

IHSAN DOGRAMACI BILKENT UNIVERSITY

ELECTRICAL AND ELECTRONICS ENGINEERING

DEPARTMENT



aselsan

Committee Meeting 4 Report

RF/Radio Signal Delta Position Calculation

Aselsan A.S., is a Turkish defense corporation headquartered in Ankara, Turkey. Its main operating area is research, development and manufacture of advanced military products for air, land and maritime forces.

The company is one of the major contractors of the Turkish Armed Forces.

Submission Date: 21 May 2025

Team Members

Orhan Eren Bıçakçı
İrem Bilgin
Zeynep Maden
Semih Özel
Müge Saribekiroğlu
Elif Beray Sarışık

Mentors

Course Coordinator: Mehmet Alper Kutay
Teaching Assistant: Ecem Şimşek
Academic Mentor: Tolga Mete Duman
Company Mentors: Tolga Sönmez
Company Mentors: Yiğiter Yüksel

List of Figures

1	Big Picture of the Project	12
2	Work Breakdown Structure of the Project	13
3	DLL Outputs For Cross Correlation Method	17
4	VCDL Approach	18
5	Examining Open Source GNSS Packages	19
6	Basic PLL Test Output	19
7	GNU Radio Implementation	20
8	1640/2 Hz Offset Detected During A Test	20
9	7Hz Frequency Offset Measured Over 15s	20
10	Received Signal After Coarse Frequency Correction	21
11	Zoomed Output of the PLL	21
12	Correlation of the PRN Code with PLL Output	22
13	The Measured Delay and Its Drift Over 4 Seconds	22
14	The measured delay and a linear fit of its drift over 4 seconds on top. Error of the fit on the bottom	22
15	Estimated Angle with respect to Sample i with Start and End of the Motion indicated	24
16	Computed Phase Difference in radian with respect to Sample i with Threshold $\epsilon = 0.15$ indicated	25
17	Estimated Delta Position in cm with respect to Sample i	25
18	Comparison of Settling Times of PLL vs Kalman Filter	26
19	Estimated Frequencies of Both Methods Compared	26
20	Steady-State Variances of Frequency Estimations	26
21	Phase Estimate from Kalman Filter.	27
22	Frequency Estimate from Kalman Filter.	27
23	Drift Estimate from Kalman Filter.	27
24	Phase Error of Kalman Filter	28
25	Phase Error during and around Movement	29
26	Phase Error and CFAR Threshold	29
27	Phase Error and CFAR Threshold around Movement	30
28	De-trended and Scaled Phase Plot where the Receiving SDR was Moved 36cms Away from the Transmitter.	31
29	Close up of the Scaled and Polynomial Fitted Phase Plot	31
30	Close up of the Scaled and Polynomial Fitted Phase Plot	32
31	The Circuit Schematic of the 40 MHz External Clock with 3.3 Linear Regulator . .	32

32	Adalm Pluto External Clock Input and Peripherals [31]	33
33	The Hardware Connection of the External Clock as a Clock Input to the SDR	33
34	Effect of Supply Variations on the Signal	34
35	Signal Waveform without Supply Noise	34
36	Received Signal Spectrum	35
37	Noise and Sample Skips in a Very-Low Frequency Signal	35
38	Signal with Three Phase Jumps Occurring within 30 ms	37
39	Effect of Phase Jumps on the Displacement Estimate (20 cm Actual Displacement) .	37
40	Estimation Error of the Kalman Filter and Errors Caused by Phase Jumps with Respect to the Error Threshold.	38
41	Effect of Phase Jumps on Frequency Estimate	39
42	Frequency Estimate when Updated Kalman Filter is Used	39
43	Effect of Phase Jumps on the Displacement Estimate with Updated Kalman Filter (20 cm Actual Displacement)	40
44	User Interface Showing Phase 1 Result	42
45	Flow Chart of the Final Design	43
46	Histogram of Error Percentages	44
47	Ground Truth vs Experimental Results	44
48	Big Picture of Phase 1 of the Project	57
49	Big Picture of Phase 2 of the Project	58
50	Gantt Chart 1/4	59
51	Gantt Chart 2/4	60
52	Gantt Chart 3/4	61
53	Gantt Chart 4/4	62

List of Tables

1	Phase 1 Error Summary	45
2	The Equipment List of the Project	49
3	The Functional System Requirements	54
4	The Signal Transmission and Reception Requirements	54
5	The Signal Processing Algorithm Requirements	55
6	The Distance Measurement Requirements	55
7	The Non-Functional System Requirements	55
8	The User-Related System Requirements	56
9	Theoretical and Experimental Distances Computed in the Setup with SMA Cable .	64

10	Theoretical and Experimental Distances Computed in Phase 1	66
----	--	----

Contents

1 Project Summary	1
2 Company Information	1
3 Motivation and Novelty	2
4 System Requirements Specifications	5
4.1 Functional Requirements	5
4.1.1 Transmission and Reception of Signal Requirements	6
4.1.2 Signal Processing Algorithm Requirements	6
4.1.3 Distance Measurement Requirements	7
4.2 Non-Functional Requirements	8
4.2.1 Cost	8
4.2.2 Environmental Interference	8
4.2.3 Temperature Range	9
4.2.4 Power Consumption	9
4.2.5 Safety Issues	9
4.2.6 Health Constraints	9
4.3 User-Related System Requirements	10
4.4 Standards of the Project	11
5 Big Picture	11
6 Methods and Implementation Details	13
6.1 Work Breakdown Structure and Project Plan	13
6.1.1 Work Packages	13
6.1.2 Milestones	15
6.2 Methods and Progress	16
6.2.1 Progress up to CM3	16
6.2.2 Progress After CM3	32
6.2.3 Risks Encountered	42
7 Results, Discussions, and Future Directions	43
7.1 Results	43
7.2 Discussions and Lessons Learned	46
7.3 Future Directions	48

1 Project Summary

The aim of this project is to design a radio-navigation subsystem capable of performing delta-position measurements using RF signals, offering a reliable alternative to GNSS-based systems that are vulnerable to signal distortion, multipath interference, and spoofing. Developed in collaboration with ASELSAN, the system is implemented in two phases using ADALM-Pluto Software-Defined Radio (SDR) devices. Phase 1 focuses on estimating one-dimensional (1D) displacement using two transmitters and a single receiver. Phase 2 extends the system to two-dimensional (2D) vectorial motion estimation through geometric reconstruction using three transmitters. Each transmitter is configured via MATLAB to emit signals at distinct center frequencies around 1.5 GHz with a bandwidth of 2 MHz. Signal separation is achieved through Frequency Division Multiple Access (FDMA), while phase and frequency synchronization are maintained using Phase-Locked Loop (PLL) and Delay-Locked Loop (DLL) structures. Kalman filtering is employed for robust estimation, effectively mitigating oscillator drift, noise, and sampling artifacts. To compensate for the frequency drift and instability of the SDRs' internal oscillators, high-precision external clocks are integrated, enabling coherent signal acquisition and enhanced synchronization between transmitters. Doppler-based methods are used to extract instantaneous velocity and compute displacement via integration. Algorithm development and signal processing workflows are conducted using MATLAB, GNU Radio, and Python. Controlled measurements are performed to evaluate system accuracy under varying operating conditions. A Raspberry Pi module is integrated with the system to create a compact, portable platform featuring an LCD screen for displacement display. The system is designed to maintain a maximum positioning error within ± 5 cm, providing a scalable and field-deployable navigation solution independent of GNSS.

2 Company Information

ASELSAN that has assigned the project is the leader of Turkey's defense industry. The company started its operation in 1975 to meet the communication needs of the Turkish Armed Forces. The company that is headquartered in Ankara, offers a wide range of products with its 10.500+ employees. ASELSAN operates in a wide range of fields which are both military and civilian such as defense electronics, communication systems, radar and electronic warfare, naval systems, energy, transportation etc. The operation in the company is separated into following five sectors:

1. Communication & Information Technologies
2. Microelectronics, Guidance & Electro-Optics
3. Radar & Electronic Warfare Systems
4. Defense Systems Technologies

5. Transportation, Security, Energy, Automation and Healthcare Systems

The company produces guided missiles, weapon systems, radar systems, electronic warfare tools, and communication systems to enhance resilience and natural security. Moreover, it develops civilian technologies such as energy management, transportation, healthcare, and security systems.

3 Motivation and Novelty

GNSS-based navigation systems are widely used and effective in many scenarios. However, they come with critical limitations that can impact their reliability and success. GNSS signals are highly vulnerable to atmospheric distortion. This can reduce signal accuracy, as well as multipath propagation, which occurs when signals are reflected off buildings or other large structures. This often results in significant errors, especially in urban environments or areas with dense structures. In addition, GNSS requires a clear line of sight to multiple satellites. In regions with a limited sky view, like deep canyons or forests, accurate positioning can be difficult to achieve [1]. Security concerns are also linked to GNSS-based structures. GNSS is suspectable to interferences, such as jamming and spoofing, which are critical issues in both civilian and military contexts. Jamming devices emit frequencies that disrupt GNSS signals, and can cause navigation systems to lose accuracy or even fail entirely. Spoofing can manipulate location data, posing risks to various industries that rely on accurate geolocation [2].

Since GNSS is an essential component of navigation for vital industries including transportation, logistics, and defense, these vulnerabilities create a serious worldwide issue. The weaknesses highlight the need for alternative navigation technologies that can operate independently of GNSS. As a solution, ASELSAN suggests a radio-navigation system using RF signals to create a more accessible and cost-effective solution. Such a system would not only enhance security by reducing reliance on vulnerable satellite networks but also improve accessibility for applications where satellite-based systems are impractical or too costly to deploy and maintain. Alternative systems like these could even have a broader application across industries providing reliable location data without the need for extensive GNSS infrastructure. It is aimed to be used by ASELSAN in defense and security applications without the need for a satellite connection.

The completed product is designed to function either as a standalone solution or as an adaptable component within ASELSAN's existing product lineup, enhancing the flexibility and reliance of its offerings across different sectors. After the project's completion, ASELSAN can further develop the system by integrating additional sensors such as IMUs to enhance positioning accuracy, or by updating the signal processing framework through software. The modular architecture also makes it suitable for integration into ASELSAN's broader platforms, including UAVs and autonomous

ground vehicles. This adaptability allows ASELSAN to broaden its applications and strengthen product versatility within both defense-focused and potentially commercial navigation solutions.

Although the primary end user of the final product will be in the defense sector and militaries, it could have users in civilian sector as well. Militaries need navigation systems that are reliable, robust, and safe enough to function in areas where GNSS is not available, such as military zones, high-security activities etc. The system's resilience to jamming and spoofing can provide operational success in high-security scenarios. Furthermore, the system has been developed to accommodate engineers and technicians who work for ASELSAN or related companies and will either maintain the solution's operation or incorporate it into larger systems. Additionally, the final product has important potential for civilian users in fields where GNSS constraints are an issue.

End-users' financial capabilities enable investment in specialized, mission-critical technologies, especially in the defense industry. The military and defense sector prioritize reliability, performance and security and puts costs of the systems at a lower priority even though cost-effectiveness is an important concern. This means that to be widely deployed across defense applications, the system needs to find a balance between price and high performance.

This project has been designed to be used among target customers who have different levels of technical ability. The system's easy-to-use interface and deployment process allow military personnel, who often have basic to intermediate technical training, to use it with minimal instruction. On the other hand, engineers and technicians provide the extensive technical knowledge needed for tasks like setup, maintenance, and integration into larger operational frameworks. These users can easily modify the system's components or incorporate it with pre-existing navigation systems thanks to its modular design. There are some products that can be alternative to GNSS-based navigation systems. SDR-Fi is the one of the alternative technologies. This solution uses SDR to achieve Wi-Fi like positioning, offering accurate location data with relatively low cost. However, SDR-Fi's precision can vary depending on the complexity of the environment, and it typically requires additional infrastructure to maintain consistent accuracy [3]. Honeywell's alternative navigation systems which use sensor-based technologies like LiDaR, radar, and star trackers to reduce reliance on GNSS is another alternative solution to the problem. High precision is provided by these systems, particularly in commercial and military settings where GNSS is not available. Nevertheless, these solutions are frequently expensive, necessitate intricate infrastructure, and are not widely accessible [4]. Similarly, under NAVISP program, the European Space Agency (ESA) promotes alternative and complementary Positioning, Navigation, and Timing (PNT) systems to GNSS. GNSS-independent technologies, like fiber optic networks and ground-based radio signals, have been developed under this program. Nevertheless, these initiatives are often successful at the local level and have a restricted range of applications in the industrial and defense sectors [5]. In addition to GNSS, hybrid systems made up of barometers, magnetometers, and inertial mea-

surement units (IMU) are employed in US military programs. When GNSS fails, these hybrid systems assure navigation continuity. However, because of its high cost and complexity, these solutions pose challenges for widespread use [6]. Another example of alternative solutions is used in some Finnish airports. These airports have implemented ground-based radio-navigation systems, which are an effective way to lessen GNSS interference. Despite guaranteeing continuous airport operations, these technologies are not appropriate for large scale applications [7].

When compared other alternative solutions, this project provides some advantages. Cost-effective Software Defined Radio (SDR) devices are used in this solution in place of expensive and complex sensor technologies like Honeywell and hybrid systems. A broader reach in both military and civilian sectors is made possible by this capability. Additionally, the system's modular design enables quick integration and user-friendliness in field operations. This solution can stand out thanks to its cost-effectiveness and modular structure as well as the complexities and costs of existing alternative systems. Moreover, the system of the SDRs does not require the implementation of complex infrastructures which makes the SDRs more advantageous and easier to use. The final advantage of using SDRs to calculate delta position could be the independence of the system to other variables such as Wi-fi, sensors and radars. All these factors make the development of the system more appealing.

The novelty of ASELSAN's system lies in its distinct approach to delta-position determination through RF signals, which differs from GNSS's full kinematic data acquisition. Instead of providing absolute positioning, it calculates relative movement and velocity, streamlining operations and reducing costs while maintaining high accuracy. This delta-position methodology not only simplifies the system design but also enhances its adaptability for diverse applications across defense, industrial, and commercial sectors. Existing solutions, while functional, do not address these unresolved challenges, particularly in situations where traditional positioning systems may fall short due to interference or high-security requirements.

The difficulties of navigation in GNSS-denied environments are addressed by number of international and national patented solutions. Utilizing techniques to reduce systematic errors in accelerometers, the Navigation System for GPS-Denied Environments (US Patent US10935670B2) enables precise navigation without the need for GPS signals [8]. Similar to this, in order to ensure reliability in the event that GNSS is not available, the Method and Apparatus for Satellite-Based Navigation Augmentation (US Patent US11598884B2) combines GNSS signals with alternative PNT data, such as inertial measurement units [9]. Similarities between these patents and this project include the use of different navigational strategies and the incorporation of sensors to either replace or enhance GNSS data. However, this project is centered on SDR technology, which provides cheaper and modular design.

Another patented solution to the problem is TerraFlite: Terrain Supported Autonomous Nav-

igation System which is created by STM for aerial platforms where GNSS cannot be used. This system matches important avionics sensor data with landscape features to generate high-precision position information [10]. While this project's solution can be used on land, in the air and on other platforms; TerraFlite is a solution designed exclusively for aerial platforms.

Because there are other navigation systems available, it might be difficult to get a patent for our project at this point. However, the fact that our project is based on solution using SDR technology and that such an approach is not yet widespread in the market increases our chances of obtaining a patent.

4 System Requirements Specifications

This final product will deliver both software and hardware components. The software deliverables consist of the algorithms for delta-position computation including both Phase 1 and 2D vectorial movement for Phase 2, error correction algorithms so as to reduce noise and oscillator drift, FDMA signal acquisition algorithms, tracking algorithms involving Kalman filtering and algorithms for movement detection.

For the hardware deliverables, there will be pre-programmed transmitters including ADALM-Pluto SDR for Phase 1 and Phase 2, as well as a receiver with ADALM-Pluto SDR, synchronized and configured simultaneously with a computer. These hardware components are illustrated with diagrams, and a block diagram of ADALM-Pluto is included to present a comprehensive hardware system. The final product will be capable of computing autonomously the delta position with respect to the initial position for 1D in phase 1 and 2D in phase 2 which is initiated by user control. The final product will be provided in a compact form by implementing it with Raspberry Pi module for portability and easiness for the end users. The project outlines functional, non-functional, and user-related system requirements specifications. The table format of the system requirements is provided in Appendix A, Table 3.

After CM3, there were many changes regarding functional requirements. With the delivery of the external clocks, we added external clock usage requirement to signal transmission requirements. Moreover, frequency division multiple access (FDMA) method is also added. To signal processing requirements, requirements related to PLL or DLL usage were deleted and instead requirements related to Kalman filtering is added.

4.1 Functional Requirements

The project's functional requirements #SRS-01 comprises three sub-requirements. The functional requirements include the requirements for signal transmission and reception, the signal processing

algorithm, and distance measurement (see Appendix A Tables 4, 5, 6 respectively).

- SRS-01-10 Transmission and Reception of Signals
- SRS-01-20 Signal Processing Algorithm
- SRS-01-30 Distance Measurement

4.1.1 Transmission and Reception of Signal Requirements

The project requires transmission and reception components that can operate at a frequency of 1.5 GHz and a minimum bandwidth of 2 MHz which is required by ASELSAN. A similar approach from literature is found as the GNSS-SDR project which operates at 1575.42 MHz and is designed to analyze GPS L1 signals [11]. The project also requires the use of FDMA signals. The signals transmitted from different SDRs will be differentiated using FDMA. A similar approach is used in SC-FDMA project, which employs frequency division to allocate unique subcarrier sets to individual users, enabling efficient communication in modern cellular networks [12]. Moreover, due to the instability of the internal clocks of the SDRs, external clock usage is another requirement of the project. External clocks oscillating at 40 MHz with 10 ppb frequency stability will be used. A similar approach is used in the SDRPlay-GPSDO setup, which integrates an external GPS-disciplined oscillator to enhance frequency accuracy and stability in SDR applications, enabling precise measurements for tasks like radio astronomy [13].

SRS-01-11	The RF transmission and reception components shall operate at a frequency of 1.5 GHz and provide a minimum bandwidth of 2 MHz to ensure effective signal transmission and reception.
SRS-01-12	The system shall utilize frequency division multiple access (FDMA) in both phases to differentiate between different transmitters.
SRS-01-13	The transmitter SDRs shall be driven with a 40 MHz, 10 ppb accuracy external clocks, to overcome the clock drift caused by the internal clock of the SDR devices.

4.1.2 Signal Processing Algorithm Requirements

The project's signal processing algorithm requirements are specified according to the modulation, phase locking, Doppler frequency tracking and movement detection. The project requires complex modulation at 1.5 GHz in order to differentiate signals. Moreover, the system requires low-pass filters after demodulating signals to be able to use the obtained signals in the Kalman filter.

In addition to modulation specifications, the Kalman filter used in the system must ensure rapid phase locking. Instead of PLLs, the Kalman filter is preferred. As it can be seen from previous studies in the context of phase tracking, Kalman filtering outperforms traditional phase-locked loops by adapting dynamically to system variations [14].

Moreover, we added a new requirement regarding Kalman filtering algorithms. To alleviate the effects of skipped samples, the Kalman filter must use an adaptive noise level to ignore the phase errors created by sample losses. As can be seen in the example from the CQG Kalman Filter, the recursive algorithm estimates the current state of a system by combining prior estimates with new measurements, effectively filtering out noise [15].

Another new requirement is added to ensure that the signal processing algorithms can find where the movement occurs and where it ends to clarify where the distance measurement should start and end. An example from literature highlights the use of an algorithm on accelerometer data to detect the start of the motion by checking the deviations from the mean of the data [16].

SRS-01-21	The system shall utilize complex modulation at a frequency of 1.5 GHz.
SRS-01-22	The system shall recover the modulated signals using low pass filters.
SRS-01-23	The system shall implement phase locking mechanisms capable of rapid signal locking within 2 seconds.
SRS-01-24	The system shall use Kalman Filtering, to alleviate the effects of skipped samples and differentiate the doppler frequency from the frequency offset.
SRS-01-25	The system shall be clearly differentiate the physical displacement of the SDR from the phase discrepancy of the SDR, ensuring that these dispositions are easily identifiable for accurate monitoring and precise control throughout the entire process.

4.1.3 Distance Measurement Requirements

The requirement for the distance measurement, which is set by ASELSAN, specifies measuring the distance between two points (A and B) with an accuracy of ± 5 cm (1-sigma error). The distance measurement requirement specification is a critical criterion for assessing project success.

This requirement ensures the system's effectiveness in precision measurements, a key performance indicator. For comparison, the SDR-Fi project achieves a mean distance accuracy of 0.99 meters and a 50th percentile accuracy of 0.77 meters [3], while another SDR-based project has demonstrated high precision with an accuracy consistently within 1.2 cm, a mean error of 6 mm, and a standard deviation of 2.2 mm [17]. This example implementation signifies the importance

of sustaining ASELSAN's accuracy standards to validate the project's success in practical applications.

- | | |
|-----------|--|
| SRS-01-31 | The system shall calculate the one-dimensional distance between two points (A and B) with an accuracy of ± 5 cm (1-sigma error) to ensure high-precision measurements. |
| SRS-01-32 | The system shall calculate the one-dimensional distance between points with an accuracy of ± 5 cm (1-sigma error) to meet precision requirements. |

4.2 Non-Functional Requirements

The non-functional constraints of the final product #SRS-02 includes the requirements for the cost, environmental interference, temperature range to ensure reliable operation, power consumption, safety issues, and health constraints (see Appendix A Table 7). The project does not include non-functional requirements for size, weight, or social factors, since ASELSAN has not provided requirements regarding the specific use case of the final product.

4.2.1 Cost

- | | |
|-----------|--|
| SRS-02-10 | The maximum allowable cost of the project is 50,000 Turkish Liras, as specified by ASELSAN. Up to the committee meeting, approximately 30,000 Turkish Liras of the budget has been utilized to acquire three ADALM-Pluto SDRs to initiate the distance calculation procedures. The estimated total cost of the final product is projected to be around 40,000 Turkish Liras. |
|-----------|--|

4.2.2 Environmental Interference

- | | |
|-----------|---|
| SRS-02-20 | Environmental interference caused substantial challenges during transmission and reception testing, especially in urban and indoor environments with considerably high RF noise. This interference has an impact on signal quality and accuracy, emphasizing the need for strong noise tolerance measures to maintain reliable operation. Therefore, a specification about high precision is required to enable RF noise tolerance tests and avoid poor performance in demanding urban and indoor contexts. |
|-----------|---|

4.2.3 Temperature Range

- SRS-02-30 With respect to the ADALM-PLUTO SDR requirements, the system is intended to operate reliably throughout a temperature range of 0°C to 70°C (32°F to 158°F), allowing it to function efficiently in both indoor and outdoor conditions [18]. Although ADALM-PLUTO's humidity tolerance is not specified, it is recommended to avoid high humidity and direct moisture exposure for maximum performance.

4.2.4 Power Consumption

- SRS-02-40 The system's power consumption ranges from 1.2W to 2.5W during operation, translating to approximately 240mA to 500mA at 5V. In Phase 1, this applies to three units (x3), while in Phase 2, four units (x4) will be used, increasing overall power requirements [9]. Additionally, the real-time processing of the computer can consume power in the range from 100W to 400W [18]. Consequently, the maximum power consumption of the final product will be approximately 420W.

