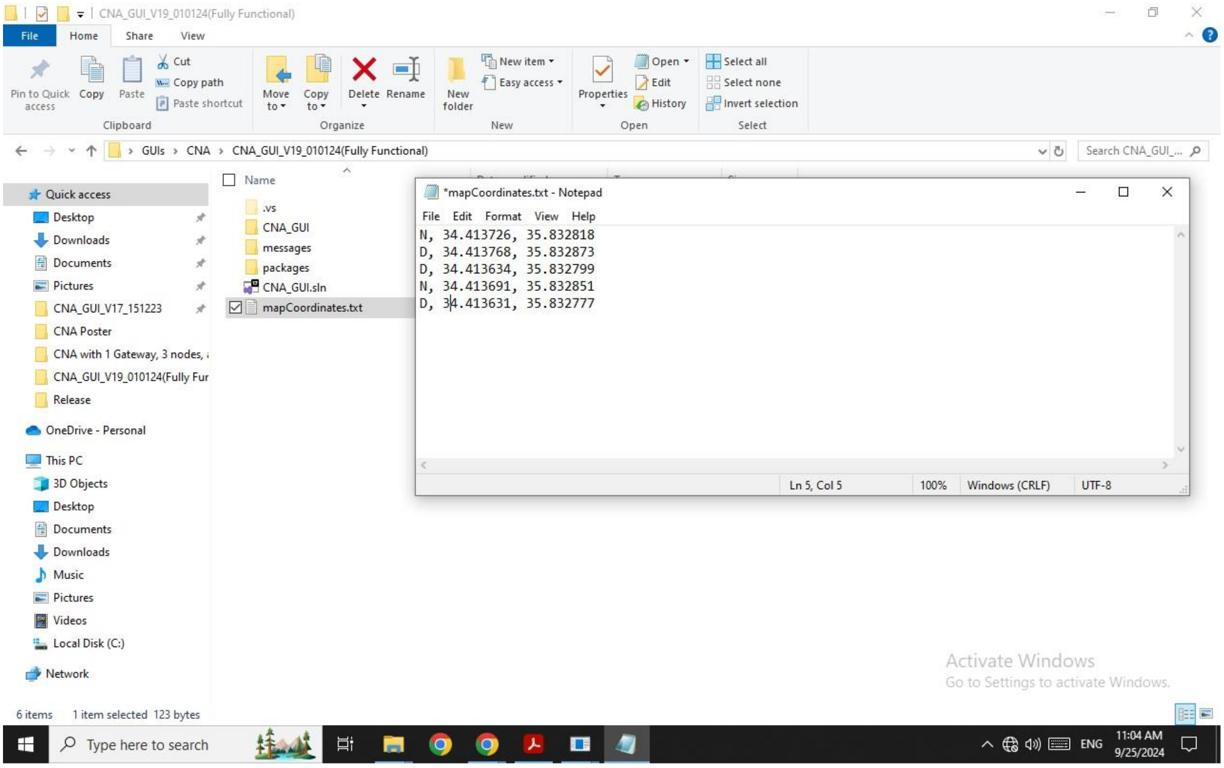
- only for fixed user locations

- Gateway and user has an identical GUI

18.7.3 GUI C# program

This program is the same for the Gateway and for the User, except the configuration file, which determines if it is for Gateway or for user.

18.7.3.1 Configuration File



In this file there is specified: which is a user/gateway (D) and which is a node (N)

18.7.4 STM32 C program, Update of 18/09/2024

Version name: CNA_Test_Node_V1

The code is written so that another developers can easily modify and add features to the CNA system. The core parts of the code are "app.h" and "app.c" files.

