The Architecture Design and Hardware Implementation of Communications and High-Precision Positioning System

by

Hanguang Yu

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved March 2020 by the
Graduate Supervisory Committee:

Daniel Bliss, Chair
Chaitali Chakrabarti
Ahmed Alkhateeb
Umit Ogras

ARIZONA STATE UNIVERSITY

May 2020

ABSTRACT

Within the near future, a vast demand for autonomous vehicular techniques can be forecast on both aviation and ground platforms, including autonomous driving, automatic landing, air traffic management. These techniques usually rely on the positioning system and the communication system independently, where it potentially causes spectrum congestion. Inspired by the spectrum sharing technique, Communications and High-Precision Positioning (CHP2) system is invented to provide a high precision position service (precision $\sim$1cm) while performing the communication task simultaneously under the same spectrum. CHP2 system is implemented on the consumer-off-the-shelf (COTS) software-defined radio (SDR) platform with customized hardware. Taking the advantages of the SDR platform, the completed baseband processing chain, time-of-arrival estimation (ToA), time-of-flight estimation (ToF) are mathematically modeled and then implemented onto the system-on-chip (SoC) system. Due to the compact size and cost economy, CHP2 system can be installed on different aerial or ground platforms enabling a high-mobile and reconfigurable network.

In this dissertation report, the implementation procedure of the CHP2 is discussed in detail. It mainly focuses on the system construction on the Xilinx Ultrascale+ SoC platform. The CHP2 waveform design, ToA solution, and timing exchanging algorithms are also introduced. Finally, several in-lab tests and over-the-air demonstrations are conducted. The demonstration shows the best ranging performance achieves the $\sim$ 1 cm standard deviation and 10Hz refreshing rate of estimation by using a 10MHz narrow-band signal over 915MHz (US ISM) or 783MHz (EU Licensed) carrier frequency.

ACKNOWLEDGMENTS

First, I want to thank my great mentor Dr. Daniel Bliss for his wisdom and patience to lead my accomplishment through many years. I want to thank my colleagues for their tremendous contributions to this fantastic project, notably including Andrew Herschfelt, Sharanya Srinivas, Yang Li, Shunyao Wu, Hyunseok Lee, Brittany McCall, Dr. Chaitali Chakrabarti, and all my other lab-mates. I am appropriate for this opportunity to work with these talented people. I also would like to thank all my committee members include Dr. Daniel Bliss, Dr. Chaitali Chakrabarti, Dr. Ahmed Alkhateeb, and Dr. Umit Ogras for their time and valuable feedback. Finally, I would like to thank my parents for encouraging and supporting me through many years' study aboard.

Airbus ExO Alpha funds this research project through the Hyper-Precise Positioning and Communications (HPPC) program. Thank you to Nunzio Sciammetta, Leslie Smith, Klaus Rueger, and the entire Airbus team for making this possible.

At this moment, the whole world is going through this unprecedented coronavirus epidemic. No matter where we are from, what colors we are, which ideologies we have, we are united to overcome this inviable enemy together.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Chapter 1

INTRODUCTION

Back in 2017, the plane MH370 disappearing on the Indian ocean shocks the whole world. Without sufficient positioning and surveillant information, the investigators only recovered the potential flight path after a couple of months later, and the exact location of missing is still a historical mystery. This tragedy reveals the issue of current aviation management. In the modern-day, the navigation and surveillant technique is not sufficient to cover every single space on earth. Most of the earth's surface is not covered by radar surveillant systems, especially over the ocean. And Global Positioning System (GPS) system only provides a one-way ranging service, and its RF signal is easy to spoofing. The Automatic dependent surveillance—broadcast (ADB-S) also replies on GPS providing the positioning information. Meanwhile, these legacy approaches provide services using independent radio systems, which increase spectral congestion and introduce mutual interference. Therefore, we use the co-designed and co-used technique to design this CHP2[1, 2] system for positioning and communication purpose of overcoming those issues for multiple vehicular application.

CHP2 is a novel radio system that serves as navigation positioning purpose while performs communication among CHP2 nodes simultaneously. The joint positioning-communication solution improves spectrum utilization efficiency. It retains much less bandwidth than those traditional ranging systems that require large bandwidth for maintaining the same precision as CHP2. Currently, CHP2 achieves $\sim$ 1cm standard deviation ranging precision. Thus, the small form factor and low cost of gives CHP2 a chance to implement itself on different platforms, for example, consumer-level unmanned aerial vehicle (UAV), autonomous driving vehicle, and even flying

vehicles in the future. This high precision enables applications such as automated landing, collision avoidance, target tracking, air traffic management, and surveillant in a cost-efficient solution without installing a large scale of infrastructure.

The CHP2 system is a team project combining the contributions of all team members. Thus, the dissertation report not only reflects the contritions from the author but also includes others' contributions. Chapter (2) introduces the CHP2 system model, including the timing model and signal model. It also reviews a simple network timing protocol (NTP) as the ToF estimation method underlying the CHP2 concept in Chapter (4), as well as the ToA estimation in Chapter (3). The author also contributes to the CHP2 waveform payload design in Chapter (6) and range-positioning converting implementation in Chapter (5). But the report focuses on the significant contribution from the author in Chapter (7) and Chapter (8), such as the hardware construction and processing chain implementation on the Xilinx SoC platform, including the CHP2 SoC architecture design and processing firmware design. Finally, the author makes major contributions to the in-lab tests and over-the-air demonstration. The experiment details and results analysis are discussed in Chapter 9.

1.1 System Overview

CHP2 is a 4×4 MIMO system that performs positioning and communications tasks simultaneously by exchanging network timing information and navigation reference sequences using a single, co-used waveform. This waveform includes a communications payload, which contains shared timing information and arbitrary communication content between users. The navigation reference sequences are used to precisely estimate the time-of-arrival (ToA) at each receive antenna. In this ToA based two-way ranging solution, these ToA estimates and the shared timing information drive a syn-

chronization algorithm that estimates the time-of-flight (ToF) between user antenna pairs and digitally synchronizes the user clocks. This synchronization algorithm can be the standard Network Timing Protocol (NTP)[3, 4] or our novel ToF estimation solution. Usually, the ranging performance naturally depends on the RF signal bandwidth. However, the high accuracy ToA measurement is achieved by the oversampling over the sub-wavelength level of the carrier signal and estimated phase information.

In a typical application, one CHP2 node is a ground-based station, and the second is on a unmanned aerial system (UAS). Both of them employ four antennae. These antennas alternate transmitting and receiving the joint positioning-communications waveform, which contains the information necessary to feed the time synchronization algorithm and achieve the ToF estimation. All 16 pairs of distance measurements from a multiple-input multiple-output (MIMO) setup are estimated. If the geometry of both nodes is known, these distance estimates can be converted into position and orientation information.

On a variety of platforms, as Figure (1.1) shown, the flexibility and reconfigurability of CHP2 enable diverse network configurations. CHP2 supports both ground and aerial platforms and can adapt to both close-range and long-range applications. Ground-to-ground links enable applications such as traffic management, knowledge distribution, and automatic navigation. Air-to-ground links enable applications such as air traffic management (ATM), flexible traffic surveillance, precise positioning, navigation, and timing (PNT), automated landing, and communications, navigation, and surveillance (CNS). Air-to-air enables airborne applications such as collision avoidance, formation control, PNT, and CNS without relying on satellite links, which are susceptible to spoofing cyberattacks, or ground links, which are not always available.

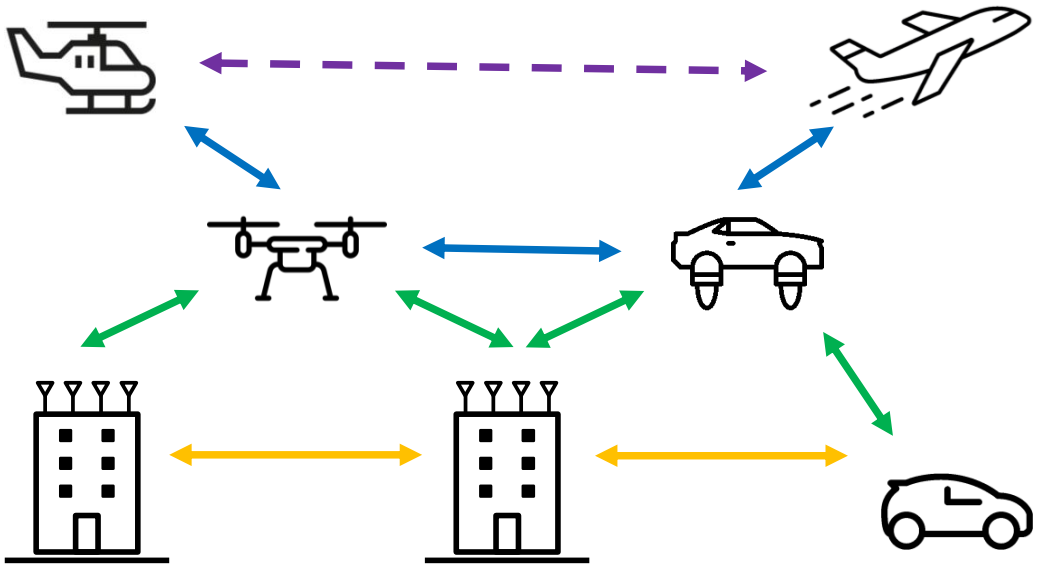


Figure 1.1: *Typical applications of the CHP2 system. It provides solution for ranging and navigation service on Ground-to-ground (G2G, orange), Air-to-ground (A2G, green), and Air-to-air (A2A, blue, purple) scenarios.*

1.1.1 Novelty

The basic premise of the CHP2 technology is simple but is only enabled by several novel innovations[1]. This includes:

1. Implement a novel ToA-based alternative positioning, navigation, and timing (APNT)[5]: The CHP2 system is implemented on consumer-off-the-shelf (COTS) hardware, which requires custom digital logical processing chain and firmware to support the system functionality. It minimizes the necessary infrastructure to implement a working system and reduces the cost and labor of building user platforms.

2. Co-use waveform design: A joint positioning-communications waveform is designed to combine both tasks into a single waveform inspired by the concept of

RF convergence [6], which is the co-design and co-use of the waveform. This mitigates mutual interference and enables high-spectrum efficiency.

3. Sub-carrier level ToA estimation: The ToF between antenna pairs are estimated by finding the difference between transmit and receive timestamps. The transmit timestamps are shared in the communications payload, but the received timestamps ToA must be estimated. A sub-carrier level accurate ToA estimator is designed and implemented to produce the high precession receiving timestamps.

4. Time synchronization and ToF estimation algorithm: An algorithm is designed that synchronizes two users during their cyclic exchanges. This algorithm precisely estimates and tracks the differences between the user clocks and allows them to achieve distributed coherence. This coherence enables high-precision ToF estimation.

1.1.2 Advantages and Disadvantages

The CHP2 system provides the positioning service and performs communication tasks simultaneously using the spectrum sharing technique. Comparing to the traditional system, such that separates positioning and communication services onto independent radio systems, it has several advantages over similar technologies, along with disadvantages. The most notable advantages are:

1. Limited infrastructure: The CHP2 system does not require infrastructure, such as a mesh of satellites, to perform its function. It eliminates issues such as inadequate coverage or limited access.

2. Low cost: The CHP2 system is implemented on consumer-grade, programmable radio platforms. It is a low-cost alternative to designing and fabricating custom

hardware for a specific implementation, allowing flexibility in applications.

3. High precision: The ranging precision of CHP2 system is centimeter-level with a 10 MHz bandwidth. It is much higher precision than similar technologies operating with similar bandwidths [7, 8].

4. Low resources: The CHP2 operates efficiently at relatively low RF signal bandwidth, allowing it to fit into crowded spectral environments and operate with limited resources.

5. Security: The positioning and communications tasks are performed simultaneously, so additional security on the communications payload, such as encryption, increases the security of the positioning system.

The CHP2 system also has several disadvantages, including:

1. Limited to line-of-sight: The nature of the link between two CHP2 users limits it to applications with line-of-sight (LOS) between the two platforms. Both the positioning and synchronization functions degrade dramatically without LOS.

2. Limited infrastructure: While minimal infrastructure is advantageous in some applications, the lack of an existing framework can be a disadvantage in others. In the case of GPS receivers, for example, a user can be added into the network at very low cost and no change to the satellite infrastructure, while the CHP2 network is less flexible in this regard.

3. Active cooperation: The CHP2 network requires active cooperation between the users. If one of the users has a failure, then the entire link will fail, whereas a passive cooperation scheme may be more robust against individual failures.

1.1.3 Applications

This report focuses on the application of Flexible Radio technology to aircraft and drone platforms. These include positioning tasks such as takeoff, landing, taxi, and formation flying, as well as remote control tasks such as drone formation flying and cruise control. We also develop specific applications, including a total position state estimation system and autonomous landing. These applications are depicted in Figures 1.2.

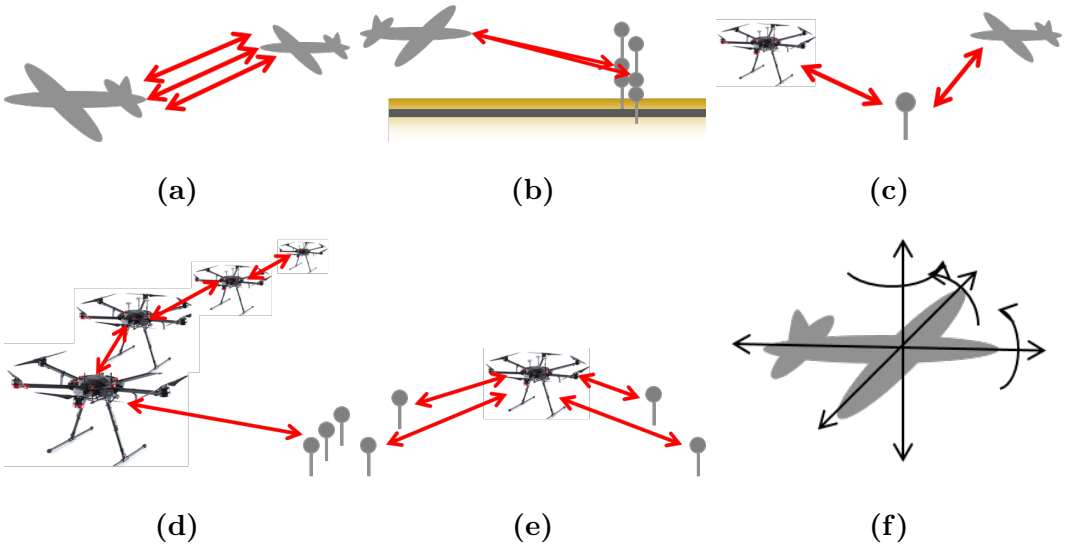


Figure 1.2: *Potential applications of the developed technologies include: (a) vehicle-to-vehicle communications and positioning, (b) landing control, (c) air-to-ground communications and positioning, (d) drone remote flight control, (e) automated drone landing, and (f) complex aircraft positioning.*

1.2 Background

The CHP2 system is a fusion of innovations in several fields of study, the most relevant being RF convergence, joint waveform design, time-of-arrival estimation, distributed coherence, and positioning systems.

1.2.1 RF Convergence

Increasing spectral congestion limits the capabilities and opportunities of legacy radio systems. Radio-Frequency (RF) Convergence refers to a growing movement of co-operation, co-existence, and co-design techniques that allow radio systems to adapt to cluttered environments and exploit their neighbors for mutual benefit [6]. Many system architectures may benefit from this modern approach, including vehicle-to-vehicle (V2V) radar and communications [9, 10], flight control [11–13], and asset tracking [14, 15].

One strategy for achieving RF convergence is to design waveforms that simultaneously perform multiple tasks. This can take various forms depending on the application but can be broadly characterized as parasitic radar or embedded communications. Parasitic radar refers to communications waveforms that have been treated to possess suitable radar performance characteristics, such as direct sequence spread spectrum (DSSS) [16–18] and orthogonal frequency-division multiplexing (OFDM). Embedded communications refer to embedding communications data into a radar waveform. One common example is chirp spread-spectrum (CSS), which encodes data in the phase of a linear frequency modulated (LFM) chirp waveforms [19–22]. Other techniques include polyphase-coded FM (PCFM) [23, 24], continuous phase modulation (CPM) [25], and FM noise radar waveforms [26].

1.2.2 Time-of-Arrival Estimation

Time-of-Arrival estimation refers to estimate the arrival time of a certain RF signal. It is limited by how well the transmitter and receiver are synchronized. Several estimators are formulated and compared in [27] and [28] in the context of ranging and clock synchronization. Using ToA estimates has previously been considered for implementing location and ranging systems [29, 30], and has recently become accessible as a low-cost localization solution [31]. The most famous example of ToA is GPS system, which consists of a large constellation of satellites provides user to triangulate its position. GPS costs tremendous resource and couple decades to provide a global scale geo-location service. Nowadays, as the software-defined radio (SDR) evolving toward more reliable and cost economy, ToA positioning are much easier to build as local APNT.

1.2.3 Distributed Coherence

Distributed coherence refers to a network of users that have been synchronized and form a coherent system. CHP2 enables distributed coherence implicitly by synchronizing each user and estimating the relative position of every node in the network. One of the original time synchronization algorithms is known as the Network Timing Protocol (NTP) [3, 4]. Many time synchronization algorithms are used in modern sensor networks [32, 33]. In this study, we use an extended version of NTP that jointly estimates clock parameters and ToF [27]. The literature [28] describes a simplified but similar setup of ranging and clock synchronization using ToA exchanging.

1.2.4 *Positioning Systems and Alternative Positioning System*

Positioning and localization are persistent problems with numerous solutions. A complete survey is beyond the scope of this investigation. The most relevant technologies related are briefly discussed below. Currently, the popular technologies for Vehicle-to-vehicle (V2V) positioning include radar systems [34], LiDAR [9, 35], optical systems [36, 37], and RFID [14]. The most common positioning system is Global Navigation Satellite System (GNSS). This system utilizes a mesh of satellites to jointly estimate position and time, which enables numerous applications [7, 8]. The GNSS is the most common PNT service, but may not meet reliability requirements for particularly sensitive applications [7, 38]. alternative positioning, navigation, and timing (APNT) systems offer similar services using non-traditional solutions, which are independent of the GNSS satellite network. In [5], the authors discuss 5 classifications of APNT technologies: distance measuring equipment (DME) [5, 39], passive wide area multilateration (P-WAM) [40–44], Pseudolite (PL) [45], VHF omnidirectional range (VOR) [5], and L-band Digital Aeronautical Communication System 1 (LDACS1) [46]. Recently, ultra-wideband (UWB)[47] system has been proved an excellent performance on RF ranging but only provides a limited range and consumes large RF bandwidth.

1.2.5 *Software Defined Radio*

The beauties of SDR are application flexibility and cost economy. The building of a multitask radio system does not require multiple dedicated radio hardware anymore. Different radio architectures can be integrated onto a single full digitized radio platform. Different tasks can be controlled and switched by replacing waveform or radio protocols on the software side. The development of a particular radio architec-

ture is aggressively simplified thanks to the entirely digitized platform. In general, most of the radio architectures are implemented onto the programmable local block (e.g., FPGA) and general processing unit (e.g., ARM, DSP). The SDR structure of CHP2 is based on the Analog Device SDR system [48].

1.3 Comapre with APNT and PNT performance

In Table 1.1, we compare CHP2 performance with the major positioning, navigation, and timing (PNT) systems, for example, Wide Area Augmentation System (WAAS)[49], Real-time kinematic (RTK)[50] and the APNT systems including DME, L band digital aviation communication systems (LDACS), Ultra-High Accuracy Reference System (UHARS), and automatic dependent surveillance – broadcast (ADS-B)[39, 40, 43, 47, 49–53]. Compared to the traditional PNT and APNT system, the CHP2 system achieves a high precision with less spectrum consumption and less infrastructure. The whole system is implemented based on the commercial available SDR platform. It is highly integrated and highly compact. Not only the high precision ranging, cost economy the CHP2 provides, but also it can be used to construct a two-way communication network which can prevent signal tapping and improve security. The hardware architecture of CHP2 supports up to sub-centimeter ranging precision. However, the precision also depends on the chosen ToF algorithm. In general, the CHP2 system ranging performance is from subcentimeter to less than 5 cm standard deviation depends on the chosen ToF method and the fine ToA estimator revisions.

Table 1.1: *Compare CHP2 performance with PNT and APNT*

Parameter	Signal Bandwidth	Range Accuracy	Scale
GPS WAAS	15.345MHz	∼1m	global
UWB[47]	1GHz	10 cm	15 m
GPS RTK	15.345MH	1.1 cm	global
DME	252×1 MHz	370.4m(standard)	50 km(Close)
		100m(improved)	75 km(Mid)
			250 km(Long)
LDACS2	200kHz	Similar as LDACS1	370 km
LDACS1	500kHz	∼20 m	370 km
UHARS	10.23MHz	<18 cm	>48 km
ADS-B (1090MHz)	50kHz	50-100 m	250 km
CHP2	10MHz	∼1 cm	>10 km

1.4 Technical Challenges

The CHP2 system achieves communications, and positioning tasks with higher precision, fewer spectrum resources, higher reliability than traditional systems. To achieve this goal, we address a couple of technical challenges.

1.4.1 Precision

Traditionally, the spectrum bandwidth constrains its precision of a ranging system. The traditional approach is to sample a received signal and then correlate it against a known reference. This correlation is usually performed at the critical sampling frequency (bandwidth) or some small multiple thereof. This approach is limited to the time difference between signal samples, or the inverse of the system bandwidth.

The target precision for the CHP2 system is 100 times more precise than this intrinsic resolution. The approach to achieving this precision requires high integrated SNR, carrier-phase-accurate time synchronization, and high precision ToA estimation.

1.4.2 Distributed Phase Coherence

For achieving high precision ToF estimation, the clocks on each radio platform must be precisely synchronized. For a sub-carrier accurate ToA estimate to be useful, this synchronization must be precise to within a fraction of a carrier cycle. The CHP2 uses a timing exchange protocol that targets this precision.

1.4.3 Implementation Difficulty

All of the signal processing algorithms and timing estimation for ToA estimation and ToF estimation are integrated onto a single SoC system, which is the Xilinx Ultrascale Plus platform[54]. Due to the high fresh rate requirement (two-way communication and ranging withing 100ms), the large portion of heavy load computation tasks are pushed onto the PL section and implemented as the HDL IP blocks. It significantly increases the development cost compared to the conventional SDR system. Meanwhile, the project requires a portable, reliable, and working system for the demonstrations on the actual field. A lot of effort and collaboration is needed for hardware implementation. For example, it includes the customized PCB design and manufacturing, hardware mounting and assembling, system interfacing, and packaging. Given a short period and lack of enough experienced developers, it is far more than a lab-level concept proof.

1.5 Objectives

This manual address the following objectives:

- System Design
  - Design the system architecture of CHP2
  - Design joint positioning-communications waveforms, protocols, and algorithms.
- Implementation
  - CHP2 radio system hardware architecture
  - Baseband processing for communication link
  - Baseband processing for positioning sequence
  - Realtime ToA estimation, NTP (and CHP2 NTP) algorithm, Kalman filter based CHP2 NTP algorithm on baseband processor
- Experimental Validation
  - Evaluate performance in laboratory experiments.
  - Evaluate performance in scaled airborne experiments.

1.6 Contributions

As a team project, CHP2 system is a summation of contributions from multiple team members. The author of this report significantly contributes the following items to the CHP2 project.

- Design
  - CHP2 system architecture on Zynq UltraScale+ MPSoC
  - CHP2 communication waveform and its baseband processing chain
  - Accelerated ToA estimator
- Implementation
  - Modification and adaptation for the SDR platform of the CHP2 system
  - Accelerator and firmware for communication signal baseband processing chain
  - Direct Memory Access (DMA) accelerated ToA estimator
  - Firmware for ToF estimation
  - Firmware for range-position conversion
  - RF hardware interfacing and tuning
  - System integration, stability, and user-interface
- Experiments
  - Setup and evaluate the laboratory experiments.
  - Setup and evaluate the airborne experiments with the Airbus team.

Chapter 2

MODEL DEFINITIONS[1]

In a perfect ideal scenario, two independent CHP2 nodes exchange the transmission and reception timestamps of each frame to estimate the time-of-flight (ToF) and timer offset between them. This case is a typical Network Time Protocol (NTP) application. However, as the CHP2 system performance requirement, all kinds of reality negative factors need to be addressed to achieve the centimeter-level accuracy. Those two nodes operate with independent clock sources and propagate a bandlimited signal over a long distance through the wireless channel. For example, local clock frequency offset, signal distortion, phase offset, and frequency dilation caused by Doppler and clock drift can contribute to the collapse of ranging accuracy. This chapter introduces the high-fidelity time and propagation models from the previous works [1, 2], upon which the ToA estimators and time synchronization algorithms are built.

2.1 Time Model

To accurately estimate the time-of-flight between two CHP2 nodes, the clocks must be carefully synchronized. To achieved this goal, the behavior of independent clocks is studied. Thus, in these two CHP2 nodes case, the two distributed and independent clocks are modeled. Also, a set of variables describing the clock parameters and modeling their behaviors are defined in this section.

[1] Andrew Herschfelt contributes to most of the timing and signal modeling of the CHP2 system.

2.1.1 Time Model Variables

In this scenario of two CHP2 radios, define a master Node A and a slave Node B. Record the exact time of transmission and reception events on each radio as the corresponding timestamps. A set of the defined variable is shown as the following table.

Table 2.1: *Time Variable Definitions*

Label	Description
$t_{A,Tx}$	Transmission timestamp of Node A
$t_{A,Rx}$	Reception timestamp of Node A
$t_{B,Tx}$	Transmission timestamp of Node B
$t_{B,Rx}$	Reception timestamp of Node B
t_A	Time on Node A
t_B	Time on Node B
τ	Time-of-Flight
T	Time offset between Node A and Node B
n	Index of a set of N time frames
$\dot{\tau}$	First order derivative of time-of-flight
$\dot{T}$	First order derivative of time offset
l	The time cycling period of CHP2 frames

A transmission from A to B takes approximately τ seconds to propagate, where $\tau = d/c$ and c is the speed of light in the medium. To estimate τ, where d is the distance between the remote nodes, the radios alternate transmitting and receiving a joint positioning-communications waveform, as depicted in Figure 2.1. Assuming a reality case of the misaligned clocks, define an offset T between the times displayed

on clocks A and B at a given instant.

$$T = t_A - t_B. \tag{2.1}$$

T and τ are then jointly estimated to synchronize the independent clocks and estimate the distance between platforms.

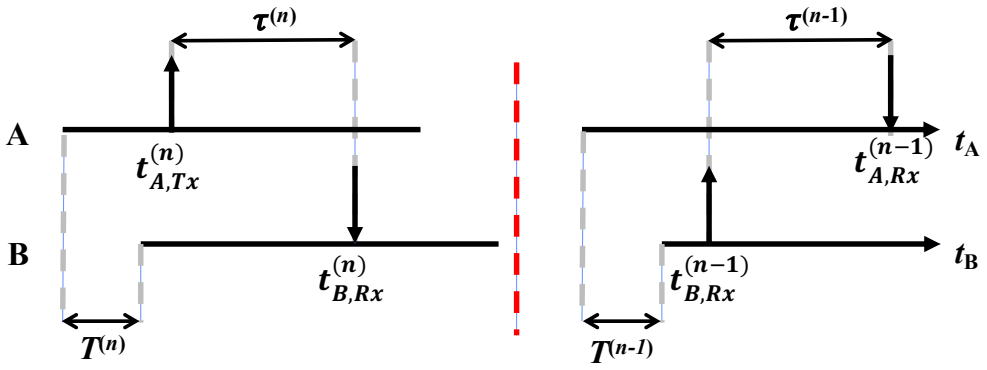


Figure 2.1: *Depiction of two interactions between radios A and B. Radio A first transmits a waveform at a time index by n, which is then received by Radio B. On the next event (indexed by n-1), Radio B transmits a waveform and later is received by Radio A.*

Define a state space including the time-of-flight τ, time offset T, and their first order derivatives $\dot{\tau}$ and $\dot{T}$. Define a set of N time frames indexed by n. Assume one interaction between radios A and B, where A sends an CHP2 frame to B, and then B responds to A by another CHP2 frame. Enumerate the state space as the instantaneous values of each variable at the start of each frame. For one interaction, denote the instantaneous values of the state space variables as $\tau^{(n)}$, $T^{(n)}$, $\dot{\tau}^{(n)}$, and $\dot{T}^{(n)}$. Model the time-of-flight in frame n as

$$\tau^{(n)} = \tau^{(n-1)} + \dot{\tau}^{(n-1)} l^{(n-1)}, \tag{2.2}$$

where $l^{(n-1)}$ is the duration of the previous frame. Model the time offset T as

$$T^{(n)} = T^{(n-1)} + \dot{T}^{(n-1)} l^{(n-1)}. \tag{2.3}$$

The quality of this model diminishes with increasing frame length l and nontrivial higher order derivatives.

2.1.2 Transmission and Reception

Every transmitting or receiving event generates a corresponding timestamp. The transmitter determines transmission timestamps. Reception timestamps are estimated at the receiver. For transmission from A to B, the receive timestamp is modeled as

$$t^{(n)}_{B,Rx} = t^{(n)}_{A,Tx} + \tau^{(n)} - T^{(n)}. \tag{2.4}$$

For a transmission from B to A, the receive timestamp is modeled as

$$t^{(n)}_{A,Rx} = t^{(n)}_{B,Tx} + \tau^{(n)} + T^{(n)}. \tag{2.5}$$

These equations are the basic idea of NTP algorithm for the ToA estimation techniques and the time synchronization.

2.2 Propagation Model

The model of a waveform prorogation through free space between two radios operating with independent clock sources is defined in this section. This waveform undergoes a time delay, phase shift, frequency shift, and channel attenuation. The transmission characteristics for a single waveform (SISO) between synchronized radios are studied. Then extend these models to the CHP2 system in which the radios are not synchronized.

2.2.1 Waveform Synthesis

The CHP2 transmitter synthesizes a baseband waveform $x_0(t)$. This waveform contains both a communications payload and navigation reference sequences. The design process is discussed in greater detail in Chapter (6). This waveform uses time reference t_{Tx} of the current transmitter, and is transmitted at time $t^n_{Tx,Tx}$ on frame index n, such that the baseband transmission $x_{bb}(t)$ is

$$x_{bb}(t_{Tx}) = x_0\left(t_{Tx} - t^n_{Tx,Tx}\right). \tag{2.6}$$

2.2.2 Transmission

Before transmission, the bassband waveform is upconverted to the transmitter carrier frequency. However, the RF carrier signal also has its phase ϕ_{Tx}, which is unknown. Therefore, the passband signal is transmitted from an antenna can be formulated as

$$\begin{aligned} x_{Tx}(t_{Tx}) &= x_{bb}(t_{Tx})e^{j2\pi\left(\phi_{Tx}+f_{\{cr,ac,Tx\}}t_{Tx}\right)} && (2.7)\\ &= x_0\left(t_{Tx} - t^n_{Tx,Tx}\right)e^{j2\pi\left(\phi_{Tx}+f_{\{cr,ac,Tx\}}t_{Tx}\right)}. && (2.8) \end{aligned}$$

where $f_{\{cr,ac,Tx\}}$ is the actual carrier frequency. This $f_{\{cr,ac,Tx\}}$ can be offset from the nominal carrier frequency value because of the imperfection on the clock source,

for example, the 40MHz OCXO source on CHP2 system has a 10ppb temperature stability. That leads to about 9Hz uncertainty on the 915MHz carrier frequency.

2.2.3 Propagation

As the waveform propagates through the wireless channel between the potential moving platforms, it undergoes a Doppler frequency shift, complex channel attenuation, and propagation time delay. The Doppler frequency shift will be introduced according to Equation (2.9).

$$f_{\{cr,dop,Tx\}} = \left(1 + \frac{v}{c}\right) f_{\{cr,ac,Tx\}}. \tag{2.9}$$

The complex attenuation a induced by the channel is modeled as

$$a = |a|e^{j2\pi\phi_a}, |a| = \sqrt{\frac{G_{Tx}G_{Rx}\lambda^2}{(4\pi d^2)}}; \tag{2.10}$$

where G_{Tx} and G_{Rx} are the transmitter and receiver gains, λ is the waveform wavelength, d is the distance between the platforms. For simple line-of-sight channels, assume that $\phi_a \approx 0$. τ represents the time of propagation such that the waveform arrives upon the receiver at time $t^n_{Tx,Tx} - \tau$ using the transmitter timer as the reference:

$$z_{pb}(t_{Tx}) = |a|e^{j2\pi\phi_a} x_{Tx}(t_{Tx} - \tau) \tag{2.11}$$

$$= |a| x_0 \left(t_{Tx} - t^n_{Tx,Tx} - \tau\right) e^{j2\pi\left(\phi_{Tx} + f_{\{cr,dop,Tx\}}(t_{Tx}-\tau)\right)} \tag{2.12}$$

2.2.4 Reception

The received signal referenced by the CHP2 receiver clock is modeled as

$$z_{pb}(t_{Rx}) = |a| x_0 \left(t_{Rx} - t^n_{Tx,Tx} - \tau \pm T\right) e^{j2\pi\left(\phi_{Tx} + f_{\{cr,dop,Tx\}}(t_{Rx}-\tau\pm T)\right)} + n(t_{Rx}), \tag{2.13}$$

where the sign is determined by which platform is receiving according to (2.1). This

signal arrives on a receive antenna. Then a down-conversion move the signal to baseband. Usually, the RF carrier for down-conversion has an unknown phase ϕ_{Rx}, thus the received signal is modeled as

$$z_{bb}(t_{Rx}) = |a|x_0\,(\sim)\,e^{j2\pi\left(\phi_{Tx}-\phi_{Rx}+f_{\{cr,dop,Tx\}}(t_{Rx}-\tau\pm T)-f_{\{cr,ac,Rx\}}t_{Rx}\right)} + n(t_{Rx}), \qquad (2.14)$$

Combine the phase terms into a single nuisance parameter $\tilde{\phi} = \phi_{Tx} + \phi_a - \phi_{Rx}$, such that (2.14) may be written as

$$z_{bb}(t_{Rx}) = |a|x_0\,(\sim)\,e^{j2\pi\left(\tilde{\phi}+f_{\{cr,dop,Tx\}}(t_{Rx}-\tau\pm T)-f_{\{cr,ac,Rx\}}t_{Rx}\right)} + n(t_{Rx}), \qquad (2.15)$$

where n is assumed to be circularly symmetric white Gaussian noise with zero mean and variance σ^2.

