Detecting Ultrahigh Energy Neutrinos with Novel Technologies for the PUEO Experiment

Cheng Xie

A dissertation submitted for the degree of
Doctor of Philosophy

Department of Physics and Astronomy
University College London

September 2024

Declaration

I, Cheng Xie, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work.

Abstract

Neutrinos with energies >10^{17} eV provide a unique source of evidence for new physics, but none have yet been detected. The Payload for Ultrahigh Energy Observations (PUEO) is a balloon-borne radio instrument under development as a direct successor to the Antarctic Impulsive Transient Antenna (ANITA) programme that placed the leading limits for ultrahigh energy neutrino flux between $10^{19.5}$-10^{21} eV.

Contributions are presented in this thesis relating to the digital beamforming trigger that provides significantly improved neutrino sensitivity for PUEO. The first software simulation of the PUEO trigger hierarchy was developed, demonstrating the benefit of this phased-array trigger and identifying design decisions corresponding to ~11% improvement in sensitivity along the energy axis. A proof-of-concept trigger was also developed using programmable hardware, achieving cross channel synchronisation to within ~10ps, which satisfies PUEO requirements. As well as informing PUEO's design, the hardware trigger developed has been deployed since May 2023 as a part of the Radio Echo Telescope for Cosmic Rays (RET-CR).

Impact Statement

The results of the software simulation and the hardware design both contribute toward PUEO's development, which is part of the experimental effort in ultrahigh energy neutrino detection that is over time "guaranteed" to identify new physics. Sensitivity improvements will result in either the first ever detection of ultrahigh energy neutrinos, or improved constraints that would become in tension with neutrino flux expected from the interaction of ultrahigh energy cosmic rays with cosmic microwave background radiation. These measurements can lead to improved understanding of the fundamental laws of the universe. Advancements in the latter have in the past led to the development of novel technologies, with impact in varied areas such as medicine, energy, economic growth and space exploration.

More immediately, the work on digital hardware design can be useful for those who are new to programmable hardware and especially the Radio Frequency System-on-Chip (RFSoC) platform. This is a broadly applicable technology across areas such as satellites, 5G communication and radar systems, as evident in the $35 billion acquisition of their original developer Xilinx in 2022 by AMD. With limited manufacturer guidance on crucial configurations such as the clocking system, this work can provide a prototype for researchers and commerical users in a wide number of scenarios.

Results from optimising the digital beamforming algorithm can also be useful in other applications where signals are combined in a phased array with the aim of increasing sensitivity to small signals. Results relating to the performance impact of the digital dynamic range, and the use of band-limiting digital filters to further improve detection sensitivity, have particularly general applicability.

Contents

List of Figures

Chapter 1

The Standard Model and Ultrahigh Energy Neutrino Physics

1.1 The Standard Model

The Standard Model (SM) of particle physics [10] is the best theory currently available for describing all known elementary particles and three of the four known fundamental forces: electromagnetic, weak and strong. Through a quantum field theory framework, it classifies and describes the 6 quarks, the 6 leptons, the 3 bosonic force carriers and the Higgs boson. Predictions can be carried out using the Standard Model Lagrangian, defined by gauge symmetries, with dynamics that depend on experimentally evaluated parameters.

While some of its predictions have been tested to very high precision, the SM is not yet a complete theory of fundamental interactions, leaving a number of phenomena unexplained, including:

Gravity The theory of general relativity is incompatible with the Standard Model [11]. While the SM is built on a quantum field theory framework, there has been no consistent quantum field theory of gravity, the fourth known fundamental force.

Dark Matter Together with the result of general relativity, a wide variety of cosmological observations are inconsistent with only the known fundamental particles of the SM. The widely accepted solution to this is an additional, hypothetical form(s) of matter that does not interact with the

electromagnetic field [12]. The accelerating expansion of the universe also cannot be accounted for, which is a problem that is referred to in terms of dark energy.

Matter Asymmetry There is no mechanism in the SM to sufficiently explain the baryogenesis required to be consistent with the much greater prevalence of matter over antimatter observed in the universe [13].

Neutrino Mass Neutrino masses must be nonzero to be consistent with experimentally observed neutrino oscillations [14]. Unlike the previous phenomena, this can be accounted for by one of several feasible extensions to the SM, and these are likely to be narrowed down in the near future. They have not yet been incorporated in the SM, due to the sensitive measurements required to determine the values of these masses, and the mechanism (Dirac vs. Majorana) in which the mass is incorporated in the theory [15].

Other Areas Outside of these areas, there are only a few specific experimental observations that are in tension with the SM. And while these may point to new physics, they may also have other explanations including experimental errors, statistical flukes arising from the large number of experiments, and uncertainties in applying known physics to produce accurate predictions. One such result is the magnetic dipole moment of the muon [16], where the combined results of the first three years of data-taking at the Muon g-2 experiment [17] found a 5.1 sigma deviation from the SM theory prediction used by the study. However, the theoretical prediction used for this result relies on dispersion relations for the hadronic vacuum polarisation contributions [18], and the use of alternative theoretical calculations that rely on lattice QCD removes the tension with the experimental result [19]. Work is ongoing to clarify the theoretical discrepancy.

Another example is ANITA's observation of upward-going cosmic-ray-like events [20]. Without the inversion expected from reflected events, these events could be hypothesised as arising from the decay of emerging τ leptons generated by ν_τ interactions beneath the ice surface. However, such events would be anomalous as the SM neutrino cross section would be expected to attenuate the flux by a factor of 10^{-5}. This could then imply either a strong transient flux, or a novel suppression of the SM cross section. These explanations are strongly constrained although not completely ruled out by the non-detection of signals expected from similar sources at IceCube [21]. In addition, explanations have been suggested that do not require upward going particles, such as the reflection from subsurface features of downgoing signals [22] as well as coherent transition radiation [23].

There are also a number of theoretical arguments that are used to motivate beyond the Standard Model (BSM) physics. These include fine tuning / the hierarchy problem [24] and the strong CP problem [25].

A large variety of extensions to the Standard Model have been proposed for dark matter and mattery asymmetry, with some parameter space for each theory that could be consistent with observation. As increasing parts of these phases spaces are ruled out with recent generations of experiments, it is hoped that data from current and future experiments will be able to pick up additional BSM signatures that can be positively identified with some of these extensions.

Terrestrial particle accelerator experiments at ever higher energies have previously yielded much of the novel experimental signatures that motivated and validated the Standard Model. The Large Hadron Collider (LHC), the highest energy man-made particle accelerator, began its third run in July 2022 [26] and the High-Luminosity LHC upgrade [27] is expected to increase instantaneous luminosity by a factor of five.

Looking further in the future, there are proposals for new colliders that will be sensitive to novel physics at higher energies and greater sensitivity. This includes the Future Circular Collider [28], a circular collider with centre-of-mass energies of up to 100 TeV. There is a choice of three possible design options for the types of collision, namely hadron-hadron, electron-positron, or electro-hadron. On the other hand, the International Linear Collider [29] and the Compact Linear Collider [30] proposals benefit from the precision measurements possible with linear collisions of electrons and positrons.

The development of colliders of the high energy frontier is increasingly costly, and they are complemented with other innovations. BSM signals may come from using novel detection methods, experiments of extremely high precision, or through novel signals. Of the latter, both gravitational waves [31] and high energy neutrinos with energies $10^{14} - 10^{15}$ eV [32] have recently been detected for the first time.

1.2 Beyond the Standard Model with Ultra-high Energy Neutrinos

Ultrahigh energy (UHE) neutrinos are of astrophysical/cosmic origin, and have energies of $\geq 10^{17}$ eV. They represent a novel messenger for astrophysics,

complementing electromagnetic radiation and gravitational wave signals [1]. At the same time, they are also a source of particles at the high-energy frontier of particle physics. Their high energies mean that neutrino-nucleon interactions would have centre-of-mass energies greater than that of terrestrial colliders, providing another opportunity to probe new physics.

In addition, neutrinos are weakly interacting and are therefore expected to have travelled long baselines, potentially from their origin of production without interactions along the way. This allows small effects, arising for example in oscillations or hidden sectors, to accumulate over cosmic distances. This can then be measured in terms of properties such as flavour ratios, energy spectrum, arrival time and spatial distribution. Because of their lack of interactions, it is expected to be possible to evaluate their direction and therefore source. This information can be combined with other signals from the same source, to enable multi-messenger analyses on e.g. the arrival time of the neutrinos compared to photons.

Figure 1.1 shows a variety of BSM physics models, outlining where new physics enters during the neutrino's journey to the detector, and the physical properties that are sensitive to the new physics.

UHE neutrinos have not been detected to date. However, there are several motivations for their existence, closely linked to cosmic rays of very large energies:

Astrophysical neutrinos UHE and extreme energy (EE) cosmic rays have been detected at energies of up to 3×10^{20} eV [33]. Their origin and composition remain open questions, but they are expected to be produced through extreme astrophysical acceleration mechanisms, and to consist mostly of either protons or heavy nuclei. Proton-photon interactions involving these cosmic rays can produce charged pions (alongside the production of neutral pions) that produce neutrinos upon decay:

$$p + \gamma \to \pi^+ (\text{or } \pi^-) + n \tag{1.1}$$
$$\pi^+ \to \mu^+ + \nu_\mu \tag{1.2}$$
$$\to e^+ + \nu_\mu + \nu_e + \bar{\nu}_\mu \tag{1.3}$$
$$\pi^- \to \mu^- + \bar{\nu}_\mu \tag{1.4}$$
$$\to e^- + \nu_\mu + \bar{\nu}_e + \bar{\nu}_\mu \tag{1.5}$$

The sources of cosmic rays are therefore expected to also produce neutrinos that typically carry 3-5% of the parent proton energy. Protons can

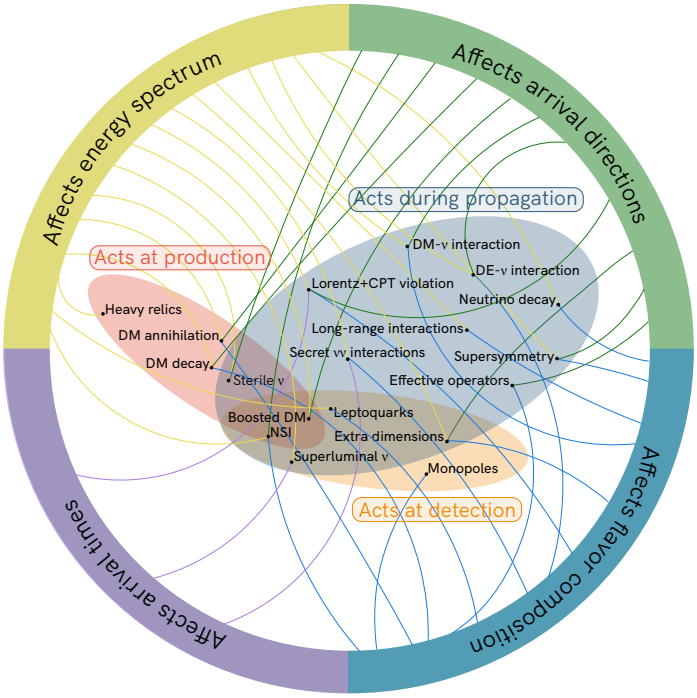


Figure 1.1: Sensitivity to new physics models can arise from interactions at the production, propagation, or detection of the UHE neutrino. These effects can be measured through a number of number of different experimental signatures - including the arrival direction, arrival time, flavour composition, and energy spectrum. Plot reproduced from [1].

also interact with surrounding matter to produce UHE neutrinos, again via charged pions.

Cosmogenic neutrinos In addition to production at astrophysical sources, highly energetic cosmic rays are expected to interact with photons of the cosmic microwave background to produce particles including UHE neutrinos via the GZK process [33], in a process with the same steps as equations 1.1 to 1.5. Since UHE and EE cosmic rays have been detected, if UHE neutrinos are not detected with a low enough limit placed, it would represent new physics in terms of the composition of cosmic rays or of fundamental interactions at these energies.

High energy (HE) neutrinos detection Finally, it is promising that

neutrinos of several 10^{15}eV have recently been detected by the IceCube experiment [32], partly bridging the gap toward UHE neutrinos. While both HE and UHE neutrinos have very high energies, only UHE neutrinos exceed the ~ 10^{13} eV centre-of-mass energy reach of terrestrial particle colliders, when involved in nucleon interactions.

1.3 Astrophysical Motivations

As well as sensitivity to new particle physics, the properties of the UHE neutrinos also make their measurement and study useful for astrophysical discoveries. Measuring their direction would help to identify the sources of UHE cosmic rays and neutrinos, with candidates including flat-spectrum radio quasars, pulsars, gamma-ray bursts, tidal disruption events, blazars, black holes, jets inside stars and dense circumstellar material [1].

Expected neutrino fluxes from some of these astrophysical sources are shown in figure 2.8. PUEO's best sensitivity to astrophysical neutrinos is at around $10^{18}eV - 10^{18.5}eV$, particularly if there are multiple flights that increase the flight time closer to 100 days. Cosmogenic models are also shown, for cosmic rays that are from nearby sources ("local") as well as those coming from distant cosmic sources ("non-local"). TA refers to the Telescope Array [34].For these models, PUEO is likely to either make a detection or place strong constraints, even with a single 30-day flight, with the most favourable sensitivity for energies between $10^{18}eV - 10^{19.5}eV$.

Properties of the detected neutrinos can then probe the acceleration mechanisms of UHE cosmic rays, including shock acceleration, shear acceleration and magnetic reconnection. The measurements include neutrino flavour composition, where even under standard oscillations the flavour composition at Earth depends on the composition at source [35, 36]. High energy neutrinos are expected to be produced by charged pion decay, as shown in equations 1.1 to 1.5. The flux flavour ratio at source is then $\Phi^0_{\nu_e} : \Phi^0_{\nu_\mu} : \Phi^0_{\nu_\tau} = 1 : 2 : 0$, which results in the observed ratio of approximately $\Phi_{\nu_e} : \Phi_{\nu_\mu} : \Phi_{\nu_\tau} = 1 : 1 : 1$. However, in sources with strong magnetic fields, for example, the intermediate muons cool via synchrotron radiation so that the source composition is $\Phi^0_{\nu_e} : \Phi^0_{\nu_\mu} : \Phi^0_{\nu_\tau} = 0 : 1 : 0$, which results in observed ratio at Earth of $\Phi_{\nu_e} : \Phi_{\nu_\mu} : \Phi_{\nu_\tau} = 1 : 1.8 : 1.8$. New physics such as Lorentz invariance violation and non-standard neutrino interactions can further impact the observed neutrino flavour ratio [37].

Chapter 2

The Payload for Ultrahigh Energy Observations (PUEO)

2.1 High Altitude Radio Detection of UHE Neutrinos in Antarctica

Compared to high energy neutrinos which have recently been detected, UHE neutrinos have both shorter interaction length as well as diminished flux. Experiments targeted at their detection must therefore be highly sensitive to neutrinos interacting in the atmosphere, or those interacting in rock, ice, or water near the surface of the Earth.

ANITA and PUEO are long duration balloon-borne detectors that search for in-ice UHE neutrino interactions, which generate compact electromagnetic showers with charge anisotropy. This results in coherent, impulsive Askaryan radiation [38] being emitted, which is the Cherenkov radiation produced by relativistic particle showers within the ice. The emission is peaked in the Cherenkov cone. This radiation is coherent in the radio, or more specifically the microwave (300MHz - 3GHz) part of the electromagnetic spectrum. Shorter wavelengths are incoherent due to the interaction lengths of the shower particles, and lower frequencies contribute less power while also requiring larger antennas [39]. The long attenuation length of radio frequencies within ice (in contrast with the Earth's crust) then allows this signal to propagate over kilometre-long distances.

This Askaryan signal detection strategy is shown on the left of figure 2.1. Due to event geometry, the sensitivity is primarily for 'Earth skimming' neu-

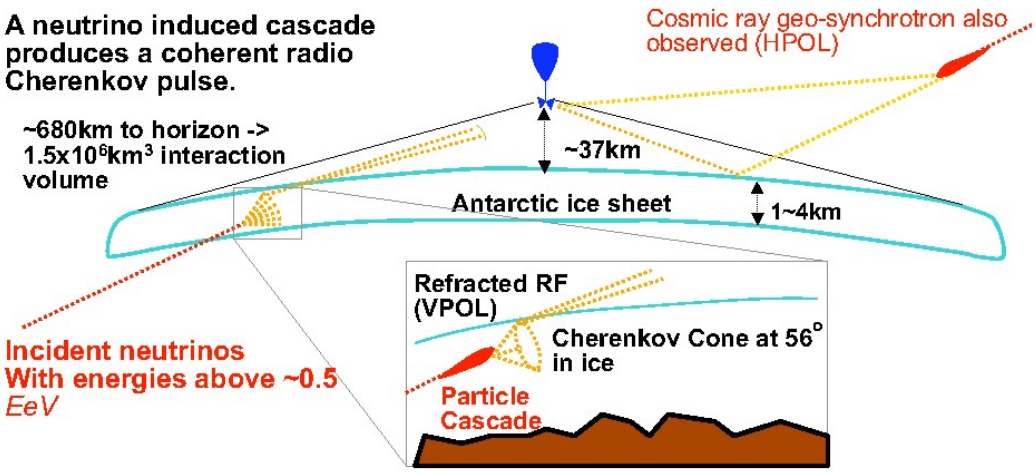


Figure 2.1: The main detection channels for ANITA and PUEO, reproduced from [2]. Earth-skimming neutrinos interact with the Antarctic ice sheet, emitting vertically polarised (VPol) signals, while extensive air showers from cosmic rays produce horizontally polarised (HPol) signals.

trinos at angles close to the horizon. More steeply up-coming neutrinos are expected to be absorbed by interactions with the Earth, and steeply down-going neutrinos emit radiation away from the instrument. The refractive index of ice yields a critical angle of $\sim 56°$, so that only the top portion of the Cherenkov cone will be emitted through the ice, with the rest being reflected (figure 2.2). Since Cherenkov emission is radially polarised relative to the cone, neutrino-induced emission are expected to be predominantly vertically polarised.

ANITA and PUEO are also sensitive to extensive air showers from cosmic rays, which have been measured during the ANITA flights. This is illustrated on the right of figure 2.1. Cosmic rays propagating in the atmosphere interact with air nuclei and initiate extensive air showers containing charged particles. The dominant source of radiation comes from geo-synchrotron radiation as the electrons and positrons are accelerated by the Earth's magnetic field, with the latter approximately vertical near the south pole. Since the synchrotron radiation is emitted perpendicular to the magnetic field, the cosmic-ray-induced emission are expected to be predominantly horizontally polarised, in contrast with the neutrino Askaryan channel.

The direction of the neutrino is determined by measuring the relative

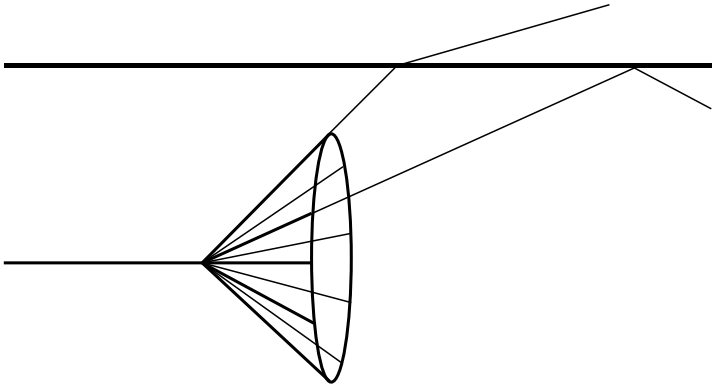


Figure 2.2: Schematic of Askaryan radiation from a neutrino interacting in the ice, reproduced from [2]. Event geometry means that sensitivity is largely for Earth-skimming neutrinos, with only the top portion of the emission cone transmitting through the ice rather than being totally reflected.

timing of the resulting the signal as detected at multiple antennas. This information is then combined with the position of the payload, to yield the location of the neutrino interaction in the ice, allowing the neutrino energy to be estimated when combined with the strength of signal detected at the antennas.

As well as the Askaryan channel for 'Earth skimming' neutrinos interacting in the ice, ANITA and PUEO are also sensitive to air showers produced by UHE ν_τ travelling through the Earth after a process of ν_τ regeneration [40]. Unlike electrons and muons that lose energy rapidly through interactions, tau leptons have a short lifetime and can quickly decay back to tau neutrinos, and subsequently convert back to a lepton through a charged current interaction.

The Antarctic ice sheet is chosen as the interaction medium. This means the detector is operating in an essentially uninhabited area, minimising the sources of man-made backgrounds for the detector. Compared to in-ice detectors such as ARA [41] and ARIANNA [42], a high altitude detector over the ice enables detection from a much greater interaction volume, at a cost to sensitivity at lower energies due to attenuation in ice and air. This provides ANITA and PUEO with the largest effective area at the highest energies among experiments.

This is contrasted with optical detectors, such as IceCube [43]. By instru-

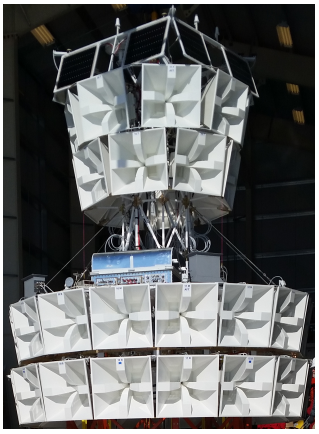


Figure 2.3: A picture of the ANITA-IV payload [3], which was ~8m tall, with each horn antenna roughly 0.95m from edge to edge.

menting the ice directly, such experiments can detect events of lower energies, with greater inference of properties such as angular and energy resolutions, but at the cost of small detector volume. Indeed, IceCube-Gen2 [44] and associated pathfinder RNO-G [45] incorporate radio arrays in their detection strategies.

2.2 ANITA

Over four flights from 2006-2016, the Antarctic Impulsive Transient Antenna (ANITA) [4] set world-leading limits for diffuse UHE neutrino flux at energies between $10^{19.5}$ eV and 10^{21} eV.

Deployed on NASA long-duration balloons, the ANITA payloads consisted of dual-polarisation horn antennas arranged in circular patterns across the azimuthal directions. Figure 2.3 shows the ANITA-IV payload. The most recent flights, ANITA III and ANITA IV, were each equipped with 48 antennas, with band frequencies of 180-1200 MHz. The four ANITA payloads were physically similar, with coincidence based triggers on 32-48 antennas, and improvements introduced over iterations across the hardware, particularly in mitigation of continuous wave contamination. Since each flight was limited to around 1 month, with the mission ending before the payload leaves Antarctica and becomes no longer retrievable, multiple flights allowed the accumulation of detector data collection time.

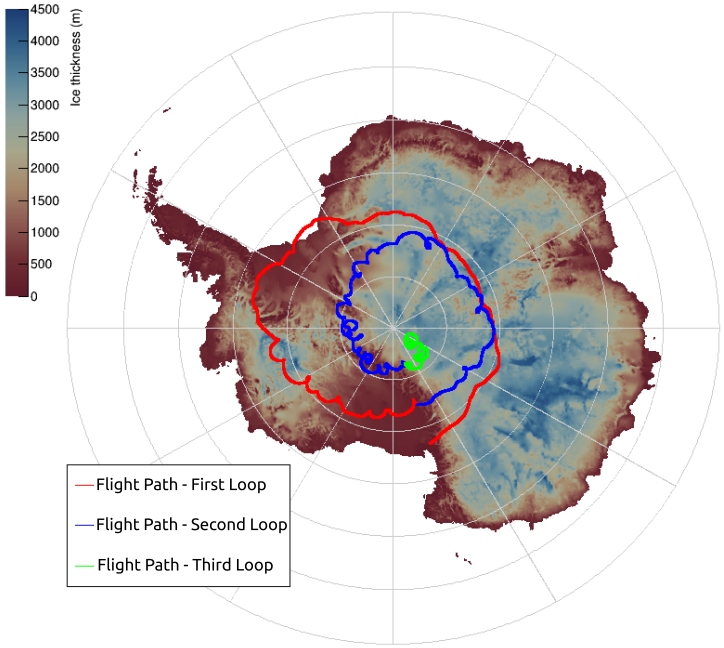
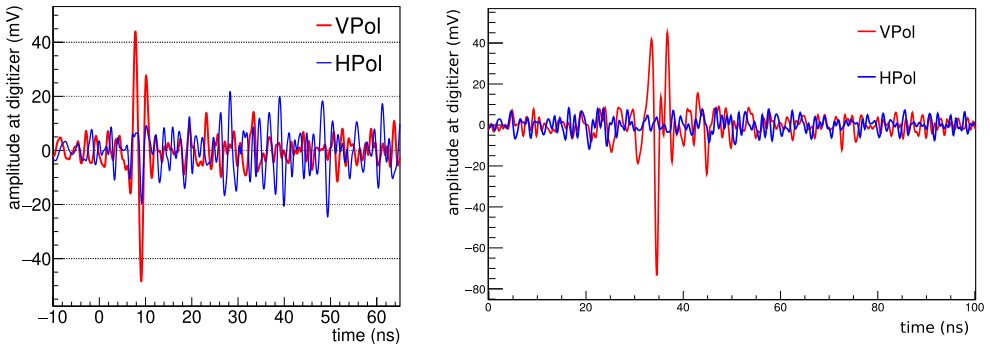


Figure 2.4: The ANITA-IV flight path [3], which lasted 28 days over Antarctica. The flight began at McMurdo Station and made two full loops around the continent before landing near South Pole Station.

Due to limited telemetry, most of the data was written to hard drives on the payload. The required digitisation limited the writing rate, so that multiple-level triggers were used to determine the mostly likely candidates for later analysis. The triggers were based on square-law power detectors on individual antennas, combined with coincidences across antennas. ANITA-IV additionally improved the effective threshold, by combining HPol and VPol channels to form circularly polarised signals. Neutrino or air shower signals are primarily linearly-polarised and therefore have approximately equal left-and-right circular polarisation signals, unlike noise.

Both flights had to manage continuous wave contamination from geostationary communication satellites, with some of the frequencies unknown prior to the flights. ANITA-III dealt with this by masking azimuthal sectors facing the interference, and ANITA-IV deployed tuneable notch filters to filter out frequencies during the flight.

Flying at altitudes of 37 to 40 km, the payloads did not have independent



(a) (b)

Figure 2.5: The ANITA-III (a) and ANITA-IV (b) neutrino candidates. Both events are impulsive and isolated, but consistent with the background expectations for each flight.

propulsion, rotating and flying freely, with location and orientation recorded for analysis. Figure 2.4 shows the flight path from ANITA-IV. On-board solar panels provided the power for the payload electronics. ANITA-III flew for 22 days and ANITA-IV for 29 days, with the flights being terminated at locations where they can still be reached for data retrieval.

Diffuse neutrino searches for ANITA-III and ANITA-IV each found one candidate event. These are shown in figure 2.5. Both events are impulsive and primarily VPol, but are consistent with the background expectation of around one event. Interpreting the candidate events as background, figure 2.6 shows the limit plot for the flux of UHE neutrinos from the ANITA flights, which are the lowest thresholds for energies between $10^{19.5}$ eV and 10^{21} eV.

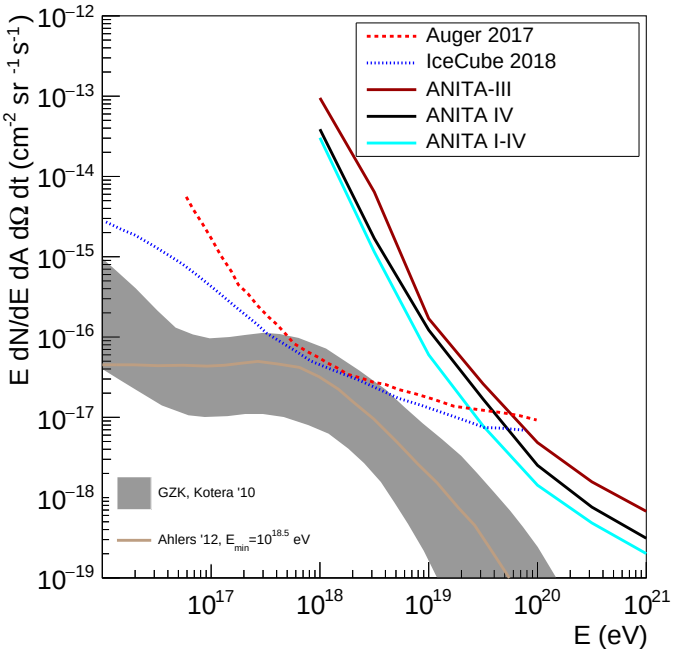


Figure 2.6: The diffuse neutrino flux limit plots [4] for ANITA-III, ANITA-IV and the combination of all ANITA flights. This is compared with some cosmogenic models and limits from Auger and IceCube.

2.3 PUEO

The Payload for Ultrahigh Energy Observations (PUEO) [5] is a direct successor to ANITA. A rendering of the PUEO payload is shown in figure 2.7. Maintaining a similar payload envelope, PUEO has a significantly improved projected sensitivity driven primarily by the use of a coherent beamforming trigger. The development and study of this trigger is the focus of this work.

PUEO's design incorporates 96 dual-polarisation antennas for the primary beamforming instrument, compared to 48 for ANITA-IV. Since the overall size and weight of the payload is limited by NASA long duration specifications, smaller antennas are used that have a higher cutoff frequency of 300 MHz. Instead of using square-law combinatoric triggers, signals from N = 16 antenna sectors are digitally summed in real time with relative delays corresponding to sets of trial direction, referred to as beams. For beams that line up with the source signal direction, the signal power is coherently

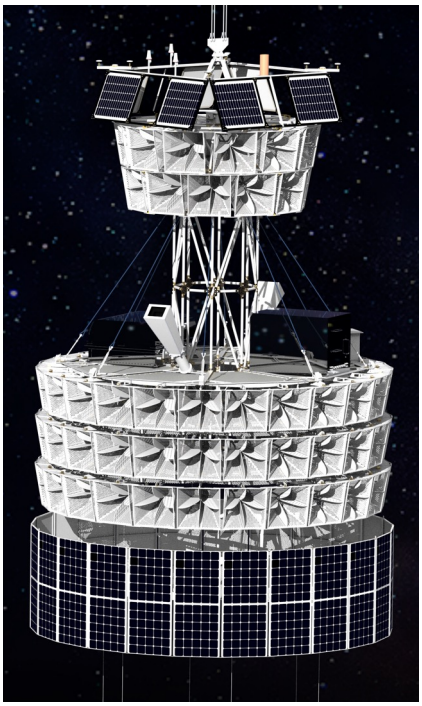


Figure 2.7: An illustration showing the main instrument of the PUEO payload. Below the main payload there will also be low frequency antennas.

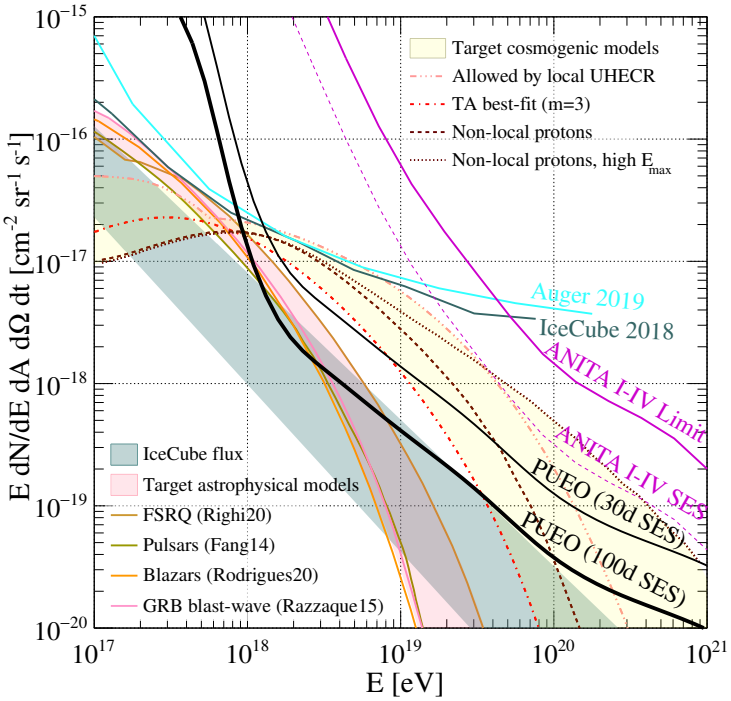


Figure 2.8: PUEO's sensitivity to diffuse UHE neutrino fluxes [5], compared to existing limits and some cosmogenic models and astrophysical models. More details on these models are given in section 1.3.

summed, whereas the thermal background is incoherently summed.

This results in a factor of $\sim\sqrt{16}$ improvement in the effective signal-to-noise ratio, which in turn results in reduced trigger threshold and thus improved sensitivity, particular in the lower energy range. Figure 2.8 shows PUEO's single-event sensitivity to diffuse UHE neutrino flux, compared to existing limits and cosmogenic/astrophysical models of neutrino flux.

Significant computational processing is required for real time beamforming. This is enabled by digitising and processing the antenna signals using Radio-Frequency Systems-on-Chip (RFSoCs). These combine Field Programmable Gate Arrays (FPGAs), Digital Signal Processing (DSP) cores and high-bandwidth GHz digitisers on a single die. Digital signal processing with the onboard FPGAs also allow the deployment of digital tuneable filters, which will be more flexible and effective for the management of continuous wave contamination.

2.4 Radio-Frequency System-on-Chip

The AMD Xilinx RFSoC platform [46] is used as the basis for PUEO's beamforming trigger. This commercially-available hardware platform is designed primarily for radio frequency communication applications, such as wireless infrastructure, phased array radar and satellite communications. A number of properties of RFSoCs make them well suited for a beamforming trigger system:

High Frequency Analogue-to-Digital Converters The RFSoCs support RF input frequency up to at least 4 GHz, which exceeds the ~3 GHz required to capture PUEO's antennas response. As well as achieving this high sampling rate, RFSoCs also provide synchronous sampling across multiple channels, which is important for coherent beamforming. This can be achieved on both the channels on one RFSoC board (usually 8), as well as across multiple RFSoC chips.

Programmable Logic RFSoCs incorporate high capacity FPGAs, programmable logic hardware whose circuitry can be configured at runtime at the gate level. This allows the parallel processing of a large quantity of data, contrasting with CPUs which are largely linear in calculations. The clock-cycle nature of hardware circuits also makes it easier for output from evaluations to be completed within a predictable amount of time. This is beneficial for the real time decisions required of the trigger. While the radio signals are sampled at ~3 GHz, the FPGA can only operate at frequencies in the hundreds of MHz. Signal processing in the programmable logic is usually chosen to be at $\frac{1}{8}$ of the sampling frequency, with several samples processed simultaneously in one clock cycle.

