Network Layer Spectral Coordination Integrated With Hadamard Projection for Multilayer Interference Mitigation

Ashton Palacios*, Devon Ward*, Dinah Bronson*, Jon Backman*, Deukhyoun Heo†, Karl F. Warnick*, Philip Lundrigan*
*Department of Electrical and Computer Engineering
Brigham Young University
Provo, Utah, USA
{apal6981, dward28, disq256, backmanj, warnick, lundrigan}@byu.edu
†School of Electrical Engineering & Computer Science
Washington State University
Pullman, Washington, USA
dheo@wsu.edu

***Abstract*—Sharing crowded radio spectrum with astronomical observatories and other passive users requires a multilayer approach. We present a protocol that allows passive devices to notify nearby active users of their spectrum utilization plans, regardless of the wireless protocol used by the receiver. For non-compliant active users, we use a Hadamard projection based cancellation algorithm to remove residual interference. Hadamard projection can be implemented before additional stages of RFI rejection such as subspace projection that may occur during beamforming and imaging. The integrated system is demonstrated with laboratory and outdoor over the air experimental results. The system is able to preserve a signal of interest to the passive user while suppressing interference from both compliant and non-compliant active spectrum users.**

***Index Terms*—spectrum sharing, radio quiet zones, cross-technology communication, RFI cancellation**

I. Introduction

According to Cisco [1], there will be over 29.3 billion network connected devices in use around the world by the end of 2023. The majority of these devices are wireless. With hundreds of devices coming online every day, the wireless spectrum is becoming ever more crowded. A spectrum user group that has been greatly affected by the increase in wireless devices is the radio astronomy (RA) community. RA observatories are passive receivers that listen for distant signals from space at a variety of frequencies. Observation frequencies can range from 360 kHz to over 275 GHz [2] and the bandwidth can be anywhere between a few MHz to hundreds of MHz with observations lasting from a few minutes to hours. The data measured at the time of interference often has to be "blanked" (i.e., deleted) because it is difficult to remove the interference and recover the signal, leading to data loss and underutilization of costly and competitively scheduled astronomical instruments.

This material is based on work supported by the National Science Foundation under Grant No. 2030165.

Some RA sites are protected by radio quiet zones where transmissions are prohibited in a certain geographic areas, but with many observatory sites this is impossible. Low-earth orbit (LEO) satellites are encroaching on passive users in nearly all locations [3]. With interference from terrestrial and spaceborne sources, new ways are needed for proactively sharing the radio spectrum between passive devices and active users.

In designing improved spectrum sharing schemes between active and passive users, it is valuable to consider all layers of the traditional wireless communications stack. Interference is ideally dealt with in a cross-layer approach. If we treat interference only as a networking problem and only develop a protocol layer solution, then if a device does not support the specific networking protocol or is non-compliant, the mitigation approach fails. Similarly, noise cancellation in digital signal processing fails when powerful transmitters can drive the RF system into nonlinearity. Other techniques for noise cancellation lead to data loss. An ideal solution is cross-disciplinary and addresses multiple layers of the network stack in a coordinated manner.

To address this challenge, we have created a synergistic approach to dealing with interference at RA observatories in three different but complementary ways. Our solution provides innovations at multiple layers of the networking stack. First, we introduce a protocol to let RA observatories communicate with interfering devices to coordinate spectrum access. By giving RA receiver the ability to transmit before they start an observation, the observatory can notify potential nearby interfering nodes what frequencies and times the observatory will be using. This scheme provides a distributed method of coordination, instead of the usual centralized database that has been used in other spectrum coordination techniques. The challenge becomes how to communicate with interfering devices. Since RA spans such diverse frequency ranges, no one wireless protocol can be selected, as supporting all possible

protocols a RA receiver might encounter is not feasible. Our solution is a protocol that is universally implementable by any wireless device. Our protocol takes the base functionality of all transmitters and receivers: the ability to transmit energy and the ability to measure energy. Using these base primitives, we create a wireless subprotocol that all wireless devices can support. This gives RA receiver the ability to transmit out information that all nearby wireless nodes can receive, regardless of wireless protocol.

Second, for non-compliant active users that fail to coordinate using active protocol coordination, we consider mitigation techniques for both narrowband and wideband RFI sources. We use a passive technique to remove wideband RFI based on a true-time delay Hadamard projection approach proposed earlier in [4] and extended in [5]. In an RF system, when strong RFI is present, it can saturate ADCs and prevent further digital signal processing. The time delay Hadamard algorithm is ultimately intended to be implemented in the analog domain, allowing for RFI removal before digitizers to prevent saturation. In our experimental tests, due to delays in fabrication of wideband true time delay chips we use a digital implementation of the Hadamard transform. For high power narrowband non-compliant interferers, at the analog layer we have developed custom IC chips including a tunable notch filter to help remove a strong narrowband RFI source before the ADC and digital processing.

Our system provides a multi-layer approach that coordinates between different techniques to provide the best interference mitigation strategy for RA observatories. This work is a path forward for RA observatories to communicate their usage to active users and block out non-compliant active users, allowing for better usage of the spectrum. We demonstrate the integrated multi-layer approach experimentally using over the air tests.

II. Related Work

Cross-Technology Communication Protocols: Recent research shows promise in the area of cross technology communication (CTC) as a viable technique to bridge the communication gap between heterogeneous devices [6]–[11]. These techniques use two general methods: 1) emulating physical layer properties [9]–[11] and 2) use of timing characteristics of transmissions [6]–[8]. The physical layer emulation techniques are able to achieve high data rates, but they are a one-to-one solution: one physical layer is only able to emulate one other physical layer. These methods exploits specific RF characteristics shared between the two target protocols, and as a result, these solutions do not work outside of their target protocols making developing large-scale inter-technology communication systems with them difficult. Our goal of large-scale inter-technology communication leads to a solution that sacrifices data rate for interoperability of devices.

Other CTC methods modify the timing characteristics of transmissions, such as beacon or data frames, to relay information in the channel's energy from devices using one protocol to those using another. For example, FreeBee [8] shifts the timing of WiFi beacon frames to convey information, and C-Morse [6] slightly adjusts data frame timing as data frames are sent out to provide communication. Other adjacent technologies that focuses on polling the channel's energy focus on long range communication [12] or ultra-low power communication [13]. While these techniques are similar to our protocol solution, they have requirements that make large-scale inter-technology communication difficult such as requiring dedicated hardware [13] or not being reliable in noisy channels [7]. Our solution focuses on using capabilities of commodity-off-the-shelf (COTS) devices allowing more types of devices to be supported and coordinated.