Chapter 3

TIME OF ARRIVAL ESTIMATION[1]

The essential of the CHP2 system is to estimate the time-of-arrival of the received CHP2 waveform (defined in Equation (2.15)). Based on the previous work [1, 2], we conclude three steps to achieve high accuracy reception timestamps. First, a coarse estimation of ToA can be estimated by optimizing an incoherent cost function that compares the received signal with a known reference waveform at different delay hypotheses. Then, the estimation is refined by cross-correlating with an over-sampled reference waveform. Finally, the estimation is further improved by incorporating the interpolation.

3.1 Estimation Preliminaries

Time-of-arrival is the time when a signal arrives upon a receiving antenna. In this report. ToA is equivalent to the received timestamp $t^{(\cdot)}_{(\cdot),Rx}$ estimated by the receiver. Traditionally, this waveform is digitally sampled by the receiver at the Nyquist sampling rate. Then a maximum-likelihood estimator, which is a matched filter, that compares the received signal with a known reference waveform at different hypotheses to find the ToA timestamps. Both basic estimation techniques and Nyquist sampling limit the ToA estimation resolution. More advanced techniques may surpass this drawback by oversampling the received and reference waveforms. And then estimate-ToA in this oversampled space. Define the following sampling frequencies:

[1] Andrew Herschfelt contributes to the fine ToA estimation mathematic model.

Table 3.1: *Estimation Variable Definitions*

Label	Description	Equation
$f_{s,c}$	Nyquist Sampling Frequency	
$f_{s,f}$	Raw IQ Data Sampling Frequency	
$f_{s,est}$	Estimator Sampling Frequency	
ρ_{sps}	Samples per Symbol	$\frac{f_{s,f}}{f_{s,c}}$
ρ_{est}	Estimator Samples per Symbol	$\frac{f_{s,est}}{f_{s,c}}$

3.2 Coarse ToA Estimation

The coarse ToA estimator maximizes an objective function that is sampled at either the Nyquist sampling rate ($f_{s,c}$), or some small multiple thereof ($f_{s,f}$). Consider the incoherent objective function [27, 28]

$$g(\tau') = \left| \int dt \; z_{bb}(t+\tau') x_0^*(t) \right|^2 \tag{3.1}$$

This is a match filter that cross-correlates the received baseband signal $z_{bb}(t)$ with the known signal $x_0(t)$ at different delay hypotheses τ'. By inspection of (2.15), this objective function is maximized for $\tau' = t_{Tx,Tx}^{(\cdot)} + \tau \mp T$, thus it indicates that the ML ToA estimate is simply

$$\hat{\tau}' = \arg\max_{\tau'} g(\tau'). \tag{3.2}$$

The receiver data processing is fully digitally, thus, the signal is represented by the discrete form such that

$$\mathbf{z}_{bb}[m] = z_{bb}\left(m f_{s,f}^{-1}\right) \;\; , \; m \in [0, 1, ..., M-1], \tag{3.3}$$

where m indexes the samples in $\mathbf{z}_{bb}$, M is the total number of collected samples, and $f_{s,f}^{-1}$ is the sampling period. Assume the reference waveform is N samples length, then the receiver approximates the objective function (3.1) as

$$\mathbf{g}(k) = \left| \sum_{m=0}^{N-1} \mathbf{z}_{bb}[m+k]\mathbf{x}_0^*[m] \right|^2 \ , \ k \in [0, 1, , ..., K-1]. \tag{3.4}$$

The coarse ToA estimator is thus

$$\hat{\tau}_c' = \hat{k} f_{s,f}^{-1} \ , \hat{k} = \arg\max_k \mathbf{g}(k). \tag{3.5}$$

This estimator is limited to the test points defined by the sampling frequency $f_{s,f}$. In general, the true value will not lie on this sampling lattice, so the accuracy is limited to the resolution between test points.

3.3 Fine ToA Estimation

The resolution of the coarse ToA estimator defined in (3.5) may be improved by performing the maximization at a higher sampling frequency, at the cost of increased computational complexity based on our previous work[1]. By upsampling $\mathbf{z}_{bb}$ and $\mathbf{x}_0$ to a higher frequency $f_{s,est}$, the distance between adjacent test points is reduced, and the resolution of the maximization is increased. This approach has the following limitations:

- **Computational Complexity**[1]: Increasing the sampling factor by a factor of ρ increases both the number of test points and the number of samples in each wavfrom by ρ, resulting in a ρ^2 multiplicative increase in the number of complex multiplications needed to evaluate $\hat{\tau}_c'$. This expansion may be mitigated by reducing the range of test points or iteratively refining the search space, but will still suffer from a massive increase in computation time. We develop a structure optimization for this arithmetic in Section (8.3.3).

- **Imperfect Upsampling**[1]: The upsampling process is imperfect, so the upsampled versions of $\mathbf{z}_{bb}$ and $\mathbf{x}_0$ are only approximations. This can introduce bias into the ToA estimator, and fundamentally limit the accuracy despite further increases in the sampling rate. Furthermore, upsampling is a computationally expensive operation.

A bank of shifted versions of the reference waveform $\mathbf{x}_0$ at a specific sampling frequency $f_{s,est}$ is designed. Instead of upsampling the reference waveform and computing the shifts in real time after each reception, the receiver instead multiplies the received signal by the correlation bank to compute the objective function.

The objective of this correlator bank is to allow the receiver to test delay hypotheses that lie on a fine sampling lattice ($f_{s,est}$), but only perform correlation at the coarse sampling frequency ($f_{s,f}$). Consider a delay hypothesis $\bar{k}$ and a range of hypotheses around it, such that $k \in [\bar{k} - \delta, \bar{k} - \delta + 1, ..., \bar{k} + \delta]$, all of which lie on the fine sampling lattice defined by $f_{s,est}$. Upsample $\mathbf{x}_0$ to $f_{s,est}$. Define the correlator

bank $\mathbf{X}_0$ as

$$\mathbf{X}_0 = \begin{bmatrix} \text{—} & \mathbf{x}_0[m+\delta] & \text{—} \\ \text{—} & \mathbf{x}_0[m+\delta-1] & \text{—} \\ & \vdots & \\ \text{—} & \mathbf{x}_0[m+1] & \text{—} \\ \text{—} & \mathbf{x}_0[m] & \text{—} \\ \text{—} & \mathbf{x}_0[m-1] & \text{—} \\ & \vdots & \\ \text{—} & \mathbf{x}_0[m-\delta] & \text{—} \end{bmatrix}$$

$$= \begin{bmatrix} x_0[\delta] & x_0[\delta+1] & \cdots & x_0[2\delta] & \cdots & x_0[N-1] & \cdots & 0 \\ x_0[\delta-1] & x_0[\delta] & \cdots & x_0[2\delta-1] & \cdots & x_0[N-2] & \cdots & 0 \\ x_0[\delta-2] & x_0[\delta-1] & \cdots & x_0[0] & \cdots & x_0[N-3] & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ x_0[0] & x_0[1] & \cdots & x_0[\delta] & \cdots & x_0[N-\delta-1] & \cdots & x_0[N-1] \\ 0 & x_0[0] & \cdots & x_0[\delta-1] & \cdots & x_0[N-\delta-2] & \cdots & x_0[N-2] \\ \vdots & \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & x_0[0] & \cdots & x_0[N-2\delta-1] & \cdots & x_0[N-1-\delta] \end{bmatrix}. \quad (3.6)$$

Each row of this matrix is a shifted version of the reference signal $\mathbf{x}_0$, where each adjacent row is shifted by 1 sample at the upsampled frequency $f_{s,est}$. Each row is then independently downsampled to the processing sampling frequency $f_{s,f}$. The result is a new correlation bank $\mathbf{B}_0$, where the adjacent rows are still separated by 1 sample at $f_{s,est}$, but the signal within a row is at the processing sampling frequency $f_{s,f}$. This allows us to test shifts at the higher sampling frequency but only perform correlation at the lower sampling frequency $f_{s,f}$, which limits the computational complexity of the problem.

The objective function (3.4) can be rewritten as a matrix multiplication with this

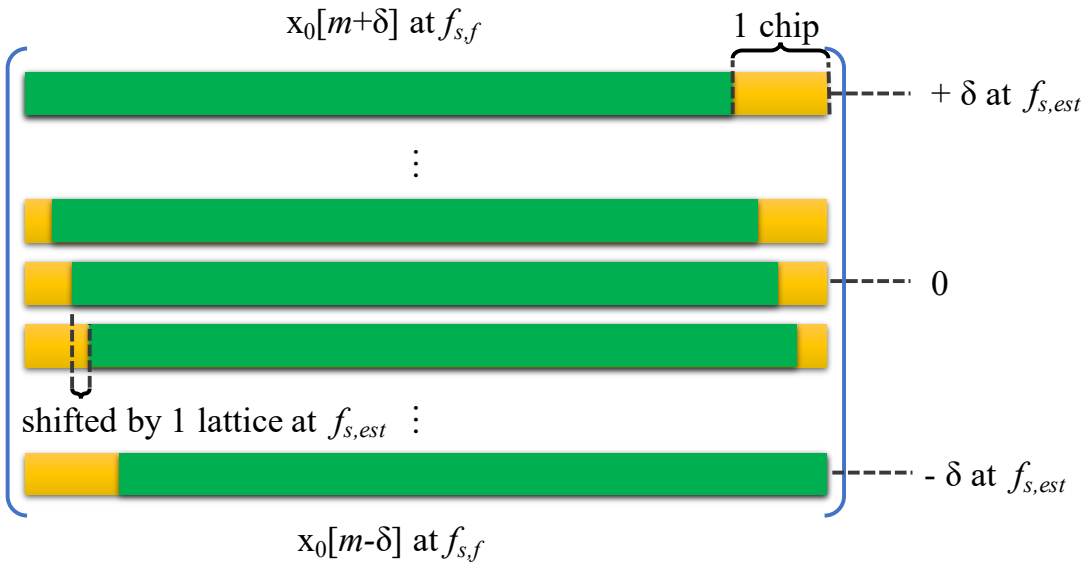


Figure 3.1: *Depiction of the correlator reference bank* $\mathbf{B}_0$*. Adjacent rows are separated by a single sample shift at the oversampled sampling frequency* $f_{s,est}$*. Each signal within a row is downsampled to the processing sampling frequency* $f_{s,f}$*.*

correlator bank. The correlation must first be aligned such that the correlator bank rows correspond to the correct indeces in the objective function. We may choose the central test point $\bar{k}$ at our convenience. The objective function is formulated such that

$$\mathbf{g}'(k) = \left| \sum_{m=0}^{N-1} \mathbf{z}_{bb}[m+\bar{k}]\mathbf{x}_0^*[m-k'] \right|^2, k' \in [-\delta, -\delta+1, ..., \delta]. \tag{3.7}$$

By noting that k is defined as the set of shifts around $\bar{k}$, we can make a change of variables $k' = k - \bar{k} \in [-\delta, -\delta+1, ..., \delta]$.

The limits of summation now align with the expression of the correlator bank in (3.6), so this operation may now be written in terms of a matrix multiplication. By defining $\mathbf{z}_f[m'] = \mathbf{z}_{bb}[m'+\bar{k}]$, i.e. the last N samples in the received sequence $\mathbf{z}_{bb}$ starting at sample $\bar{k}$, the objective function can now be written as

$$\mathbf{g}(k') = \mathbf{z}_f \mathbf{B}_0^\dagger, \tag{3.8}$$

where $\mathbf{B}_0$ is the downsampled version of $\mathbf{X}_0$. Both $\mathbf{z}_f$ and $\mathbf{B}_0$ are sampled at the coarse sampling frequency $f_{s,f}$, but the shifts k' are at the fine sampling frequency $f_{s,est}$. The ToA estimate may then be extracted from $\mathbf{g}(k')$ as the sum of $\bar{k}$ and k' both normalized to seconds, such that

$$\hat{\tau}'_f = \bar{k} f_{s,f}^{-1} + \hat{k}' f_{s,est}^{-1},\ \hat{k}' = \arg\max_{k'} \mathbf{g}[k']. \tag{3.9}$$

This form allows for negative indeces k'. If it is necessary for $\mathbf{g}$ to be indexed by positive integers, then we may apply another change of variables $k'' = k' + \delta \in [0, 1, ..., 2\delta]$, and the estimate is appropriately shifted

$$\hat{\tau}'_f = \bar{k} f_{s,f}^{-1} + \left(\hat{k}'' - \delta\right) f_{s,est}^{-1},\ \hat{k}'' = \arg\max_{k''} \mathbf{g}[k'']. \tag{3.10}$$

Chapter 4

TIME OF FLIGHT ESTIMATION[1]

A couple of time-of-flight estimation algorithms are designed for the CHP2 system. Those algorithms estimate and track the distance between users and synchronizes the distributed clock sources. Most of them are inspired by the Network Timing Protocol (NTP). There are the Hyper-precise Timing Protocol (HTP)[1, 2], Extended Kalman Filters based Time-of-Flight tracking, first-order one-shot tracking, and second-order one-shot tracking. Currently, only NTP, HTP, and EKE based Time-of-Flight estimation are implemented on the application layer of the CHP2 system. The HTP, EKE filter, and one-shot tracking methods are belongs to other persons' intellectual properties and not discussed in this manual. This chapter covers the necessary information about the NTP algorithm.

4.1 NTP Timing Protocol

Assume a scenario of two CHP2 nodes. When Node B receives a message, it records the timestamp of this message arrival ($t_{B,Rx}^{(n)}$) and determines the time event for transmitting the next message ($t_{B,Tx}^{(n+1)}$) to the other Node A. These two timestamps are packed onto the next transmission frame such that Node A has access to these values. After a transmission cycle A $\rightarrow$ B $\rightarrow$ A, Node A estimates the time of flight τ and time offset T for each of the two frames. Index these two frames by n and $n-1$, as depicted in Figure 4.1.

For the transmission A $\rightarrow$ B, the received timestamp is modeled[1] as

$$t_{B,Rx}^{(n-1)} = t_{A,Tx}^{(n-1)} + \tau^{(n-1)} - T^{(n-1)}. \tag{4.1}$$

[1] Andrew Herschfelt and Sharanya Srinivas contribute to the ToF estimation mathematic models.

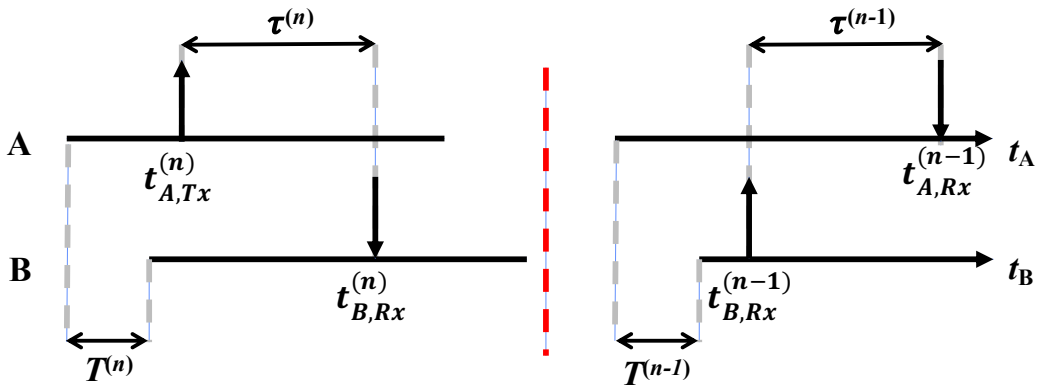


Figure 4.1: *Depiction of transmission cycle from $A \to B \to A$. The variables necessary to execute the acquisition stage of the algorithm are labeled.*

For the transmission B $\to$ A, the received timestamp is modeled as

$$t_{A,Rx}^{(n)} = t_{B,Tx}^{(n)} + \tau^{(n)} + T^{(n)}. \tag{4.2}$$

This stage in the protocol makes the following assumptions:

1. Both Nodes have been registered on the network.

2. Both Nodes have agreed to cooperate and have defined a frame length l.

3. The time offset T does not change significantly from frame $n-1$ to frame n, i.e. $T^{(n-1)} = T^{(n)}$.

4. The time delay τ does not change significantly from frame $n-1$ to frame n, i.e. $\tau^{(n-1)} = \tau^{(n)}$.

When a radio receives a transmission, it decodes the timestamps information embedded in the message. With these timestamps and assumptions 3 and 4, Equations (4.1) and (4.2) become a system of 2 linear equations with 2 unknowns. Each radio solves this system as follows.

Algorithm[1]:

A: $\forall n \in [2, 4, 6, ...]$,

1. Compute $\hat{\gamma}_A^{(n)}$ using:

$$\hat{\gamma}_A^{(n)} = (\hat{t}_{A,Rx}^{(n)} - t_{A,Tx}^{(n-1)}) - (t_{B,Tx}^{(n)} - \hat{t}_{B,Rx}^{(n-1)}) \tag{4.3}$$

2. Estimate $\tau^{(n)}$ and $\tau^{(n-1)}$ using assumption 4 and (4.3):

$$\hat{\tau}^{(n)} = \frac{\hat{\gamma}_A^{(n)}}{2} \tag{4.4}$$

$$\hat{\tau}^{(n-1)} = \frac{\hat{\gamma}_A^{(n)}}{2} \tag{4.5}$$

3. Estimate $T^{(n)}$ and $T^{(n-1)}$ using Equations (4.1) and (4.2):

$$\hat{T}^{(n)} = \hat{t}_{A,Rx}^{(n)} - t_{B,Tx}^{(n)} - \hat{\tau}^{(n)} \tag{4.6}$$

$$\hat{T}^{(n-1)} = t_{A,Tx}^{(n-1)} - \hat{t}_{B,Rx}^{(n-1)} + \hat{\tau}^{(n-1)} \tag{4.7}$$

4. If $n > 2$, track the first order derivatives. Let M be the total number of iterations performed, such that $\hat{l}_A^{(n-2,M)}$ is the frame length estimate of the last iteration of the previous processing cycle. For the current cycle use the nominal value of l_A.

$$\hat{\dot{\tau}}^{(n)} = \frac{\hat{\tau}^{(n)} - \hat{\tau}^{(n-2)}}{l_A^{(n-1)} + \hat{l}_A^{(n-2,M)}} \tag{4.8}$$

$$\hat{\dot{\tau}}^{(n-1)} = \hat{\dot{\tau}}^{(n)} \tag{4.9}$$

$$\hat{\dot{T}}^{(n)} = \frac{\hat{T}^{(n)} - \hat{T}^{(n-2)}}{l_A^{(n-1)} + \hat{l}_A^{(n-2,M)}} \tag{4.10}$$

$$\hat{\dot{T}}^{(n-1)} = \hat{\dot{T}}^{(n)} \tag{4.11}$$

B: $\forall n \in [3, 5, 7, ...]$,

1. Compute $\hat{\gamma}_B^{(n)}$ using:

$$\hat{\gamma}_B^{(n)} = (\hat{t}_{B,Rx}^{(n)} - t_{B,Tx}^{(n-1)}) - (t_{A,Tx}^{(n)} - \hat{t}_{A,Rx}^{(n-1)}) \tag{4.12}$$

2. Estimate $\tau^{(n)}$ and $\tau^{(n-1)}$ using assumption 4 and (4.12):

$$\hat{\tau}^{(n)} = \frac{\hat{\gamma}_B^{(n)}}{2} \tag{4.13}$$

$$\hat{\tau}^{(n-1)} = \frac{\hat{\gamma}_B^{(n)}}{2} \tag{4.14}$$

3. Estimate $T^{(n)}$ and $T^{(n-1)}$ using Equations (4.2) and (4.1):

$$\hat{T}^{(n)} = t_{A,Tx}^{(n)} - \hat{t}_{B,Rx}^{(n)} + \hat{\tau}^{(n)} \tag{4.15}$$

$$\hat{T}^{(n-1)} = \hat{t}_{A,Rx}^{(n-1)} - t_{B,Tx}^{(n-1)} - \hat{\tau}^{(n-1)} \tag{4.16}$$

4. If $n > 3$, track the first order derivatives. Let M be the total number of iterations performed, such that $\hat{l}_B^{(n-2,M)}$ is the frame length estimate of the last iteration of the previous processing cycle. For the current cycle use the nominal value of l_B.

$$\dot{\hat{\tau}}^{(n)} = \frac{\hat{\tau}^{(n)} - \hat{\tau}^{(n-2)}}{l_B^{(n-1)} + \hat{l}_B^{(n-2,M)}} \tag{4.17}$$

$$\dot{\hat{\tau}}^{(n-1)} = \dot{\hat{\tau}}^{(n)} \tag{4.18}$$

$$\dot{\hat{T}}^{(n)} = \frac{\hat{T}^{(n)} - \hat{T}^{(n-2)}}{l_B^{(n-1)} + \hat{l}_B^{(n-2,M)}} \tag{4.19}$$

$$\dot{\hat{T}}^{(n-1)} = \dot{\hat{T}}^{(n)} \tag{4.20}$$

The NTP algorithm is easy to implement, but has the following limitations[1]:

1. τ is assumed to be constant between each frame. But it is not true for the moving platform.

2. T may not be constant between each frame. The practical oscillator has frequency offset from its label specification and phase noise.

3. Transmission timestamps are assumed to be known perfectly. The actual timestamps cannot be precisely acquired without calibration.

4. The frame length l may not be known and constant because of significant drift or misalignment on clock source.

5. Estimates of the state space occur only every other frame.

These limitations are addressed in the tracking stage of the CHP2 timing protocol not including in this report.

Chapter 5

POSITION AND ATTITUDE ESTIMATION[1]

So far, the time-of-flight or range estimation techniques of the CHP2 system are discussed in previous chapters. In this chapter, the method of extending MIMO range estimates to the actual geometry location of the remote CHP2 node is introduced. For demonstration, consider two CHP2 nodes communicating over a 4 $\times$4 MIMO channel remotely such that one is the ground station, and the other one is the remote node. It is presumed to generate 16 ToF/range estimates between each pair of antennae. The ground station should estimate the geometry location of the remote CHP2 node based on these MIMO range estimates and its initial location. This chapter focuses on determining the position and orientation of each CHP2 node.

5.1 CHP2 Geometry Calibration

CHP2 system measures the physical propagation delays between each antenna pair of two CHP2 nodes. Without any calibration, the CHP2 processing chain and RF-related hardware adds extra delay to the total propagation delay. Therefore, before any range geometry conversion, the estimated raw range values have to be correctly calibrated to the exact range estimates by removing that extra system delay.

In a case of two CHP2 nodes, assume the CHP2 antenna geometry configurations are known to each node initially. It is so easy to compute the actual range values between two nodes. Subtract this true range by the raw estimated range to get an offset when both nodes are on their initial positions. Later then, raw range estimates subtract this offset to get the corrected range estimated, even though both nodes are

[1]This is a joint work with Andrew Herschfelt and Yang Li.

not on their initial positions.

There are CHP2 Node A and B. Node A is the ground station, and Node B is on a flight platform. Each of them has four antennas. Initially, both nodes are placed close together on the ground statically. An external measurement instrument can measure the antenna geometry. Then the antenna geometry information is loaded onto both nodes. This information is defined as the exact coordinates of all physical antennas in a three-dimensional Cartesian coordinate system (x, y, z). For the Antenna pair A1 and B1, the initial exact range measurements are defined as follows,

$$\tau_{\text{true\_init\_1}} = \sqrt{(x_{\text{A1}} - x_{\text{B1}})^2 + (y_{\text{A1}} - y_{\text{B1}})^2 + (z_{\text{A1}} - z_{\text{B1}})^2}. \tag{5.1}$$

Then the CHP2 system can generate an estimated range as $\tau_{\text{raw\_init\_1}}$ from the CHP2 timing algorithm for antenna pair A1 and B1. In this step, a range offset can be calculated by subtracting the true and raw value.

$$\tau_{\text{offset\_1}} = \tau_{\text{raw\_init\_1}} - \tau_{\text{true\_init\_1}}. \tag{5.2}$$

This offset value estimation procedure is repeated a couple of times to get an average value to minimize the error. Later then, all the raw range estimates can be corrected by this offset value.

$$\tau_{\text{correct\_1}} = \tau_{\text{raw\_1}} - \tau_{\text{offset\_1}}. \tag{5.3}$$

Repeat this procedure for all 16 antenna pairs of Node A and B to compute the system offset and correct the range estimates. Once this initial geometry calibration finishes, the remote Node B can move with the flight platform for the long-range measurement.

This calibration solution relies on several prerequisites.

1. A three-dimensional Cartesian coordinate system is required. As long as it is

consistent, the origin point can be chosen anywhere. This coordinate system represents the geometry related information

2. The external measurement instrument is precise, such as a laser tachymeter. It measures the exact coordinates of antennas on both Node A and B during the calibration stage.

3. The antenna geometry configuration is consistent. Once the configuration is changed, the range correction is no longer accurate.

4. The CHP2 initial raw range estimates are stable and accurate. During the calibration stage, both Node A and Node B have to maintain a very well line of sight communication and enough environment clearance to eliminate any multi-path disturbance.

5.2 Position Estimation

Let us consider a setup portrayed in figure 5.1, where a ground node (G) and an aerial node (P) are communicating via four antennae each, indexed by $j\ \forall j \in [1, 4]$. The ground node antennae G_j are tethered to the ground at known locations (x_j, y_j, z_j). The CHP2 algorithm results in 16 delay estimates, which can be transformed to distance estimates via $d_{i,j} = c\,\tau_{i,j}$, where $c = 3 \times 10^8 m/s$ is the speed of light. This implies, to locate each of the aerial antennae P_j we have access to four distances $d_{1,j}$, $d_{2,j}$, $d_{3,j}$ and $d_{4,j}$, as shown in figure 5.1. We shall now explore different multilateration techniques to locate each of the $P_j \forall j \in [1, 4]$.

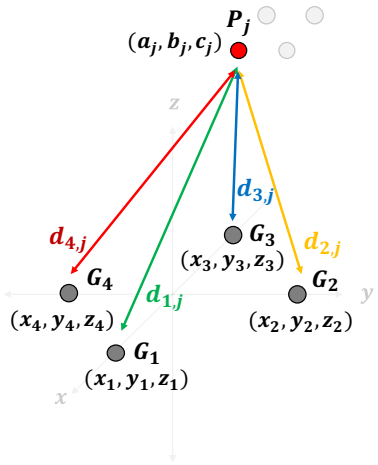


Figure 5.1: *Setup of a ground node (G) and an aerial node (P) communicating over a 4 ×4 MIMO channel. Four distances $d_{1,j}$, $d_{2,j}$, $d_{3,j}$ and $d_{4,j}$ are used by ground node to locate the aerial antennae P_j.*

5.2.1 Multilateration Estimators

In this section we explore 3 multilateration techniques that solve a least-squares estimation problem - 1) Ordinary Least Squares Estimators (OLS), 2) Iteratively Re-weighted Least Squares Estimators (IRLS) and the 3) Non-Linear Least Squares Estimator (NLLS), refer to [55]. We shall discuss the pros and cons of each of these methods.

The problem at hand can be formulated as a linear set of equations, and Ordinary Least Squares Estimator provides a least-squares solution. Often due to shielding, all the measurements are not of equal significance. The OLS solution does not consider this. It is the limitation that the Iteratively Re-weighted Least Squares Estimator is trying to overcome. It constantly reweighs the measurements to adaptively and come up with the best estimate. IRLS solution is still solving a linear problem and hence restricted to this regime. The Non-Linear Least Squares Estimation (NLLS) resolves this issue by solving an optimization problem every instance a measurement comes in.

5.2.2 Iterative Searching

During the CHP2 development phase, we experiment with these three multilateration methods. However, none of these algorithms stably outputs correct coordinates. They perform well on the scenario where ranges estimates have a significant value of difference, such as a traditional large scale triangulation. The CHP2 system places its 4×4 MIMO antennas very close together and can measure the range accurate up to centimeter-level. However, this configuration leads to all tiny differences on all 16 range estimates and causes a sizeable numerical error on the above the multilateration techniques.

Therefore, for the sake of demonstration and simplifying implementation of the CHP2 system, an iterative searching method is realized. Since the initial geometry coordinates of the ground station and remote node are known. During updating the range estimates, we can iteratively search the space around the aerial node. Choose the most likely location, which gives the minimum difference from the ranges estimates, as the estimated coordinate. For further simplification, a cylindrical coordinate system is established and later then converted back to three-dimensional Cartesian coordinates. Specifically, there are searching in three aspects: azimuth angle, range, and elevation.

Let K be the number of CHP2 frame, M be the total iteration times.

1. Acquire ground station and remote node initial location coordinates.
2. Define number of test point N and initial resolution for azimuth, range and elevation.
3. For $k = 1 : K$ CHP2 frames...
   (a) For $m = 1 : M$ iterations...

i. Set the test points' azimuth range.

ii. Evaluate the test points around the remote node location. Compute a set of ranges that are from the test points to the grounds station antennas.

iii. Find the azimuth that gives the minimum difference between computed ranges from the test point to the ground station and the estimated ranges.

iv. Update the test points' azimuth range. Update the estimated remote node azimuth.

v. Set the test points' distance range (from test point to ground station)

vi. Evaluate the test points around the remote node location. Compute a set of ranges that are from the test points to the grounds station antennas.

vii. Find maximum likelihood distance that gives the minimum difference between computed ranges from the test point to the ground station and the estimated ranges.

viii. Update the test points' distance range. Update the estimated remote node distance.

ix. Set the test points' elevation range (from test point to ground station)

x. Evaluate the test points around the remote node location. Compute a set of ranges that are from the test points to the grounds station antennas.

xi. Find maximum likelihood elevation that gives the minimum difference between computed ranges from the test point to the ground station and the estimated ranges.

xii. Update the test points' elevation range. Update the estimated remote node elevation.

xiii. End for loop.

(b) Update estimated remote node azimuth, distance, and elevation.

(c) Convert the cylindrical coordinate system onto the three-dimensional Cartesian system.

4. End for loop.

The number of iteration in actual system implementation is 5 times. It gives a good estimation and less computation loading.

Chapter 6

CHP2 JOINT POSITIONING-COMMUNICATIONS WAVEFORM DESIGN[1]

The CHP2 frame mainly consists of communication waveform and positioning waveform. The communication portion contains data payload of time stamps and other necessary information. The positioning waveform is feed to the high precision time-of-arrival estimator. Based on the CHP2 frame structure, a data link layer and a duplex mode are proposed to adapt this specified system. These configurations guarantee the reliability of communication while maintaining high precision position estimation between remote CHP2 nodes.

6.1 Data Link Layer Overview

In the current CHP2 data link layer configuration, the transmission and reception are alternating between two nodes. Each transmission of a node happens exactly on a continuous cycling period. The other node receives the CHP2 frame, then processes the payload and estimates time-of-arrival simultaneously. Then this receiver node continues to prepare the frame for its next transmission as Figure 6.1. The CHP2 time-of-flight algorithm requires a bidirectional communication to preciously estimate the range between two nodes. Generally, the length of a valid frame is much shorter than the constant cycling period. It can guarantee the feasibility of extending the system onto multi-user or multi-node network topology. More frames from different nodes can be inserted into an empty time slot sequentially.

The CHP2 system implements a 4×4 MIMO CHP2. The MIMO scheme can provide 16 different ranges of measurement between two nodes. The tiny differences

[1]This is a joint work with Andrew Herschfelt and Shunyao Wu.

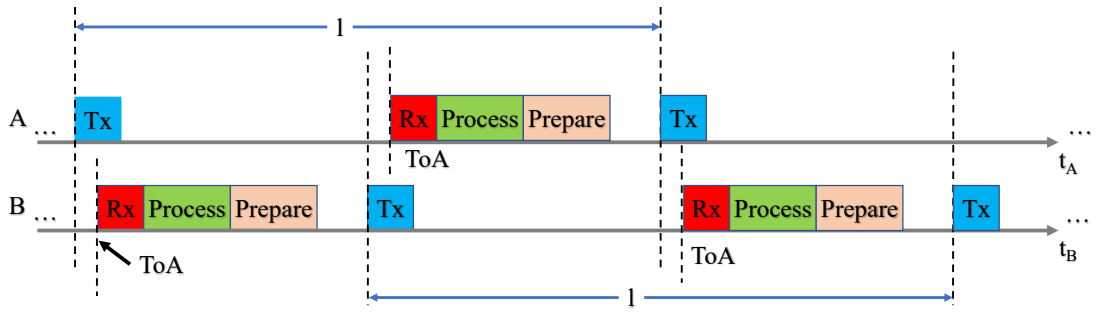


Figure 6.1: *Depiction of data link layer for two users. Users alternate transmitting and receiving in each frame. The receiver estimates the ToA timestamp and schedules the transmit timestamp, both of which are packaged into the next payload and transmitted on the next frame.*

among these measurements contribute to the estimation of geolocation and orientation of the remote target node. Therefore, a CHP2 frame utilizes all four transmission antenna (uplink channel) as Figure 6.2.

A CHP2 frame contains a communication segment and a positioning segment. The communication segment is only transmitted on uplink channel one, then received and processed on downlink channel one as a SISO case. The positioning segment is deployed to all four MIMO channels. Based on the actual physical CHP2 of the over-the-air demonstration, the SISO communication link is still very robust and easy for implementation. In the chapter of baseband processing implementation, the details will be discussed. Extending the communication section onto a MIMO case is feasible but not necessary and may increase the implementation workload dramatically.

The method of deploying the positioning segment uses a time-division duplex (TDD) scheme. The TDD strategy places the same waveform on each uplink channel but transmitting in different sequential time slots, as depicted in Figure 6.2. For a given length of a time slot, TDD allows short waveform, which drops integrated SNR (ISNR) comparing to a code-division duplex (CDD) scheme. However, the TDD

scheme is not suffering from channel crossing interference and has less ISNR.