Power Consumption Instead of a multi-component solution that combines FPGAs with external analogue-to-digital (ADC) data converters, RFSoCs integrate them into a single chip. This significantly reduces power consumption, a key constraint for a balloon payload. This is achieved through reduced circuitry, as well as replacing complex FPGA-to-analogue interfaces such as JESD204 with simpler alternatives. While ANITA's trigger rate was ultimately limited by the power required to digitise the signals, PUEO would be able to digitise all signals, removing a key limitation from ANITA.

Digital Signal Processing Cores The Xilinx RFSoCs incorporate a large number of digital signal processing (DSP) slices. DSPs are most commonly used for a multiply-accumulate operation, with a multiplication followed by an addition. This is the basic building block of a digital filter, pro-

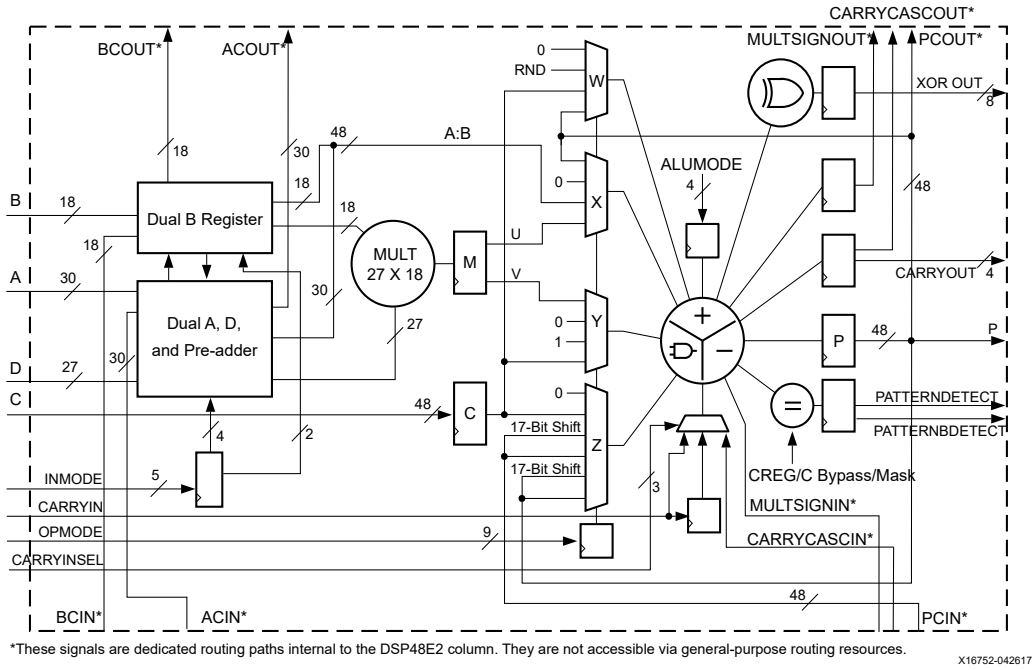


Figure 2.9: A schematic for the DSP48E2 slice of the Xilinx RFSoC [6], showing the number of bits supported at each input. The slice consists of a 27-bit pre-adder, 27 x 18 multiplier, and a flexible 48-bit ALU that serves as a post-adder/subtracter, accumulator, or logic unit.

viding a tap that multiplies a coefficient with an input signal value. Figure 2.9 shows the schematic for the Xilinx DSP slice. The large number of bits supported for the operation, means that several multiplications can be carried out simultaneously in a single clock cycle. The DSP slices are therefore able to enable digital filtering capabilities required for PUEO. Specifically, high-quality notch filters can be tuned to filter out only a narrow band of the received waveform around the specific frequencies of continuous wave emission from satellites, increasing the signal that can be retained for the trigger.

Commercial Availability While each 8-channel RFSoC still costs thousands of dollars, the 24 units required for the 192 channels (96 dual-polarisation antennas) is not a prohibitive expense. This is despite many features from the RFSoC being unused or underutilised, such as the digital-to-analogue converter, on-board CPUs, and communication features such as decoding/encoding

and amplitude modulation. This is made possible by using off-the-shelf products that serve a number of industry applications. The commercial scale of FPGAs and RFSoCs is illustrated by the $35 billion acquisition of Xilinx by AMD in 2022.

In chapter 3 and chapter 4, the design of the PUEO beamforming trigger is outlined and simulated in software, providing results for the expected trigger performance as well as design optimisations. In chapter 5, a proof-of-concept trigger is implemented for a single RFSoC board, outlining key steps in the hardware design process and implementation of the complex system of clocks required for synchronising across channels.

Chapter 3

Improving Neutrino Detection: Beamforming Trigger

One of PUEO's key improvements over ANITA involves using digital coherent beamforming to increase the number of signal events that are detected, by lowering the threshold strength for a signal to be triggered. This enables digitised waveform associated with the signal to be stored, and later analysed. Since only a small fraction of data can be stored, successful trigger is a prerequisite for a signal's detection.

PUEO's main instrument antennas have a bandwidth of 300-1200 MHz, so that all waveform information is retained when this is digitised at a frequency of, or just below, 3 GHz. This digitisation is carried out at 12 bits. While this is reduced to a smaller number of bits for efficient beamforming, as will be discussed, the higher fidelity data is stored for triggered waveforms.

A software simulation was developed and integrated into the PUEOSim [47] simulation framework. This chapter outlines design considerations for such a beamforming trigger. The next chapter provides more details for the simulated trigger, and quantifies design decisions that maximise the sensitivity gained by the coherent beamforming.

3.1 Signal-to-noise Ratio

We define the signal-to-noise ratio (SNR) as half the peak-to-peak amplitude of the time domain signal, divided by the root mean square of the noise in the time domain:

$$\text{SNR} = \frac{\text{Amplitude}_{\text{signal peak-to-peak}}}{2 * \text{RMS}_{\text{noise}}}. \tag{3.1}$$

Due to limits in onboard storage and telemetry (for storage away from the instrument), only a very small proportion of the data can be stored for later analysis and detection. A signal event therefore needs to be triggered to be detected. With a given set of signals, the trigger efficiency is the proportion of signals that are triggered and therefore stored for later analysis. A benchmark for trigger performance, independent of the received neutrino energy spectrum, is the SNR where the trigger is 50% efficient, i.e. where 50% of signals at that SNR successfully trigger, resulting in the digitised waveform being written to storage. This performance metric will be denoted $\text{SNR}_{50\%}$, with lower values indicating better trigger performance.

The sensitivity of the PUEO and ANITA experiments has typically been measured in terms of volumetric acceptance $\langle V\Omega \rangle(E)$ [48]:

$$\langle V\Omega \rangle(E) = \frac{n_{\text{pass}}(E) V_0 \Omega}{N(E)}, \tag{3.2}$$

where $n_{\text{pass}}(E)$ is the number of events that pass the trigger at a given energy E, V_0 is the volume of ice in Antarctica viewed by the payload, Ω is 4π steradians, and $N(E)$ is the number of neutrino events at the given energy E.

The volumetric acceptance therefore depends on the trigger rate at a given SNR, through its dependence on the trigger rate at a given event energy. Other factors also determine the SNR received at the antennas - such as the depth of the event within the ice, the distance between the event and the detector, or the position of the detector relative to the Askaryan cone. The SNR simplifies trigger comparisons by studying acceptance rates of signals as they are received at the antenna, after taking into account these factors.

However, there are limitations in using $\text{SNR}_{50\%}$ as a measure of neutrino detection sensitivity. It does not account for variations in the shape of the SNR curve, since there is no specification of the trigger rate of events at SNRs other than that of the 50% trigger rate. This is demonstrated with two examples SNR curves in figure 3.1. In addition, the SNR here is defined only on the peak-to-peak amplitude of the signal in time, and does not account for its shape, including its extent in time. The trigger performance is then implicitly associated with and dependent on assumptions for the signal pro-

file in time. Nevertheless, volumetric acceptance is expected to be strongly correlated with the SNR at 50% trigger rate.

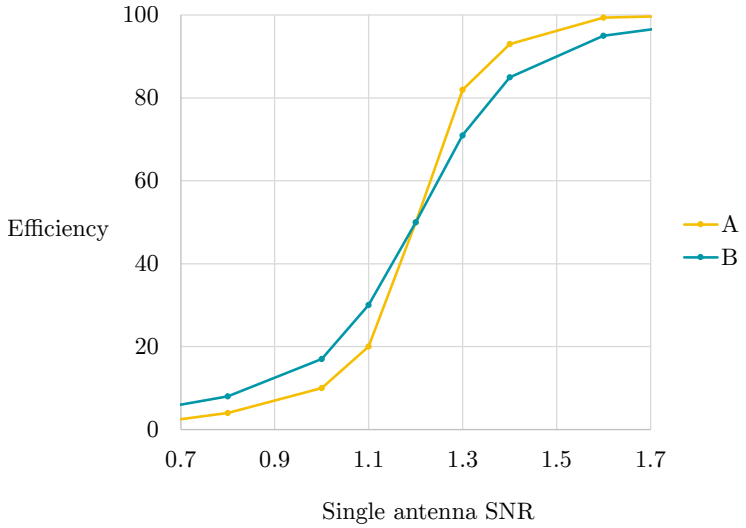


Figure 3.1: Although triggers A and B share the same threshold SNR where 50% of signals are triggered, A performs better at higher SNR and B outperforms at lower SNR. Trigger B may be preferable if we expect the signal flux to rapidly diminish with increasing energy, and thus yield a preference for improving the detection likelihood of a signal with lower SNR. This difference is neglected in the analyses in this and the next chapter.

Note that unless otherwise specified, SNR will refer to single antenna SNR, the relative strength of the signal to the noise received at one antenna. This is distinct from an effective SNR, which includes a multiplier from coherent summation across antennas.

3.2 Simulating Neutrino Signal

To develop, test and optimise the coherent trigger simulation, functions were developed for directly generating the combined signal and noise amplitudes received at each antenna, for signals from a given direction and SNR. This

enables module testing independently of the settings and potential issues in the broader PUEO simulation, while also reducing computation time.

A 1D parametrisation [49] is used to find the peak Askaryan signal, viewed from the Cherenkov angle, at 1m from the point of interaction,

$$\mathcal{E}^{1\mathrm{m}}_{\theta_{\mathrm{Ch}}} = 2.53 \times 10^{-7} \cdot \frac{E}{\mathrm{TeV}} \cdot \frac{\nu}{\nu_0} \cdot \frac{1}{1 + (\nu/\nu_0)^{1.44}}, \tag{3.3}$$

where ν is the frequency, $\nu_0 = 1.15\,\mathrm{GHz}$ and E is the shower energy which will be absorbed in the normalisation when choosing the SNR.

When viewed at an angle θ_{view} compared to the Cherenkov angle θ_{Ch}, the field strength becomes [50]:

$$\mathcal{E}^{1\mathrm{m}}(\theta_{\mathrm{view}}) = \mathcal{E}^{1\mathrm{m}}_{\theta_{\mathrm{Ch}}} \cdot \frac{\sin\theta_{\mathrm{view}}}{\sin\theta_{\mathrm{Ch}}} \cdot \exp\left[-\left(\frac{\theta_{\mathrm{view}} - \theta_{\mathrm{Ch}}}{\Delta\theta}\right)^2\right], \tag{3.4}$$

where $\Delta\theta$ is the width of the Cherenkov cone. Although it is often parametrised separately for electromagnetic and hadronic components of the shower, for simplicity it is treated here with a single width [49]:

$$\Delta\theta = 2.2^\circ \cdot \frac{1\mathrm{GHz}}{\nu} \tag{3.5}$$

To find the incident electric field at the PUEO payload, this is multiplied by a factor of i for a 90° phase, and assumed to have been attenuated in 500m of ice, using the following parametrisation based on the all ice attenuation measured in [51] with a bedrock field reflectivity R=0.3:

$$\alpha = \exp\left[\frac{-10}{9.4 - 0.003 \cdot \frac{\nu}{\mathrm{GHz}}}\right], \tag{3.6}$$

where the overall normalisation is again absorbed later by choice of SNR. The 500m attenuation is chosen for simplicity. Once integrated as part of the full PUEO simulation, the traversal of individual neutrinos through the ice would be simulated with a model of the Antarctic ice sheet.

To yield the signal received, the impulse response is multiplied by the electric field given by the product of equations 3.4 and 3.6. This is dependent on $\theta_{\mathrm{view}} - \theta_{\mathrm{Ch}}$, the viewing angle of the antenna away from the Askaryan cone. Fig. 3.2 shows this for a variety of different angles, with a broadening of the signal width at larger viewing angles, but where the signal amplitude is also

significantly diminished. For definiteness, the simulations in this chapter were carried out at $\theta_{\text{view}} - \theta_{\text{Ch}} = 1°$.

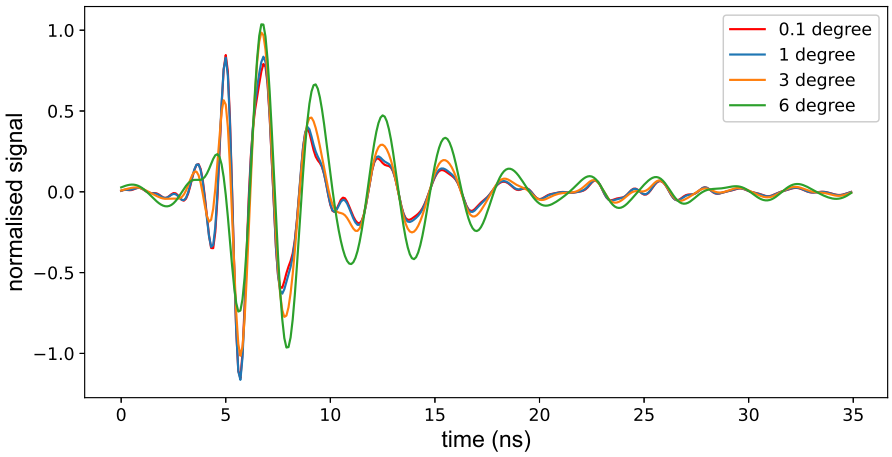


Figure 3.2: Simulated signal received by a PUEO antenna, at a variety of angles away from Askaryan cone. while each signal is plotted with peak-to-peak normalisation, amplitude of the the same signal is significantly diminished when viewed away from the cone. For definiteness, simulations in this chapter have signal with $\theta_{\text{view}} - \theta_{\text{Ch}} = 1°$.

The signal is received at different antennas at times which are offset due to their relative position to the source. This relative differences in distance travelled can be accounted for using the following expression:

$$\Delta L_i = \cos(\theta)[z_i \tan(\theta) - r_i \cos(\phi - \varphi_i)], \tag{3.7}$$

where z_i, r_i and φ_i are the cylindrical coordinates of the antenna i, and θ and ϕ correspond to the source angle of the signal. While there is an additional additive constant, one of the antennas j can be chosen as the reference, so that $\Delta L_i - \Delta L_j$ yields the difference in distance a signal has to travel before arriving at antenna i.

All trigger processing is based on digitised data, assumed to be at a sampling rate of 2.94912 GHz, chosen as a multiple of the high-fidelity MHz clock used to synchronise the multiple converters in the RFSoC hardware that will power the digitisation. The signal impulse is based on the ANITA

3 impulse response, with a filter used to limit the response to above 300Mhz, a result of the smaller antenna size used in PUEO. This combined with the incident electric field from eqs. (3.4) and (3.6) by complex multiplication at each frequency bin in frequency domain. This is then interpolated, using the difference in relative arrival of the signal derived from eq. (3.7). The role of beamforming will be to reverse these differences, although that will only be in whole units of the sampling rate.

3.3 Simulating Thermal Noise

The incident signal is now combined with simulated thermal noise, resulting from radiation emitted by the Antarctic ice received at different antennas. There are also other sources of noise, including anthropogenic noise such as satellite transmitted continuous wave signals. These are received across multiple antennas, and while the same benefit cannot be achieved through coherent beamforming, there is still some limited improvement from pointing beams away from these sources. Additional efforts to reduce their impact on trigger effectiveness through the use of tuneable narrow band notch filters are not studied in this work, but are also enabled by the digital processing capabilities of the RFSoC.

The electric field due to thermal noise, polarised by the antenna, consists of two uncorrelated components (real and imaginary) each with a Gaussian distribution of the same variance and zero mean. The combined phasor has a uniformly random phase distribution, with the amplitude represented by a Rayleigh distribution [52], with the probability distribution function:

$$p(x) = \frac{x}{\sigma^2} e^{-x^2/2\sigma^2}, \tag{3.8}$$

where σ is the scale parameter, with a mean of $\sigma\sqrt{\pi/2}$. Note that the amplitude therefore does not follow a Gaussian distribution.

Thermal noise for testing the coherent trigger was generated in frequency domain, with a phasor generated at each frequency bin. Using a C++ pseudorandom number generator engine, a random phase for the phasor was drawn from the interval $[0, 2\pi)$. The phasor's amplitude was generated, using the inverse transform sampling method, with

$$A = \sigma\sqrt{-2\ \mathrm{lnU}}, \tag{3.9}$$

where U is pseudorandomly drawn from the interval (0,1). These are independently generated at each frequency bin, for every antenna. The scale parameter σ across frequencies depends on the electronics response. For simplicity, this is approximated as uniform between 240Mhz and 1.3GHz and zero outside. Given a particular SNR being simulated, there is also an overall factor in the scale parameter determined by the signal peak-to-peak and the definition for SNR in eq. (3.1). This requires the noise to be first Fourier transformed to the time domain. Once the appropriate scale has been chosen, the noise and signal at each antenna are then combined through (complex) addition at each frequency bin in the frequency domain.

3.4 Coherent Summation

In this and the next section, the main features of coherent summation are illustrated using simplified signal waveforms and neglecting antenna separation. Let us consider inputs received at two different antennas, with waveforms of amplitude $a(t)$ and $b(t)$, both with average intensity i:

$$i = \frac{\kappa}{T} \lim_{T\to\infty} \int_0^T a(t)^2\, dt = \frac{\kappa}{T} \lim_{T\to\infty} \int_0^T b(t)^2\, dt, \tag{3.10}$$

where κ is a constant.

The combined waveform then has the intensity:

$$I_2 = \frac{\kappa}{T} \lim_{T\to\infty} \int_0^T \left[a(t) + b(t)\right]^2 dt \tag{3.11}$$

$$= \frac{\kappa}{T} \lim_{T\to\infty} \int_0^T \left[a(t)^2 + b(t)^2 + 2\, a(t)\, b(t)\right] dt \tag{3.12}$$

$$= \begin{cases} 2p, & \text{if } a(t) \text{ and } b(t) \text{ uncorrelated} \\ 4p, & \text{if } a(t) \text{ and } b(t) \text{ perfectly correlated,} \end{cases} \tag{3.13}$$

where we used the relationship $\frac{1}{T} \lim_{T\to\infty} \int_0^T a(t)\, b(t)\, dt = 0$ when $a(t)$ and $b(t)$ are uncorrelated.

Generalising to N antennas, with waveforms of amplitude a_i,

$$I_N = \frac{\kappa}{T} \lim_{T\to\infty} \int_0^T \left[\sum_{i=1}^{N} a_i(t)\right]^2 dt \tag{3.14}$$

$$= \begin{cases} Ni, & \text{if } a_i(t) \text{ uncorrelated} \\ N^2 i, & \text{if } a_i(t) \text{ perfectly correlated.} \end{cases} \tag{3.15}$$

Consider now waveforms that are each a sum of uncorrelated thermal noise and perfect correlated signal. Due to linearity, the intensities of these two parts can be considered separately using eq. 3.15. The relative intensity of the signal compared to the noise in the combined waveform is therefore multiplied by a factor of $N = N^2\, i/(N\, i)$ as a result of coherent summation.

The enhancement to the root mean square is the square root of the enhancement for intensity. Using the definition of SNR given by eq. (3.1), the signal-to-noise ratio, which is a ratio of amplitudes, is scaled by a factor of $\sqrt{N}$. This is the improvement in amplitude for signal detection sensitivity enabled by using a coherent sum of waveforms from different antennas.

Some modifications to these expressions are made when we consider the coherent summation used in the trigger of the PUEO detector. The impulsive signals we are interested in are localised in time, and the summation is over a finite time interval T without taking the limit of large T. T is referred as the window size. This also means the coherent sum needs to be continuously evaluated at different times, which we can denote by the starting time, t_0. The geometry of the antennas located on a detector means signal from a given direction arrives at different antennas at different times, requiring a set of offsets o_i^b to enable the sum to be coherent, with b referring to a specific beam associated with a particular source direction. Finally, the signal is not continuous but instead a series of discretised samples, replacing the integral with a sum. We therefore define the coherent sum of amplitudes $a_i(t)$ of N antennas as:

$$P_N(t_0) = \sum_{t=t_0}^{t_0+T} \left[\sum_{i=1}^{N} a_i(t - o_i^b)\right]^2. \tag{3.16}$$

The relationships in equation 3.15 remain largely true, under these changes. This is demonstrated in figures 3.3 and 3.4. In figure 3.3, uncorrelated noise across 4 waveforms was generated following a Rayleigh distribution. The

RMS of the combined noise waveform is approximately a factor $\sqrt{4} = 2$ larger than each input waveform.

On the other hand, in figure 3.4, the RMS of 4 identical waveforms, when coherent summed, is approximately a factor of 4 larger than each input waveform. The peak-to-peak measurement of the amplitude had a similar enhancement.

In combination, consider four waveforms where each is the sum of noise (from the corresponding waveform in figure 3.3) and signal (from figure 3.4). Using the definition of signal-to-noise ratio in equation 3.1, we find the coherent summation of 4 waveforms enabled a factor of $2 = \sqrt{4}$ improvement in SNR. This is illustrated in figure 3.5.

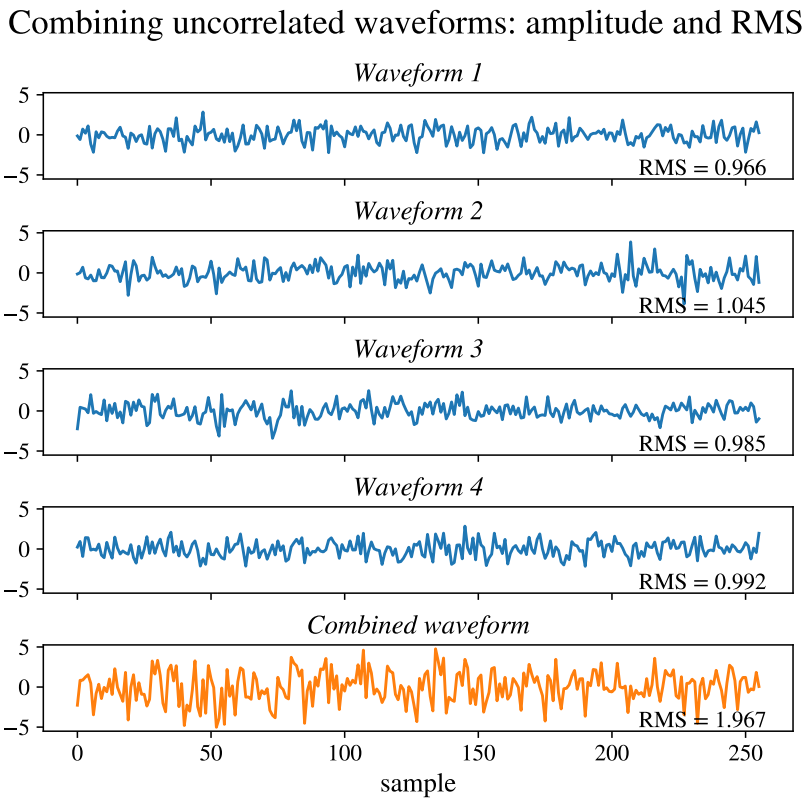


Figure 3.3: The amplitude and RMS of N uncorrelated waveforms are increased by a factor of $\sqrt{N}$ when summed.

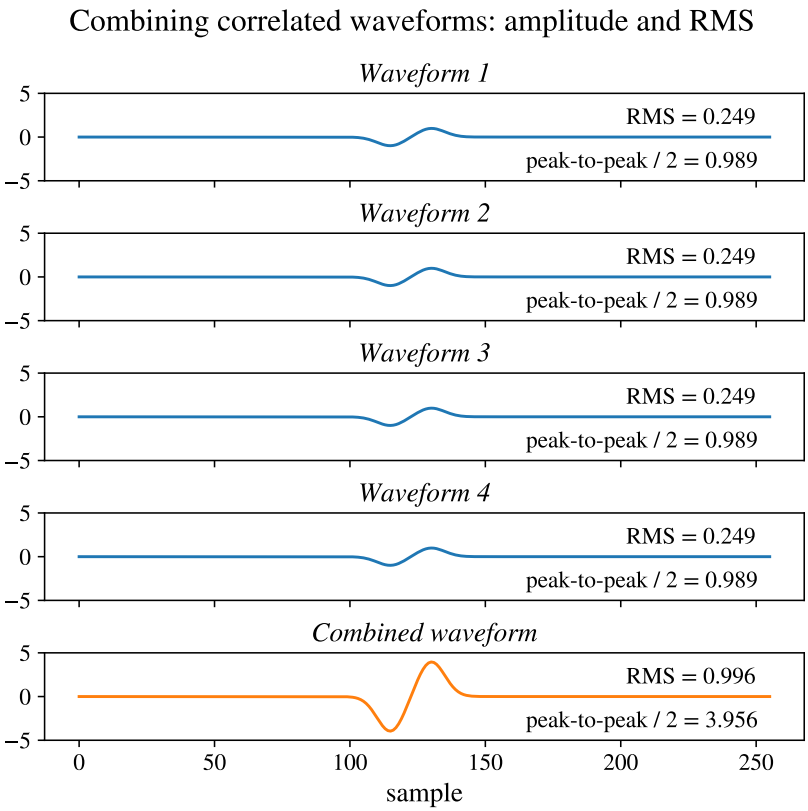


Figure 3.4: The amplitude and RMS of N identically correlated waveforms are increased by a factor of N when summed.

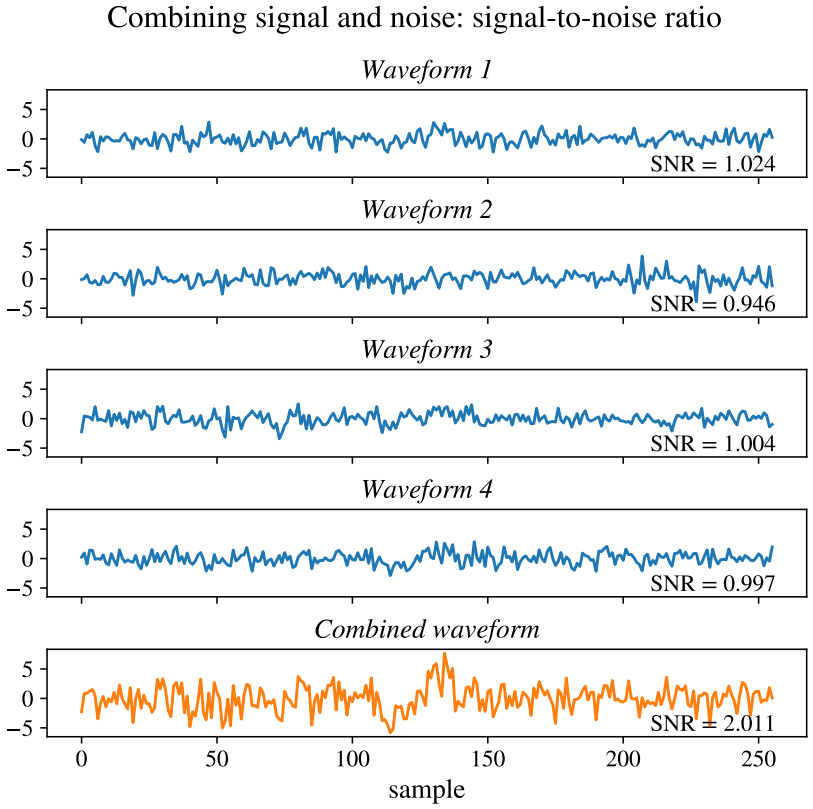


Figure 3.5: Coherent summation of N waveforms yields a $\sqrt{N}$ increase in signal-to-noise ratio.

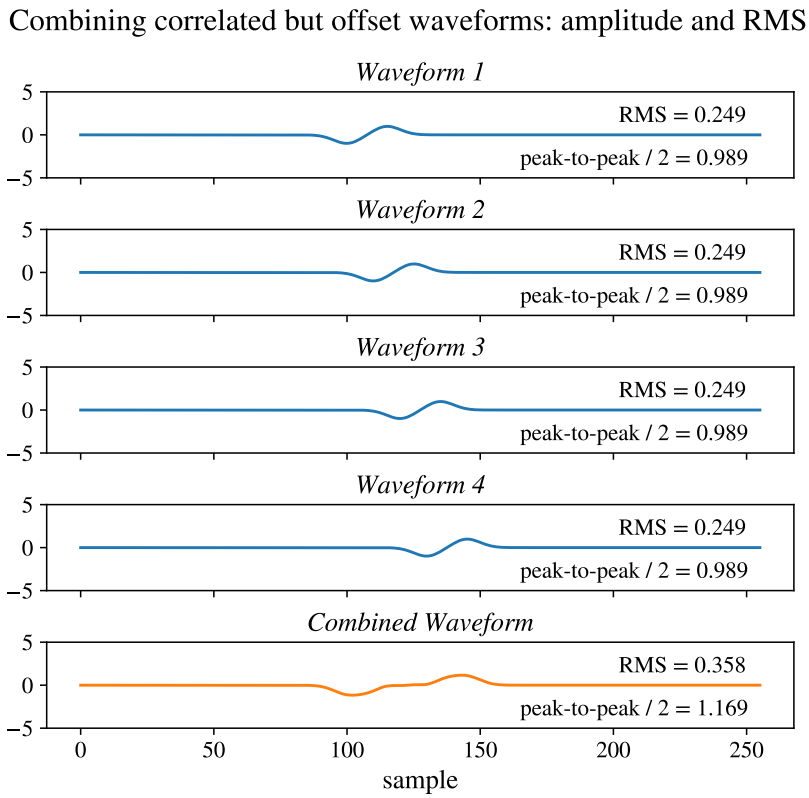


Figure 3.6: Summation of waveforms containing the same signal, but offset in time, do not yield the expected benefit in RMS or peak-to-peak amplitude of a coherent summation.

3.4.1 Waveform Alignment

The above enhancement in SNR from coherent summation assumes the signal parts of different waveforms to be identically correlated, with the same part of identical waveforms aligned across antennas for the sum. If the alignment of the waveforms deviate from this, the relative gain in SNR is diminished, as can be seen in figure 3.6.

Misalignment of two otherwise correlated waveforms causes the contribution from the cross term in equation (3.13) to reduce, not only to zero, as in two uncorrelated signals, but to a potentially negative value. Destructive interference can result in combined SNR that is less than that of the unsummed single antenna. This is accentuated by the expected signal response

having more fluctuations than the toy signal used here, making destructive interference more likely. It is therefore crucial to understand the relative alignments of signals across antennas, which is dependent on the geometry of the antennas on the payload as well as the direction of the source signal.

3.5 Beams and Pointing

In general, different antennas will lie at different distances from a given source, due to their positions at different parts of the payload. Their relative differences can be determined using equation (3.7). In the context of waveform alignment, this results in offsets such that the peaks and troughs of signal waveforms are received at different times.

To achieve the gain in effective SNR, a series of offsets is then required to realign signals across the antennas, based on the geometry of the detector layout relative to that direction. This associates a given series of offsets with

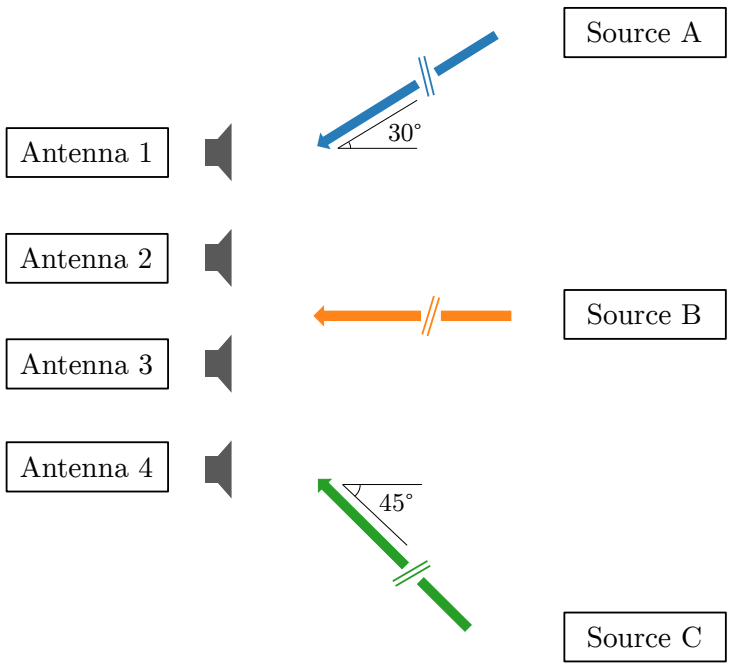


Figure 3.7: A illustrative 2D example, with physically separated antennas each receiving signals from one of 3 source directions.

a directionality, effectively "pointing" the detector and giving rise to the term "beamforming".

Figure 3.7 demonstrates this in a simplified 2D example, with 4 antennas placed in a row with equal separations. Three sources are considered, each at a distance away from the antennas that is much greater than the separation between the antennas. Source B is perpendicular to the line of the antennas, such that the signals are already coherently aligned across the antennas (in the large distance approximation). Source A is at an angle 30° to source B, so that antenna 1 is the earliest to receive the signal. Similarly, source C is at an angle −45° to source B, so that antenna 4 is the earliest to receive the signal.