18.7.4.1 app.h

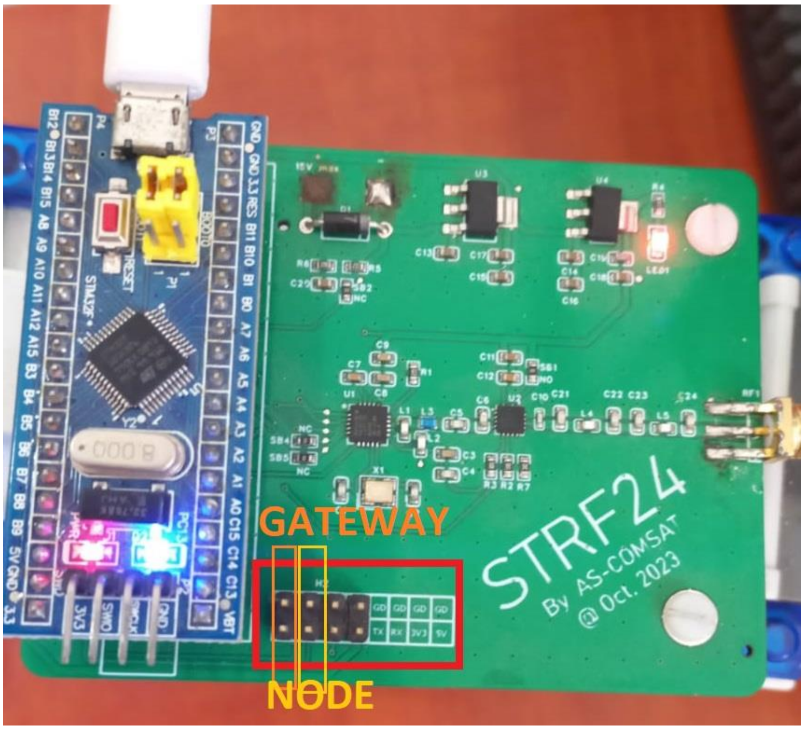
```
/*
```

```
 * app.h

 *

 *  Created on: Sep 10, 2024

 *      Author: Raja

 */



#ifndef INC_APP_H_

#define INC_APP_H_



#include "usbd_cdc_if.h"

#include "string.h"

#include "stdbool.h"

#include "trx_interface.h"

#include "board_type.h"



#define NRF24L01P_PAYLOAD_LENGTH 32



enum {
```

```
        _ID_FRAME = 0x1A,

        _COORDINATES_FRAME = 0x50,

        _CNA_MESSAGE_FRAME = 0x60,

        _CMD_FRAME = 0x70,

        _DEBUG_FRAME = 0x80,



        _SW_VER = 100,

};



// Function that manages the application

void AppHandler(board_type_t board_mode);



// Declare the callback function to process received data

void AppProcessReceivedUSBData(uint8_t *data, uint16_t length);



// Function to process the IRQ interrupt of nrf24l01

void processIRQ_Callback(void);

#endif /* INC_APP_H_ */
```

This header file is the only header that should be modified by the developers. It has 3 main functions: AppHandler, AppProcessReceivedUSBData, and processORQ_Callback.

* void AppHandler(board_type_t board_mode): This is the main loop of the software. It navigates between all the states of the system. The states are defined in a "typedef enum app_states". 3 different board types can be configured (GATEWAY, NODE, or USER) using the header pins located on the board (see figure below). If the orange header pins are connected together, the board is configured as GATEWAY. And if the yellow headers are connected together, the board behaves as NODE. If both headers are not connected, the board behaves as a USER board. The system keeps looping between its states according to some external conditions (such as: new usb data received, RF data received, errors in reading, and so on..).




```
typedef enum app_states {

        _RF_RECEIVE,

        _TRANSMIT,

        _ERROR,

        _IDLE,

        _SET_ADDRESSES,
```

```
        _HANDLE_ID,

        _HANDLE_COORDINATES,

        _HANDLE_USB_MESSAGE,

        _HANDLE_CMD,

        _HANDLE_SW_VER,

        _HANDLE__DEBUG

} app_state;



// The main loop of the application.

void AppHandler(board_type_t board_mode) {



        if (!addresses_set)

                current_state = _SET_ADDRESSES;



        if (nrf24l01p_get_comm_mode() != _RX) {

                nrf24l01p_prx_mode(); // switch back to RX mode

        }
```

```
        switch (current_state) {

        case _HANDLE_ID:

                read_multiple_bytes(NRF24L01P_REG_TX_ADDR, addresses,

                                sizeof(addresses));



                CDC_Transmit_FS(addresses, sizeof(addresses));



                current_state = _IDLE;

                break;



        case _SET_ADDRESSES:



                unique_id[0] = HAL_GetUIDw0();

                unique_id[1] = HAL_GetUIDw1();

                unique_id[2] = HAL_GetUIDw2();



                // Copy the first 5 bytes of the unique ID into uid[5]

                uid[0] = (uint8_t) (unique_id[0] >> 24) & 0xFF; // Most significant
byte of the first 32-bit word
```

```
                uid[1] = (uint8_t) (unique_id[0] >> 16) & 0xFF; // Next byte

                uid[2] = (uint8_t) (unique_id[0] >> 8) & 0xFF;  // Next byte

                uid[3] = unique_id[0] & 0xFF; // Least significant byte of the first
32-bit word

                uid[4] = (uint8_t) (unique_id[1] >> 24) & 0xFF; // Most significant
byte of the second 32-bit word



                addresses_set = true;

                current_state = _IDLE;

                break;



        case _HANDLE_USB_MESSAGE:

                memset(rf_tx_pld, 0, sizeof(rf_tx_pld));



                memcpy(rf_tx_pld, usb_rx_buffer, sizeof(rf_tx_pld));

                CDC_Transmit_FS(rf_tx_pld, sizeof(rf_tx_pld));



                nrf24l01p_ptx_mode(); // switch to TX mode
```

```
        nrf24l01p_write_tx_fifo(rf_tx_pld);



        current_state = _IDLE;

        break;



    case _RF_RECEIVE:

        CDC_Transmit_FS(rf_rx_pld, sizeof(rf_rx_pld));

        memset(rf_rx_pld, 0, sizeof(rf_rx_pld));



        current_state = _IDLE;

        break;



    case _HANDLE__DEBUG:

        memset(debug, 0x30, sizeof(debug));



        debug[0] = board_mode + 0x30;

        debug[1] = _SW_VER;

        debug[2] = nrf24l01p_get_fifo_status() + 0x30;
```

```
                CDC_Transmit_FS(debug, sizeof(debug));

                nrf24l01p_flush_tx_fifo();

                current_state = _IDLE;

                break;

        case _IDLE:

                break;

        }

}
```