4.2.5 Safety Issues

- SRS-02-50 The design of the product considers safety over both electrical and radio frequency exposure factors. All electronic components must follow specific safety standards to avoid hazards such as electric shocks, overheating, and potential malfunctions when subjected to variable power conditions. Besides, RF transmission power will be rigorously managed to ensure that it remains below acceptable limits, in accordance with regulatory regulations governing human exposure to RF fields. This can be crucial for longer exposure to the system, since persistent adherence to RF safety requirements shields users from the potential health concerns associated with long-term RF exposure.

4.2.6 Health Constraints

- SRS-02-60 The design of the project does not directly violate any health constraints, but the RF exposure levels must be considered and kept within safe limits, according to guidelines issued by the International Commission on Non-Ionizing Radiation Protection (ICNIRP) or comparable organizations. This

procedure alleviates the possible health concerns about the longer exposure to the final product.

4.3 User-Related System Requirements

The user-related constraints of the final product #SRS-03 includes the requirements for user control, portability of the system, accessibility, so as to maintain the expectations of the end users by sustaining operational effectiveness. No additional user-related requirements are included, as ASELSAN has not specified any further criteria beyond those stated. User-related system requirements consist of constraints about user control, portability, monitoring, accessibility, and documentation and support (see Appendix A, Table 8).

- | | |
|-----------|--|
| SRS-03-10 | The system shall provide users with an easy interface for initiating and controlling signal transmission, receiving, and processing activities. The interface must allow users to effortlessly transition between 1D and 2D distance measuring modes. An example from the literature is the GNSS-SDR platform provides a graphical user interface (GUI) allowing users to configure signal processing parameters [19]. |
| SRS-03-20 | The system shall display calculated distances on the LCD screen connected to Raspberry Pi enabling users a clear and accessible format to monitor system performance effectively. Moreover, the signal transmission should be controlled by the Raspberry-pi module. To exemplify it from the literature, the Raspberry Pi GPS Distance Tracker Project demonstrates how distances are calculated and displayed on an LCD screen connected to the Raspberry Pi, providing users with real-time feedback on system performance [20]. |
| SRS-03-30 | The system shall be designed to be lightweight and compact, ensuring portability for users. The final product's dimensions and weight shall allow it to be easily transported and deployed in various environments. Indeed, this portability is critical for enabling users to calculate distances effectively in field conditions without the need for extensive setup. An example of portability in literature can be found in the Ettus Research USRP B200 Mini SDR, which emphasizes a compact and lightweight design weighing only 280 grams [21]. Nonetheless, any related example for positioning setup is not found. |

SRS-03-40 The system shall include related user documentation in order to provide guidance on setup, operation, and troubleshooting. An example of user documentation designed to provide guidance on setup, operation, and troubleshooting can be found in the RTL-SDR Blog V3 Software Defined Radio Documentation [22].

4.4 Standards of the Project

Due to the fact that the hardware deliverables of the project are mainly combination system of ADALM-Pluto Software Defined Radios, the standards of the project coincides with the ADALM-Pluto Regulatory Compliance [23]. The device is complied with the requirements of a Class B digital device following the FCC Rules in Part 15, which approves the usage and installation of the device within an environment. Apart from its FCC statement, the device acquires Industry Canada licence exempt in RSS standards, which states that device meets the requirement for CAN ICES-3(B)/NMB-3(B).

Additionally, the overall system satisfies the safety requirement for the testing, measurement, control, and laboratory usage of the electrical equipments provided in IEC 61010-1, UL/CSA 60950-1 [24]. Since the operational frequency of the project which is 1575.42 MHz included in the range of 0 Hz to 300GHz, the device and the overall hardware system complies with IEEE C95.1, which highlights the standard for the safety levels of human exposure the magnetic, electric, and electromagnetic field within the range [25]. Besides, the device also sustains a declaration of conformity with respect to EN ISO/IEC 17050-1:2010 [26]. Furthermore, the hardware components adhere to the IS-GPS-200N standard, which defines the interface requirements between the GPS space segment and the navigation user segment, ensuring compatibility with GPS signal structures and protocols [27].

Apart from the safety regulations, the technical standards for the wireless communication IEEE 802.11 is also highlighted within the final product so as to acquire an architecture within the standards of the 802.11 [28]. Consequently, due to the limited range of products utilized in the project, mostly the standards of the device itself are followed in addition to technical standards listed.

5 Big Picture

The Big Picture of the project is provided in the Figure 1. A larger version of the Big Picture of the *Phase 2* can be seen in Appendix B. Additionally, a larger version of the Big Picture for *Phase 1*, which has been successfully implemented, can be found in Appendix B. The Big Picture demonstrates *Phase 2* of the project, which involves calculating the total distance in a two-dimensional

environment using four ADALM-PLUTO SDRs: one for receiving purposes and three for transmitting purposes (see Figure 1). The transmitter SDRs are preprogrammed using MATLAB on a computer via the USB 2.0 protocol. Additionally, the SDRs, powered by a 5V supply, continuously transmit pure tone signals around 1.5 GHz center frequency. The receiver SDR, which is in motion, is initially connected to a computer via the USB 2.0 protocol for power supply and algorithm development. Later, it will be directly connected to a Raspberry Pi module via a Micro-USB to USB-A connection, allowing for an easily transportable hardware setup.

Initially, the SDR is connected to a computer to facilitate the analysis and implementation of algorithms supported by the GNSS SDR package, which is downloaded from GitHub. During the algorithm development stage, GNU Radio and MATLAB are used concurrently to ensure comprehensive software development. Once the coding is optimized for real-time processing, it is implemented in Python, and a user interface via HDMI protocol is developed for demonstrating the script.

Finally, a Raspberry Pi module is utilized to meet the end-user requirements by providing a portable device. The Raspberry Pi module, powered by a 9V power supply, is connected to an LCD screen via GPIO to visualize the distance traveled by the hardware block. The Raspberry Pi is pre-programmed by connecting it to a computer via Ethernet.

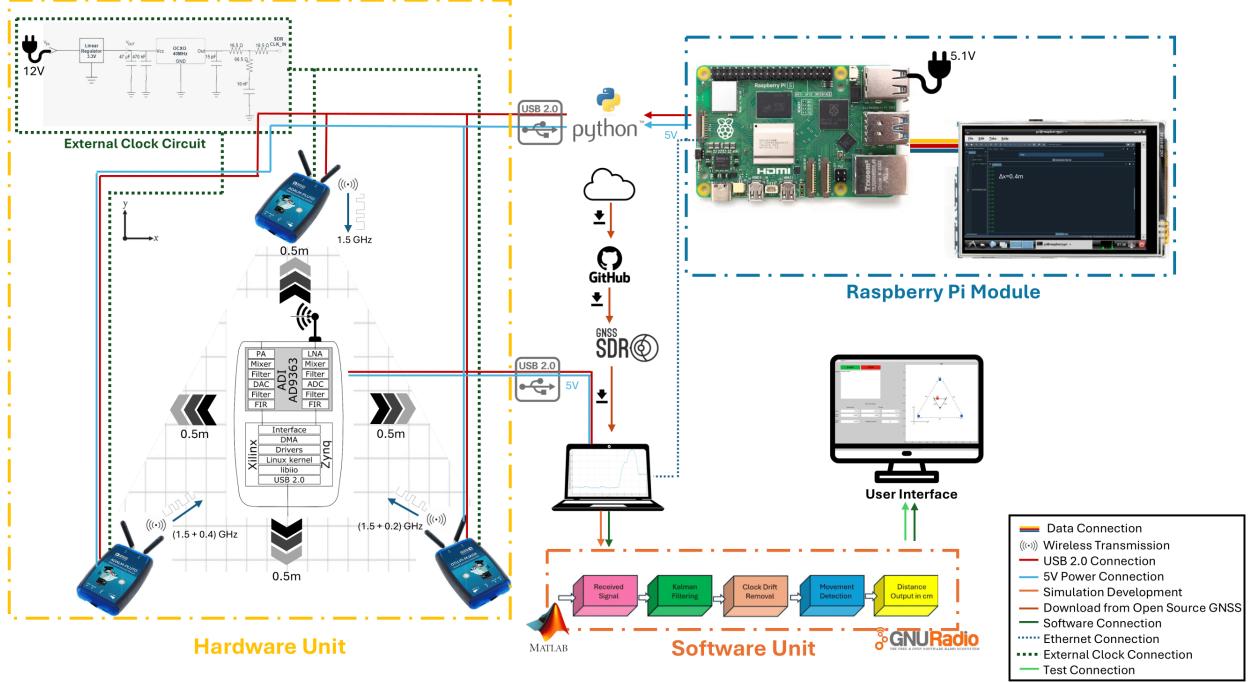


Figure 1: Big Picture of the Project

6 Methods and Implementation Details

6.1 Work Breakdown Structure and Project Plan

Work breakdown structure (WBS) of our project consists of 7 main work packages. The general structure of the WBS can be seen in Figure 2. After CM3, some changes were made in work packages 5 and 6, which are explained in the following section. The detailed version of the Gantt Chart provided in Appendix C.

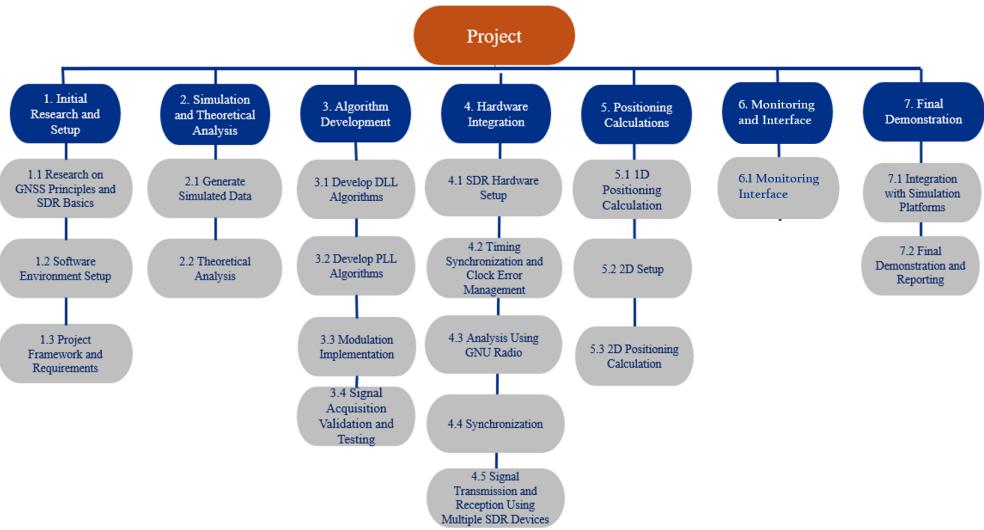


Figure 2: Work Breakdown Structure of the Project

6.1.1 Work Packages

- Package 1: First work package includes the initial research steps. It consists of three subsections, which are 1.1, research on GNSS principles and SDR basics, 1.2 , software environment setup and 1.3, project framework and requirements. For WBS 1.1, the GNSS signal structures were investigated, and several DLL and PLL algorithms were tested out for existing GNSS-SDR projects. For WBS 1.2, all team members installed the necessary toolboxes for MATLAB and Simulink implementations, which will then be used in signal processing. In WBS 1.3, a general specification list for the project is determined based on the literature review that has been carried out.
- Package 2: The Second work package includes the early simulation and data analysis parts, and consists of two subsections, which are 2.1, generation of simulated data and 2.2, theoretical analysis. For WBS 2.1, using MATLAB, intermediate frequency samples were created and using these samples, the SDR working principles were investigated. For WBS 2.2, the

theory of delay-locked loop and phase-locked loop algorithms was studied, and the necessary equations were derived.

- Package 3: Algorithm development includes four subsections, which are 3.1, DLL algorithm development, 3.2, PLL algorithm development, 3.3, modulation implementation and 3.4, signal acquisition validation and testing. For WBS 3.1, two DLL algorithms were tested out to minimize the delay difference between the received and reference signals. First uses a voltage-controlled delay line method, and the other uses the cross-correlation method. For WBS 3.2, the PLL algorithm was implemented and tested on the data collected from the SDR. For WBS 3.3, signal acquisition using two SDRS, properties of binary phase shift keying modulation were investigated, and a frequency constellation correction method was implemented. For the last subsection for this package, 3.4, validation of PLL and DLL algorithms was tested on the data collected from SDRs. Later, research on calculating the delta position using PLL and DLL outputs was conducted.
- Package 4: Hardware integration is broken into five subsections. In WBS 4.1, the hardware setup of the ADALM-Pluto is completed. This includes the acquisition and configuration of the SDR, installing of the necessary drivers and software, verifying basic operations and transmission and reception, and calibration of the hardware to have an accurate frequency response. WBS 4.2 focuses on timing synchronization and clock error management. Clock drift and possible error sources are analyzed. Then, several possible solutions are researched. These solutions include a shared external clock, shared GPS, utilizing a beacon SDR, and an algorithm to compensate for the clock offset. In WBS 4.3, GNU Radio analysis are executed. Signal is transmitted and received, and PLL and local oscillator frequency offset correction are tried to manage clock errors. In addition, how to do offline simulations is researched. In WBS 4.4, aim is to manage synchronization. PLL, averaging, polynomial fit are the utilized methods. After these implementations, synchronization accuracy is calculated. WBS 4.5 consists of the transmission and reception of the signal using multiple SDR's. It includes correlation check, frequency offset removal using FFT, and stabilization with PLL and DLL algorithms to obtain suitable signals for positioning calculations. WBS 4.6 consists of supplementary assignment which is provided by ASELSAN to provide an alternative plan for the project in case of incompleteness of the project which includes delta position estimation by using one SDR and 50cm SMA cable for the movement of the receiver antenna which was also demonstrated in CM3 as the first setup.
- Package 5: Positioning calculation has three subsections. In WBS 5.1, 1D positioning calculation will be achieved. Linear delta-position algorithms will be implemented and tested for accuracy. Error will be minimized to achieve the accuracy criteria. After CM3, we added external clock synchronization step in this work package as well to ensure the clock drift is

minimized. Last item in this subsection is the detection of start and end points of the motion to make sure that the algorithm captures the movement automatically. WBS 5.2 consists of the setup of the 2D system. Optimal signal transmission and reception in 2D environment will be managed, and clocks will be synchronized. Moreover, Kalman filter parameters will be fine tuned to minimize error. In WBS 5.3, 2D position calculation will be executed. After CM3 we decided to remove DLL algorithms and instead add FDMA method to this package. FDMA algorithm will be used to differentiate between the transmitters and then the vectorial distance traveled will be calculated. Lastly, the performance of the system will be evaluated via experiments.

- Package 6: This package included real-time signal processing. After CM3 we decided to make the real-time processing an alternative plan and focus more on the regular monitoring of the movement. Monitoring and integration section consists of one section. In WBS 6.1 a user interface will be implemented. The distance traveled will be displayed on a monitor and the user will be able to control the signal reception. A Raspberry-pi module will be programmed to handle signal transmission from the SDRs and this Raspberry-pi module will be connected to an LCD screen will show the distance traveled. This will improve the systems portability.
- Package 7: Final implementation and demonstration has one subsection. In WBS 7.1 final demonstration and reporting of the project will be completed. A detailed demonstration plan will be prepared. Feedback will be collected based on the demonstration, and the final report will be written.

6.1.2 Milestones

For our project we determined 4 milestones.

- First Milestone: The first milestone is related to the algorithm development. This milestone includes the successful implementation of the PLL, DLL and signal acquisition algorithms. This milestone include the work packages 3.1, 3.2, 3.3 and 3.4 which are, DLL algorithm development, PLL algorithm development, modulation implementation and signal acquisition validation and testing using only one SDR and later two SDRs. The due date for this milestone was 15/12/24 and we have successfully completed our signal acquisition, PLL and DLL algorithms for the multiple SDR case as well, so this milestone can be considered successful.
- Second Milestone: The second milestone of the project is to achieve 1D positioning calculation, regarded as Phase 1 of the project. Two SDR's will be utilized simultaneously to achieve a linear delta-position calculation. This milestone is considered sufficient to meet

ASELSAN's project requirements. The criterion for success is to achieve a positioning accuracy within the range of +/- 5 cm. The milestones' success will also be evaluated based on the completion of the following WBS packages: 5.1.1, 5.1.2, 5.1.3, 5.1.4. The preliminary due date for the third milestone is set as 11/04/25.

- Third Milestone: The third milestone is related to a major problem that is faced while working with multiple SDRs, which is the synchronization problem. While using two SDRs due to the clock and phase difference between the SDRs, the data obtained from the receiver is not suitable for delta-position calculation. This problem will be solved by utilizing different methods to eliminate this difference. The criteria of success for this milestone will be the SDRs being synchronized. This milestone includes the work packages 4.2, 4.4 and 5.1.5 which are timing synchronization and clock error management, and synchronization of the SDRs using external clocks. Due to the complexity and the importance of the problem the due date for this milestone is arranged as 27/04/25.
- Fourth Milestone: The fourth milestone of the project is achieved with the implementation of a monitoring system for both Phase-1 and Phase-2. The success of this milestone is defined by the system's ability to display results with minimum error. This milestone validates performance by checking the accuracy of the distance measurement of the system for both 1D and 2D movement and checking the displayed data on the monitor and the LCD screen. Achieving this milestone is challenging due to the operational characteristics of the ADALM-Pluto SDR. To ensure the success of the fourth milestone, WBS packages 5.2, 5.3 and 6.1 must completed. The due date for the fourth milestone is decided to be 16/05/25.

6.2 Methods and Progress

6.2.1 Progress up to CM3

1. Initial Research and Setup

A solid project foundation was established through extensive research on GNSS principles, SDR technology, and PLL/DLL structures. Open-source GNSS-SDR tools and relevant literature were explored for technique selection and improvements. Development tools such as MATLAB, Simulink, and GNU Radio were configured, while project requirements, risk assessments, and version control mechanisms were also finalized to streamline collaboration.

2. Simulation and Theoretical Analysis

CDMA Codes: The CDMA codes were implemented for the use of the same channel among multiple transmitters and synchronization. The phase shift of periodically transmitted CDMA codes is estimated to measure the distance variations between the transmitter

and receiver. The receiver calculates the delay through circular cross-correlation between the received and reference codes. Each transmitter utilises a unique CDMA code from a predefined set known to the receiver. This unique code enables signal separation even when multiple transmissions occupy the channel.

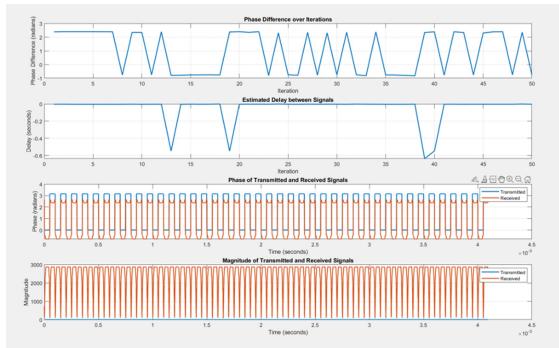
MATLAB Simulation: The algorithm's performance was tested with generated MATLAB IF data before real SDR data. The simulation environment incorporates three essential elements: Implementation of binary phase shift keying (BPSK), Phase-Locked Loop (PLL) design, and Delay-Locked Loop (DLL) algorithm. Algorithms were optimized before deploying and testing on hardware.

3. Algorithm Development

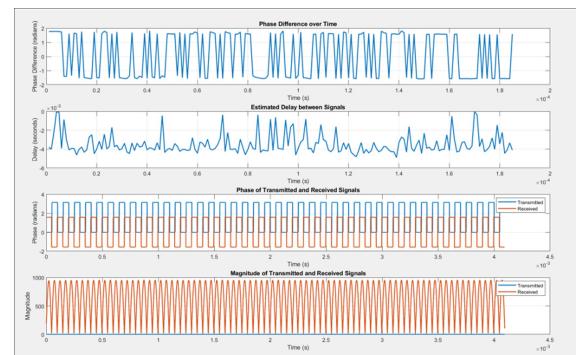
DLL: Under the WBS 3.1 two distinct DLL approaches were tested:

System Principles and Initial Research (CM1 Phase): The DLL synchronizes input and output signals by adjusting delay. It consists of a delay line, a phase detector, a loop filter, and a voltage-controlled oscillator. In our approach, the phase difference is computed by cross-correlating the input and the delayed output signals, it attains the maximum value where the signals are in phase. An error signal is processed by the loop filter, generating a control signal and adjusting the delay until the phase difference is below the required threshold.

Implementation and Refinement (CM2 Phase): The first approach used cross-correlation to compare the transmitted and the received signals with a single SDR. More delay is added based on the index of the maximum cross-correlation. Figure 3a and Figure 3b display the outputs for the cases where the antennas are connected with a wire and not connected, respectively.



(a) Output of DLL When Transmitter and Receiver were Connected



(b) Output of DLL with Antennas

Figure 3: DLL Outputs For Cross Correlation Method

The first model could not cancel the delay sufficiently when the transmitter and receiver are not connected directly. Therefore, we tried another method by directly implementing the circuit schematic in Figure 4a. This implementation adds a delay to the system based on the error signal. Figure 4b shows the phase difference goes to zero over iterations.

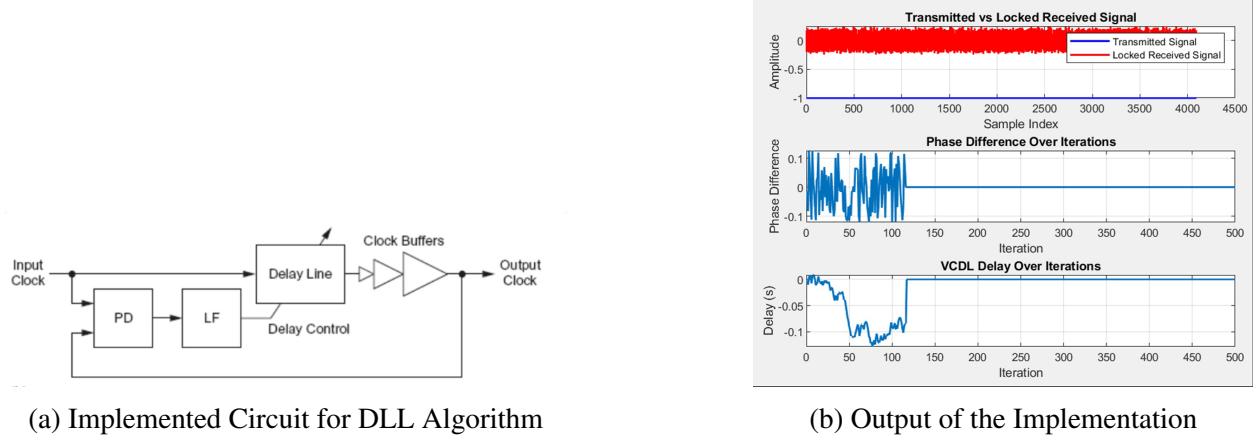


Figure 4: VCDL Approach

PLL

System Principles and Initial Research (CM1 Phase): A phase-locked loop (PLL) is a critical control system designed to synchronize the phase of an input signal with a reference signal by continuously adjusting their phase difference. Our initial design made use of this property and focused on creating a framework for frequency and phase synchronization between the transmitter and receiver. We began with theoretical analysis and continued with simulations to understand the structure. The binary discrete signal $m[n]$ is transmitted as $m(nT_s) \cos(2\pi f_c t)$, where T_s is the sampling period, f_c is the center frequency, and $m[n] \in \{-1, 1\}$. The received signal, accounting for oscillator discrepancies, is expressed as $r(t) = m(nT_s) \cos(2\pi f_c t) [\cos(2\pi(f_c + f_\Delta)t + \theta) - j \sin(2\pi(f_c + f_\Delta)t + \theta)]$, where f_Δ represents the frequency offset and θ the phase offset. The baseband message is given by $r[n] = m[n] \exp(j2\pi f_\Delta nT_s + j\theta)$.