Spectrum Management Techniques: The primary proposed alternative to our solution for regulatory spectrum management is having a centralized database, such as the one proposed by [14], that coordinates the CBRS bands [15] or the controller that enables 5G dynamic spectrum sharing [16]. While these techniques have proven to be effective within the scope they are deployed, they have limitations. First, they assume the devices are Internet connected and require a constant connection to a centralized node. Second, the spatial and temporal resolution of such solutions is quite coarse. Third, database approaches require the use of radio propagation models that can be overly constraining or incorrect. While this is not a problem for early deployments, as spectrum sharing gets more integrated into our devices, this will become a larger issue. A few examples of those limitations is how to handle mobile transmitters that are not aware of current zone restrictions [17], devices that wander into a RA telescope's interference range directly or indirectly through multipath or slidelobes, or devices that do not have the compute power to access a central database. These cases cannot always be handled by a centralized coordination node leading to a solution that needs to be ad hoc. Our decentralized solution allows us to coordinate with a wide range of devices that are spatially close together. To coordinate with a wide range of devices our solution communicates to the lowest ability of wireless devices, the lowest ability being able to poll energy on a channel. Our solution does not replace the role of a centralized database, but rather could work in conjunction to coordinate and mitigate a wider range of interfering devices.

Wideband Analog RFI Cancellation: A standard practice to remove strong RFI interference is through the use of beamforming and spatial nulling. For astronomical arrays, this is done in the digital back end. Digital beamforming for RFI mitigation has been demonstrated by many groups [18], [19]. This approach is well known and we assume that digital beamforming and spatial nulling is implemented in the final digital signal processing stages of the RA receiver.

The major remaining concern is high power range RFI that exceeds the dynamic range of the digital system. To mitigate high power RFI, analog suppression methods are needed before digitization. Wideband beamforming algorithms typically utilizing an FIR filter structure implemented on integrated circuits [20], [21] can reduce RFI in the analog domain before the ADC. The performance of these IC wideband

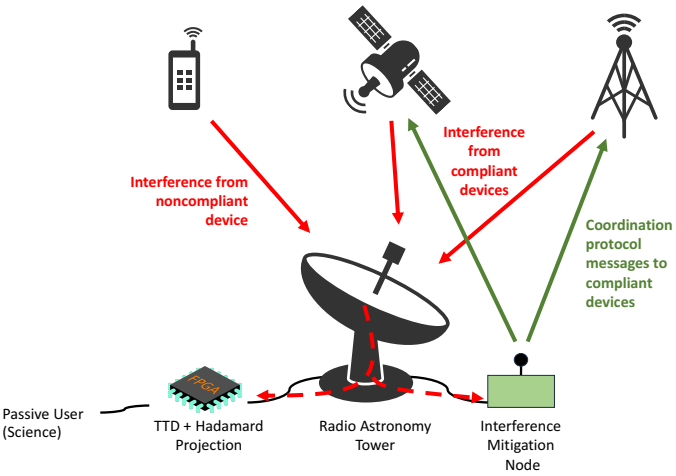

Fig. 1: System design with the three different components working together to share the spectrum with active users.

beamfomers depends on the analog filter taps in the IC, which adds complexity to the design. Another difficulty of the IC wideband beamformer is the calibration of the FIR weights, where the weights to form nulls are calculated based on simulations and are often inaccurate.

III. Design

Our multi-layer approach consists of three parts: a RF front end, a DSP algorithm, and a coordinating network protocol as all shown in Fig. 1. Each system has been designed to mitigate RFI in a unique but complimentary way. The first layer is the proactive coordinating network protocol. The protocol informs potential interfering devices of the RA receiver's frequency plan. Any interfering device that does not adhere to the protocol's request is then mitigated through the RF front end and DSP back end. Although each system has been designed specifically for RA receivers, they can be used for other applications as well. The following subsections detail each system in our multi-layer approach.

A. Coordinating Protocol

To be able to communicate the spectral needs of the RA receiver to surrounding wireless devices, we develop a coordinating protocol. The protocol allows an RA tower to communicate with widely varying of physical layer implementations making the protocol highly scalable to the number of devices and number of wireless protocols that interfere with RA receiver. To achieve this functionality, we focus on the fundamental capability of all receiving devices: the ability to measure the energy of a channel, often in terms of RSSI. To encode information in RSSI, we use a transmitter to modulate information in the energy of wireless channel that can be detected and decoded by devices that can poll the energy on the wireless channel.

Our protocol requires three entities: our protocol transmitter, interfering active spectrum users, and the RA receiver that is trying to receive signals from space that is experiencing interference. The transmitter is coupled with the RA receiver to enable the RA receiver to broadcast its spectrum listening needs either proactively or reactively. Proactively, the RA receiver informs active users of the duration of the listening cycle and the frequencies needed of an upcoming scheduled experiment. This is done before an observation is about to be made. Reactively, the RA receiver identifies interference and then informs the active users of the interference. If the interference is not mitigated, the data is "blanked", i.e. deleted. When data is blanked, that is wasted time and money, which our solution aims to minimize. The following subsections detail how each of these entities operate within our protocol.

1) Transmitter: To communicate with a wide range of devices, our protocol transmitter needs to be able to communicate to the lowest common denominator of receivers, the ability to measure the channel's energy level. Incidentally, energy levels of a channel are affected by any transmission. Our transmitter exploits this fact by using arguably the simplest modulation scheme, on-off keying (OOK), to manipulate the channel's energy level to encode data. To improve reliability of the transmissions, we use pseudo-random number (PN) codes to encode information, which is similar to direct-sequence spread spectrum (DSSS) techniques.

2) Receiver: Any device that has the ability to measure the channel's energy level can receive and decode one of our packets. A packet can be received and decoded by collecting the most recent channel energy level (RSSI) samples and correlating them against predetermined symbols. If the correlation value is higher than a set threshold, a symbol is detected. If a valid sequence of symbols is detected within an appropriate time span, the packet symbols are decoded into a spectrum sharing action the active user should take.

The advantage to the outlined approach is that it can be implemented completely in software and run on COTS hardware. This makes the receiver implementation easily deployable to devices with a software update. This is an important requirement since the receiver implementation needs to be widely deployable to minimize the number of non-compliant devices.