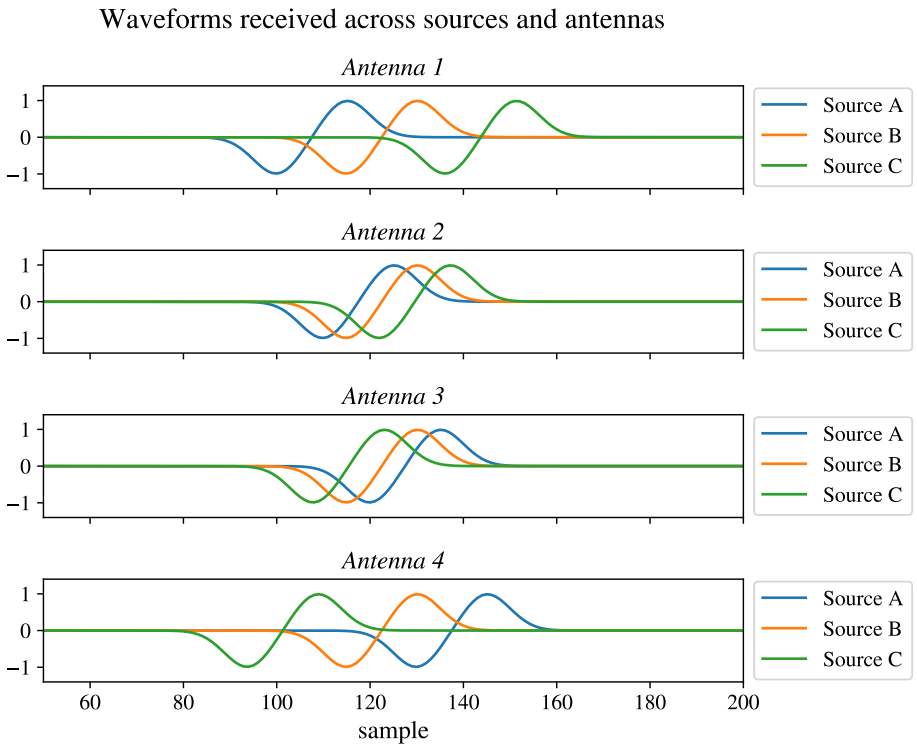


Figure 3.8: For each source direction, antennas on a payload receive them at different times due to the geometry of the antennas on the payload. Signal from source B reaches all antennas at the same time, in the large distance limit. Source A reaches antenna 1 first, whereas source C is first picked up by antenna 4.

Figure 3.8 illustrates these relative offsets in the waveforms received by different antennas. It can be seen that, in general, direct summation of the waveforms from multiple antennas would lead to a combination that is not quite coherent, leading to limited benefit for SNR similar to figure 3.6, rather than the $\sqrt{N}$ benefit of the coherent sum in figure 3.4. For each source direction, this can be rectified by defining a beam, consisting of the number of samples the waveform from each antenna must be offset before the coherent sum, so that a signal from that source direction is aligned across the antennas.

Equipped with the offsets from these beam definitions, figure 3.9 shows the combined waveforms formed using them. Along the diagonal, where the beams match the source directions, benefits of coherent summation are realised. Elsewhere, mismatched beams and source directions yield combined waveforms that have amplitudes not significantly improved on the initial waveforms. This is even more so when considering the impact on SNR, where noise amplitudes also increase with summation as $\sqrt{N}$.

This illustrates a key consideration for computation in coherent beamforming. Since we are in general interested in signals coming from a broad view of different directions, coherent sums using different offsets must be computed simultaneously and continuously, to direct beams across all directions of interest. Even when a signal is present, the majority of beams being evaluated are not coherent - but need to be evaluated in case they are. This presents significant computational requirements.

For a general signal, there is an infinite number of possible sets of offsets corresponding to every possible direction. However, relevant scales help to limit the number of beams required. The signal is digitised at ~2.9 GHz, so that alignments are defined by offsets that are rounded to the nearest integer numbers of samples. The offsets of two nearby directions are then only distinct, if they are separated enough for the value associated with at least one antenna to be rounded to a different integer. Each set of offsets can thus be associated with not just an infinitesimal direction, but a beam of some width, represented by a field of view subtended by some steradians of solid angle. The size of each beam can be characterised by widths along two orthogonal directions. The width along a direction is inversely proportional to the largest separation of antennas along the same axis.

Note that we can in theory align the beams better than this, by interpolating the digitised signal. In practice, this would be too computationally expensive to continuously evaluate across a large number of beams. Fur-

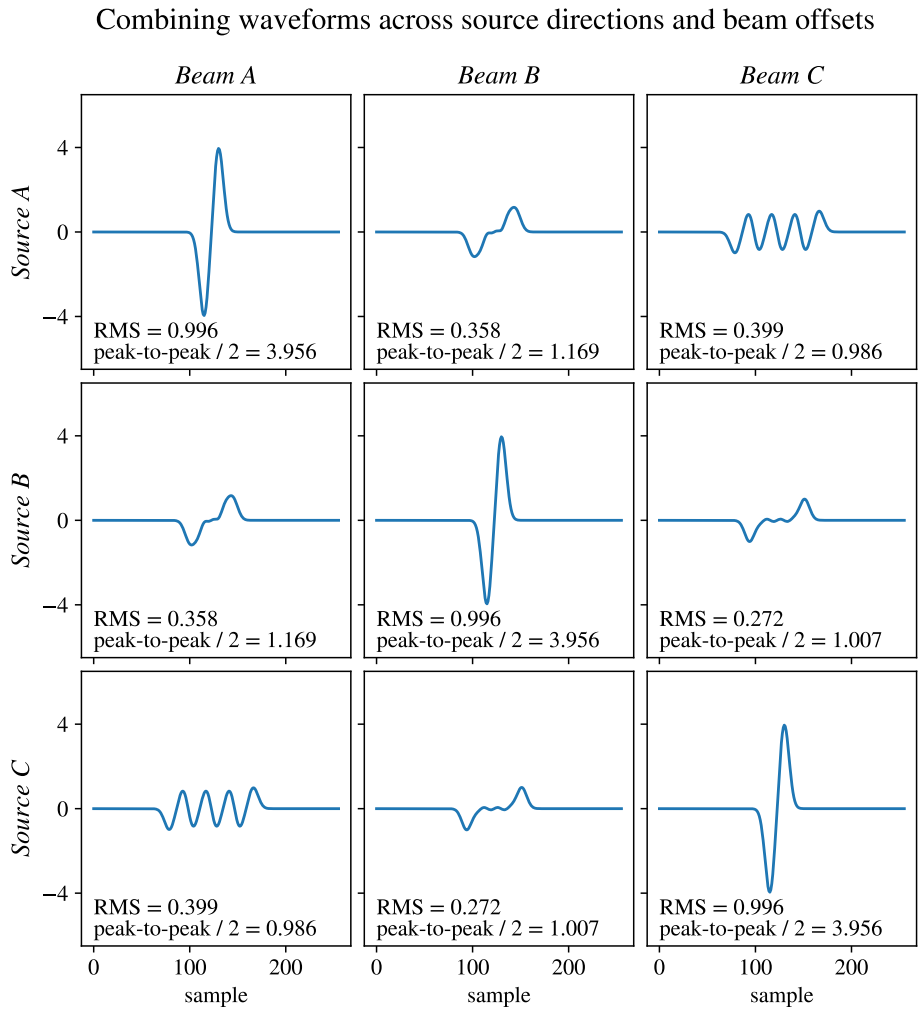


Figure 3.9: Beam definitions consist of offsets for adding waveforms from antennas, pointing the beam at a direction. Signal from the vicinity of that direction achieve the benefits of coherent summation in terms of amplitude, measured here as RMS and peak-to-peak amplitude.

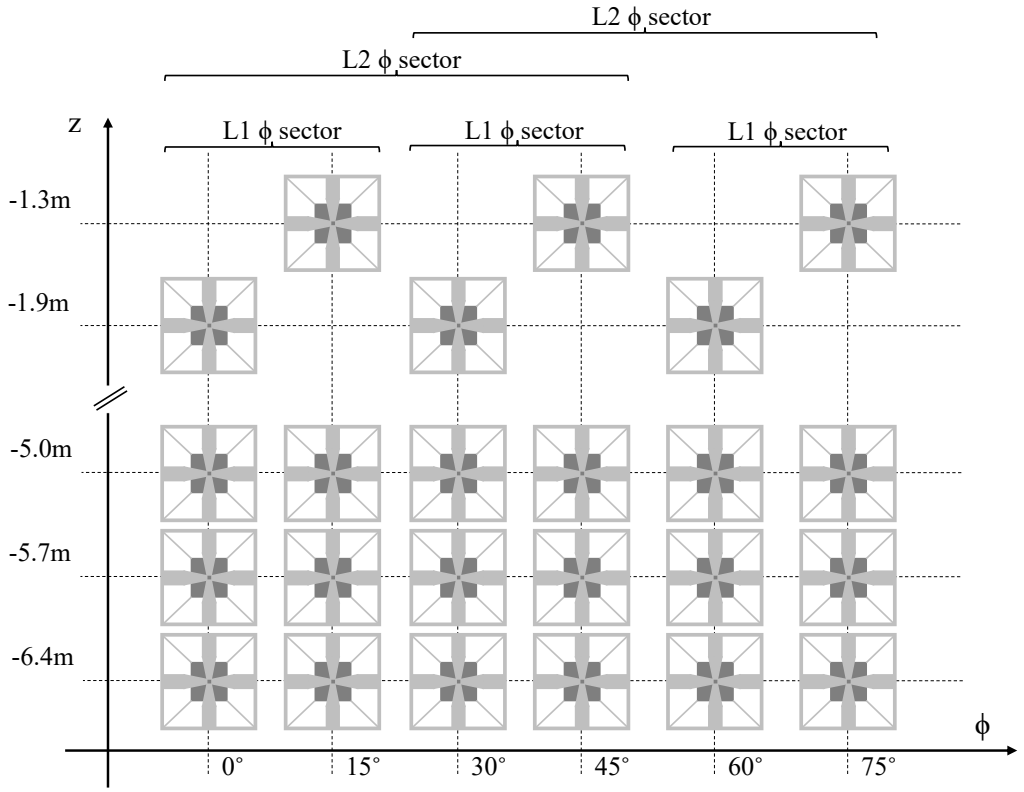


Figure 3.10: Schematic of the antennas and coherent summation ϕ sectors in a 75° arc of the azimuth, in cylindrical coordinates. At each ϕ, the top antenna is at $r \sim 1$m, with the other 3 at $r \sim 2.2$m.

thermore, for two sinusoidal waveforms with frequency ω, a coherent sum between them maintains approximately the same value if they are offset by time δt such that $\omega \delta t \ll 2\pi$.

In the next chapter, simulations will also study whether the resolution can be further reduced to several samples without loss of sensitivity, allowing the consolidation of closely located beams into one single coherent sum.

3.6 Antenna Sectors

The effective SNR of the coherent sum increases by a factor of $\sqrt{N}$, where N is the number of antennas in the sum. Maximising the number of antennas

included in the coherent sum therefore provides for the greatest sensitivity. However, the total number of antennas is limited by the weight and power that can be supported by the payload. The number of antennas used for each sum is also limited by the field of view of the antennas, as well as the real-time inter-antenna data communication and computations required.

Using smaller antennas enables 96 primary beamforming horn antennas to be carried by PUEO. This compares with the 48 antennas on ANITA IV. The antennas are arranged in 4 rings of 24 antennas, each ring at a different height along the main axis of the payload. Adjacent antennas in each ring are separated by an azimuthal angle of $\sim 360^\circ/24 \sim 15^\circ$. Figure 3.10 shows 24 of these antennas, in a 75° arc of the azimuth angle.

Each antenna has a limited field of view of $30^\circ \sim 40^\circ$, either side of the antennas centre along each of the directions perpendicular to its centre. There is significant distortion of the waveform as this limit is reached. However, at least along the azimuthal direction, a signal from a direction close to this limit would also be visible, more on-axis, from a different ϕ sector.

The coherent sum that combines $N = 16$ antennas, forms a level 2 or L2 ϕ sector. However, it is not possible to evaluate the L2 coherent sum continuously, due to the limited bandwidth available to bring together data from all 16 antennas, since they are digitised on different RFSoC boards. Instead, level 1 or L1 ϕ sectors, combining 8 adjacent antennas that are digitised on the same RFSoC board, are evaluated continuously. Only when a L1 ϕ sector is evaluated to be above some threshold, suggesting the possibility of a signal from the vicinity of that beam's direction, the two L2 ϕ sectors corresponding to it are then evaluated.

3.7 Windows and Steps

Once a beam has been formed from the summation of a number of antennas, a combined waveform is yielded similar to the last plot in figure 3.5. Recall that the goal of the coherent trigger is to determine, in real time, the most likely signal candidates. It is expected the presence of a signal increases the magnitude of the waveform's amplitude above the typical variation of the thermal noise.

One can therefore compare the magnitude of the waveform amplitude at each point in time to a threshold. The value of the threshold is determined by analysing randomly generated noise, given the maximum rate that signal

candidates can be triggered and saved to storage for later analysis. The sensitivity of this approach would be limited by large values in the thermal noise. Although this has now been improved by a factor of $\sqrt{N}$ when compared to triggering on a single antenna, individual values can be much larger than the RMS of the noise, limiting the sensitivity to signals with very large SNR.

To further improve digital coherent beamforming efficiency, we should note that even if an Askaryan signal is impulsive, the signal waveforms are not purely impulsive, i.e. they have finite extent in time and can be observed over several samples. Figure 3.11 shows a modeled impulse response of the PUEO antenna and signal chain, based on an ANITA impulse response with a high pass filter at ~ 300 Mhz to reflect PUEO's smaller antenna size. A lower SNR signal can be detected, if the sum is formed by combining the waveform not only over antennas but also across time. This time is the window size T as denoted in the definition of the coherent sum in equation 3.16.

By summing over a period of time similar to the signal's duration, the signal contribution is expected to remain roughly the same order of magnitude,

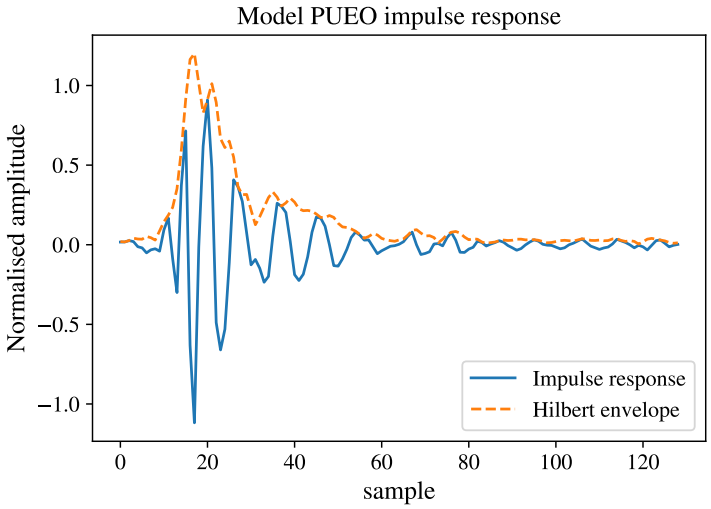


Figure 3.11: Modeled impulse response for the PUEO antenna and signal chain, based on the ANITA 3 impulse response with a high pass filter at ~300 Mhz to account for the smaller antenna size and thus a higher low frequency cutoff. Also shown is the Hilbert envelope, which enables a quantitative measure of the signal width.

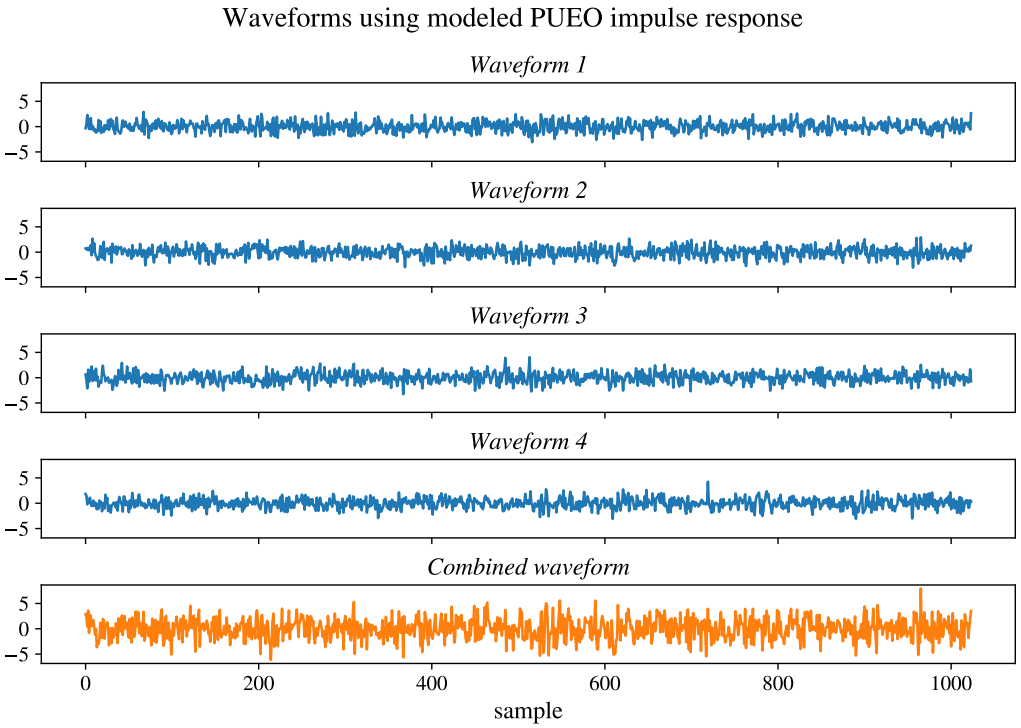


Figure 3.12: Coherently aligned waveforms from four antennas. Compared to the previous toy signal, the impulse response is more quickly varying and longer in duration. A longer duration noise waveform was therefore used, and combined with a signal of SNR = 1.4, with the peak of the impulse response at ~ sample 550.

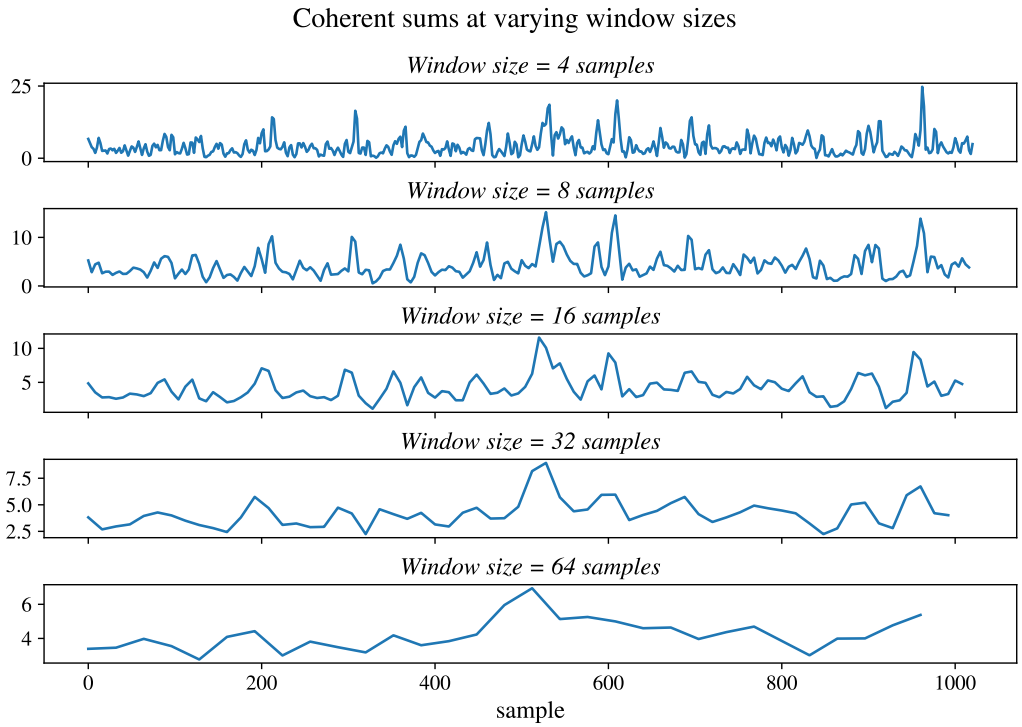


Figure 3.13: Coherently sums computed at different window sizes, from the waveforms in figure 3.12. For each plot evaluated at window size T, the first sum is evaluated using the T samples starting from $t_0 = 0$. The next coherent sums are then evaluated from $t_0 = T/2, T, ...$ i.e. with steps of $T/2$ between adjacent windows.

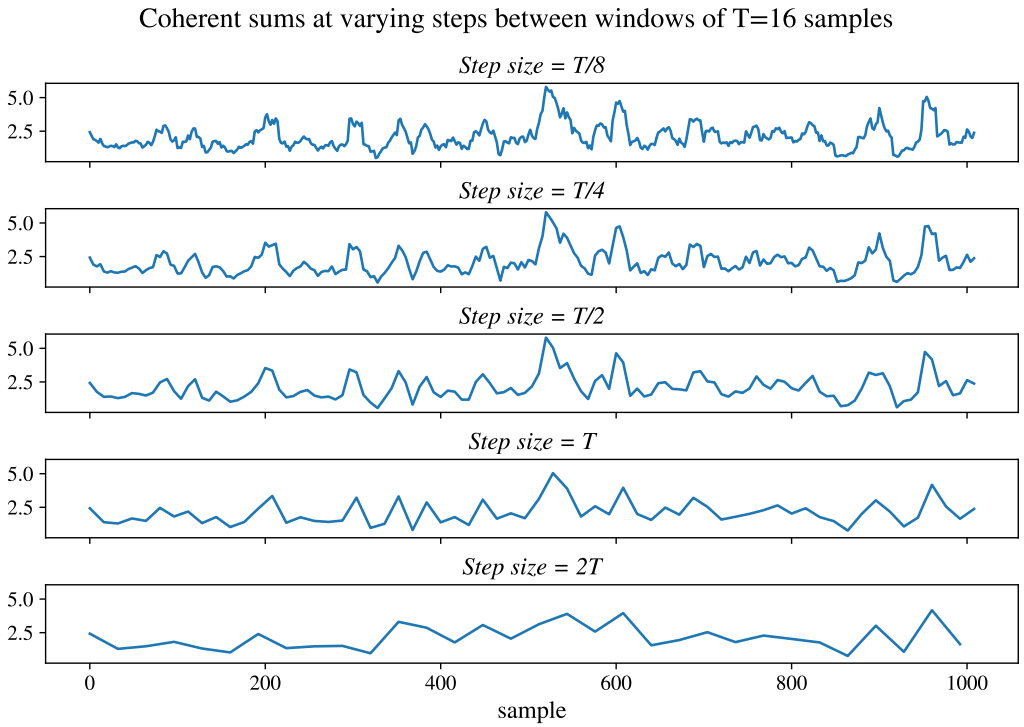


Figure 3.14: Adjacent coherent sums start at samples separated by different step sizes.

noting that the RMS of a sinusoidal waveform is $1/\sqrt{2}$ the peak amplitude. Large spikes in thermal noise, on the other hand, are expected to be averaged down.

Let us study the effect of varying the window size. Replacing the previous toy signal waveform with the more realistic impulse response in figure 3.11, the total number of samples was also extended to account for the longer duration signal. This is illustrated in the waveforms in figure 3.12, which are already coherently aligned, each waveform with signal-to-noise ratio (as defined in equation 3.1) of 1.4.

Due to do the highly oscillating behaviour of the impulse response, even in the combined waveform the presence of the signal is not visually obvious. In fact, the largest single peak is at ~ sample 950, which is noise in origin. Combining coherent sums across different window sizes is shown in figure 3.13. With a window size of 4 samples, there is a peak associated with the

signal, but it is a smaller value than other peaks of thermal noise origin. As the window size is increased, the signal peak grows in prominence compared to peaks from the noise, until it becomes the dominant feature at a window size of 16 samples.

The appropriate window size for optimising the triggering of signal signatures can be related approximately to the duration of the signal. To quantify this, refer to the dotted orange line in figure 3.11. This is the Hilbert envelope (or instantaneous amplitude) of the impulse response, given by its analytic signal, which is a sum of the signal and its Hilbert transform [53]. If the width of the signal is defined to be delimited by the locations where the envelope falls by a factor of 2 compared to the peak, then it is measured to be 12 samples.

If the window size is further increased to be much greater than the duration of the signal, the sensitivity of the coherent sum trigger is expected to drop. The window sizes of 8 and 16 are natural units of calculation, when evaluated in the FPGAs.

In the above evaluations, adjacent windows of size T were evaluated from starting samples t_0 separated by steps of size $T/2$. Figure 3.14 illustrates the effect of varying the size of the step for T = 16. Smaller steps increase the chance that the coherent sum is lined up with the signal, at the cost of greater computation and increased threshold due to a larger rate of potential triggers. The next chapter will study the quantitative effects on trigger efficiency of varying the window and step sizes.

3.8 Trigger Hierarchy and Threshold

The beamforming sum serves to increase the relative contribution of signal to noise, in a metric where they cannot be separated. The presence of a signal increases that sum, especially when it is at the most coherent - when the beam is pointing at a direction close to the source. This makes it more likely to exceed a threshold value and lead to a successful trigger.

To evaluate the effect on trigger efficiency, we must also determine the value of this threshold. With a given coherent summation algorithm, the value of the trigger threshold is determined by the noise profile and the target trigger rate. The target trigger rate reflects constraints on the rate at which the system is able to process and store data for later analysis.

Figure 3.15 shows the hierarchy of the trigger structure. The 96 main

instrument antennas are subdivided into 12 groupings of 8 adjacent antennas, yielding 24 L1 beamforming sectors due to two polarisations. There are also 24 (overlapping) L2 beamforming sectors, consisting of all combinations of pairs of adjacent L1 sectors, as shown in 3.10. The total PUEO payload trigger rate is around 100 Hz, so that each L2 sector can target a trigger rate of ~3 Hz. A small contribution to the payload trigger rate is expected from an 8-channel low frequency instrument, sensitive to frequencies between 50-300 MHz, designed to enhance PUEO's sensitivity to air showers generated by cosmic rays and tau lepton decays [5].

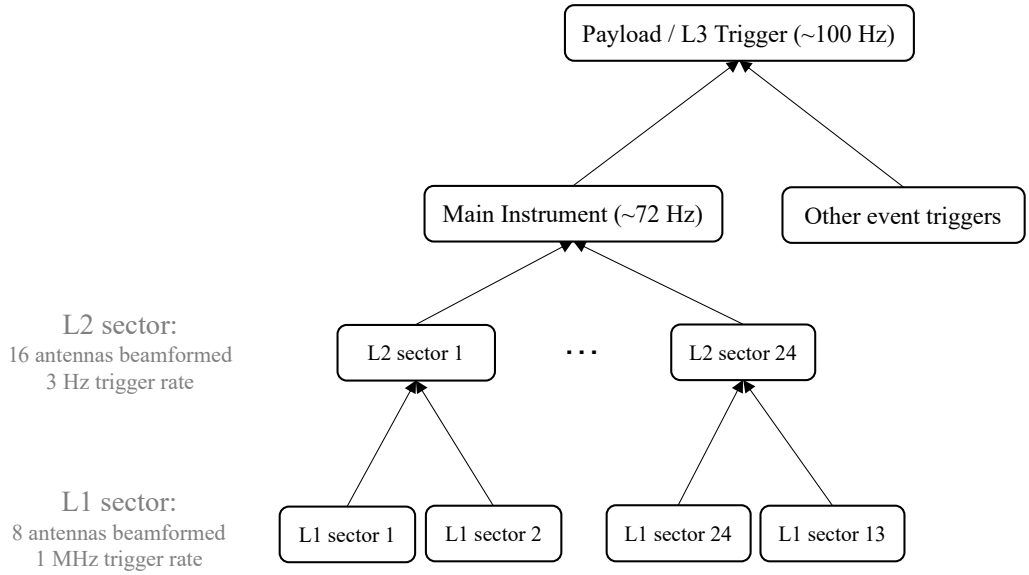


Figure 3.15: Each L2 sector evaluates when either of two corresponding 8-antenna L1 sectors are triggered. When any L2 sector is triggered, the payload is triggered and data is written to storage. This represents a ~ 10^6 reduction in data writing requirements, depending on the duration of data stored for each trigger. The L1 and L2 sectors are both numbered here such that 1-12 and 13-24 correspond to the two polarisations respectively.

Hardware constraints mean that the L2 sectors cannot be evaluated continuously, for all windows at the underlying sampling rate. Each 16-antenna L2 sector is associated with two 8-antenna L1 sectors which share the same antennas. The role of the L1 sectors is to reduce the L2 evaluation rate. Each L2 sector is only evaluated when one of its two corresponding L1 sectors trig-

ger, with the former maintained at a rate of ~ 1 MHz. As L1's output rate is still ~ 10^5 greater than that for L2, it is not expected to significantly affect the performance of the L2 trigger.

The threshold value also depends on the RMS of the noise, whose physical value varies based on temperature, but this varies only at macroscopic timescales and can be normalised in the digitisation process.

During flight the trigger threshold for each L2 sector can be algorithmically configured to provide the target trigger rate, continuously adjusting the threshold down (or up) as the trigger rate trends below (or above) this reference rate. In a simulation context, this would be an expensive computation and is replaced with a predetermined trigger threshold for a given noise profile and coherent summation algorithm. Every change to the coherent summation, such as changing the window size, or other parts of the signal processing, such as changing the digitisation scheme or adding a digital filter, would require a new trigger threshold to be evaluated.

All of the above calculations for every L1 and L2 sector is evaluated at different beam directions using unique offsets between inputs. This represents a vast amount of real time computation, particularly given the power and weight constraints of a balloon payload. This will be made possible by the use of AMD's RFSoC boards.

Chapter 4

Beamforming Trigger Simulation and Optimisation

A simulation of the PUEO coherent beamforming trigger was developed with several overlapping objectives:

- Validate the trigger efficiency benefits of coherent beamforming
- Optimise beamforming components to maximise detector sensitivity
- Demonstrate the extent to which beamforming fidelity can be reduced without significantly impacting performance, as likely to be necessary due to computational constraints
- Contribute as a constituent part of, and enable the evaluation of PUEO simulations.

The trigger simulation was implemented in C++ and ROOT, and has been integrated as part of PUEOSim, the simulation of PUEO developed from the earlier ANITASim [54]. This chapter first outlines the components of the simulated trigger. The findings of the above investigations are then discussed, including an analysis of the coherent beamforming improvements and how they can be impacted by design parameters.

4.1 Overview of Trigger Simulation

Figure 4.1 shows a schematic of the L2 trigger, each of which corresponds to the 16 antennas pointing in one 45° span of the azimuth, as shown in

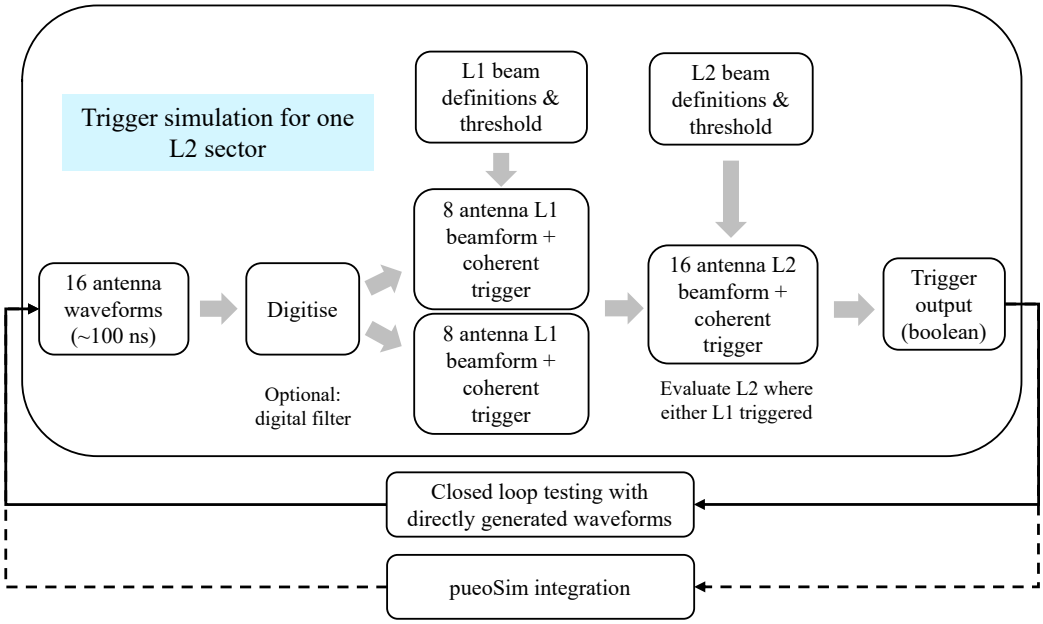


Figure 4.1: Schematic of the PUEO coherent trigger simulation.

figure 3.10. Different L2 sectors are most sensitive to different source directions. The L2 sectors each trigger independently, and can therefore be treated largely in isolation.

The core areas of the trigger design are as follows:

Trigger levels. PUEO's primary beamforming coherent trigger consists of three levels. The L1 sectors each coherently sum the signals received and digitised at a 2.94912 GHz sampling rate from the 8 antennas of two adjacent ϕ directions, as shown in figure 3.10. The beamformed sums are continuously evaluated using the definition 3.16, and at each set of offsets corresponding to a beam direction. Both the rate at which they are evaluated, as well the beams evaluated, are subject to additional study.

Each L1 sector is triggered when any beamformed sum exceeds the L1 threshold, which is set such that the L1 trigger rate is approximately 1 MHz.

The purpose of the L1 trigger is to reduce the rate at which the L2 trigger is evaluated. Each L2 sector is evaluated only when either of the L1 sectors that share antennas with it, is triggered. This is motivated by the limited data transport bandwidth between L1 sectors available in the hardware. Each

L2 trigger coherently sums the signals from the 16 antennas of four adjacent ϕ directions, at each chosen beam direction, with L2 threshold set such that each L2 sector triggers at around 3 Hz. This L2 output rate is motivated by the overall payload L3 trigger rate of ~ 100 Hz, which is constrained by data recording and storage capacity. Note all L1 and L2 sectors are evaluated at both polarisations, which are treated as independent in this analysis.

Threshold evaluation. Both the L1 and L2 sectors discussed above trigger by comparing beamformed sums to thresholds, which are chosen to produce a particular output rate, with given thermal background, trigger configuration, beam selection and beamforming sum parameters. During PUEO's operation this can be determined through a feedback process until the desired rate is achieved. During simulation, generating an adequate number of noise samples to estimate the threshold during runtime of the simulation would be too computationally expensive. This is instead evaluated, for each distinct beamforming configuration, by a configuration of the threshold value through evaluating the distribution of beamformed sum values. Details are presented in section 4.3.