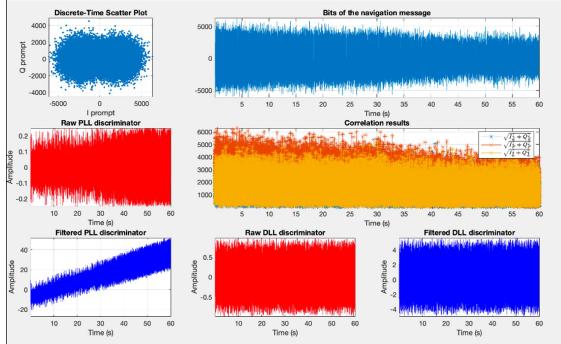


Figure 5: Examining Open Source GNSS Packages

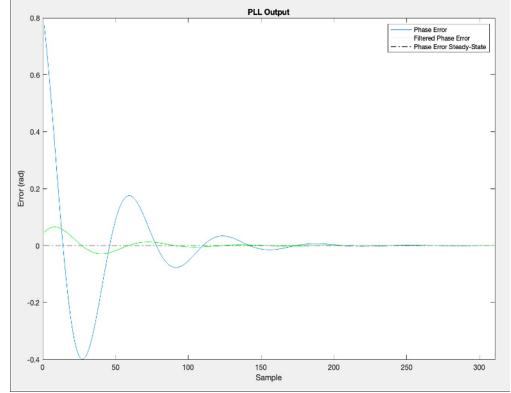
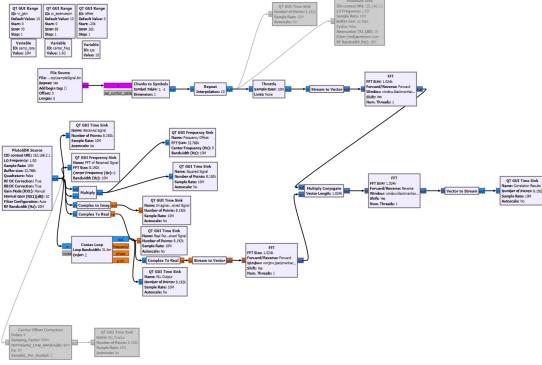


Figure 6: Basic PLL Test Output

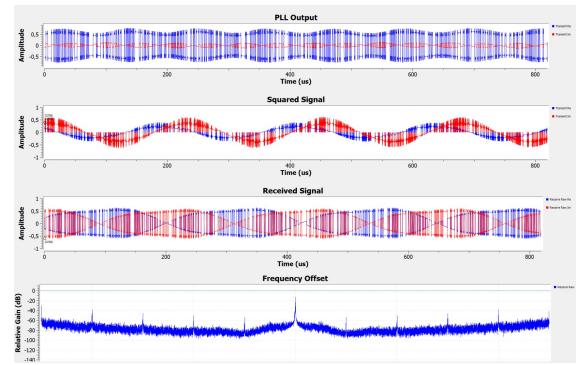
During the CM1 phase, we researched PLL algorithms in open-source GNSS implementations (Figure 5) before developing our own solutions. Initial testing with basic signals demonstrated successful convergence, with phase errors approaching zero over successive samples and reaching steady state (Figure 6). These results validated our fundamental approach.

Implementation and Refinement (CM2 Phase): Our developed PLL implementation was based on a basic structure consisting of three key components, shown in Figure 7. The phase error detector block computes the error signal using the function, $e[n] = \text{sgn}(\Re\{y[n]\}) \cdot \Im\{y[n]\}$. This function utilizes the properties of BPSK-modulated signals. The error signal is processed through a loop filter implemented as a PI controller. Gains are tuned carefully to reach an optimal performance regarding the pull-in range, locking speed and the variance of the error after locking. The filtered signal drives a direct digital synthesizer (DDS) adjusting the reference signal's phase.

GNU Radio Implementation: We have also used GNU Radio to implement the procedure. We prepared to transition to GNU Radio for faster programming and real-time implementation. Signal frequency offset detection and frequency recovery with PLL has been implemented in GNU Radio.



(a) GNU Radio Block Diagram



(b) Output of the Implementation

Figure 7: GNU Radio Implementation

Figure 7b shows the frequency recovered signal, the raw signal received, the squared signal used for frequency offset detection, and the spectrum of the signal, from the top to the bottom plot.

4. Hardware Integration

Local Oscillator Offset: In SDRs' local oscillators there exists frequency offsets that modifies the baseband BPSK signals according to $r[n] = x[n] \exp(j2\pi f_\Delta nT_s + j\theta)$, where f_Δ represents the frequency offset and θ the phase offset. To detect these offsets, we utilize the BPSK signal property $x[n] \in \{-1, 1\}$ by squaring the received signal: $r^2[n] = \exp(j4\pi f_\Delta nT_s + j2\theta)$. An FFT of this squared signal results in the frequency offset components. The ADALM-Pluto SDR's clock has specified 25 ppm frequency stability [29] translates to a maximum ± 37.5 KHz offset at our 1.5 GHz center frequency. Our measurements showed typical offsets ranging from 100 Hz to 20 kHz, with observed drift of 7 Hz over 15 seconds (Figure 9). This non-stationary behavior requires continuous offset tracking in our implementation.

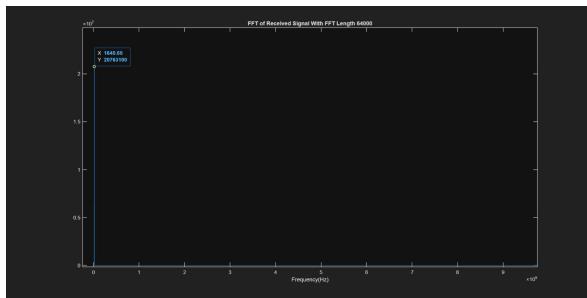


Figure 8: 1640/2 Hz Offset Detected During A Test

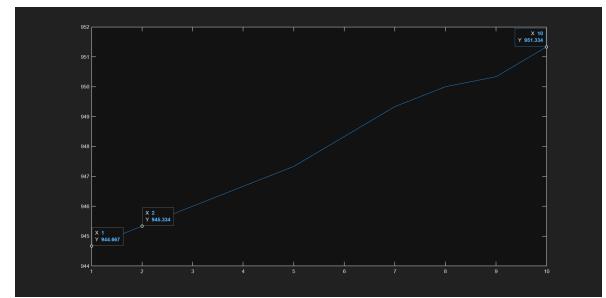


Figure 9: 7Hz Frequency Offset Measured Over 15s

Coarse Frequency Correction: Using the FFT method, we remove the complex envelope by multiplying the received signal with the conjugate of the detected complex envelope. However, due to the FFT's finite resolution, residual frequency offsets smaller than FFT resolution remain after this coarse correction. As visible in Figure 10, the top zoomed plots show successful removal of high-frequency modulation, while the bottom plots reveal low-level offsets over longer periods.

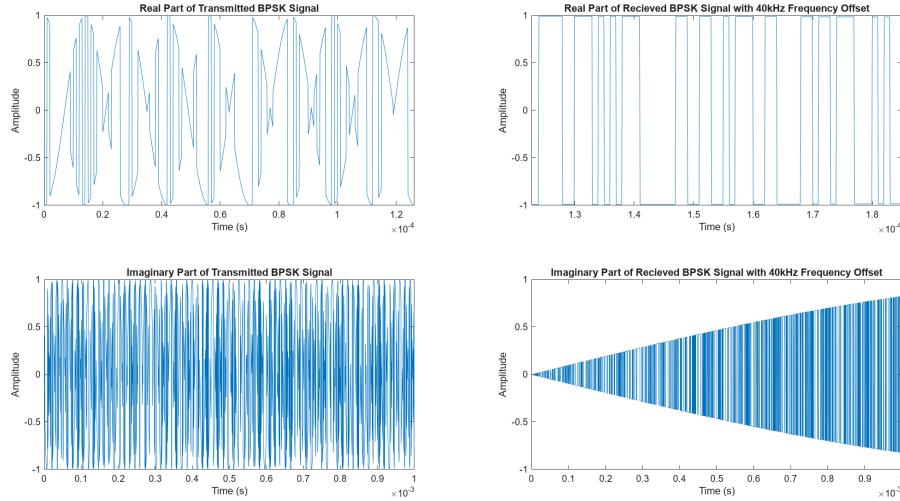


Figure 10: Received Signal After Coarse Frequency Correction

Fine Frequency Correction: The PLL algorithm provides complete frequency offset recovery. With 10 MHz sample rate, the maximum 37.5 KHz offset corresponds to a normalized frequency of 0.00375. This enables effective PLL operation without requiring an excessive pull-in range. The PLL's frequency recovery performance is demonstrated in Figure 11.

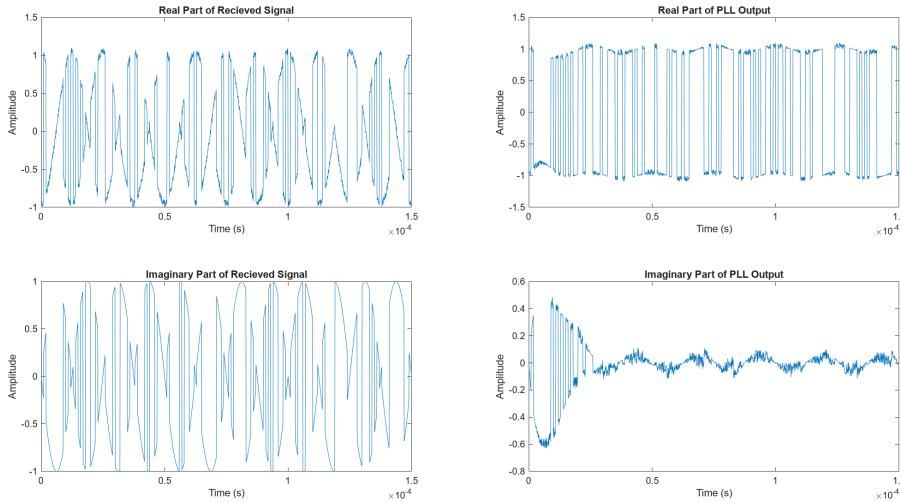


Figure 11: Zoomed Output of the PLL

Measuring Delays with Autocorrelation: The transmitted baseband signal consists of a 1000-element pseudo-random noise (PRN) code. As discussed in the CDMA codes section, the correlation properties of these PRN sequences enable precise delay measurement through cross-correlation with the reference code. For stationary cases, periodic transmission of the same PRN code results in consistent delay measurements across periods, as demonstrated in Figure 12.

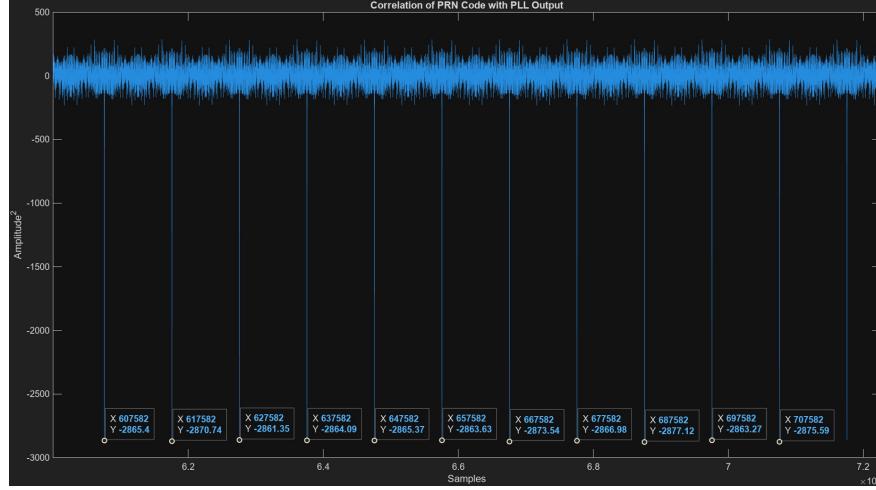


Figure 12: Correlation of the PRN Code with PLL Output

However, longer measurements reveal a linear drift in delay samples, even under stationary conditions (Figure 13). This drift varied in rate and direction between transmissions, but was independent of which SDR is utilized as the receiver or transmitter. Despite these variations, the drift followed a linear pattern. A linear fit derived from 1 second of the data accurately predicted the drift behavior throughout the reception period, as demonstrated in Figure 14.

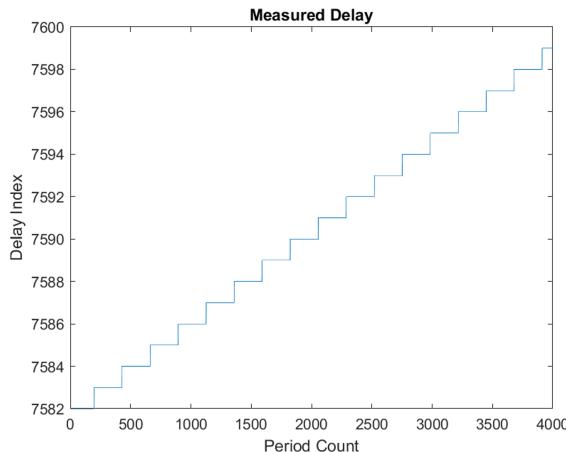


Figure 13: The Measured Delay and Its Drift Over 4 Seconds

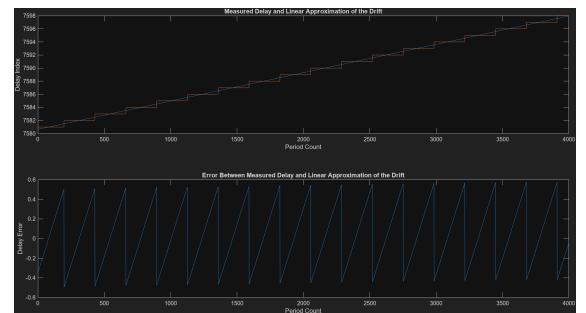


Figure 14: The measured delay and a linear fit of its drift over 4 seconds on top. Error of the fit on the bottom

5. Positioning Calculations

After CM2, the project was divided into two separate experimental setups. During this phase, the first setup was developed as an alternative solution in case the main configuration encountered technical difficulties. The second setup continued with the core objective of the project by performing delta-position estimation using two SDR devices, as originally required.

The primary objective of the initial setup was to eliminate clock mismatches between two SDRs, which can introduce significant challenges in signal processing. To address this, a single SDR device functioning as both transmitter and receiver was used, thereby completely eliminating the issue of clock discrepancy.

A 50 cm long SMA cable was utilized to connect the receiver antenna to the SDR, allowing for controlled physical movement of the antenna along a linear axis without affecting the internal clock of the device. The main goal of this setup was to estimate the displacement of the receiver antenna solely based on the phase difference of the received signal.

Since the system operated with a shared clock reference, the received signal remained phase-aligned with the transmitted one. This property enabled the application of a phase-based method to calculate the delta position of the receiver between two locations. By analyzing phase shifts between successive received signals, the change in position was estimated. Under the assumption of minimal reflection-induced interference and a stable environment, the observed phase changes were directly mapped to distance variations, expressed in terms of the signal wavelength.

In this experimental setup, one ADALM-Pluto SDRs operating at 1.5 GHz was used. A continuous complex sinusoidal waveform was transmitted, and the receiver acquired this signal in real-time. To detect movement, the received data was divided into two equal segments. The average phase of each segment was calculated, and the phase difference between two segments was calculated.

$$\Delta\phi = \phi_2 - \phi_1$$

Assuming a linear movement along the signal path and no clock mismatch (due to using matched SDRs), the delta-position Δd can be calculated using the relation:

$$\Delta d = \frac{\lambda}{2\pi} \cdot \Delta\phi$$

where λ is the wavelength of the RF signal, computed as $\lambda = \frac{c}{f_c}$ with c being the speed of light and f_c the carrier frequency. To enhance sensitivity and robustness, the phase difference was scaled and accumulated iteratively. Additionally, the angle of arrival θ was approximated using:

$$\theta = \arcsin \left(\frac{\Delta\phi \cdot \lambda}{2\pi d} \right)$$

where d is the known separation between two successive phase samples. The system continuously monitored θ to detect the onset and completion of motion. Once movement was detected, cumulative phase changes $\Delta\phi_{\text{total}}$ were tracked. The corresponding linear displacement Δd was calculated by $\Delta d = \frac{\lambda}{2\pi} \cdot \Delta\phi_{\text{total}}$ which is previously computed and converted to centimeters as:

$$\Delta d_{\text{cm}} = \left(\frac{\lambda}{2\pi} \cdot \Delta\phi_{\text{total}} \right) \cdot 100$$

This method allows for sub-wavelength resolution in displacement measurement, expressed in centimeters, without requiring external synchronization or absolute position reference. The algorithm for the first setup follows the approach. The received signal is split into two signals.

The results were obtained when the receiver antenna was moved by 40 cm can be seen from Figure 15, 16, and 17.

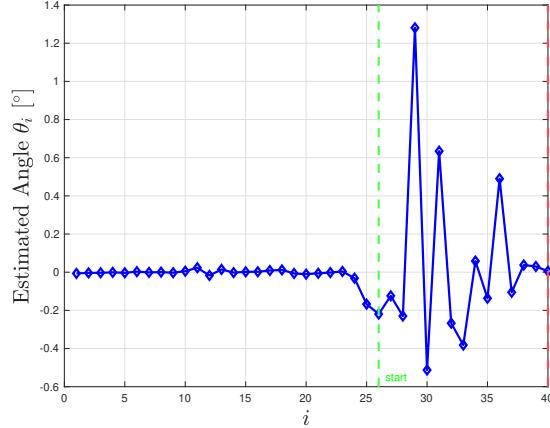


Figure 15: Estimated Angle with respect to Sample i with Start and End of the Motion indicated

Then, the average phase of each sub-signal is computed. Consequently, the phase difference between them is then used to estimate the direction and angle of movement Θ . Movement is detected if the calculated angle exceeds a certain threshold which is determined according to the trial-and-error to make it suitable for the environment. The phase differences are accumulated throughout the motion which can be seen in the Figure 16.

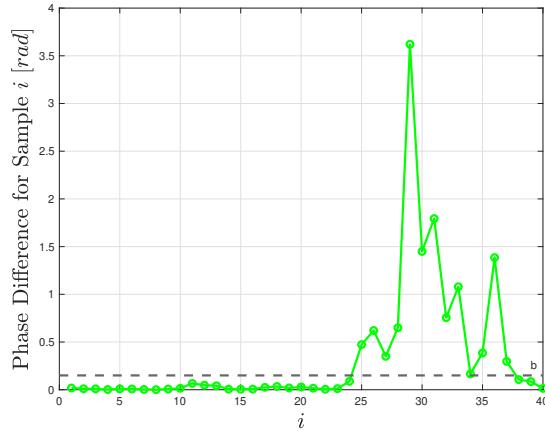


Figure 16: Computed Phase Difference in radian with respect to Sample i with Threshold $\epsilon = 0.15$ indicated

Once the angle falls below a lower threshold, indicating the end of movement, the total accumulated phase difference is used to calculate the displacement of the object based on the signal's wavelength, and the result is output in centimeters (see Figure 17). The overall algorithm for the first setup can be found in Appendix D.

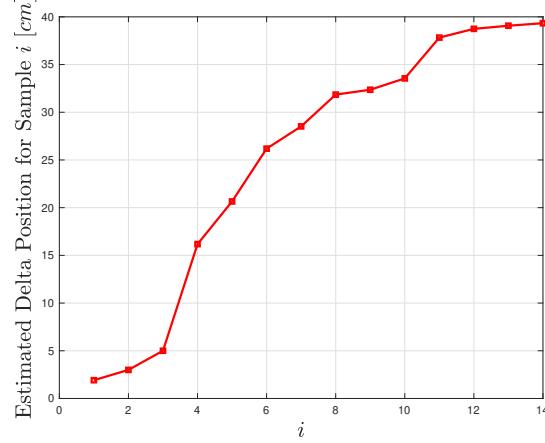
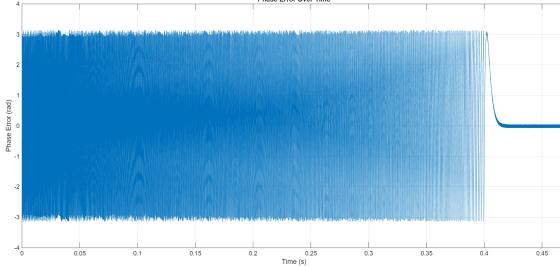


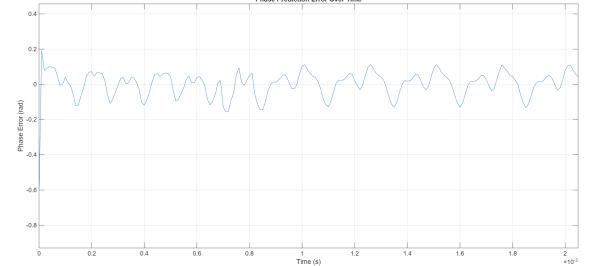
Figure 17: Estimated Delta Position in cm with respect to Sample i

The two SDR setup, consisting of a transmitting and a receiving SDR driven by different internal oscillators, estimated the relative disposition of the receiver by constantly tracking the phase of the incoming $1.5GHz$ signal from the transmitter with Kalman Filtering, detecting the start and end of the movement, and fitting a polynomial to the clock drift of the stationary device; thus differentiating the phase shift acquired by motion from the phase shift due to clock mismatches and LO drifts.

Kalman Filtering: Until CM2, phase tracking was done by using a second order Phase Locked Loop, consisting of a PI controller plus an integrator inside the loop. A Kalman Filter was decided as it had faster settling times and better steady-state variance overall.

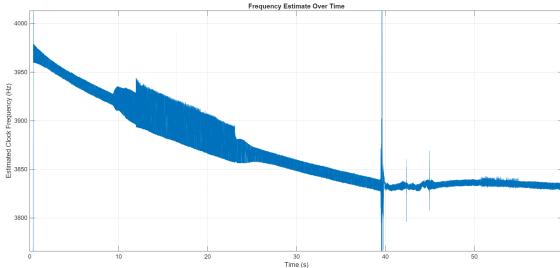


(a) Settling of phase error when PLL is used.

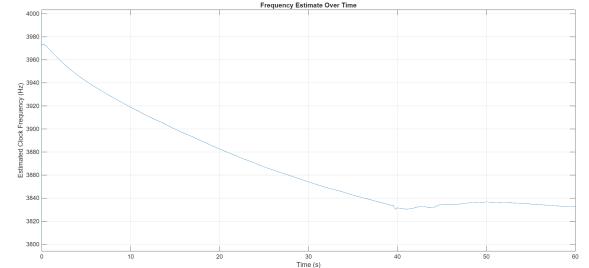


(b) Settling of phase error when KF is used.

Figure 18: Comparison of Settling Times of PLL vs Kalman Filter

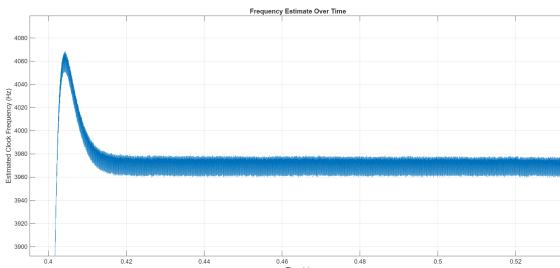


(a) Estimated frequency offset using PLL.