3) Protocol: We design the packet structure of our protocol to be reliable and use as little air time as possible to quickly coordinate with devices that would or are interfering with RA receivers. By doing so, we minimize both the time it takes to receive our protocol and how long interference persists. To minimize airtime while still being reliable, the building blocks of our protocol packets are sequences of PN codes. The PN codes we use are maximum length sequences (m-sequences). These sequences have favorable cross-correlation properties [22] making them easily detectable amongst busy channels. M-sequences have varying bit lengths of $2^N - 1$ where N is the number of linear-feedback shift registers. The individual bits within the sequence are called chips. Longer sequences have increased resilience to noise, but take longer to transmit. To minimize transmit time and have enough resilience to noise, we select an m-sequence size of 63 chips. While there are many m-sequences that have 63 chips, there

TABLE I: The meanings of the symbols under the different protocol contexts. The subscript denotes the symbol number and the superscript denotes whether the symbol is inverted or not.

Symbol	Duration (Min)	Frequency	Bandwidth (MHz)
S_0^+	5	0	10
S_1^+	10	1	20
S_2^+	20	2	40
S_3^+	40	3	80
S_4^+	60	4	160
S_5^+	90	5	320
S_0^-	120	6	640
S_1^-	180	7	-
S_2^-	240	8	-
S_3^-	300	9	-
S_4^-	360	10	-
S_5^-	PILOT	PILOT	PILOT

are only 6 orthogonal m-sequences. We make use of these 6 orthogonal sequences and their inverses to design our protocol.

With the constraint of minimized airtime and having 12 symbols to build our protocol, we design our protocol to give different meaning to a symbol based on its location with the packet structure. The packet structure consists of a sequence of eight symbols. Table I provides a comprehensive list of the different meanings of each symbol under the different symbol locations within a packet. The packet is broken into a preamble and a payload. The preamble constitutes the first two symbols which can be used for synchronization by the protocol receiver. After the preamble, the first symbol in the payload is an action time duration symbol. When an active user receives and decodes one of our packets, an appropriate spectrum sharing action needs to be performed for a specified amount of time. We acknowledge that the only spectrum sharing action that may be appropriate is turning off the interfering radio for the specified time. RA receivers listen to the spectrum from a few minutes to a few hours [23], dictating listening times with minute granularity. Following the duration symbol, the next four symbols constitute a center frequency in MHz that the RA receiver is requesting. The final symbol to be transmitted informs the active user of the bandwidth size the RA receiver needs to be free.

As shown in Fig. 2, if a RA receiver wants to listen to the frequencies between 5.84 and 5.93 GHz for 50 minutes, the RA receiver will create one of our packets with a duration of 60 minutes, a center frequency of 5.890 GHz, and a bandwidth of 10 MHz.

B. RF Front end

An alternative approach to the standard FIR wideband beamforming approach that utilizes time delays and a Hadamard projection has been shown to remove wideband RFI [24] in the analog domain to avoid saturating the ADC. It can also be cascaded with a downstream beamformer to

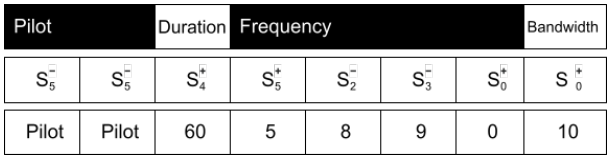

Fig. 2: Our protocol packet structure (top), an example packet with its corresponding symbol sequence (middle), and the meaning of the example packet (bottom).

increase the SNR post-ADC [25] in the analog or digital domain.

Our proposed RF system design is capable of removing two types of RFI sources that are not compliant with the coordinating protocol discussed earlier: a strong narrowband RFI source and a strong wideband RFI source. The RF chain consists of LNAs, mixers, filters, and two custom IC components. The first custom IC component is a reconfigurable filter with tunable notch bandwidth and center frequency that cancels a strong in-band narrowband RFI source. The second custom component is a combined TTD Hadamard projection IC chip. The RF system is shown below in Fig. 3.

The filter notch location in the CMOS IC chip is based on variable capacitors that control the notch frequency and bandwidth. In order to increase the notch depth, multiple filters can be cascaded together inside the CMOS IC chip to achieve a deeper null depth. When there is no strong narrowband RFI source within the frequency band of interest, the notch can be placed outside the band of interest to not interfere with the wideband SOI. This analog filter would be placed in front of the mixers and would operate on the RF signals before being mixed down to the IF frequency.

The true-time delay (TTD) Hadamard projection IC circuit forms a wideband spatial null down to the noise floor at a specific angle of arrival to the system. First, we consider the array response vector $\boldsymbol{v}$. The array response vector $\boldsymbol{v}$ is a column vector containing the collection of the time-shifted signal of interest and RFI source present at each port of the receiving aperture array. By applying a wideband time delay operator $\boldsymbol{D}$ on $\boldsymbol{v}$ tuned to a specific interference source based on its angle of arrival, we can align in time the targeted interference signal to look like a common bias on $\boldsymbol{v}$. Then by applying a Hadamard projection operator $\boldsymbol{H}_p$ the resulting response vector $\boldsymbol{v}_p$ will induce phase shifts to the narrowband signal of interest and remove the wideband interference signal from the remaining channels.

$$\boldsymbol{v}_p = \boldsymbol{H}_p \boldsymbol{D} \boldsymbol{v} \tag{1}$$

The time delay operator $\boldsymbol{D}$ has previously been demonstrated in [26] over four channels. The implementation of the Hadamard projection operator $\boldsymbol{H}_p$ has also previously been demonstrated in [4]. Ultimately we will combine both operations into a custom CMOS IC chip to operate on the signals coming in from the four channels at the IF frequency. The output of the TTD Hadamard Projection circuit is amplified and filtered before being sampled using the RFSoC ZCU216.

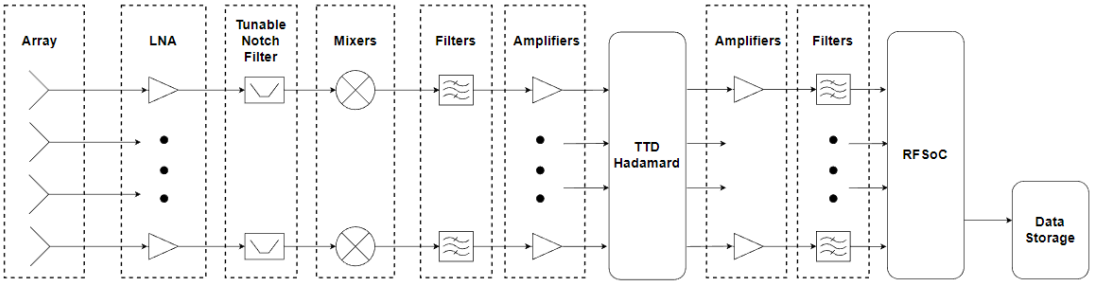

Fig. 3: Block diagram of proposed RF system, containing standard RF components, along with the two custom IC chips: narrowband notch filter and TTD Hadamard projection chip.