Beam selection and density. As discussed in the previous chapter, a given beamforming sum evaluated for some L1 or L2 sector is coherent in the vicinity of a particular direction, encoded in the relative delay offsets applied to the signal at each antenna before they are coherently summed. Because of this, beamforming sums are also referred to in this work as coherent sums, but it is understood that in most cases the sum's direction does not match an incoming signal.

As well as an associated direction, each such beam has a width that measures the decline in coherence as the source of the signal deviates away from the direction of the beam.

Given digitised data at a 2.94912 GHz sampling rate, and since only delay offsets of integer numbers of samples can be efficiently implemented, the delay offsets will be the nearest integers to the idealised values. The beams retain an approximate direction and width, but no longer have a single well defined direction. This also limits the number of distinct beams, with approximately 400 L1 and 1400 L2 beams available given PUEO's geometry.

This maximum number of beams is not expected to be optimal. A dense array containing most of these may exhaust computational capacity, as well as plausibly reduce the trigger efficiency by raising the trigger threshold due

to a larger number of beamforming sums evaluated. On the other hand, a sparse array of beams would reduce the coherence of reconstructed sums from the directions that have been omitted. With PUEO's greater vertical profile compared to the other dimensions, a different density of beams is expected along the azimuth compared to elevation.

The relationship between the selection of L1 and L2 beams and trigger performance will be presented in section 4.5, optimising triggering efficiency while using a significantly reduced number of beams.

Digitisation scheme. Before the antenna waveforms are fed into the triggers, they are digitised by the analogue-to-digital converters (ADCs) in the RFSoC boards into a discrete representation. Although the ADCs digitise to 12 bits, a lower bit count is necessary given the high computational requirements of beamforming combined with the constraints of the hardware. A digitisation scheme can be defined in terms of 2 parameters: the number of bits used to represent the signal and the value of the noise RMS in that representation.

It is expected that beamforming trigger performance can be largely maintained with a significant reduction of the bit count. However, a systematic analysis would help quantify the relationship between performance and computation. An analysis is presented in section 4.6 of the performance impact of using different digitisation schemes.

Note that this digitisation scheme only applies to the beamforming, with the full fidelity waveforms stored for all triggered waveforms.

Beamforming sum window and step sizes. In the beamforming sum definition given by equation 3.16, a value must be chosen for T, the number of samples in each coherent sum window. Intuitively this should be similar to the duration of the signal to maximise the relative contribution of signal vs. noise in the coherent sum. Another choice is the step size, which is the number of samples between the first sample t_0 of consecutive coherent sums. Similarly to the beam selection, a large step size might result in lower trigger performance, while a very small step size increases computational requirements and increase the trigger threshold. The results relating to these are found in section 4.7.

Digital filtering. The RFSoC hardware being used on PUEO lends itself to digital signal processing, including digital filtering. There are two

primary ways this is expected to be used for PUEO. Firstly, PUEO's digital notch filters - which remove small ranges of frequencies - will be able to mitigate narrow band satellite interference. This takes the place of analogue notch filters on ANITA-IV, which improved the fractional live time of the instrument from 32% in ANITA-III to 91% in ANITA-IV. While both are dynamically reconfigurable, PUEO's digital filters are expected to be more flexibly tuned during flight. This is important as experience from prior flights found additional radio frequencies being used over time, with no reliable way before the flight to identify the relevant bands through publicly available information.

The second application of digital filtering is to improve the signal to noise ratio and therefore trigger efficiency by applying a low pass filter. Since the impulse response drops off rapidly above 800 Mhz, whereas the noise continues up to 1300 Mhz, a low pass filter centred around 800 MHz could enable a reduction in thermal noise within the waveform without significantly impacting the signal, improving PUEO's sensitivity to signals with smaller SNRs. An investigation of this is presented in section 4.8

4.2 Testing Environment

The complexity of the full PUEOSim implementation makes it unwieldy as the input source of most of the above trigger testing. As well as the additional computation time required, the full simulation allows less direct control of the properties of the input signal, exposing the results to impact from a large number of configuration options. A simplified noise and signal model was developed, directly generating signal waveforms at a given SNR as outlined in the previous chapter.

Unlike the integer samples required for the beamforming offsets, the delays introduced in the simulated signals reflect continuous variations in physical distance. The delays are achieved using interpolation of the signals at the relative delays given by equation 3.7. The simulated waveform at each antenna then consists of this signal, which has a size of 128 samples, although with most of the power within a Hilbert envelope of size ~ 40 samples, embedded roughly in the middle of thermal noise of size 256 samples. The exact position of the signal is randomly chosen with a uniform distribution in time, before discretisation and for a chosen reference antenna, to be anywhere within a consecutive range of 32 samples. The signal position for the

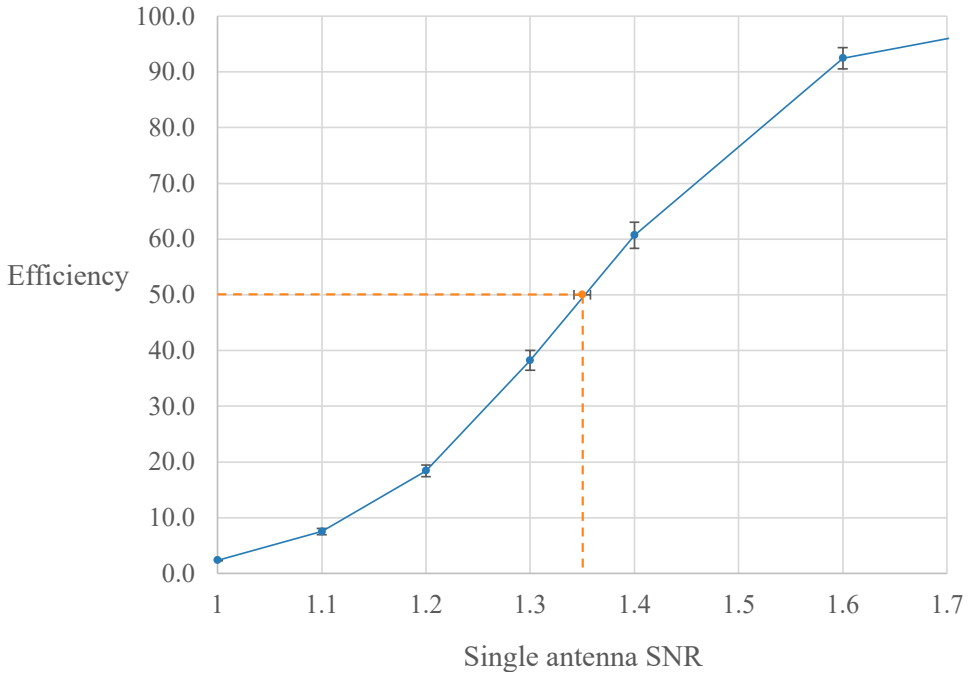


Figure 4.2: An example of the plot showing trigger efficiency variation with single antenna SNR of the received signal. The standard error is shown for both the percentage of signals triggered at each SNR, as well as $SNR_{50\%}$.

other antennas then follow from their geometry relative the the reference antenna. This processes randomises the position of the signal profile within the coherent sum windows. In addition, this also randomises the sampling locations relative to the peaks and other features of the signal profile.

Starting from a window including the first sample, all windows are evaluated at L1 across all chosen L1 beams. If any of them trigger, the same window is also evaluated at L2.

At some SNR and for a given set of coherent beamforming configuration, the measurement of trigger efficiency used in this study is the number of L2 triggers in 1000 repeated simulations, each with independently generated noise. The error in the measurement is evaluated by then repeating this 10 times and taking the standard error. To evaluate $SNR_{50\%}$, in order

to directly compare performance between configurations, this is repeated at several different SNRs. 10 values are evaluated for $\mathrm{SNR}_{50\%}$, each a linear interpolation between the two SNR values whose trigger rates are closest to and either side of the 50% value. The standard error calculated from 10 such calculations yields the error in $\mathrm{SNR}_{50\%}$.

The source direction of the signal is randomly chosen, 30° either side along the azimuthal ϕ direction from the centre of the L2 sector, and 30° either side along the elevation θ of each antenna. Note that the antenna boresights point at 10° below the horizontal plane perpendicular to the payload axis. These chosen source direction ranges are quite large compared to expected sources of event signals based on ice geometries. With a narrower range chosen, fewer total number of beams would be required than outlined in section 4.5, but with the same separations.

Figure 4.2 shows an example of this measurement of the variation of trigger efficiency with signal-to-noise ratio, demonstrating a characteristic S shape. Although $\mathrm{SNR}_{50\%}$ does not fully capture the trigger performance, as each curve can differ in their trigger efficiency at other SNRs, this quantity is a useful indicator that will be used to facilitate comparison.

One consideration not included in these studies is the effect of relative group delay. This is caused by the differential phase distortions from different antennas receiving the signal at different angles away from the antenna's phase centre.

4.3 Threshold Evaluation

Each beamformed sum value is compared to a threshold value to determine the trigger output. This threshold is chosen such that, the target trigger rate is achieved given purely thermal fluctuations. The target rate is sometimes referred to as the accidental rate. The accidental rate for each L1 sector is 1 Mhz, due to hardware resource constraints on the rate at which the L2 trigger is evaluated. The target rate for each L2 sector is 3 Hz, which is imposed by the overall payload trigger rate of 100 Hz.

To reduce computation costs, in the simulation the appropriate threshold is evaluated by generating a large number of noise only waveforms, and calculating their beamformed sum values to find the noise only distribution for that beamformed sum configuration. For L1, this set of steps can be followed:

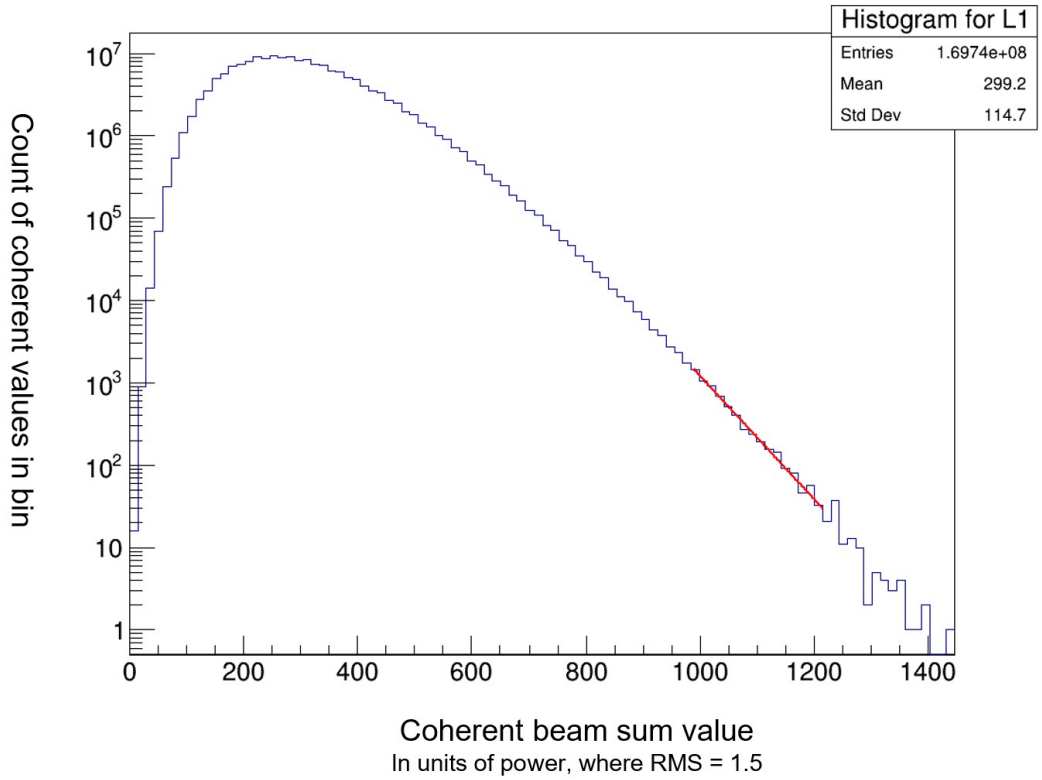


Figure 4.3: Distribution of beamformed sum values for an L1 trigger with 8 noise only waveform inputs. Every window is evaluated using equation 3.16, across all beams for the chosen trigger. The red line shows a log linear fit for large values, which is then used to evaluate the number of beamformed sum values above any given threshold value.

1. Generate thermal noise for each of the 8 input channels of an L1 sector, equivalent to trigger simulation with zero signal. Bit reduction and any other digital processing (such as filters) chosen for the trigger being evaluated, are applied.

2. Generate beamformed sums, using the algorithm in equation 3.16, across all chosen beams as well as all windows with the chosen step size between them.

3. Plot the log of the counts histogrammed into approximately 100 bins. An example can be seen in fig. 4.3 with 1.7×10^8 beamformed sum

values generated

4. To yield a given accidental rate, the corresponding number of windows triggered in this distribution is given by:

$$\frac{\text{L1 accidental rate * Step between windows * count of all windows}}{\text{Sample rate * L1 beam count}}$$

5. Fit a log linear distribution at large beamformed sum values of the histogram

6. Use a geometric sum to evaluate of the number of points which exceed a given beamformed sum value V, as a function of V

7. Combine this function with the total number of beamformed sum values to yield the accidental rate as a function of V. Solve for V at the target accidental rate. For the example in fig. 4.3, the target rate of 1 MHz is achieved with a threshold of 1080

8. Increase the total number of beamformed sum values generated if the bins near the threshold are sparsely populated. A minimum of 10^7 beamformed sums is found to be required.

Similar steps can be followed to evaluate the L2 threshold, requiring 16 inputs to now be simulated. Each of the constituent L1 sectors is evaluated first (across beams and windows), so that only windows which triggered at L1 are evaluated at L2, requiring an update to the equation above. An example evaluation is shown in figure 4.4. The L2 threshold evaluation is more computationally expensive, since generated noise must first be triggered on L1, as well as due to the much lower output rate.

The distributions of beamformed sum values are similar to a χ^2 distribution, if we consider the sum across windows to form a normally distributed variable. There are however a number of distinctions. In most beamforming algorithms there are overlaps between neighbouring windows, so they are no longer independent. Similarly, different beams offset the same antenna data at different values, yielding non-independent statistics. The L2 beamformed sum is only evaluated if the same window is triggered at L1, but the latter can be at any beam direction and involve different input data. The bit reduction scheme also alters the statistics through discretisation and saturation, and the antennas band limit the input noise.

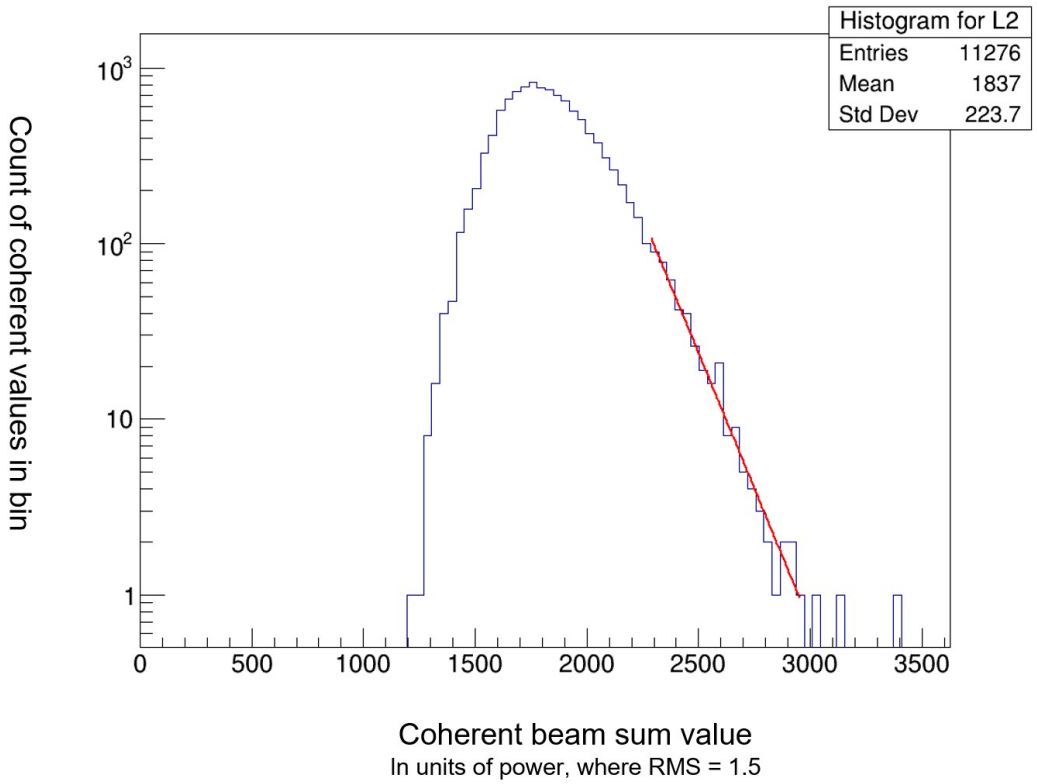
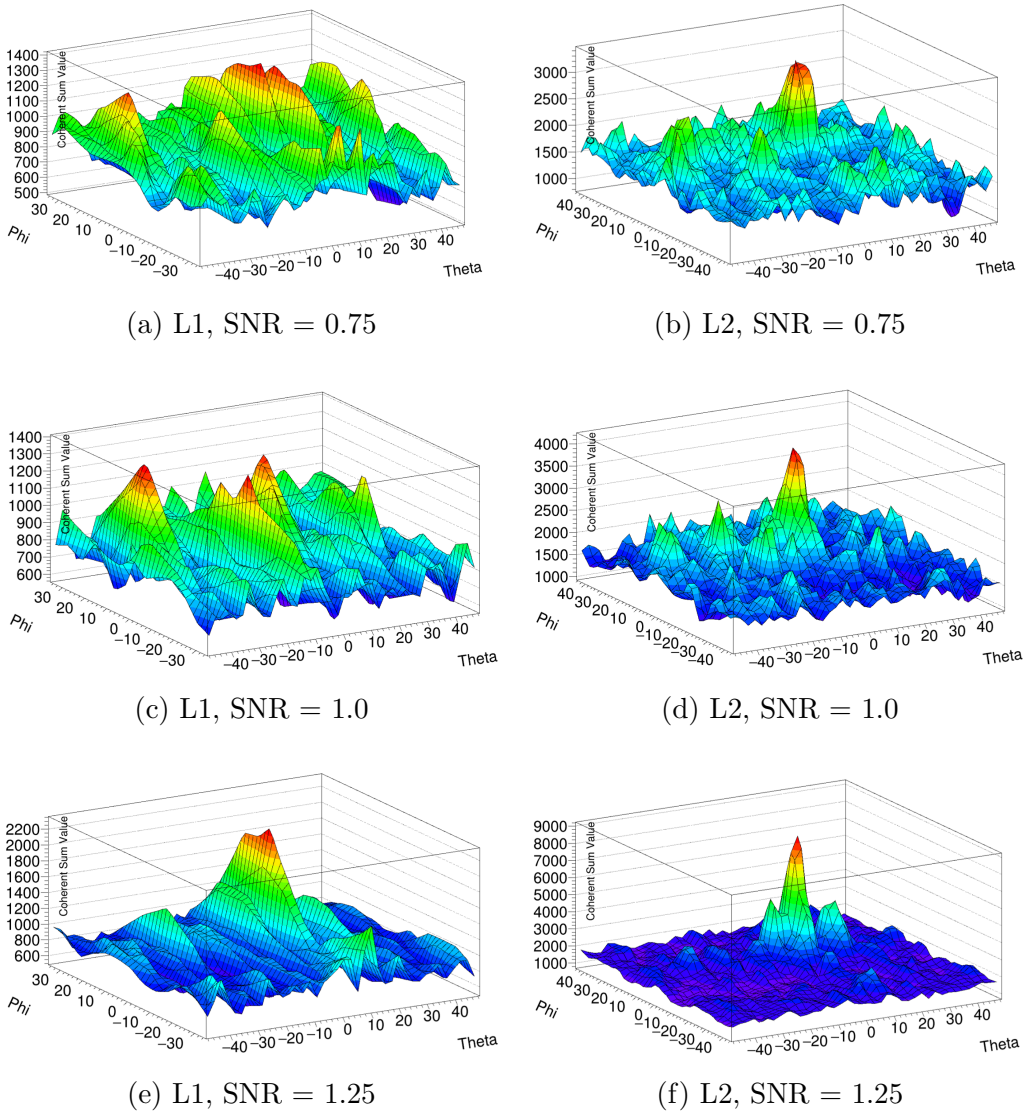


Figure 4.4: Distribution of beamformed sum values for the L2 trigger using 16 noise only waveform inputs, using equation 3.16. A beamformed sum is only evaluated and plotted here, if the corresponding window is triggered at L1 for any beam.

4.4 Trigger Efficiency Benefits

For an illustration of the $\sqrt{N}$ benefit of beamforming on SNR, figure 4.5 shows visualisations of the coherent sums formed using L1 and L2 beams pointed in directions across the θ and ϕ plane. In each plot, a single signal with a source angle $\theta = \phi = 0$ and at the given SNR is embedded within noise of 256 samples. All possible windows are evaluated across different beam directions, with size 16 samples and steps of 8 steps. For each beam, the largest beamformed sum value from among all windows is plotted along the z-axis.

At each signal-to-noise ratio, the beamforming coherent sum can be clearly



(a) L1, SNR = 0.75

(b) L2, SNR = 0.75

(c) L1, SNR = 1.0

(d) L2, SNR = 1.0

(e) L1, SNR = 1.25

(f) L2, SNR = 1.25

Figure 4.5: Visualisation of beamformed sum values at a range of signal to noise ratios, with the L1 and L2 triggers, combining 8 and 16 antennas respectively. As the signal-to-noise ratio becomes higher the peak becomes more visible. L2 results in more prominent peaks due to the summation of twice as many channels. Since the additional antenna channels are at different azimuthal angles, the max beamformed sum value is also more sharply peaked in the ϕ direction. Y-axis shows the beamformed sum value, in units of electric field squared, with different scales used as stronger signals are used at higher SNR, and more antennas are included at L2 vs. L1.

seen to increase the relative prominence of the signal peak compared to the thermal noise background. This is already evident in the case where SNR = 0.75, in plots 4.5a and 4.5b where the signal peak-to-peak is below the root mean square of the noise and thus well below the thermal fluctuations. When data from 16 antennas are added for the L2 trigger, the effective SNR is $0.75 \times 4 = 3$ and the peak corresponding to the signal can be visually identified.

Due to the very low L2 accidental rates that are targeted, signals at SNR < 1 will not generally be triggered despite being apparently visually identifiable. This is because over the very large number of samples, peaks of similar and greater sizes appear from random fluctuations, requiring a high threshold to maintain the small accidental rate. Only at SNR = 1.25, are $30 \sim 40\%$ of the signals expected to be triggered at L2. As it can be seen in plot 4.5f, positive L2 triggers correspond to dramatically elevated values for the coherent sum.

To quantify the benefit of the coherent beamforming trigger, the simulation code was modified to analyse the trigger efficiency of two other trigger scenarios. First, a trigger based on individual antennas was analysed. Due to the limited viewing angle of each antenna, 12 antennas are arranged on this payload with equal azimuthal separation, each corresponding to one L1 sector. Each antenna triggers individually, and is given equal allocation of the same total trigger rate as the 12 L2 sectors of the PUEO trigger. The trigger threshold was then evaluated on this basis, which was then applied to simulated signals of different SNRs. In figure 4.6, the trigger rates at various SNR of this 1 antenna trigger are compared to PUEO's 16 antenna setup.

There is a measured improvement by a factor of ~ 3.3 in $\mathrm{SNR}_{50\%}$. This is lower than the theoretical improvement of $\sqrt{16} = 4$. This could be due to a number of reasons, including integer sample offsets providing imperfect coherence. Another reason could be SNR being defined on peak-to-peak amplitude, where as beamformed sums are over a larger window size (16 or 32 samples), reducing the benefit of the coherent summation.

Another design was analysed, using the same antenna layout as PUEO, but with each coherent trigger carrying out beamforming on vertical arrays of 4 antennas at the same azimuthal angle. This enables a comparison with the proof of concept trigger developed on the RFSoC, in a later chapter. Compared to this 4 antenna trigger, the 16 antenna trigger provides a measured improvement by a factor of ~ 1.9 in $\mathrm{SNR}_{50\%}$.

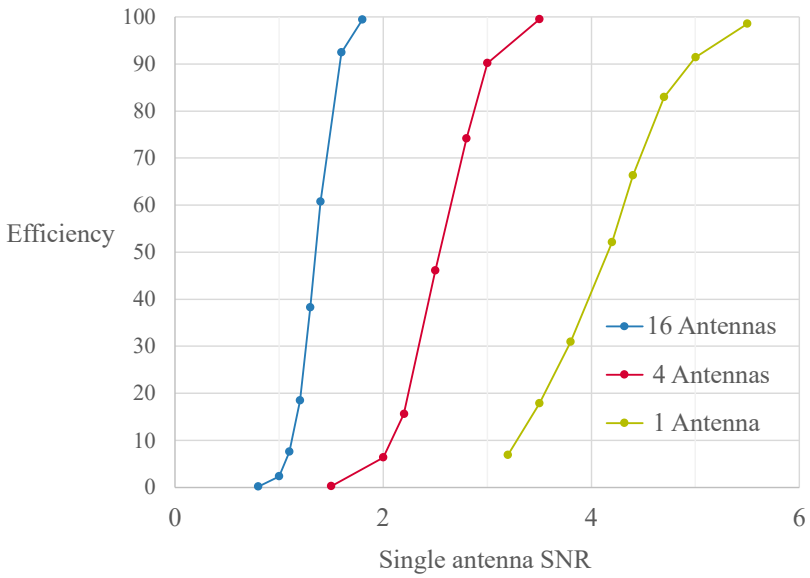


Figure 4.6: Comparison of trigger efficiencies, at varying single antenna SNR, for beamforming with N = 1, 4 and 16. PUEO's coherent beamforming trigger with 16 antennas achieves $SNR_{50\%} \sim 1.3$, a single-level 4-antenna beamforming trigger achieves $SNR_{50\%} \sim 2.5$, and triggering on a single antenna yields $SNR_{50\%} \sim 4.2$.

4.5 Beam Selection and Density

Each beamformed sum in PUEO's beamforming relies on a beam definition, consisting of a set of offsets for the constituent antennas and a corresponding pointing direction. The offsets should be the opposite of the relatively delays of a signal arriving at the antennas from the source direction. They can be evaluated using equation 3.7, together with the geometry of the antennas and the pointing direction.

To define the set of beam directions, we can start from the finite set of all sets of unique integer beam offsets. The limited number is due to the discrete sampling of the data, so that only integer offsets can be implementing, avoiding computationally expensive interpolations. Rounding all offsets to the nearest integer, one can then systematically scan the θ and ϕ plane until

one of the rounded offsets changes to the next integer. A subset of these can then be chosen, in an effort to retain trigger performance while reducing computational costs.

In the above approach, the beam density changes away from the centre of the antenna, and it is not clear what would be the best way to choose the subset of beams. A simpler, alternative approach can be taken instead. Instead of starting from all possible antennas, begin with a grid of beams equally spaced in the θ and ϕ directions. Starting with the beam pointing at $\theta = \phi = 0°$, identify beams at fixed separations of $\Delta\theta_1$ and $\Delta\phi_1$ between adjacent L1 beams, and $\Delta\theta_2$ and $\Delta\phi_2$ between adjacent L2 beams. The offsets thus derived are then rounded to the nearest integers.

To evaluate the impact of beam density on trigger efficiency, $\mathrm{SNR}_{50\%}$ is evaluated for different values of the above beam separations. As noted in section 4.2, the source direction is randomised, in the range of 30° either side along the azimuthal ϕ direction from the centre of the L2 sector, and 30° either side along the elevation θ.

The results for ϕ and θ are plotted in figures 4.7 and 4.8 respectively. At each combination of $\Delta\theta_1$ / $\Delta\theta_2$ and $\Delta\phi_1$ / $\Delta\phi_2$, $\mathrm{SNR}_{50\%}$ is measured and benchmarked against the best performing combination. An increase in $\mathrm{SNR}_{50\%}$ represents decreased trigger performance. The standard error in $\mathrm{SNR}_{50\%}$ is ±0.01 throughout, and the error in the increase in $\mathrm{SNR}_{50\%}$ is ±0.02.

Since the L2 trigger output is $\sim 10^5$ smaller than than L1, and therefore much more restrictive, only configurations where L2 has a higher or equal density as L1 were measured. Red colour indicates effects where the impact on $\mathrm{SNR}_{50\%}$ exceeds the standard error measured.

It can be seen that higher beam densities through reduced beam separations resulted in trigger performance improvements, until approximately $\Delta\theta_1 = 4°$, $\Delta\theta_2 = 2°$, $\Delta\phi_1 = 13°$, $\Delta\phi_2 = 5°$. Additional beam density does not offer significant improvement in $\mathrm{SNR}_{50\%}$, and can be reasonably neglected if necessary to conserve computational resource. In figure 4.7, where ϕ is varied, $\Delta\theta_1 = 2°$ and $\Delta\theta_2 = 1°$ are chosen based on the results of the other plot to ensure performance is not limited by the θ direction. Similarly, in figure 4.8 $\Delta\phi_1 = 10°$ and $\Delta\phi_2 = 5°$.

The higher optimal density in the θ directions makes sense, if one considers that within an L2 sectors, the greatest horizontal separation between two antennas is ~1.7 metres, whereas the largest vertical separation is ~5.1 metres, requiring a finer resolution in θ.

In figure 4.7, there is a region (toward the bottom right) where increased

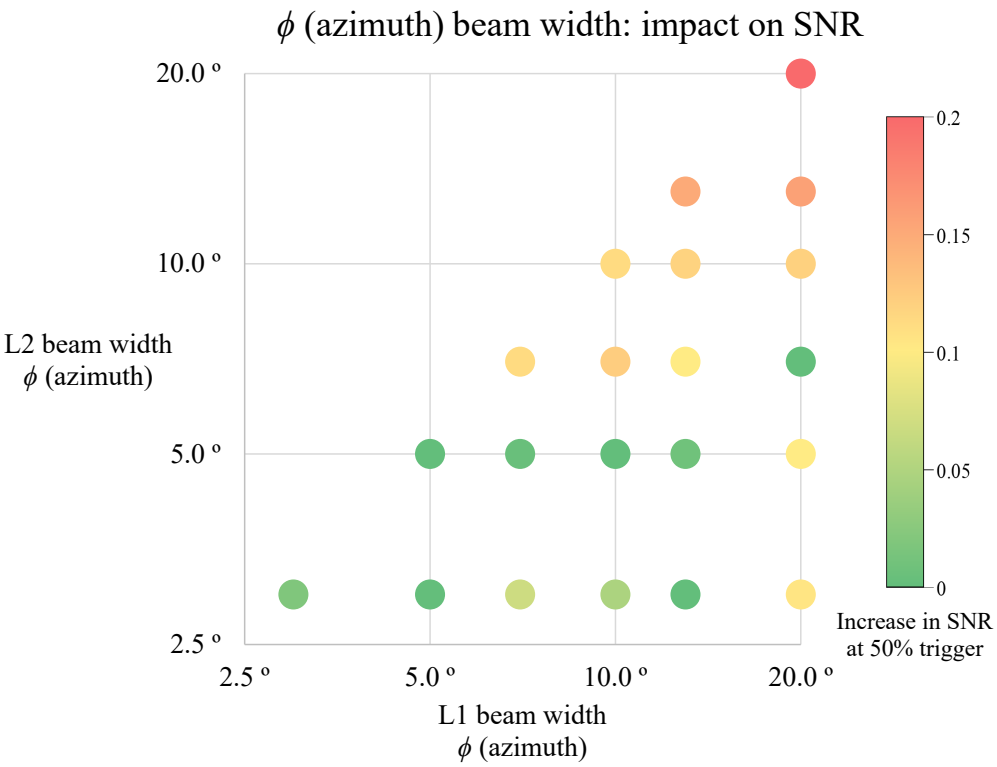


Figure 4.7: Impact on $\text{SNR}_{50\%}$ from variations in ϕ separations between adjacent L1 and L2 beams. The colour of each point shows the difference in $\text{SNR}_{50\%}$ compared to the best trigger performance, i.e. lowest value. Higher density of beams was found to yield greater trigger efficiency, until saturation at $\Delta\phi_1 \sim 13°$, $\Delta\phi_2 \sim 5°$. The standard error in the increase in $\text{SNR}_{50\%}$ is ± 0.02.

L2 beam density appears to be detrimental to the overall SNR when a large L1 beam width of 20 degrees is used. This appears at first counterintuitive. However, this is because a sparser L1 beam was analysed in the azimuth direction than the elevation, similarly motivated as above, since the greatest horizontal separation between two antennas in an L1 sector is ~0.9 metres compared to vertical separation of ~5.1 metres. When a set of beams is used that is very sparse (20°) in L1 azimuth but very dense (3 – 5°) in L2 azimuth,

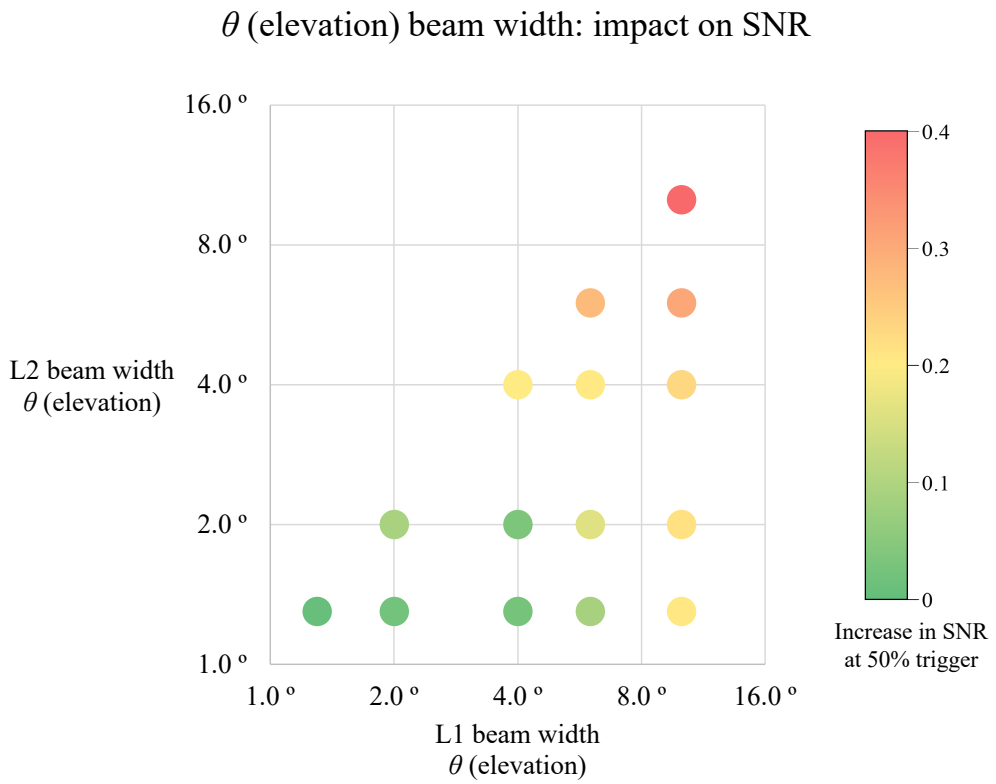


Figure 4.8: Impact on $\mathrm{SNR}_{50\%}$ from variations in θ beam density. Higher density of beams yield greater trigger efficiency, until $\Delta\theta_1 \sim 4°$, $\Delta\theta_2 \sim 2°$. The standard error in the increase in $\mathrm{SNR}_{50\%}$ is ± 0.02.

there is a detriment due to the combined effect of the low chance of L1 beams pointing close to the signal, together with a high L2 trigger threshold due to the larger number of beams.