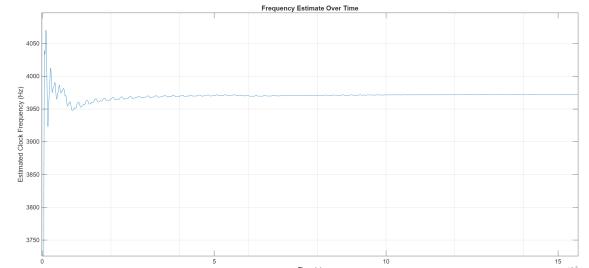


(b) Estimated frequency offset using KF.

Figure 19: Estimated Frequencies of Both Methods Compared



(a) Settling and variations after settling of estimated frequency offset when PLL is used.



(b) Settling and variations after settling of estimated frequency offset when KF is used.

Figure 20: Steady-State Variances of Frequency Estimations

From Figure 18 it can be seen that settling of PLL almost takes half a second while KF settles to steady-state in less than ten milliseconds. And in Figures 19 and 20 , the steady state of the estimated frequency offsets are shown. It can also be seen that KF has much less steady-state variance than PLL.

The Kalman Filter estimates three state variables from the measured the complex phase of the received IQ signal: the phase of the signal, frequency of the signal, and the frequency drift of the signal, which is the time derivative of the signal's frequency.

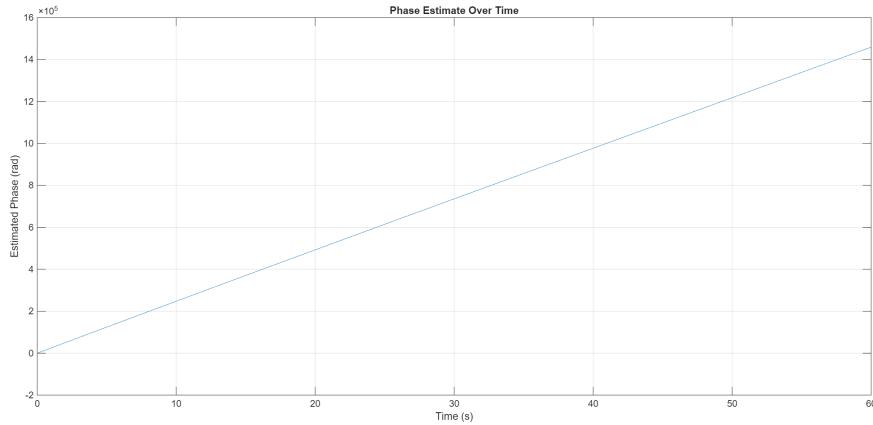


Figure 21: Phase Estimate from Kalman Filter.

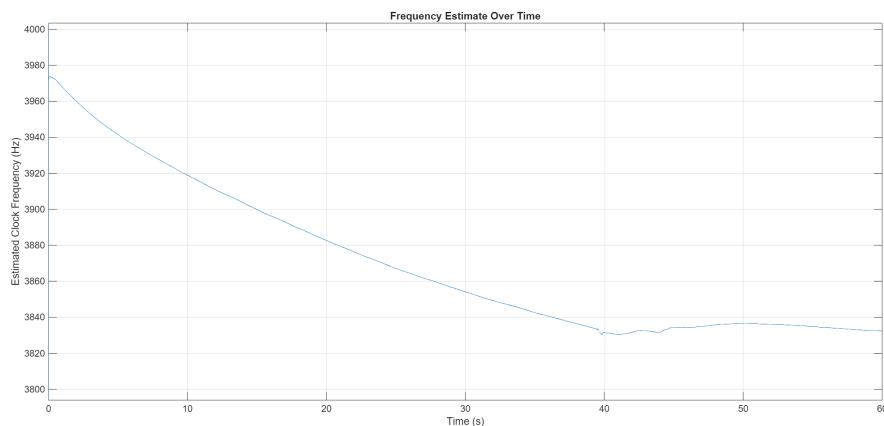


Figure 22: Frequency Estimate from Kalman Filter.

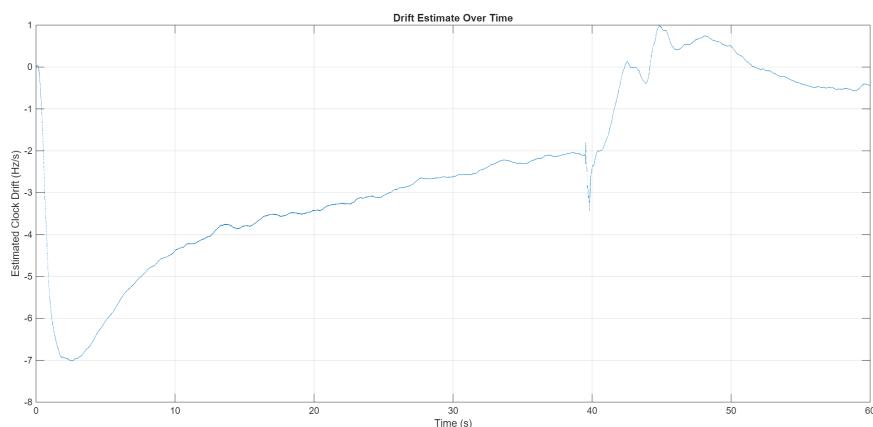


Figure 23: Drift Estimate from Kalman Filter.

While only the phase estimation was used for any disposition calculation, frequency and drift estimations were useful for observation and analysis purposes. For example, as it can be seen in Figure 23 after the 40 second mark where movement occurs, there is a sudden and drastic change in clock drift; thus the clock frequency starts to follow a different polynomial trend immediately after acceleration (22), probably due to shock sensitivities of SDR internal oscillators. Also during the first 10 seconds of operation higher order change occurs compared to the steady-state and static operation of the SDRs (between 15-40 seconds interval), likely the result of the internal settling of SDR oscillators and PLLs. These patterns were present in all realizations, and they have given us a general insights about the transient behaviour of SDR clocks.

Movement Detection: Every time the receiving SDR was displaced with sufficient acceleration, high level (almost π radians) spikes were present in the phase error, also with a higher error baseline (Figures 24 and 25). In order to accurately detect the spikes caused by the beginning and the ending of the movement Cell Averaging Constant False Alarm Rate (CA-CFAR or CFAR) algorithm was used.

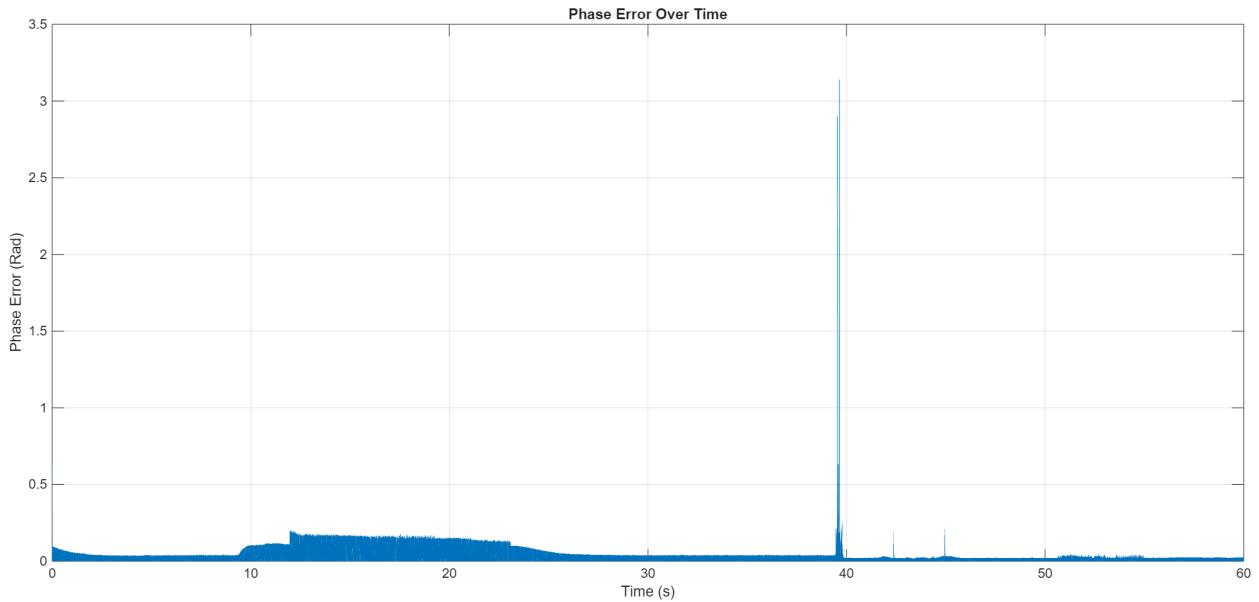


Figure 24: Phase Error of Kalman Filter

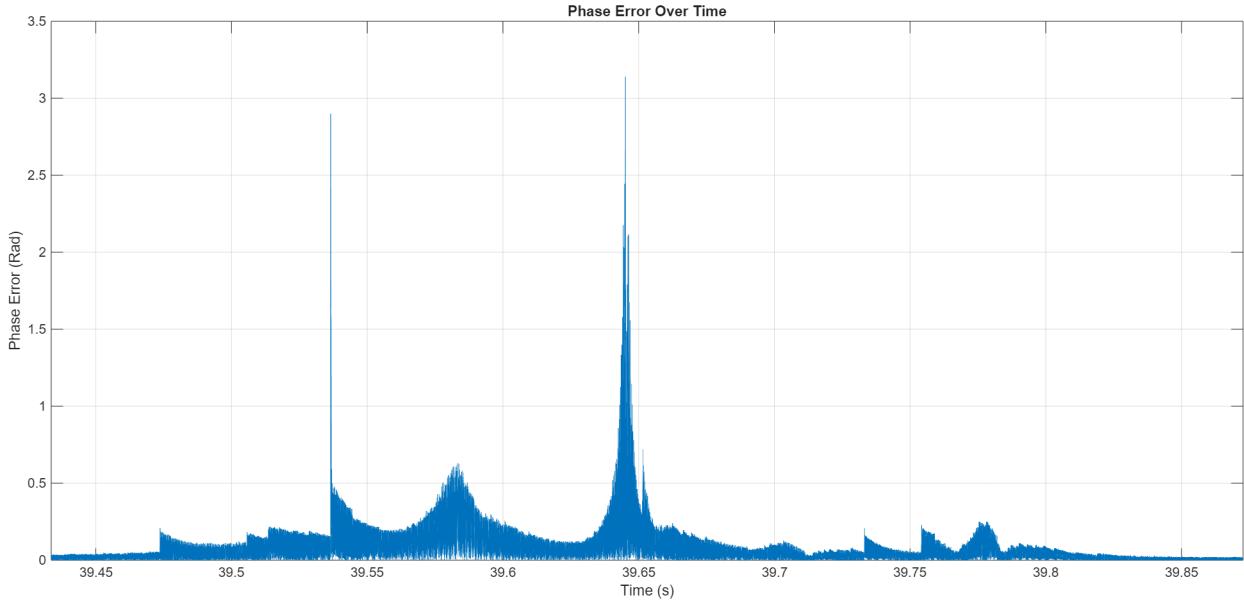


Figure 25: Phase Error during and around Movement

In CFAR algorithm a different level of threshold is constructed for each element in the array. This process allows for peaks to be distinguished in a signal where the background noise, in this case error, level can vary.

As it can be seen from Figures 26 and 27, the threshold varies with the baseline error and only the high level and narrow error spikes, resulted from the impulsive acceleration of the receiving SDR, can pass through the threshold.

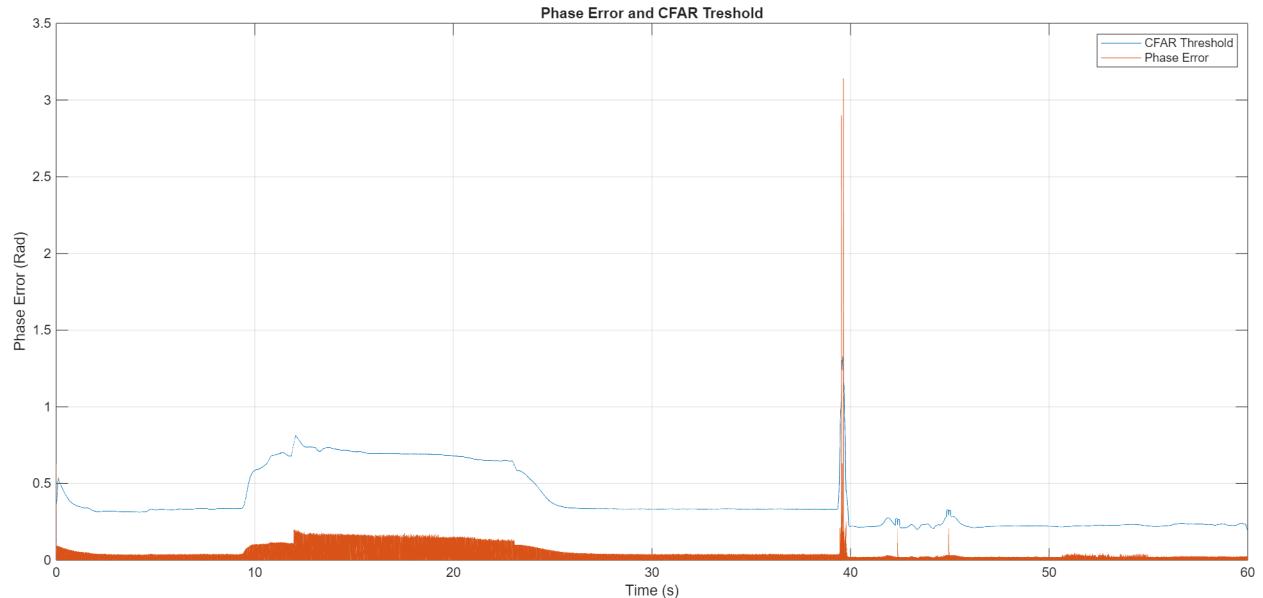


Figure 26: Phase Error and CFAR Threshold

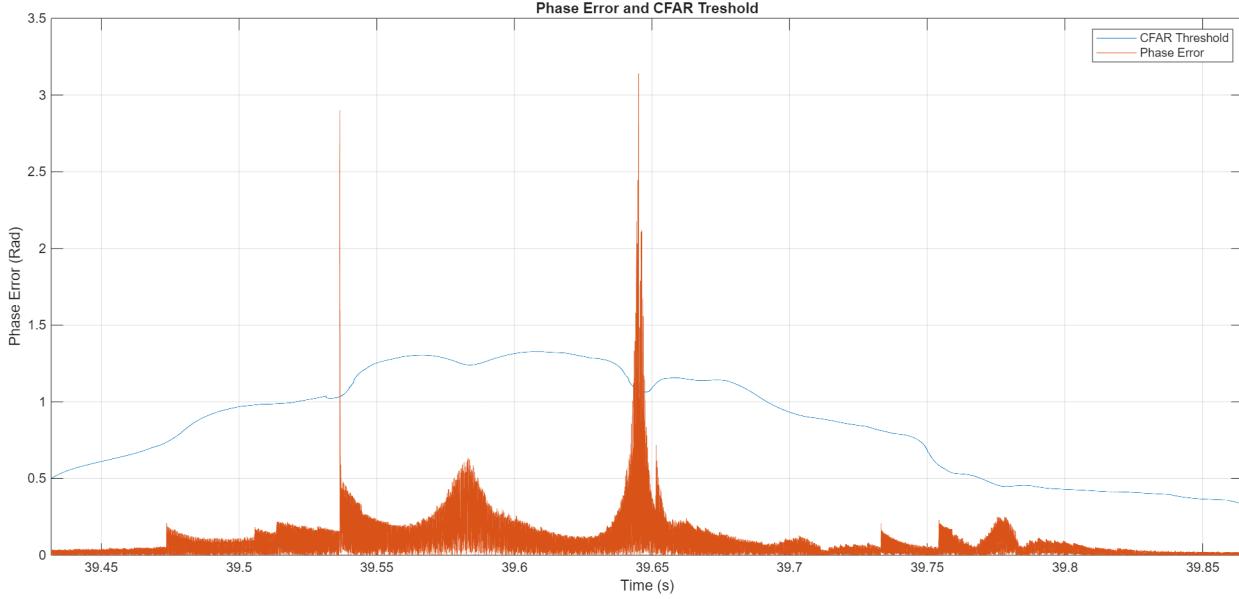


Figure 27: Phase Error and CFAR Threshold around Movement

After CM3, the CFAR algorithm is left as movement detection, after de-trending the phase plot, and can accurately track the displacement over time.

Clock Drift Removal: As mentioned under the section **Kalman Filtering** and shown in Figures 21, 22 and 23; the polynomial that the clock difference between the SDRs follow changes drastically, immediately after movement. Also, the first 10-20 seconds after booting the SDRs, their local oscillators are still settling. Therefore, the polynomial fitting to the phase had to start after the initial settling time had passed and before movement had started. The start point of the polynomial fit was also a parameter which had to be tuned. When random fluctuations occurred in close time positions to the movement, shorter fits captured the background trends better. However, longer fits had more accurate results when random fluctuations had less amplitude, or were further from the movement's timestamp.

After inspecting many realizations of the frequency and clock drift estimations (Figures 22 and 23), it was seen that the steady-state behaviour of the SDR clock frequencies were mostly second order with random fluctuations superposed. So, the polynomial fit made to the estimated phase had to be third order.

After the polynomial fit was made and the trend was removed from the phase measurement, the phase was scaled by $\frac{9.993}{\pi}$, convert angles of wavelength traveled in radians to distance traveled in centimeters.

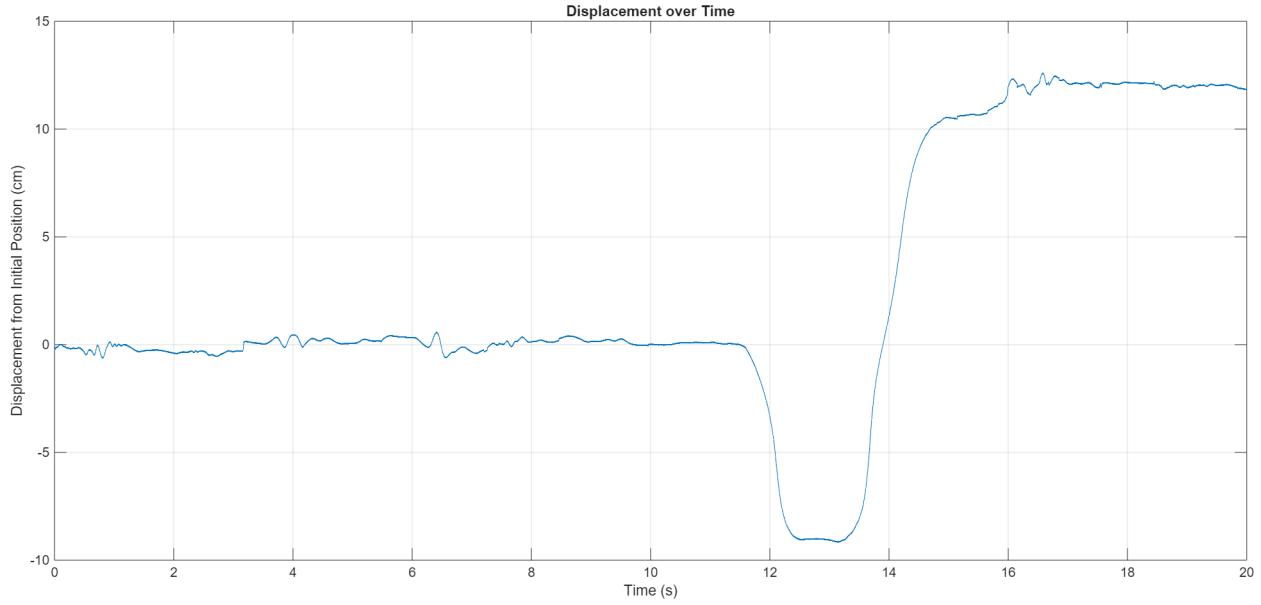


Figure 28: De-trended and Scaled Phase Plot where the Receiving SDR was Moved 36cms Away from the Transmitter.

As it can be seen from Figure 28, after the fit interval (25 - 40 seconds), clock's behavior starts to change immediately.

In Figures 29, and 30; results from different realizations are shown. In each test, the receiving SDR was moved 36cm away from the transmitting SDR. The start and end points detected by the CA-CFAR algorithm are marked on the plots.

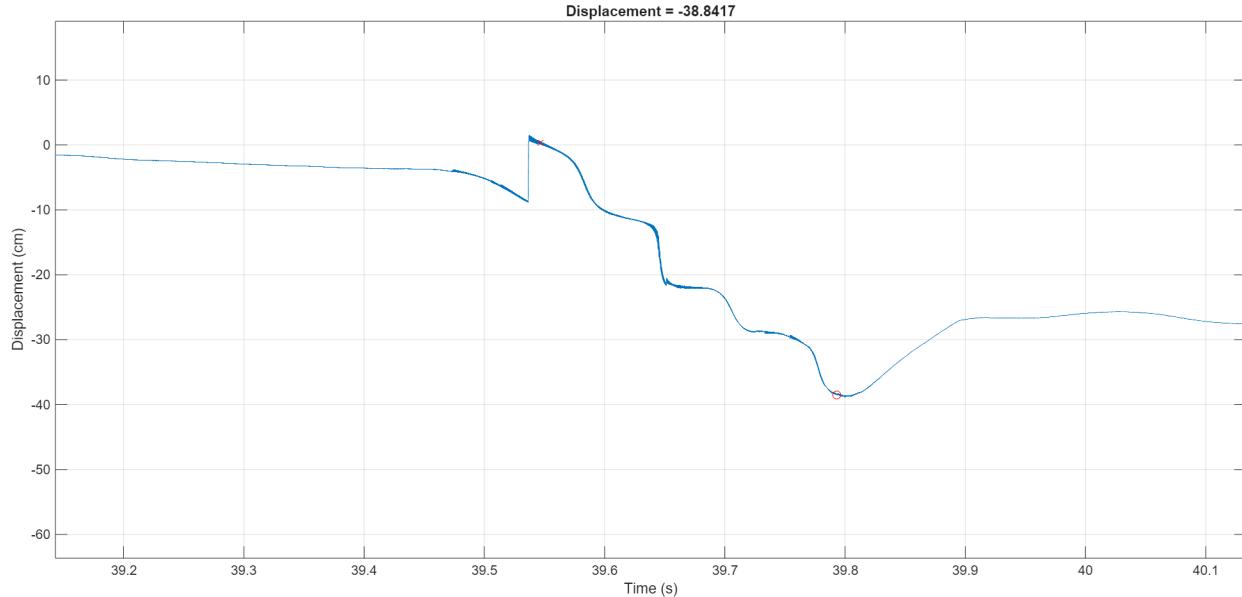


Figure 29: Close up of the Scaled and Polynomial Fitted Phase Plot

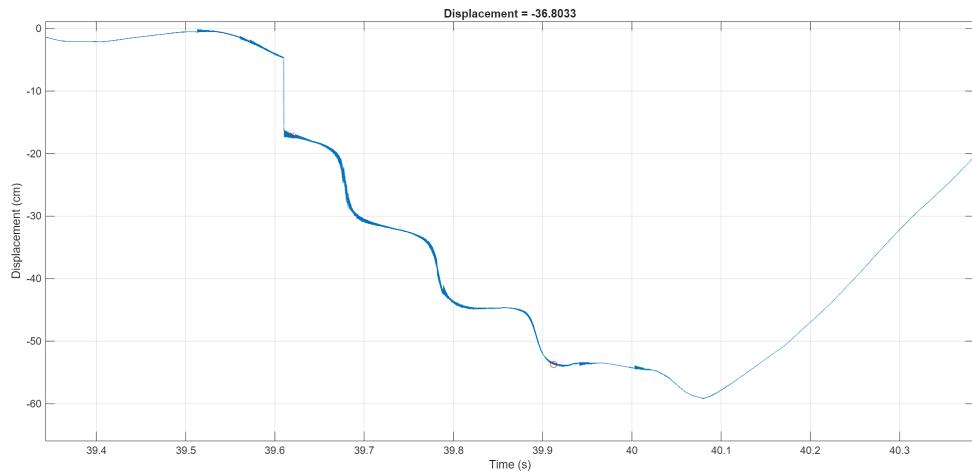


Figure 30: Close up of the Scaled and Polynomial Fitted Phase Plot

6.2.2 Progress After CM3

Common External Clock and Driver Circuit After CM3, it was seen that clock drifts between the receiver and the transmitter required a hardware solution. Therefore, two transmitters for Phase 1 and three transmitters for Phase 2 were used; where each transmitter was driven by the same external clock hardware.