The time delay weights used to steer the spatial null toward the wideband RFI are controlled through DC pins coming from the RFSoC.

C. DSP

In our system, the RFSoC functions as the processing unit of the system, and has two main functions: controlling the RF front end, and sampling the output of the RF front end to perform further digital signal processing on the output such as narrowband beamforming to increase the SNR at the ADCs.

One novel feature of this architecture is the ability of the RFSoC to control the operation of the RF front end such as the LO frequency, narrowband notch filter location, and analog TTD values. The LO frequency used by the mixers is digitally generated using the DACs from the ZCU216 by employing a polyphase DDS [27] allowing the RFSoC to control the RF bandwidth of interest we want to analyze. The location of the tunable notch filter in the frequency domain is also controlled directly by the RFSoC through the DACs. The individual channel delays induced by the TTD block are controlled by DC pins coming from the RFSoC.

After sampling the signal at the end of the RF system, the RFSoC saves the data and passes the saved data into a digitally implemented spectrometer, allowing the operator to view the output of the RF system in the frequency domain in real time. After the data is saved, the RFSoC performs various estimation algorithms for improving the performance of the RF system, such as angle of arrival estimation. Because it is impossible to know the exact angle of arrival a priori, the RFSoC samples from the the RF front end and estimates the angle of arrival of the wideband RFI, and then calculates the needed time delays and subsequent DC voltages required to steer the wideband spatial null towards the wideband RFI.

IV. Implementation

We discuss the implementation of our multi-layer approach. The first layer of our approach is our coordinating protocol that is implemented on COTS and SDR hardware. If interference is ongoing after the coordinating protocol has been transmitted, the next layers of our solution work on mitigating the interference which consists of the RF front end notch filter and the DSP algorithm. The RF front end notch filter is packaged and connected to the rest of our RA receiver front end to remove narrowband RFI. An all-digital version of our DSP algorithm is implemented on an FPGA with RFNoC capabilities to remove wideband RFI.

A. Coordinating Protocol

We implement our coordinating protocol receiving logic on three wireless technologies to demonstrate its flexibility: WiFi, LoRa, and LTE. These protocols are used by a large portion of consumer electronics that would interfere with RA receivers. We use COTS hardware for the WiFi and LoRa receiving devices and a software-defined radio (SDR) for LTE. In addition to the receivers, we develop a transmitter that can accommodate a wide range of spectrum coordination scenarios on an SDR to evaluate our protocol on the three wireless technologies.

1) Transmitter: To evaluate the functionality and scalability of our protocol, our transmitter needs to be versatile and easy to use over a wide range of frequencies; RA receivers operate in subsets of frequencies beginning at 360 kHz to over 275 GHz [2]. Since most consumer electronics can operate in the range between 10 MHz and 6 GHz, we choose to focus on this range, although our protocol solution will work on other frequency ranges. We select the Ettus USPR X410 to be our transmitter. This radio supports an approximate frequency range of 10-7200 MHz. We use GNU Radio Companion (GRC) to create and modulate our packets. Using GRC and the RFNoC capabilities of the X410, we create a modulated signal that can be either narrow- or wide-band and uses between 0.5 MHz to 300 MHz of instantaneous bandwidth. These capabilities allows our transmitter to be flexible and powerful enough to accommodate many spectrum coordination scenarios.

2) WiFi: The COTS WiFi hardware we built our protocol on top of is the Atheros ATH9K chipset. The ATH9k chipset has capabilities [28] that provides physical and MAC layer information, like RSSI, to the Linux virtual debug file system in userspace. To configure and control the output of these capabilities, the chipset exposes a few control files. Most importantly, one of the files allow us to control how often new information is provided to userspace. We configure a sample rate that allows us to detect 5 ms symbol chips in real time. To enable sampling at this rate and for other configurations, we use a custom Python script.

Our WiFi implementation has two spectrum sharing options. The device can either shut off its radio or switch to another frequency band. We use the Linux command line program `nmcli` [29] to control different functionalities of the WiFi adapter than discussed in the previous paragraph, such as disabling the radio or switching to different frequency bands.

3) LoRa: We implement our protocol on the COTS Pycom LoPy4 board [30] specifically targeting its LoRa modulation capabilities. To make this board behave similarly to the Atheros WiFi hardware in terms of RSSI polling, we make a minor addition to the default Pycom Python bindings exposing its internal RSSI polling. Our program can reliably poll the channel's energy with intervals as low as 500 μs. For consistency and in order to avoid excess load on the processor, we sample at 5 ms intervals. Because of hardware limitations, the LoPy4 board cannot report valid RSSI values during active transmissions. However, because of the LoRaWAN 1% duty cycle [31] for many devices, this only minimally affects the device's response for most realistic scenarios. Unlike the WiFi implementation, our LoRa receiver does not have another frequency it could switch to upon receiving our packet. Instead, after receiving our packet, the device is put to sleep for the specified amount of time.

4) LTE: To avoid broadcasting on licensed cellular frequencies, our LTE implementation uses srsRAN, a 4G/5G SDR implementation of the LTE protocol stack [32]. The use of an SDR enables us to transmit on ISM bands instead of cellular bands. Additionally, srsRAN provides a lot of real time information about the physical layer. However, the default provided RSSI value for srsRAN is only reported in conjunction with a valid LTE packet reception. To access the signal strength readings regardless of a LTE packet reception, we modify the srsRAN source code. We access the received I/Q sample buffer and calculate an RSSI value once every millisecond.

We use a pair of Ettus USRP B200minis to simulate a user equipment (UE) device and an evolved Node B (eNodeB). The two Ettus B200minis run on separate computers and connect wirelessly in the 2.4 GHz ISM band. To ensure connectivity, `iperf3`, a command line network throughput evaluation application, is used between the UE to the eNodeB. While srsRAN takes care of the active connection, a userspace program reads and processes the RSSI values from the srsRAN application looking for one of our protocol packets. When a one of our packets is received by the UE device, the UE device turns off. Consumer LTE devices can avoid turning off their radios by simply switching frequencies or enabling airplane mode.