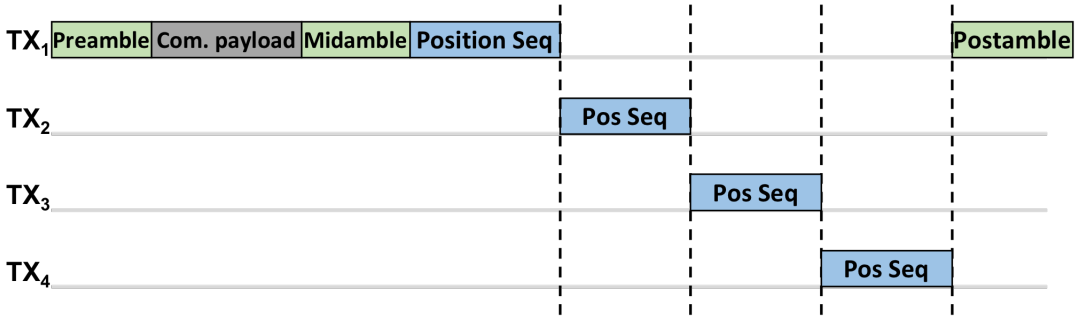


Figure 6.2: *Time-division duplex strategy for assigning the positioning sequences. The same sequence is placed on each transmit channel but are transmitted sequentially. This makes the waveform design easier but reduces the integration time.*

6.2 Waveform Structure

The CHP2 frame structure is defined as Figure 6.3. It includes communication waveform and positioning waveform. Precisely, the commutation section consists of a preamble waveform, a data payload, and a midamble waveform. The positioning section consists of multiple positioning waveforms, which are transmitted in a TDD scheme over different uplink channels, as Figure 6.2, and a postamble waveform for fine frequency offset estimation. The preamble and midamble waveform are used for coarse frequency offset estimation. The empty slots are reserved for waveform pulse shaping residue. The pulse-shaping filter helps to reduce inter-symbol interference and limit the signal bandwidth within the design requirement. The design details of each component are discussed within the subsection of this chapter.

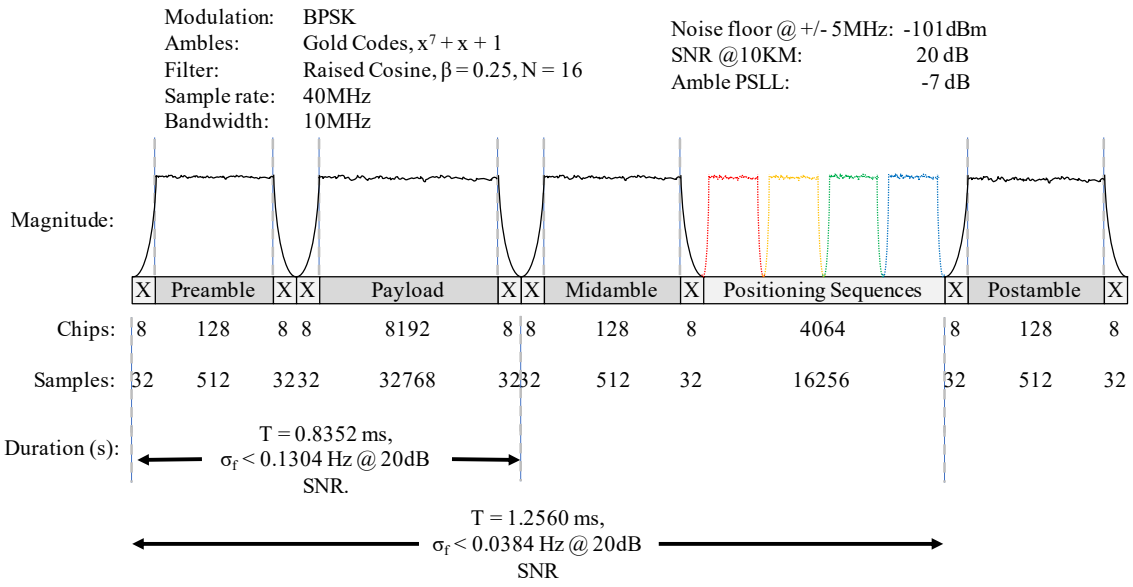


Figure 6.3: *Depiction of the individual components on a joint waveform. The length of each component is defined in the number of critical samples or chips. The communications component consists of a preamble, a midamble, and a payload. The positioning component consists of a reserved segment that is filled based on which positioning strategy is selected (See Figures 6.2) and a postamble waveform. Each component is separated with an empty slot to mitigate multi-path and inter-symbol interference.*

6.3 Communications Waveform Design

The communication waveform in a CHP2 frame carries the time stamps and other necessary information for completing the time-of-flight estimation algorithm. Data throughput is not the top priority, but link reliability is. In the demonstration setup, there are a ground station and a drone segment that takes off nearby and flies over a maximum range of about 10 kilometers. Also, this UAV (unmanned aerial vehicle) scenario limits transmission signal power (about 1 watt maximum on each antenna). Thus, the CHP2 communication waveform design is targeting link reliability, long-range communication, and low power consumption requirement.

6.3.1 Payload Compression

The communications payload consists of the timestamp to implement the synchronization algorithm and the other necessary information. A payload contains 16 transmit timestamps and 16 receive timestamps, each one corresponding to each of the 16 links in this 4×4 MIMO system. All 32 raw timestamps consume a large number of bits in communication payload if there is no data compression. However, those timestamps in the CHP2 system are naturally high related and can be compressed to save bits in the payload.

The positioning waveforms are transmitted in a TDD mode and sequentially in time. Although those waveforms are sent on different time slots, they can be treated as transmitting at the beginning of a CHP2 frame simultaneously. Sine the time length from the beginning of a frame to each corresponding positioning waveform is fixed. As long as this assumption is kept in the receiving chain, there is no impact on the current algorithm. In a word, all 16 transmission timestamps are treated as a single one, which is the exact timing of the preamble in a CHP2 frame.

The reception timestamps of positioning waveforms should be recorded sequentially in different fixed time slots. However, they can be treated such that they are all received at the same coarse ToA with a slight time difference due to the antenna geometry. Thus, the receiving timestamps can be compressed by reporting a single coarse ToA of the preamble, then reporting the finer timing deltas of all 16 positioning waveforms. This idea also matches the CHP2 ToA estimation architecture design, where ToA estimation is separated as ToA coarse estimation and ToA fine estimation.

In such a way, the actual CHP2 communication payload is reduced to contain a transmitting timestamp, a coarse receiving timestamp, 16 delta receiving timestamps, and other necessary information. The transmitting timestamp only contains an integer component because each transmission is only assigned on a designated time periodically. This is determined by the primary system timer, which is driven by the primary clock source (40MHz OCXO source). So this transmitting timestamp precisely maps to the system timer value. The minimum resolution of the primary timer is the period of the system clock ($0.25\mu s$). The time value of the transmitting timestamp is a common multiple of the system's primary clock period. A coarse receiving timestamp reflects the receiver's timer content when a CHP2 frame is received. However, its minimum resolution is limited by the system's primary clock frequency. The 16 delta receiving timestamps represent the differences from the exact receiving timestamps to this coarse receiving timestamp. These 16 deltas are determined by the fine ToA estimation of digital logic, not the system timer. Combining the coarse receiving timestamp and delta receiving timestamp can generate the full receiving timestamps of all 16 MIMO channels. Since a delta value is very tiny comparing to a full receiving timestamp, therefore, fewer bits are used to represent the full timestamp.

1. Integer Bits: This is the number of bits used to report the integer component of the full timestamps. The number of integer bits will determine the maximum amount of time that the clock counter can report. For n integer bits, the clock timer can count up to $(2^n - 1)$ before resetting. At a clock rate of f_{clock}, the maximum amount of time that this clock can run before resetting is $(2^n - 1) f_{clock}^{-1}$. This determines the length of time for which the system can run before resetting.

2. Delta Bits: This is the number of bits used to report the delta between the coarse receive timestamp and all the 16 full receiving timestamps. The antennas on the CHP2 system are close together. Thus the prorogation delay differences are small compared to the maximum value of the clock counter. As such, fewer bits can be used to report the deltas. The number of delta bits will determine the maximum separation between antennas that can be supported by this compression scheme. For p delta bits, the maximum separation between antennas that can be supported is $(2^p - 1) f_{subclock}^{-1} c$, where c is the speed of light and $f_{subclock}$ is the equivalent clock rate of the subsample (the fraction of a regular sample) and is determined by the hardware architecture.

3. Decimal Bits: This is the number of bits used to report the decimal components of the full timestamps. The number of decimal bits will determine the subsample precision of the timestamps. For m decimal bits, the subsample precision of the timestamps is 2^{-m} per subsample clock cycle, which can be converted to seconds by multiplying the clock period $2^{-m} f_{subclock}^{-1}$. This defines a maximum performance threshold on timing precision.

Given the system requirements, the following values for these three parameters are chosen to be $n = 42$, $m = 10$, $p = 9$, $f_{clock} = 40MHz$ and $f_{subclock} = 2GHz$. The resulting clock duration, subsample precision, and maximum supported separation

are:

$$\text{Duration} = (2^n - 1) f_{clock}^{-1} = 30.5420\text{hrs}, \tag{6.1}$$

$$\text{Accuracy} = 2^{-m} f_{subclock}^{-1} = 0.00049ns = 0.1465\text{mm}, \tag{6.2}$$

$$\text{Separation} = (2^p - 1) f_{subclock}^{-1} c = 0.2555\mu s = 76.65\text{m} \tag{6.3}$$

This theoretical result of the data format is only the upper limit of the system. The actual performance is determined by the hardware architecture and time-of-flight estimation algorithm.

The structure of timestamps for these parameters is depicted in Figure 6.4. An additional sign bit is included for each delta. The 64 additional bits are reserved for the HW debugging information, and 8 bits are reserved for the power control word. The final assembled packet is depicted in Figure 6.5.

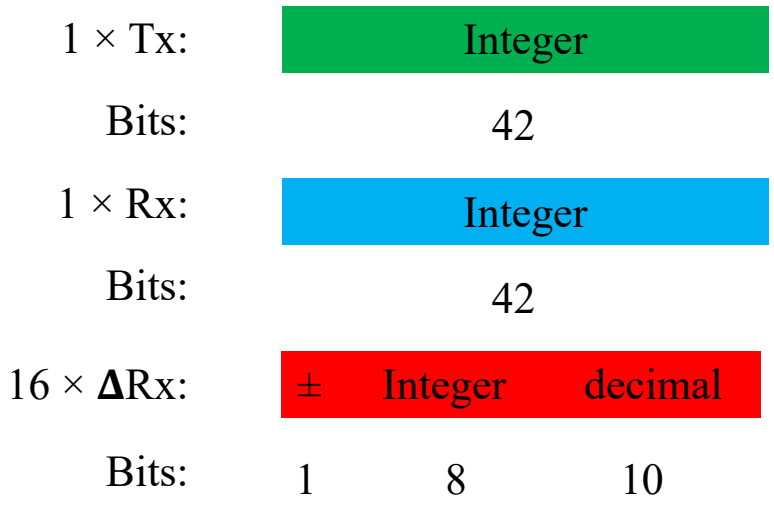


Figure 6.4: *Depiction of the timestamp compression.*

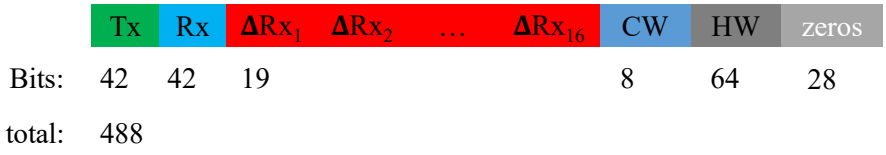


Figure 6.5: *Depiction of the assembled compressed payload.*

6.3.2 *Payload Assembling*

After assembling the data bits, this raw payload has to pass through CRC checksum, scrambler, error-control encoder, modulator, code-spreading, and pulse-shape filter. All timestamps and extra information are compressed onto 488 bits long. First of all, the payload bits are appended with extra empty 24 bits suffix for storing CRC checksum value and preventing tail-biting on the convolution code. Secondly, the appended bits are passed through a scrambler to add randomness. Thirdly, the payload is encoded by a 1/2 rate convolution code. The length of encoded bits is extended to 1024 bits. Fourthly, the encoded bits are modulated by binary phase shift key (BPSK) and upsampled to four samples per symbol. Fifthly, the modulated symbols are then code-spread by a Gold code of length 8 to generate 32768 samples. Finally, the spread samples go through an FIR filter for pulse shaping to limit inter-symbol interference and constrain the bandwidth within system requirements.

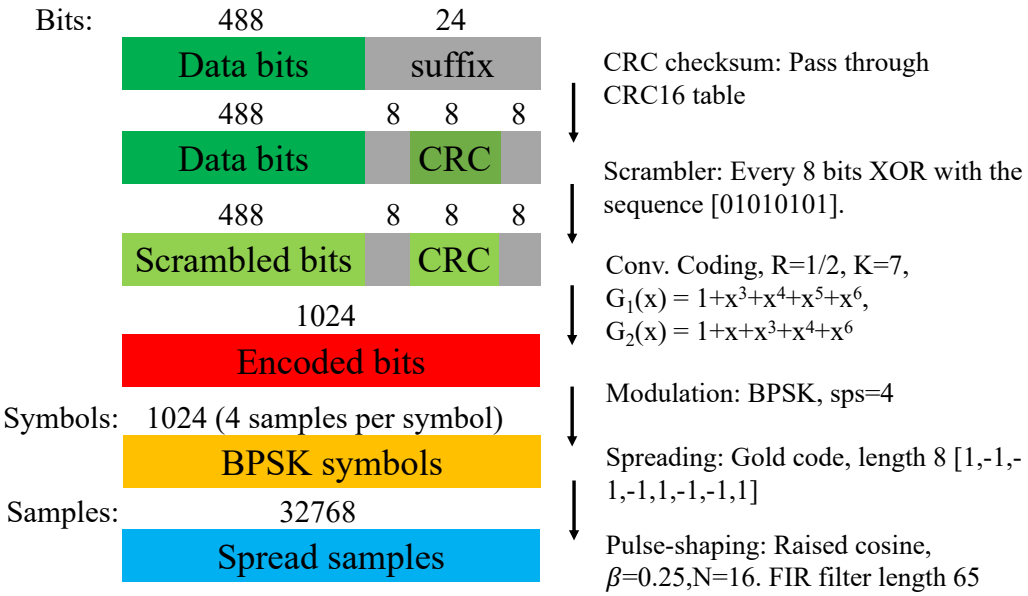


Figure 6.6: *Depiction of assembling a communications payload.*

6.3.3 Link Budget

Based on Free Space Path Loss Equation and system design parameters, the path loss at 10km is:

$$\begin{aligned} FSPL &= 20\log_{10}(d) + 20\log_{10}(f) + 20\log_{10}(\frac{4\pi}{c}) - G_{Tx} - G_{Rx} \\ &= 20\log_{10}(10\text{km}) + 20\log_{10}(1\text{GHz}) + 20\log_{10}(\frac{4\pi}{c}) - 1\text{dB} - 1\text{dB} \\ &= 110.44\text{dB}. \end{aligned} \tag{6.4}$$

The signal bandwidth is 10MHz. The required transmitting power is 30dBm per antenna. The receiver noise figure is about 2.5 dB [56, 57] around 1GHz carrier of the chosen SDR platform. Then the noise floor at receiver is about:

$$\begin{aligned} \text{Noise floor} &= 10\log_{10}(\kappa \times T_0 \times 1000) + \text{NF} + 10\log_{10}(\text{BW}) \\ &= 10\log_{10}(1.38 \times 10^{-23} \times 290 \times 10^3) + 3 + 10\log_{10}(10\text{MHz}) \\ &= -101\text{dBm}. \end{aligned} \tag{6.5}$$

The receiver SNR at 10 km distance is about:

$$\text{SNR} = 30\text{dBm} - 110.44\text{dB} - (-101\text{dBm}) \approx 20\text{dB}. \tag{6.6}$$

For a given payload of 512 effective information bits, it is sent out over a 32768 samples ($819.2\mu s$) at 40MHz per frame. The required SNR to close link is:

$$\begin{aligned} C &= BW\log_2(1 + \text{SNR}), \\ \text{SNR} &= 2^{C/BW} - 1 \\ \text{SNR} &= 2^{625\text{kbps}/10\text{MHz}} - 1 \\ \text{SNR} &= -13.5385\text{dB}. \end{aligned} \tag{6.7}$$

Therefore, the communication link can be properly closed far more than 10km range. The calculated maximum range limit for closing communication link is about 500km.

6.3.4 Pilot Sequences Design

The preamble is used for frame acquisition and the coarse time-of-arrival estimation. Both the preamble and midamble are used for coarse frequency offset estimation. It guarantees communication payload decoding properly. The extra postamble is used for fine frequency offset estimation. To prevent frame detection (cross-correlator) misfires, we design the preamble and midamble with low cross-correlation properties. We draw sequences from a Gold code of length 128 and modulate them using BPSK.

6.3.5 Frequency Estimation

The communications processing chain performs a coarse frequency offset estimation before decoding the data payload. The coarse frequency offset is estimated by computing the phase difference between the preamble and midamble. This estimator produces maximum unambiguous frequency offset estimates proportional to the inverse of time between the test points (a small phase shift between test points looks the same as a small phase shift plus a full phase rotation). These ambiguities occur at the true value plus or minus integer multiples of the π/T where T is the time length between the test points.

$$\begin{aligned} \Delta f_{\text{coarse}} &= \frac{\phi_{\text{Mid}} - \phi_{\text{Pre}}}{2\pi T} \\ &= \frac{(\phi_{\text{Mid}} - \phi_{\text{Pre}})\, f_s}{2\pi l} \end{aligned} \tag{6.8}$$

$$|\phi_{\text{Mid}} - \phi_{\text{Pre}}| < \pi$$

$$\begin{aligned} |\Delta f_{\text{coarse}}| &< 0.5 \times 40\text{MHz}/33408\text{samples} \\ &= 598.66\text{Hz} \end{aligned}$$

$$\begin{aligned}
\Delta f_{\text{fine}} &= \frac{\phi_{\text{Post}} - \phi_{\text{Pre}} + n}{2\pi T} \\
&= \frac{(\phi_{\text{Post}} - \phi_{\text{Pre}+n})\, f_s}{2\pi l}, \qquad (6.9) \\
\text{where } n &= arg \min_n \left(\Delta f_{\text{coarse}} - \Delta f_{\text{fine}}\right). \\
|\phi_{\text{Post}} - \phi_{\text{Pre}}| &< \pi \\
|\Delta f_{\text{fine}}| &< 0.5 \times 40\text{MHz}/50240\text{samples} \\
&= 398.09\text{Hz}
\end{aligned}$$

We ensure that the time between the preamble and the midamble is short enough that the ambiguities lie well outside the tolerance of the system. A 40MHz Clock source with ±20ppb stability synthesizes a 1GHz RF carrier signal results in frequency offsets below 40Hz when comparing two CHP2 systems due to clock source frequency drifting. The coarser frequency offset ambiguities are multiples of 1.1973kHz, and the fine frequency offset ambiguities are multiples of 796.1783Hz, so the estimators can dismiss the ambiguities for sufficient SNR, even when a reasonable Doppler frequency offset is included.

The performance this estimator improves with the increasing length between test points at the cost of close ambiguities. The Cramer-Rao lower bound for a simple frequency estimator is

$$\text{var}(\hat{f}) \geq \frac{12\ f_s^2}{(2\pi)^2 \eta N(N^2-1)} \approx \frac{12}{(2\pi)^2} \frac{1}{\text{ISNR}} \frac{1}{T^2}, \qquad (6.10)$$

where η is the signal to noise ratio, N is the length of the integration in samples, and f_s is the sampling rate. The strategy for producing the best frequency offset estimate is to first perform a coarse estimate between the preamble and midamble to disambiguate the possible solutions. Then perform a fine estimation between the preamble and the postamble. This is depicted in Figure 6.7. The first estimate has a standard deviation of less than 0.1304Hz at 10km, and the second has a standard deviation of

less than 0.0384Hz at 10km when the receiver SNR is about 20dB. However, these are frequency offset estimation performance lower bound. They may not be achieved due to system limitations and other factors.

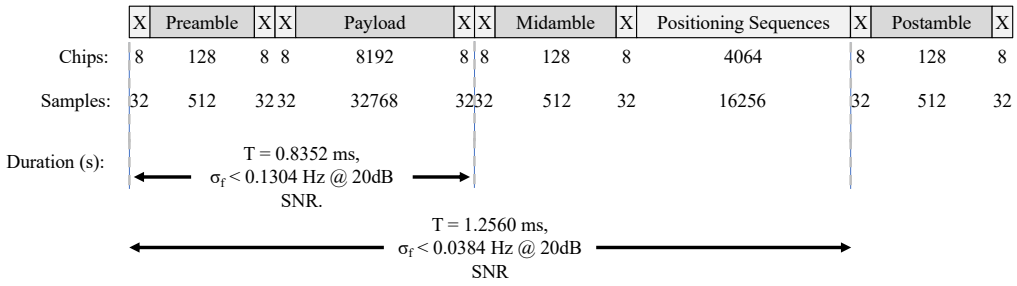


Figure 6.7: *Depiction of two frequency estimators using the preamble, midamble and postamble. The first estimator easily disambiguates the estimate but has a worse estimate. The second estimator has a better estimate but is susceptible to the ambiguities. The first estimate is used to disambiguate the second estimator. The standard deviations listed are absolute upper limits at 10 km.*

6.4 Positioning Waveform Design

For the positioning waveform payload, we consider a TDD mode that uses shorter waveforms within the positioning slot, but each antenna transmits while the others are off. It eliminates the inter-symbol interference and allows more options for waveform selection but limits the length of the sequences, which reduces integration time. Briefly, the positioning waveform is constructed from a random binary sequence of length 2000 from a Gold code and modulates them by the MSK modulation scheme to 1000 chips long.

Chapter 7

CHP2 HARDWARE IMPLEMENTATION[1]

This chapter describes the physical composition of CHP2 system. Except for the core of software defined radio, it also requires extra hardware to complete the CHP2 system. We finish the whole physical prototyping system for the in-lab tests and over-the-air demonstration.

7.1 Hardware Overview

The highlight of the CHP2 system is its decentralized application. Each node is identical on both hardware and software architecture. As a typical radio system, the CHP2 system requires a processing platform for digital baseband processing, system status control, user interface, monitoring, debugging, and data exchange interface with the upper-level system. Also, RF signal generation and acquisition require a set of RF front end components. Furthermore, a set of customized antennas for this particular application is required. Finally, supplement components, including power supply module, battery over-drain protection, power battery, PCB mounting frame, cable management, and others, are necessary. Those components can be classified into the following categories:

- **Motherboard: Xilinx ZCU102**: Xilinx Zynq UltraScale+ MPSoC ZCU102 Evaluation Kit. The MPSoC of this development board is Zynq UltraScale XCZU9EG-2FFVB1156, which is a combination of SoC IPs and FPGA programmable logic together.

[1]This is a joint work with Airbus ExO Alpha team.

- **RF frond-end: ADI FMCOMMS5**: This Analog Devices RF evaluation board includes dual AD9361 transceivers and supports up to 4x4 MIMO. **TRS Switch Board**: A customized transmission-reception switching card with external RF amplifiers.

- **Antenna**: a set of customized Omni-directional antenna cover US 915MHz ISM band, and EU 783MHz licensed experimental band.

- **Power Module and Battery**: Provide a DC-DC voltage converting and protection for the motherboard and RF front end cards. Power battery provides a large current and enough capacity for the whole system.

- **Accessory components**: It includes PCB mounting frame and cable management. The weight constraint and material strength are required.

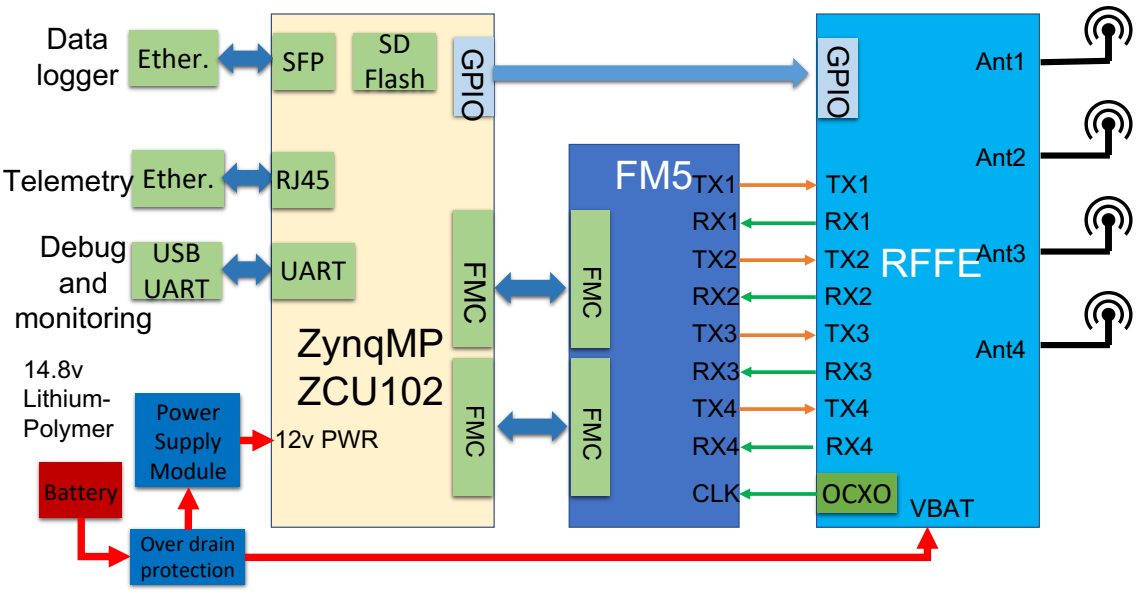


Figure 7.1: *A brief block diagram of CHP2 assembly.*

7.2 CHP2 System Hardware Design Requirements

As a high precision radio localization system. the final performance of CHP2 can only be achieved under certain design prerequisites.

- **Carrier Frequency**: CHP2 uses a 915MHz ISM band in the USA and a 783MHz licensed experimental band under European regulation. Also, lower carrier frequency performs better on long-distance propagation and has less attenuation over the atmosphere.
- **Signal Bandwidth**: The RF bandwidth of the CHP2 system is 10MHz.
- **RF Power Requirement**: The goal of the CHP2 system is to work over 5 kilometers range properly. Thus, the maximum instantaneous transmission power is limited to 30dBm. Due to its power control and RF signal duplex mode, the average power is much less 20 dBm.
- **MIMO Setup**: CHP2 can measure range and angle by the 4×4 MIMO configuration. However, all MIMO links have to be phase coherent. A total of 16 links generate a large number of data. The baseband process has to catch up with the data rate.
- **Data I/O and Interface**: CHP2 will be finally integrated with other systems. It has to provide data interface and monitoring options.
- **Payload Integration**: The weight of CHP2 assembly on the drone has to less than 3kg, not including battery.
- **Antenna Configuration**: The antennas on the ground station and drone have to be onmi-direction and support dual bands for the US and EU.

7.3 Signal Processing Hardware Architecture

The processing platform of the CHP2 system is chosen as the Xilinx ZCU102 development kit. This kit integrates an MPSoC (Multi-Processor System-on-Chip) device XCZU9EG from Xilinx Zynq UltraScale plus series. As the highlight of this series, it is a powerful combination of the Processing System (PS) and Programmable Logic (PL). In other words, this MPSoC is a fusion of multicore generic processor and FPGA. Specifically, it includes a quad-core ARM Cortex A53 processor, a dual-core Cortex R5 Real-time processor, a Mali400 MP2 graphics processing unit, and a large section of FPGA with 60K logical cells including over two thousand DSP slices and multiple data interface IPs. Utilize the generic processor for general processes while takes advantage of the powerful FPGA such as flexibility in customized logic, being specialized in heavy arithmetic loading, and high data throughput application.

Because of such benefits from Zynq MPSoC (ZynqMP), it implemented the base-band processing chain of CHP2 on this MPSoC. The components of demanding massive mathematical computation and raw sampling data handling are directly fabricated on the FPGA section, including frame detector, massive correlation, data buffer/handing, DMAs, timestamp/timer related logic, and timing control logic. Those components strictly require a short amount of execution time, when handling tremendous of raw sampling data, and also highly demands timing precision. The processes of less data throughput and light computation loading are running over the quad-core ARM processor, such as channel equalization, demodulation, trellis-decoding, Time of arrival algorithm. The details of the processing chain will be discussed in the later chapter.

Moreover, as the most significant feature of the ZCU102 board, it is equipped with dual FMC extension ports to mount external daughter card for high-speed data rate

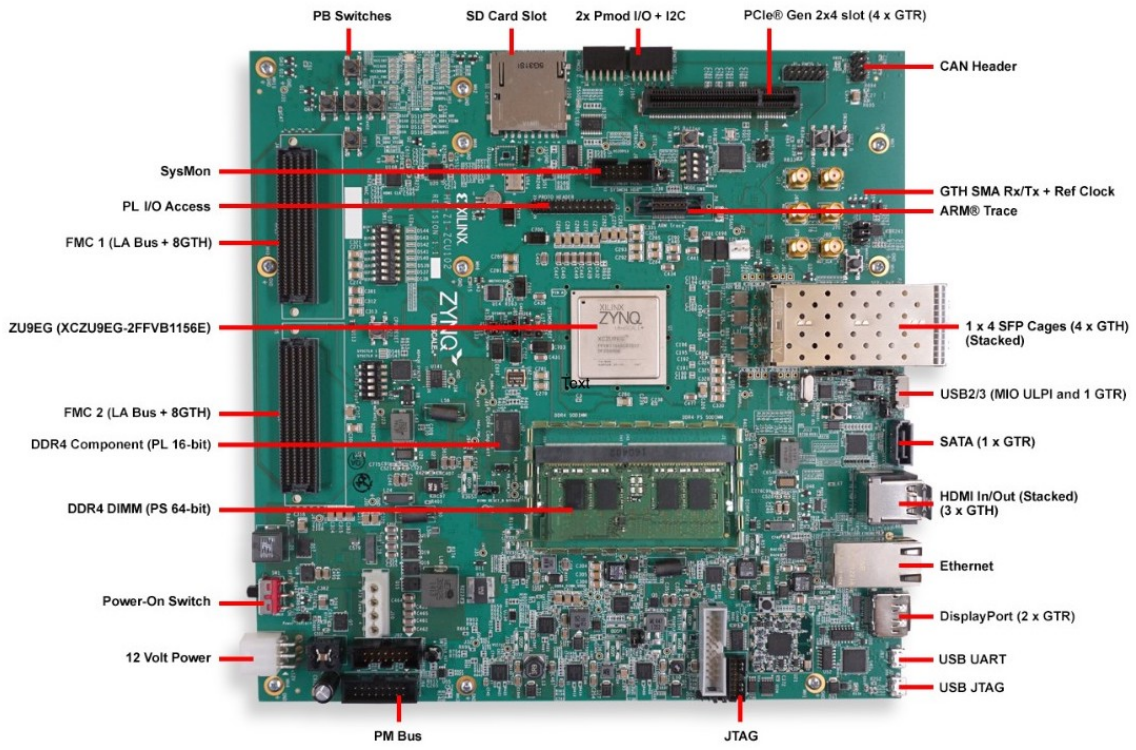


Figure 7.2: *Depiction of major components and interfaces on ZCU102 development board[58].*

applications. Especially in this CHP2 system, those FMC extension ports connect the ADI FMCOMMS5 transceiver card for handing up to 4×4 MIMO full-duplex data streams with a maximum RF signal sampling rate at 56MHz. Typically, a maximum IQ data rate of 8847.36Mbps on both transmitting and receiving directions under 4Tx4Rx mode can be reached on the FMCOMMS5 card. The ZynqMP should catch up with this data rate without much difficulty. Also, the ZCU102 board provides multiple external interfaces for data exchange, debugging, monitoring such as Gigabytes Ethernet ports, JTAG, UART in CHP2 system implementation.

As the brief diagram of CHP2 hardware show in Figure (7.3), both PS and PL sections are inside the ZynqMP chip. The PS section contains processors and some general IP cores for running operating system or bare-metal programs and general

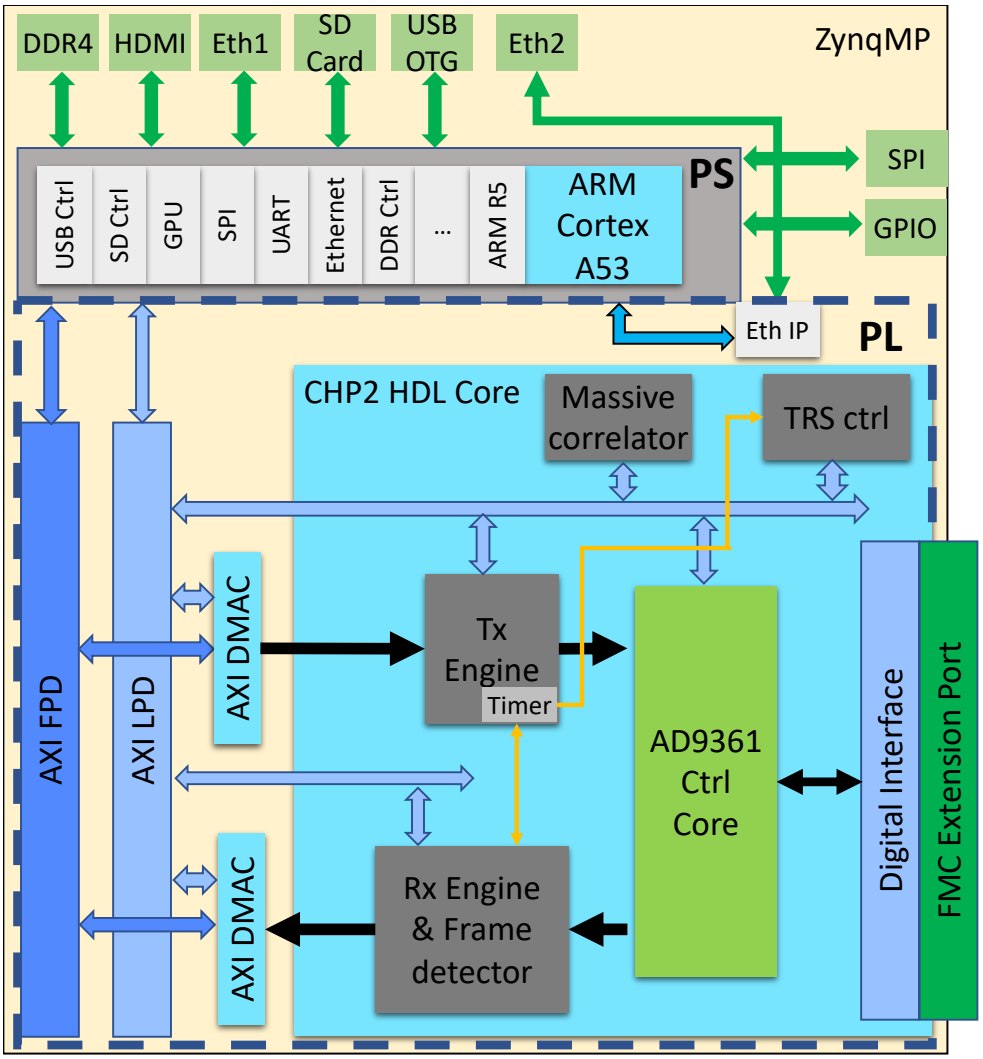


Figure 7.3: *Depiction of signal processing architecture diagram implemented on the ZynqMP Platform.*

data I/O operation. The PL section is equivalent to the traditional FPGA. It contains the IPs which are synthesized from HDL(hardware description language) coding. The CHP2 customized IPs and Analog Device provided IP(AD9361 controlling core) are listed in the PL section. The IO ports and DDR4 surrounding the PS section can be mapped to the physical port and memory on the ZCU102 board in Figure (7.2).