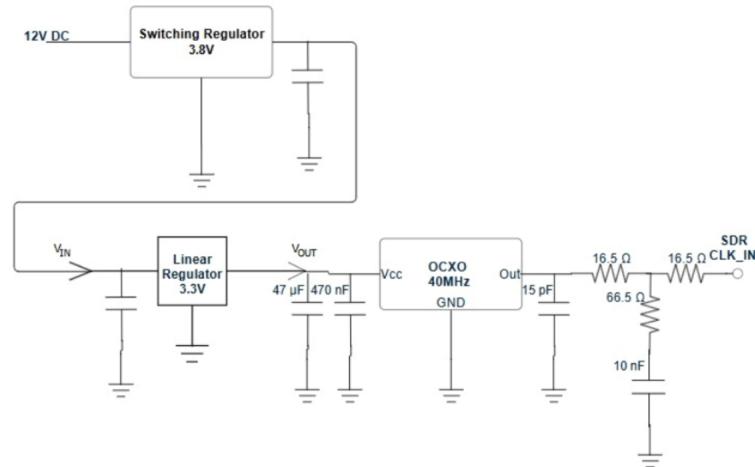


Figure 31: The Circuit Schematic of the 40 MHz External Clock with 3.3 Linear Regulator

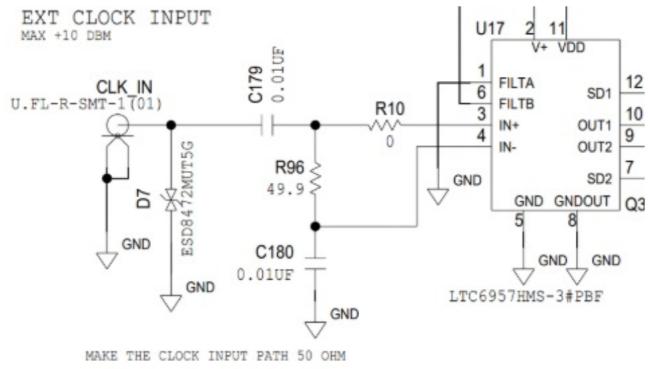


Figure 32: Adalm Pluto External Clock Input and Peripherals [31]



Figure 33: The Hardware Connection of the External Clock as a Clock Input to the SDR

The external clock was a 40 MHz oscillator (DOCSC012F-040.0M) with 3.3 V input voltage. At power on, the oscillator drew 0.6 A, and at steady state 0.25 A current. The clock supply was fed with 12 V DC, which was converted down to 4 V DC with a Buck Converter, and to 3.3 V DC with a Low-Dropout Linear Regulator (LDO). The LDO allowed for a more stable voltage source for the oscillator, since the switching ripple from the buck converter caused high frequency noise which affected the stability of the oscillator, and interfered with the transmitted waveform. The effect of supply noise can be seen in Figures 34 and 35.

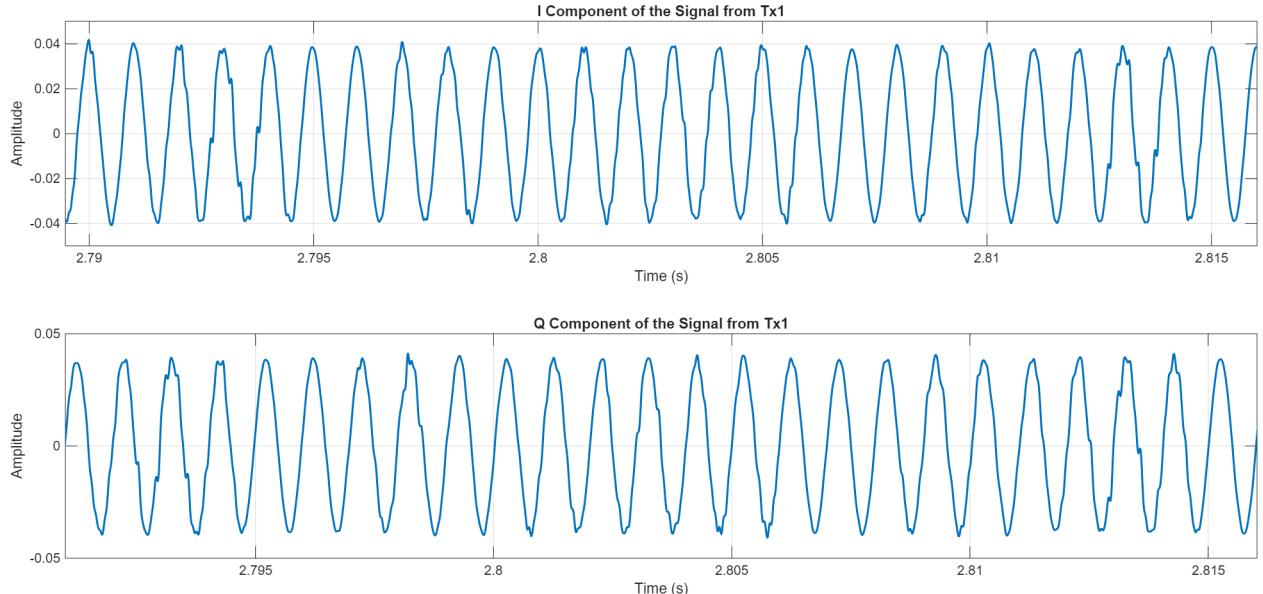


Figure 34: Effect of Supply Variations on the Signal

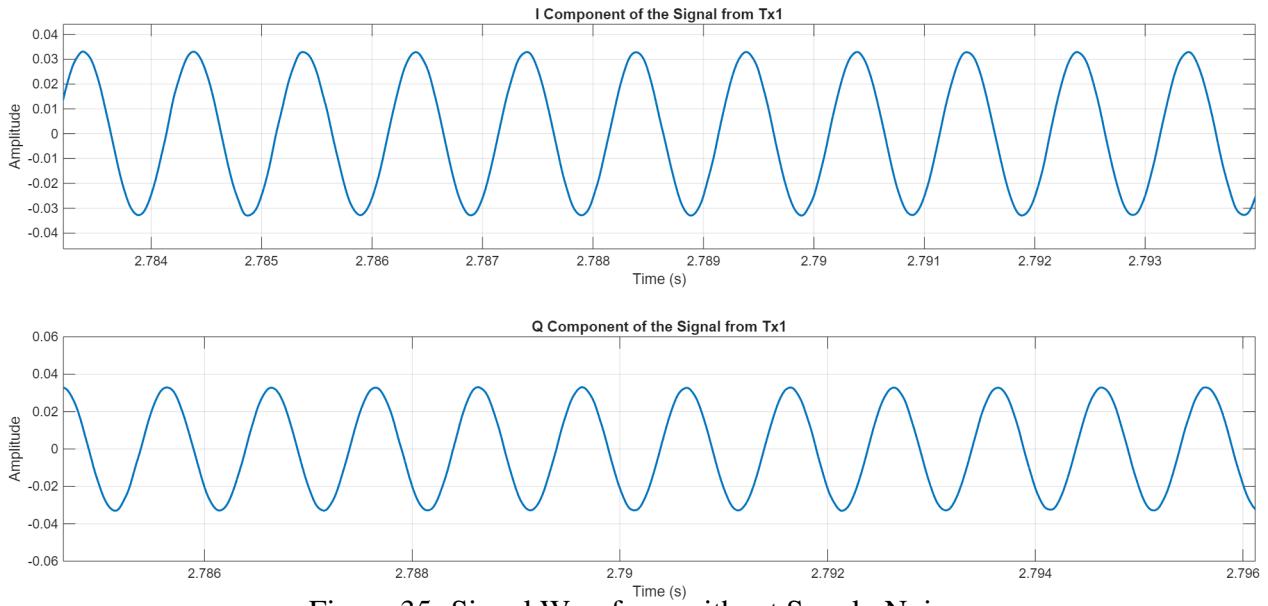


Figure 35: Signal Waveform without Supply Noise

FDMA and Signal Processing The two transmitters for Phase 1 and three transmitters in Phase 2 are differentiated using Frequency Division Multiple Access (FDMA). The transmitters were programmed to transmit at 0 Hz, 100 kHz, and 200 kHz with respect to baseband; and with a center frequency of 1500 MHz. On the receiver side, 1500 MHz center frequency was sampled with 2 MHz at the baseband. The resulting frequencies were then selected using bandpass filters with Butterworth filters with 100 kHz passband and demodulated down to 1 kHz intermediate complex frequency.

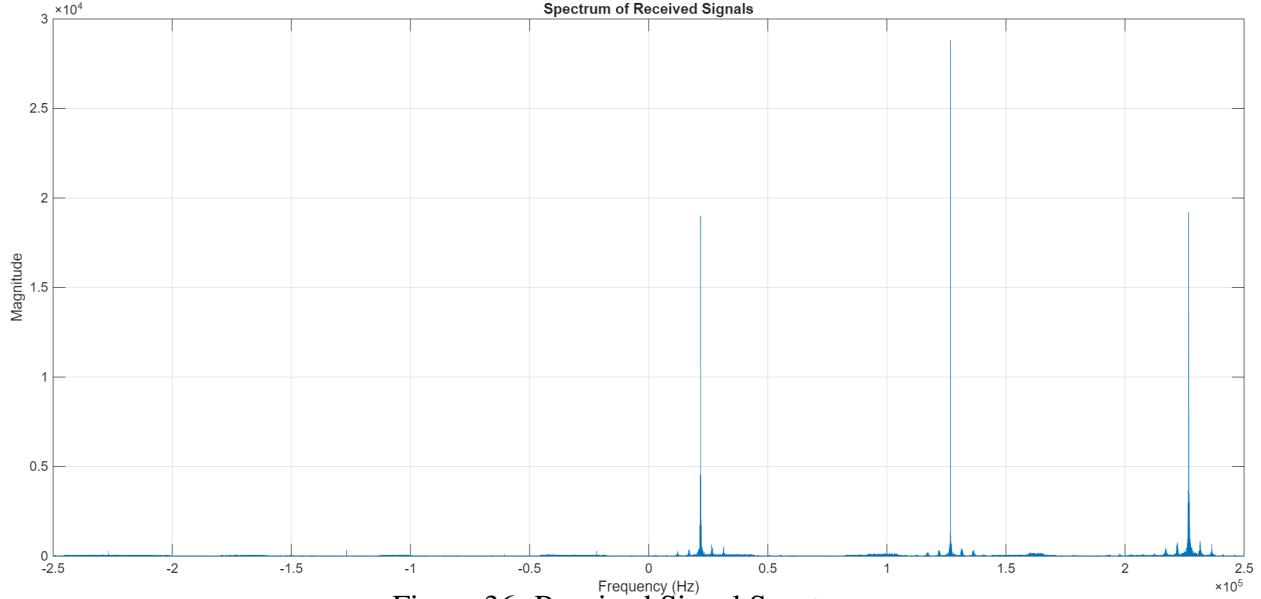


Figure 36: Received Signal Spectrum

The reasoning behind demodulating down to an intermediate frequency but not the baseband was the high variation in signal amplitude. The reason for this problem was most likely due to the SDR's Automatic Gain Control (AGC) circuitry. The variation in amplitude of a baseband or very-low frequency signal highly affected the steady-state noise of the Kalman Filter, whereas a low frequency signal allowed the filter to be more robust to amplitude variations. Two waveforms can be compared from 35 and 37.

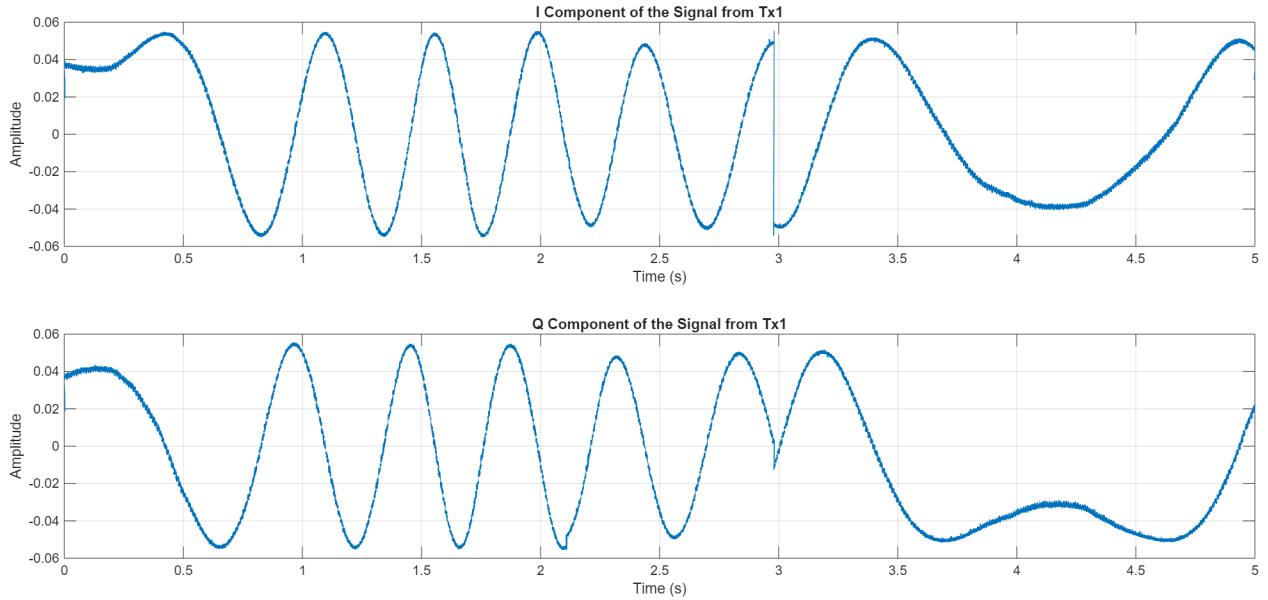


Figure 37: Noise and Sample Skips in a Very-Low Frequency Signal

Modified Kalman Filter and Doppler Frequency Tracking The main fundamental change of the Delta - Position Estimation algorithm from CM3 has been the addition of second transmitter with a common clock with the first transmitter for Phase 1. This allowed for distinguishing Doppler frequency shifts from the random walk of the clock offset by differential signaling. This can be explained as:

- Both transmitters, Tx1 and Tx2, transmit at a carrier frequency ω_c .
- The receiver, Rx, observes the incoming signals with a frequency offset due to clock differences. It receives the signals at the frequency $\omega_c + \delta_{\text{clk}}$, where δ_{clk} denotes the random clock difference between the receiver and the transmitters.
- When Rx moves toward Tx1 and away from Tx2, Doppler shifts are introduced:
 - Signal from Tx1 is observed at $\omega_c + \delta_{\text{clk}} + \omega_D$
 - Signal from Tx2 is observed at $\omega_c + \delta_{\text{clk}} - \omega_D$
- By subtracting the two received frequencies, the common terms (carrier frequency and clock offset) are eliminated:

$$(\omega_c + \delta_{\text{clk}} + \omega_D) - (\omega_c + \delta_{\text{clk}} - \omega_D) = 2\omega_D$$

Hence, the Doppler frequency ω_D can be isolated.

Since the difference between center frequencies of Tx1 and Tx2 are 100kHz, their difference in Doppler effect is only about $1.3 \cdot 10^{-4}$ Hz. This value is way below the steady state variance of frequency estimation, so it can be disregarded. However, despite using common external oscillators, a constant frequency offset still occurs between the center frequencies of the transmitters. Probably caused by the interface between the SDR and the PC, constant frequency offset causes an additive linear trend in the displacement. As it is visible in Figure 42, this offset required the receiver to be stationary for a certain duration, long enough to correctly estimate this constant offset.

Another fundamental change was the updated Kalman Filter. Now that the clock drift issue was solved, the main source of error had become the effect of skipped samples. Either due to the Analog to Digital Converter, the Linux Serial Peripheral Interface of the SDR, or the interface between the SDR and the PC; between many frames there had been a significant loss of samples, resulting in waveforms which can be seen in Figure 38. When the received complex signal phase is tracked, this had the effect of a step gain in displacement, seen in Figure 39.

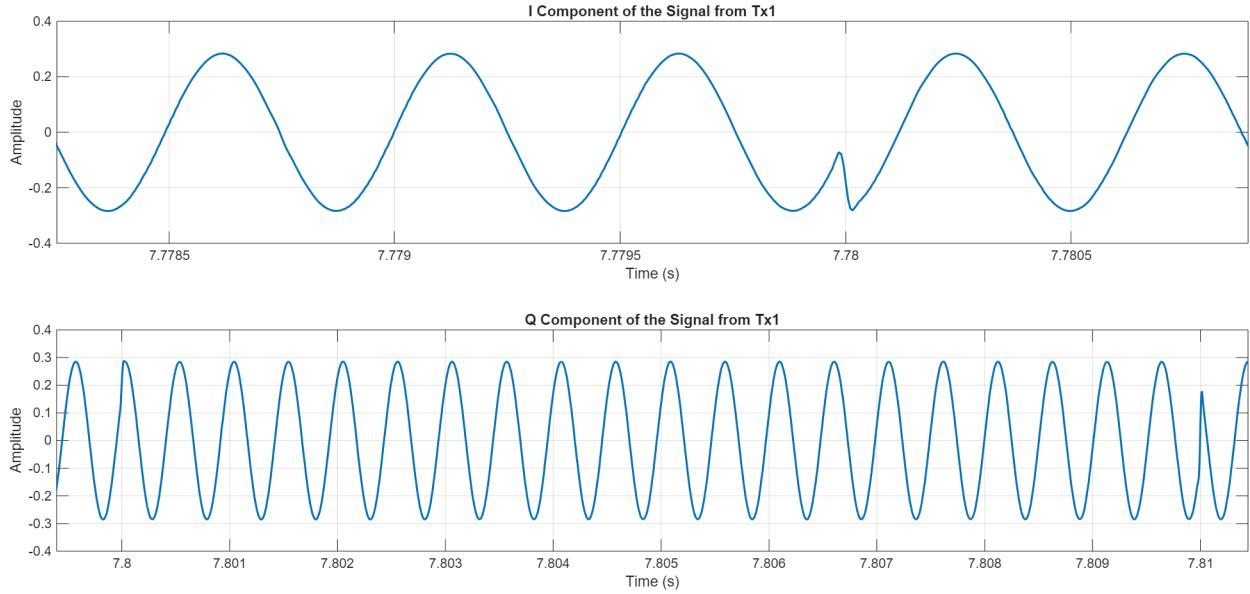


Figure 38: Signal with Three Phase Jumps Occurring within 30 ms

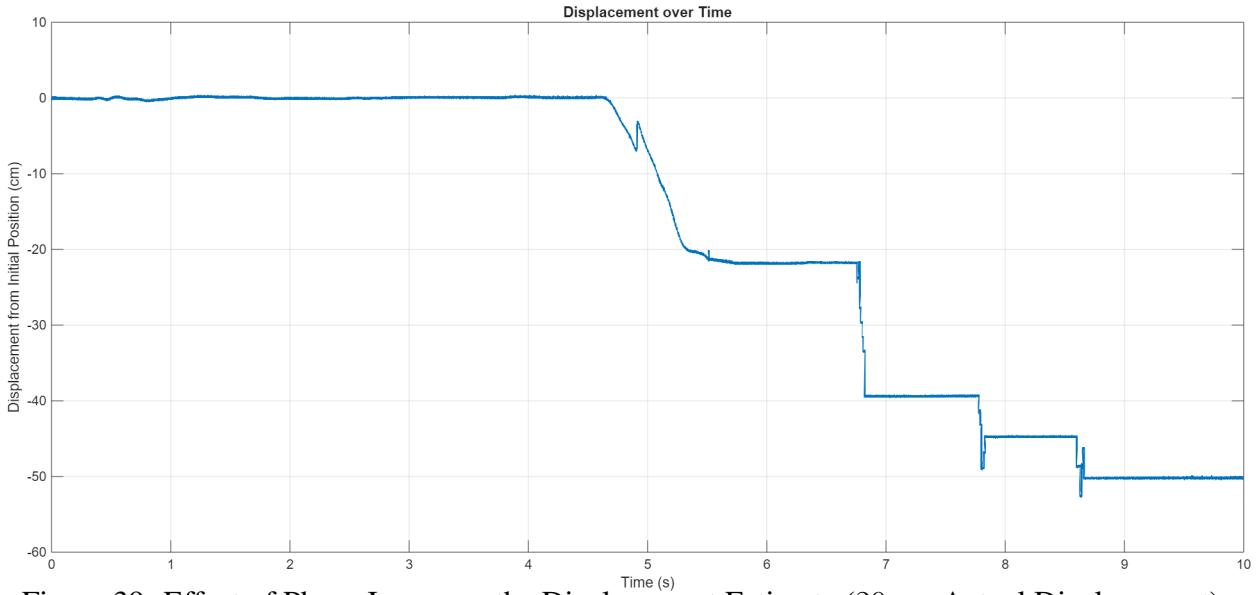


Figure 39: Effect of Phase Jumps on the Displacement Estimate (20 cm Actual Displacement) .

The effect of these phase jumps was tried to be mitigated by embedding a decision threshold to the Kalman Filter. Since the estimation error would become much higher when a phase jump had occurred, with respect to the steady state variance or the error caused by movement; these jumps could be detected by using an adaptive threshold. The thresholds were determined by averaging a certain amount of the previous error values, and scaling this average with a constant gain. The estimation error and the error threshold can be seen in Figure 40. When these phase jumps are detected, the updated Kalman Filter only updates its phase estimate but freezes its frequency estimate, preventing the frequency estimation to be affected as much as the phase estimation. The

differences of these effects can be seen in Figures 41 and 42. In Figure 42, there is an actual movement which corresponds to a Doppler frequency around 2-3 Hz, and small transients caused by phase jumps. However, In Figure 41, phase jumps caused 6 Hz frequency spikes, larger than the actual Doppler caused by the movement. By using the updated Kalman Filter and integrating the Doppler frequency for displacement calculation instead of purely tracking the phase, the effect of these jumps can be mitigated. This effect can be seen by comparing Figures 39 and 43. Figure 43, is the same signal as in Figure 39; the only difference is that in Figure 39 the standard Kalman Filter is used whereas in Figure 43 the updated version is used.