B. RF Front End

If the coordinating protocol is not able to eliminate the interference, the RFI mitigation layers of our approach are used. The ultimate goal of these layers is to deploy a RF front end with a tunable notch filter for narrowband non-compliant RFI sources and an analog implementation of the Hadamard projection with analog true-time delay chips for wideband interferers. This would allow high dynamic range RFI to be mitigated before digitization and DSP.

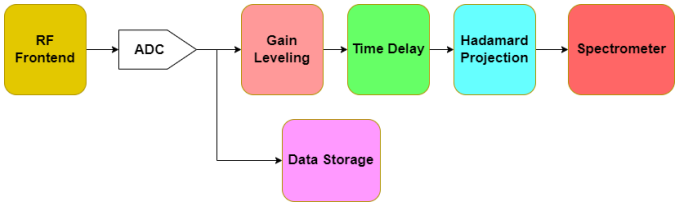

Fig. 4: Block diagram of digital system, containing the gain leveling, time delay, Hadamard projection, and spectrometer blocks.

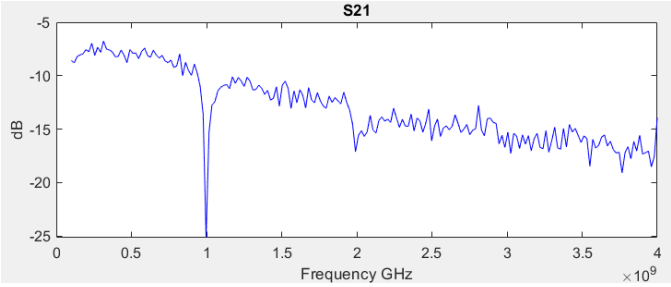

Fig. 5: Measured S parameters of packaged notch filter.

Preliminary testing of the IC notch filter chip measures the S-parameters of the IC notch filter using a probe station and a network analyzer. The results of the S_{21} from the 2-stage cascaded filter shows a clear 25 dB notch present spanning 100MHz at the target frequency shown in Fig. 5. After the IC chip was packaged and connected to the rest of the RF front end system, RF leakage occurred on the notch filter between the output RF channel and the input clock signal used by the notch filter, resulting in sharp periodic tones at the output of the notch filter making it unusable. This will need to be addressed in a future revision of the notch filter chip design.

C. DSP

Although the TTD Hadamard algorithm was intended to be located in the analog signal chains before ADCs, due to chip fabrication delays, an all-digital implementation is developed using the ZCU216 RFSoC to test the Hadamard algorithm and verify its performance with over-the-air signals. The block diagram of the digital implementation is shown in Fig. 4. The RFSoC performs the delay operator $\boldsymbol{D}$ in the digital hardware using an interpolation block, and the Hadamard operator on the quantized digital signals. The signals are then fed into a digital spectrometer to analyze the frequency spectrum of the output of the algorithm.

There are many different methods that a digital time delay can be implemented using interpolations. For the purpose of testing the time delay operator, $\boldsymbol{D}$ is created using a 3rd-order polynomial interpolation. In the digital hardware, this is done using a Vandermonde matrix structure to improve latency, and reduce the needed hardware to implement the interpolator.

The Vandermonde matrix $\boldsymbol{V}$ takes the following form

$$\underbrace{\begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & x_m^2 & \cdots & x_m^n \end{bmatrix}}_{\boldsymbol{V}} \tag{2}$$

where we assume regularly spaced samples occur every T_s seconds at normalized sample times $t_0 = -1$, $t_1 = 0$, $t_2 = 1$ and $t_3 = 2$. The Vandermode matrix elements are then replaced by the normalized time samples of t. The interpolation values $I(t_i)$ from the points $\{(t_0, y_0), (t_1, y_1), \ldots, (t_m, y_m)\}$ using the Vandermonde matrix will be

$$I(\tau) = \boldsymbol{q}\boldsymbol{V}^{-1}\boldsymbol{s} \tag{3}$$

where $\boldsymbol{q}$ forms a vector of powers of the desired fractional delay value τ

$$\begin{bmatrix} 1 & \tau & \tau^2 & \cdots & \tau^m \end{bmatrix}^T \tag{4}$$

and $\boldsymbol{s}$ forms the vector of inputs to the delay block

$$\begin{bmatrix} y_0 & y_1 & \cdots & y_m \end{bmatrix}^T \tag{5}$$

which functions as a FIFO for the input data on the digital channels. The multiplication of the $\boldsymbol{q}$ and the inverse Vandermonde matrix $\boldsymbol{V}^{-1}$ can be pre-computed and pipelined into the FPGA, making the evaluation of the $\boldsymbol{s}$ occur in real time. For example, our four channel $\boldsymbol{V}^{-1}$ digitally will be

$$\boldsymbol{V}_4^{-1} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{1}{3} & -\frac{1}{2} & 1^2 & -\frac{1}{6} \\ \frac{1}{2} & -1 & \frac{1}{2} & 0 \\ -\frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \end{bmatrix} \tag{6}$$

In this way the cubic polynomial interpolation can be performed using only 11 multiplication and 12 addition operations, with a total latency of 10 clock cycles. When operating with four channels, there is a total of 48 addition blocks, with 44 multiplication blocks.

The calibration of the time delay weights $\boldsymbol{q}$ is done by assuming the dominant RFI source is the RFI source we wish to cancel. If the RFI is wideband, a narrowband assumption on the RFI is assumed for the calibration step, where the narrowband RFI is selected to be in the bandwidth of the wideband RFI. In order to find the time delay weights to remove the RFI, the signal correlation matrix R_Ω is estimated. The eigenvector v_Ω corresponding to the largest eigenvalue from R_Ω is selected. The time delay weights $\boldsymbol{q}$ are then estimated as

$$\boldsymbol{q} = \frac{-\arg(v_\Omega)}{\omega} + b \tag{7}$$

where b is a bias term to force the time delays to be positive numbers. The minimum bias b_{min} required can be solved mathematically, where $b_{\text{min}} = \min(\boldsymbol{q})$. However, to avoid extra computation the bias term b can be set as a constant value based on the maximum angle of arrival of the RFI, thus reducing the computation needed to run the algorithm in real time.