7.4 CHP2 RF Front End

The ZynqMP device dominates the whole digital domain of the CHP2 system while a set of RF front end components have to be responsible for RF and Mixed RF-digital domain. The essential RF component on the RF front end of the CHP2 system is the Analog Devices AD9361 integrated transceiver. Thanks to the rapidly evolving mixed Analog-Digital technology, AD9361 is the first commercially available integrated radio transceiver in the world. It provides a completed RF solution by combining DAC/ADC, RF mixer, RF filter, programmable RF amplifier, FIR filter, high-speed data interface onto a single chip. It coves 70MHz to 6GHz with a sampling rate of up to 56MHz. A single AD9361 is also capable of 2 Rx and 2 Tx MIMO channels with a data interface full-duplex mode (not RF signal full-duplex).

7.4.1 FMCOMMS5 Transceiver Board

CHP2 is a 4×4 MIMO system, not only measuring range but also finding direction and disentangling traveling orientation. Therefore, the RF front-end used in CHP2 is chosen as an Analog Device FMCOMMS5 card. Two AD9361 transceivers are being able to work simultaneously on this RF card. Also, this FMCOMMS5 can be mounted onto the ZCU102 platform through FMC ports. Also, this RF daughter card contains an external carrier synthesizer ADF5355, which synthesizes carrier signal and distributes carrier to both AD9361 for maintaining phase-coherent across all transmission and receiving channels. Phase coherent is critical for the CHP2 system.

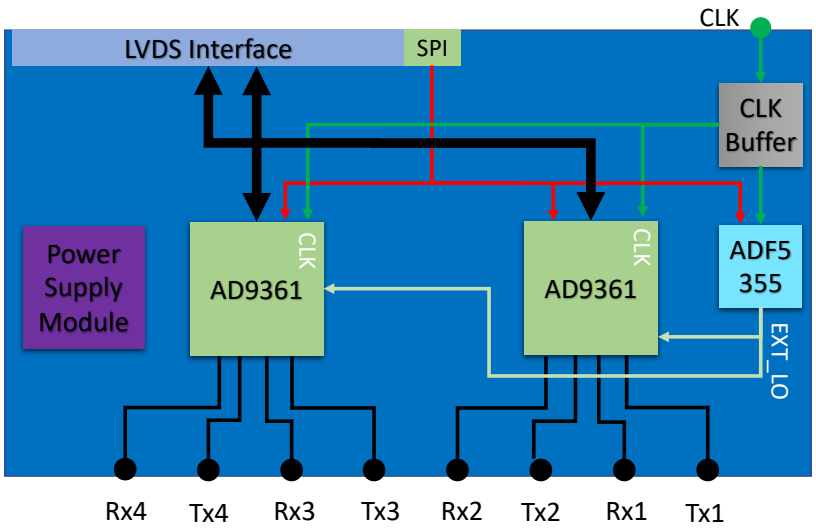


Figure 7.4: *Depiction of FMCOMMS5 transceivers block diagram.*

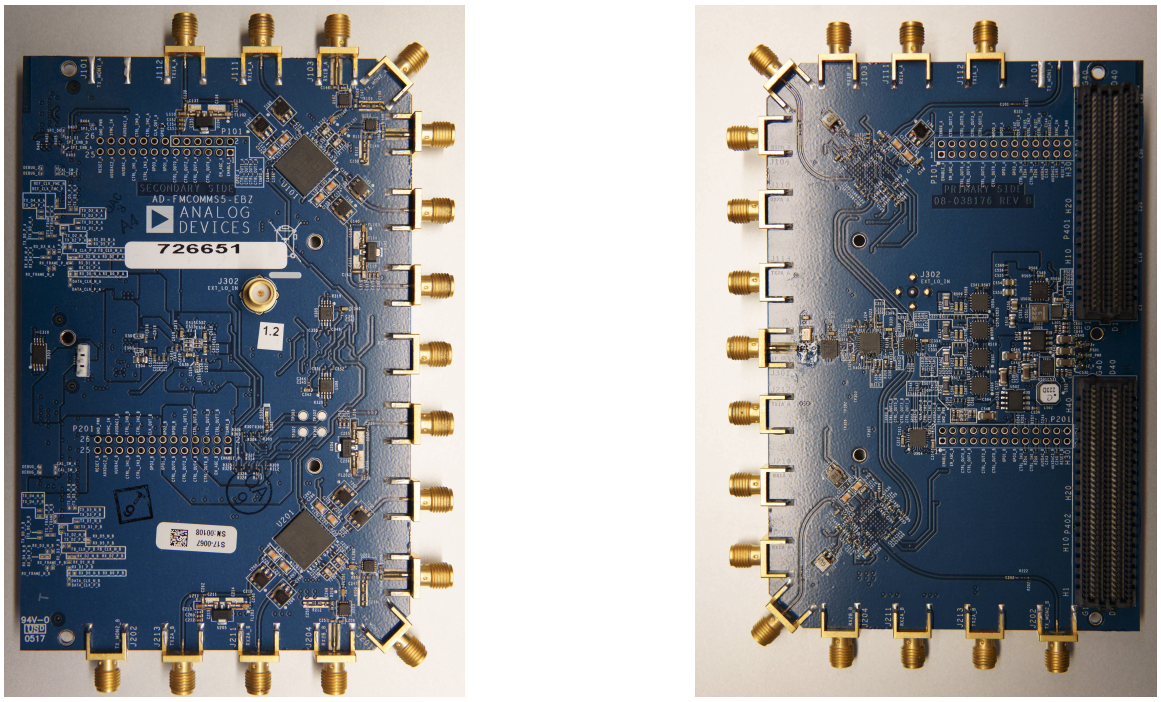


Figure 7.5: *The actual images of the FMCOMMS5 transceiver card. It clearly shows two AD9361 with 4 Tx and 4 Rx channels on the front. The external clock input port and the external carrier synthesizer ADF5355 distribution circuit is on the back*

7.4.2 TR Switching Board

Moreover, because of the MAC layer structure of CHP2 as Figure (6.1), the RF signal transmission and reception are sequential in time similar to Time Division Duplex (TDD). This duplex mode requires a TR switch on the RF chain. Although the AD9361 transceiver has TDD mode, there is no enough amplifier to meet the transmission power target. Also, with other concerns (calibration, RF signal filtering), a customized TR switching amplifier board is proposed for the CHP2 project. Its schematic diagram is shown as Figure (7.6).

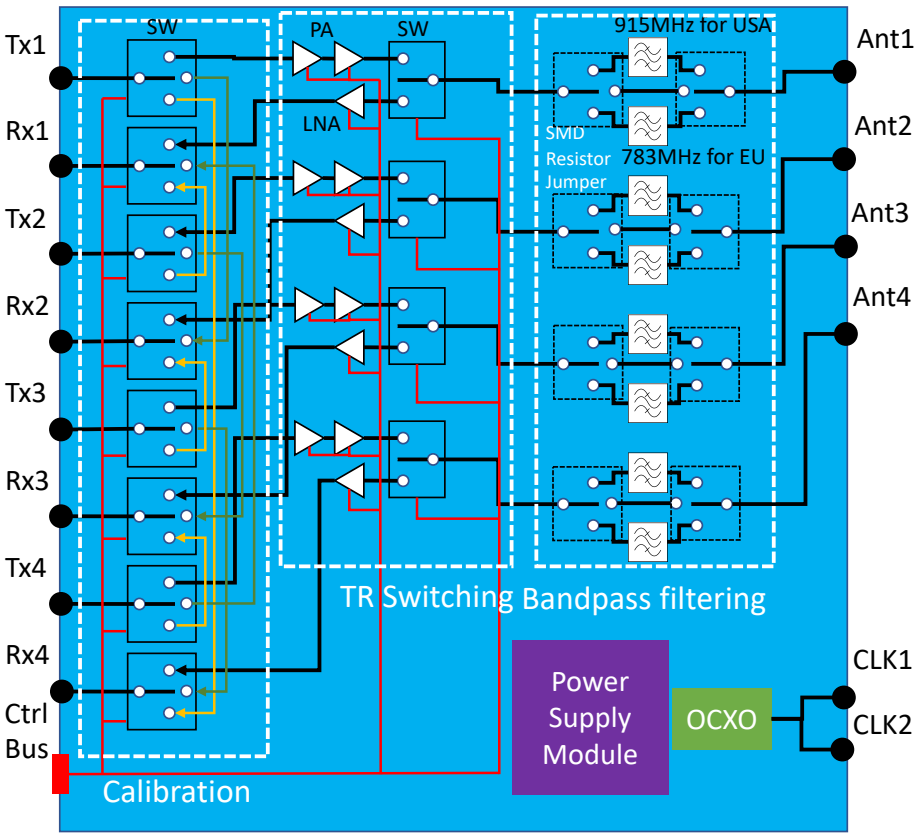


Figure 7.6: *Depiction of CHP2 TR Switching block diagram.*

On this TR switching board, there are four sections: phase calibration switching, transmission-reception switching, dual-band filtering section, power supply module,

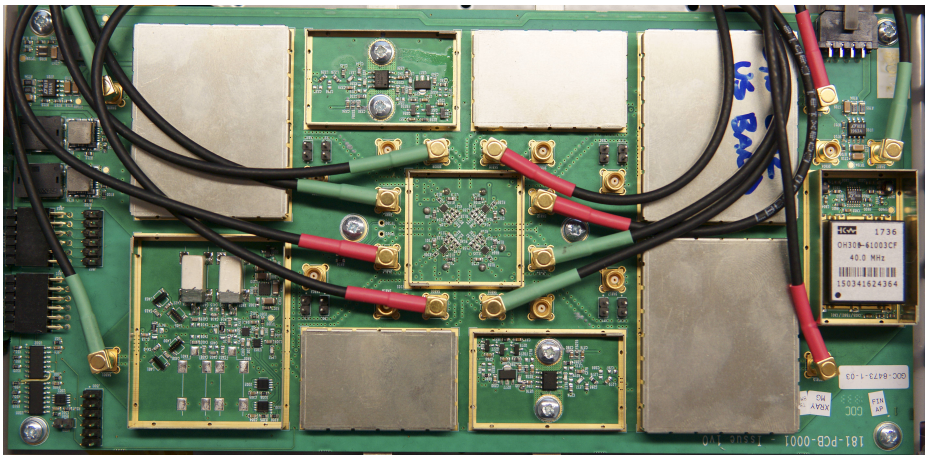


Figure 7.7: *The customized CHP2 TR switching amplifier board.*

and a 40MHz oven-controlled crystal oscillator (OCXO). The calibration section is for phase calibration purposes to make sure all transmission and reception channels have the phase alignment. The TR switch section is the primary purpose of this PCB. Also, extra PAs and LNAs are mounted to make sure the maximum transmission power can reach 30 dBm when the input power is around 0 dBm. The reception signal is strong enough when the distance of the two systems is far away. Both PAs and LNAs can be bypass or disabled by the controlling bus. The filter section provides bandpass filtering on the CHP2 waveform. It consists of three modes: the USA spec filter, the EU spec filter, and the direct-pass, which are chosen by the corresponding zero ohm resistors. In the clock source section, compared with the crystal clock on the FMCOMMS5 card, this OXCO source gives superior frequency stability (10∼20ppb) and helps to decrease the number of system imperfections during the prototyping stage. The power supply module converts the proper voltage for all electrical components on this PCB from the primary battery source.

7.5 CHP2 Antennas

The antennas of the CHP2 system are also fully customized. There are two categories of the antenna in the CHP2 system: the ground station antenna and the drone segment antenna. The ground station antennas are installed on the top of four height-adjustable stands. The drone antenna can be mounted rigidly on the drone. Both antennas' gain is 1.9dB (EU band) and 2.1dB (US band) with an omnidirectional plane. As Figure (7.8) shows, the return loss is excellent across the entire frequency range of interest.

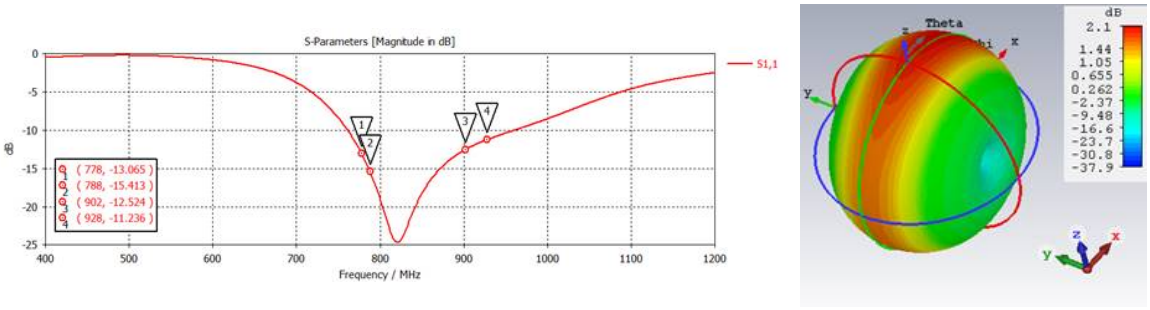
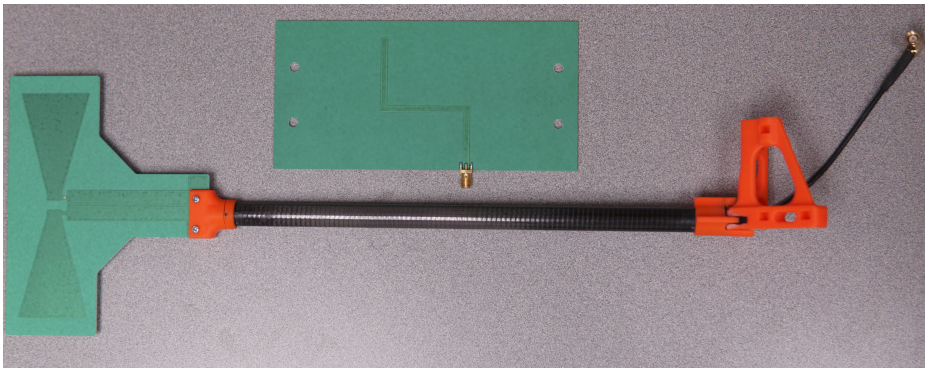
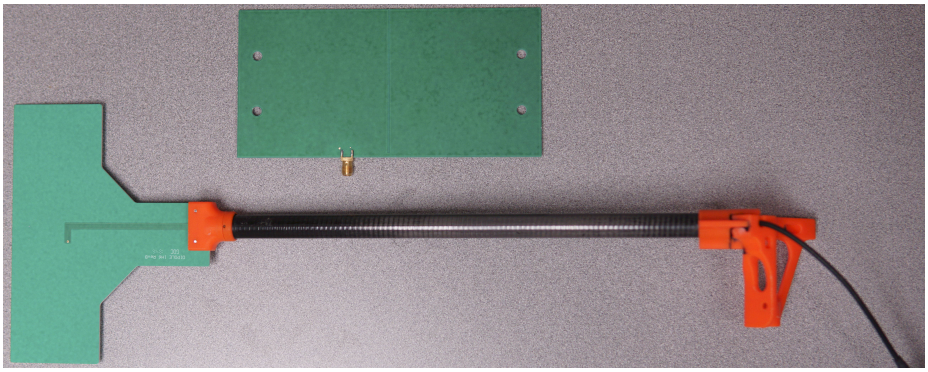


Figure 7.8: *Depiction of S11 return loss and omni-directional plane of CHP2 system antenna. The CHP2 antenna perfectly covers the range of EU band and USA band with exceptional return loss.*



(a) *The front of antennas*



(b) *The back of antennas*

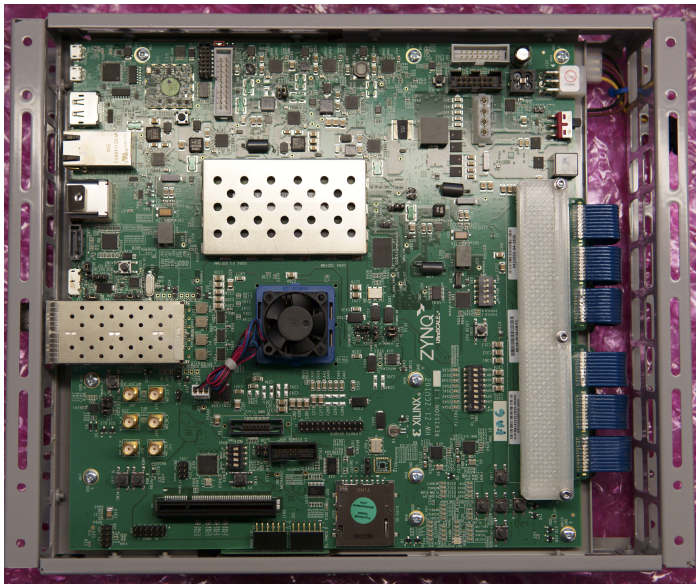
Figure 7.9: *Depiction of the CHP2 antenna. The top one is the ground station antenna. The one on the bottom is the drone antenna, which is pre-mounted onto a rigid carbon filter tube with mounting bracket. They are mounted to the four corners of CHP2 assembly.*

7.6 CHP2 Assembly

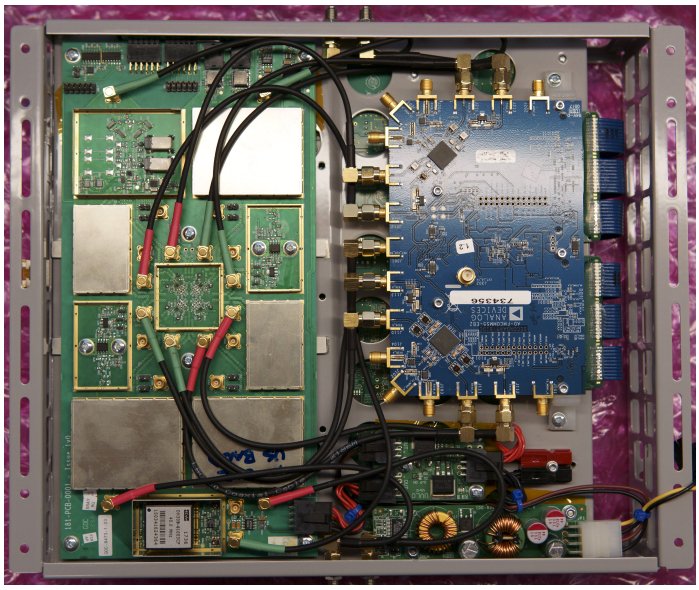
The CHP2 assembly is the final integration of all hardware onto a single aluminum alloy frame. This metal frame is mounted onto as the payload of an experiential drone with transmission antenna on four corners. The ground station also uses the same assembly but inside a portable weatherproofing box.

As described in Figure (7.1), the CHP2 assembly consisted of ZCU102 board, FMCOMMS5 board, and TR switching board. Except for the major components, power supply, and battery over-drain protection modules are also installed. Moreover, all data exchange interfaces, such as dual RJ45 ports and USB UART port, are preserved for telemetry and data-logging devices as debugging and monitoring purpose. The SD card installed onto ZCU102 is the primary firmware storage. Once the system is power-on, the embed Linux OS, the CHP2 baseband process, and other higher-level applications are loaded from here and then executed.

The metal frame not only integrates every component but also provides a safe environment for all electronics properly running. It improves the strengths of hardware assembly and the convenience during handling and installation. The light metal material deduces total weight and increases the heat dispensation of some device, such as a power amplifier. Additionally, all necessary electrical connectors can be accessed seamlessly, such as RJ45 ports, antenna ports, USB port, and HDMI, as in Figure (7.11) and (7.10).

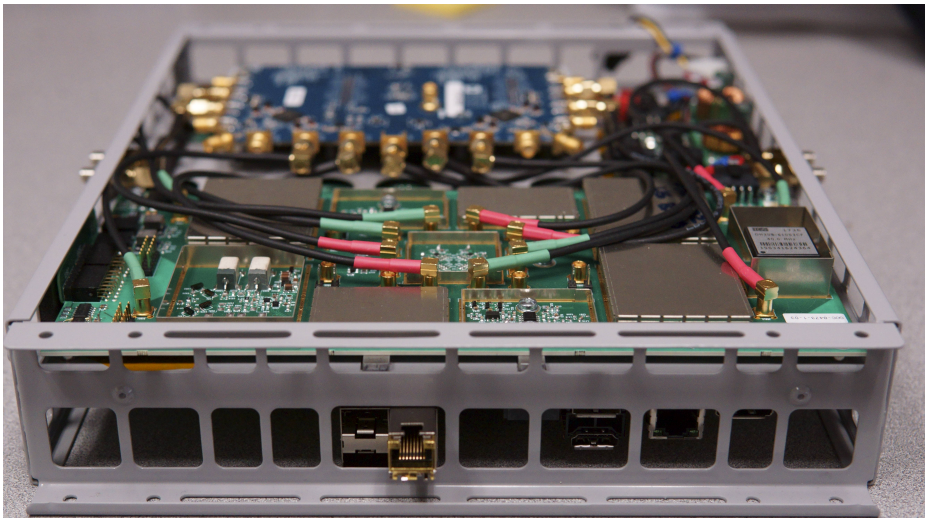


(a) *The top view of the CHP2 assembly*

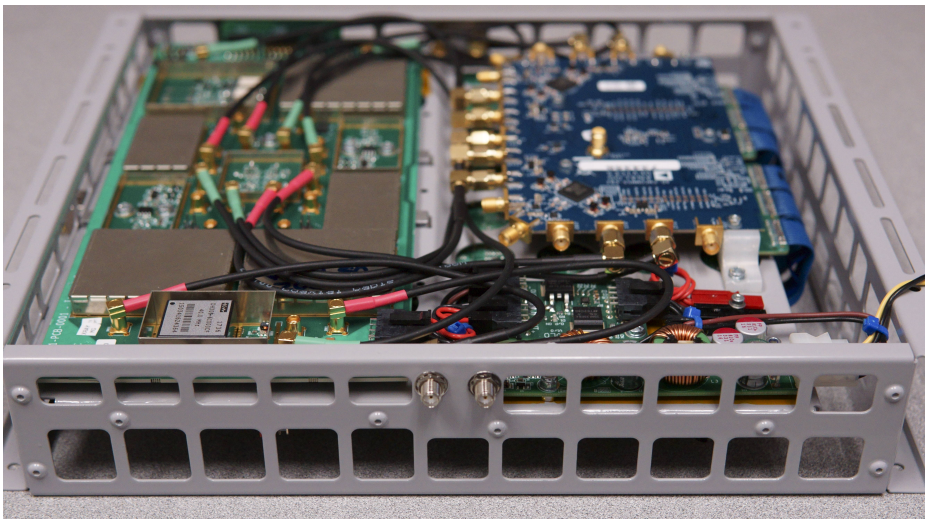


(b) *The bottom view of the CHP2 assembly*

Figure 7.10: *The top and bottom view of the CHP2 assembly.*



(a) *The side view A of the CHP2 assembly*



(b) *The side view B of the CHP2 assembly*

Figure 7.11: *The side views of the CHP2 assembly. All external port can be access easily. The side drilled holes are used to mount drone antenna support bracket on the bottom of Figure (7.9).*

Chapter 8

CHP2 PROCESSING ARCHITECTURE[1]

This chapter outlines the signal process chain of the CHP2 system. It covers the waveform transmission and reception signal processing chain. The main focus of this chapter is to explain the acquisition of the high precision timestamp from the CHP2 frame.

8.1 Overview

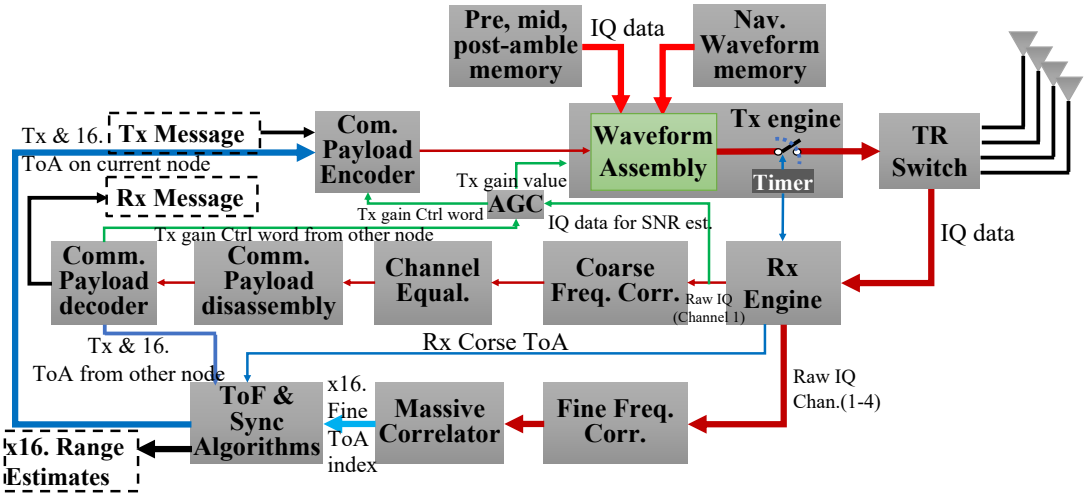


Figure 8.1: *Depiction of CHP2 processing chain.*

The CHP2 processing chain, as in Figure (8.1), is built on the SDR platform ZCU102 with the FMCOMMS5 RF front card. Analog Device provides an open-source C-language based AD9361 driver library, and Verilog based HDL core library for necessary reception and transmission. However, all the available functions only

[1]This is a joint work with Hyunseok Lee.

operate the raw IQ data on the low physical layer. In this case, the CHP2 processing chain has to be built from scratch. Meanwhile, this SDR platform has a powerful potential capacity but is entirely new to the SDR community. Much extra effort is spent on making the SDR basic skeleton working. However, the benefit of this SDR platform is that the RF transceiver AD9361 is pretty mature and covering a complete RF related features. Therefore, the processing chain is entirely digitized without touching too much on the RF domain. It helps to accelerate radio design rapidly.

The following components are built on transmission chain:

- **CHP2 Data Payload Composite**: Gathering the data bits from existed time stamps and other necessary information by using payload compression mentioned in Section (6.3.1).
- **Payload Assembling**: Payload data bits go through CRC checksum, scrambling, error control coding, modulation, and spreading.
- **CHP2 Frame Assembling**: Add preamble, midamble, and postamble to payload data waveform. Then attach all four navigation waves between midamble and postable sequences.
- **Pulse Shaping Filtering**: A fir filter reduces inter-symbol interference and constrains the signal bandwidth within 10MHz.
- **Transmission Engine**: A buffer temperately holds the raw IQ data of the CHP2 frame wave and then sends these IQ data to the AD9361 control core based on the exact transmission timestamp.

The following components are built on reception chain:

- **Frame Detection or Receiving engine**: A cross-correlator acquires the incoming preamble wave and records the coarse receiving timestamp. Then output the following CHP2 frame wave to the Rx processing chain.
- **Frequency Correction**: Estimate and correct the carrier frequency offsets on the CHP2 waveform.
- **Channel Equalization**: A standard communications equalizer using MSE (mean square error) method.
- **Payload dissembling**: A reverse process of payload assembling.
- **Massive Correlation**: Cross-correlate the received navigation waveform with the stored up-sampled waveform to estimate the fine receiving time stamp.
- **ToA algorithms**: The algorithms use available timestamps from local CHP2 node and remote one to estimate the high accuracy ranges between two nodes. It also includes the converting MIMO ranging results to actual geometry coordinates.

Among the above components, The transmission engine, the receiving engine and massive correlator are implemented onto the programmable logic section on ZynqMP. Other components are implemented onto ARM processor with a particular parallelism optimization technique.

8.1.1 CHP2 Processing Chain Design Requirement

As a physical prototyping radio ranging system, the design of CHP2 signal processing chain has to meet the following requirements and conditions.

- **40MHz Main Clock**: This source is not only the IQ data sampling clock but also the main clock for all digital logic components involving IQ data directly manipulation. Especially, the CHP2 main timer, the transmission and receiving engines are driven by this clock source.

- **CHP2 Frame Refresh Rate: 10Hz**. In this prototyping two CHP2 nodes setup, the range estimates refresh every $100ms$.

- **Maximum Processing Time Per Frame: 50**ms. Due to the two nodes setup, the receiving processing has to be finished under $50ms$ because there is a timestamp exchange involving two nodes within 100ms interval. Each node only has a maximum of $50ms$ to finish all the receiving and transiting process.

- **Massive Correlation resolution: 0.5ns**: The designed navigation sequences has a chip rate of 40MHz. Applying the massive correlation is equivalent to cross-correlate with a 50x up-sampled sequence to find the exact time of arrival. The accuracy of this fine ToA is 1/40MHz/50 = 0.5ns about 200 times better than coarse ToA from the envelope of 10MHz bandwidth waveform. The resolution is $0.5\text{ns} \times 3 \times 10^8\text{m} = 0.15\text{m}$ about half of the carrier wavelength.

8.1.2 AD9361 HDL Core and interface

The AD9361 HDL core provides the controlling, data stream exchanging and driver clock input/output with the AD9361 chips on the FMCOMMS5 card. Also, there is a software driver interface (Analog Device Industrial I/O subsystem, IIO) provided by Analog Devices for operating AD9361 transceivers. On the lower-level physical layer, the existed HDL core and driver interface are critical because they provide the non-trivial features to the developer. These features include RF related configuration on transceiver chip and IQ data exchange mechanism through DMA to LVDS port. The developer can directly operate and configure transceivers by accessing an IQ data memory and a list of mapped function registers. It helps the developer to focus on customizing the processing chain and seamlessly operate the transceiver without understanding too many details of AD9361. The original HDL core and driver interface of the ZCU102 FMCOMMS5 platform is shown in Figure (8.2). The programmable logic (PL) section contains the AD9361 controller core, AXI bus-based DMAs. The software section running over the ARM processor (PS) includes the IIO driver and programmable interfaces for user processes. As an SDR skeleton, the existed components are good enough for low bandwidth radio applications. However, for the CHP2 system such that demands precise timing and carries heavy mathematical loading, it requires much additional development over the PS and PL section.

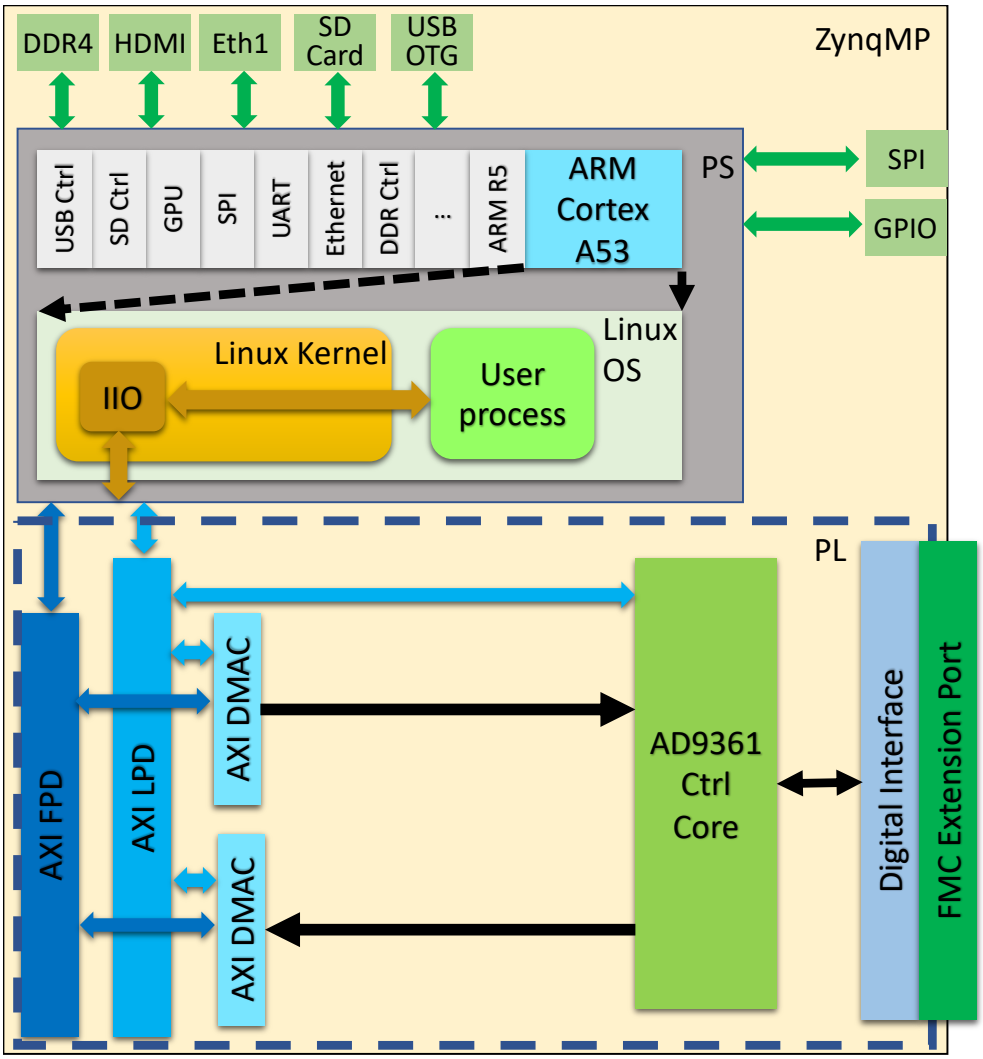


Figure 8.2: *The original Architecture of ZCU102 FM5 SDR from Analog Devices*

The CHP2 system is a typical communication system following a simplified Open Systems Interconnection model (OSI model). The bottom is the RF layer, which contains all the RF related hardware and setup, including the transceivers and TR switching amplifier board.