The above beam densities correspond to 35 beams per L1 sector and 270 beams per L2 sector, which equates to ≈ 7000 beams for the payload. (This number can likely be reduced by a factor of $2 \sim 3$ by reducing the very large elevation angle surveyed here.) In the PUEO whitepaper, beam separations of $\approx 3°$ for θ and $\approx 10°$ for ϕ were assumed instead for the L2 beam density. Such a configuration would require only around 30% of the number of beams, but is expected to be less sensitive based on this study. Figure 4.9 shows the trigger efficiency comparing the performance of this density compared to the

above minimal-optimal configuration. The additional beams result in a ~0.07 improvement in $SNR_{50\%}$, or an additional ~14% of signals triggering at SNR = 1.4.

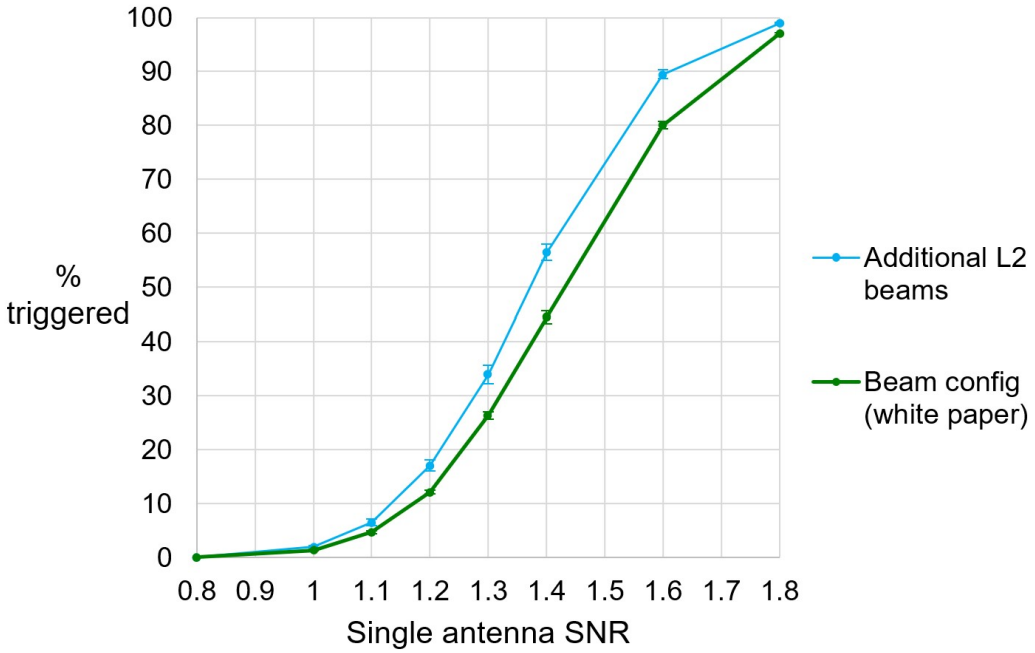


Figure 4.9: Comparison of trigger efficiency across different SNRs between 2 beam densities. In green is the PUEO whitepaper configuration, with $\Delta\theta_1 = 4°$, $\Delta\theta_2 = 3°$, $\Delta\phi_1 = 13°$, $\Delta\phi_2 = 10°$. In blue is a configuration with additional beams, with $\Delta\theta_1 = 4°$, $\Delta\theta_2 = 2°$, $\Delta\phi_1 = 13°$, $\Delta\phi_2 = 5°$.

4.6 Bit Reduction Scheme

FPGAs (Field Programmable Gate Arrays) are components within the RF-SoCs will carry out digital signal processing, including coherent beamforming and digital filtering. The source of the digitised inputs is the ADCs onboard the RFSoCs, that will convert the antenna voltage signals into 12-bit digitised values.

A significant reduction in the number of bits retained for the calculation is required due to the limited computational resources of the onboard FPGAs.

PUEO's sensitivity is to signals with SNR of order 1. This relatively low dynamic range gives reason to expect the trigger performance can be maintained, with a bit reduction scheme that retains only a few of the leading values.

A bit reduction scheme can be primarily characterised by two parameters. The first is the relative scaling of the representation compared to the original values. For this application, it can be specified in terms of the value of the noise RMS in the bit reduced representation. Note that this value does not need to be precisely representable in that representation. For example, in a 3 bit reduced scheme where the only possible values are the integers between -3 and 3, it is still meaningful to choose the scale such that the noise RMS is 1.3 in the reduced scheme. Choosing this scaling allocates the relative available dynamic range above and below the noise, setting the smallest and largest values that can be represented in this scheme.

During flight, the thermal background can be measured as the voltage of the noise RMS varies with temperature. This allows the RMS of the bit reduced waveform to be maintained at a constant value. This is possible since temperature variation is expected to be several orders of magnitude slower than evaluation required to measure their change. This step ensures that the noise does not over or under-saturate the representation.

The second parameter used for characterising a bit reduction scheme is the number of bits used for the representation. This defines the total dynamic range available, beyond which values are saturated to positive / negative values with the maximum possible magnitude, and under which the values are represented as 0. Using a 3 bit example, 8 possible values can be represented, allowing the 7 integers from -3 to 3 to be represented. The remaining value is unused to achieved a symmetric representation. In general, the n-bit representation has possible values from $-(2^{n-1}-1)$ to $2^{n-1}-1$.

Given these two parameters, the 12 bit representation can be multiplied by be a conversion factor and then rounded to the required leading bits. From here, values derived during beamforming from sums and multiplications are provided with increasing numbers of bits commensurate with no additional loss of precision, although reductions can also be considered here.

Figure 4.10 shows the variation of triggered efficiency using a 5 bit representation, with a range of different values for the first parameter, the noise RMS in the bit reduced representation. Window sizes were chosen to be 16 samples and step between windows of 8 samples. Beam separations were chosen as $\Delta\theta_1 = 2°$, $\Delta\theta_2 = 1°$, $\Delta\phi_1 = 10°$, $\Delta\phi_2 = 5°$.

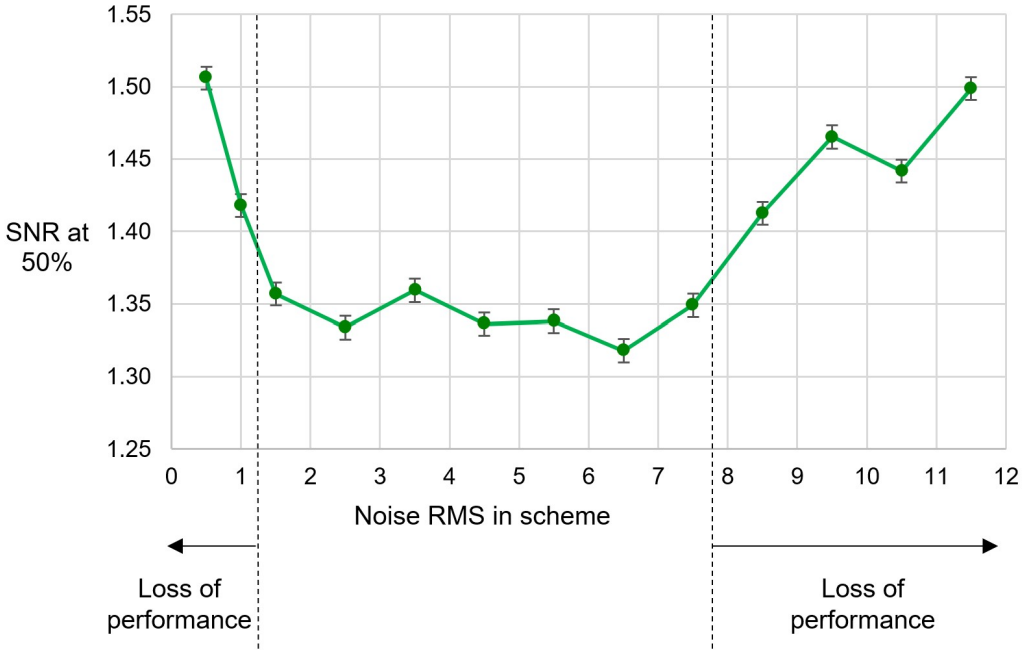


Figure 4.10: The scaling of the input can be parametrised as the value of the noise RMS in the digitisation scheme, shown here for a 5 bit representation. The trigger efficiency is negatively impacted when the RMS scaling is very small or very large is the representation.

In 5 bits, integers from -15 to 15 are possible values. It can be seen that performance deteriorates if the noise RMS is set to above 8, as it reduces the upper dynamic range and truncates the signal peak. $SNR_{50\%}$ is also elevated if the RMS is under ≈ 1.3, where there is not adequate lower dynamic range to achieve the full benefit of coherent summation. There is no noticeable improvement in trigger performance when an intermediate value is chosen, so that additional dynamic range is provided at either end.

These results imply that the noise RMS should be chosen such that, there is at least one bit of dynamic range above and a half bit below. One may then expect that even 3 bits would enable the trigger performance to be maintained. A 3 bit scheme was the working assumption in the PUEO collaboration, prior to this analysis.

Figure 4.11 plots the trigger performance measured as $SNR_{50\%}$, for bit reduction schemes using 3, 4, 5 and 6 bits. Similarly, figure 4.12 compares

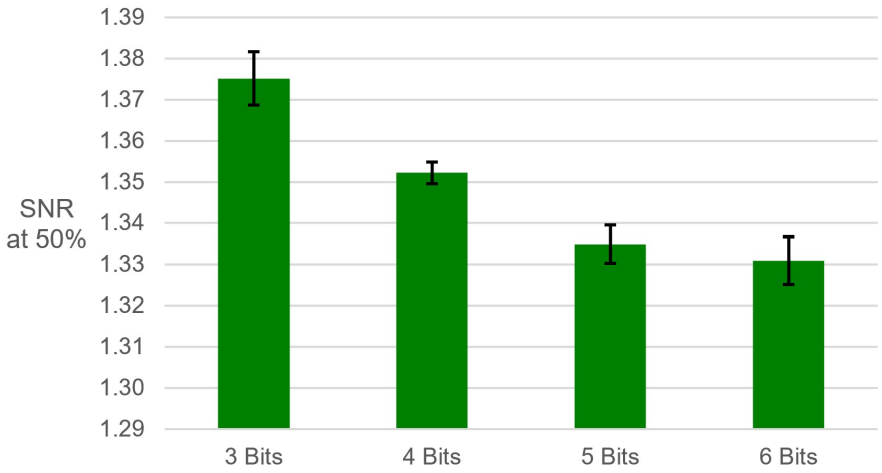


Figure 4.11: Comparison of $SNR_{50\%}$ using increasing number of bits improves the performance up to around 5 bits. Standard error bounds are shown in black.

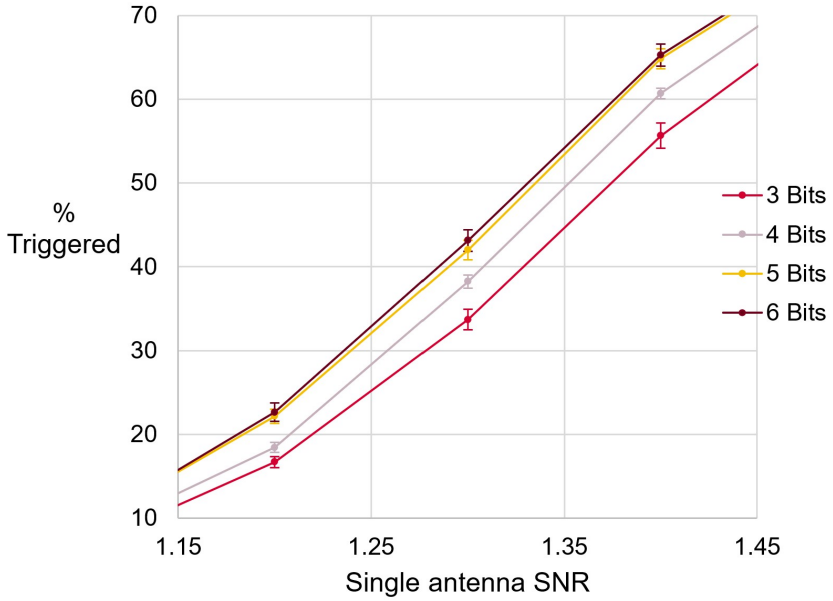


Figure 4.12: Comparison of trigger efficiency at a range of SNR, using an increasing number of bits in the digitisation scheme.

these schemes across a range SNRs. In each case, a number of different scalings are chosen in the range between 1.3 and $2^{n-1} - 2$, and the best performing one is chosen. It can be seen that small but clear improvement in trigger performance is achieved by increasing the scheme from 3 to 4 bits, and again from 4 to 5 bits. Increasing from 5 to 6 bits, the improvement can no longer be discerned from error. A 5 bit scheme is therefore optimal, provided there are available resources to support the evaluation. Compared to a 3 bit representation, it offers a 0.04 increase in $\mathrm{SNR}_{50\%}$.

4.7 Window and Step Sizes

The window size T in equation 3.16 determines the number of samples over which the coherent sum is carried out, before it is compared with the threshold value to determine the trigger output. The sum across antennas aims to increase the effective signal-to-noise ratio, since large random spikes in noise are unlikely to be across multiple antennas, but coherently combined signals do coincide. Similarly, the windowing process sums across time to average out large random spikes in noise, which are unlikely to be across the whole window, while coherently combining the signal.

On this basis, the window size would be expected to improve trigger efficiency if it is similar in magnitude to the width of the signal, where the latter is understood to be the number of samples where the signal is similar in size to the peak. An additional increase could be expected to have a negative impact on trigger efficiency as the signal is 'over-diluted'. Given the variety of ways signal width can be defined, a quantitative analysis would narrow down this relationship, as well as provide a sketch of the sensitivity of the trigger efficiency when the window size deviates from the optimal value.

With a given window size, one is also free to choose the step size which defines the offset between values of t_0 in equation 3.16 for consecutive sums. If the step size is similar to the window size, there will be scenarios where the window does not align well to capture the full width of the signal within a window, reducing the maximum SNR that is reached in any coherent sum. On the other hand, increasingly small steps increase the total number of windows evaluated. To maintain a fixed output trigger rate, this is expected to increase the trigger threshold. This would have the effect of reducing trigger efficiency, though that may be offset by a potential increase in the coherent sum evaluated for the signal through additional windows.

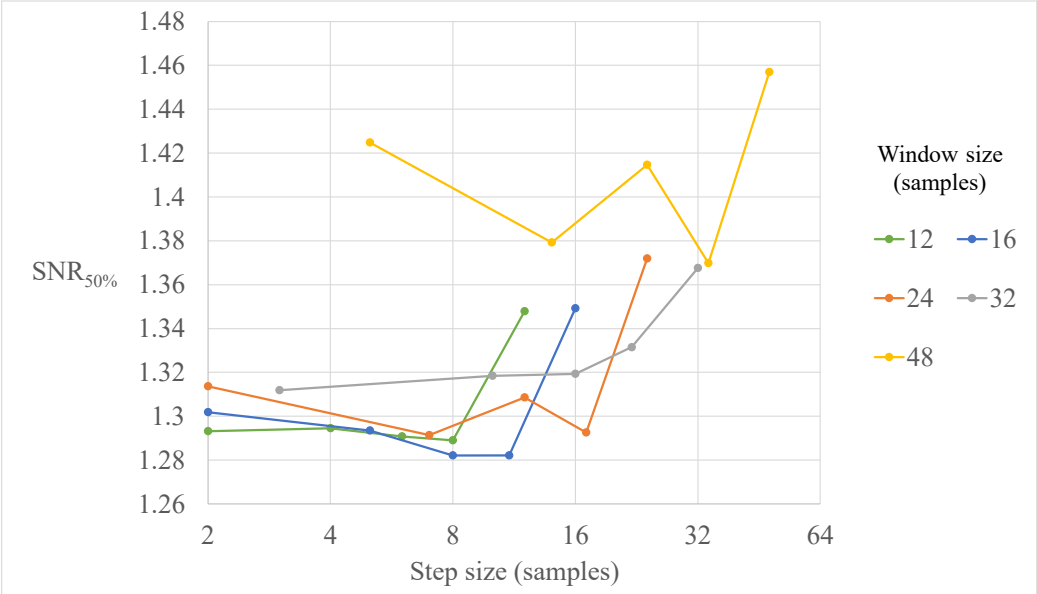


Figure 4.13: Comparison of $SNR_{50\%}$ at different values of window and step sizes. The standard error in $SNR_{50\%}$ is ±0.01 throughout.

Figure 4.13 shows the trigger efficiency when evaluated with a range of different window sizes, each at a number of different step sizes. As with all other analyses in this chapter, the signal was chosen at an angle of observation $\theta_{view} - \theta_{Ch} = 1°$, seen in figure 3.2. The best trigger performance is achieved with a window size of up to 16 samples, and step size of around half of that. In terms of the Hilbert envelope of the signal, the optimal window size corresponds to a drop in amplitude from the peak by a factor of ~3.5. The step size corresponds to a drop in amplitude from the peak by a factor of ~1.5.

It can also be seen that the window size is more deterministic of the trigger efficiency than the step size. At a given window size, $SNR_{50\%}$ remains roughly constant as long as the step size is around half or less that of the window. With a larger window size, even small steps do not offer improved trigger efficiency.

During flight, signals are expected to be observed at a range of different off-cone angles. Compared to the 1° chosen here, a smaller angle of observation yields a largely similar signal as can be seen in figure 3.2, while the signal width increases at larger angles. It makes sense therefore to choose

a window size of ~16. While at the upper range of the optimal values for a signal observed at a small off-cone angle, it is likely to remain optimal for wider signals observed at a larger off-Askaryan-cone angle or dispersed by other effects such as being at a large off-antenna-axis angle compared to the antenna centre.

Prior to this study, analyses for PUEO beamforming frequently used window sizes of 32 samples with step sizes of 16 samples. In the context of a small angle off-cone observations, which have the largest amplitudes, a 16 sample window size combined with an 8 sample step size yields a ~0.04 improvement in $\mathrm{SNR}_{50\%}$.

4.8 Digital Filtering

Figure 4.14 shows the frequency dependence of the magnitude of the signal in figure 3.2 when observed at an angle of 1° off cone. While the thermal noise is flat between 0.24-1.3 Ghz, the Askaryan signal is peaked with majority of power below 800 Mhz. This means that applying a low pass filter with a cutoff frequency of around 800 Mhz would be expected to significantly reduce the thermal noise power while having little impact on the signal power.

The amplitude response of an example low pass filter is shown in figure 4.15. The 11 symmetric coefficients were evaluated using MATLAB's *designfilt* function, with their mirror symmetry reducing the required computational resource. When applied to each channel before coherent summation, the RMS of the thermal noise is reduced to 66% of its original value. This contrasts with the impact on the signal, with the signal peak-to-peak retaining 94% of the unfiltered value and signal RMS at 95% of the original value. There is therefore an ~ 30%, or 0.40 improvement in SNR, defined as the ratio of these two quantities.

However, an analysis found only a ~ 6%, or 0.08 improvement in $\mathrm{SNR}_{50\%}$. This apparent discrepancy is due to the low pass filter altering the statistical distribution of the noise. This is illustrated in a study of the thresholds. The square root of the L1 threshold, which has the same units as the signal strength, only had an 18% improvement, and the square root of the L2 threshold is reduced by 13% after the filter. When considered with a 6% reduction in the signal peak-to-peak, a 7% improvement in $\mathrm{SNR}_{50\%}$ is expected, much closer to the 6% measured.

For a qualitative interpretation of this result, one can consider that the

windowing scheme, which sums over adjacent coherent values, already acts similarly to a low pass filter. This therefore reduces the benefit of an additional low pass filter.

4.9 Summary of Coherent Trigger Simulations

Studies were carried out to answer the questions set out at the beginning of this chapter, using simulated signal and noise profiles as well as beamforming trigger algorithms based on the PUEO payload design. Overall, the beamforming trigger is verified to provide a factor of 3.3 in improved sensitivity measured in $SNR_{50\%}$, compared to the factor of 4 expected. The additional benefit of a low pass filter was found to be a 6% rather than a $\sim 30\%$ reduction in $SNR_{50\%}$ previously assumed in the whitepaper.

There are a number of parameters that can be optimised in design decisions for the trigger, and the simulation was used to identify values for them that can maximise performance. Measured in terms of lowered $SNR_{50\%}$, a

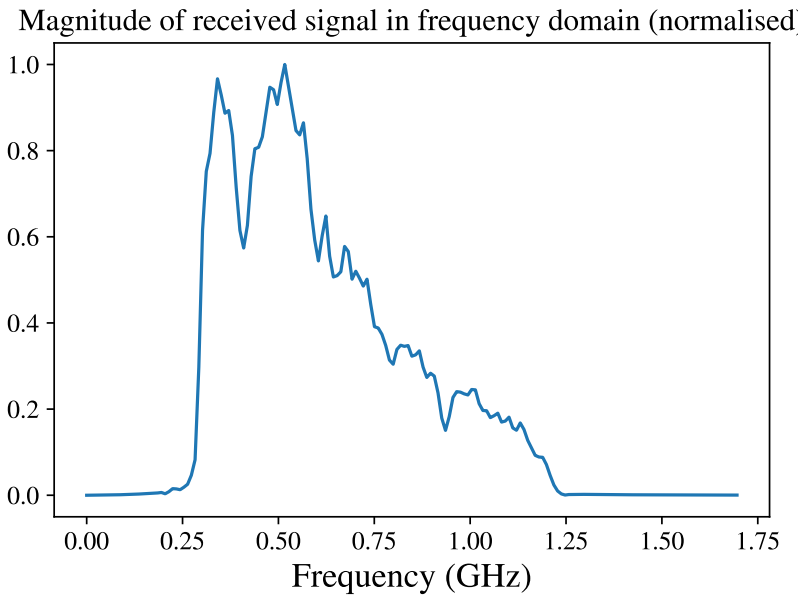


Figure 4.14: Askaryan signal in the frequency domain, when observed at 1° off-cone angle, including the antenna response (but not the cable response / signal chain).

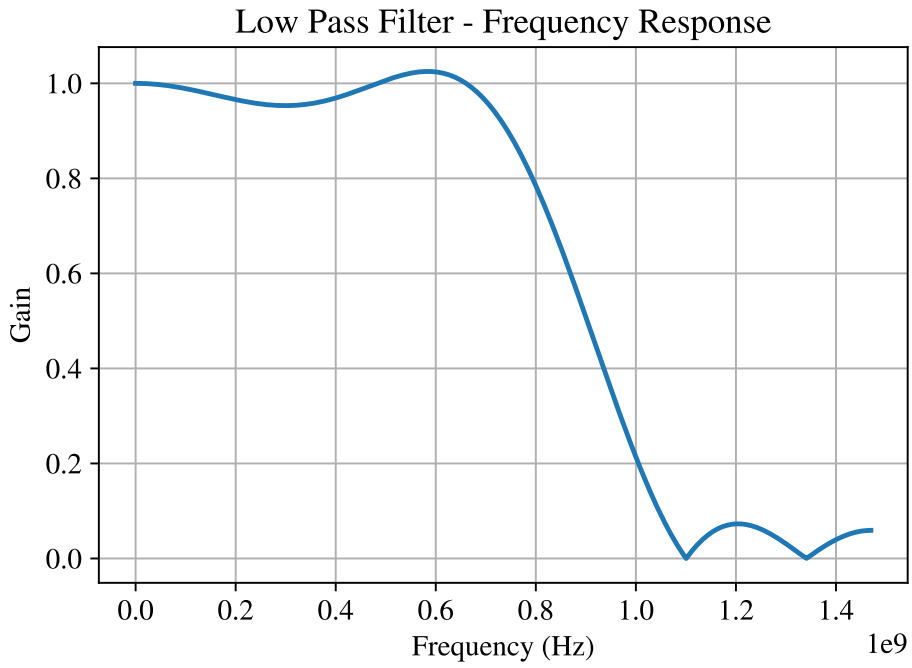


Figure 4.15: Magnitude response of an example low pass filter, designed to reduce the power in noise while largely maintain the signal power.

0.07 improvement was achieved through a denser array of beams compared to the PUEO whitepaper, a 0.04 reduction from using 2 more bits in the digitised representation, and a 0.04 improvement in using 16 rather than 32 samples in the beamformed sum window. The combined improvement of 0.15 is equivalent to a $\sim 11\%$ improvement in $\mathrm{SNR}_{50\%}$.

At the same time, the simulation was used to identify how resolution and fidelity can be reduced in beamforming, given limited computational and storage capacity, without negatively impacting performance. Compared to the extreme scenario of using all 12 bits, the 5 bit scheme used here equates to a factor of ~ 6 reduction in FPGA resources, due to the calculations being dominated by amplitude-squared quantities. Additional order of magnitude reductions in computation are enabled by not evaluating all distinct beam directions, and using steps of 8 samples rather than smaller increments between beamformed sum windows.

This simulation has been integrated within the PUEOSim simulation. This enables full simulations of PUEO's payload over flights, accounting for

factors such as the Antarctic ice sheet and the distribution of the geometry of source signals, with results contributing toward detailed design decisions as PUEO prepares for flight.

Chapter 5

RFSoC Trigger Design

Development for the RFSoC platform is complex, combining the hardware development of the FPGA (or programmable logic, PL), software development for the CPU (or processing system, PS), configuration of onboard radio frequency components, and proprietary technologies for functionality such as clocking. With a wide array of applications, including many for radio communications, much of the development framework and documentation are not directly applicable to PUEO's RFSoC development. Nevertheless, specific steps must be followed to enable functionality necessary for PUEO, in particularly for clocking, synchronisation and the analogue-to-digital converter(ADC). These required processes are spread across hundreds of pages of documentation, or sometimes not documented at all.

This chapter provides an outline of the aspects of RFSoC hardware and software development leading up to the implementation of a proof of concept beamforming coherent trigger. Manufacturer provided tools were used, including AMD Vivado [55] for hardware development and AMD SDK [56] for software development (the latter is now part of the AMD Vitis software development environment [57]).

An overall block design for the hardware was developed using Vivado, linking together key components including the Zynq UltraScale+ MPSoC (Multiprocessor-System-on-Chip) processing system, the RF Data Converter consisting of the ADC and the digital-to-analogue converters (DACs), together with clocking systems and beamforming logic developed on the FPGA fabric. Verilog, a hardware development language (HDL), was used to create blocks of hardware logic such as delay offsets, coherent summation, and threshold comparison. These logic units could be tested using simulation

functionality within Vivado. AMD-provided hardware logic blocks, including block RAM controllers and FIR controllers, were also used to provide functionality.

Once connected to the FPGA through Vivado, AMD SDK was used to develop C++ code running on one of the CPUs of the PS. As well as configuring the RFSoC on startup and resets, software was used to generate simulated noise and signals that were then sent through to the ADC via the DAC channels of the data converter.

The AMD Integrated Logic Analyzer (ILA) [58] was used to monitor the signals of the FPGA. This allowed taps to be created to record the signal at different parts of the FPGA logic. Thus results of intermediate programmable logic could be verified, and the trigger results recorded. To facilitate a large number of runs, as is necessary for measuring trigger efficiency over many simulated signals, an automated approach was developed for communication between these interconnected parts. Automation in the ILA is managed

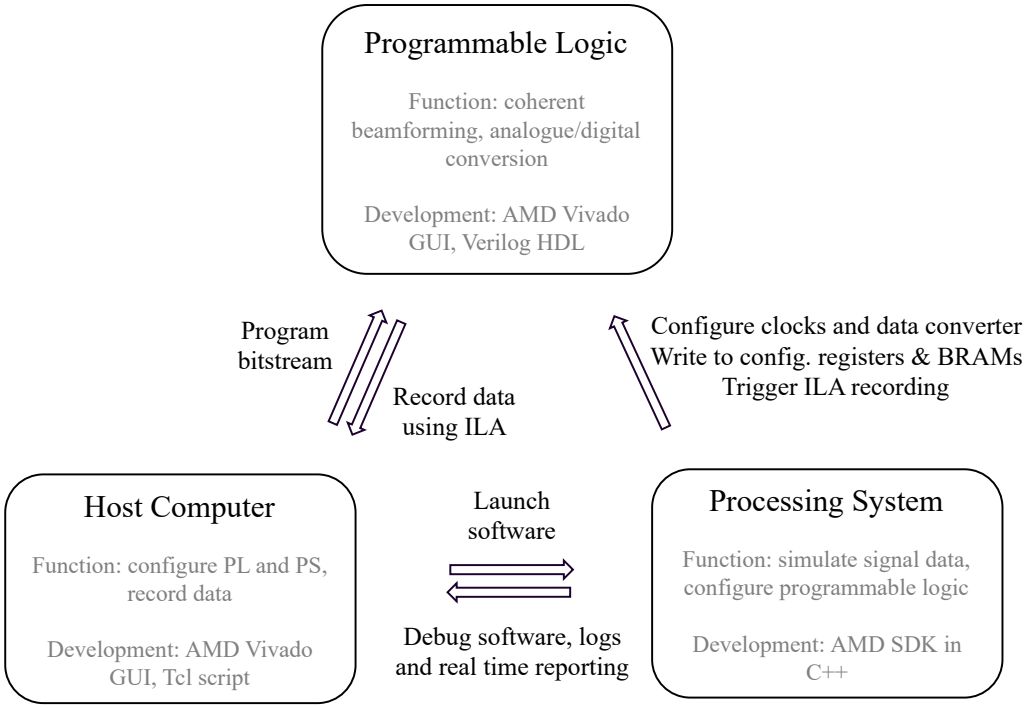


Figure 5.1: Outline of the key components and data/control flows, in the development framework presented for the RFSoC.

through a Vivado Tcl script [59] running on a connected computer. Figure 5.1 shows a schematic of these components of the development framework.

As well as the proof of concept beamforming trigger, another objective of this work is a foundational framework from which future RFSoC developments can draw from. While there are reference designs from the manufacturer, as well as the open source PYNQ project [60], development of the necessary features using these as bases was found to be challenging. This was due to a number of reasons, including limited documentation, a broad range of applications, and complex designs that did not lend themselves to modification.

In contrast, the board design shown here is modular, with key features such as multi-tile-synchronisation enabled through a small number of specific procedures. Except for the clocking setup specific to the ZCU111 evaluation board, the set up here is directly transferrable to other RFSoC hardware. Earlier iterations of this work was published in the proceedings of the 37th International Cosmic Ray Conference [61]. Implementation details such as code and screenshots are provided, both in this chapter and in the appendices, in the interest of providing a reference for similar developments, which was found to be lacking.

This work has been used by S. Prohira, assistant professor at the University of Kansas, as the basis of the design for the RFSoC-based radio trigger detector of the Radio Echo Telescope for Cosmic Rays (RET-CR) [62], which was deployed in May 2023.

5.1 Vivado Design Suite - Hardware Design

The AMD (formerly Xilinx) Vivado Design Suite is an integrated design environment for the design, synthesis and analysis of board designs that are ultimately based on hardware description languages (HDLs). A screenshot of the work flow is shown in figure 5.2.

The majority of user input is through the block design editor within the *IP Integrator*. IP, or Intellectual Property, references hardware functionality coded through HDLs such as Verilog or VHDL, and integrated within the board design as a component. *Synthesis* translates the abstract HDL design into a technology-mapped description of the connections in the target hardware device's available resources such as slices, LUTs and flip-flops, carrying out optimisations along the way. Minimal user input is required in this step.

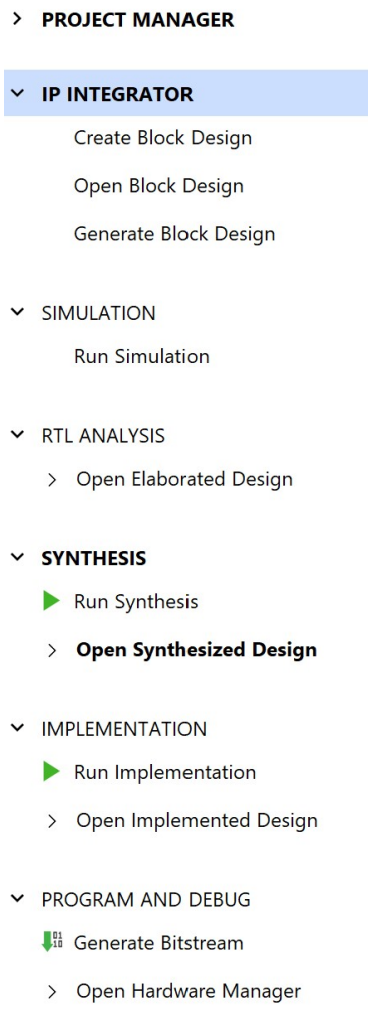


Figure 5.2: Vivado's Flow Navigator, showing the main steps of the development process.