In theory, each antenna is identical, and the array has no mutual coupling or different losses on each array channel. In real systems, there will always be variations of the signal on the array channels. To compensate for the variations in the channels, a digital gain leveling block is inserted before the digital TTD block which attempts to digitally level all channels in order to increase the spatial null depth we can form. The maximum likelihood estimate of a sinusoidal amplitude $\hat{A}$ is used to determine the amplitude at each channel given as

$$\hat{A} = \frac{2}{N}|X(\hat{\omega})| \tag{8}$$

where $\hat{\omega}$ is already known since we can easily estimate the frequency of the non-compliant RFI by inspection of the power spectral density, or by a priori knowledge of the interference sources. If the interference source is wideband, a narrowband assumption is used where $\hat{\omega}$ lies within the interference bandwidth. Once all the channel gains are known, the channel gains $\boldsymbol{g}$ to digitally level the ADC counts can also be calculated. The channel leveling operator $\boldsymbol{G}$ is found by diagonalizing the leveling coefficients, where $\boldsymbol{G} = \text{diag}(\boldsymbol{g})$. We can now model the output of the Hadamard algorithm $\boldsymbol{v}_p$, which has the wideband RFI removed, as

$$\boldsymbol{v}_p = \boldsymbol{H}_p\boldsymbol{D}\boldsymbol{G}\boldsymbol{v} \tag{9}$$

The Hadamard block in the digital hardware is likewise created using addition blocks. The four-channel implementation that is used for testing the TTD Hadamard algorithm uses an additional 8 blocks per channel, yielding a total of 56 addition blocks over the entire design. The output of the Hadamard block is then fed directly into the spectrometer for real-time analysis.

V. Evaluation

We evaluate each part of our solution separately and then integrate together. We demonstrate the coordinating protocol has minimal impact on native transmissions and works with a variety of protocols on different frequencies. We evaluate the performance of the RF front end with the digital system. Lastly, we evaluate the full multilayer integrated system by having our protocol remove compliant devices and the Hadamard algorithm remove the uncompliant RFI while preserving a signal of interest.

A. Native Network Impact

With normal device communication and our packet transmissions potentially coinciding in time and frequency, it is easy to ask how our protocol affects the primary channel of communication and how the primary channel affects our protocol. To examine this duality, we use two WiFi nodes that support our protocol and the `iperf3` application. Using `iperf3`, we create a client-server connection between the two nodes to simulate network traffic in conjunction with interference from many other wireless devices already communicating on the same frequency within the same physical space. The first node is a wired server and the second node is the

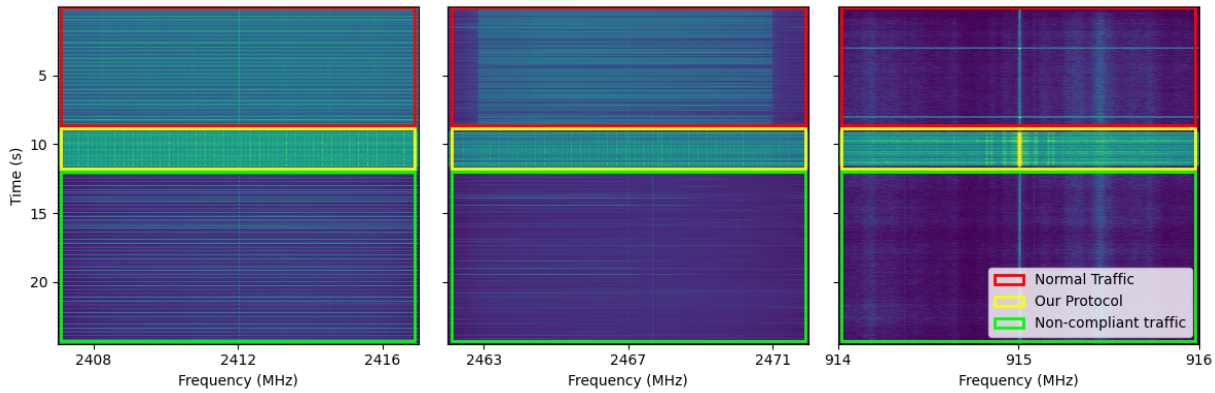

Fig. 6: Spectrograms demonstrating three distinct wireless protocols receiving and acting on a coordinating packet. Left: WiFi, Middle: srsRAN, Right: LoRa. The top portion encircled in red is the normal device communication. The yellow box is the transmitted coordinating packet. The bottom green portion is other non-compliant interfering traffic that is on the same frequency that needs other techniques to mitigate.

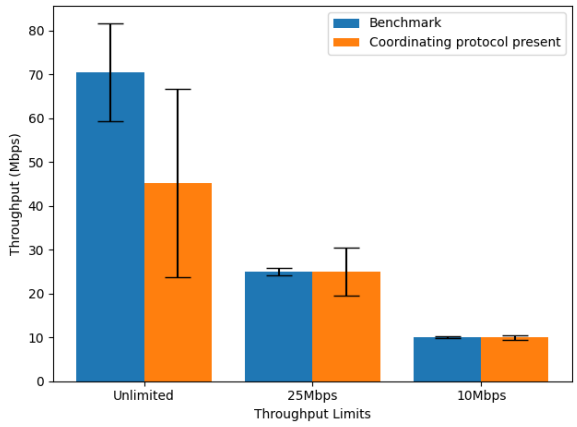

Fig. 7: Average throughput with and without coordinating packets being transmitted.

wireless client. The client connects to the server via a wireless router and sends UDP packets at different throughput values. Only the second node in this configuration is running the our protocol detection algorithm. The spectrum sharing action feature is disabled for this test to allow capturing multiple of our packets in a single test. We run two sets of tests: a benchmark test where no coordinating packets are transmitted and one where 10 coordinating packets are transmitted. Each test is repeated 5 times for each of the selected throughput rates, which are unlimited, 25 Mbps, and 10 Mbps.

1) Affect of Network Traffic on Our Protocol: Our modified WiFi node is able to fully decode all but one out of the 150 coordinating packets transmitted under all the different network traffic configurations. This demonstrates that our protocol works despite the amount of network utilization from other active wireless transmissions. The use of m-sequences in the protocol's design allows the signal to be detected regardless of other transmissions. In addition, our transmitter does not adhere to any channel sensing scheme and instead transmits whether or not the channel is busy. This shows that our design will work even if the channel is congested.

2) Affect of Our Protocol on Network Traffic: Our protocol uses active transmission, which will affect the throughput of other WiFi transmissions. Fig. 7 shows the impact of our protocol on other network traffic. Our protocol has little to no average effect when the traffic limit is set to 25 Mbps and 10 Mbps, which are realistic scenarios.The unlimited throughput configuration, a typically unrealistic scenario, shows an average drop in throughput of 36% when our protocol is being used compared to when it is not. This results show expected behavior. WiFi's CSMA mechanism prevents WiFi transmissions while the channel is busy. Our transmissions put energy into the spectrum that WiFi devices detect and will not transmit at that time. It is important for our protocol to not be totally disruptive to the interfering transmitters. They may need to transmit a higher priority communication that will need to interrupt RA receivers regardless of the RA receiver requests, communications such as emergency communications. These communications, under normal throughput usage, will experience no impact and ignore the protocol. Employing a multilayer approach in these scenarios is advantageous. Where the coordinating protocol cannot shutdown the transmitter, the Hadamard projection can notch out the interference.