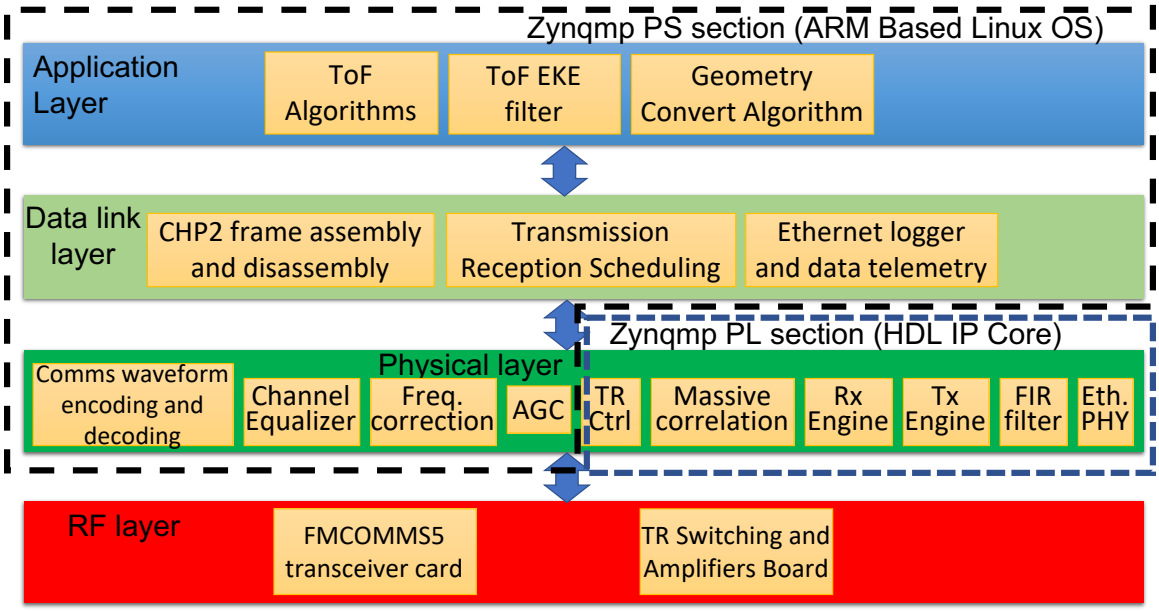


Figure 8.3: *Depiction of a simplified OSI model of the CHP2 system.*

The layer above the RF layer is the physical layer which is responsible for most of the signal processing functions during this wireless communication. As Figure (8.3) shows, part of the physical layer is implemented onto the PL section, which contains the most timing critical tasks and massive throughput mathematical computation tasks. The remaining of the physical layer is deployed onto the PS section, which is the signal processing program running over the ARM cores.

The next layer is the data link layer. This layer is functioning as the CHP2 frame transmission and reception scheduling, frame assembly, and disassembly. It schedules

the transmitting and receiving events while acquires and packs timestamps from or to the CHP2 frames. An Ethernet logger program is also running on this layer for data logging and telemetry data sending purposes.

The most upper layer is the application layer. Time-Of-Arrival estimation algorithms are living on this layer. The typical ToA algorithm implemented here is a modified NTP algorithm, and a ToA extended Kalman filter. They are different methods but serve the same purpose. Another one is the geometry converting algorithm, which converts the MIMO range estimates onto the target coordinate in real-time. The data link layer and application layer are implemented as part of the CHP2 program running on the ARM-based Linux OS.

8.3 CHP2 Physical Layer

Similarly to the original Analog Device released SDR architecture[48], the CHP2 physical layer is implemented over the Programmable Logical (PL) section and the Processing System (PS) section on ZynmqMP SoC. In the PL section, most of the timing critical components of the physical layer are implemented as HDL IP cores. Except for the original AD9361 controller core and AXI DMAs are reserved, the CHP2 Tx engine and the Rx engine are customized and inserted onto the IQ data paths. Moreover, the CHP2 massive correlator, the TRS board controller, and the carrier-synthesizer phase-reset controller are inserted onto the AXI LPD bus.

In the PS section, the CHP2 processing programs are running over the ARM Cortex-A53 cores. This customized processes on Linux user space can access the PL section through the IIO driver library and the memory mapping function provided from Linux kernel. Partial of the physical layer, including communication waveform encoding and decoding, channel equalizer, frequency offset estimation, and correction, as well as automatic gain control, are implemented as the CHP2 programs. As in Figure (8.3), the application layer and data link layer are also implemented in the CHP2 programs.

In the aspect of the transmission chain, once the raw IQ data are ready from CHP2 processes, the Tx engine delivers them to the AD9361 controller core. Then the IQ data are packed onto a unique format by the AD9361 controller core and exchanged with AD9361 transceivers on the FMCOMMS5 card through the FMC extension port as in Figure (8.4). On the reception chain, once the Rx engine acquires the CHP2 frame from the AD9361 controller core, it delivers the raw Rx IQ data to CHP2 processes in the PS section. On the actual hardware, the boundary of the layers is not always clear. The transmission and reception engine involves both the physical

layer and the data link layer.

The CHP2 PS accesses the PL section through the AXI bus. There are two types of AXI bus in this particular design: AXI FPD (full power domain) and AXI LPD (low power domain). The AXI FPD bus is particularly used for the IP cores, which demands high data throughput and high refresh rate data streams such as DMAs. It directly handles the IQ data streams between the CHP2 PS and the AD9361 transceiver. The DMA moves IQ data onto the PS memory space, which is open to the user application. The AXI LPD bus is used for the IP cores which do not demand high data throughput but use memory mapping as the data exchange method in this architecture. Almost all of the IP cores used in the CHP2 PL section are mounted to the AXI LPD bus for register-memory mapping access purposes.

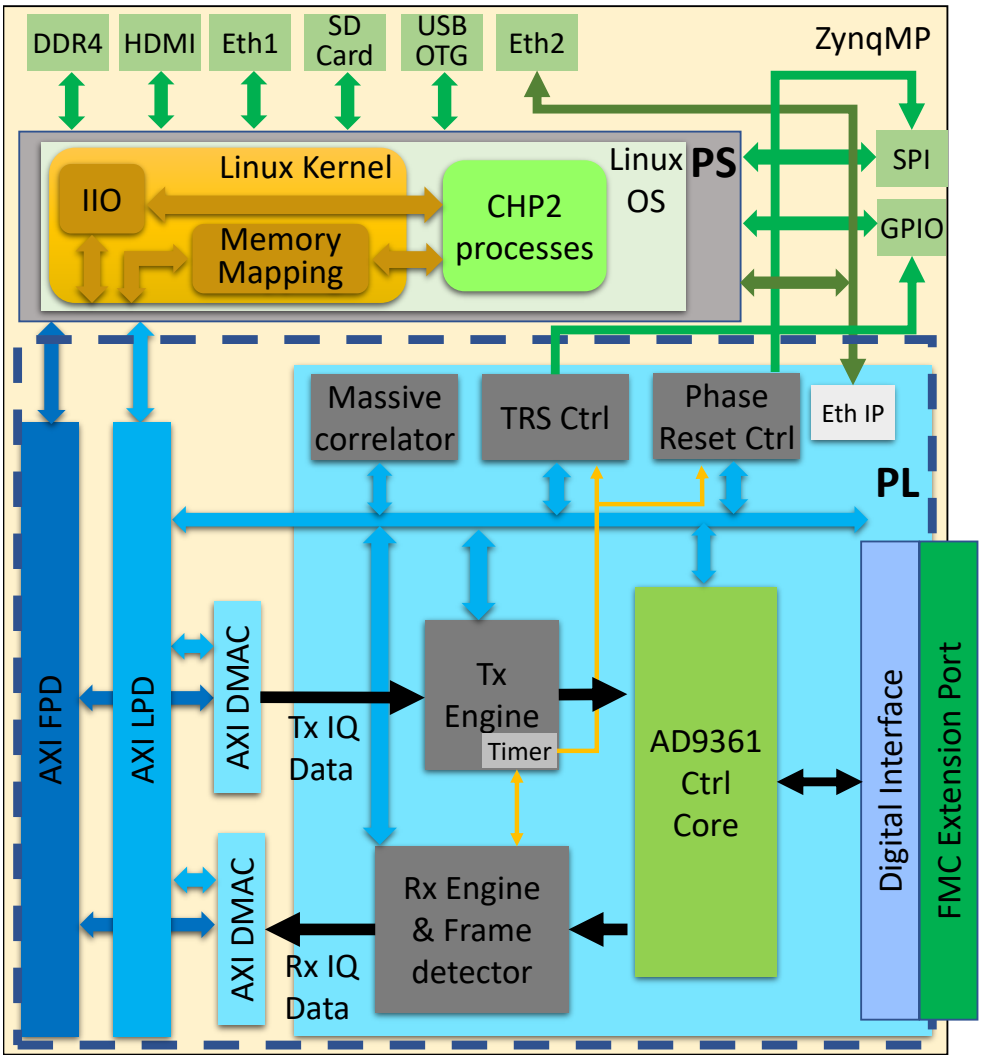


Figure 8.4: *Depiction of the CHP2 system architecture on ZynqMP platform*

8.3.1 Transmission Engine

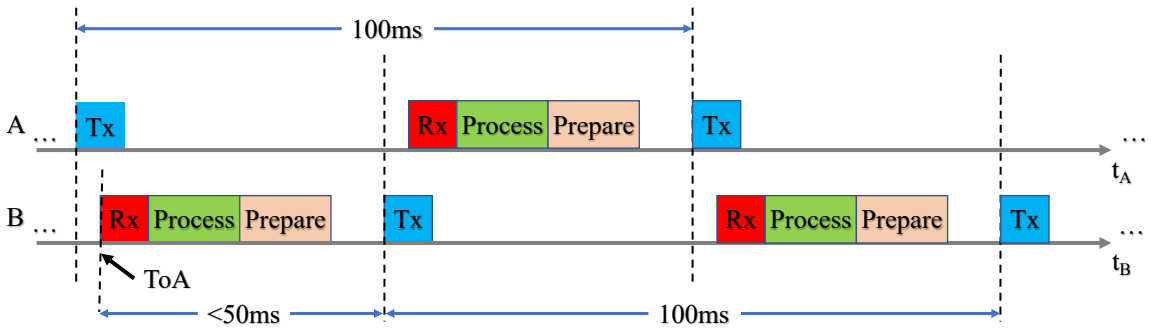


Figure 8.5: *Depiction of the CHP2 system MAC layer behavior*

As the two CHP2 nodes scenario in Figure (8.5), from the view of a single Node, the transmission occurs every 100ms periodically and precisely. Based on this fact, the transmission engine is a large IQ data buffer collaborating with a timer-controlled finite status machine. There is a CHP2 primary timer. It is driven by the 40MHz CHP2 primary clock, which comes from the OXCO source on the TR switchboard. All time-related events or operations use this primary timer as a reference source. During the CHP2 system properly operating, the finite state machine of the transmission engine is designated a transmission timestamp through the AXI LPD bus from CHP2 processes periodically. Also, the buffer is filled with the raw IQ data from the PS section. Once the transmission time is reached, the buffer content will be dump to the next AD9361 control core for the actual transmission. Since the path length of the Transmission engine - AD9361 control core - AD9361 transceivers is constant and fully digitized, there is no too much uncertainty on the timing. Therefore, the actual transmission time should have a fixed bias. The bias can be corrected during the calibration stage.

As Figure (8.6) shows, the CHP2 transmission IQ data comes from the PS section and is stored onto local block memory through the AXI bus instead of original

using the AXI DMAC. By using a simple logic switch, the CHP2 implementation is fully compatible with any existed Analog Devices driver library, and Analog Devices provided applications.

In the transmission engine IP block, the block memory is filled by the preamble sequence, the payload wave, the midamble, the postamble, and the navigation sequence. The generation of these waveform and payload composite are described in Section (6). Usually, the transmission IQ data preparation is much faster than the reception processing and less than 5ms on the current system implementation. In a word, as a crucial function of the Tx engine, it maintains the 100ms transmission interval precisely.

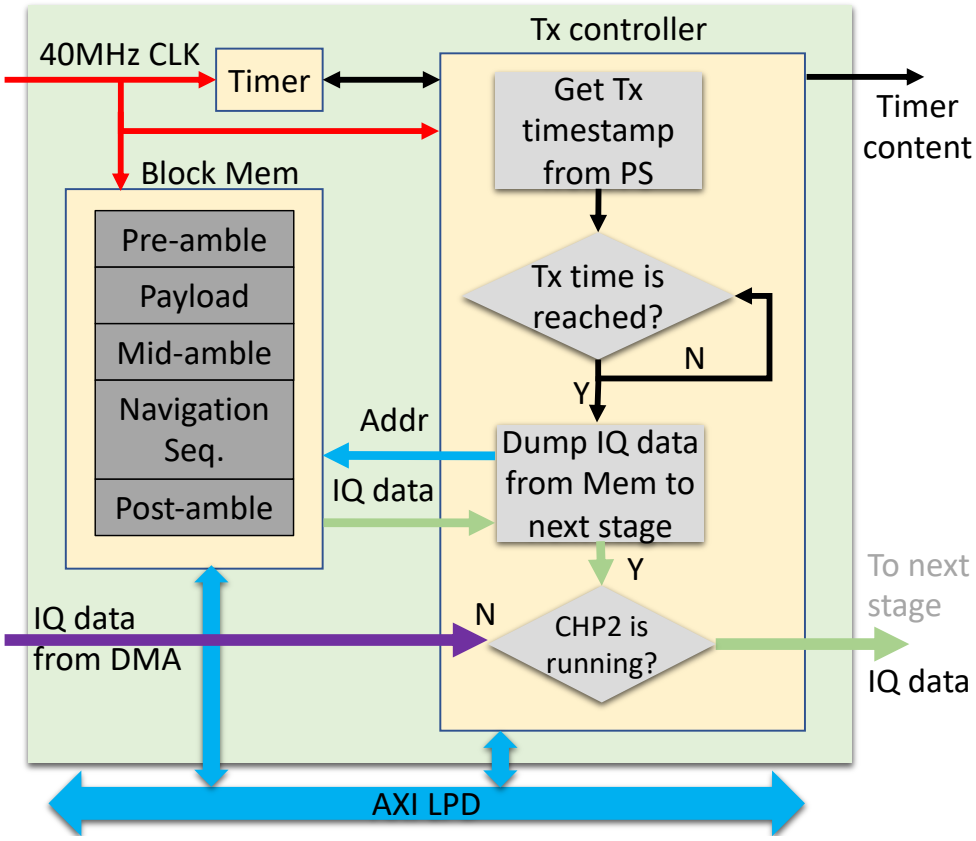


Figure 8.6: *Depiction of the CHP2 TX engine.*

8.3.2 Receiver Engine

The receiver engine is for the CHP2 frame acquisition purpose. It consists of a frame detector, an FIR filter, a block memory, and other miscellaneous. During the signal acquisition stage, the frame detector is continuously cross-correlating the received signal with a preloaded preamble wave stored in the block memory. Once the correlation result goes beyond the threshold value, the frame detector declares the CHP2 frame is detected and then dumps the waveform onto the DMA on the reception chain. It then moves IQ sample data onto the memory of the PS section. Also, the frame detector records the coarse receiving timestamp and reports it to the CHP2 system program on the PS section. The coarse receiving timestamp is label as $\hat{\tau}_c'$ as Equitation (3.5).

The core component of the frame detector is a cross-correlator. It cross-correlates the received signal with the stored 128 symbols preamble waveform. Instead of cross-correlating with full 512 IQ samples of the preamble waveform, it only cross-correlates the first samples in each symbol. Therefore, there are 128 taps on this match filter design. This design can achieve a similar performance from cross-correlating full 512 samples but saves a vast resource. Since according to Cramér–Rao upper bound, the coarse ToA estimation variance is theoretically related to SNR. It may have some numerical error, but the variance is still within one symbol or four samples. This error can be compensated by applying the massive correlation, which can handle ± 1 symbol shifting, during the fine reception timestamp estimation.

As Figure (8.8) shows, the complex number cross-correlation operation is designed as a systolic array. The arithmetic format of this array is a fixed-point number. The pipeline length is six stages. Meanwhile, the correlation peak estimation is not only comparing with the threshold but also finding the maximum local peak. This method

prevents the frame detector from duplicated triggering on the same received preamble wave due to the multi-path effect.

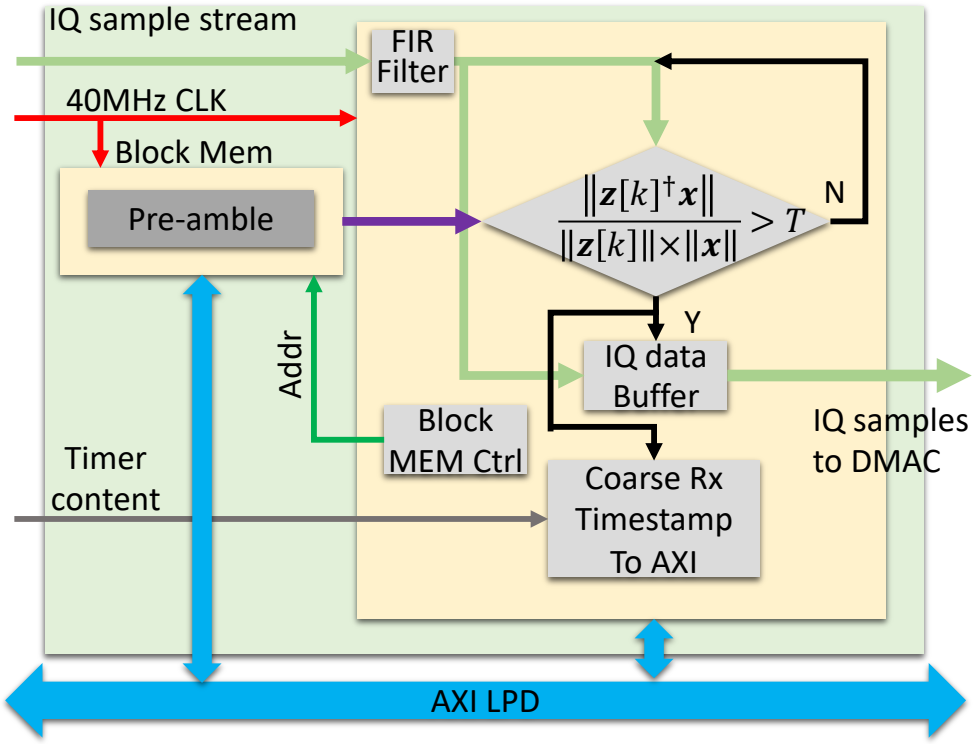


Figure 8.7: *Depiction of the CHP2 RX Engine.*

- $\mathbf{z}[k] : n \times 1$ is the received signal stream. 1 sign bit, 15bits fraction. bits.
- $\mathbf{x} : 128 \times 1$ is the preamble waveform. 1 sign bit, 1 integer bit.
- $\mathbf{g}[k] : n \times 1$ is the normalized cross-correlation result. 0 sign bit, 15 integer bits, 30 fraction bits.
- $\|\mathbf{z}[k]^\dagger \mathbf{x}\|$ is the cross-correlation result. 0 sign bit, 15 integer bits, 30 fraction bits.
- $\|\mathbf{z}[k]\| \times \|\mathbf{x}\|$ is the normalization value. 0 sign bit, 7 integer bits, 30 fraction bits.

The threshold value range is from $0 \sim 1$ due to the normalized test statistics as the equation in Figure (8.8) . The specific value is chosen as 0.6 based on numerical simulation results. During the actual implementation, first multiple the threshold value with the normalization value and then compare it with the correlation result.

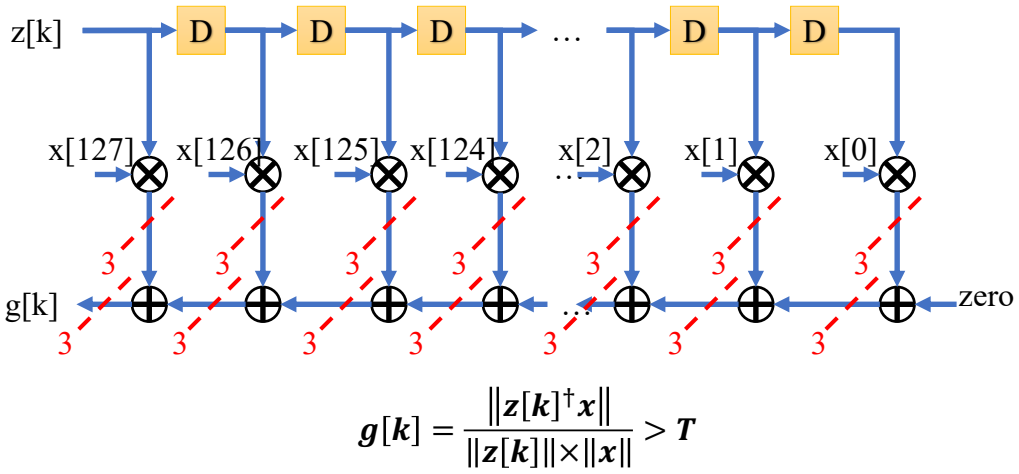


Figure 8.8: *Depiction of the CHP2 frame detector*

After the frame detection, the corresponding CHP2 frame is acquired while the coarse Rx timestamp is reported to the CHP2 programs on the PS section. This received frame is separated into two pieces: communication payload and positioning waveforms. The communication payload waveform is only collected from receiving channel one and then sent to the ZynqMP PS section for further baseband processing. The positioning waveforms are collected from all four receiving channels, and each channel contains the waveforms of all four transmitters on the other side CHP2 node. Thus, the total waveforms ready for the fine timestamp estimation are 16. In a word, the communication is using SISO mode while the range estimation is using the

4×4 MIMO mode. Later then, after applying a fine frequency offset correction, all 16 positioning waveforms are sent to the CHP2 massive correlator for the fine ToA estimation algorithm on the PL section.

8.3.3 Time-Of-Arrival Massive Correlator

The CHP2 massive-correlator is in charge of estimating fine receiving timestamp (fine ToA estimation). For finding a more precise time of arrival, it has to cross-correlates an up-sampled received waveform with an up-sampled reference waveform. Since the bandwidth of navigation waveform is 10MHz, the ranging precision only relies on the envelope, which is only 30m. If it is up-sampled by 200 times on each chip, then, the range resolution is increased to 15cm. However, a general upsampling also increases the sequence length by 200 times. In practically, it becomes an extreme difficulty for implementation. Not only the cross-correlation length is increased by a lot, but also an up-sampling operation is required.

Instead of directly cross-correlating the up-sampled waveform against the reference, we construct a match filter bank. As the Figure (8.9) shown, over-sample the original waveform by 200 times and shift it back/forward by a certain fractional sample value. Then down-sample 200 times to recover the same length of reference waveform. Go through this process multiple times with different fractional sample values to generate multiple versions of shifted reference waveform. Then assemble these waveforms as the filter bank. Each row of this filter bank is a different fractional sample-shifted version of the original reference waveform. The index of the row can represent the shifted fractional time.

By this simple method as in Figure (8.10) shown, this filter bank can be pre-constructed and loaded onto the memory. During the operation, it multiple a vector of the received positioning waveform with this reference matrix. Then the index of the output vector represents the shifted fraction time, and the value of each output vector element represents the cross-correlation result. Choose the index with the maximum magnitude value. Report this index to the CHP2 process running on the PS section

along with its phase information and then generate the fine ToA estimation.

However, this multiplication method needs to load a large filter bank coefficients onto the block memory on the PL section, which may not be realistic. Therefore, a further structure optimization is applied as Figure (8.11). Instead of building a filter bank with ± 1 chip shifting, a smaller filter bank with only $\pm 1/4$ chip shifting can be constructed. Furthermore, the received waveform can also be reconstructed as a waveform bank with fractional chip-level back-forward shifting. In this design, to achieve exactly the same 400 lengths of cross-correlation result, the received waveform bank contains the $-3/4 \sim 1$ chip shifted versions of the originally received waveform. This structure naturally is suitable for parallel computing as the operations on eight rows of received waveform bank can be executed parallelly. In this design, the received waveform bank can also be simplified as that each row is the sample delayed version of the originally received waveform.

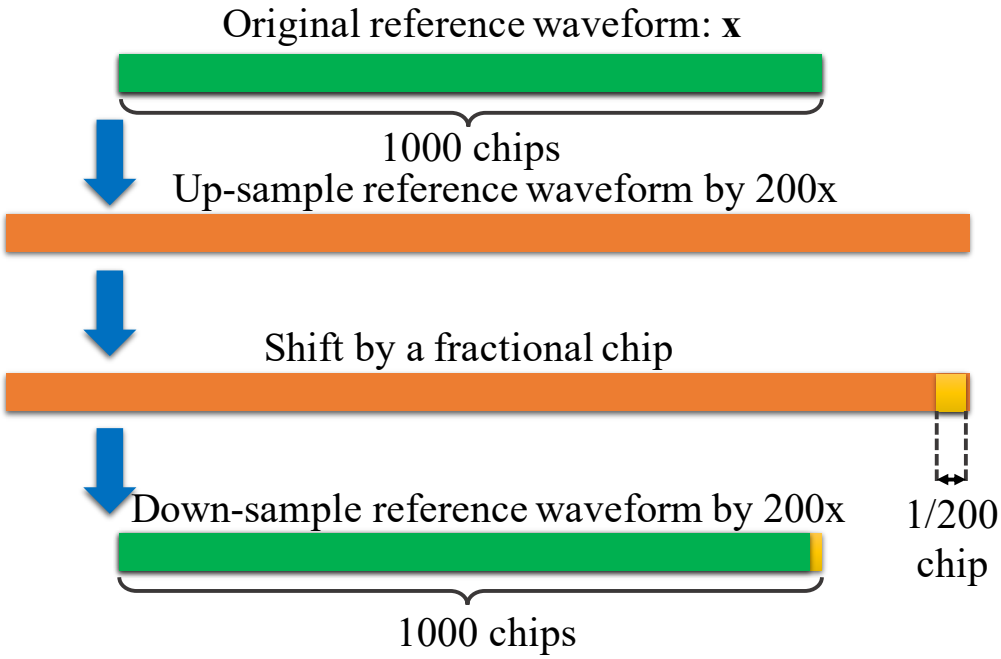


Figure 8.9: *Depiction of generating a micro-shifted reference waveform.*

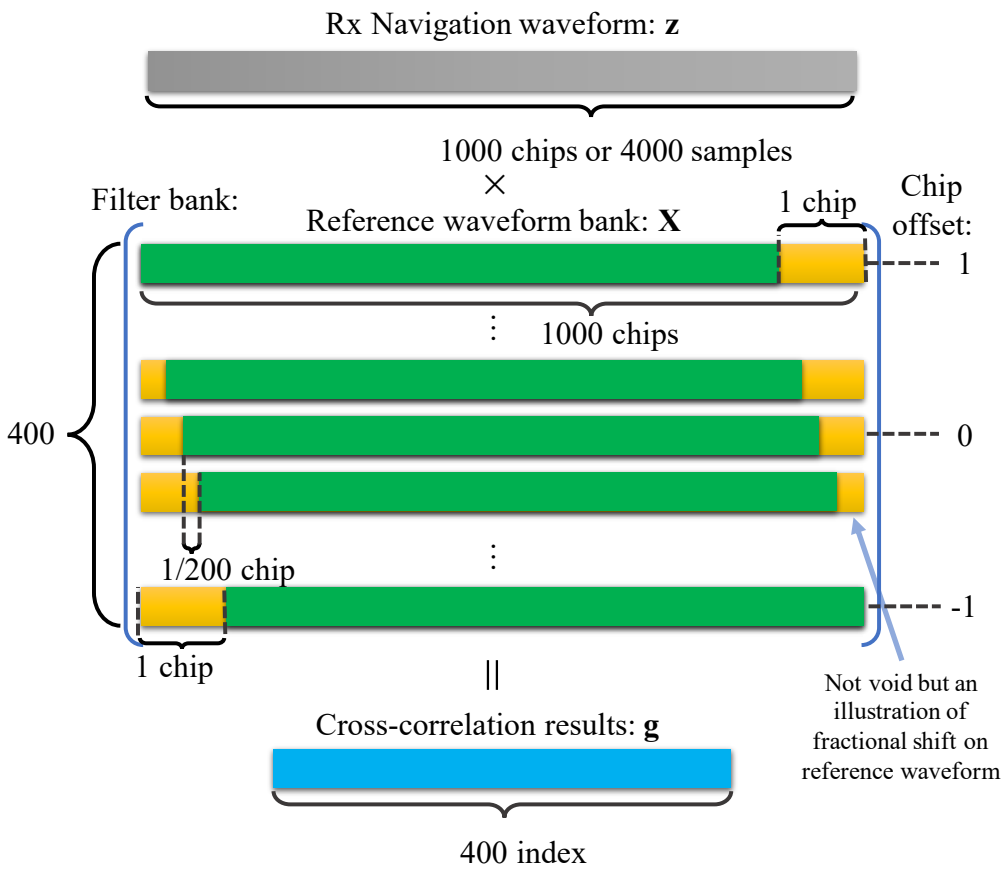


Figure 8.10: *Depiction of the CHP2 massive correlation arithmetic. Use matrix multiplication instead of an expensive match filter. Each row of filter bank is a fractional shifted version of the reference waveform. In this example, consecutive waveforms are shifted by 1/200th of a critical sample.*

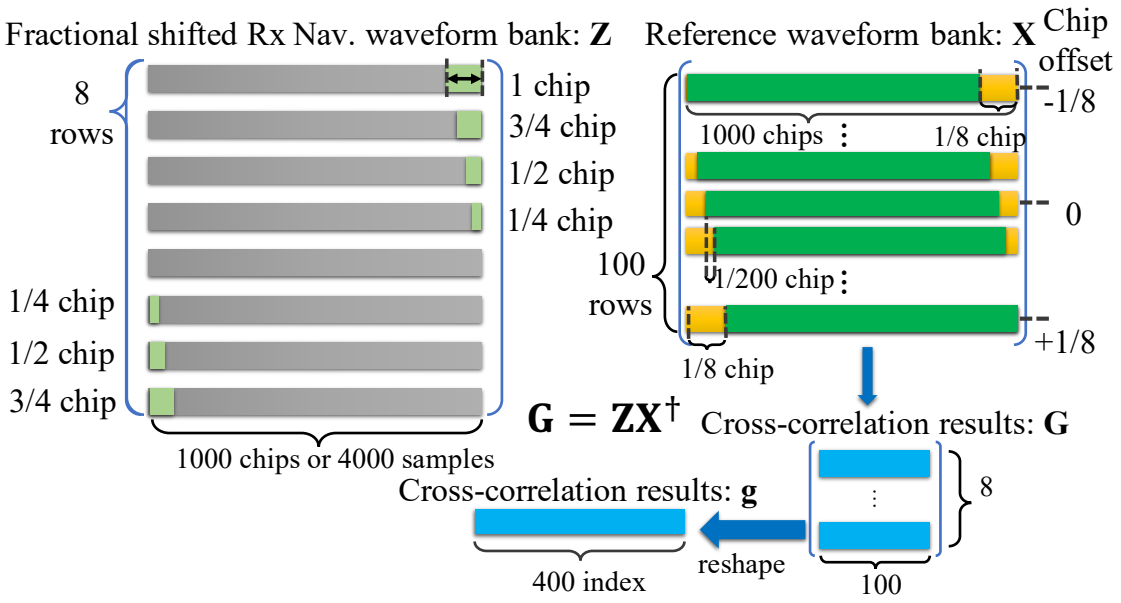


Figure 8.11: *A further optimized structure for the cross-correlation arithmetic*

Later then, this structure is implemented on the ZynqMP PL section as Figure (8.12 and 8.13). The IP block is mounted onto the AXI bus. Then all the input IQ data are allocated and controlled by CHP2 processes running on the PS section. During a massive correlation computation, the received positioning waveforms are loaded onto the input block memory. After the computation, the results are pushed onto the result block memory and mapped to the Linux userspace. The result is an array of 400 complex numbers. Each index corresponds to each 1/400 chip fractional time-shifting. All 400 index covers -1 to 1 chip shifting range, which corresponds -4 to 4 samples shifting range. The index of the maximum magnitude value among those 400 indexes is used for calculating the fine timestamp.

Furthermore, a polynomial fitting is applied to narrow down the exact fine timestamp further. This is a second-order polynomial regression. Once the index of the element with maximum magnitude on this cross-correlation result is declared, we use the adjacent $\pm$ 7 index values of magnitude around this peak to construct a second-order regression model as following Equations (8.1, 8.2, 8.3). Once the polynomial vector $\hat{\boldsymbol{\beta}}$ are calculate, the new peak index can be expressed as $\hat{k}_r = -\beta_1/(2\beta_0)$.

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \epsilon. \tag{8.1}$$

$$\begin{bmatrix} y_{-7} \\ y_{-6} \\ \vdots \\ y_0 \\ \vdots \\ y_6 \\ y_7 \end{bmatrix} = \begin{bmatrix} 1 & x_{-7} & x_{-7}^2 \\ 1 & x_{-6} & x_{-6}^2 \\ \vdots & & \\ 1 & x_0 & x_0^2 \\ \vdots & & \\ 1 & x_6 & x_6^2 \\ 1 & x_7 & x_7^2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} \epsilon_{-7} \\ \epsilon_{-6} \\ \vdots \\ \epsilon_0 \\ \vdots \\ \epsilon_6 \\ \epsilon_7 \end{bmatrix} \tag{8.2}$$

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}^\dagger \mathbf{X}\right)^{-1} \mathbf{X}^\dagger \mathbf{y} \tag{8.3}$$

This second-order regression is implemented as part of the CHP2 process program in the PS section. The challenge is that there is 16 total of second-order fittings required for finer massive cross-correlation peak finding. They have to be executed parallelly over multi ARM cores to save processing time.