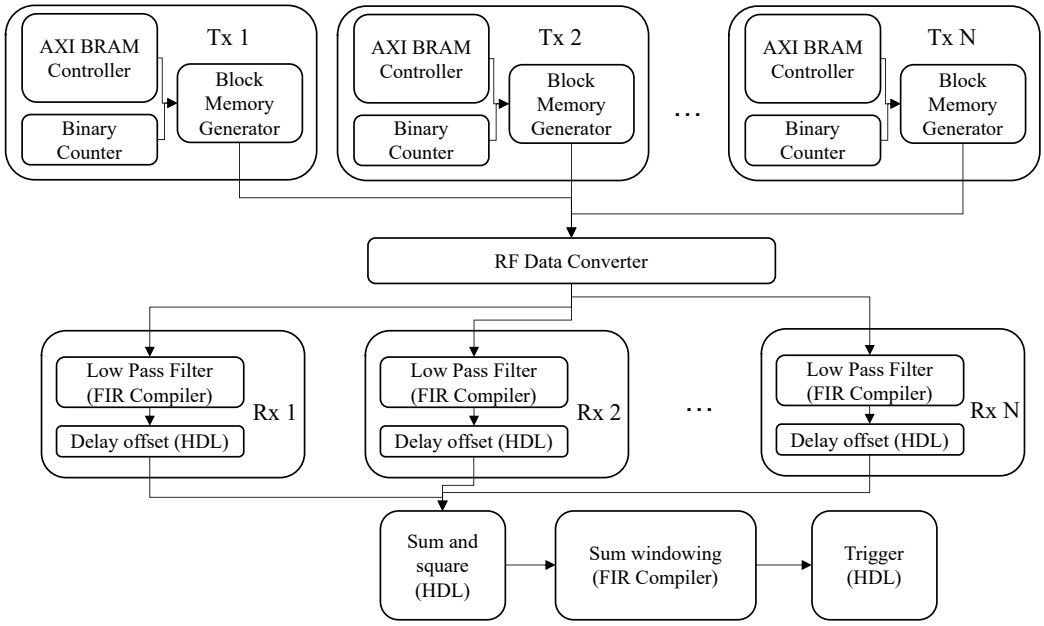


Figure 5.3: A schematic of the main functional components of the proof of concept trigger.

After *Synthesis*, the *Set Up Debug* step can be used to add ILAs to record and analyse data in particular locations of the FPGA fabric.

Following this, the *Implementation* step determines for each required logic resource, a specific choice such that their physical positions ('placement')and interconnections ('routing') enable signals to travel between logic blocks optimally. Specifically, the timing of signals will be assessed against timing constraints to ensure that all such requirements are met. Timings and other errors raised at this step can be resolved by additional iterations of the block design.

Generate Bitstream compiles and encodes the implemented design for configuration. This also includes board support files that are useful for the PS software's interaction with the PL, such as addresses for configuration registers and block RAM. The *Hardware Manager* allows the bitstream to be programmed on the RFSoC, although this can also be done through the processing system. This is also where the previously configured ILA can be used to capture internal signals of the FPGA.

Figure 5.3 shows an overview of the proof of concept hardware design

developed, which corresponds to a trigger of four antenna at the same ϕ angle, forming a vertical line, and consisting of half the antennas of an L1 ϕ sector in figure 4.1. Only 4 channels are used to do a limitation of the XM500 balun board. The trigger performance of the design is as expected from the C++ software simulation of the previous chapter. In addition, the proof of concept trigger provides a framework for further hardware and software development on the RFSoC, with a solution for the elaborate and complex clocking systems shown to have minimal jitter between channels.

Vivado-provided configuration blocks enable implementation of some of the required hardware functionality, with GUI-based connections and built-in links with the hardware available. Customisation options provide additional flexibility, allowing e.g. the FIR compiler to fill different roles. As they are provided by the manufacturer, they are also expected to be robust as well as efficient. Details for the configuration of these manufacture provided block design modules are given in Appendix A.

5.1.1 Creating Logic with HDL

Additional functionality can also be implemented directly using a hardware description language (HDL), most commonly in either VHDL or verilog. HDL code describe the structure and behaviour of digital circuits. Unlike software code running on CPUs, where instructions are largely sequential, HDL describe hardware where operations in different parts are concurrent. Data flow between such operations at the regular intervals provided by the relevant clock.

Functional modules are referred to in Vivado as IP (intellectual property), and user created modules can be set up using the *Create and Package New IP* wizard. The wizard allows the creation and management of AXI4, AXI4 lite and AXI4 stream interfaces, as shown in figure 5.4. AXI4 lite implements a subset of the AXI4 protocol, with a simpler interface that enables easy implementation of device registers that can be read from and written to. The AXI4 full interface implements the complete AXI4 protocol, supporting higher performance memory-mapped data access. AXI4 stream is another simplification of the AXI4 protocol, with unidirectional data flow that do not require addressing and instead rely on simple control signals.

Vivado will then automatically implement the selected ports in the chosen protocol in HDL code. Unique names will be assigned to any registers, whose addresses can then be accessed from C code running on the processing system

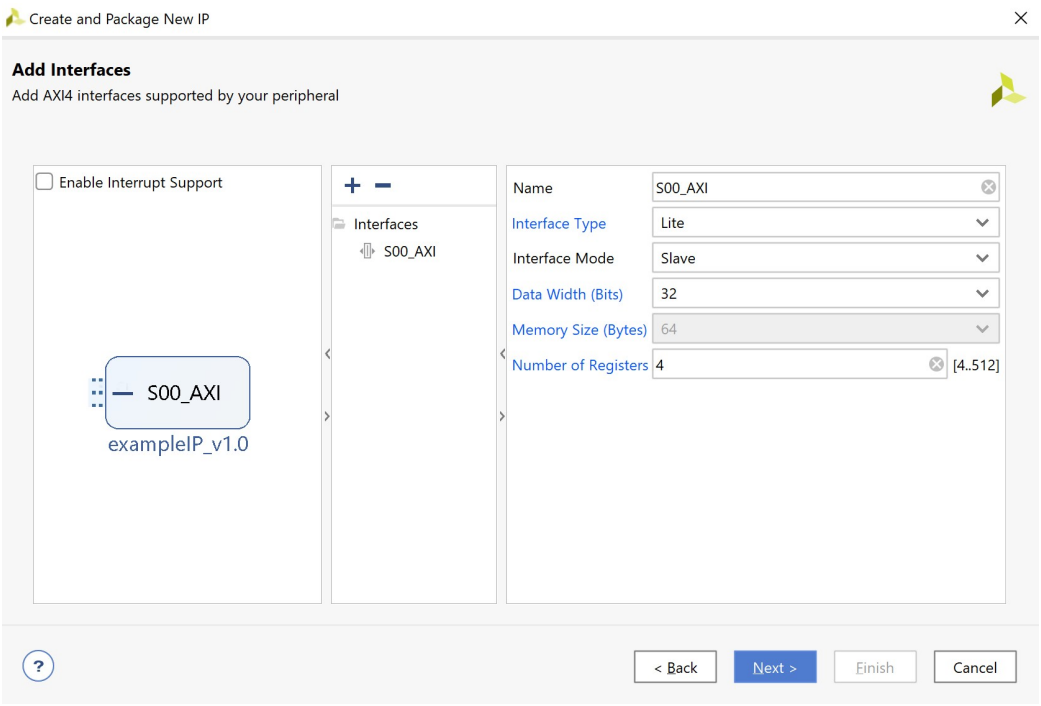


Figure 5.4: Vivado's new IP wizard enables the automatic creation of AXI4 interfaces. Shown here is the AXI4 lite interface, which can be connected to the PS and is a streamlined way to integrate registers that are readable and modifiable during runtime. This allows hardware settings to be dynamically varied at runtime.

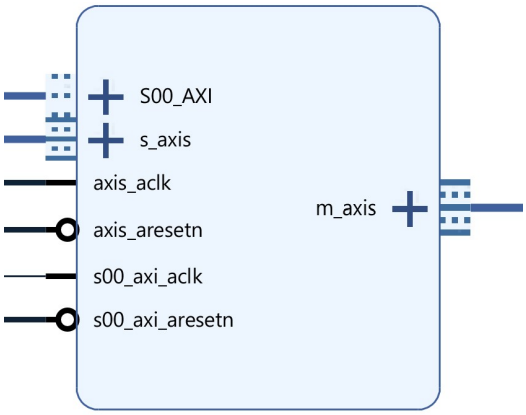


Figure 5.5: A custom board design module developed with ports managed by Vivado, along with custom logic from verilog code for beam delays.

through a look up of the board support package exported as part of the board design. User logic can then be added to this machine generated code, with suggested locations bookended by comments. Verilog is chosen for the implementation of the HDL code.

Delay / Beam Offset

The delay IP, shown in figure 5.5, implements the beam offsets by inserting the required additional sample latency along the data path for each antenna. The operation corresponds to the inclusion of the offsets o_i^b in equation 3.16, in addition to any offsets relating to differences in the physical cabling of different channels. Below is the portion of the verilog code implementing the logic of the delays, inserted in the position indicated by comments provided in code prepared by the new IP wizard:

```
1     // Add user logic here
2     localparam integer DIN_WIDTH = 96;
3     reg [5:0] delay_samples = 6'b000000;
4     reg [DIN_WIDTH-1:0] sampleShiftReg [1:0];
5     reg [DIN_WIDTH-1:0] blockShiftReg [7:0];
6     reg [DIN_WIDTH-1:0] outputReg;
7     integer i;
8
```

```
9       always @(posedge s_axis_aclk)
10      begin
11          delay_samples <= slv_reg0[5:0];
12          sampleShiftReg[0] <= {s_axis_tdata[127:116],
            ↪  s_axis_tdata[111:100], s_axis_tdata[95:84],
            ↪  s_axis_tdata[79:68], s_axis_tdata[63:52],
            ↪  s_axis_tdata[47:36], s_axis_tdata[31:20],
            ↪  s_axis_tdata[15:4]};
13          sampleShiftReg[1] <= sampleShiftReg[0];
14
15          case (delay_samples % 8)
16              0: blockShiftReg[0][DIN_WIDTH-1:0] <=  sampleShiftReg[0][0+:
                ↪  8*12];
17              1: blockShiftReg[0][DIN_WIDTH-1:0] <= {sampleShiftReg[0][0+:
                ↪  7*12],sampleShiftReg[1][DIN_WIDTH-1-: 1*12]};
18              2: blockShiftReg[0][DIN_WIDTH-1:0] <= {sampleShiftReg[0][0+:
                ↪  6*12],sampleShiftReg[1][DIN_WIDTH-1-: 2*12]};
19              3: blockShiftReg[0][DIN_WIDTH-1:0] <= {sampleShiftReg[0][0+:
                ↪  5*12],sampleShiftReg[1][DIN_WIDTH-1-: 3*12]};
20              4: blockShiftReg[0][DIN_WIDTH-1:0] <= {sampleShiftReg[0][0+:
                ↪  4*12],sampleShiftReg[1][DIN_WIDTH-1-: 4*12]};
21              5: blockShiftReg[0][DIN_WIDTH-1:0] <= {sampleShiftReg[0][0+:
                ↪  3*12],sampleShiftReg[1][DIN_WIDTH-1-: 5*12]};
22              6: blockShiftReg[0][DIN_WIDTH-1:0] <= {sampleShiftReg[0][0+:
                ↪  2*12],sampleShiftReg[1][DIN_WIDTH-1-: 6*12]};
23              7: blockShiftReg[0][DIN_WIDTH-1:0] <= {sampleShiftReg[0][0+:
                ↪  1*12],sampleShiftReg[1][DIN_WIDTH-1-: 7*12]};
24              default: blockShiftReg[0][DIN_WIDTH-1:0] <=
                ↪  sampleShiftReg[0][0+: 8*12];
25          endcase
26
27          for (i=0; i<7; i=i+1)
28                  blockShiftReg[i+1] <= blockShiftReg[i];
29          outputReg <= blockShiftReg[delay_samples/8];
30      end
31
32      assign m_axis_tvalid = 1'b1;
```

```
33      assign s_axis_tready = 1'b1;
34      assign m_axis_tdata[DIN_WIDTH-1:0] = outputReg[DIN_WIDTH-1:0];
35      // User logic ends
```

Lines 2 to 7 create internal parameters and registers for the data processing. Note that 12 bits are kept for each sample from the 16 provided at input, and 8 samples are held in parallel at each AXI stream clock cycle. Since negative delays cannot be implemented in a circuit, the delay offset values for all beams are defined with a bias to be all positive.

slv_reg0 is one of the AXI4 lite registers implemented by the custom IP wizard, allowing the beam delay to be dynamically set by the PS through the *S00_AXI* port. *sampleShiftReg* rearranges the samples within one clock cycle to align with the total offset, with the remaining delay samples actioned by *blockShiftReg*. For simplicity, the *tvalid* and *tready* AXI stream control signals are set to always true.

To test the functionality of the module against a number of different scenarios, Vivado also provides a simulation environment in the *Create and Package New IP* workflow. Using a verilog file to set up the required clock and test data input, this functionality was used to test the HDL code is working as expected.

Sum and Square

Similarly to the above, the following code implements the digital circuits for the sum over antenna i, and taking to a power of 2, in equation 3.16. Note that *assign* statements are used for continuous assignment without regard to clock cycles, driving the value on the right-hand side to the left-hand side wires. The *generate* block between lines 28 and 37 enables the instantiation of a repetitive set of such assignments.

On the other hand, the *always* block between lines 15 and 26 is used to carry out the instruction at the specified event, which in this case is the positive edge of the AXI stream clock signal. The timing separation allows evaluations to be carried out in order, preventing race conditions. The use of pipe registers allow complex operations, such as a multiplication, additional clock cycles to complete.

```
1
2       // Add user logic here
3       reg signed [12:0] int_1a_sum [7:0];
4       reg signed [12:0] int_1b_sum [7:0];
5       reg signed [13:0] int_2_sum  [7:0];
6       reg signed [27:0] int_sq_pipe1 [7:0];
7       reg signed [25:0] int_sq_pipe2 [7:0];
8       reg signed [25:0] int_sq [7:0];
9       wire signed [11:0] s00_signed [7:0];
10      wire signed [11:0] s01_signed [7:0];
11      wire signed [11:0] s02_signed [7:0];
12      wire signed [11:0] s03_signed [7:0];
13      integer i;
14
15      always @(posedge s_axis_aclk)
16      begin
17          for(i=0; i< 8; i = i+1)
18              begin
19              int_1a_sum[i] <= s00_signed[i] + s01_signed[i];
20              int_1b_sum[i] <= s02_signed[i] + s03_signed[i];
21              int_2_sum[i] <= int_1a_sum[i] + int_1b_sum[i];
22              int_sq_pipe1 [i] <= int_2_sum[i] * int_2_sum[i];
23              int_sq_pipe2 [i] <= int_sq_pipe1 [i][27:2]; //DSP input
                 ↪  restricts to top 26 bits
24              int_sq [i] <= int_sq_pipe2 [i];
25              end
26      end
27
28      generate
29          genvar j;
30          for (j=0; j<8; j=j+1)
31              begin
32                 assign s00_signed[j][11:0] = s00_axis_tdata[12*j+: 12];
33                 assign s01_signed[j][11:0] = s01_axis_tdata[12*j+: 12];
34                 assign s02_signed[j][11:0] = s02_axis_tdata[12*j+: 12];
35                 assign s03_signed[j][11:0] = s03_axis_tdata[12*j+: 12];
36              end
```

```
37      endgenerate
38
39      assign m00_axis_tdata = {6'b000000, int_sq[7], 6'b000000, int_sq[6],
        ↪  6'b000000, int_sq[5], 6'b000000, int_sq[4], 6'b000000, int_sq[3],
        ↪  6'b000000, int_sq[2], 6'b000000, int_sq[1], 6'b000000,
        ↪  int_sq[0]};  //pad to be byte aligned
40      assign m_summed_axis_tdata = {2'b00, int_2_sum[7], 2'b00,
        ↪  int_2_sum[6], 2'b00, int_2_sum[5], 2'b00, int_2_sum[4], 2'b00,
        ↪  int_2_sum[3], 2'b00, int_2_sum[2], 2'b00, int_2_sum[1], 2'b00,
        ↪  int_2_sum[0]};
41      assign m00_axis_tvalid = 1
42      // User logic ends
```

Trigger Control

The trigger control block compares the value of the coherent sum to the trigger threshold, resulting in a trigger if the latter is exceeded. In the verilog code below, it can be seen that the threshold value can be set through software running on the PS by writing to the register *slv_reg0*.

This module has additional roles in the interactions between the PS, PL and the host system capturing signals using the ILA. When a new signal candidate is provided to the coherent trigger, the PS is used to update an AXI4 lite register *slv_reg1* to indicate this. The value in this register is connected to an output, that can then trigger the ILA running on Vivado on the host computer. A distinct runID associated with this signal is similarly written to *slv_reg2*, and recorded using the ILA. Although it is commented out below, it can be seen that intermediate calculations such as the coherent sum values, can also be output here to be recorded by the ILA for analysis.

```
1       // Add user logic here
2       wire signed [31:0] threshold = slv_reg0[31:0];
3       wire signed [31:0] toCompare [7:0];
4
5       reg record_switch;
6       reg [31:0] runID;
7
```

```
8       reg trigger;
9
10      always @(posedge s_axis_aclk)
11      begin
12
13          // we chop last two bits from threshold, since the preprocessor
             ↪  also did this to the data
14          trigger <=    (toCompare[0] > threshold)
15                      | (toCompare[1] > threshold)
16                      | (toCompare[2] > threshold)
17                      | (toCompare[3] > threshold)
18                      | (toCompare[4] > threshold)
19                      | (toCompare[5] > threshold)
20                      | (toCompare[6] > threshold)
21                      | (toCompare[7] > threshold) ;
22          record_switch <= slv_reg1[7:0];
23          runID <= slv_reg2[31:0];
24      end
25
26      generate
27          genvar j;
28          for (j=0; j<8; j=j+1)
29              begin
30                 assign toCompare[j] = s00_axis_tdata[PACKAGE_WIDTH*j+:
                    ↪  DATA_WIDTH];
31              end
32      endgenerate
33
34      //assign m_summed_axis_tdata = {int_2_sum[7], int_2_sum[6],
         ↪  int_2_sum[5], int_2_sum[4], int_2_sum[3], int_2_sum[2],
         ↪  int_2_sum[1], int_2_sum[0]};
35
36      assign m_triggered_any_axis_tdata = trigger;
37      assign m_recordSwitch_axis_tdata [0] = record_switch;
38      assign m_runID_axis_tdata = runID;
39      assign s00_axis_tready = 1;
```

```
40      // User logic ends
```

5.1.2 Additional Board Design Considerations

As well as using IP modules provided by Vivado, and creating custom logic using an HDL such as verilog, there are also other tools available for creating hardware design. MATLAB's System Generator module provides a graphical environment for creating HDL code at a higher level of abstraction, allowing rapid prototyping as well as the integration of MATLAB's library of functionality, all without any prerequisite HDL knowledge. This includes digital signal processing (DSP) block and libraries that are optimised for AMD devices. Similarly, Vivado's High-Level Synthesis (HLS) tool allows HDL code to be created using C++ code, providing a higher level of abstraction than coding directly in VHDL or verilog. HLS also provides a number of techniques for HDL code optimisation along different criteria, allowing control of the balance between performance, area, and power. Indeed, it was found that the same logic produced using these tools used fewer FPGA resources than the simple verilog implementation above.

The block RAM and its associated generator were used to provided a connection between data stored and addressed in memory, and that being processed in the PL fabric through AXI streams. This is commonly handled instead using Direct Memory Access (DMA) transfers through a DMA controller. If using DMA, care needs to be taken for the correct and non-trivial configuration of the transfer protocols, including control signals such as *TLAST*.

Numerical representation should also be treated with care, ensuring that blocks transmitting and receiving data are interpreting binary values in the same way, e.g. whether they are signed or unsigned. The RF data converter AXI stream interface, as set up in this design, requires 128 bits for each channel. This corresponds to 8 parallel samples of 16 bits each. Only the most significant 12 of these bits correspond to data, with the convention that the leading bit is the least significant.

5.1.3 Design Analysis and Reporting

In the majority of cases, a board design will encounter a range of issues and errors during the synthesis and implementation stages. These were resolved for the proof of concept trigger through references to the Vivado user guide as well as manufacturer managed forums. After implementation is successfully completed, the device window displays an interactive graphical representation of the die for the target device, shown in figure 5.6. One can zoom into specific device resources and make any required updates to placement and routing.

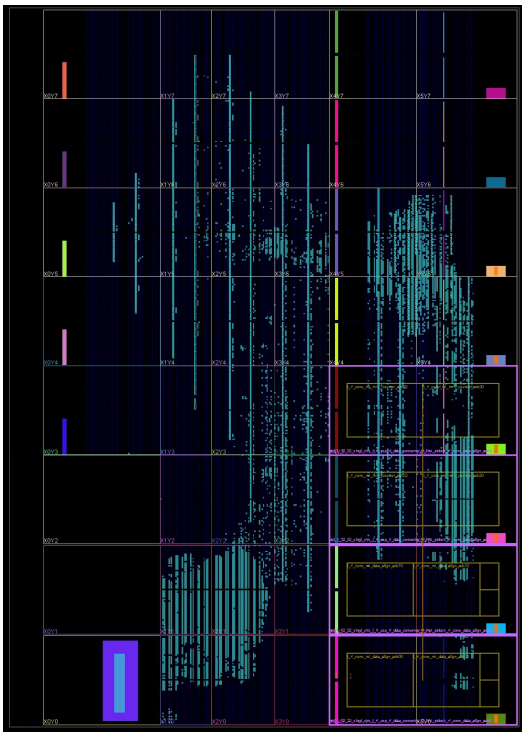


Figure 5.6: The device window shows the placement after implementation of resources, such as block RAMs, DSPs, configurable logic blocks and clock regions. At higher zoom levels, different levels of abstraction are used to show approximate placement and congestion. Black spaces correspond to unused resources, and teal showing active logic cells consisting of lookup tables and flip-flops that implement the FPGA logic.

In addition, a number of reports can be generated on the timing perfor-

mance, power consumption, and resource utilisation of the design. Figure 5.7 displays the timing summary report of the proof of concept trigger implemented. There are three main categories of timing slack measurements, which are the margin of time compared to that required by timing constraints to fulfil the functions of the design components. Positive setup slack ensures data input is stable before being captured at the clock event. Positive hold slack means that, after the clock event, the data input remains stable long enough to propagate through the operation. Positive pulse width slack, on the other hand, means the clock pulse itself is long enough to allow reliable capture of the data.

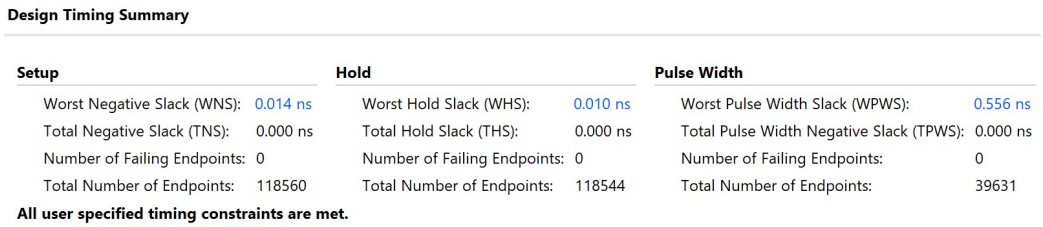


Figure 5.7: The timing results summary of the implemented design, showing that all timing requirements were met, reflected in the positive slack values.

Iterative improvements were made to the design, until all negative slack issues were resolved. These were primarily resolved through the addition of pipeline stages, introducing additional registers to break up long evaluation paths. In the board design, this can be done through the addition of shift register slices and data FIFO blocks. In the verilog HDL code, this involves the declaration of additional intermediate registers through which the data is passed after a complex operation. Correct clock domains must be used for each module, and the use the same clock source for multiple frequencies, through the use of an MMCM block, helps keep them in sync. A reduction in clock frequency also helps improve timing slack. Reducing the depth and number of ILA cores, restricting to data-only cores, as well as inserting pipeline stages before the ILA, also help to improve timing for the design. Finally, a number of options can be adjusted in the Vivado synthesis and implementation processes, such as reducing hierarchy flattening, enabling retiming so that registers can move across combinatorial logic, and specifically targeting timing improvements using directives such as *ExtraTimingOpt* and *AggressiveExplore*.

Vivado also provides an analysis of power and resource utilisation from the design, split out across the constituent components including digital signal processors (DSPs), clocking, block RAM and configurable logic blocks (CLBs). Optimising these two measures was not the focus of this design. Although both were within the working constraints of the RFSoC development board for the proof of concept, they must be significantly streamlined in order to implement the large number of beams required in PUEO.

Utilisations of power and hardware resources can be improved in future work in a number of ways. This includes adopting the beamforming optimisations from the previous chapter, retaining only 5 bits, while reducing the size of the window and increase the steps between windows. The Vivado HLS tool can also be used to optimise the design of user logic currently implemented directly using HDL code.

Finally, the bitstream, a binary configuration file for the FPGA, can be generated and exported as part of a .hdf file. This is a renamed standard .zip file, containing both the bitstream as well as hardware and software handoff files for further development. This includes *psu_init.c*, which initialises the hardware during the boot processes. Also part of the export is the board support package (BSP) that provide software interfaces for the hardware, which will be used for software development with the PS.

5.1.4 Clocking

A variety of clocks play key roles in the operation of RFSoC-based coherent beamforming. This includes communication between the processing system and the programmable logic, high speed data processing in the PL fabric, and synchronised gigahertz sampling between digital converters. All the clock sources were connected to required modules using the Vivado GUI, but this can be done equivalently using HDL code.

Figure 5.8 illustrates the relationship of these clocks relative to the modules from the block design of figure 5.3. Note that the PL clock is connected to all module except the processing system, but these are omitted for clarity.

A series of specific configurations of these clocks was found to be necessary for the operation of the hardware. User and products guides are provided by the manufacturer, including [8,9,63] as well as the ZCU111 Evaluation Board User Guide UG1271 [7] and the ZCU111 System Controller GUI Tutorial [64]. However, limited information is provided in these documentations for the programming of the LMK04208 and LMX2594 clock sources. Additional

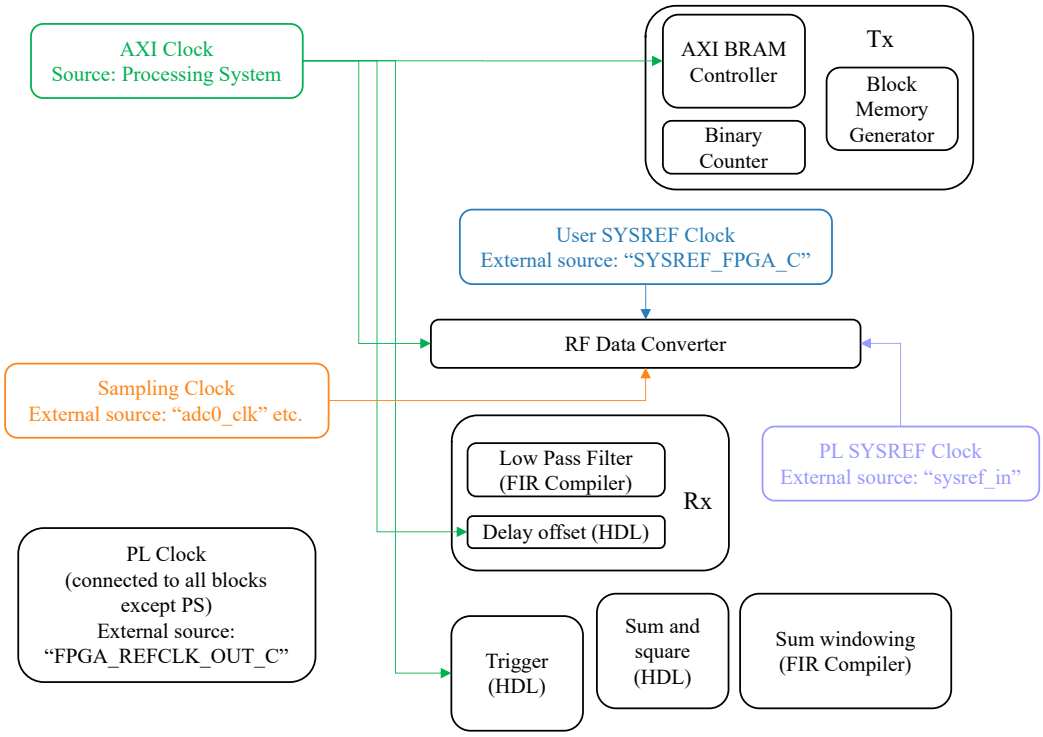


Figure 5.8: Schematic showing the clocks connected to the main functional block modules of the board design.

details of the clock configuration was therefore included in this thesis as a reference.

The sources and configurations of following clocks in the proof of concept are provided in Appendix B. Note that some of these have multiple names and are not always easily distinguished from each other:

PL AXI Clock

Enables communication between software and hardware using the AXI4 interface. Also referred to as AXI4 lite clock, or CPU clock.

Sampling Clock

Provides a stable clocking source for the analogue-to-digital, and digital-to-analogue converters. Also referred to as reference clock.

PL AXI Stream Clock

Drives the rate of the FPGA fabric data path, responsible for digital signal processing including coherent beamforming and digital filtering. Also known as sample clock, PL ref clock, fabric clock or FPGA REFCLK.

User SYSREF and Analogue SYSREF Clocks

Responsible for multi-tile synchronisation, correcting for sources of latency uncertainty between ADC (and DAC) converters. The source of user SYSREF is also referred to as PL SYSREF or SYSREF FPGA, and the analogue SYSREF is often referred to as simply SYSREF.

5.2 Timing Measurements

In order to achieve effective coherent beamforming, the sampling jitter should be minimised between different analogue-to-digital converters, so that there is a consistent relationship over time between converter timings. If the relative timings are to vary by amounts similar to the time between samples, the coherent sums enabled by carefully adjusted offsets would begin to lose their coherence. The ADCs on the RFSoC are arranged over 4 independent tiles, providing multiple source of phase variations.

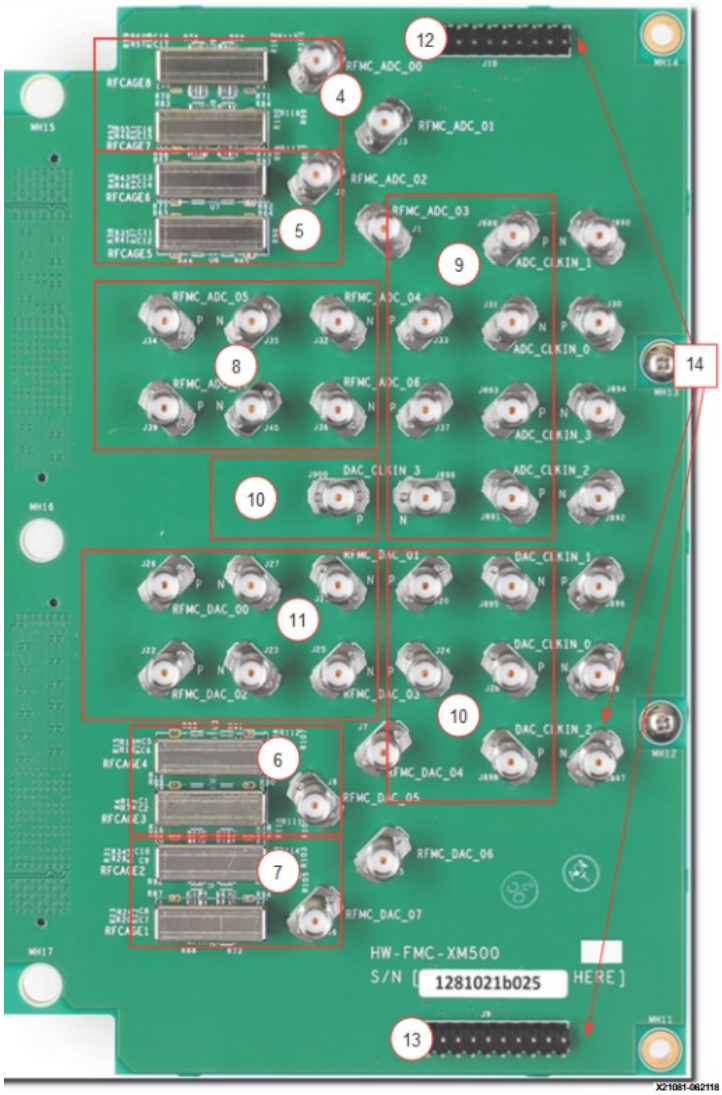


Figure 5.9: The XM500 balun board that forms part of the ZCU111 evaluation board, providing SMA inputs to the RFSoC. ADCs 00 to 03 were used to test phase jitter with externally generated signal.

Figure 5.9 shows the XM500 balun board of the RFSoC, providing SMA pins for signal input into the ADCs. While the RFSoC accepts differential signal inputs through a RFMC connections, this cannot be easily connected to by the user. Instead, the XM500 balun board provides conversion from these to standards SMA connections. Single-ended interfaces are provided for the first four ADCs shown at the top of the board, labelled *RFMC_ADC_00* to *RFMC_ADC_03*. An external signal generator was used to provide a single frequency sinusoidal signal into the 4 different ADCs through the use of a splitter. Since this source is single ended, the four single ended inputs were used, with ADC 00 and ADC 01 on one tile, ADC 02 and ADC 03 on another.

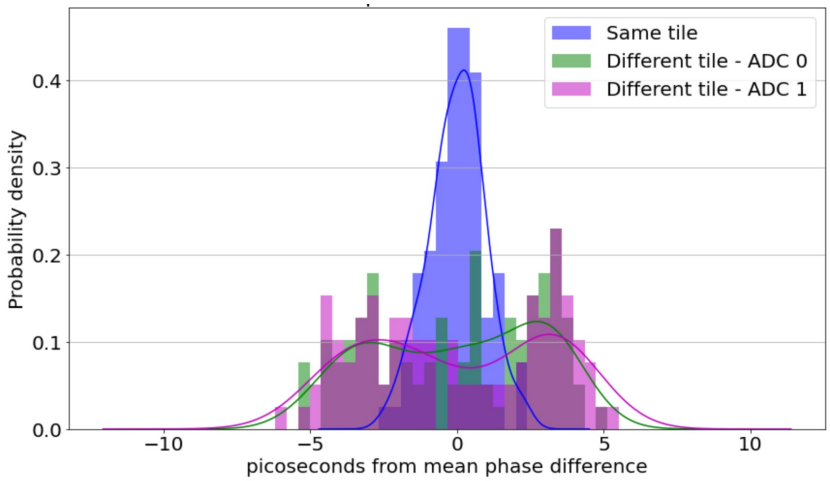
5.2.1 Measured Phase Variations

Multi-tile synchronisation was configured using the clocking framework discussed in this section and in Appendix B. The goal is to achieve a consistent timing relationship across converters, both on the same as well as different tiles. Using ADC 00 as a reference, ADC 01 is used to evaluate phase jitter for a converter on the same tile, and ADCs 02 and 03 (which are also often labelled ADC 0 and 1 on tile 1) are used to evaluate phase jitter for converters on different tiles.