B. Multi-device Coordination

We implement the detection of our protocol on top of three vastly different hardware sets: WiFi, LoRa, and LTE. In this experiment, we show that we can coordinate with all three device types from one transmitter. For simplicity of setup we place our protocol transmitter approximately one meter from the LoRa, WiFi, and srsRAN devices. The LoRa device periodically transmits data every 5 seconds to another LoRa device across the room. The WiFi device is connected to our access point which operates on channel 1 (2.412 GHz center frequency). The uplink frequency for the srsRAN devices is set

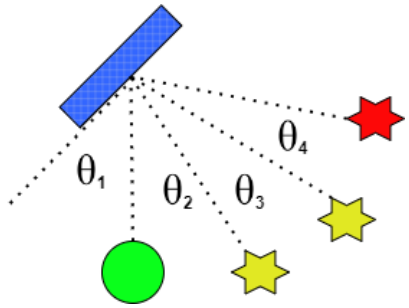

Fig. 8: Roof testing setup. Blue: Phased array and DSP backend. Green: signal of interest, with $\theta_1 = 60°$. Yellow: Compliant RFI sources, $\theta_2 = 20°$ and $\theta_3 = 10°$. Red: Non-compliant RFI, $\theta_4 = 15°$.

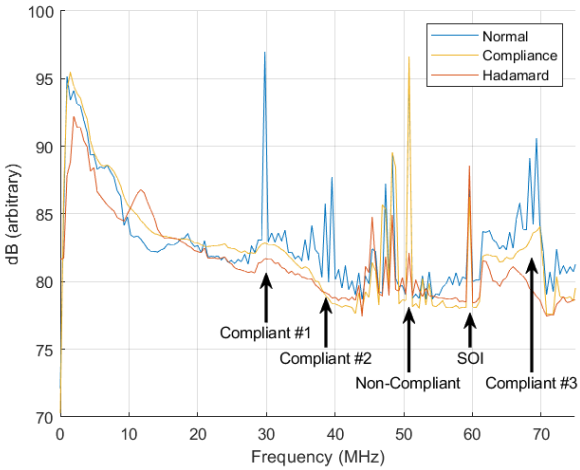

Fig. 9: Roof test results. The normal frequency spectrum before employing UWP and the TTD Hadamard algorithm is shown in blue. The spectrum after UWP is shown in yellow. The spectrum with the non-compliant RFI removed is shown in red. Compliant RFI sources are located at 30MHz, 40MHz, and 70MHz are turned off through UWP. Non-compliant RFI removed by the TTD Hadamard algorithm is located at 50MHz. The signal of interest located at 60MHz is preserved by the Hadamard algorithm for further downstream processing.

to the center frequency of WiFi's channel 12 (2.467 GHz). The WiFi and srsRAN devices use `iperf3` to simulate network traffic.

The spectrogram plots found in Fig. 6 show the normal spectrum use of the individual devices between 0 and 9.5 seconds as denoted by the red boxes in the figures. It is to be noted that the LoRa sub-figure does not exhibit the tell-tale chirp-spread spectrum pattern that is normally associated with LoRa because of the time scale in the figure. At 9.5 seconds a coordinating packet is transmitted. For LoRa, the packet is transmitted on 915 MHz, and for the other devices the packet is transmitted on channel 6 (2.437 GHz center frequency). After the devices receive and decode our packet, they take appropriate spectrum sharing actions. The LoRa and srsRAN devices merely shutoff their radios for the rest of the experiment. The WiFi device, on the other hand, switches to the access point's 5 GHz interface and resumes transmitting its normal network traffic. From this experiment, it is evident that our protocol can communicate with many different devices and coordinate the use of spectrum. Our protocol is dynamic in its bandwidth usage and the devices it can be received on.

This experiment also shows that there are non-compliant devices on the channel that did not adhere to any spectrum sharing schemes. The interference caused by these devices needs to be mitigated through other means which is demonstrated in the following Section V-D.

C. System Cable Testing

We perform cable testing to verify the performance of the RF front end in conjunction with the digital system implementation. For testing purposes, a signal generator and a 4:1 splitter are used to simulate the RFI coming directly boresight to the array. Due to the angle of arrival of the RFI in the cables, no time delay is required, and we can determine the maximum null depth we can form under ideal conditions. We can also test the calibration of the time delay weights (they should be zero) and test the impact that the gain leveling block has on the overall notch depth we can form.

When the gain leveling block is bypassed in the digital system, we are only able to form a notch of 1-2 dB over all frequencies as we sweep over the RFI frequency. However, once we enable the gain leveling block, we are able to form a maximum notch of 36 dB over the IF frequencies from 0-250MHz. The time delay weight calibration accurately estimates the angle arrival is exactly $0°$ from boresight of the array.

D. Over The Air Testing

We perform a coordinated, over the air test outside on a roof. Our test setup is shown in Fig. 8. We use Ettus B100mini and Ettus N210 SDRs as the compliant and non-compliant RFI sources, and as the signal of interest. The down-converted frequency spectrum at the output of the RF front end is shown in Fig. 9.

Before the proposed protocol sends a message to all compliant active transmitters, the frequency spectrum at the input to the DSP backend is shown in Fig. 9 in blue. After our protocol is transmitted to all receivers, the compliant receivers turn off, leaving only the non-compliant transmitter and signal of interest at 50 MHz and 60MHz, respectively, shown in yellow. At this point, the calibration previously mentioned above is used to tune the time delay weights in the DSP backend to remove the non-compliant RFI transmitter. After the application of the TTD Hadamard algorithm, the spectrum shown in red shows that the non-compliant RFI source is attenuated by 15 dB to the noise floor of the system, which is the theoretical maximum depth we can lower the interference source. The signal of interest has been preserved for further downstream processing. While the signal is preserved, SNR is

critical for RA receivers. In future work we plan to assess the SNR degradation after Hadamard projection.