Finally, the fine timestamp can be label as $\hat{\tau}'_f = \hat{k}_c f_{s,f}^{-1} + \left(\hat{k}_f + \delta k_f\right) f_{s,est}^{-1}$, where $\delta k_f = \hat{k}_r - \hat{k}_f$ and $\hat{k}_f = \arg\max_{k_f} \mathbf{g}[k_f]$ is from Equation (3.9). $\hat{k}_c$ is the coarse receiving timestamp. $f_{s,f}$ is the system sample rate which is 40MHz in current design. $f_{s,est}$ is the sub-sample rate which is 2GHz.

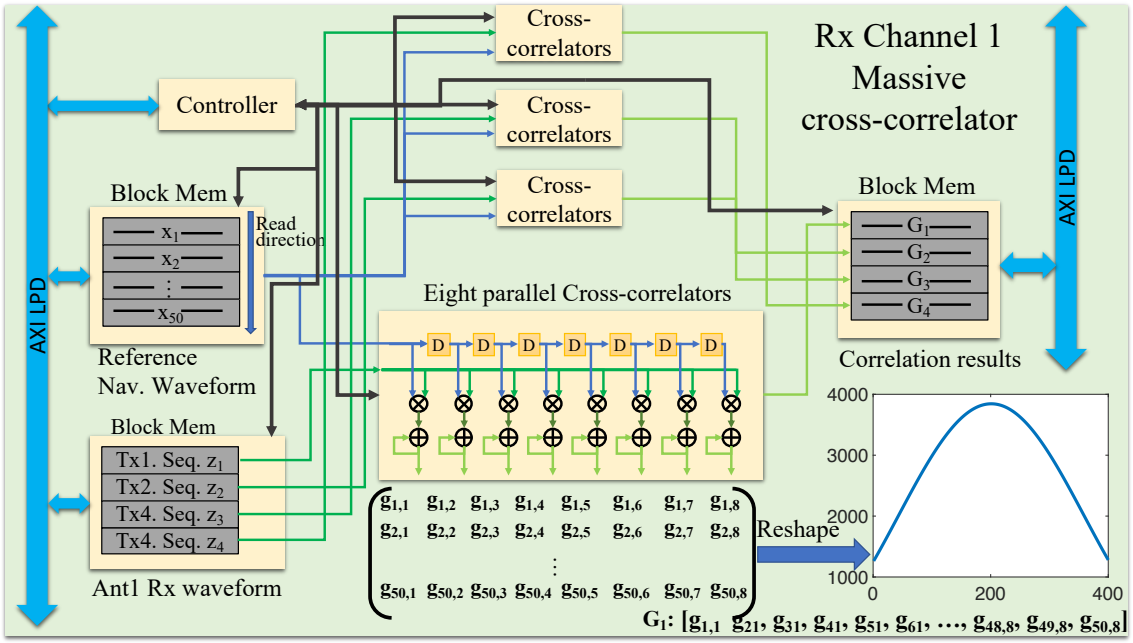


Figure 8.12: *The implementation of the CHP2 massive correlator on a single receiving channel. Each receiving chain has its own dedicated four correlators*

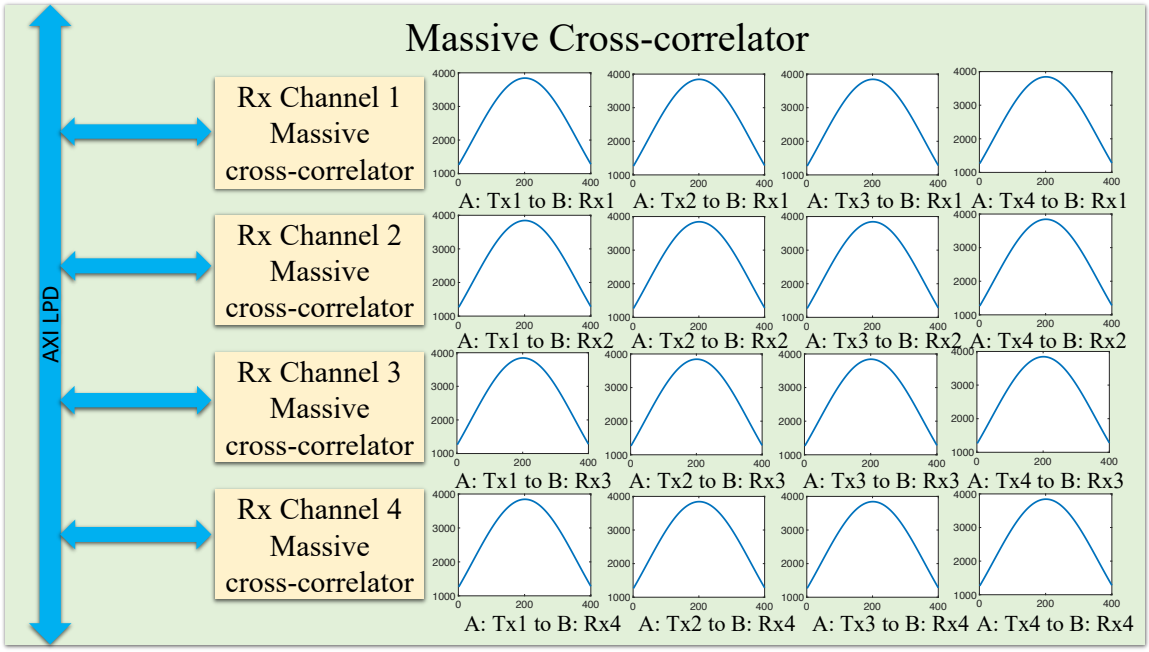


Figure 8.13: *The implementation of massive correlator for all four receiver channels.*

8.3.4 Optimize the Massive Correlation Accelerator

The massive correlation is the most critical component of the CHP2 system processing chain. The fine ToA estimation algorithm heavily depends on this IP block. Therefore, accelerating this IP benefits the whole system performance a lot. We optimize it in two stages. The first stage is pipelining its complex number arithmetic. Secondly, add DMA (direct memory access) IP block to accelerate its data transfer.

The massive correlation is equivalent to apply the array dot production between the reference waveform and the input waveform. The essential arithmetics are the complex number multiplication and the accumulation. The initial design of the method, shown as Figure (8.14) and Figure (8.12), requires this complex number operation to finish with one cycle. The long critical path causes a lower clock rate. Thus, we redesign this operation and reduce the number of the multiplier as Figure (8.15). Also, we add a three stages pipeline to this operation. It helps to increase the maximum operational clock rate. The number of the multiplier is reduced to three per operation unit with the cost of adding two extra adders. A multiplier costs more sources than two adders. So, the whole logical source is still reduced. Furthermore, we double the cross-correlation length from the initial 400 test bins to the current 800 test bins, where the euqivlent range cell size is decreased from 15cm to 7.5cm.

The second stage optimization is adding the DMA block to interface with the accelerator IP with the system memory. The initial design of the massive correlator uses the AXI4 lite bus through the memory mapping function. This protocol is only suitable for IP core's registers reading or writing but not for high throughput data IO operation. Each IO operation on the AXI4 lite bus requires individual handshaking, and cannot provide the burst transferring mode. Comparing to the computation time consumption, moving the data costs a significant amount of time. Therefore,

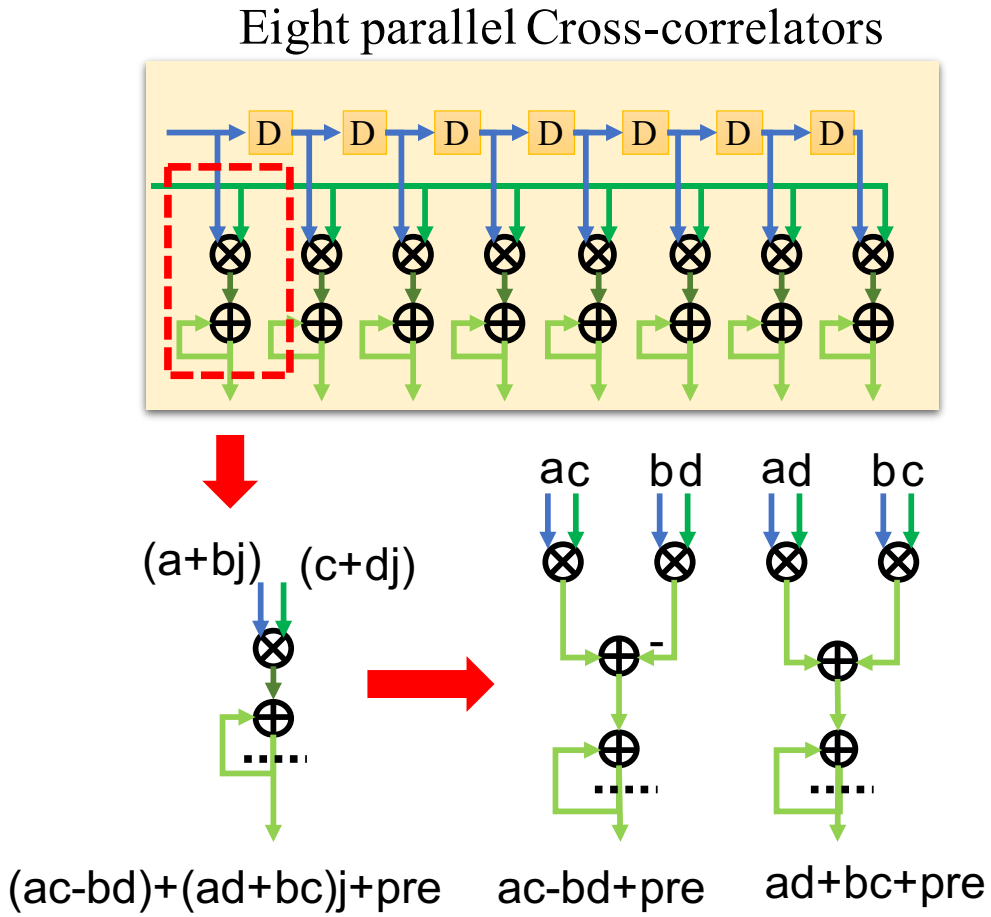


Figure 8.14: *Depiction of the initial design of massive correlation arithmetic. The complex numbers multiplication and accumulation operation is constraint to finish within one clock cycle. It consumes four multipliers*

the DMA is the only choice to overcome this drawback. It can feed the input data onto the massive correlator or moves the correlation results to the system memory in a customizable burst length. Once the initial handshaking established between the IP core and the DMA, the data can flow through the data bus continuously. It is much faster than the AXI4 lite bus IO operation. Since the provided Xilinx AXI DMA IP requires interfacing the user IP with the AXI Stream protocol, thus, we

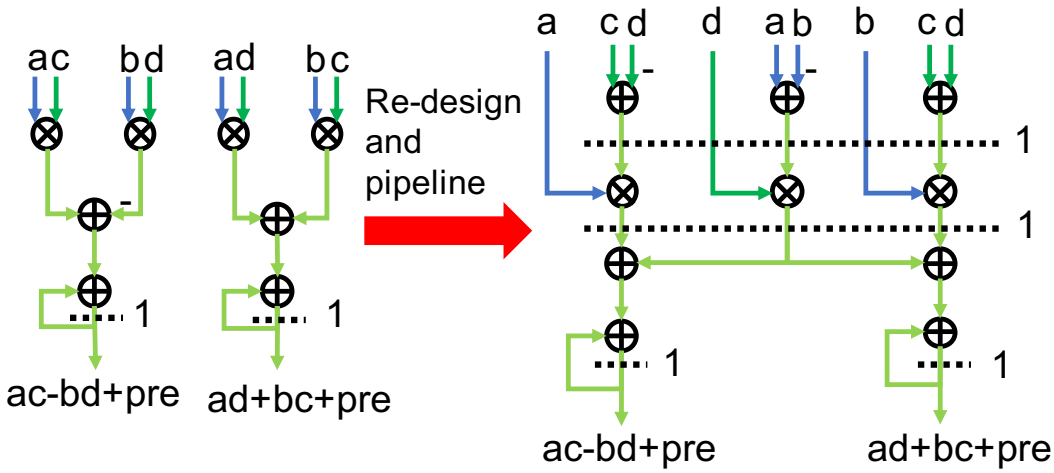


Figure 8.15: *Depiction of the optimized design of massive correlation arithmetic. The complex numbers multiplication and accumulation operation are reduced to use three multipliers. Adding three stages pipeline can easily double or triple the clock rate.*

preserve the AXI4 lite bus interface of the massive correlator and add extra AXI4 Stream interfaces. The AXI4 lite bus still serves the purpose for register control and monitoring while the AXI4 stream interface only serves the data transferring. We customized an AXI4 Stream wrapper connecting to the IP core's local BRAM, which serves as the data exchanging buffer. The whole design diagram is shown as Figure (8.16). The IP cores used is the Xilinx AXI DMA IP. The CHP2 program accesses the DMA through the Linux kernel DMA engine, which is a standard DMA API interfacing with the Xilinx AXI DMA kernel driver. It provides the automatic data coherence feature and hides all the low-level registers setup, which is easier to use compared with some third party userspace DMA driver.

Finally, the optimized massive correlation accelerator achieves a 200MHz clock rate after the new synthesized firmware is implemented on the CHP2 processing board. The computation consumption time is approaching the theoretical minimum

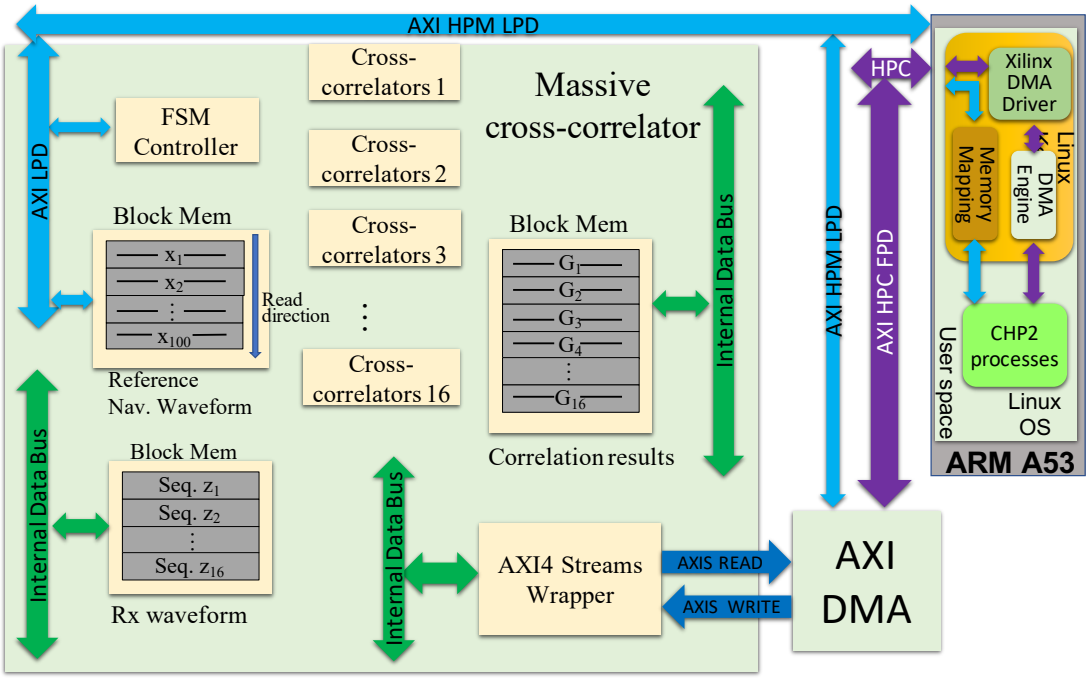


Figure 8.16: *Depiction of the AXI DMA interfacing the massive correlator through AXI Stream wrapper. The DMA then interfaces to the ARM core through AXI4 full bus. The internal connections within massive correlator are hidden on the digram.*

value, which is $4000 \times 100 \times 5\text{ns} = 2\text{ms}$. The data transferring rate is also closing to the theatrical maximum data throughput bandwidth, which is $200\text{MHz} \times 4\text{bytes} = 763\text{MBytes}$. The total cross-correlation time consumption for processing a single CHP2 frame is reduced to 4 ms. It saves considerable time compared to the initial design, as shown in Table (8.1).

Table 8.1: *The CHP2 Massive correlation performance before and after optimization.*

Item	Before	After
Cross correlation length	400	800
Minimum range cell size	15cm	7.5cm
Equivalent sampling rate	2GHz	4GHz
Data clock rate	100MHz	200MHz
Load input data bandwidth	76MBytes	667MBytes
Load result bandwidth	17MBytes	667MBytes
Computation time	2.03ms	2.14ms

8.3.5 Channel Equalizer

The channel equalizer is applied for the receiving chain for extracting communication payload. It is nothing related to navigation waveform signal processing but only removes the physical propagation channel effect on communication payload. This is an adaptive filter using least square solution as Equation (8.4).

$$\begin{aligned} \mathbf{y} &= \mathbf{w}^{\dagger}\mathbf{z}, \\ \mathbf{w} &= \mathbf{R}^{-1}\left(\mathbf{Z}^{\dagger}\mathbf{s}\right), \end{aligned} \tag{8.4}$$

where,

$\mathbf{w} : n \times 1$ the filter taps weights,

$\mathbf{z} : n \times 1$ the received signal stream,

$\mathbf{R} : m \times m$ the received signal covariance matrix,

$\mathbf{Z} : n \times m$ the received signal tap-input,

$\mathbf{s} : n \times 1$ the reference preamble waveform

In this particular design, it assumes the multi-path on propagation channel results in a convolution channel. And the channel impulse response is assumed to be five taps. The distance between taps is two samples. Then the received signal tap-input $\mathbf{Z}$ and

the covariance matrix $\mathbf{R}$ can be constructed as follows:

$$\mathbf{Z} = \begin{bmatrix} z_4 & z_2 & z_0 & z_{-2} & z_{-4} \\ z_5 & z_3 & z_1 & z_{-1} & z_{-3} \\ z_6 & z_4 & z_2 & z_0 & z_{-2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ z_{n+3} & z_{n+1} & z_{n-1} & z_{n-3} & z_{n-5} \end{bmatrix},$$

$$\begin{aligned} \mathbf{R} &= \mathbb{E}\left(\mathbf{Z}^{\dagger}\mathbf{Z}\right) \\ &\simeq \frac{1}{n}\mathbf{Z}^{\dagger}\mathbf{Z}. \end{aligned} \tag{8.5}$$

Each column of this matrix $\mathbf{Z}$ is a sample shifted version of the received preamble waveform. The center column should be perfectly aligned with the complete preamble waveform. During the training stage, bring (8.5) to Equation (8.4) and compute the adaptive filter coefficient $\mathbf{w}$. Then apply this filter to the remaining of the communication payload. Each coming new CHP2 frame refreshes the filter coefficients. After the equalization, the output IQ data are ready for decoding. The procedure of extracting communication payload content is the reverse order of Figure (6.6).

The channel equalization is implemented in the PS section as part of the CHP2 physical layer signal processing program. The challenge of this realization is to solve the matrix inverse and accelerate the computation time.

8.3.6 RF Power Control

As one of the CHP2 system design tasks, it is required to establish a long-range stable communication link between the ground node and the remote node. Also, maintaining enough precision of ranging without saturation requires a proper receiving SNR for both nodes over a couple of kilometers distance in actual application. Therefore, an RF power controlling mechanism has to be implemented for the CHP2 system.

Currently, the physical hardware has already provided many options to tune RF signal parameters dynamically, including transmission and receiving gains. The Analog Device AD9361 transceiver offers up to 75 dB effective receiving gain and up to 90 dB transmission attenuation around 900MHz center frequency. Also, its noise figure is as lower as about 2dB to 3 dB, which guarantees an extreme sensitivity in a limited receiving power scenario.

Except for this AD9361 transceiver, the RF front end TR switch card in the CHP2 system provides an extra 30 dB transmission gain and up to 1W output RF power by an external power amplifier(PA). Meanwhile, It utilizes an on-board low noise amplifier (LNA) for an additional 20 dB receiving gain.

According to the previous Section (6.3.1), in a scenario of two CHP2 nodes separated by a 10km range, the receiver SNR is about 20dB when the other one transmits its maximum power 30dBm. Based on some empirical knowledge, the SNR around 20dB$\sim$25dB prevents saturating the massive correlator computation. Therefore, for the current design, within a 10km range, it requires an automatic gain controlling (AGC).

This feedback control consists of four components. The first component is the receiving SNR estimation. There is a total of 16 channels SNR estimates because of

the MIMO feature of the CHP2 system. For a single tx stream, we use the maximum SNR value among these four receiving antennas to indicate the signal strength of itself. This SNR represents the transmission power of the corresponding antenna on the other node. The second one is the decision-making logic. This function keeps monitoring the receiving SNR value of the corresponding tx stream and comparing it with the predefined threshold. If the SNR is lower than the threshold, the decision-making logic packs a "transmission power increase" command onto the communication payload. If the SNR is higher than the threshold, the decision-making logic packs a "transmission power decrease" command onto the communication payload. If the SNR is close to the threshold within a specific delta value, the decision-making logic sends a "hold on" command to the other node. It helps to avoid a potential oscillation on SNR values. After the decision-making, the power control word is received by the other node. And then the third component: the gain value generator starts. It generates the specified transmission gain value based on the current node TX RF layer parameters and the received power control command from the other node. The fourth one is the AD9361 transceiver driver interface. It directly tunes the analog RF parameters based on those values from the gain value generator.

A illustration of power control loop is shown in Figure 8.17. It only shows one feedback loop from one node to the other one. However, the actual two nodes have their feedback loop individually and simultaneously. An example of the receiving SNR versus time is shown as Figure (8.18) when the AGC is targeting 25 dB SNR value.

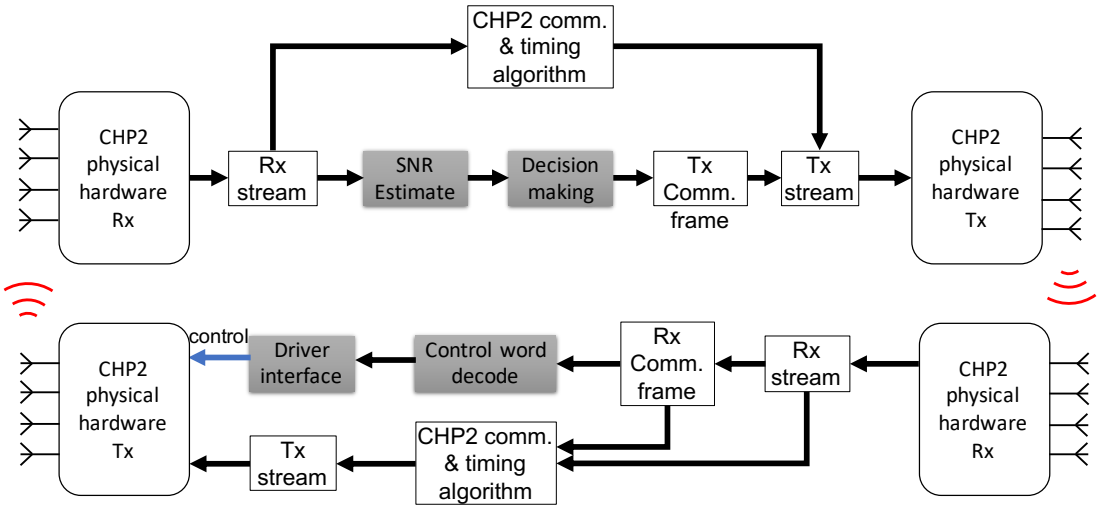


Figure 8.17: *Depiction of a power control loop from one CHP2 node to the other one. On a node, the Rx and Tx antennas are actually the same. They will be switched between Rx and Tx mode.*

Algorithm 1 CHP2 Power Control Algorithm

1: Estimate all 16 channels' SNR values of one CHP2 node.

2: Pick up the maximum SNR value among four receiving channels for each signal stream as the transmission power indicator.

3: Compare this SNR with the threshold to determine the power control word.

4: Pack control word onto the CHP2 communication frame and then send it out to the other CHP2 node.

5: Receive on the other CHP2 node and decode the control word.

6: Generate the transmission gain value based on the power control word and current gain value and value range.

7: Write the transmission gain value onto the hardware RF analog chain.

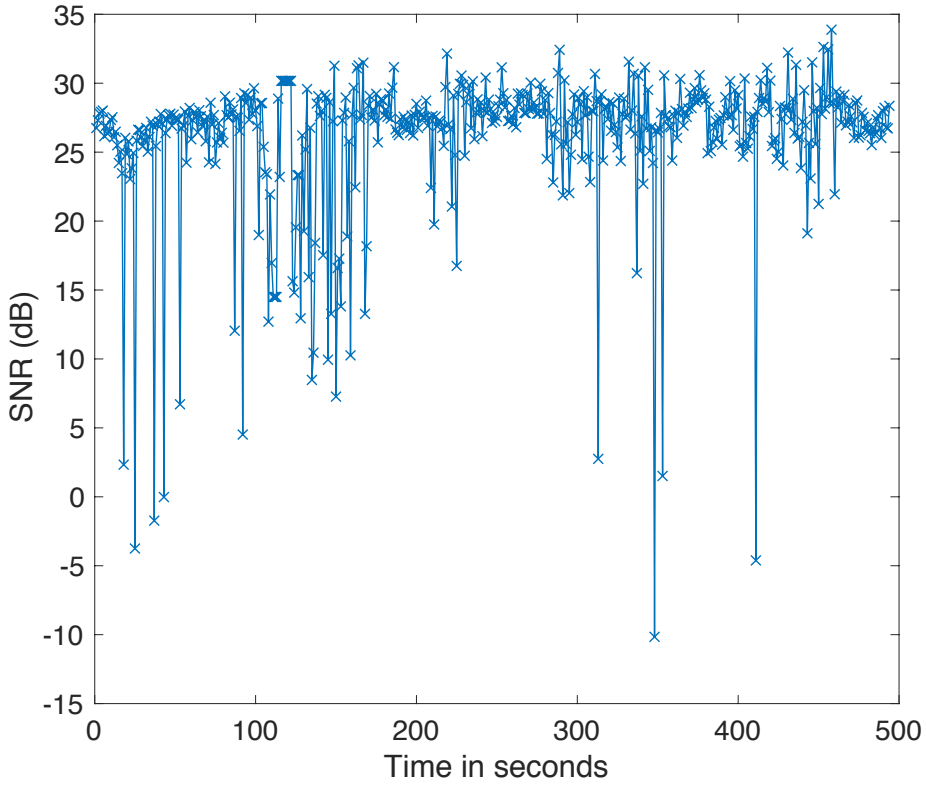
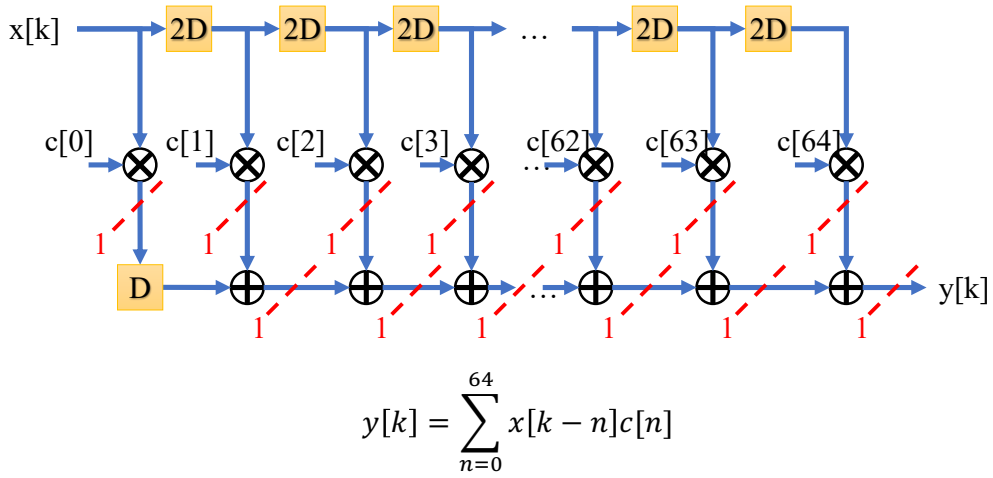


Figure 8.18: *Depiction of the relation between SNR (from remote node antenna 1 to ground node antenna 1) and time when the power control loop is applied. The two nodes sit on two pushcarts, which are separated by about 200 meters. The remote node moves toward the ground node at a constant velocity. The target SNR is 25dB. It clearly shows the power control loop maintains the SNR around 25 dB. However, some artifacts also distribute the SNR estimation*

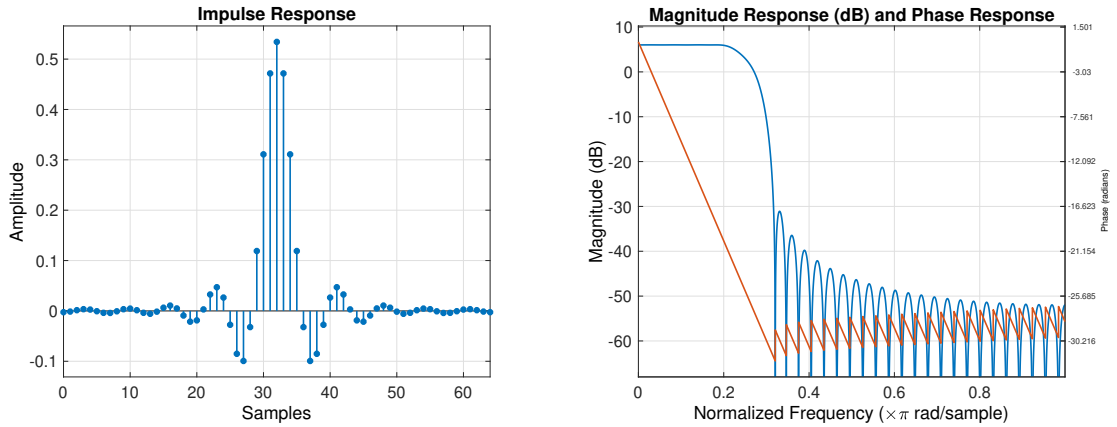
8.3.7 Pulse Shaping filter

To constrain the signal bandwidth within 10MHz and decrease the inter-symbol interference, we design eight FIR filters based on Raised-cosine, as implemented in Figure (8.19). With a 0.25 roll-off factor, 16 symbols span and 4 samples per symbol, a filter with 65 coefficients is designed as the impulse response in Figure (8.20a). Then the filter is mounted onto each transmission and reception chain inside Tx Engine and Rx Engine. The filter input and output port are reserved as a signed 16 bits fix-point format to guarantee the IQ data format compatibility.



$$y[k] = \sum_{n=0}^{64} x[k-n]c[n]$$

Figure 8.19: *Depiction of the 65 taps FIR filter structure and its timing relation.*



(a) *the impulse response* **(b)** *the frequency and phase response*

Figure 8.20: *Depiction of the 65 taps FIR filter performance. The wide opened eye shows less inter-symbol interference.*

8.3.8 Frequency Offset Correction

There are two frequency corrections on the CHP2 system and applied to different sets of incoming IQ data. The first one is a coarse frequency offset correction applied to the raw IQ data for decoding communication payload. Although the adaptive equalizer has a similar feature, this correction increases the accuracy of equalized data. The second one is only used for the navigation waveform IQ data before the massive cross-correlation. It prevents the large frequency offset from corrupting the cross-correlation curve. As the Section (6.3.4) and Figure (6.3), the preamble, midamble and postamble waveforms should be used to estimate each individual phases $(\hat{\phi}_{pre}, \hat{\phi}_{mid}, \hat{\phi}_{pos})$.

Assume the reference waveform is $\mathbf{x} : n \times 1$ and the received reference waveform is $\mathbf{y} : n \times 1$, then

$$\hat{\phi} = \frac{\text{imag}\left(\mathbf{x}^{\dagger}\mathbf{y}\right)}{\text{real}\left(\mathbf{x}^{\dagger}\mathbf{y}\right)}. \tag{8.6}$$

The coarse frequency offset estimate is

$$\hat{f}_1 = \frac{\hat{\phi}_{mid} - \hat{\phi}_{pre}}{2\pi T_{mid-pre}}. \tag{8.7}$$

The fine frequency offset estimate is

$$\hat{f}_2 = \frac{\hat{\phi}_{pos} - \hat{\phi}_{pre} \pm n}{2\pi T_{pos-pre}},\ n = \arg\min(\hat{f}_2 - \hat{f}_1), \tag{8.8}$$

The correction can be described as following

$$z_{\text{corrected}}[k] = z[k] \cdot e^{\frac{2\pi \hat{f}_{\text{offset}} k}{f_s}}. \tag{8.9}$$

where $z[k]$ is the received raw IQ data, k is the discrete time index and f_s is the sampling frequency.

The above phase estimation and frequency offset correction are implemented onto C language as part of the CHP2 process program running on the PS section.

8.3.9 Communication Payload Encoding and Decoding

The communication payload encoding and decoding are implemented as part of the CHP2 process program running on the PS section. The details of implementation is exactly following Section (6.3.1). The payload decoding is a reverse procedure of the encoding. The main difference is that trellis decoding is used to decode the convolution code.

8.3.10 Transmission Reception Switching Controller

As the previous chapter mentioned, the CHP2 is running on TDD mode, which needs cycling though transmission and reception. A controller, as in Figure (8.5) in the PL section, is built to dedicate controlling the switching action. The procedure is based on the CHP2 primary timer and transmission timestamp. The TR switching controller keeps monitoring the primary timer and then switches the mode to transmission earlier by a fixed amount of micro-seconds before the tx time indicated by the transmission timestamp. After the transmission is finished, the controller switches the mode back to reception.

8.4 CHP2 Data Link Layer

The data link layer is responsible for scheduling the frame transmission and reception. It also schedules the extracting from a frame or packaging timestamps onto a frame. As the following the scheduling diagram, on a CHP2 node, each transmission happens every 100ms. Before each transmission, the reception process has to be finished within 50ms as Figure 8.21. Thus, the schedule is crucial. Meanwhile, the scheduler also controls the TRS amplifier board for transmission and reception mode switching.