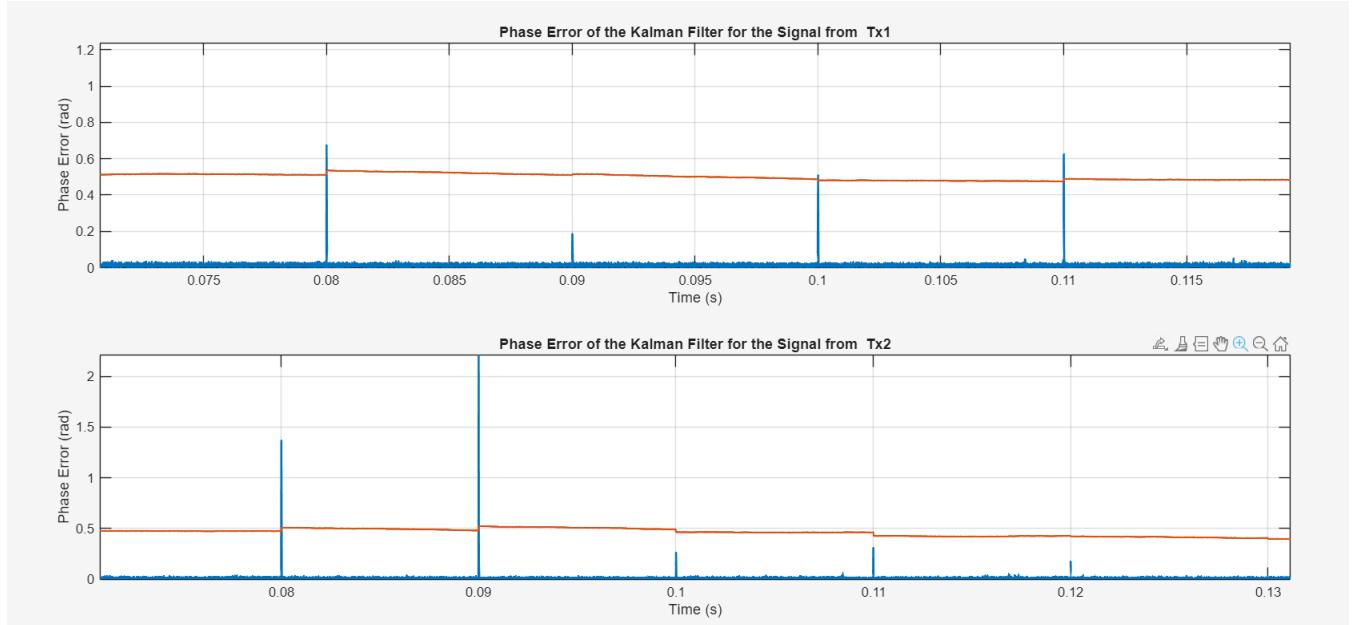


Figure 40: Estimation Error of the Kalman Filter and Errors Caused by Phase Jumps with Respect to the Error Threshold.

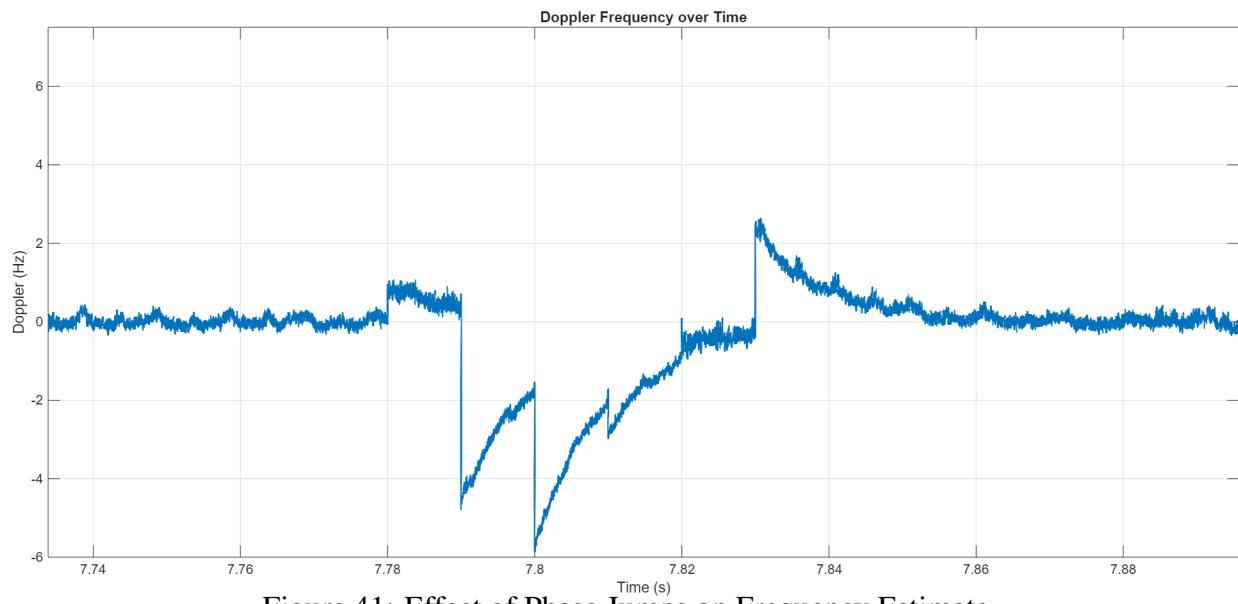


Figure 41: Effect of Phase Jumps on Frequency Estimate

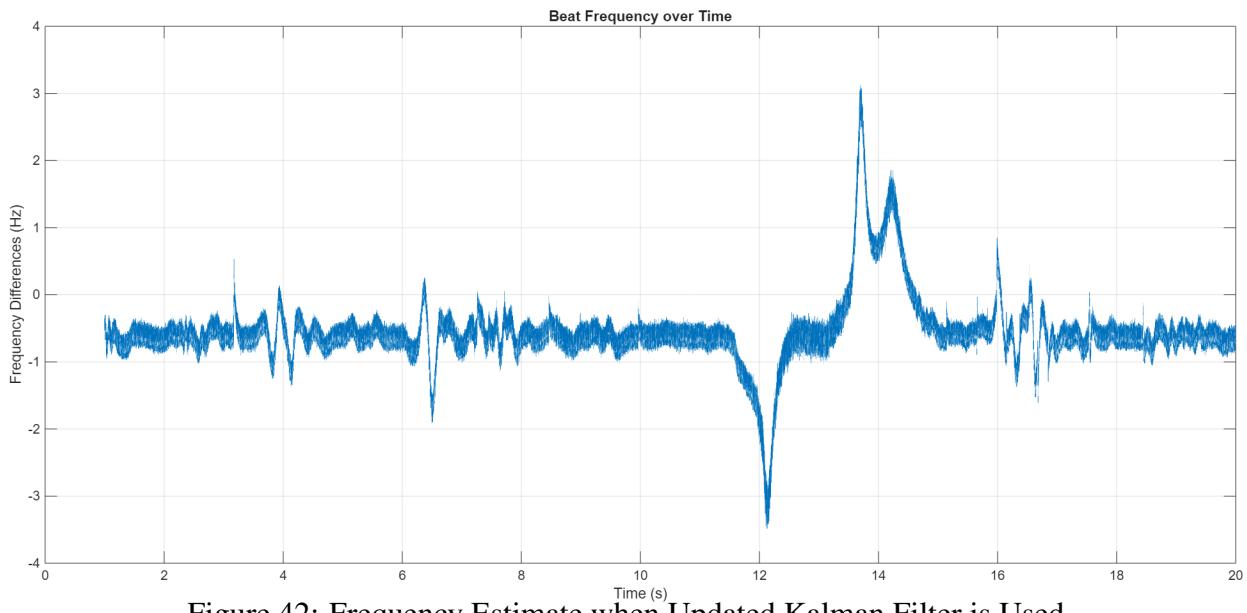


Figure 42: Frequency Estimate when Updated Kalman Filter is Used

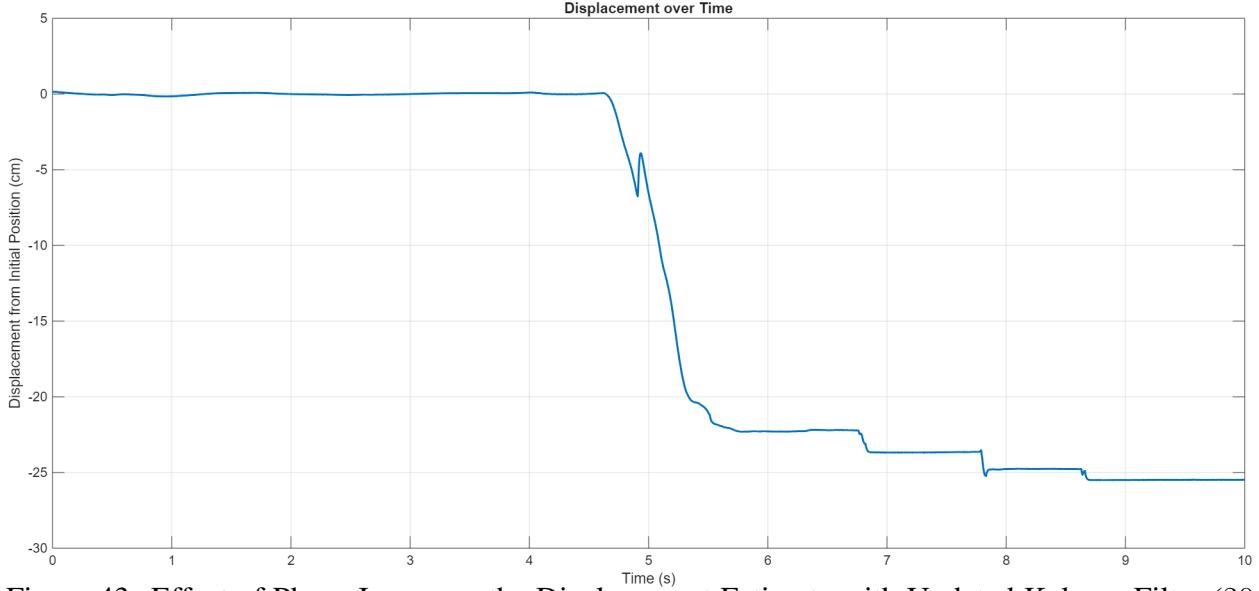


Figure 43: Effect of Phase Jumps on the Displacement Estimate with Updated Kalman Filter (20 cm Actual Displacement)

2D Distance Calculation The 1D displacement derivation in Phase 1 is pretty straightforward. Considering the transmitters and the receiver are colinear, allowing the displacement to be computed directly by integrating the Doppler frequency, which is the difference of frequencies from the two transmitters. In Phase 2, this is no longer the case. Since the unit vectors of the radial velocities are changing with respect to the position of the receiver in the 2D plane, the pairwise differences of transmitted frequencies do not give the velocity in a constant direction. Instead, the integrals of the pairwise frequency differences result in the pairwise differences of radial changes. To clearly explain, let the initial position of the receiver be \mathbf{p}_0 , the position of the i^{th} transmitter $\mathbf{p}_{\text{Tx}i}$, and the final position of the receiver \mathbf{p}_1 . So, the initial distance of the receiver to the i^{th} transmitter is,

$$r_{0i} = \|\mathbf{p}_{\text{Tx}i} - \mathbf{p}_0\|_2$$

And the final distance of the receiver to the i^{th} transmitter is,

$$r_{1i} = \|\mathbf{p}_{\text{Tx}i} - \mathbf{p}_1\|_2$$

Then, the radial change in distance with respect to the i^{th} transmitter becomes,

$$d_i = r_{1i} - r_{0i} = \|\mathbf{p}_{\text{Tx}i} - \mathbf{p}_1\|_2 - \|\mathbf{p}_{\text{Tx}i} - \mathbf{p}_0\|_2$$

Finally, the integral of the pairwise frequency difference of the i^{th} and the j^{th} transmitter can be

expressed as,

$$d_{ij} = d_i - d_j = \|\mathbf{p}_{\text{Tx}_i} - \mathbf{p}_1\|_2 - \|\mathbf{p}_{\text{Tx}_i} - \mathbf{p}_0\|_2 - (\|\mathbf{p}_{\text{Tx}_j} - \mathbf{p}_1\|_2 - \|\mathbf{p}_{\text{Tx}_j} - \mathbf{p}_0\|_2)$$

$$= c_{\text{Doppler}} \int_0^t \omega_{\text{Tx}_i} - \omega_{\text{Tx}_j} dt = \|\mathbf{p}_{\text{Tx}_i} - \mathbf{p}_1\|_2 - \|\mathbf{p}_{\text{Tx}_i} - \mathbf{p}_0\|_2 - (\|\mathbf{p}_{\text{Tx}_j} - \mathbf{p}_1\|_2 - \|\mathbf{p}_{\text{Tx}_j} - \mathbf{p}_0\|_2) \quad (1)$$

which gives three equations as constraints on the final position \mathbf{p}_1 , where the initial position of the receiver and the positions of the transmitters are known, and c_{Doppler} is the relevant scalar transforming radians to centimeters.

Thus, the estimated final position, $\hat{\mathbf{p}}_1 = [\hat{x} \ \hat{y}]^T$ can be computed as,

$$\hat{\mathbf{p}}_1 = \arg \min_{\mathbf{p}_1} \sum_{i,j} d_{ij} - (\|\mathbf{p}_{\text{Tx}_i} - \mathbf{p}_1\|_2 - \|\mathbf{p}_{\text{Tx}_i} - \mathbf{p}_0\|_2 - (\|\mathbf{p}_{\text{Tx}_j} - \mathbf{p}_1\|_2 - \|\mathbf{p}_{\text{Tx}_j} - \mathbf{p}_0\|_2))$$

The above minimization is necessary since the equation in Eq.1 does not have an analytical solution, so it can only be solved numerically by minimizing the residuals between the integrated Doppler pairs (left hand side of the equation, direct derivations from measurements) and the radial distance changes (right hand side, estimation using \hat{x} and \hat{y}).

User Interface For the user interface, the App Designer in MATLAB is used. Here, two tabs for Phase 1 and Phase 2 are created. The signal reception is controlled by the “START” and “STOP” buttons. When the signal reception starts, the “Message” area displays “Reception started” and after a specified time, which can be changed by the user, the “Move” message appears on the screen. After reception ends “Reception ended” message is displayed. Later on, after the post processing steps, in the plot area the displacement is plotted. In Phase 1, this plot shows the continuous tracking of the displacement and in Phase 2, the plot shows how the location is changed with respect to the initial location. For both phases, the distance traveled part will show how much displacement occurred with respect to the initial position.

During the development phase, Kalman filter parameters were integrated in this design for easy usage. This enabled us to modify the Kalman filter parameters without running the script each time. For the final setup, the Kalman filter parameters will not be visible to the user. An example user interface with a Phase 1 result can be seen in Figure 44.

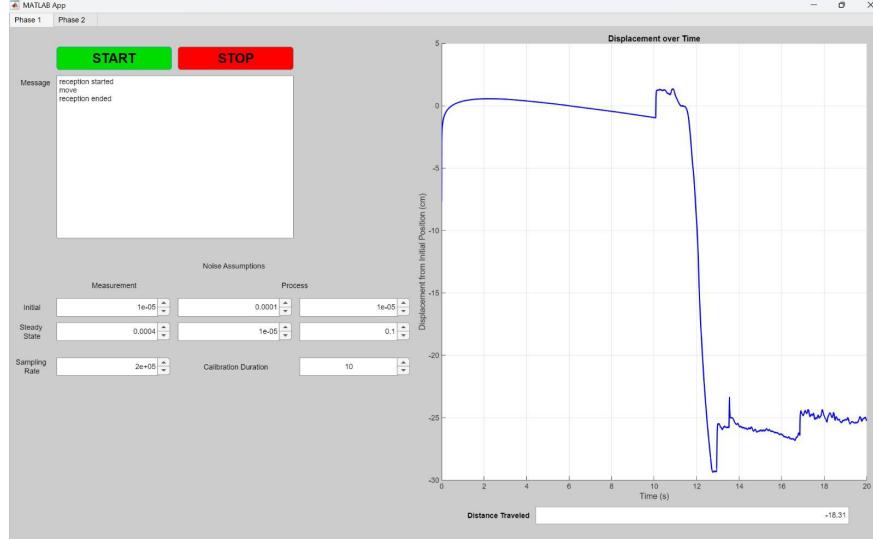


Figure 44: User Interface Showing Phase 1 Result

In the fair, the UI will not include any parameters which can be modified for Phase 1. In Phase 2, the UI will only include the initial position input from the user as a parameter to be modified.

6.2.3 Risks Encountered

We encountered several risks throughout the project across different work packages that significantly influenced both the timeline and the outcomes of our work.

- **WBS 4.2 – Timing Synchronization and Clock Error Management:** Another risk was the instability and drift of the internal clocks in ADALM-Pluto SDRs, which made phase synchronization extremely challenging. Firstly, we attempted to compensate algorithmically, but as the clocks behaved randomly, the aimed accuracy could not be obtained. To solve this problem, we integrated an external clock to synchronize the transmitter SDRs, which required extensive circuit design and power supply filtering. This approach, ultimately resolved the clock drift issue and provided better performance compared to the internal clocks of the SDR.
- **WBS 4.4 – Synchronization:** One of the risks was committing to PLL for phase tracking. However, we observed its limitations in both steady-state accuracy and convergence speed as experiments progressed. We adopted Kalman Filtering for phase tracking. With this choice we were able to track the clock drift as well as the phase. This decision significantly improved overall performance.
- **WBS 5.1, 5.3 – Positioning under Motion:** Clock drift and skipped samples create the basis of the problems we encountered throughout the project. Clock drift problem is solved using

external clocks as mentioned in the timing synchronization and clock error management section. To solve the skipped samples problem which were responsible from sudden jumps in the phase estimate, we used a moving average algorithm in our Kalman Filter design. By doing this, we alleviated the effects of skipped samples.

7 Results, Discussions, and Future Directions

7.1 Results

Both phases of the project are completed successfully. For Phase 1 the functional specifications for the final product can be listed as:

- SRS-01-11 Frequency and Bandwidth: Signal transmission and reception at 1.5 GHz center frequency and minimum 2 MHz bandwidth.
- SRS-01-12 FDMA: Signal differentiation using frequency division multiple access.
- SRS-01-13 External Clock Usage: Transmitters will be driven with a 40 MHz 10 ppb accuracy external clocks.
- SRS-01-21 Modulation: Utilization of complex modulation at 1.5 GHz.
- SRS-01-22 Filtering: Low pass filtering to recover modulated signals.
- SRS-01-23 Phase Locking: Rapid phase locking within 2 seconds.
- SRS-01-24 Doppler Frequency Tracking: Use of Kalman Filtering to alleviate effects of skipped samples.
- SRS-01-25 Movement Detection: Differentiation of movement from the phase discrepancy of the SDR.
- SRS-01-31 1D-Distance Calculation: Calculation of 1D distance between two points with ± 5 cm accuracy.

For this phase, all of the listed requirements are satisfied in the final design. The system follows the flow graph in Figure 45 and the components in this flow graph are specified according to the requirements listed as they were explained in detail in the methods section.

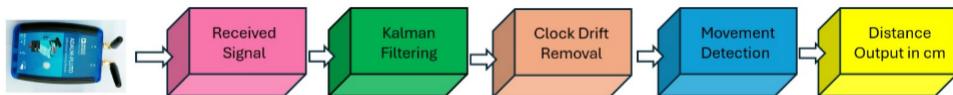


Figure 45: Flow Chart of the Final Design

Comprehensive testing was done for Phase 1 to ensure that the distance measurement in 1D works within the specified error interval. The table including test results can be seen in Appendix

F. The overall mean error of Phase 1 for the collected 100 samples was 12.11%. The histogram of error percentages can be seen in Figure 46.

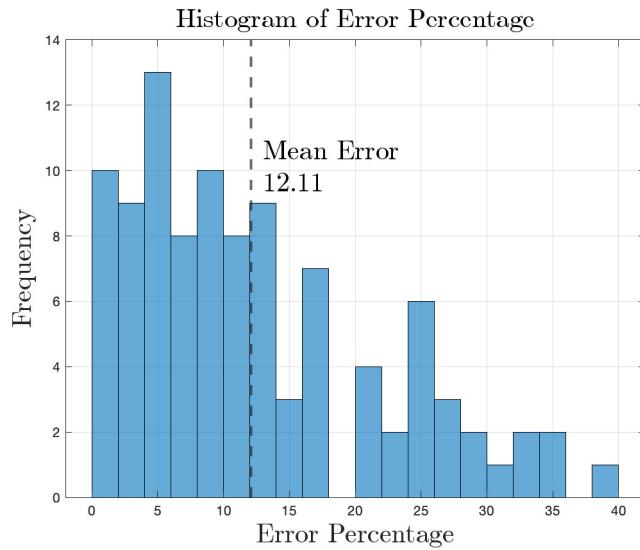


Figure 46: Histogram of Error Percentages

As it can be observed from the plot, the error is accumulated around 5-10 % interval. With 95% confidence the error is 12.11 ± 1.90 . For the collected samples, the plot of ground truth versus the experimental results can be seen in Figure 47. This plot clearly shows that the measured values does not vary much from the true expected values of the displacement.

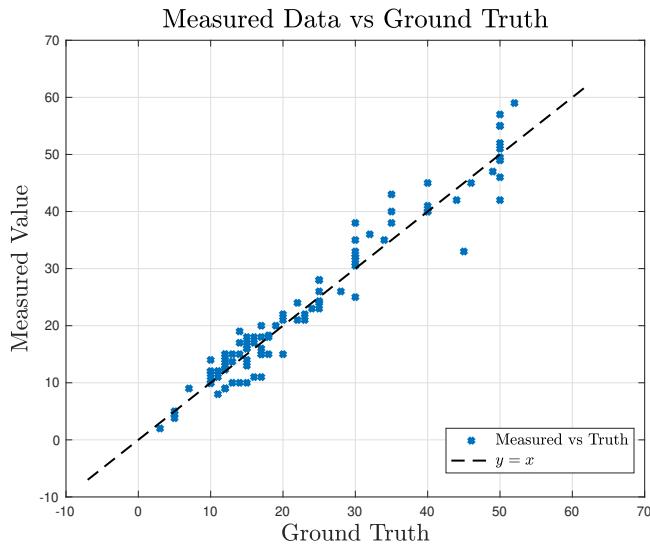


Figure 47: Ground Truth vs Experimental Results

The mean error for different displacement intervals can be seen in the table below. It can be seen

that, the error percentage for smaller displacement values increase. This occurs as a consequence of the frequency jumps created by the sample losses. Since these jumps created by this sample losses result in more or less the same displacement values which are close to 5 cm. Thus, the system is more suitable for measuring displacements larger than 20 cm as the displacements caused by this jumps will be relatively small compared to the actual displacement.

Error Summary of Phase 1	
Condition	Error (%)
Average Error	12.11
Displacement > 30 cm	8.2
Displacement > 20 cm	8.3
Displacement > 10 cm	11.5

Table 1: Phase 1 Error Summary

The user-related specifications related to Phase 1 can be listed as:

- SRS-03-10 User-Control: The ability of users to control signal transmission and reception via a user interface.
- SRS-03-11 Monitoring: The distance traveled must be displayed on the interface and the LCD screen.
- SRS-03-12 Portability: The system shall be implemented in a compact way to ensure easy deployment in various environments.

For both phases of the project, the listed user-related requirements are satisfied. The user interface designed for the project can start and stop the signal reception. The output can be viewed at the plot section of the interface and distance output in cm can be seen seen at the related section. Moreover, the signal transmission is controlled by a Raspberry-pi module. This increases the portability and easy deployment of the system.