VI. Conclusion

Spectrum sharing with passive users has become an important issue that needs to be dealt with in a holistic way. Our solution takes three synergistic approaches to spectrum coordination. Our coordinating protocol allows for a RA observatory to transmit out a coordination message for all potential nearby active devices to receive. We can send this coordinating protocol both proactively, before an observation, and reactively, if interference is detected and blanking needs to occur. For non-compliant active users, we implemented a passive technique to remove wideband RFI based on a true-time delay Hadamard projection algorithm. In our tests, the Hadamard projection was implemented digitally, but in future work it could be implemented in the analog domain along with tunable notch filters to prevent ADC saturation for high dynamic range interferers. Using experimental over the air tests, we demonstrate these techniques working together to show that they are effective at removing RFI.

References

[1] Cisco, "Cisco annual internet report - cisco annual internet report (2018–2023) white paper," Jan 2022.

[2] I. T. Union, *CHAPTER II Frequency*, 2020th ed., ser. Radio Regulations. Geneva, Switzerland: International Telecommunication Union, 2020, vol. 1, p. 62–63.

[3] Di Vruno, F., Winkel, B., Bassa, C. G., Józsa, G. I. G., Brentjens, M. A., Jessner, A., and Garrington, S., "Unintended electromagnetic radiation from starlink satellites detected with lofar between 110 and 188 mhz," *A&A*, vol. 676, p. A75, 2023. [Online]. Available: https://doi.org/10.1051/0004-6361/202346374

[4] S. Jang, R. Lu, J. Jeong, and M. P. Flynn, "A true time delay 16-element 4-beam digital beamformer," *2018 IEEE Radio Frequency Integrated Circuits Symposium (RFIC)*, 2018.

[5] J. W. Kunzler and K. F. Warnick, "Wideband radio frequency interference cancellation for high-sensitivity phased array receivers with true time delays and truncated hadamard projection," *IEEE Transactions on Antennas and Propagation*, vol. 70, no. 9, pp. 8069–8076, 2022.

[6] Z. Yin, W. Jiang, S. M. Kim, and T. He, "C-morse: Cross-technology communication with transparent morse coding," in *IEEE Conference on Computer Communications*. Atlanta, GA, USA: IEEE, 2017.

[7] K. Chebrolu and A. Dhekne, "Esense: Communication through energy sensing," in *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 85–96.

[8] S. M. Kim, S. Ishida, S. Wang, and T. He, "Free side-channel cross-technology communication in wireless networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 5, pp. 2974–2987, Oct. 2017.

[9] Z. Li and T. He, "Webee: Physical-layer cross-technology communication via emulation," in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 2–14.

[10] R. Liu, Z. Yin, W. Jiang, and T. He, "Wibeacon: Expanding ble location-based services via wifi," in *The 27th Annual Inter-national Conference On Mobile Computing And Networking*, ser. ACM MobiCom '21. New York, NY, USA: ACM, 2021.

[11] X. Guo, Y. He, J. Zhang, H. Jiang, Z. Yu, and X. Na, "Taming the errors in cross-technology communication: A probabilistic approach," *ACM Transactions on Sensor Networks*, vol. 18, no. 1, Oct. 2021.

[12] P. Lundrigan, N. Patwari, and S. K. Kasera, "On-off noise power communication," in *The 25th Annual International Conference on Mobile Computing and Networking*, ser. ACM MobiCom '19. New York, NY, USA: ACM, 2019.

[13] B. Kellogg, A. Parks, S. Gollakota, J. R. Smith, and D. Wetherall, "Wi-fi backscatter: Internet connectivity for rf-powered devices," in *Proceedings of the 2014 ACM Conference on SIGCOMM*, ser. SIGCOMM '14. New York, NY, USA: ACM, 2014, pp. 607–618.

[14] PAWR, "Pawr announces successful demonstration of intelligent spectrum sharing," 2022, last accessed 28 July 2022.

[15] FCC, "Citizens broadband radio service rules," 2015.

[16] "Dynamic spectrum sharing: How it works & why it matters."

[17] C. Beaudet, P. Woody, W. Sizemore, R. McCullough, J. Ford, F. Ghigo, C. Niday, and C. Clark, "The green bank interference protection group: Policies for rfi management," 2007.

[18] W. A. Baan, "Implementing RFI mitigation in radio science," *Journal of Astronomical Instrumentation*, vol. 8, no. 01, p. 1940010, 2019.

[19] R. A. Black, B. D. Jeffs, and K. F. Warnick, "Deep, broad null formation for canceling moving RFI in radio astronomical arrays," *Journal of Astronomical Instrumentation*, vol. 8, no. 01, p. 1940003, 2019.

[20] S. Park, L. Huang, and S. Raman, "A wideband beamformer using 3.25 gs/sec discrete-time analog fir filter ics," in *2019 49th European Microwave Conference (EuMC)*, 2019, pp. 260–263.

[21] A. Madanayake, V. Ariyarathna, N. Udayanga, L. Belostotski, S. K. Perera, and R. J. Cintra, "Design of a low-complexity wideband analog true-time-delay 5-beam array in 65nm cmos," in *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2017, pp. 1204–1207.

[22] D. Sarwate and M. Pursley, "Crosscorrelation properties of pseudorandom and related sequences," *Proceedings of the IEEE*, vol. 68, no. 5, pp. 593–619, 1980.

[23] Green Bank Observatory, "GBT schedule," 2022.

[24] E. Ghaderi, A. S. Ramani, A. A. Rahimi, D. Heo, S. Shekhar, and S. Gupta, "Four-element wide modulated bandwidth MIMO receiver with 35-dB interference cancellation," *IEEE Transactions on Microwave Theory and Techniques*, vol. 68, no. 9, 2020.

[25] J. Kunzler and K. Warnick, "Wideband radio frequency interference cancellation for high-sensitivity phased array receivers with true time delays and truncated hadamard projection," *IEEE Transactions on Antennas and Propagation*, vol. 70, 2022.

[26] E. Ghaderi, A. S. Ramani, A. A. Rahimi, D. Heo, S. Shekhar, and S. Gupta, "An integrated discrete-time delay-compensating technique for large-array beamformers," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 9, 2019.

[27] J. Bartschi, "Development of compact phased array receivers on rfsoc prototyping platforms," *BYU ScholarsArchive*, 2022.

[28] L. Wireless, "Linux wireless - ath9k spectral scan," 2022.

[29] NetworkManager, "nmcli," 2022.

[30] Pycom, "Pycom lopy4," Aug 2022.

[31] L. A. T. C. R. P. Workgroup, "Rp002-1.0.2 lorawan regional parameters," 2020.

[32] srsRAN, "srsran: Open source sdr 4g/5g software suite from software radio systems (srs)," 2022.