* void AppProcessReceivedUSBData(uint8_t *data, uint16_t length): This function will be automatically called once new USB data are recieved. The function will save the incoming usb data in "usb_rx_buffer" and the received data length in "usb_data_len". The first byte received from the USB determines the desired function, thus, the "current_state" is updated accordingly. The frames can be seen in app.h.

```
uint8_t usb_rx_buffer[APP_TX_DATA_SIZE];

uint16_t usb_data_len;

void AppProcessReceivedUSBData(uint8_t *data, uint16_t length) {

        memset(usb_rx_buffer, 0, sizeof(usb_rx_buffer));
```

```
        memset(usb_tx_buffer, 0, sizeof(usb_tx_buffer));



        usb_data_len = length;

        memcpy(usb_rx_buffer, data, length);



        if (usb_rx_buffer[0] == _ID_FRAME)

                current_state = _HANDLE_ID;

        else if (usb_rx_buffer[0] == _COORDINATES_FRAME)

                current_state = _HANDLE_COORDINATES;

        else if (usb_rx_buffer[0] == _CNA_MESSAGE_FRAME)

                current_state = _HANDLE_USB_MESSAGE;

        else if (usb_rx_buffer[0] == _CMD_FRAME)

                current_state = _HANDLE_CMD;

        else if (usb_rx_buffer[0] == _DEBUG_FRAME)

                current_state = _HANDLE__DEBUG;

        else if (usb_rx_buffer[0] == _SW_VER)

                current_state = _HANDLE_SW_VER;

}
```

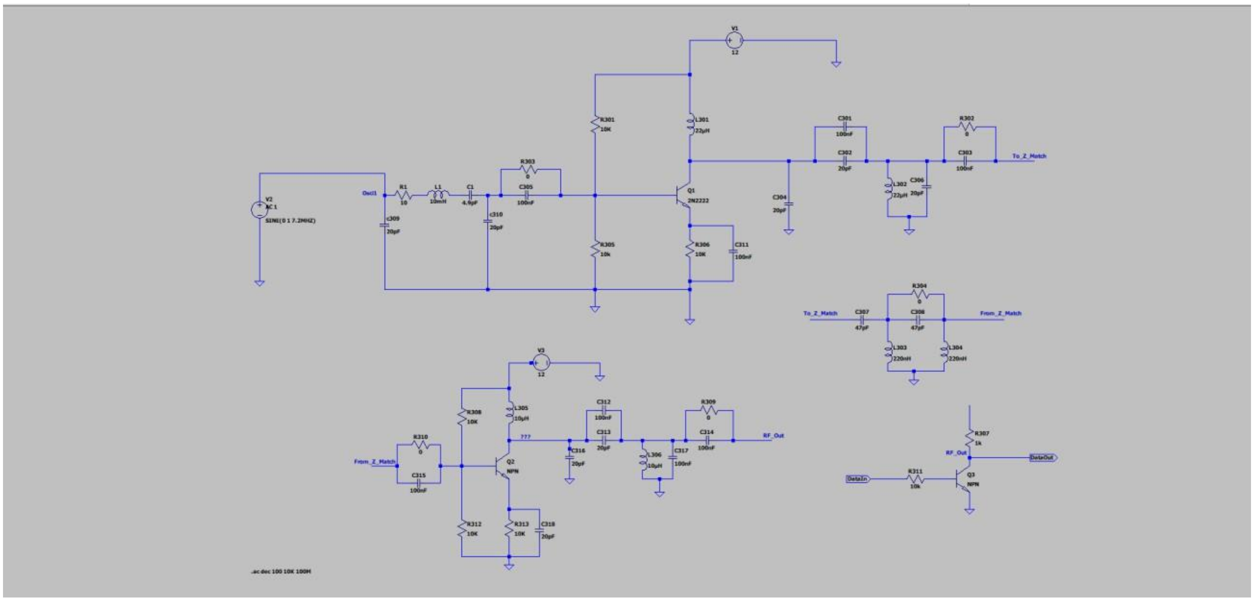
***** void processIRQ_Callback(void): This function is called automatically once a new packet is received via RF. The received packet will be saved in "rf_rx_pld" and can be accessed from the AppHandler. Once a new RF message is received, the current state is updated to RF receive which is handled in AppHandler cases.

```
uint8_t rf_rx_pld[NRF24L01P_PAYLOAD_LENGTH];

uint8_t rf_tx_pld[NRF24L01P_PAYLOAD_LENGTH];



void processIRQ_Callback() {



        uint8_t fifo_status = nrf24l01p_get_fifo_status();



        // if new message has arrived

        if ((fifo_status & 0x02) == 1 || (fifo_status & 0x01) == 0) {

                // read incoming message

                nrf24l01p_read_rx_fifo(rf_rx_pld);



                // clear rx fifo

                nrf24l01p_flush_rx_fifo();



                // update the application status

                current_state = _RF_RECEIVE;
```

```
        }

        HAL_GPIO_TogglePin(LED_GPIO_Port, LED_Pin);

        // clear interrupt flags

        nrf24l01p_clear_rx_dr();

        nrf24l01p_clear_tx_ds();

        nrf24l01p_clear_max_rt();

}
```