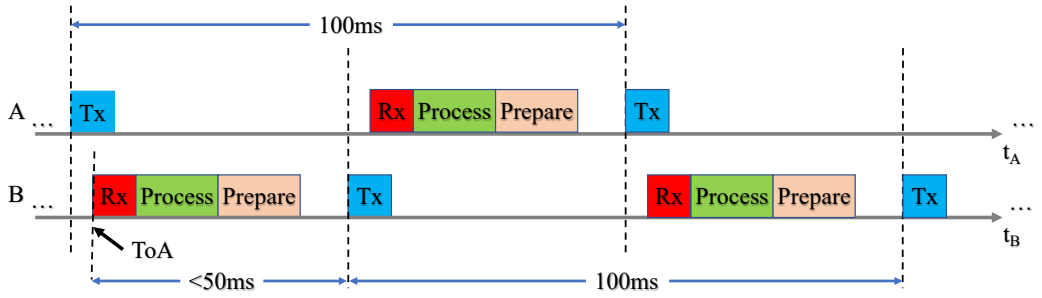


Figure 8.21: *Depiction of the data link layer for two users. Users alternate transmitting and receiving in each frame. The receiver estimates the ToA and schedules the transmit event. Both of the timestamps are packaged into the next payload and transmitted on the next frame.*

Another import component on the link layer is the Ethernet logger program. It collects all the ToF estimates from the application layer and other debugging information from the physical layer and then sends to the external system for telemetry and remote monitoring purpose. The Ethernet logger interfaces with the external systems. At a rate of 10Hz, coming out of the CHP2 system, an Ethernet frame contains all the range estimates, the estimated coordinates, the receiver channel SNR value, and other necessary information for system monitoring and debugging.

8.5 CHP2 Application Layer

This layer contains several of Time-of-flight estimation techniques. They are the essence of this ranging-positioning system. By the timestamps provides by the lower layers, the ToF algorithms estimate the distance, clock frequency offset, and time offset between two CHP2 nodes. Also, the geometry conversion techniques are implemented. It helps to convert the 4×4 MIMO range estimates of the remote node onto the geometry coordinates and the orientation information. It is discussed in the later chapters.

8.6 ZynqMP Implementation Result

The whole processing chain and system layers are built onto ZynqMP XCZU9EG. Most of mathematics heavy loading components are implemented onto the PL section. The remaining are implemented as low-level program and high-level application on the PS section. However, ZynqMP implementation is similar to traditional FPGA. The design can be evaluated by the Xilinx Vivado EDA tool.

Table (8.2) shows the specifications of programmable resource consumption within ZCU102. The total LUT resource is used up to 55%, and DSP blocks are used up to 77%. The remaining space may be utilized for future extension. The implementation satisfies the timing constraint as shown on Table (8.3). The Tx engine, Rx engine, and the FIR filters are inserted onto the direct IQ data path from the transceiver AD9361. Therefore, the data clock rate of the above IP cores is variable, but the sample as the IQ data sampling rate 40MHz, where the upper limit is up to 56MHz. The massive correlator is working on a fixed 200MHz clock domain, which is limited by the maximum rate on internal data buffers to the AXI DMA IP. But the cross-correlation arithmetic can smoothly run beyond 300MHz because we implement it by the DSP48E2 blocks instead of the pure lookup tables.

Table 8.2: *CHP2 Logical resource utilization on ZynqMP XCZU9EG*

Resource	Utilization	Available	%
LUT	149779	274080	54.65
LUTRAM	5135	144000	3.57
FF	153109	548160	27.93
BRAM	705	912	77.30
DSP	1928	2520	76.51
IO	143	328	43.60
GT	1	24	4.17
BUFG	26	404	6.44

Table 8.3: *The data clock rate of the major IP components within the CHP2 system*

IP(with quantity)	Clock Frequency	Input Date Width
Tx Engine × 1	40MHz	128bits
Rx Engine × 1	40MHz	128bits
FIR Filter × 8	40MHz	32bits
Massive Correlator × 1	200MHz	32bits

8.7 CHP2 Processing Acceleration

In a typical two nodes setup scenario, the CHP2 system requires a two-way communication within 100ms. That means the receiving frame processing and the transmission frame preparation has to be finished within 50ms before the next transmission. And there are a total of 208000 complex numbers of 12 bits fix-point format generated per frame at the 40MHz sampling rate. They have to be correctly processed and generate the estimated ToA timestamps. The correct timing is crucial for the system performance Furthermore, no only the heavy load computations, for example, the frame detector and massive cross-correlation, are migrated to HDL coded IP accelerators, but also the raw data transferring among the CPU memory and these IPs should be accelerated.

Considering the architecture of ZynqMP SoC, there is a multi-core ARM A53 processor. It is not that faster as the X86 CPU on the regular software-defined radio platform, for example, the USRP series. In conclusion, much effort in accelerating the processing speed on CHP2 has to be made to satisfy the design requirements. There are a couple of optimization strategies used in this CHP2 system processing chain.

8.7.1 Coding with a SIMD style

Some of the CHP2 processing programs have the computation involving a large number of IQ, and the arithmetic behind are the vector operations. For example, phase estimation, frequency offset correction, PSK modulation, code-spreading are all operated on a vector of IQ data with similar arithmetic inside an iteration loop. It implies a SIMD feature should help to accelerate the computation somehow.

As one of the modern CPU, the ARM A53 core is using armv8 architecture, which provides NEON Microarchitecture for implementing SIMD (single instruction multiple data) feature. Initially, implementing SIMD on ARM requires coding on assembly language using architecture specified instructions or coding on high-level language but using intrinsics functions. Those methods are still widely used and efficient. However, they are still not trivial and require much experience. However, thank compiler technology evolving, the current majority of compilers, for example, GCC 4.6(or newer), are featuring automatic vectorization a for loop on high-level languages like C/C++. It enables high-level parallelism with few compiler parameters tuning and a specified coding style [59].

8.7.2 Use OpenMP Library

After the above optimization, the total cost of CHP2 processing operations is within around 15ms. However, the raw IQ data moving speed becomes the bottleneck. The original Analog Device released SDR HDL section uses DMAs to move IQ data between the AD9361 transceiver controller and Linux OS kernel memory. However, the Analog Device driver interfaces and the current applications are all implemented over the Linux userspace. In this case, it requires one extra copy operation between kernel space and userspace, which consumes plenty of CPU time. To avoid the high risk of touching any Linux kernel drivers and device tree, we decide to parallelize this memory copy operation. Let this load is shared by all four A53 cores instead of a single core initially. It turns out that using the OpenMP library can achieve excellent performance while maintaining a minimum modification of C code. The initial memory copying costs about 30ms. The OpenMP based memory copying costs only less than 7ms.

Meanwhile, there are a couple of matrix-decomposition based matrix inverse operation involving channel equalization, polynomial regression, and EKE filter. Using OpenMP to parallelize those operations saves a lot of CPU time. Finally, the total processing time consumption is reduced to about 30ms. It is much less than the limit of 50ms.

8.8 Dynamic Range Analysis

As a radio system with limited computation resource and physical hardware fundamental properties, the CHP2 system discretely samples the RF signal by the ADCs. It uses finite precision number representation through the whole signal processing chain. In reality, the finite precision number can only represent a specific range of values. Any input above the maximum allowed amount can saturate the corresponding finite precision algorithm. Vice-verse, any input below the minimum value can cause starving. The ratio of this maximum over the minimum amount is defined as the dynamic range. Indeed, when designing the finite precision algorithm, we consider all possible input values and decide a proper dynamic range. But the initial input finite precision number usually comes from the ADC. Consequently, the behavior of the ADC and the finite precision arithmetic could potentially impact the system's whole performance dramatically. Therefore, it is crucial to analyze the dynamic range of the system to characterize the performance. Meanwhile, it helps to find the system limitation and possible design faults.

In this section, we analyze the dynamic range of the AD9361 transceiver ADCs and the massive-correlator as the core component of the CHP2 system. It includes both theoretical and practical dynamic range analysis. The remaining finite precision computation components are also constrained by their dynamic ranges but not the major factors in determining the system performance.

8.8.1 Dynamic Range of ADC on AD9361 Transceiver

The AD9361 provides two receiving and two transmitting channels. Each of them contains an ADC or DAC in the complex domain to sample or generate the analog signal. The transmitting channel is not analyzed because the transmission chain is not complicated. The CHP2 default pushes the finite precision number to the full scale on transmission DACs. The receiving chain ADC dynamic range is more interested because it is directly related to the receiving baseband processing performance.

The receiving channel uses a 12 bits sigma-delta ADC covering from less than 200 kHz to 56 MHz sampling rate. Assume the uniform distributed quantization error is the only noise source, then divide the ADC full-scale magnitude square by the variance of the quantization error results in an expression for an ideal converter.

$$\begin{aligned} \text{SNR} &= (1.763 + 6.02N) \\ &= 1.763 + 6.02 \times 12 \\ &= 76 \text{ dB} \end{aligned} \tag{8.10}$$

where N is the number of ADC bits. Thus, we can find an ideal dynamic range or SNR of this ADC is about 74 dB.

However, this is not a practical case. The thermal noise and clock jitter also contribute to the error on ADC. The extra receiving gain will boost the thermal noise. From Equation (6.5), we can find the thermal noise on ADC at a 10MHz bandwidth is about -101 dBm or -171 dBm/Hz. For the sake of simplification, we drop the error caused by clock jitter. Meanwhile, the current system setting is adding about 52 dB receiving gain, and we apply four times complex sampling over the 10MHz bandwidth (every twice the Nyquist band oversampling lowers the noise floor by 3 dB). So, the thermal noise is increased to $-101\text{dBm} - 6\text{dB} + 52\text{dB} = -55\text{dBm}$. From the AD9361 datasheet[56, 60], its full range scale peak voltage $V_p = 0.625v$. The RF signal input

impedance is about 100Ω. We can imply the maximum full scale signal power level is

$$\begin{aligned} P_{max} &= \frac{V_{rms}^2}{Z_{in}} \\ &= 10 \times log\frac{(0.625 \times 0.707)^2}{100} + 30 \text{ dB} \\ &= 2.9 \text{ dBm}. \end{aligned} \tag{8.11}$$

We assume input power is 0.5 dB below full-scale. Then we can calculate a effective SNR as

$$\text{SNR} = 2.9 \text{ dBm} - 0.5 \text{ dBFS} - (-55 \text{ dBm}) = 57.4 \text{ dB}. \tag{8.12}$$

If this inference is corrected, assuming there is no clock jitter, the actual effective ADC dynamic range is about 57 dB over a 10 MHz bandwidth, which is also equivalent to the maximum allowed receiving signal SNR without saturation the ADC. However, there is not further released information from the AD9361 manufacturer except the existed datasheet. The conclusion cannot be verified but only through the actual experiment. This is beyond this report content.

Furthermore, if we set the receiving gain to 0 dB, the effective ADC dynamic range seems to increase to $2.4 - (-101) = 103.4$ dB. But this is not true. The AD9361 ADC internally over-samples the analog signal at the much faster sampling rate. Then it combines the samples to achieve the effective sampling rate, which is from 200kHz to 56MHz. The oversampling technique drops the thermal noise density, but the dominated error source becomes the quantization error at this point.

8.8.2 Dynamic Range of Massive Correlator

The massive correlator serves two purposes for the whole system. The first is to find the fine ToA timestamps, which is crucial to the ToF estimation algorithm. The second is to estimate a super fine receiving phase of the navigation waveform. That can help refine the ToF estimates and potentially lever the range precision to the millimeter level. Therefore, a high dynamic range on its correlation result guarantees the precision of these two major services.

The input IQ data of the massive correlator is passed from ADC through the FIR filter and frequency correction. In the receiving chain design specification, the intermediate processes do not change or slightly change the dynamic range of the IQ. So we do not count their effects. The maximum value of the input to the massive correlation can be as large as up to the full scale of 12 bits finite precision number. Assume there is a received waveform with infinite SNR (noise is zero) and using the full scale of 12 bits finite precision number, perfectly aligned with the massive correlation reference waveform (which is also the full scale of 12 bits fixed-point number in design). According to the design specification, the massive correlation is equivalent to perform 4000 samples long array dot production, which will increase the output value magnitude about $10 \times log(4000) = 36$ dB compared to the input data magnitude as the figure (8.22). This assumes the input waveform uses the 12 bits fix-point format with a magnitude of 1 on each sample. The cross-correlation will give the maximum results of less than 4000. This 36 dB is the upper limit of the dynamic range.

Ideally, if the quantization error is the only noise source. Then input the quantization noise will set the lower limit of the dynamic range on the cross-correlation results. In this scenario, the quantization error is a uniform distribution as $U(-\frac{1}{\Delta}, \frac{1}{\Delta})$,

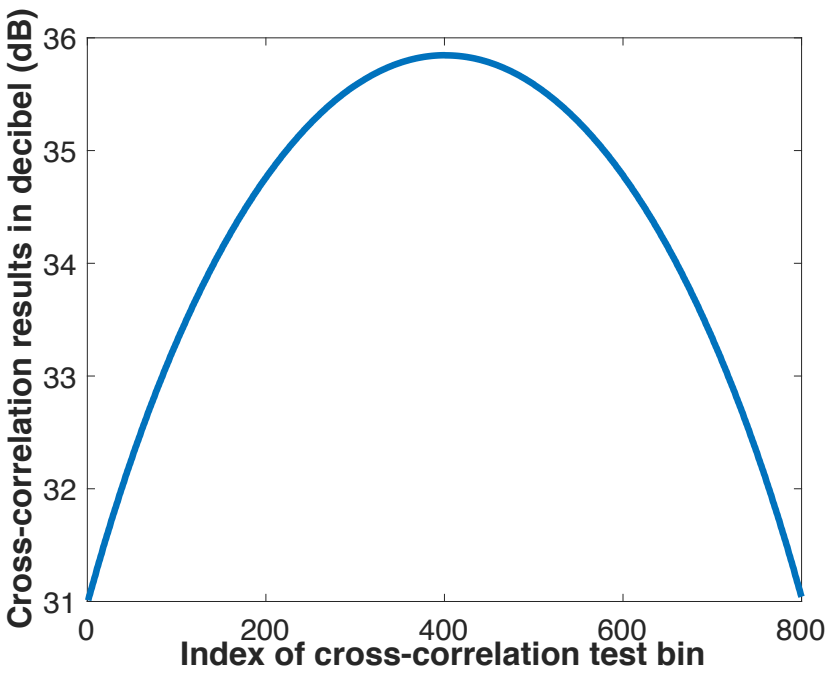


Figure 8.22: *The magnitude of massive correlation results increase up to 36dB assuming each input complex number has a magnitude of one which is the full scale of the 12 bits fixed-point format.*

where $\Delta = \frac{2}{2^{12}}$ assuming the peak to peak value is 2. So we synthesize the uniform distribution noise and input it onto the massive correlation multiple times. Then collect the output and plot the results in Figure (8.23). The multiple runs show the cross-correlation results are not beyond -13 dB. Therefore, the ideal dynamic range of the massive correlation is about 49 dB when the 12 bits fix-point representation is used.

In a more practical scenario, the thermal noise from the ADC will dominate the error source. We use the same scenario on Equation (6.5) and the same receiving gain setting of 52 dB. So the thermal noise power is about -55 dBm. We can convert this thermal noise to the 12 bits fixpoint format domain (where the peak value is 1).

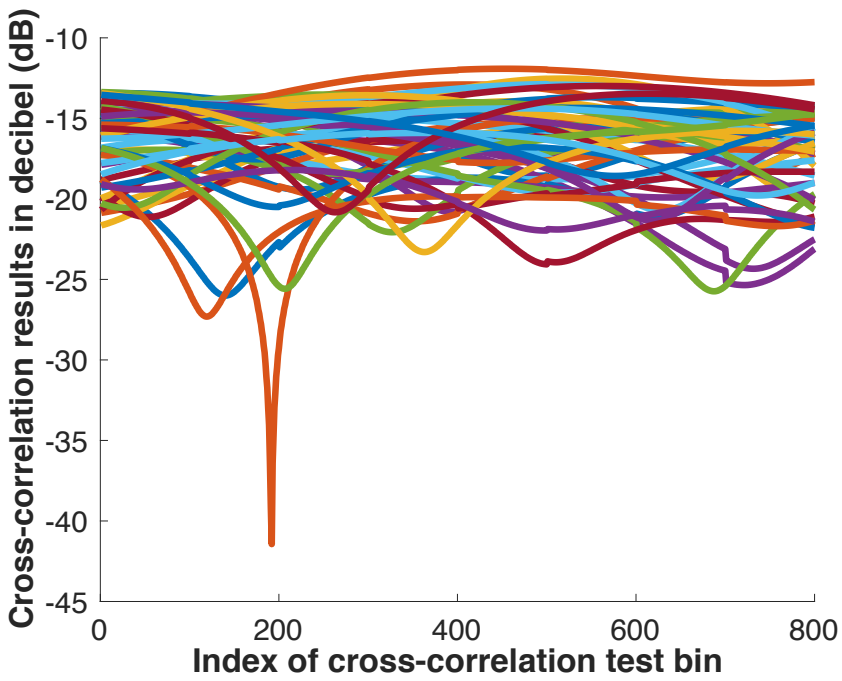


Figure 8.23: *Multiple iterations show the lower limit is not higher than -13 dB when only the quantization noise waveform are input to the massive correlator.*

The variance of thermal noise is calculated by the following

$$\begin{aligned} \sigma_{\text{noise in voltage}} &= \sqrt{P_{\text{thermal noise}} \times Z_{\text{IN}}} \\ &= \sqrt{10^{-55\text{dBm}/10} \times 100\Omega} \\ &= 0.0178 \text{ volts.} \end{aligned} \tag{8.13}$$

Then we convert the thermal noise standard variance from the voltage into the fix-point number domain.

$$\begin{aligned} \sigma_{\text{noise}} &= 0.0178/v_p \\ &= 0.0178/0.625 = 0.0285 \end{aligned} \tag{8.14}$$

Later then, we synthesize multiple thermal noise only waveforms by the white Gaussian distribution $n(0, \sigma^2_{\text{noise}})$ and then quantize it by 12 bits fix-point format. Input

the thermal noise only waveform multiple times and collect the results as in Figure (8.24). This sets the minimum correlation value. In a conclusion, the maximum achievable correlation result is about 36dB. The minimum value is about 5dB when the thermal noise is dominated. Thus, the practical dynamic range is about 31 dB.

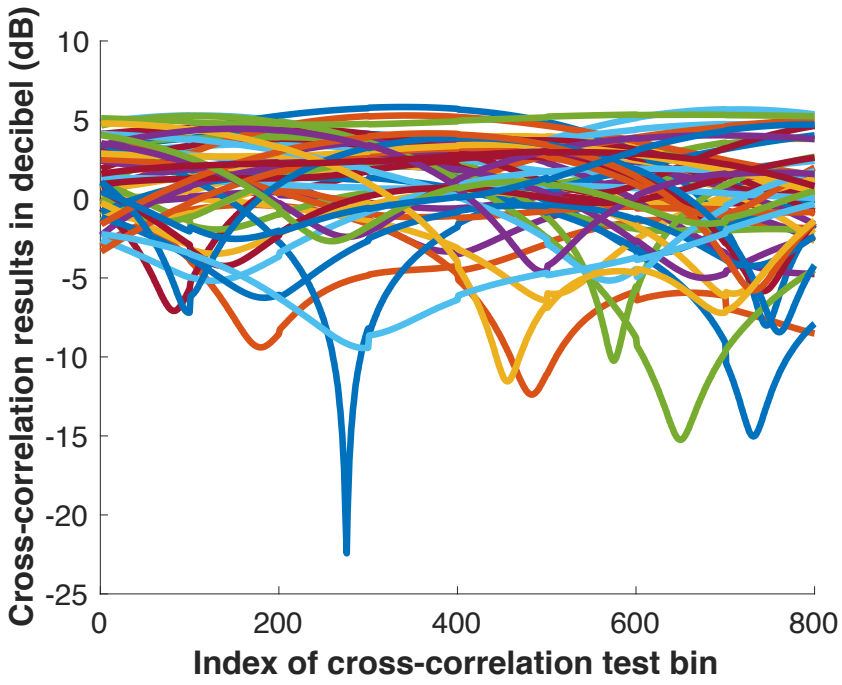


Figure 8.24: *Multiple Iterations show the lower limit is not higher than 5 dB when only the thermal noise are input to the massive correlator.*

Chapter 9

EXPERIMENTAL RESULTS[1]

For the performance verification and demonstration purpose, in-lab cabling testing, and several times of on-field trials were conducted during the development of CHP2 software and hardware. The scenario of two CHP2 nodes is assumed for all experiments. One node is the ground station, and the other one is the remote node, which is mounted on an aerial platform. The on-field testing is the actual over-the-air experiment, while in-lab testing has a similar setup but uses RF cables instead of over-the-air through the antenna. The ToA estimators are the same for both cases, but different ToF estimation algorithms' performance is evaluated.

The CHP2 system keeps iteration during implementation and verification. Those differences in performance only reflect the status during certain stages. The cabling test achieves better performance than flight demonstration since the firmware and algorithm are the most updated version. In this case, we accomplish a sub-centimeter ($\leq$ 1cm) level ranging precision.

9.1 Cable Testing

Due to the nature of the CHP2 system, any multiple path effects will distribute the ranging accuracy. Cable testing helps to debug the ToA and ToF estimator without introducing unknown external errors. Two CHP2 nodes are connected by RF cable, RF attenuation, and signal combiner to emulate an ideal line of sight prorogation channel as Figure 9.1 shows. The estimated ToF is not corresponding to the actual cable length because it is not feasible to measure the antennas' geometrical coordi-

[1]This is a joint work with Yang Li and Airbus ExO Alpha team.

nates in the cable test. Some default geometry information is used. The estimation performance can only be evaluated by the variance of the result data set. Also, the system emits a 10MHz bandwidth signal with a 915MHz carrier in the USA. The AGC maintains the SNR about 25 dB.

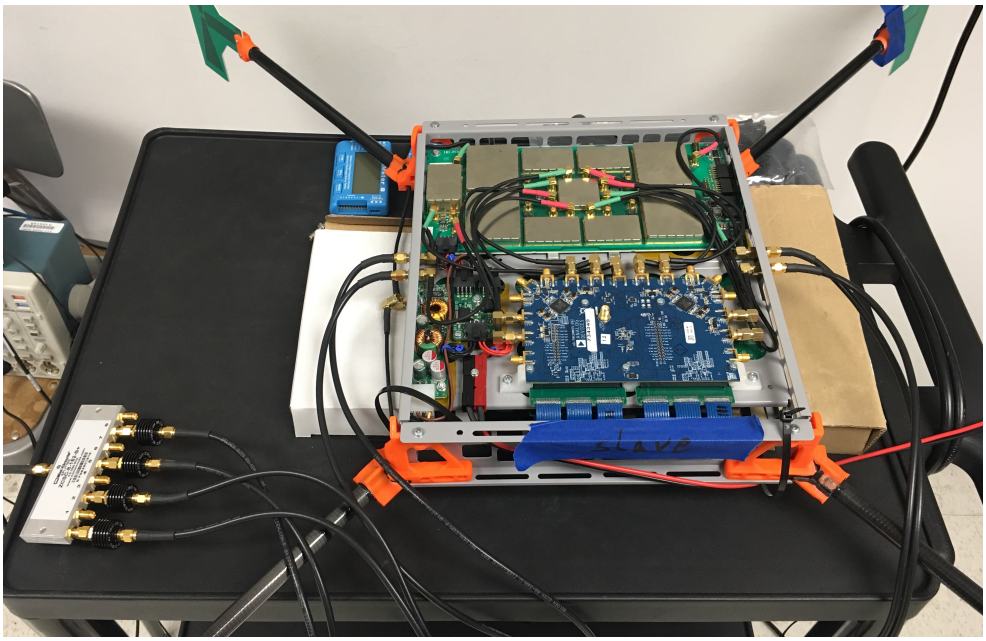


Figure 9.1: *The CHP2 assembly connects to the each other through RF signal attenuators and combiner.*

9.1.1 NTP Performance

For the same data set, the performance using peak detection ToA estimator and standard NTP ToF estimator is shown as Figure 9.2. The second-order polynomial interpolating improves the ToF estimation performance significantly as Figure 9.3. The precision is further increased to $\sim$ 1cm level.

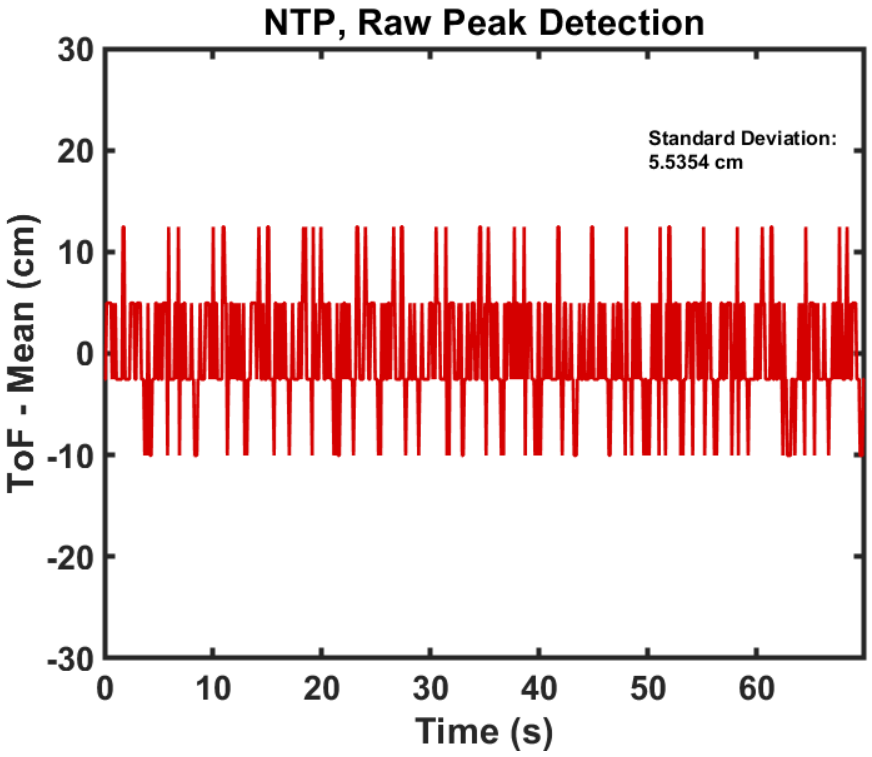


Figure 9.2: *The collected set of timestamps from CHP2 system is processed through NTP algorithm. The estimated ToF is centralized to zero mean. Its standard variance of this data set is 5.5cm*

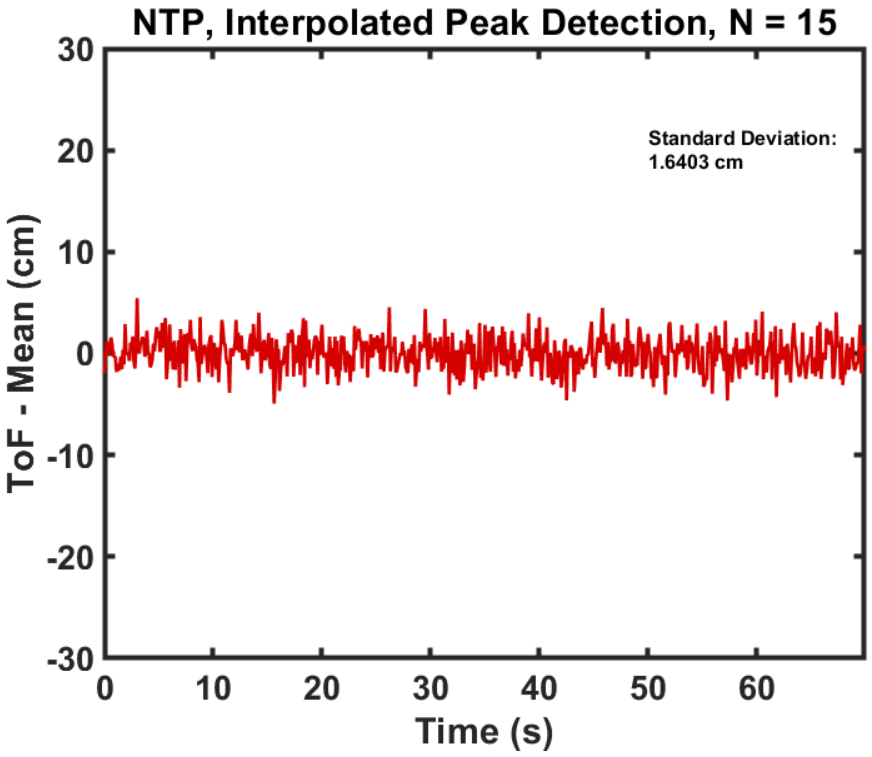


Figure 9.3: *The collected set of timestamps from CHP2 system is processed through NTP algorithm. The estimated ToF is centralized to zero mean. Its standard variance of this data set is 1.6cm. A significant ToA estimation improvement is achieved by a second order polynomial interpolating method.*

9.1.2 Kalman Filter Performance

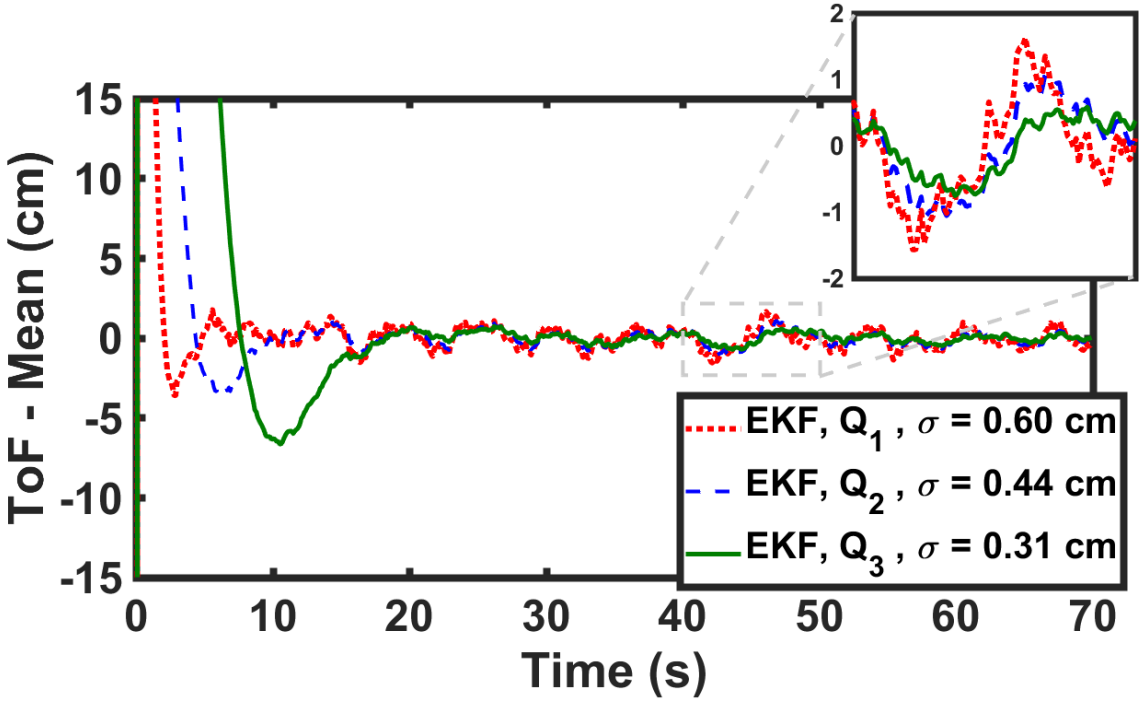


Figure 9.4: *The performance of EKE filter with different initial Q estimates.*

Kalman filters are implemented to improve ToF estimates performance. A data set runs through first-order extended Kalman filters with adaptive Q estimation every iteration. The EKE performance is shown as Figure 9.4. The standard variance is further improved to the sub-centimeter level. During the onboard verification, we find Kalman Filter is very sensitive to large distortion on input ToA timestamps value. The filter may lose tracking once a large distortion occurs but may slowly recover later than.

9.2 Over-the-Air demonstration

The flight demonstration was performed at a general aviation airport in Nordlingen, Germany. The CHP2 ground station and the remote node were set up on the end of the runway as Figure 9.5. The ground station antennas were configured as Figure 9.7 to minimum distortion of the RF signal bouncing the ground. A drone was carrying the CHP2 remote node on the origin point of the chosen coordinate system, as Figure 9.6. During the flying testing campaign, several flight paths were conducted, including a vertical take-off and landing (VTOL) test, a short-range test, a u-turn testing, and long-range testing. The primary purpose of the flying tests was to verify whether the ranging estimates were reasonable and stable as well as the whole CHP2 system stability.

The CHP2 system for this over-the-air demonstration was running under a licensed RF carrier frequency of 783MHz. Its bandwidth was 10MHz width, as same as the in-lab test in the US. As mentioned before, this demonstration was at the early stage of CHP2 performance verification. The ToF estimation solution used in this demonstration was the NTP algorithm. Since the firmware was not ready to include the interpolated massive correlation peak detection, the EKE time-of-flight estimation, and the geometry conversion, this demonstration exclusively focused on range estimation. We expected the estimation performance might degrade slightly in actual wireless channel comparing to the in-lab cabling test. The performance during this flight test was not the final goal of the CHP2 but expected.