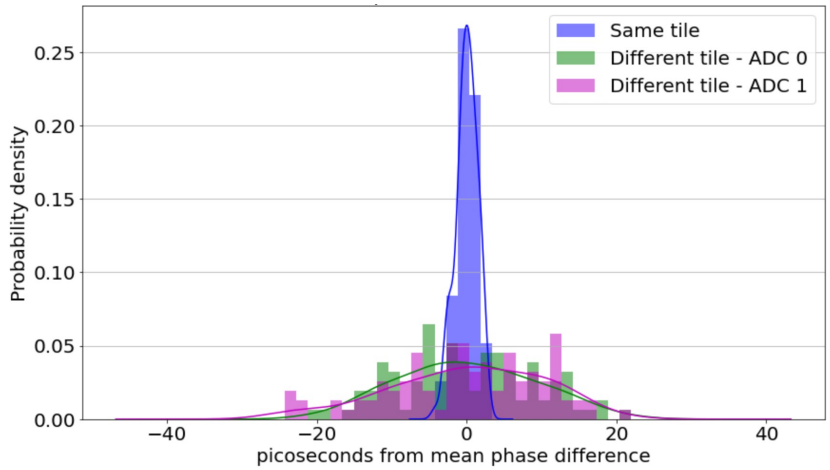
The measurements are made over a 1 hour period. Each phase measurement is made by fitting a sinusoid to 16 consecutive samples, with free variables for the phase, frequency, peak-to-peak amplitude and an overall DC bias. Cabling and wiring introduce fixed timing differences between ADCs, which is calibrate for by evaluating variations from the mean phase difference between a given pair of ADCs. Since the RFSoCs may need to be reconfigured during flight, the board was also powered-off and reinitialised several times during the data collection.

Figure 5.10 plots these phase jitter results, with both low (300 MHz) and high (1 GHz) frequency signals. At 1 GHz the phase variation of ADCs on the same tile were found to have a standard deviation of 0.9 ps, and 3.0 ps across different tiles. At 300 MHz, same tile variation remains similar at 1.4 ps, and the cross tile variation increases to 9.5 ps. However, at least some of this increased jitter is likely to be due to different baluns used for ADC 02 and 03 (U7 Anaren, designed for 1 - 4 GHz operation) and ADC 00 and 01 (U8 Mini-Circuits, designed for 0 - 1 GHz operation), with loss in performance at 300 MHz for ADC 02 and 03.

The great latencies across tiles is expected, due to a greater accumulation



(a) With external signal at 1 GHz, standard deviation of phase jitter for ADCs on the same tile is ~ 0.9 ps, increasing to ~ 3.0 ps between tiles.



(b) With external signal at 300 MHz, standard deviation of phase jitter for ADCs on the same tile is ~ 1.4 ps, increasing to ~ 9.5 ps between tiles.

Figure 5.10: Histogram and kernel density estimate of phase difference variation, showing jitter across tiles much smaller than the 340 ps between samples. The larger apparent cross-tile variation at 300 MHz is likely driven by the greater attenuation due to the reduced low frequency response of the baluns of one of the tiles.

of phase jitter sources, including clock distribution paths, temperature variations and positions relative to noises such as the power supply. The possibly bimodal distribution of phase jitter for the 1 GHz signal may correspond to the resolution of the underlying measurement and adjustment of latencies.

5.2.2 Implication of Deterministic Latency

Although the measured jitter is greater between converters across different tiles, the variations are much smaller than 340ps, which is the time between consecutive samples at 2.95 GHz. Since the frequency content of the received Askaryan signal is between 300 MHz - 1.2 GHz, features of the signal do no vary when sampled at these subsample jitters. This provides confidence in the RFSoC's ability to perform coherent beamforming at the required sample rate.

Future work can explore the use of additional external baluns to verify that the remaining four ADCs offer the same degree of cross-converter alignment. With access to multiple RFSoC boards, multi-device synchronisation can also be tested to measure inter-RFSoC phase jitter.

5.3 Trigger Testing

5.3.1 Software

The AMD Software Development Kit (SDK) can be used to write C++ software to run on the Processing System. The proof of concept trigger design uses this capability to initialise the RFSoC, which is essential for the synchronisation achieved in the previous section. Details for this initialisation are provided in Appendix C.

A few other SDK functions are highlighted here that enable dynamic trigger testing. *init_platform* initialises aspects of the hardware platform, including caches and UART. When run outside of the SDK, *init_platform* also initialises subsystems within the PS including the CPU and memory controllers, the DDR memory controller and peripherals. When running the application inside the SDK, these initialisations are done automatically.

```
1     init_platform();
```

BRAM and AXI4 lite register addresses can be identified from names found in *xparameters.h*.

```
/* Definitions for peripheral TRIGGER_CONTROL_0 */
#define XPAR_TRIGGER_CONTROL_0_DEVICE_ID 0
#define XPAR_TRIGGER_CONTROL_0_S00_AXI_BASEADDR 0xA0001000
#define XPAR_TRIGGER_CONTROL_0_S00_AXI_HIGHADDR 0xA0001FFF
```

These addresses can be written to using length-specific functions such as *Xil_Out16(address, value)*, and read from with functions such as *Xil_In16(address)*.

```
uint32_t bram_tx_base_addr_0 =
 ↪  XPAR_TX_TX0_AXI_BRAM_CTRL_0_S_AXI_BASEADDR;
Xil_In16(bram_tx_base_addr_0)

uint32_t bram_thresh_0 = XPAR_TRIGGER_CONTROL_0_S00_AXI_BASEADDR;
Xil_Out32(bram_thresh_0, threshold);
```

These functions are used to assign generated signal data to the BRAMs, which are then transmitted by the DACs and received by the ADCs. Delay offsets and trigger thresholds are controlled by writing to the corresponding memory addresses. The following code manages the trigger processes through communication with the ILA managed by the host system

```
//slv_reg2 stores runID
Xil_Out32(XPAR_TRIGGER_CONTROL_0_S00_AXI_BASEADDR+8,
 ↪  strtol(unique_ID, &ptr2, 10));
sleep(1);
//slv_reg1 stores recordSwitch, to turn on trigger
Xil_Out16(XPAR_TRIGGER_CONTROL_0_S00_AXI_BASEADDR+4, 1);
//turn off trigger
Xil_Out16(XPAR_TRIGGER_CONTROL_0_S00_AXI_BASEADDR+4, 0);
sleep(9);
xil_printf("Completed run %d \n\r", run_count);
cleanup_platform();
```

sleep is used to allow time for a new signal to propagate through the coherent trigger and recorded by the ILA. Note that *xil_printf* allows message to be printed to the SDK terminal.

5.3.2 Integrated Logic Analyser

The final part of the automated testing environment outlined in figure 5.1 is the Integrated Logic Analyser (ILA) running on the FPGA but managed by Vivado running on the host computer. The ILA must be configured during synthesis, using the *Set Up Debug* wizard shown in figure 5.11.

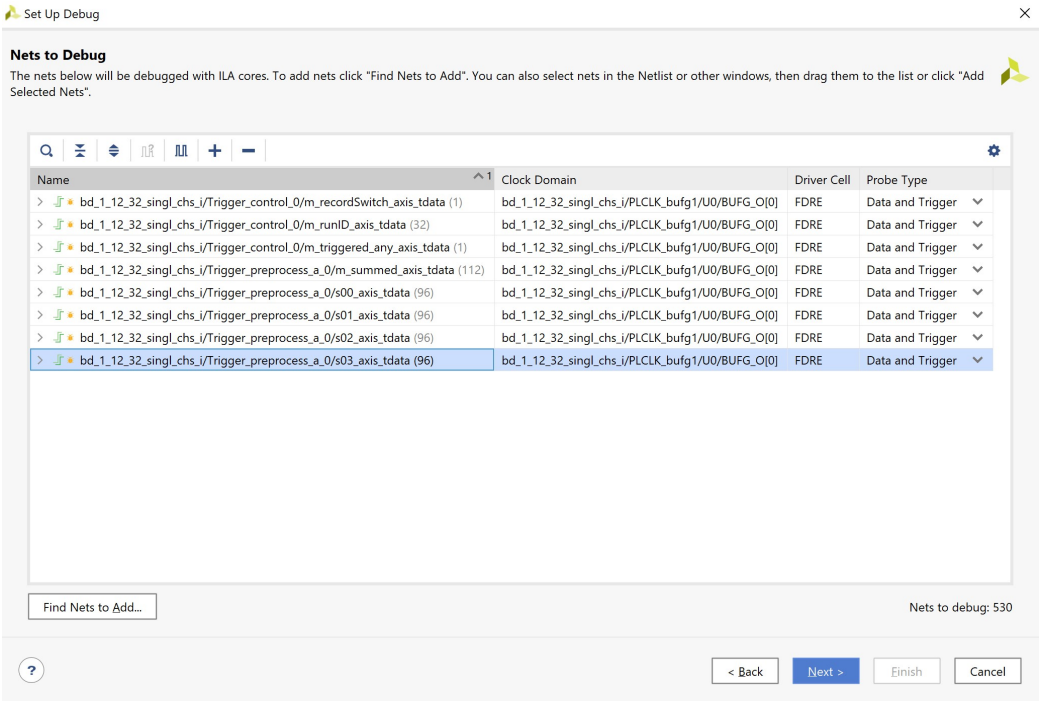


Figure 5.11: *Set Up Debug* under synthesis in the Vivado workflow allows ILA cores to be connected to nets that connect different components of the FPGA fabric.

Groups of connections whose signals will be probed can be selected, as well as the corresponding clock source. Note that the probes can also be limited to subsets of the full connection, or individual wires. Probes can be those that will trigger at given values, or have their data recorded when the

triggers are activated, or both. Additional options also allow the selection of the ILA data depth, determining how much data can be stored and analysed, and potential ILA pipe stages which help with timing. The binary value in *m_recordSwitch_axis_tdata*, activated via software running on SDK on the host machine, is used as the ILA trigger, resulting in the capturing of the data in the other ILA cores.

Finally, a tcl script is used to create a looping automated control system for the ILA, for batch analysis of a large number of signals. In each iteration, the ILA is re-armed, and upon the trigger and new provided provided by a corresponding C++ loop in the CPU, waits for the data buffers to fill up before writing data from the ILA cores to a csv file.

```
1    for {set i 0} {$i <100} {incr i} {
2        run_hw_ila [get_hw_ilas -of_objects [get_hw_devices xczu28dr_0]
         ↪  -filter {CELL_NAME=~"u_ila_0"}]
3        wait_on_hw_ila [get_hw_ilas -of_objects [get_hw_devices
         ↪  xczu28dr_0] -filter {CELL_NAME=~"u_ila_0"}]
4        upload_hw_ila_data [get_hw_ilas -of_objects [get_hw_devices
         ↪  xczu28dr_0] -filter {CELL_NAME=~"u_ila_0"}]
5        after 100
6        write_hw_ila_data -csv_file -force C:/path_to_data/data_$i.csv
         ↪  hw_ila_data_1
7        puts "Completed iteration $i"
8    }
```

For the timing measurements in section 5.2 using an external signal with a single ended connection, the ADC channels with single ended balun inputs were used - *RFMC_ADC_00* through to *RFMC_ADC_03*. For testing the proof of concept trigger, where the DAC is used to supply the generated signal data, the four differential channels are used instead - *RFMC_ADC_04* through to *RFMC_ADC_07*. Using the same external balun on each differential channel, this allowed similar responses to be applied in the four channels to enable a coherent sum. In contrast, the single channels have built in baluns, with two of them designed for low frequency operation and another two designed for high frequency operation, resulting in large differences for the broad-frequency signal being tested. Thus only the differential channels were used for the proof of concept beamforming trigger.

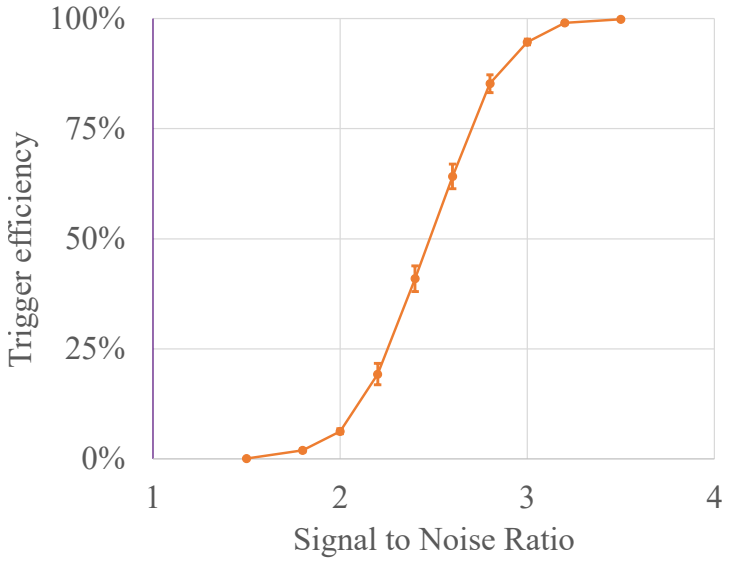


Figure 5.12: The trigger efficiency plot for a 4-channel proof of concept trigger developed on the RFSoC. $SNR_{50\%}$, the signal-to-noise ratio at 50% trigger efficiency, is measured to be 2.5, which is consistent with the results from the software simulation in figure 4.6.

5.3.3 Trigger efficiency

Figure 5.12 shows the trigger efficiency of the 4-antenna coherent beamforming trigger when running with signals simulated as per the software simulation. $SNR_{50\%}$ is evaluated to be 2.5, which is compatible with the result from the software simulation in figure 4.6.

Although this is only half of the number of channels in an L1 sector, this is a useful result showing that the hardware trigger implemented can achieve the benefits expected from software simulation for coherent beamforming. This is a verification of the beamforming performance, which ultimately carries through to the expected PUEO sensitivity of figure 2.8.

Chapter 6

Conclusions and Outlook

Ultrahigh energy neutrinos provide a unique signature through which to look for BSM physics, due to a combination of their large energies, potential to accumulate small effects over long distances, as well as providing the direction of their origin.

ANITA and PUEO are high altitude balloon payloads designed to detect ultrahigh energy neutrinos. They are sensitive to the coherent Askaryan signals generated from a neutrino interaction in ice, aided by the long interaction length of radio. In addition, their high altitude allow a large effective volume to be surveyed for the low flux of UHE neutrinos. This comes at the cost of smaller signal strengths that have attenuated over longer distances. PUEO builds on this by using a coherent beamforming trigger that provides an order of magnitude improvement in the single-antenna signal-to-noise ratio sensitivity. This trigger is made possible by the use of high performance RFSoC hardware.

6.1 Impact Summary

This work laid out a road map of key building blocks in implementing a beamforming trigger on the RFSoC platform. This included the first software simulation of the PUEO trigger hierarchy, confirming the benefit to single antenna SNR sensitivity from beamforming. An improved understanding has also been achieved on the sensitivity impact of a low pass filter.

The trigger simulation and optimisation analyses carried out in this work are important for demonstrating that gains in PUEO's sensitivity can be

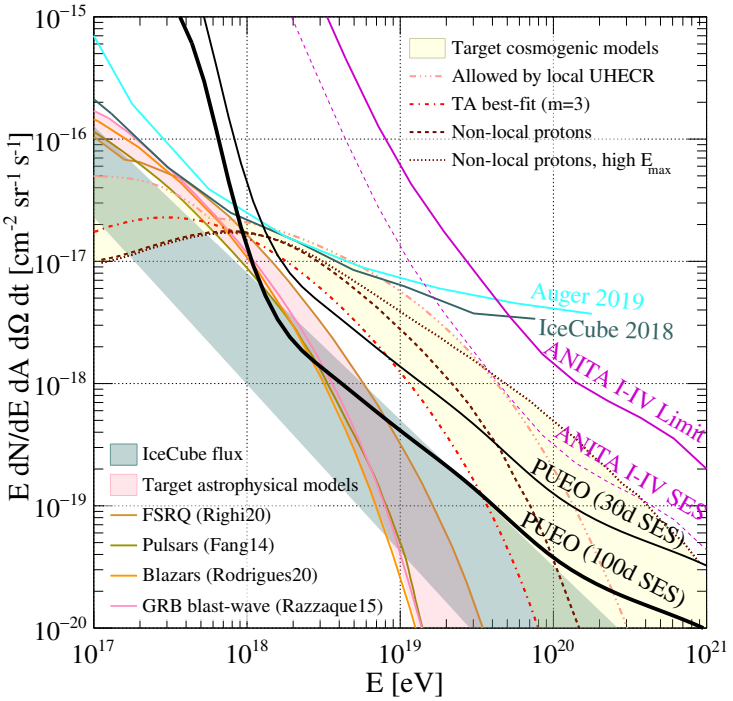


Figure 6.1: PUEO's sensitivity to diffuse UHE neutrino fluxes [5]. Trigger optimisations evaluated in this work correspond roughly to an 11% shift to the left along the x-axis for the two PUEO sensitivity lines, representing 30 and 100 day flights.

achieved through trigger design choices, and for a first quantification these improvements.

Compared to design choices from the PUEO whitepaper and previous studies, a ~ 11% improvement has been demonstrated in the SNR where the trigger is 50% efficient. This can be achieved by retaining a higher number of bits, additional beam density and a smaller window size. Simplistically, the SNR scales linearly with the neutrino energy, as in equation 3.3. The identified improvement therefore represents an approximately 11% shift left along the x-axis in PUEO's diffuse neutrino sensitivity estimates in figure 6.1. For neutrinos of energy $10^{18}eV$, this improvement is similar to that gained from a flight of 100 days compared to 30 days.

The analysis also identified orders of magnitude reductions in hardware resources required, without compromising trigger performance, by using a

5 bit digitisation scheme and L2 beam separations of 2° in elevation and 5° in azimuth. The trigger software simulations has been integrated into PUEOSim, the PUEO simulation framework that extends the simulation to the full payload and beginning from the initial neutrino interaction.

Work was also carried on the hardware design of a proof-of-concept trigger using 4 channels on a single RFSoC board. This included the use of the Vivado software for the configuration of the board as well as designing hardware logic for the trigger algorithm. A test bench platform was developed for the trigger hardware, using C++ code running on the onboard CPU, together with Integrated Logic Analysers that capture data on the FPGA and saved for later analysis. Trigger performance consistent with the software simulation was achieved. Crucially, the system of clocks was configured to enable a high level of synchronisation across channels, with the standard deviation of mean phase differences no greater than 10 picoseconds.

Outside of PUEO, the prototype RFSoC hardware design outlined in this work was used as the basis for the radio trigger detector of the Radio Echo Telescope for Cosmic Rays (RET-CR) [62], which was deployed in May 2023. Using 8 transmitting and 8 receiving channels, the cascade of cosmic-ray initiated air shower is detected through their reflection of the transmitted radio. By testing the radio echo technique, RET-CR is a step toward a next generation experiment to detect ultrahigh energy neutrinos using the same technology.

6.2 Future Work and Outlook

Using the PUEOSim framework, within which the software trigger simulation developed here has been integrated, all 96 antennas of PUEO's main instrument can be simulated rather than a single L2 sector of 16 antennas as has been done here. This will enable more representative simulations and trigger optimisations to be carried out, where neutrino fluxes are derived from models of UHE neutrino production, with in-ice interactions and event geometries based on models of the Antarctic ice sheet.

The payload simulation would also be able to incorporate flight paths based on previous ANITA flights, together with continuous wave noise also simulated based on past data. Simulations of digital notch filters should be included to study their effectiveness in maintaining trigger sensitivity in the presence of continuous wave noise. The measured PUEO antenna and

signal chain responses can be used to reflect the hardware as it is acquired and tested. Indeed, the current simulation has ignored the distortion of signals when received at angles that are off-cone to the antennas, which would affect coherent summation, and this is being incorporated in iterations of the simulation.

Through the inclusion of these improvements, the full simulation enables a calculation of the effective area of a PUEO flight, including an updated limit that can be placed on the diffuse flux if there is no detection. This could be further extended to a study of the analysis pipeline, to understand the impact of PUEO's improvements on analysis efficiency given the presence of UHE neutrino signal. Studies are also underway to evaluate the ability to differentiate neutrino flavours through the identification of multiple closely located vertices.

In terms of hardware, work has been ongoing for the framework enabling clock synchronisation and communication across multiple RFSoC boards. In addition, data buffers will be included, as well as an overall L3 trigger that must be able to deal with potential trigger build-up scenarios. At the same time, much more efficient implementations of the trigger algorithms are being developed, directly in HDL code, to make fuller utilisation of the resources available. Efficient notch trigger implementations are also being developed, which will likely place the greatest constraints on available computational resource.

These efforts form parts of PUEO's broader development, delivery, testing and review process. The assembly and integration of the payload is scheduled for early 2025, with the PUEO launch window expected for December 2025.

PUEO is expected to have world-leading sensitivity to ultrahigh-energy neutrinos above 1 EeV, with either the first detection of ultrahigh-energy neutrinos, or set the best limits for ultrahigh-energy neutrino flux. Crucially, the predicted flux of several cosmogenic UHE neutrino models are expected to be accessible by PUEO at energies between 1 and 10 EeV. Excitingly, even in the case of non-detection, PUEO's limits on the UHE neutrino flux would be evidence of something unexpected either in the nature of highly-energetic cosmic rays or in fundamental physics at these energies.

On the other hand, the first detection of UHE neutrinos would provide a new messenger for understanding the highest-energy accelerators in the universe, as well as a source of particles that interact at energies exceeding that of terrestrial particle accelerators. PUEO is joined by a number of other experiments in this endeavour, both operational and in development, that use

a variety of detection techniques. This includes optical detectors in ice or water, which have in general lower detector volume but higher resolution compared to radio experiments. These experiments include IceCube [65], IceCube-Gen2 [66], ANTARES [67], KM3NeT [68] and P-ONE [69]. As well as ANITA and PUEO, ARA [70], ARIANNA [71], and RNO-G [72] also use radio detection of Askaryan signals, but with detectors in the ice instead high altitude balloon. IceCube-Gen2 is also expected to incorporate in-ice radio detection. ν_τ regeneration and the lifetime of the tau enable detection of air-shower produced by UHE ν_τ specifically. The Pierre Auger Observatory [73] detects the Cerenkov light of air showers particles passing through a large array of surface water tanks. GRAND [74] and TAROGE [75] detect radio emission generated by the air showers, similarly to ANITA/PUEO, but using terrestrial detectors. The Cerenkov and fluorescence light radiated by the air-showers particles can also be detected, either from the ground with Trinity [76] and Ashra NTA [77], or from a satellite for POEMMA [78].

Appendix A

RFSoC Block Design Modules

An overview is provided here for the key components used in the proof of concept beamforming design. In particular, careful configuration of the clocks and interfaces of both the Processor System (CPU) and the RF Data Converter (ADC) is required for the high frequency hardware design.

A.1 Processing System

Using a new board design wizard, or from a blank design, the first module that is added is that of the Processing System, named *Zynq UltraScale+ MPSoC*. Figure A.1 shows some of the connections available for this block.

The following connections were found to be important. *pl_clk0* should be connected to the AXI clock inputs for all PL blocks that might require configuration or other interaction with the PS. Typically these inputs are named *s_axi_aclk* or *s00_axi_aclk*. AXI, or Advanced eXtensible Interface, is a protocol for interfacing between PL cores and PS, and AXI here refers to AXI4 or AXI4 lite connections. In particular, this clock should not be connected to the AXIS, or AXI Stream clock inputs, typically named *axis_aclk* or possibly only *aclk*. AXI Stream is used for the continuous delivery of data, rather than configuration interface with the CPU. The rate of the AXI interface clock can be chosen freely, and is here selected to be 100 MHz. This same clock is connected to the three clock inputs shown here: *maxihpm0_fpd_aclk*, *maxihpm1_fpd_aclk* and *saxihp0_fpd_aclk*.

The AXI interface channels, *M_AXI_HPM0_FPD* and *M_AXI_HPM1_FPD*, should be connected to the same IP blocks as above, using an AXI Intercon-

nect, shown to the right of figure A.2. Similarly, the associated reset for the AXI interface *pl_resetn0* should be connected to the IP clocks via the *Processor System Reset*, shown to the left of figure A.2.

A.2 RF Data Converter

Figure A.3 shows the board design module for the radio frequency data converters, which has perhaps the most involved set of interface connections within the design.

Firstly, the inputs *s_axi*, *s_axi_aclk* and *s_axi_aresetn* are connected to the Processing System, as previously outlined. Next, the inputs numbered from *s00_axis* are associated with the DAC, and can be connected to AXI stream outputs. Note that the first number is associated with the tile, and in each tile there are multiple converters, which is identified by the second number. *s0_axis_aclk* and *s0_axis_aresetn* are the clock and reset associated with the AXI Stream. Similarly, *m00_axis* is the AXI stream output of the first converter in the first ADC tile, and can be connected to the PL for additional processing. *m0_axis_aclk* and *m0_axis_aresetn* are the related clock and reset.

The voltage channels *vin0_01* to *vin3_23*, and *vout00* to *vout03* should

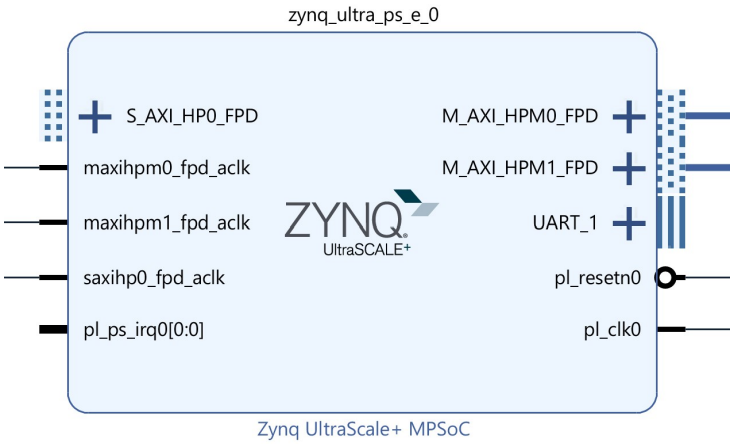


Figure A.1: The board design module for the PS, or Processing System, which is also known as the MPSoC (Multi-Processor System-on-Chip).

be connected to external interfaces using Vivado's automated prompt. The clocks named from *adc0_clk* and *dac0_clk* should also be connected to external clock interfaces using Vivado's automated prompt, and not the outputs of this block named from *clk_adc0* and *clk_dac0*.

The AXI stream clocks, named from *m0_axis_aclk* and *s0_axis_aclk* should be connected to the external interface named *FPGA_REFCLK_OUT_C*, with intermediate connections that are discussed later in section 5.1.4. A multiple (368.64 MHz) of the external interface (122.88 MHz) is used as the core clock within the FPGA PL fabric. By processing 8 samples in parallel, this enables the RF data converter to work at the desired gigahertz frequency of 2.94912 GHz.

sysref_in is connected directly to an external interface with the same name. *user_sysref_adc* is connected to an external interface named *SYSREF_FPGA_C*, with intermediate connections that are discussed in section 5.1.4. This is a low frequency reference clock (7.68 MHz) used to align the

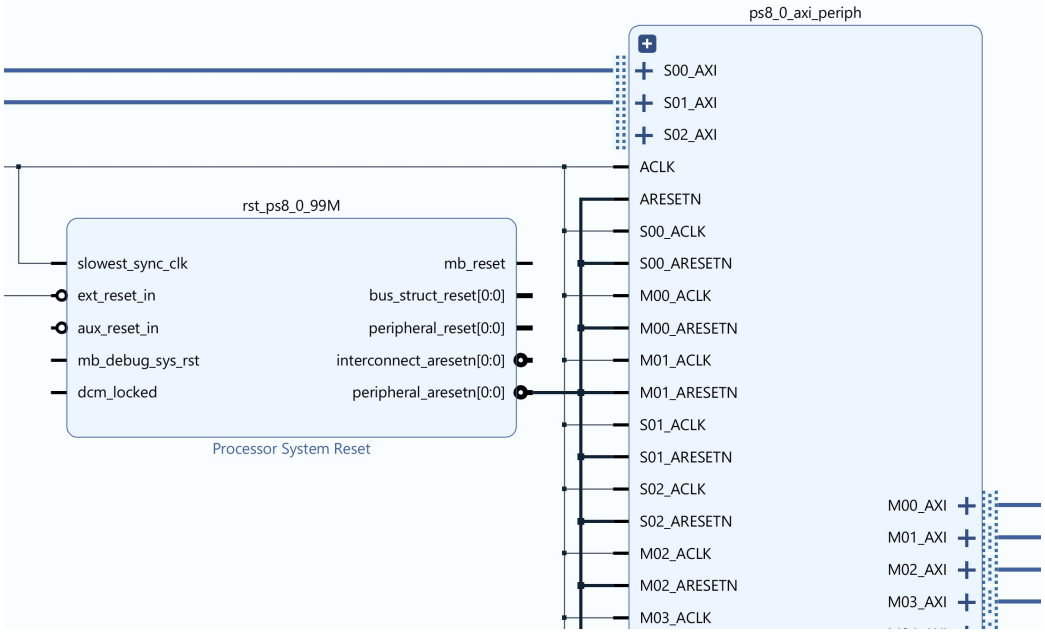


Figure A.2: The board design modules Processor System Reset (left) and AXI Interconnect. The AXI Interconnect is truncated for clarity. The inputs to the left are connected, from the top down, to *M_AXI_HPM0_FPD*, *M_AXI_HPM1_FPD*, *pl_clk0*, and *pl_resetn0*.

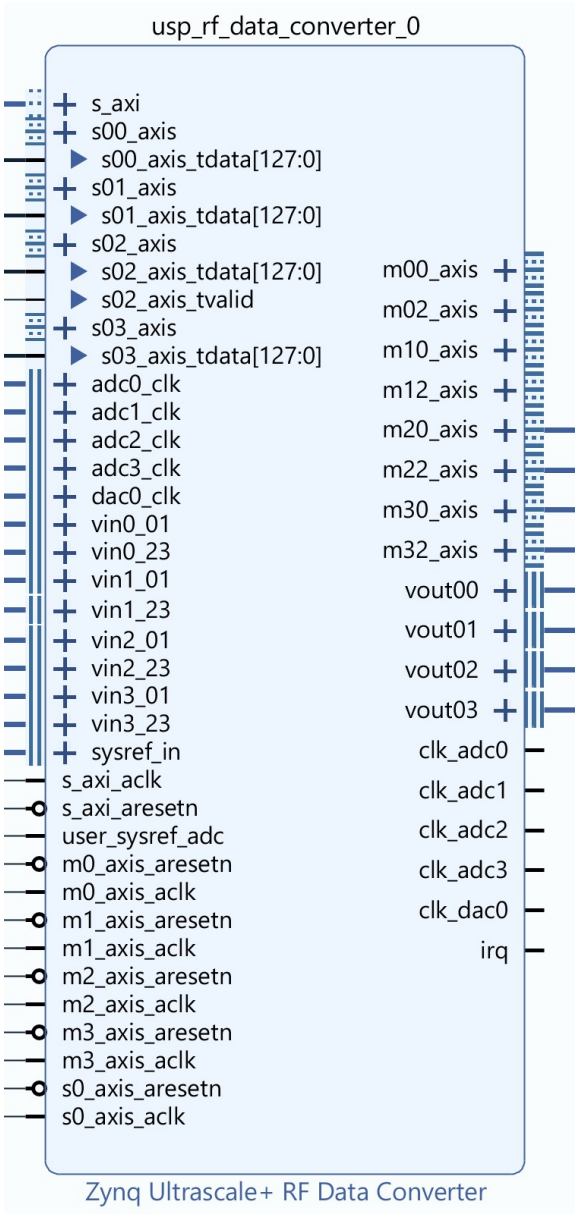


Figure A.3: The board design module *Zynq Ultrascale+ RF Data Converter*, associated with the radio frequency analogue-to-digital and digital-to-analogue converters.

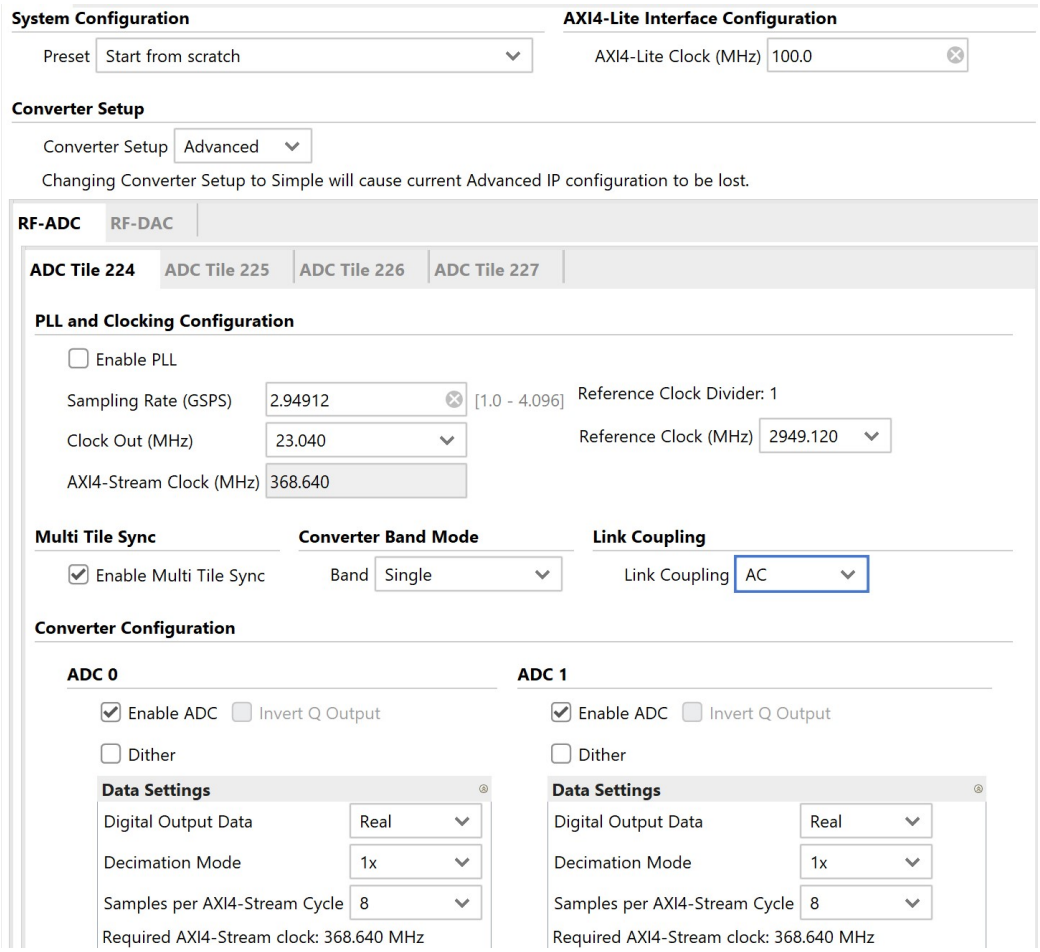


Figure A.4: Settings page for the board design module *Zynq Ultrascale+ RF Data Converter*.

data converters according to a simplified version of the JESD204B standard [79].