For Phase 2, in addition to the functional requirements listed above, an additional requirement is satisfied which can be listed as:

- SRS-01-32 2D-Distance Calculation: Calculation of 2D vectorial distance between two points with ± 5 cm accuracy.

A comprehensive test is not yet conducted for this phase, but the initial trials showed promising results. Based on the initial position inputted by the user, the algorithm can determine to which position the receiver SDR has moved to.

7.2 Discussions and Lessons Learned

The overall goals of the project were achieved based on ASELSAN's expectations. Phase 1 achieved its objective of providing one-dimensional delta-position estimation using SDRs with an accuracy within the required ± 5 cm. This milestone validated our signal processing pipeline, synchronization methods, and displacement calculation algorithms. Phase 2, successfully extended the system to two-dimensional positioning. Additionally, full real-time integration which is as a further demand from the company although not originally within the project scope has been taken into consideration will be implemented if time permits. The project delivered a portable, SDR-based, and GNSS-independent positioning system aligned with ASELSAN's requirements.

Several aspects of the project went well. Especially, the implementation of Kalman filtering significantly improved both phase and frequency estimation, outperforming the previously used PLL in terms of accuracy and convergence speed. Furthermore, the integration of a common external clock into the transmitter chain was another important success of the project which resolved drift-related issues between SDRs and enabled coherent signal reception. This was a challenging task for us, as it required extensive hardware configuration as well as custom circuit design. Additionally, by fine tuning Kalman filter parameters and adding moving average algorithms to the filter added robustness to the system against sample losses. Finally, the Raspberry Pi integration and user interface implementation went smoothly and contributed to a more robust and portable experimental setup.

Despite these successes, the project also faced challenges. One key difficulty was clock synchronization between multiple SDRs, especially without an external timing reference. The SDRs' internal oscillators exhibited unpredictable drift, which complicated phase-based displacement estimation. Besides, signal amplitude variations and environmental RF noise occasionally interfered with detection performance, particularly in indoor settings. These risks required iterative tuning of signal filters, thresholds, and polynomial drift compensation strategies. Furthermore, achieving robust movement detection under variable signal conditions proved difficult and demanded additional algorithmic refinements.

Some aspects of the project could have been handled better if approached differently. For instance, external clock integration could have been prioritized earlier in the timeline, as it proved crucial to achieving stable and coherent measurements. Additionally, more time allocated to filter optimization and adaptive thresholding might have improved performance under high-noise conditions. More structured testing across varied RF environments and earlier integration of the Raspberry Pi and LCD hardware could have ensured smoother final demonstrations. If the expectations regarding real-time processing had been clarified earlier in discussions with the company, development on this front could have progressed in parallel rather than being deferred to later

stages. Moreover, we lost a lot of time due to delays in the delivery of critical hardware. This could have been avoided if the hardware requirements were finalized earlier and the orders were placed sooner.

One recommendation for future projects of a similar type would be carefully evaluating the hardware specifications beforehand, especially the clock stability of SDRs before starting. In our case, we lost significant time trying to achieve phase correlation using the internal clocks of the ADALM-Pluto SDRs, which turned out to have poor stability. Several methods were used and later disregarded as the need for them diminished when we used external clocks. If an external clock usage would have been considered from the initial stages of this project, no time would have been lost to these unneeded algorithms. Hence, an external clock should be considered from the beginning if precise phase tracking is required. Another important point is the selection of the SDR platform. We proceeded with the ADALM-Pluto because one unit was provided early by the company, allowing us to start development quickly. Nonetheless, alternative SDRs such as HackRF offers better specifications and have broader community support, including more open-source project which could have saved us time for algorithm development and provided greater flexibility.

One major factor beyond our control was the delivery time of critical hardware components. Despite early planning, we experienced unexpected delays which impacted the timeline of the project. Besides, the inherent limitations of the ADALM-Pluto SDR's internal oscillator such as drift and instability posed significant challenges. These issues could not be fully mitigated without external hardware, which was not available at the start of the project.

One of the most unexpected challenges was the behavior of the internal clocks of the SDRs. During experiments, their behavior was often unpredictable and inconsistent unlike our initial expectations. We initially anticipated that the internal clocks of the SDRs would provide sufficient stability for our phase tracking tasks. However, their instability made it difficult to maintain reliable phase alignment. We observed that the clock drift changed rapidly over time, especially after movement or sudden temperature changes. This behavior caused delays in our phase tracking and synchronization processes which is solved using an external clock circuit (see Figure 1). Throughout these processes we avoided mistakes such as committing to a predetermined algorithm without testing alternatives. Instead, we iterated through several implementations, which allowed us to switch from PLL to Kalman Filtering when it proved to be more effective.

We learned a lot throughout the project, including algorithms such as CA-CFAR and Kalman Filtering, as well as external oscillator design for synchronization. We also gained experience with ADALM-Pluto SDRs, built signal processing workflows using GNU Radio, and carried out hardware integration and display implementation on Raspberry Pi. Since most of these topics were not covered in our undergraduate curriculum, we had to learn them independently. We relied on

academic papers, technical documentation from Analog Devices, open-source projects (such as GNSS-SDR), online tutorials for MATLAB and GNU Radio. One of the key sources we used to gain sufficient information about the ADALM-Pluto SDR was the book Software-Defined Radio for Engineers, which provided both theoretical background and practical insights on SDR architecture and usage [30].

7.3 Future Directions

This project has demonstrated the viability of a GNSS-independent delta-positioning system using SDR technology. The system successfully calculated relative motion in both 1D and 2D configurations, achieving a positioning error margin of ± 5 cm, which validates the effectiveness of the signal processing algorithms developed throughout the project. One of the future directions is the complete transfer of signal processing tasks which are currently handled on an external computer onto an embedded platform like the Raspberry Pi. Currently, transmitter programming is the main application of the Raspberry Pi. Nonetheless, a completely independent and small system would be made possible by moving all functions, such as Kalman filtering, and displacement estimation, onto Raspberry Pi. The system can also be modified for real-time operation with additional optimization. This would eliminate the need for post-processing or outside computation by enabling the receiver to continually process incoming signals and update position estimates in real time. The system's applicability would be greatly increased by real-time operation, particularly in dynamic use situations. Finally, the system's modular architecture makes it possible to grow into complete 3D motion estimation. The system may be expanded to track displacement in all three spatial dimensions by adding more transmitters and using geometric processing techniques. The system's technical capabilities and application broadness would be greatly expanded by this improvement, increasing its adaptability for further growth and integration.

8 Equipment List

The equipment list comprises a variety of items needed to complete the project (see Table 2). An Adalm-Pluto SDR was both provided (1 unit) and purchased (3 units) from the company, with the purchased units totalling 27,119.55 TL. The GNU Radio software was downloaded as an open-source tool at no cost. Six MATLAB licenses were provided by the department free of charge. Additionally, a Baofeng UV8R analog radio was borrowed from a team member at no cost. The main component of the project is Adalm-Pluto SDR including components to support RF-to-IQ signal conversion and processing. Its fundamental component is the AD9363 Transceiver, which handles much of the signal processing, including input multiplexing, lownoise amplification, gain control, and signal mixing in the analog domain. Additionally, the AD9363 includes the analog

filters, ADCs/DACs, and fixed digital filters, with programmable 128-tap FIR filters for baseband processing. For demonstration purposes, we used a linear rail to achieve smoother motion and obtain more systematic results. Additionally, we included a stopper and a carriage as part of the setup. Since the internal clocks of the SDR were not functioning reliably, we purchased an external oscillator with better specifications to achieve the desired accuracy.

The Equipment List of the Project					
Equipment	Quantity	Cost per Unit	Total Cost	Acquisition Method	Source
Adalm-Pluto SDR	1	Free	0	Download	Company
Adalm-Pluto SDR	3	9039.8504 TL	27119.5512 TL	Purchase	Company
MATLAB Licence	6	USD 55	0	Download	Department
GNU Radio	6	Free	0	Download	Open Source
Analog Radio Baofeng UV8R	1	USD 40	0	Borrowed	Team Member
Raspberry Pi 5	1	2.499,31 TL	0	Borrowed	Team Member
LCD 2x16 Screen	1	99,47 TL	99,47 TL	Purchase	Company
9V Battery	1	230 TL	230 TL	Purchase	Company
HG15 Linear Rail Stopper - 15 mm	2	110,23 TL	220,46 TL	Purchase	Team Member
HGW 15 CC Wide Linear Carriage	1	359,23 TL	359,23 TL	Purchase	Team Member
HG 15 15 mm Linear Guide Rail	1	718,11 TL	718,11 TL	Purchase	Team Member
40.0000 MHz OCXO Crystal Oscillator, LVCMOS Output	4	71,92 USD	287.68 USD	Purchase	Company

Table 2: The Equipment List of the Project

References

- [1] Inertial Labs, "What Are the Limitations of GNSS?" [Online]. Available: <https://inertiallabs.com/what-are-the-limitations-of-gnss/>. [Accessed: 11-Nov-2024].
- [2] Skybrary, "GNSS Jamming and Spoofing." [Online]. Available: <https://skybrary.aero/articles/gnss-jamming-and-spoofing>. [Accessed: 11-Nov-2024].
- [3] E. Schmidt, D. Inupakutika, R. Mundlamuri, and D. Akopian, "SDR-Fi: Deep-Learning-Based Indoor Positioning via Software-Defined Radio," IEEE Access, vol. 7, pp. 1-1, 2019. doi: 10.1109/ACCESS.2019.2945929.
- [4] Honeywell, "Alternative Navigation Systems," Honeywell Aerospace. [Online]. Available: <https://aerospace.honeywell.com/us/en/products-and-services/product/hardware-and-systems/sensors/alternative-navigation-systems>. [Accessed: Dec. 25, 2024].
- [5] European Commission Joint Research Centre, "Assessing Alternative Positioning, Navigation, and Timing Technologies for Potential Deployment in the EU," Publications Office of the European Union, Luxembourg, 2023. [Online]. Available: https://publications.jrc.ec.europa.eu/repository/bitstream/JRC132737/JRC132737_01.pdf. [Accessed: Dec. 25, 2024].
- [6] DARPA, "Micro-Technology for Positioning, Navigation, and Timing," Defense Advanced Research Projects Agency. [Online]. Available: <https://www.darpa.mil/research/programs/micro-technology-for-positioning-navigation-and-timing>. [Accessed: Dec. 27, 2024].
- [7] Reuters, "Three Finnish airports mitigate Russian GPS interference with radio navigation," Reuters, Nov. 7, 2024. [Online]. Available: <https://www.reuters.com/business/aerospace-defense/three-finnish-airports-mitigate-russian-gps-interference-with-radio-nav> [Accessed: Dec. 28, 2024].
- [8] Google Patents, "Alternative navigation system using terrain, magnetic, and visual sensors," U.S. Patent 10,935,670 B2, issued Mar. 2, 2021. [Online]. Available: <https://patents.google.com/patent/US10935670B2/en>. [Accessed: Dec. 29, 2024].

- [9] Google Patents, "Systems and methods for alternative navigation," U.S. Patent 11,598,884 B2, issued Mar. 7, 2023. [Online]. Available: <https://patents.google.com/patent/US11598884B2/en>. [Accessed: Dec. 29, 2024].
- [10] STM, "Terraflite," STM. [Online]. Available: <https://www.stm.com/tr/tr/cozumlerimiz/komuta-kontrol/terraflite>. [Accessed: Dec. 29, 2024].
- [11] J. A. García, E. S. Lohan, and F. Dovis, "GNSS-SDR: An Open Source Global Navigation Satellite System Software Defined Receiver," 2018 International Conference on Localization and GNSS (ICL-GNSS), IEEE, 2018, pp. 1-6.
- [12] H. G. Myung, J. Lim, and D. J. Goodman, "Single Carrier FDMA for Uplink Wireless Transmission," IEEE Vehicular Technology Magazine, vol. 1, no. 3, pp. 30–38, Sep. 2006.
- [13] W. D. Reeve, "Using the SDRPlay SDR Receivers with an External Frequency Reference," Reeve Observatory, 2020. [Online]. Available: https://reeve.com/Documents/Articles%20Papers/Reeve_SDRPlay-miniGPS.pdf
- [14] A. Patapoutian, "On Phase-Locked Loops and Kalman Filters," IEEE Transactions on Communications, vol. 47, no. 5, pp. 670–672, May 1999.
- [15] CQG Inc., "Kalman Filter," CQG Integrated Client User Guide, [Online]. Available: <https://help.cqg.com/cqgic/25/default.htm#!Documents/kalmanfilter.htm> [Accessed: May 20, 2025].
- [16] P. Machek, "Movement Detection in the Accelerometer Data," 2013. [Online]. Available: https://bmeg.fel.cvut.cz/wp-content/uploads/2013/06/Machek_MOVEMENT_DETECTION_IN_THE_ACCELEROMETER_DATA.pdf. [Accessed: Apr. 3, 2025].
- [17] G. Piccinni, F. Mazzenga, and F. Santucci, "Analysis and Modeling of a Novel SDR-Based High-Precision Positioning System," ResearchGate, [Online]. Available: https://www.researchgate.net/publication/327063429_Analysis_and_Modeling_of_a_Novel_SDR-Based_High-Precision_Positioning_System. [Accessed: Nov. 1, 2024].
- [18] Analog Devices Inc., "ADALM-PLUTO SDR Data Sheet," Analog Devices. [Online]. Available: <https://www.analog.com/en/products/adalm-pluto.html>. [Accessed: Nov. 4, 2024].

- [19] B. J. Ace, "GNSS-SDR Monitor," GitHub Repository, 2023. [Online]. Available: <https://github.com/acebrianjuan/gnss-sdr-monitor>. [Accessed: Dec. 27, 2024].
- [20] J. Smith, "Raspberry Pi GPS Distance Tracker," GitHub Repository, 2023. [Online]. Available: <https://github.com/example/rpi-gps-distance-tracker>. [Accessed: Dec. 27, 2024].
- [21] Ettus Research, "USRP B200 Mini SDR," 2024. [Online]. Available: <https://www.ettus.com/all-products/usrp-b200mini/>. [Accessed: Dec. 27, 2024].
- [22] RTL-SDR Blog, "RTL-SDR V3 Documentation," 2024. [Online]. Available: <https://www rtl-sdr com>. [Accessed: Dec. 27, 2024].
- [23] Analog Devices, Inc., "ADALM-PLUTO Regulatory Compliance," Analog Devices Wiki. [Online]. Available: https://wiki.analog.com/university/tools/pluto/common/regulatory_compliance. [Accessed: Nov. 14, 2024].
- [24] LVT Test Laboratories, "TS EN 61010-1: Safety Requirements for Electrical Equipment for Measurement, Control, and Laboratory Use," LVT Test Laboratories, 2019. [Online]. Available: <https://www.lvt.com.tr/en/makale/ts-en-61010-1-olcme-kontrol-ve-laboratuvarda-kullanilan-elektriksel-don> [Accessed: Nov. 14, 2024].
- [25] IEEE Standard for Safety Levels with Respect to Human Exposure to Electric, Magnetic, and Electromagnetic Fields, 0 Hz to 300 GHz, IEEE Std C95.1-2019, Oct. 2019.
- [26] Turkish Standards Institution, "TS EN 61010-1: Safety requirements for electrical equipment for measurement, control, and laboratory use - Part 1: General requirements," 2011. [Online]. Available: <https://intweb.tse.org.tr/standard/standard/Standard.aspx?081118051115108051104119110104055047105102120088111043113104073101089076> [Accessed: Nov. 14, 2024].
- [27] U.S. Department of Defense, *IS-GPS-200N: Navstar GPS Space Segment/Navigation User Interfaces*, Revision N, May 2021. [Online]. Available: <https://www.gps.gov/technical/icwg/IS-GPS-200N.pdf>
- [28] IEEE 802.11 Working Group, "IEEE 802.11: The Working Group Setting the Standards for Wireless LANs," [Online]. Available: <https://www.ieee802.org/11/>. [Accessed: Nov. 14, 2024].

- [29] Analog Devices, Inc., ADALM-PLUTO SDR Active Learning Module, <http://www.analog.com/media/en/news-marketing-collateral/product-highlight/ADALM-PLUTO-Product-Highlight.pdf>
- [30] A. M. Wyglinski, R. Getz, T. Collins, and D. Pu, *Software-Defined Radio for Engineers*, Artech House, 2018.
- [31] Analog Devices Inc., “ADALM-PLUTO SDR Data Sheet,” Analog Devices. [Online]. Available: <https://www.analog.com/en/products/adalm-pluto.html>. [Accessed: Nov. 4, 2024].

Appendix A - System Requirements

The Functional System Requirements	
#	Requirements
SRS-01-10	Transmission and Reception of Signals
SRS-01-20	Signal Processing Algorithms
SRS-01-30	Distance Measurement

Table 3: The Functional System Requirements

The Signal Transmission and Reception Requirements		
#	Requirements	Project's Requirements
SRS-01-11	Frequency and Bandwidth	The RF transmission and reception components shall operate at a frequency of 1.5 GHz and provide a minimum bandwidth of 2 MHz to ensure effective signal transmission and reception.
SRS-01-12	FDMA	The system shall utilize frequency division multiple access (FDMA) in both phases to differentiate between different transmitters.
SRS-01-13	External Clock Usage	The transmitter SDRs shall be driven with a 40 MHz, 10 ppb accuracy external clocks, to overcome the clock drift caused the internal clock of the SDR devices.

Table 4: The Signal Transmission and Reception Requirements

The Signal Processing Algorithm Requirements		
#	Requirements	Project's Requirements
SRS-01-21	Modulation	The system shall utilize complex modulation at a frequency of 1.5 GHz.
SRS-01-22	Filtering	The system shall recover the modulated signals using low pass filters.
SRS-01-23	Phase Locking	The system shall implement phase locking mechanisms capable of rapid signal locking within 2 seconds.
SRS-01-24	Doppler Frequency Tracking	The system shall use Kalman Filtering, to alleviate the effects of skipped samples and differentiate the doppler frequency from the frequency offset.
SRS-01-25	Movement Detection	The system shall be clearly differentiate the physical displacement of the SDR from the phase discrepancy of the SDR, ensuring that these dispositions are easily identifiable for accurate monitoring and precise control throughout the entire process.

Table 5: The Signal Processing Algorithm Requirements

The Distance Measurement Requirements		
#	Requirements	Project's Requirements
SRS-01-31	1D-Distance Calculation	The system shall calculate the one-dimensional distance between two points (A and B) with an accuracy of ± 5 cm (1-sigma error) to ensure high-precision measurements.
SRS-01-32	2D-Distance Calculation	The system shall calculate the two-dimensional vectorial distance between points with an accuracy of ± 5 cm (1-sigma error) to meet precision requirements.

Table 6: The Distance Measurement Requirements

The Non-Functional System Requirements	
#	Requirement
SRS-02-10	Cost
SRS-02-20	Environmental Interference
SRS-02-30	Temperature Range
SRS-02-40	Power Consumption
SRS-02-50	Safety Issues
SRS-02-60	Health Constraints

Table 7: The Non-Functional System Requirements

The User-Related System Requirements		
#	Requirements	Project's Requirements
SRS-03-10	User-Control	The system shall provide users with an easy interface for initiating and controlling signal transmission, receiving, and processing activities. The interface must allow users to effortlessly transition between 1D and 2D distance measuring modes.
SRS-03-20	Monitoring	The system shall display distance traveled with direction on the user interface. Moreover the distance shall be displayed on the LCD screen connected to Raspberry Pi enabling users a clear and accessible format to monitor system performance effectively [17].
SRS-03-30	Portability	The system shall be designed to be lightweight and compact, ensuring portability for users. The final product's dimensions and weight shall allow it to be easily transported and deployed in various environments. Indeed, this portability is critical for enabling users to calculate distances effectively in field conditions without the need for extensive setup.
SRS-03-40	Documentation and Support	The system shall include related user documentation in order to provide guidance on setup, operation, and troubleshooting [18].