Figure 9.5: *Depiction of flying test setup on the end of airport runway.*

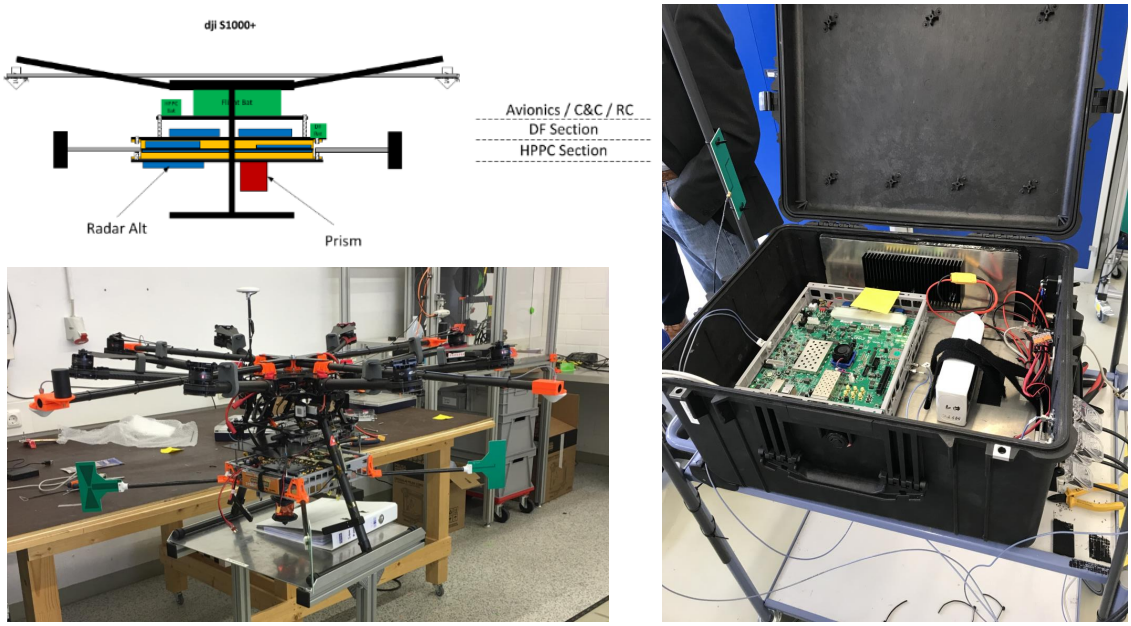


Figure 9.6: *Depiction of the CHP2 remote node on the drone and the ground station. A crystal prism is mounted on the button of drone payload for the laser tachymeter tracking.*

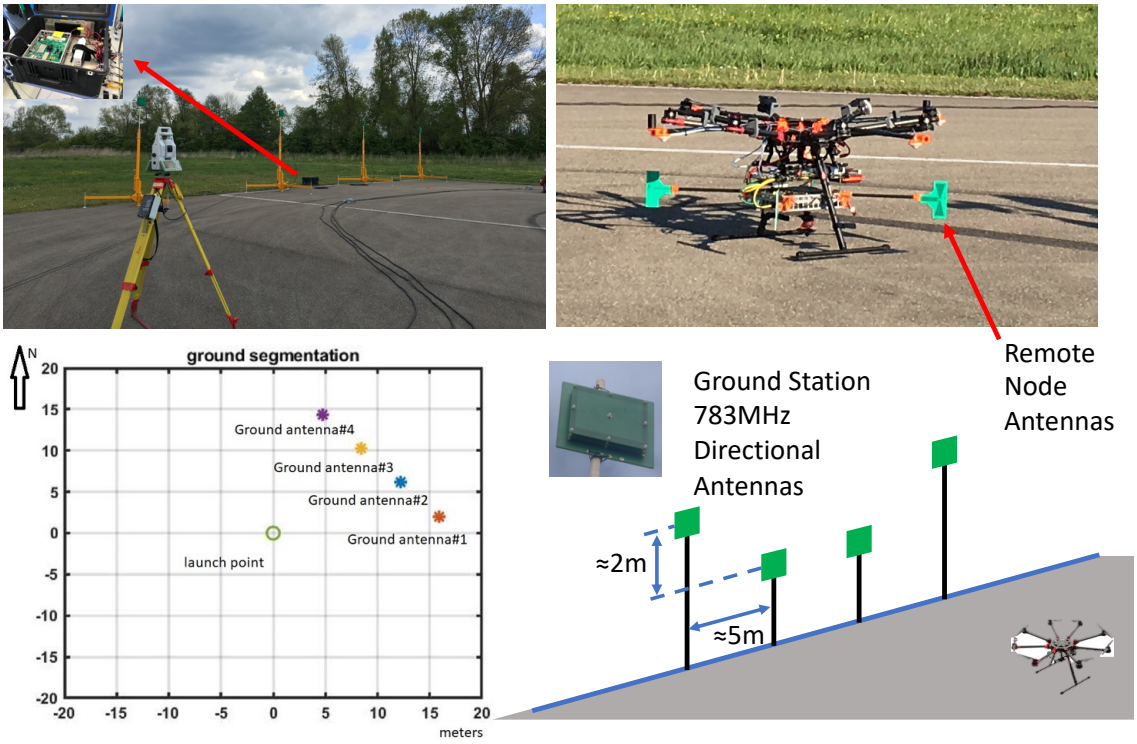


Figure 9.7: *Depiction of the ground station antenna geometry configuration. A laser tachymeter was used for measuring all antenna geometry. It can real-time track the flying drone and generate true reference during the flying test.*

9.2.1 VTOL Test

This test demonstrated the range varying during the drone taking off and land vertically. As Figure 9.8 shows, it lifted off from the initial location vertically and hovered over about 10m for a while. Then vertically landed on the ground. Right before the taking off, there was a manually moving drone on the ground to see the range estimation change. However, when the operators moved the drone, the CHP2 RF signal was blocked and caused much multi-path disturbance on the estimates. The tachymeter tracked most of the flight path through the whole test. Compared to the CHP2 estimation, the tachymeter has a slight offset, but as expected. The laser prism was on the center of the drone while the CHP2 antennas were on the four corners of the drone. It leads to a half meter bias naturally. If the drone rotated its direction, this bias would be changed as the relative distance between the ground station antenna and the remote antenna on the drone can be changed.

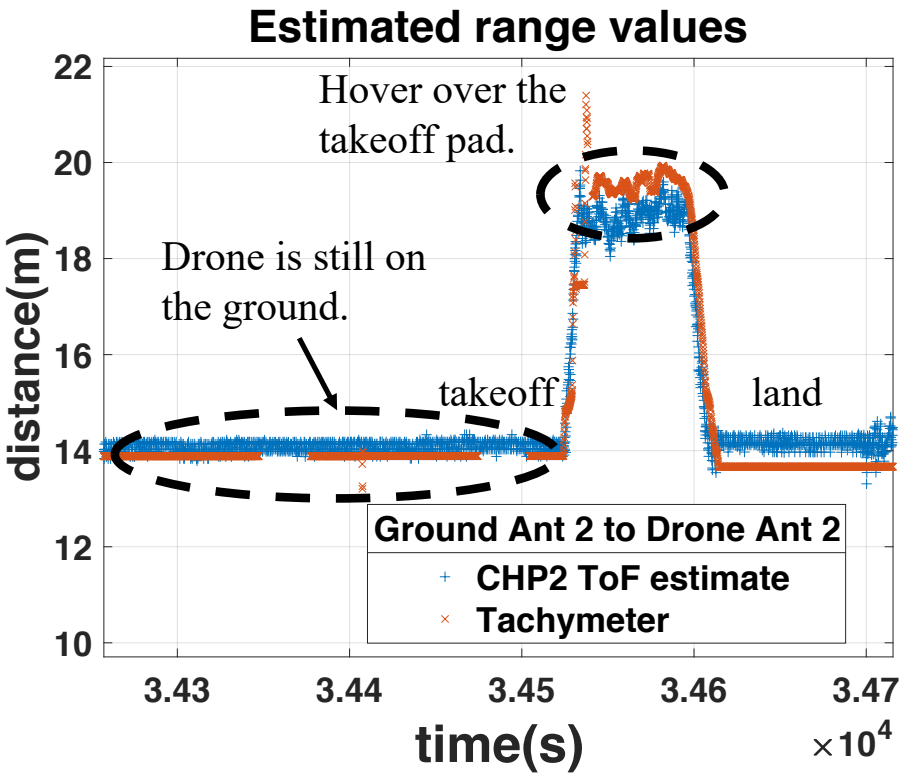


Figure 9.8: *Range estimation results for the manual relocation and vertical take-off and landing tests. During the manual test, the operators picking up the UAV obstruct the line-of-sight path between the base station and UAV, which significantly degraded the system performance. During the VTOL test, the range estimation closely followed the tachymeter reference.*

For precision verification purposes, we analyzed the measure performance when the drone was in the static on the ground. Zoom in the distance measurement result as in Figure (9.9) and (9.10). We clearly see some artificial perturbation. It turns out this phenomenon is related to quantization issues during the massive correlation. We solved this issue after this flight testing. The remaining disturbance should come from the multi-path effect as in Table (9.1). The worst antenna pair achieved 16.59cm while the best pair achieved a 7.96cm standard deviation.

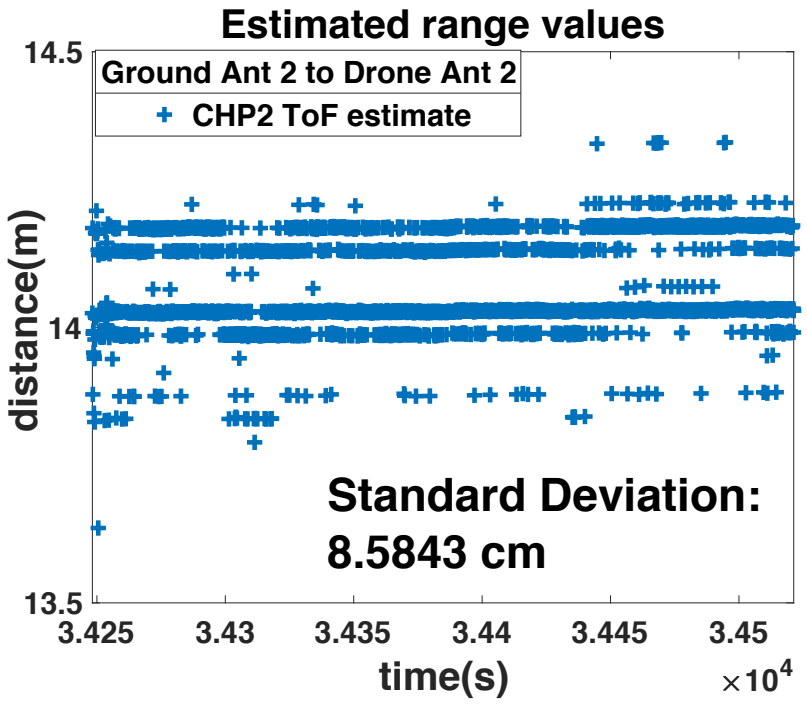


Figure 9.9: *The range estimation results over a small chunk of time on the VTOL test from Ground Ant 2 to Drone Ant 2. The estimates were analyzed during the drone was still on the ground before taking off. The standard deviation of this interval is about 8.5cm.*

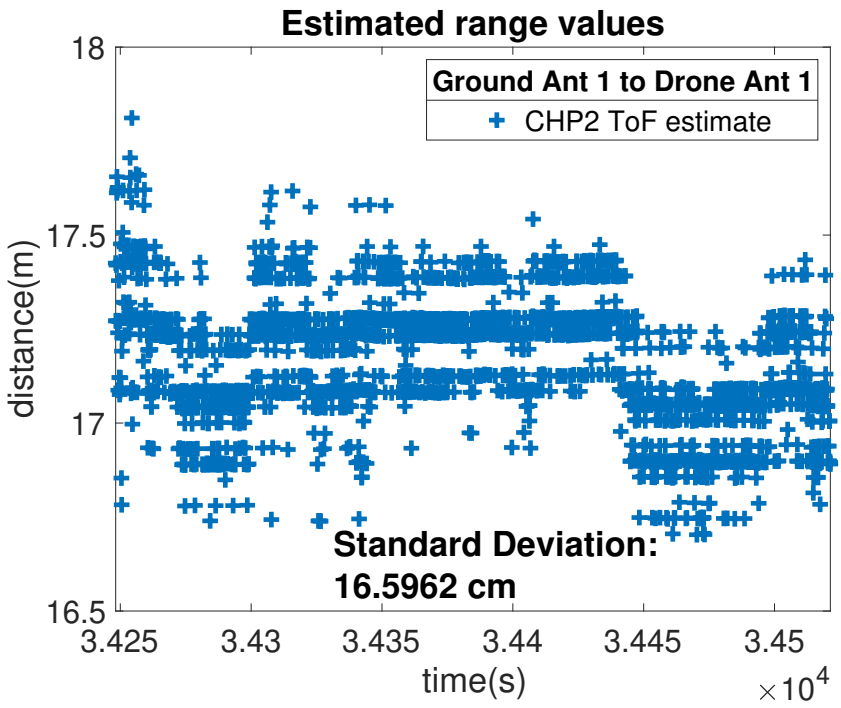


Figure 9.10: *The range estimation results over a small chunk of time on the VTOL test from Ground Ant 1 to Drone Ant 1. The estimates were analyzed during the drone was still on the ground before taking off. The standard deviation of this interval is about 16.592cm. It seems distortion happens at some time interval.*

Table 9.1: *The standard deviations of range estimation results*

Unit: centimeter(cm)				
Antenna Pair	Ground Ant1	Ground Ant2	Ground Ant3	Ground Ant4
Drone Ant1	16.59	14.12	11.62	11.24
Drone Ant2	13.54	8.58	7.96	8.78
Drone Ant3	11.39	8.91	8.43	9.11
Drone Ant4	10.68	9.18	9.02	10.18

9.2.2 *Short Range Test*

This test demonstrates a short-range estimation as Figure 9.11. The drone took off and flies along the runway. It then landed on the midway. After a brief inspection, the drone lift off again and flew back to the origin. During the inspection, the laser beam and CHP2 signal were blocked by the operator as it is shown in the figure. The laser tracking was lost, and CHP2 range estimates have a disturbance.

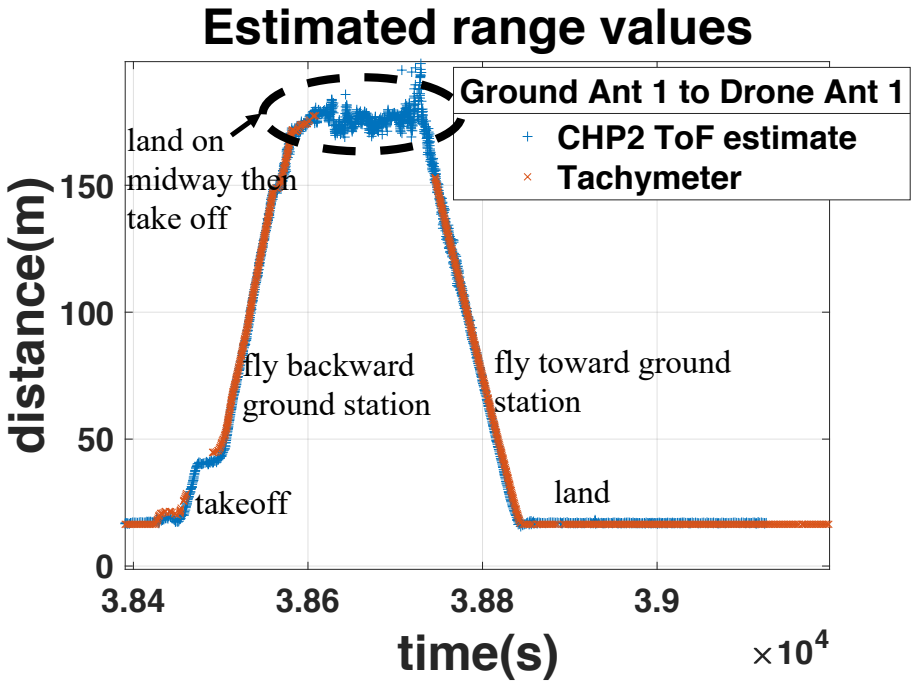


Figure 9.11: *Range estimation results for midway landing and take-off. After the drone lands on the midway, the laser tracking was blocked when the operator was routinely inspecting the drone. The CHP2 range estimates were also disturbed during this process.*

9.2.3 U-turn Test

This test demonstrates the range estimation changing as Figure 9.12 shown. The drone lifted off 20m above the landing pad. Then it kept climbing until about 50m high and flew toward the end of the runway. Then it turned around and flew back to the initial location. From Figure 9.12, it clearly shows the range estimates reach about 760m, which is the maximum length of the runway. The range estimates are very stable, and the curve is very smooth. There were slight offsets among different antenna pairs. It is expected since each pair has a slight geometry difference. The flight path reaches longer than the laser tracking range limitation 200m. So it is disabled.

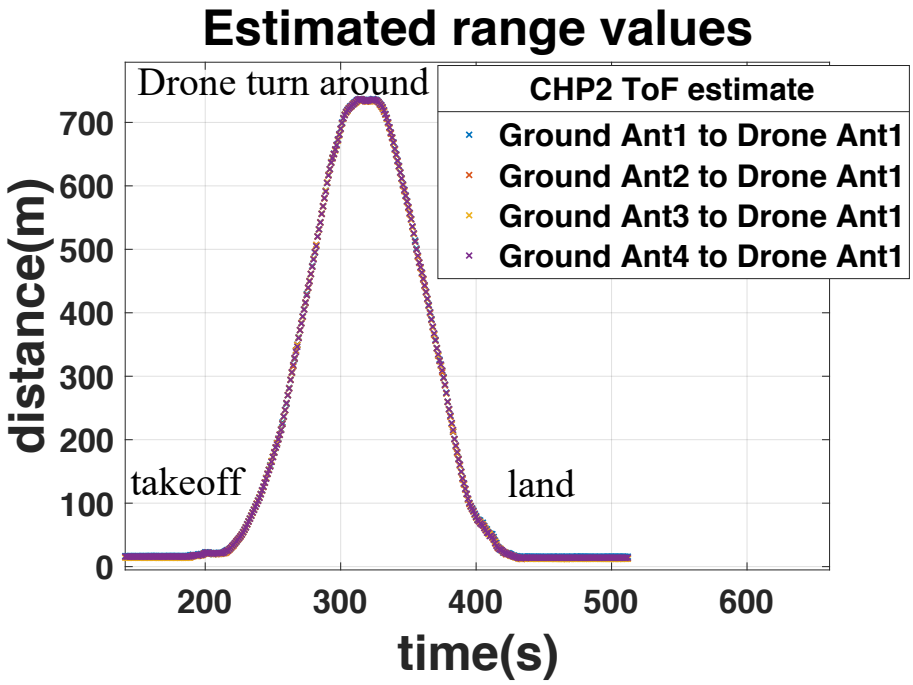


Figure 9.12: *Range estimation results for a flying forward and backward along the airport runway.*

9.2.4 Long Range Test

The long-range test should verify whether the CHP2 system can properly work over kilometers level distance. However, due to the safety issue and regulation, we did not fly the drone. Instead, we put it inside an SUV and drove it away on an unpaved road from the airport. During this test, the communication link was kept connected until about 2.6km away when the car went downhill, and the line-of-sight was totally blocked.

9.3 Conclusion

This over-the-air demonstration was very successful. It validated the idea of the CHP2 system and provided a solid example for the UAV positioning application.

From the aspect of accuracy, the CHP2 system range estimates almost aligned with laser tachymeter measurements. There was an artificial bias because the laser tracks the prism while the CHP2 system determined the range from the antenna to the antenna. The artificial bias can be easily corrected if the coordinate of the prism is known.

From the aspect of precision, the over-air-test results were very close to the in-lab tests without the polynomial fitting peak finding. Some of the channels achieved a reasonable standard deviation, which was less than 10 cm, but were are not. In Figure 9.9 and 9.10, it seems there was a distortion on the channel from the ground Antenna 1 to the drone Antenna 1. And in Table 9.1, the good estimation variances came from certain antenna pair. Since the CHP2 antennas were mounted on four corners of the drone, It implied a multi-path effect might happen on certain antennas.

From the aspect of robustness, the communication link worked much better than the expectation. The link losing happens when the signal was totally blocked during the SUV downhill at 2.6km far. The AGC mechanism worked properly, and the RF amplifier switching board also performed excellently. It should be no issue once the over-the-air distance between two nodes is more than 10km.

Chapter 10

SUMMARY AND FUTURE WORK

The Communication and Hyper-Precision Positioning system provides a high performance, low-cost solution for positioning and navigation of the modern aviation system. This decentralized point-to-point system offers an extremely high ranging precision about 1cm only using a narrow bandwidth of 10MHz. The small form factor of the CHP2 enables integration itself onto numerous platforms with minimal resource consumption to support a potential high-density CHP2 network.

The system architecture development, hardware, and firmware implementation satisfy the initial requirement of the CHP2 project. This universal SDR platform (ZCU102 and AD9361) with minimum customized hardware (for example, TR switch card and antennas) maintain the low cost, high performance, and flexibility of the CHP2 system. To achieve this 100ms estimation fresh rate and centimeters level precision, we implement the cross-correlation for signal acquisition and coarse ToA estimation, the massive correction for fine ToA estimation as HDL IP cores, the complete communication processing chain as low-level firmware program, the NTP, and EKE filter based ToF estimation methods as application-level program.

Currently, we narrow down the CHP2 ToA precision to four of the wavelength, which is about 7.5cm (assuming the center carrier frequency is the ISM band 915MHz). And furthermore, improve the precision down to about 1cm by interpolating the peak detection on fine ToA estimation. We achieve sub-centimeter precision goal by applying EKE filter.

Although the CHP2 system is very stable in terms of communication link robustness, based on the experiment, we observe the range estimation is easily interrupted by

a multi-path effect, and its precision can degrade due to an improper receiving SNR. This may cause an issue for urban environment applications, for example, air traffic management within a city area. This issue has to be addressed in the future. We continue to develop applications of this technology, including communications, navigation, and surveillance (CNS), air traffic management (ATM), automated vehicles, collision avoidance, intelligent transport systems (ITS), and distributed beamforming.

REFERENCES

[1] A. Herschfelt, H. Yu, S. Wu, H. Lee, and D. W. Bliss, "Joint positioning-communications system design: Leveraging phase-accurate time-of-flight estimation and distributed coherence," in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2018, pp. 433–437.

[2] A. Herschfelt, H. Yu, S. Wu, S. Srinivas, Y. Li, N. Sciammetta, L. Smith, K. Rueger, H. Lee, C. Chakrabarti, and D. W. Bliss, "Joint positioning-communications system design and experimental demonstration," in *38th AIAA/IEEE Digital Avionics Systems Conference*. IEEE, 2019.

[3] D. L. Mills, "Internet time synchronization: the network time protocol," *IEEE Transactions on communications*, vol. 39, no. 10, pp. 1482–1493, 1991.

[4] D. Mills, "Network time protocol (version 3) specification, implementation and analysis," 1992.

[5] S. Han, Z. Gong, W. Meng, C. Li, and X. Gu, "Future alternative positioning, navigation, and timing techniques: a survey," *IEEE wireless communications*, vol. 23, no. 6, pp. 154–160, 2016.

[6] B. Paul, A. R. Chiriyath, and D. W. Bliss, "Survey of rf communications and sensing convergence research," *IEEE Access*, vol. 5, pp. 252–270, 2017.

[7] B. W. Parkinson, P. Enge, P. Axelrad, and J. J. Spilker Jr, *Global positioning system: Theory and applications, Volume II*. American Institute of Aeronautics and Astronautics, 1996.

[8] P. Misra and P. Enge, "Global positioning system: signals, measurements and performance second edition," *Massachusetts: Ganga-Jamuna Press*, 2006.

[9] A. Cailean, B. Cagneau, L. Chassagne, S. Topsu, Y. Alayli, and J.-M. Blosseville, "Visible light communications: Application to cooperation between vehicles and road infrastructures," in *Intelligent Vehicles Symposium (IV), 2012 IEEE*. IEEE, 2012, pp. 1055–1059.

[10] C. Sturm and W. Wiesbeck, "Waveform design and signal processing aspects for fusion of wireless communications and radar sensing," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1236–1259, 2011.

[11] V. A. Orlando, "The mode s beacon radar system," *The Lincoln Laboratory Journal*, vol. 2, no. 3, pp. 345–362, 1989.

[12] M. Strohmeier, M. Schafer, V. Lenders, and I. Martinovic, "Realities and challenges of nextgen air traffic management: the case of ads-b," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 111–118, 2014.

[13] O. A. Yeste-Ojeda, J. Zambrano, and R. Landry, "Design of integrated mode s transponder, ads-b and distance measuring equipment transceivers," in *Integrated Communications Navigation and Surveillance (ICNS), 2016*. IEEE, 2016, pp. 4E1–1.

[14] N. Decarli, F. Guidi, and D. Dardari, "A novel joint rfid and radar sensor network for passive localization: Design and performance bounds," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 1, pp. 80–95, 2014.

[15] P. Bidigare, "The shannon channel capacity of a radar system," in *Signals, Systems and Computers, 2002. Conference Record of the Thirty-Sixth Asilomar Conference on*, vol. 1. IEEE, 2002, pp. 113–117.

[16] X. Shaojian, C. Bing, and Z. Ping, "Radar-communication integration based on dsss techniques," in *Signal Processing, 2006 8th International Conference on*, vol. 4. IEEE, 2006.

[17] M. Jamil, H.-J. Zepernick, and M. I. Pettersson, "On integrated radar and communication systems using oppermann sequences," in *Military Communications Conference, 2008. MILCOM 2008. IEEE*. IEEE, 2008, pp. 1–6.

[18] X. Li, R. Yang, Z. Zhang, and W. Cheng, "Research of constructing method of complete complementary sequence in integrated radar and communication," in *Signal Processing (ICSP), 2012 IEEE 11th International Conference on*, vol. 3. IEEE, 2012, pp. 1729–1732.

[19] Y. Xie, R. Tao, and T. Wang, "Method of waveform design for radar and communication integrated system based on css," in *Instrumentation, Measurement, Computer, Communication and Control, 2011 First International Conference on*. IEEE, 2011, pp. 737–739.

[20] X. Chen, X. Wang, S. Xu, and J. Zhang, "A novel radar waveform compatible with communication," in *Computational Problem-Solving (ICCP), 2011 International Conference on*. IEEE, 2011, pp. 177–181.

[21] G. N. Saddik, R. S. Singh, and E. R. Brown, "Ultra-wideband multifunctional communications/radar system," *IEEE Transactions on Microwave Theory and Techniques*, vol. 55, no. 7, pp. 1431–1437, 2007.

[22] A. Springer, W. Gugler, M. Huemer, L. Reindl, C. Ruppel, and R. Weigel, "Spread spectrum communications using chirp signals," in *EUROCOMM 2000. Information Systems for Enhanced Public Safety and Security. IEEE/AFCEA*. IEEE, 2000, pp. 166–170.

[23] S. D. Blunt, M. Cook, J. Jakabosky, J. De Graaf, and E. Perrins, "Polyphase-coded fm (pcfm) radar waveforms, part i: implementation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 3, pp. 2218–2229, 2014.

[24] S. D. Blunt, J. Jakabosky, M. Cook, J. Stiles, S. Seguin, and E. Mokole, "Polyphase-coded fm (pcfm) radar waveforms, part ii: optimization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 3, pp. 2230–2241, 2014.

[25] C. Sahin, J. Jakabosky, P. M. McCormick, J. G. Metcalf, and S. D. Blunt, "A novel approach for embedding communication symbols into physical radar waveforms," in *Radar Conference (RadarConf), 2017 IEEE*. IEEE, 2017, pp. 1498–1503.

[26] C. A. Mohr, P. M. McCormick, S. D. Blunt, and C. Mott, "Spectrally-efficient fm noise radar waveforms optimized in the logarithmic domain," in *Radar Conference (RadarConf18), 2018 IEEE*. IEEE, 2018, pp. 0839–0844.

[27] P. Bidigare, U. Madhow, R. Mudumbai, and D. Scherber, "Attaining fundamental bounds on timing synchronization," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 5229–5232.

[28] P. Bidigare, S. Pruessing, D. Raeman, D. Scherber, U. Madhow, and R. Mudumbai, "Initial over-the-air performance assessment of ranging and clock synchronization using radio frequency signal exchange," in *Statistical Signal Processing Workshop (SSP), 2012 IEEE*. IEEE, 2012, pp. 273–276.

[29] T. E. McEwan, "Time-of-flight radio location system," Apr. 23 1996, uS Patent 5,510,800.

[30] L. W. Fullerton, J. L. Richards, and I. A. Cowie, "System and method for position determination by impulse radio using round trip time-of-flight," Aug. 26 2003, uS Patent 6,611,234.

[31] S. Lanzisera, D. Zats, and K. S. Pister, "Radio frequency time-of-flight distance measurement for low-cost wireless sensor localization," *IEEE Sensors Journal*, vol. 11, no. 3, pp. 837–845, 2011.

[32] F. Sivrikaya and B. Yener, "Time synchronization in sensor networks: a survey," *IEEE network*, vol. 18, no. 4, pp. 45–50, 2004.

[33] B. Sundararaman, U. Buy, and A. D. Kshemkalyani, "Clock synchronization for wireless sensor networks: a survey," *Ad hoc networks*, vol. 3, no. 3, pp. 281–323, 2005.

[34] S. Tsugawa and S. Kato, "Energy its: another application of vehicular communications," *IEEE Communications Magazine*, vol. 48, no. 11, 2010.

[35] S. Eckelmann, T. Trautmann, H. Ußler, B. Reichelt, and O. Michler, "V2v-communication, lidar system and positioning sensors for future fusion algorithms in connected vehicles," *Transportation Research Procedia*, vol. 27, pp. 69–76, 2017.

[36] K.-D. Langer and J. Grubor, "Recent developments in optical wireless communications using infrared and visible light," in *Transparent Optical Networks, 2007. ICTON'07. 9th International Conference on*, vol. 3. IEEE, 2007, pp. 146–151.

[37] A. Belle, M. Falcitelli, M. Petracca, and P. Pagano, "Development of ieee802. 15.7 based its services using low cost embedded systems," in *ITS Telecommunications (ITST), 2013 13th International Conference on*. IEEE, 2013, pp. 419–425.

[38] J. M. Dow, R. E. Neilan, and C. Rizos, "The international gnss service in a changing landscape of global navigation satellite systems," *Journal of geodesy*, vol. 83, no. 3-4, pp. 191–198, 2009.

[39] S. Lo, Y. H. Chen, P. Enge, B. Peterson, R. Erikson, and R. Lilley, "Distance measuring equipment accuracy performance today and for future alternative position navigation and timing (apnt)," in *Proceedings of the 26th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2013), Nashville, TN*, 2013, pp. 711–721.

[40] K. Pourvoyeur, A. Mathias, and R. Heidger, "Investigation of measurement characteristics of mlat/wam and ads-b," in *2011 Tyrrhenian International Workshop on Digital Communications-Enhanced Surveillance of Aircraft and Vehicles*. IEEE, 2011, pp. 203–206.

[41] M. A. Garcia, R. Mueller, E. Innis, and B. Veytsman, "An enhanced altitude correction technique for improvement of wam position accuracy," in *2012 Integrated Communications, Navigation and Surveillance Conference*. IEEE, 2012, pp. A4–1.

[42] H. Neufeldt and S. Stanzel, "An operational wam in frankfurt airspace," in *2013 14th International Radar Symposium (IRS)*, vol. 2. IEEE, 2013, pp. 561–566.

[43] S.-S. Jan, S. L. Jheng, Y. H. Chen, and S. Lo, "Evaluation of positioning algorithms for wide area multilateration based alternative positioning navigation and timing (apnt) using 1090 mhz ads-b signals," in *27th International Technical Meeting of the Satellite Division of the Institute of Navigation, ION GNSS 2014*. Institute of Navigation, 2014, pp. 3016–3028.

[44] "ADS-B frequently asked questions (faqs)," https://www.faa.gov/nextgen/programs/adsb/faq/#i2, accessed: 2010-10-17.

[45] J. Barnes, J. Wang, C. Rizos, and T. Tsujii, "The performance of a pseudolite-based positioning system for deformation monitoring," in *2nd Symp. on Geodesy for Geotechnical & Structural Applications*, 2002, pp. 21–24.

[46] D. Shutin, N. Schneckenburger, M. Walter, and M. Schnell, "Ldacs1 ranging performance-an analysis of flight measurement results," in *2013 IEEE/AIAA 32nd Digital Avionics Systems Conference (DASC)*. IEEE, 2013, pp. 3C6–1.

[47] A. Yassin, Y. Nasser, M. Awad, A. Al-Dubai, R. Liu, C. Yuen, R. Raulefs, and E. Aboutanios, "Recent advances in indoor localization: A survey on theoretical approaches and applications," *IEEE Communications Surveys Tutorials*, vol. 19, no. 2, pp. 1327–1346, Secondquarter 2017.

[48] Analog Devices Inc. (2018, June) HDL Reference Designs (2018 Release 1). [Online]. Available: https://github.com/analogdevicesinc/hdl/tree/hdl_2018_r1

[49] NSTB/WAAS T&E Team. (2018, April) Wide-area augmentation system performance analysis report: report #64. [Online]. Available: https://www.nstb.tc.faa.gov/reports/waaspan64.pdf

[50] NovAtel. AdVance® RTK Competitive Analysis. [Online]. Available: https://www.novatel.com/assets/Documents/Papers/AdVance_RTK_Competitive_Analysis.pdf

[51] M. Schnell, U. Epple, D. Shutin, and N. Schneckenburger, "Ldacs: future aeronautical communications for air-traffic management," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 104–110, May 2014.

[52] D. Shutin, N. Schneckenburger, M. Walter, and M. Schnell, "Ldacs1 ranging performance - an analysis of flight measurement results," in *2013 IEEE/AIAA 32nd Digital Avionics Systems Conference (DASC)*, Oct 2013, pp. 1–21.

[53] A. Trunzo, R. Ramirez, and J. Baldwin, "The uhars non-gps-based positioning system," *Proceedings of the 27th International Technical Meeting of the Satellite Division of The Institute of Navigation*, pp. 3243 – 3248, September 2014.

[54] Xilinx Inc. (2019, October) Zynq UltraScale+ MPSoC Data Sheet: Overview v1.8. [Online]. Available: https://www.xilinx.com/support/documentation/data_sheets/ds891-zynq-ultrascale-plus-overview.pdf

[55] N. Patwari, A. O. Hero, M. Perkins, N. S. Correal, and R. J. O'dea, "Relative location estimation in wireless sensor networks," *IEEE Transactions on signal processing*, vol. 51, no. 8, pp. 2137–2148, 2003.

[56] Analog Devices, Inc. RF Agile Transceiver AD9361: Data Sheet Rev. F. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/AD9361.pdf

[57] Qorvo, Inc. (2017, July) QPL9095 Ultra Low-Noise Bypass LNA Datasheet. [Online]. Available: https://www.qorvo.com/products/d/da006183

[58] Xilinx Inc. (2019, June) ZCU102 Evaluation Board User Guide UG1182 (v1.6). [Online]. Available: https://www.xilinx.com/support/documentation/boards_and_kits/zcu102/ug1182-zcu102-eval-bd.pdf

[59] ARM. (2013) NEON Programmer's Guide Version: 1.0. [Online]. Available: https://developer.arm.com/docs/den0018/a/neon-programmers-guide-version-10

[60] Analog Devices, Inc. AD9361 Reference Manual UG-570. [Online]. Available: http://www.farnell.com/datasheets/2007082.pdf