19 28.8 MHz Transceiver Development

We are working on a 28.8 MHz transceiver, and instead of directly using a 7.2 MHz crystal oscillator, I designed a circuit where R1, L1, and C1 work together to generate the 7.2 MHz signal. This signal is then processed through multiple stages to achieve the desired output frequency of 28.8 MHz.



19.1 Oscillator Stage (R1, L1, C1 - Replacing Crystal Oscillator)

- In a traditional oscillator, a 7.2 MHz crystal would set a precise frequency, but I designed a tuned circuit instead.
- R1, L1, and C1 create an LC tank circuit, which resonates at 7.2 MHz and sustains oscillations.
- L1 (10mH) and C1 (4.GpF) determine the oscillation frequency based on the formula: f = 1/2π rad(lc)
- C0G and C10 (20pF capacitors) help stabilize the frequency by reducing noise and unwanted harmonics.

19.2 Frequency Multiplier Stage (C0S, C10, R303):

- The output from the oscillator stage needs to be multiplied by 4 to reach
- 28.8 MHz.
- C09 and C10 help filter and amplify the fundamental 7.2 MHz signal while allowing higher harmonics to pass through.
- R303 is 0Ω, meaning it acts as a direct wire rather than a resistor. This ensures the signal is transferred without any voltage drop.
- Q1 (2N2222 Transistor) acts as an amplifier, boosting the signal strength so that it can drive the next stage effectively.

19.3 Amplification and Filtering Stage (L301, L302, C301):

- The transistor Q1 amplifies the signal, but it also introduces unwanted harmonics.
- L301 (22µH) and L302 (22µH) help in filtering out undesired frequencies and ensuring that the 28.8 MHz harmonic is dominant.
- C301 (100nF) and C302 (20pF) act as bypass capacitors, removing high- frequency noise from the power supply.

- R302 (0Ω) acts as another direct connection, ensuring maximum signal transfer.

19.4 Impedance Matching Network (C307, C308, L303, L304):

- Before sending the signal to the next stage, it needs to be properly
- matched to the circuit’s impedance.
- C307 and C308 (47pF each), along with L303 and L304 (220nH each), form a Pi-network to adjust the impedance and improve power transfer.
- This ensures that the signal is not distorted when it reaches the next stage.
- Modulation and Output Stage (Q2, Q3, C315):Q2 (NPN Transistor) is used to process the modulated signal, amplifying it before transmission.
- C315 (100nF) helps block DC components, allowing only AC signals to pass.
- R311 (10kΩ) and Q3 (NPN Transistor) act as a final signal processor before sending the output signal.
- The final RF signal is sent through RF_Out, which is the transmission point of the transceiver.

We have not conducted the simulation for this circuit at this time. However, once we complete the simulation, we will analyze the results and share them accordingly.

18.4.25:

دكتور بس عم خبرك شوي عن شو صار معنا : نحنا لتقينا مع الأستاذ محمود بالمركز الخميس وشفنا circuit عط يطلع في غلط بال output عطينا ال circuit للأستاذ محمود بعد ما عدلت عليها شوي هوي رح يشوف وين باقي المشاكل