Table 8: The User-Related System Requirements

Appendix B - Big Picture

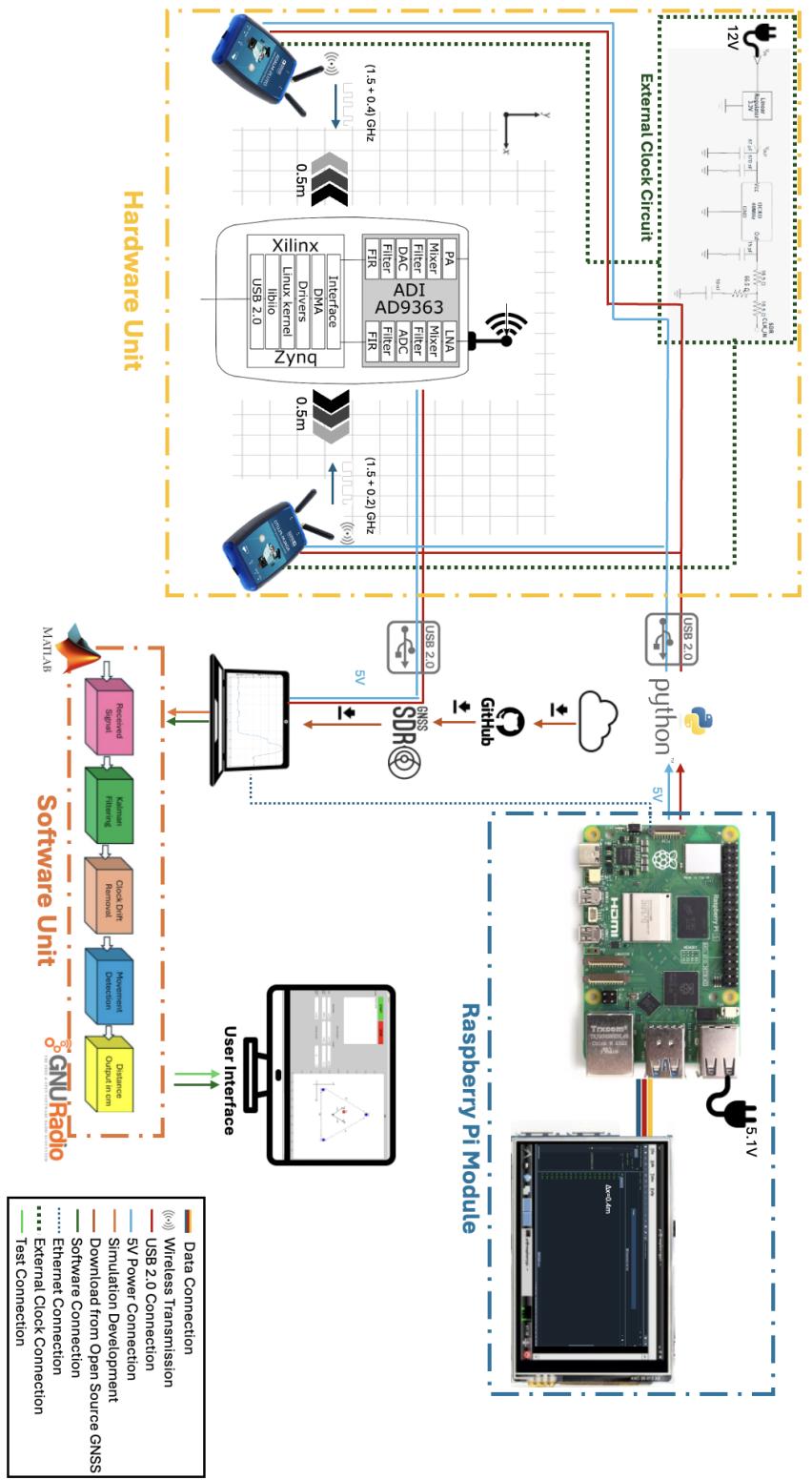


Figure 48: Big Picture of Phase 1 of the Project

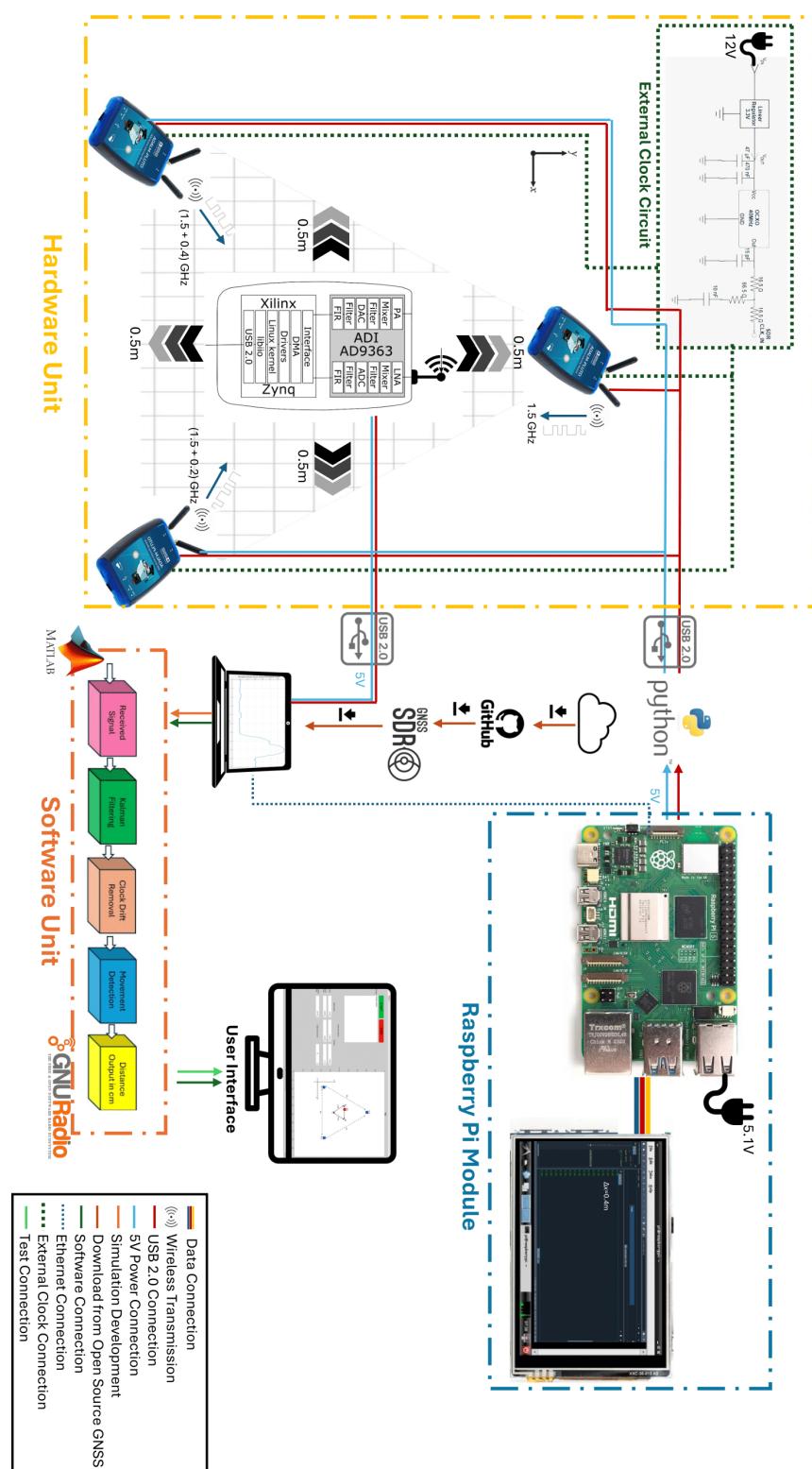


Figure 49: Big Picture of Phase 2 of the Project

Appendix C - Gantt Chart

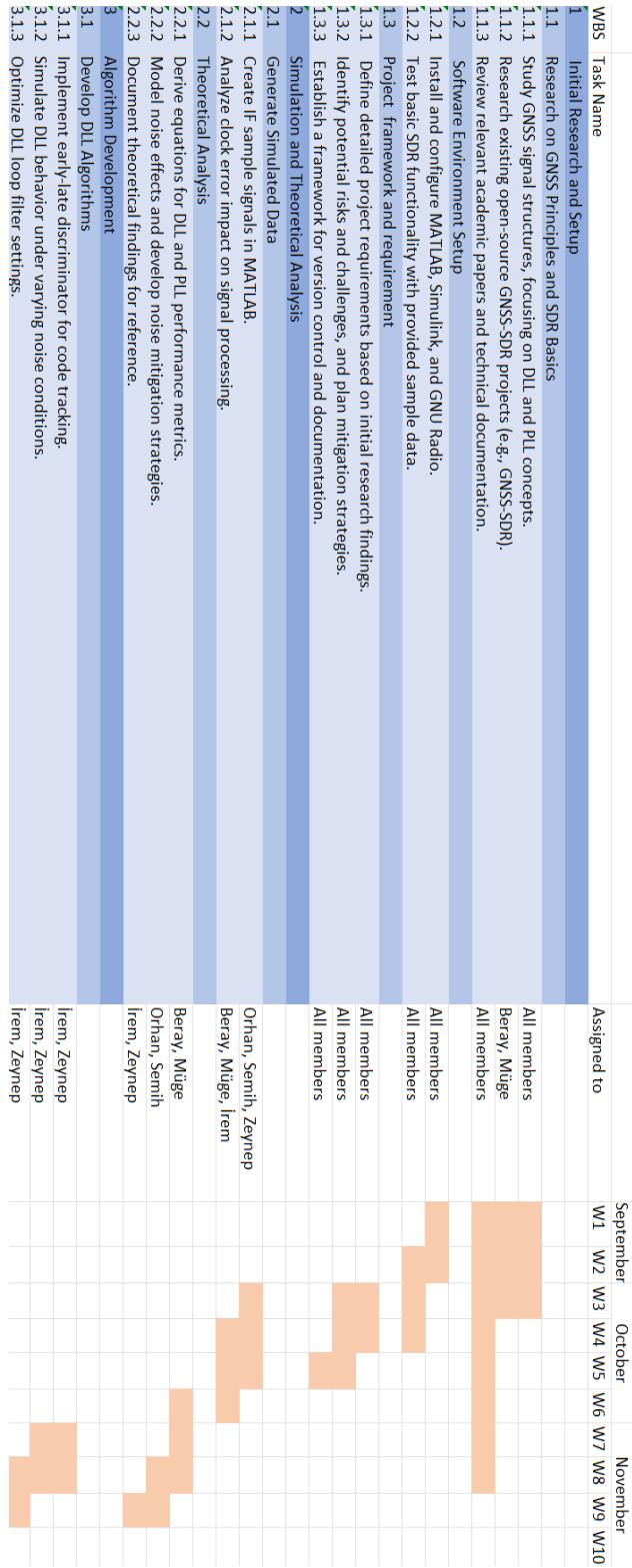


Figure 50: Gantt Chart 1/4

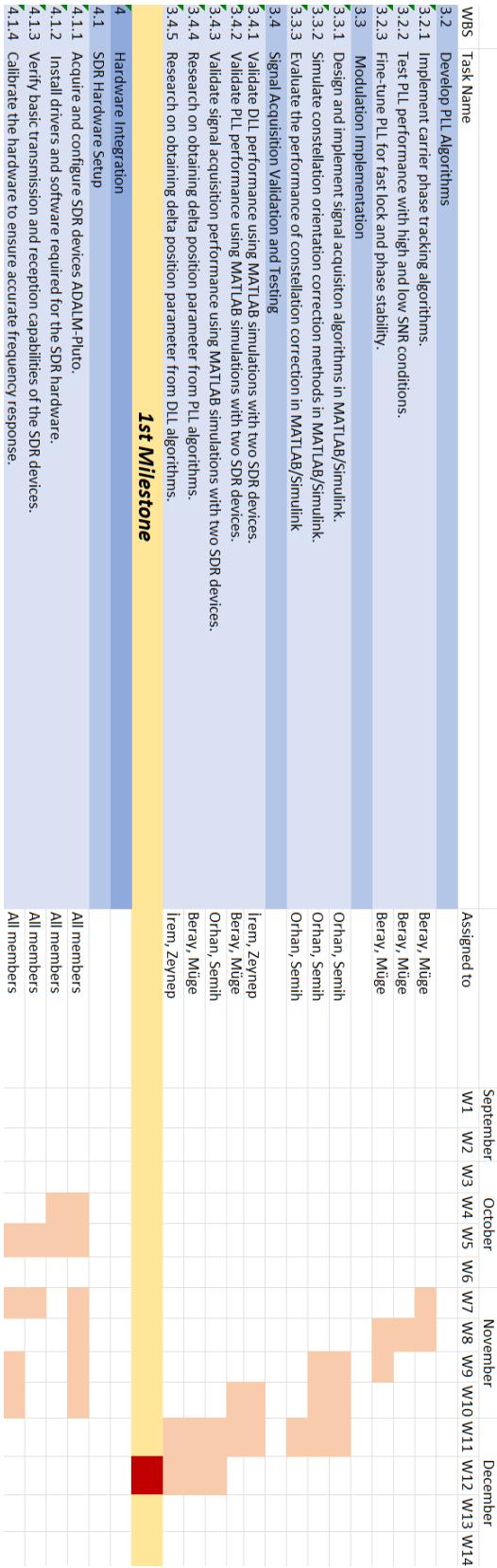


Figure 51: Gantt Chart 2/4

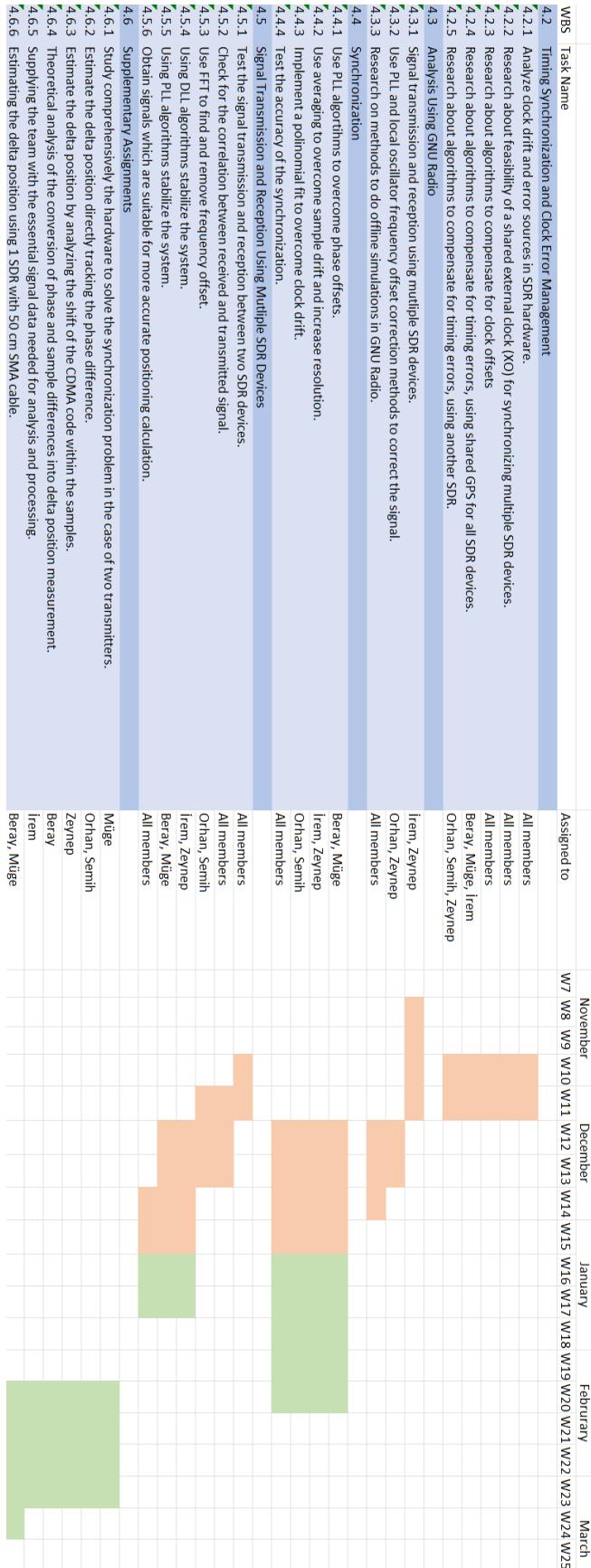


Figure 52: Gantt Chart 3/4
61

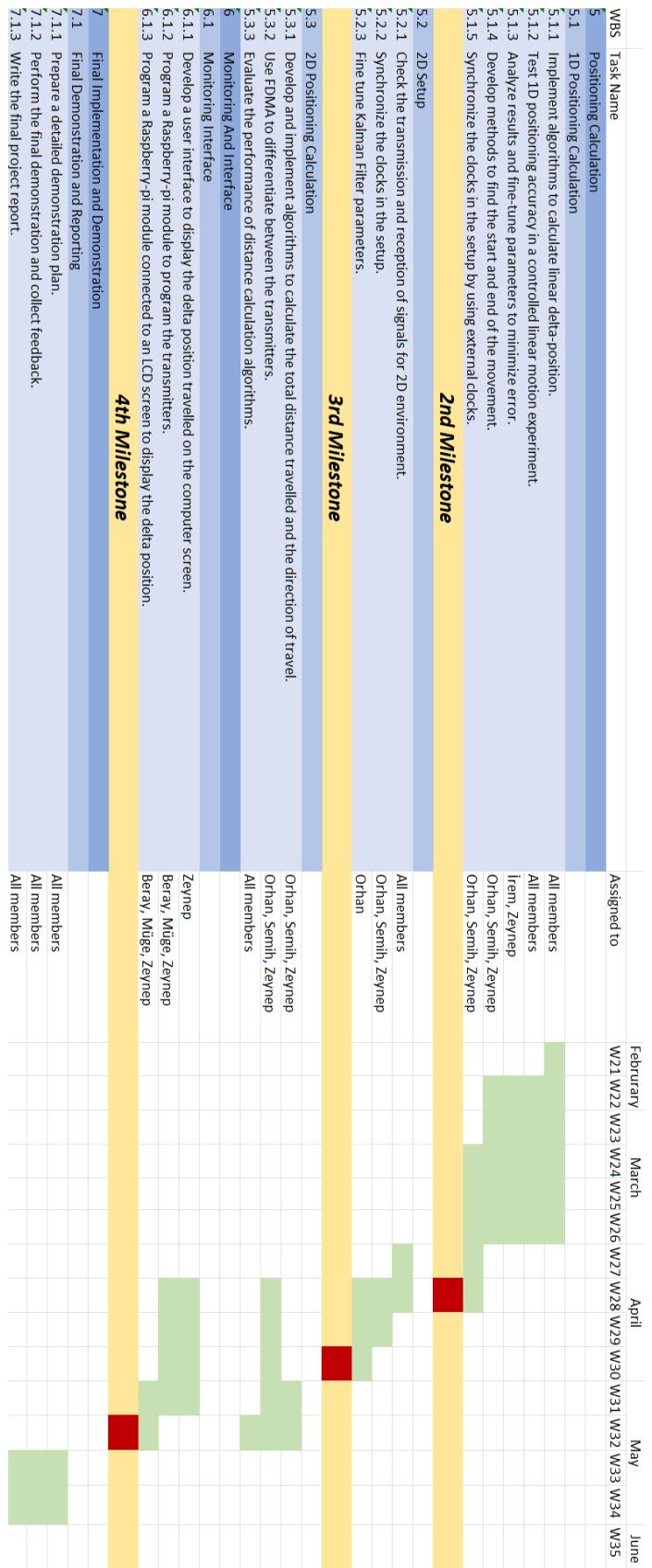


Figure 53: Gantt Chart 4/4

Appendix D - First Setup Algorithm

Algorithm 1 Phase-Based Movement Detection in First Demonstration

```

1: Inputs:  $f_c, f_s, G_{tx}, G_{rx}, c, \lambda = c/f_c, d, N, \epsilon$ 
2:  $t = (0 : 1/f_s : 5)'$ 
3: TX Init: PlutoSDR( $f_c, f_s, G_{tx}$ )
4: TX Send: transmitRepeat(tx, tx_waveform)
5: RX Init: PlutoSDR( $f_c, f_s, G_{rx}$ , frameLength =  $N$ )
6: while true do
7:    $[x, \text{valid}, \_] = \text{rx}()$ 
8:   if valid then
9:      $\phi_1 = \text{mean}(\angle x(1 : N/2))$ 
10:     $\phi_2 = \text{mean}(\angle x(N/2 + 1 : N))$ 
11:     $\Delta\phi = \text{unwrap}(\phi_2 - \phi_1)$ 
12:     $\theta = \arcsin\left(\frac{\Delta\phi \cdot \lambda}{2\pi d}\right) \cdot \frac{180}{\pi}$ 
13:    if  $|\theta| > \epsilon$  then
14:      phase  $\leftarrow$  [phase,  $10 \cdot \Delta\phi$ ]
15:      total_phasediff  $\leftarrow$  total_phasediff +  $|10 \cdot \Delta\phi|$ 
16:    end if
17:    if  $\neg$ move_triggered and  $|\theta| > 0.15$  then
18:      move_triggered  $\leftarrow$  true
19:      data_count  $\leftarrow$  0
20:    end if
21:    if move_triggered then
22:      data_count  $\leftarrow$  data_count + 1
23:      if  $|\theta| < 0.03$  then
24:         $\Delta x = \frac{\text{total\_phasediff}}{2\pi} \cdot \lambda \cdot 100$ 
25:        distance_array  $\leftarrow$  [distance_array,  $\Delta x$ ]
26:        move_triggered  $\leftarrow$  false
27:      end if
28:    end if
29:  end if
30: end while

```

Appendix E - Experimental Results for the Setup with SMA Cable

Table 9: Theoretical and Experimental Distances Computed in the Setup with SMA Cable

Theoretical Distance [cm]	Experimental Distance [cm]	Error (%)
20.00	22.20	11.00
10.00	15.41	54.10
15.00	13.80	8.00
7.00	8.60	22.86
10.00	13.90	39.00
7.00	6.30	10.00
18.00	21.00	16.67
50.00	56.00	12.00
50.00	51.00	2.00
50.00	58.00	16.00
4.00	4.10	2.50
14.00	24.00	71.43
20.00	22.00	10.00
7.50	7.70	2.67
7.00	10.00	42.86
18.00	16.70	7.22
4.00	7.70	92.50
30.00	20.00	33.33
15.00	13.40	10.67
20.00	37.00	85.00
20.00	20.00	0.00
9.00	7.90	12.22
10.00	10.00	0.00
5.00	5.00	0.00
9.00	12.00	33.33
10.00	10.00	0.00
15.00	20.00	33.33
15.00	13.50	10.00

Continued on next page

Table 9 – continued from previous page

Theoretical Value [cm]	Experimental Value [cm]	Error (%)
9.00	6.90	23.33
50.00	50.60	1.20
50.00	33.00	34.00
10.00	8.00	20.00
10.00	8.30	17.00
10.00	11.00	10.00
5.00	3.40	32.00
20.00	21.00	5.00
40.42	42.16	4.29
15.23	13.74	9.79
4.88	5.30	8.43
34.21	37.49	9.58
22.01	24.05	9.26
6.10	5.66	7.19
24.76	22.42	9.45
1.72	1.66	3.62
45.47	43.48	4.37
12.94	12.51	3.32
33.13	31.38	5.28
15.59	14.69	5.72
26.00	24.73	4.90
27.34	24.93	8.80

Appendix F - Experimental Results for Phase 1

Table 10: Theoretical and Experimental Distances Computed in Phase 1

Run ID	Theoretical [cm]	Experimental [cm]	Error (%)
1	11.00	12.00	9.09
2	15.00	17.00	13.33
3	10.00	9.91	0.90
4	30.00	33.00	10.00
5	50.00	42.00	16.00
6	50.00	51.00	2.00
7	50.00	55.00	10.00
8	35.00	40.00	14.29
9	50.00	49.00	2.00
10	50.00	57.00	14.00
11	40.00	45.00	12.50
12	50.00	46.00	8.00
13	49.00	47.00	4.08
14	35.00	43.00	22.86
15	50.00	49.00	2.00
16	50.00	52.00	4.00
17	52.00	59.00	13.46
18	50.00	55.00	10.00
19	12.00	12.30	2.50
20	17.00	15.00	11.76
21	11.00	11.00	0.00
22	14.00	17.00	21.43
23	14.00	10.00	28.57
24	16.00	11.00	31.25
25	14.00	19.00	35.71
26	22.00	21.00	4.55
27	12.00	9.00	25.00
28	5.00	4.50	10.00
29	10.00	12.00	20.00
30	15.00	14.00	6.67
31	30.00	30.50	1.67
32	10.00	10.00	0.00
33	25.00	23.00	8.00
34	17.00	11.00	35.29
35	23.00	22.00	4.35

Continued on next page

Table 10 – continued

Run ID	Theoretical [cm]	Experimental [cm]	Error (%)
36	30.00	25.00	16.67
37	45.00	33.00	26.67
38	11.00	12.00	9.09
39	11.00	8.00	27.27
40	15.00	10.00	33.33
41	15.00	13.00	13.33
42	30.00	32.00	6.67
43	50.00	49.50	1.00
44	20.00	21.00	5.00
45	20.00	22.00	10.00
46	15.00	16.00	6.67
47	30.00	32.00	6.67
48	30.00	38.00	26.67
49	12.00	14.00	16.67
50	11.00	12.00	9.09
51	19.00	20.00	5.26
52	13.00	15.00	15.38
53	40.00	41.00	2.50
54	44.00	42.00	4.55
55	14.00	15.00	7.14
56	13.00	10.00	23.08
57	5.00	3.80	24.00
58	7.00	9.00	28.57
59	10.00	12.00	20.00
60	30.00	35.00	16.67
61	35.00	38.00	8.57
62	12.00	14.00	16.67
63	10.00	14.00	40.00
64	5.00	5.00	0.00
65	20.00	15.00	25.00
66	32.00	36.00	12.50
67	17.00	16.00	5.88
68	3.00	2.00	33.33
69	17.00	20.00	17.65
70	25.00	28.00	12.00
71	18.00	15.00	16.67
72	16.00	18.00	12.50
73	23.00	21.00	8.70
74	12.00	13.00	8.33

Continued on next page

Table 10 – continued

Run ID	Theoretical [cm]	Experimental [cm]	Error (%)
75	17.00	18.00	5.88
76	25.00	26.00	4.00
77	13.00	13.70	5.38
78	18.00	18.00	0.00
79	24.00	23.00	4.17
80	25.00	24.00	4.00
81	46.00	45.00	2.17
82	22.00	24.00	9.09
83	16.00	17.00	6.25
84	15.00	16.00	6.67
85	17.00	15.00	11.76
86	28.00	26.00	7.14
87	12.00	15.00	25.00
88	30.00	31.00	3.33
89	40.00	40.00	0.00
90	12.00	9.00	25.00
91	15.00	18.00	20.00
92	25.00	24.30	2.80
93	12.00	13.00	8.33
94	25.00	28.00	12.00
95	10.00	11.00	10.00
96	34.00	35.00	2.94
97	15.00	17.00	13.33
98	40.00	40.00	0.00
99	12.00	15.00	25.00
100	18.00	18.30	1.67