Additional configurations are required for the RF data converter, using the block customization page, shown in figure A.4. The AXI4 lite clock should be chosen to match that of the PS. As above, each AXI4 stream cycle is chosen to have 8 samples, which together with the sampling rate, determines the AXI stream clock rate. This lowers the PL clock rate to hundreds of MHz, which is necessary for timing closure of the FPGA.

The output data type should be chosen as real, and decimation selected as

1x. Multi tile sync should be enabled. The desired ADC and DAC channels should be enabled, with settings on different tabs of the page. Note that ADC 0 of ADC tile 224 always needs to be enabled. Since the DACs are used for data generation in this proof of concept, they are also enabled, with equivalent settings around frequencies, 8 samples per cycle, etc. Further down and not shown on this figure, mixer type should be chosen as 'bypassed'.

A.3 FIR Compiler

The FIR compiler core enables the configuration of finite impulse response filters, which has as output the sum of the current and a number of previous inputs, each multiplied by a coefficient. This is used for the implementation of low pass filters in the design, discussed in the trigger optimisation chapter, for improving trigger performance. Although not implemented in this proof of concept, the FIR compiler can also be used to configure tuneable notch FIR filters, which will be crucial for the mitigation of narrow band satellite interference. Note that infinite impulse response (IIR) filters can also be implemented for these tasks [80]. Although they are more complex to implement, especially with parallel samples in each clock cycle, they generally require fewer FPGA resources.

FIR compiler cores were also used for the summation across windows in the coherent sum. Figure A.5 shows the connections of the module, with only AXI stream input and output.

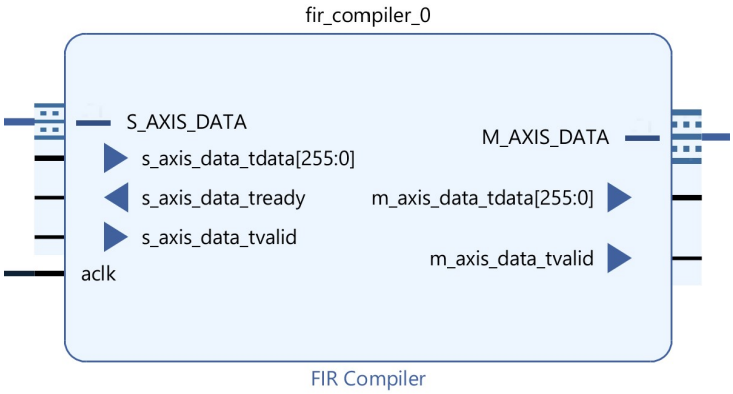


Figure A.5: Module connections for board design module *FIR Compiler*.

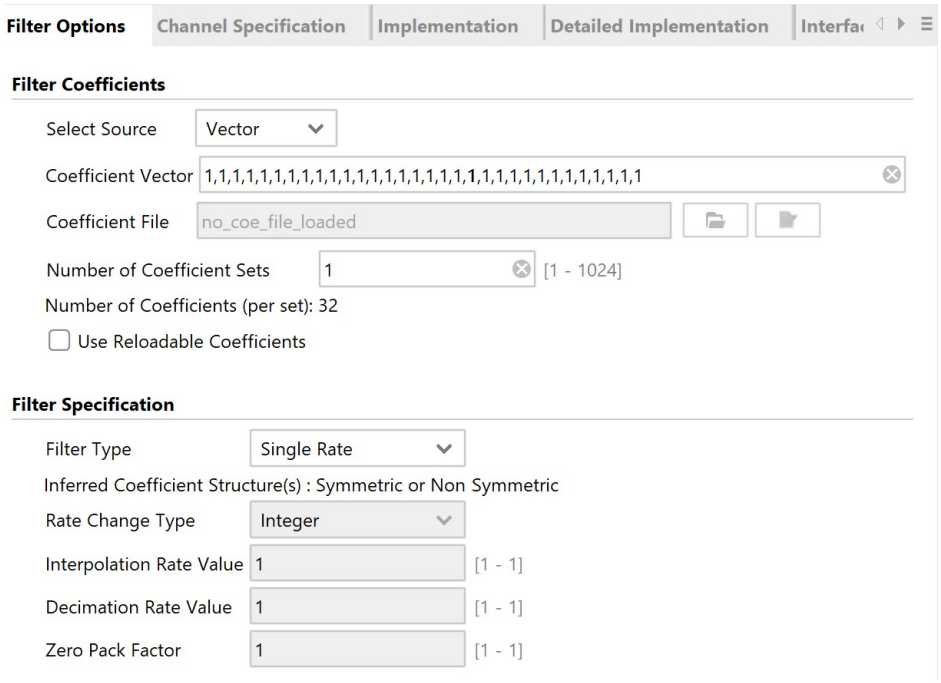


Figure A.6: Module customisation: filter options for board design module *FIR Compiler*.

A number of relevant options are in the block customisation. On the filter options tab, shown in figure A.6, the FIR filter coefficients can be entered, either directly or through a file. 32 coefficients are used in this filter, for a coherent sum of window size 32.

Note that there is no AXI connection for editing the coefficients. However, reloadable coefficients can be enabled which adds additional AXI stream for the coefficients to be edited after initialisation. 'Single Rate' should be chosen for the filter type, which does not carry out any decimation or interpolation.

In the channel specification settings, shown in figure A.7, the 2.94912 GHz sampling frequency is entered, as well as the PL clock frequency. Using these two, the configuration chooses the number of parallel inputs. The coefficients chosen on the previous page account for these parallel samples, so that in this example 8 of the coefficients are multiplied with the samples from the same clock cycle.

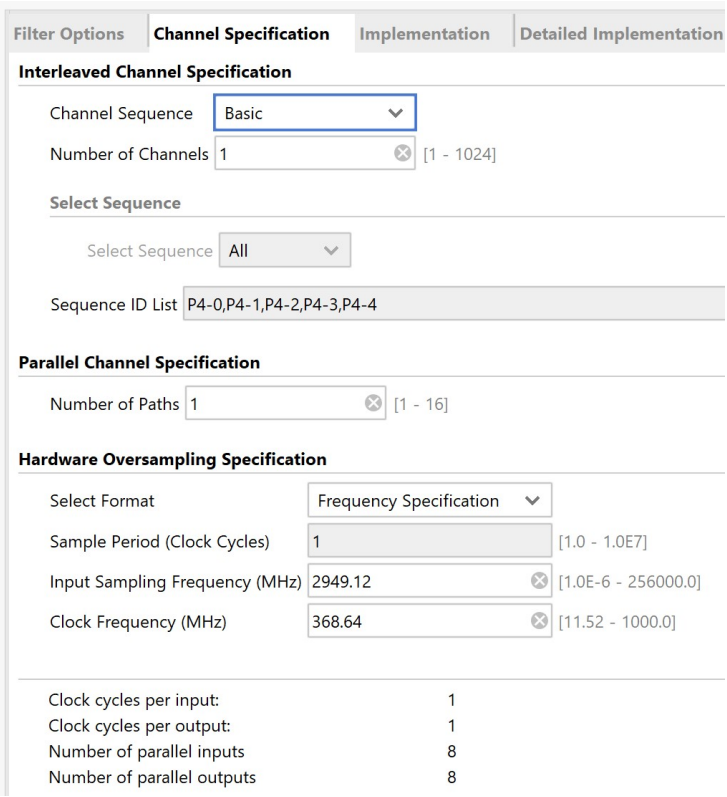


Figure A.7: Module customisation: channel specification for board design module *FIR Compiler*.

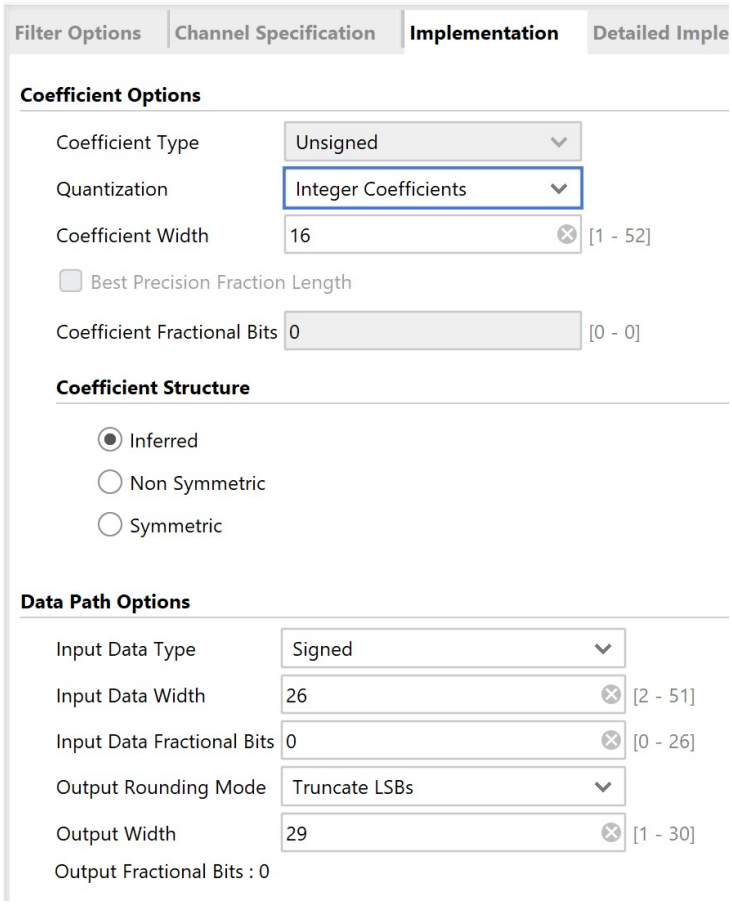


Figure A.8: Module customisation: implementation settings for board design module *FIR Compiler*.

Finally, on the implementation page in figure A.8, the input and output data widths are chosen, including choices of decimal points and rounding modes. Signed data type should be selected. The input data width will depend on the output choices of the previous block. Note that the total input/output width will be rounded up to integer multiples of 64 with padded bits. In this case, the 26 bit width would correspond to a total AXI stream width of 208, but the total width is 256 as can be seen in figure A.5.

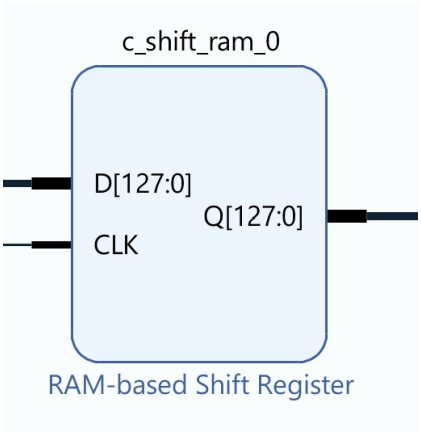


Figure A.9: *RAM-based Shift Register.*

A.4 Shift Register

The RAM-based shift register (figure A.9) has data input D and output Q connections of chosen width, which can be connected to the constituent tdata ports of AXI streams. Note that where required by the subsequent block, tready and tvalid, the remaining components of an AXI stream, must be provided for the signal to be correctly accepted. *CLK* should be connected to the AXI stream clock associated with the data input. Finally, a depth can be selected in the block customisation options, which is equal to the number of values held by and the latency of the shift register. In the proof of concept trigger, the role of the shift register is the provision of additional slack for meeting timing constraints.

A.5 AXI BRAM Controller and Generator

Block RAM provides data that is readily accessible to both the PS and PL. In the coherent trigger example, it is used to enable software running on the Processing System to provide simulated signal and noise to the DACs, so that it can then be coherently beamformed after being received at the ADCs.

The module block for the AXI BRAM controller is shown in figure A.10. The AXI input, clock and reset should be connected to the Processing System block, via AXI interconnect and the PS Reset blocks, using a Vivado automation wizard. Within the customisation settings, a width of 128 bits

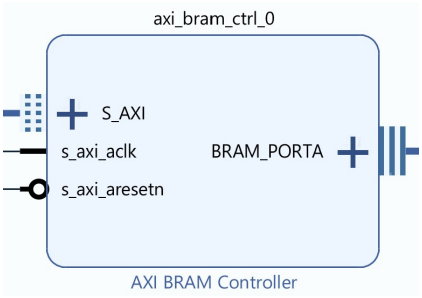


Figure A.10: *AXI BRAM Controller.*

should be chosen so that the block RAM can output 8 parallel samples of 16 bits to the DAC.

The *BRAM_PORTA* output is connected to the *BRAM_PORTA* input of the block memory generator module, shown in figure A.11. Whereas port A is connected as a group, port B has wires that are individually connected. *clkb* is connected to the AXI stream clock. *enb* is connected to a boolean

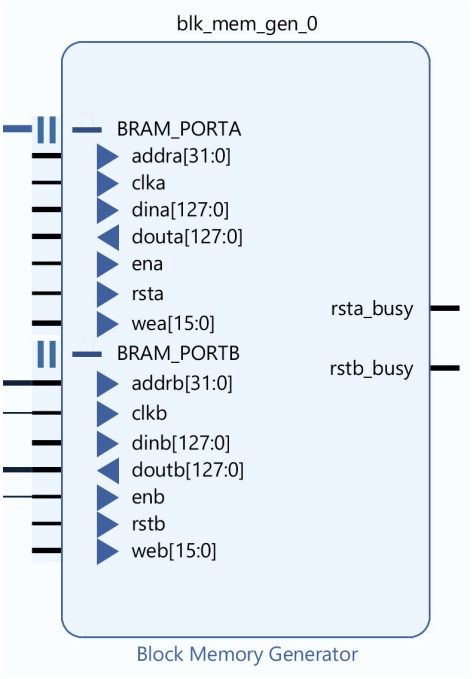


Figure A.11: *Block Memory Generator.*

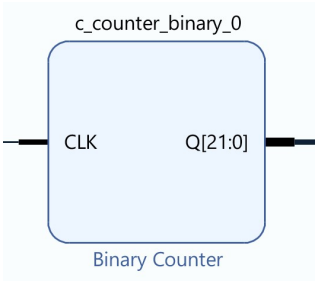


Figure A.12: *Binary Counter.*

control, which for simplicity in this case is a constant block with the value 1. *doutb* is connected to the shift register and subsequently the DAC input. Finally, *addrb* is the address of the BRAM which will be output at a given clock cycle from *doutb*. This is connected to a binary counter block, shown in figure A.12. Settings of the binary should be updated, with output width and increment value corresponding to the size of the block RAM, and the width of each stored value.

Appendix B

RFSoC Clock Configuration

B.1 PL AXI Clock

The PL AXI clock provides timing for communication between software and hardware using the AXI4 interface. The interface enables software running on the processing system CPU to write to memory registers on the PL. The clock is sourced from the *pl_clko* port of the processing system (figure A.1), and configured to be 100 MHz using the block customisation options in Vivado. This is found under Clock Configuration, Low Power Domain Clocks and PL Fabric Clocks, as can be seen in figure B.1.

B.2 Sampling Clock

The sampling clock determines the rate at which analogue signals are sampled and converted into digital signals, and vice versa, in the RF data converter. In the Vivado board design, the clock appears as external connections made available by a Vivado wizard prompt, displayed in figure B.2 . They should be connected to identically named clock inputs in the RF data converter, as previously shown in figure A.3.

The RF clocking structure can be seen in figure B.3, reproduced from the ZCU111 Board User Guide UG1271 [7]. The clock signals are sourced from the RFoutA/RFoutB outputs of the onboard LMX2594 RF synthesiser, and chosen in this design to be at 2.94912 GHz. It is configured using the RF data converter driver using software running on the PS, which is further detailed in section C.

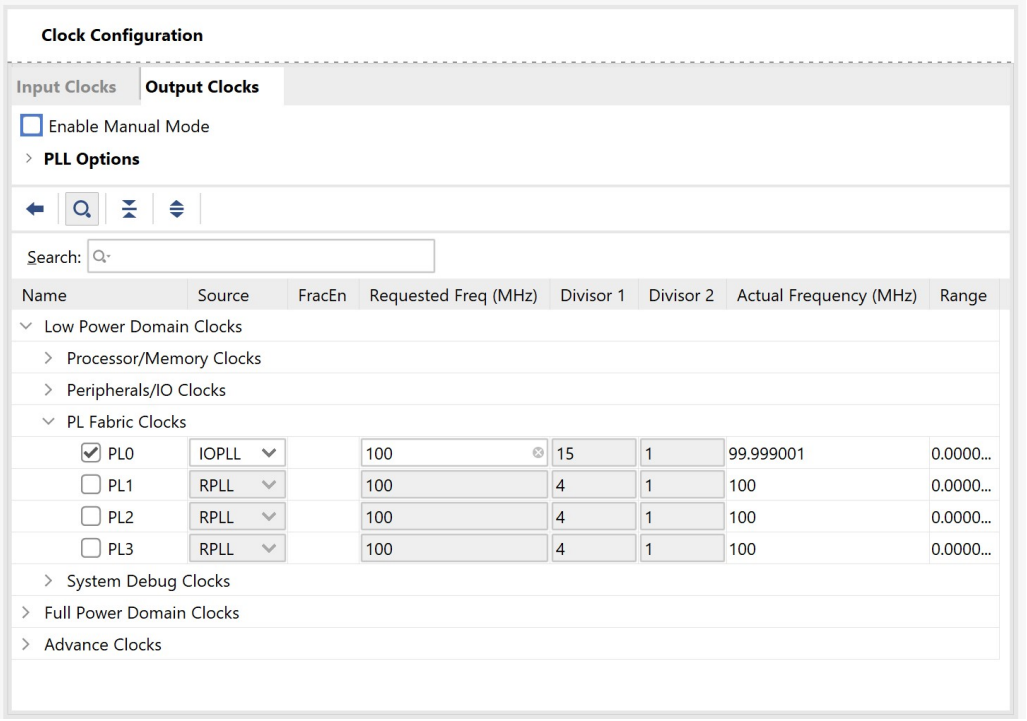


Figure B.1: Configuring the PL AXI clock, in the customisation page for MPSoC, the processing system.

There is a large degree of configurability in LMX2594, which can be accessed in the software driver by editing the *ClockingLmx* array in the *xrfdc_clk.c* configuration application provided with the driver. 123 hex register values are required in the configuration array, which can be interpreted and modified using the Texas Instruments Clocks and Synthesizers (TICS) Pro software [81].

TICS Pro's visual interpretation of the LMX2594 PLL settings used is shown in figure B.4. The desired output frequency at *RFoutA* and *RFoutB* can be achieved through the modification of dividers and multipliers. Additionally, the setting *VCO_PHASE_SYNC* needs to be enabled for the design to correctly enable multi tile synchronisation. The default register settings provided in *xrfdc_clk.c* have this toggled off. Note that some of the multipliers/dividers may need to be adjusted again when this setting is enabled, in order to maintain the same sampling clock frequency.

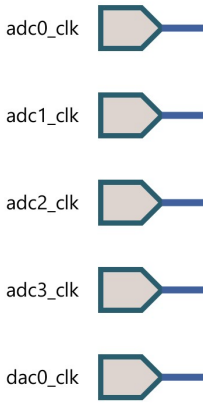


Figure B.2: Sampling clock made available in Vivado board design as external connections, on the left side of the board design diagram.

B.3 AXI Stream / PL Fabric Clock

The AXI stream clock is the primary driver of the continuous flow of data movement and processing within the FPGA, i.e. the programmable logic. In general, the PL clock can be sourced from a number of places, including from the processing system. However, the current design uses output from the onboard LMK04208 low-noise clock jitter cleaner, to enable the functionality of synchronisation features. Figure B.5 reproduces the illustration from the Zynq UltraScale+ RFSoC RF Data Converter Evaluation Tool UG1287 [63], with similar schematic references in [8] and [9]. PL REF CLK in this diagram can be identified with the *FPGA_REFCLK_OUT* port at the top of figure B.3.

Similarly to the LMX2594 that provided the sampling clock, LMK04208 is configured using the RF data converter driver using software running on the PS, with additional details in section C. The 32 hex register values required for the configuration array can once again be imported and modified using TICS Pro. This design takes the AXI stream clock from *CLKout2*, toggling the *powerdown* option for that output, and choosing *LVDS* as the clock type. An output frequency of 122.88 MHz is chosen, compared to the 368.64 MHz frequency for the fabric clock, facilitated by an MMCM clocking wizard block provided by Vivado. 8 samples are processed in parallel through the data path, enabling the 2.94912 GHz sampling frequency.

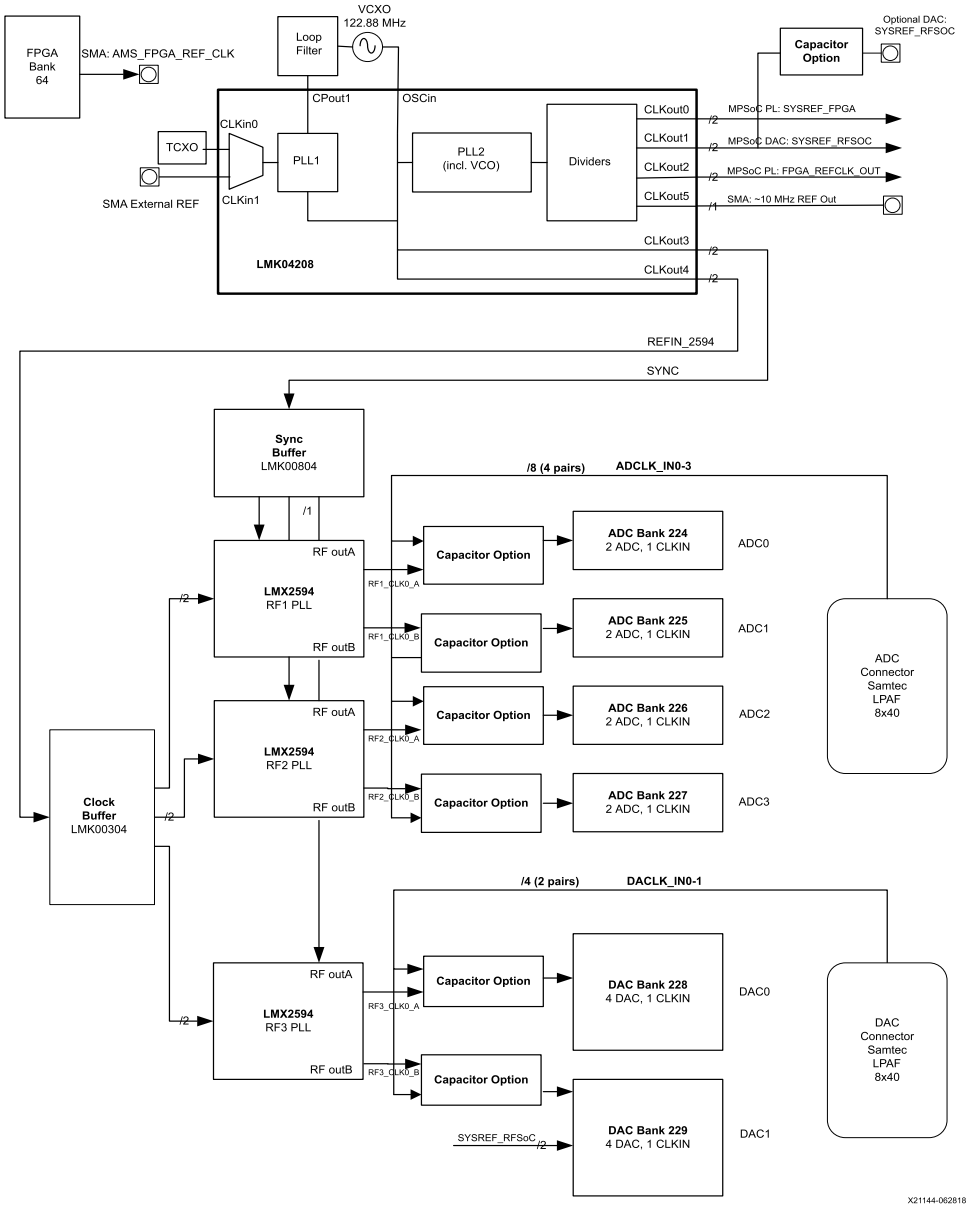


Figure B.3: RF clocking structure for the ADCs and DACs, reproduced from [7], showing LMX2594's provision of RF sampling clocking.

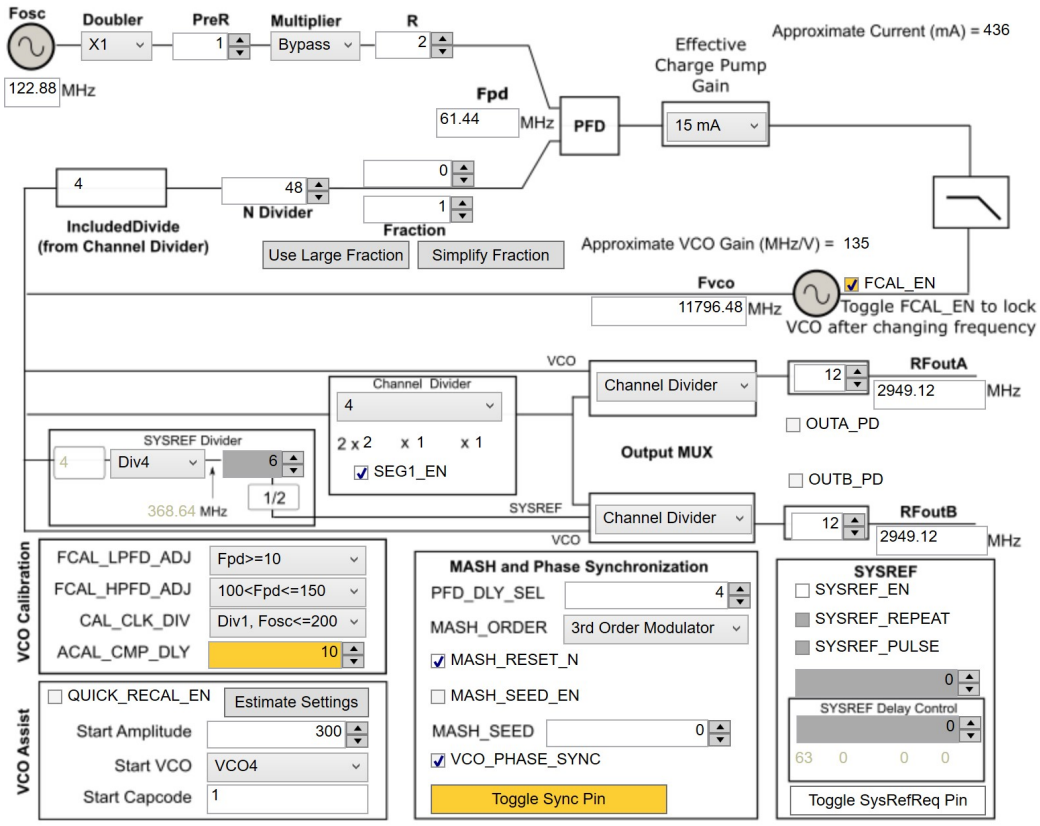


Figure B.4: The TICS Pro software GUI can be used to interpret and update the register values responsible for configurations of the onboard LMX2594 RF synthesiser.

Several additional steps are required for the implementation of this AXI stream clock in the Vivado board design. The .xdc constraints file of the board design wrapper must be edited explicitly to connect the pins of the LMK04208 to create the corresponding external port in the board design. This can be done by inserting the following lines:

```
set_property IOSTANDARD LVDS [get_ports {FPGA_REFCLK_OUT_C_clk_n[0]}]
set_property IOSTANDARD LVDS [get_ports {FPGA_REFCLK_OUT_C_clk_p[0]}]
set_property PACKAGE_PIN AL15 [get_ports {FPGA_REFCLK_OUT_C_clk_n[0]}]
set_property PACKAGE_PIN AL16 [get_ports {FPGA_REFCLK_OUT_C_clk_p[0]}]
```

The *FPGA_REFCLK_OUT_C* port will then appear as an external clocks that can be connected using the Vivado GUI, and identified with the PL AXI

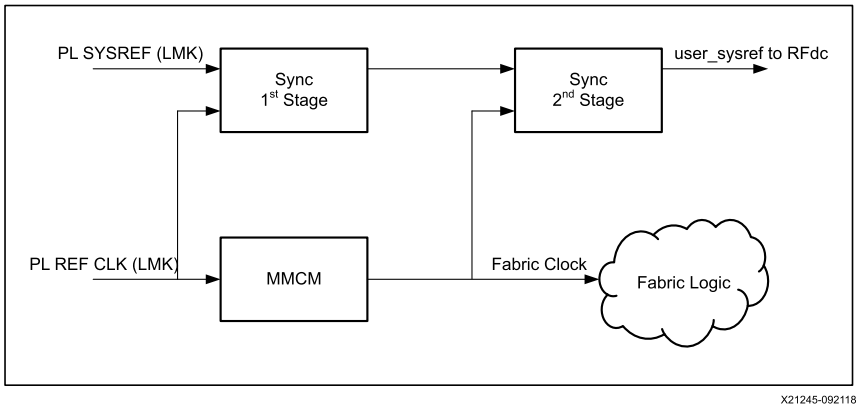


Figure B.5: The synchronisation of the PL AXI stream clock with the PL SYSREF is required for multi-tile (and multi-chip synchromisation). This schematic is from UG1287, showing LMK04208 as the source of the PL REF CLK.

stream clock. The pin locations can be found in the schematics (XTP508) from ZCU111 evaluation board documentation [82]. The relevant diagram is reproduced in figure B.7.

Furthermore, the synchronisation processes referenced in figure B.5 are implemented by following the more detailed schematic provided in the UltraScale Architecture PCB Design User Guide UG583 [8] and the UltraScale+ RFSoC RF Data Converter Product Guide PG269 [9]. These are reproduced in figure B.8. Flip flop modules are inserted both before and after the MMCM clocking wizard, following [8], with BUFG and IBUFG utility buffers added directly after the clock inputs, following [9].

B.4 User SYSREF and Analogue SYSREF Clocks

Between the independent RF data converters, there are a variety of sources of latency uncertainty, including clock divider phase, FIFO latencies, clock skew and data skew [9]. The AMD RFSoCs use a SYSREF synchronisation scheme across multiple data converter tiles and RFSoCs, and is based on the

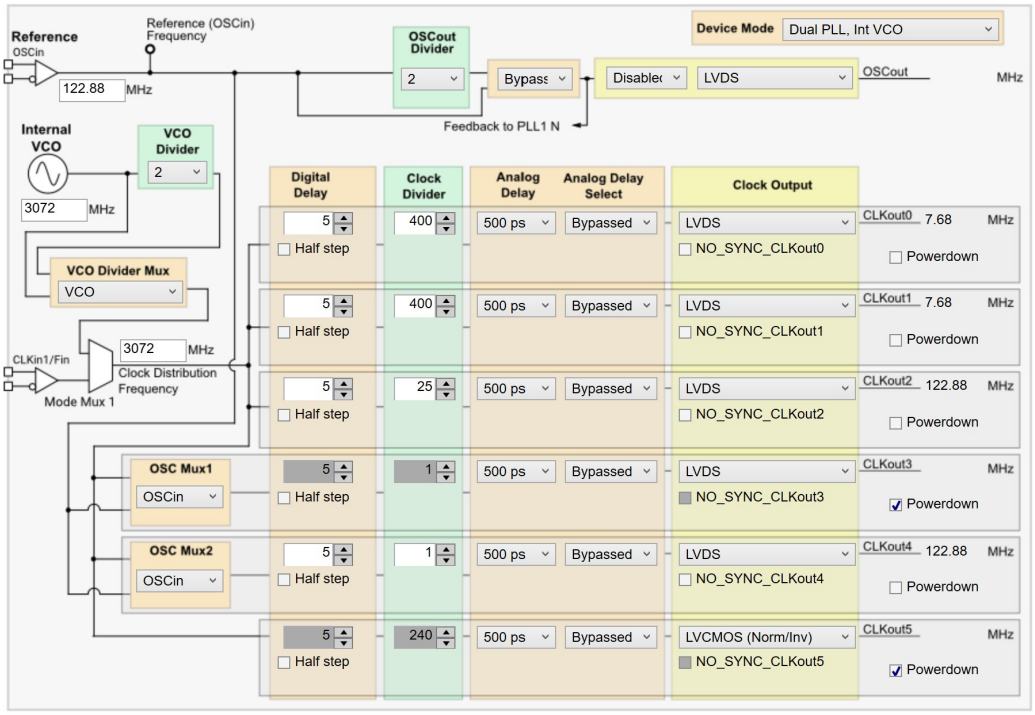


Figure B.6: The TICS Pro software GUI can be used to interpret and update the register values responsible for configurations of the onboard LMK04208 low-noise clock jitter cleaner.

JESD204B standard.

The user SYSREF and analogue SYSREF clocks are sourced from *CLKout0* and *CLKout1* of LMK04208, with the clock outputs shown in figures B.3 and B.6. The frequency is chosen to be 7.68 MHz, as it is required to be both a submultiple of the data path clock as well as be less than 10 MHz. As before, the configuration is done using the software driver and TICS Pro.

As they come from the same source, time of flight in the PL and converter tile data paths are determined and corrected for, through comparison of the user SYSREF clock signals with the directly connected analogue SYSREF signal. The analogue SYSREF appears as an external connection in the board design, *sysref_in*, automatically created by a Vivado wizard prompt and connected directly to the RF data converter input of the same name. The user SYSREF also appears as an external connection in the board design, *SYSREF_FPGA_C*, which must be manually connected through the following

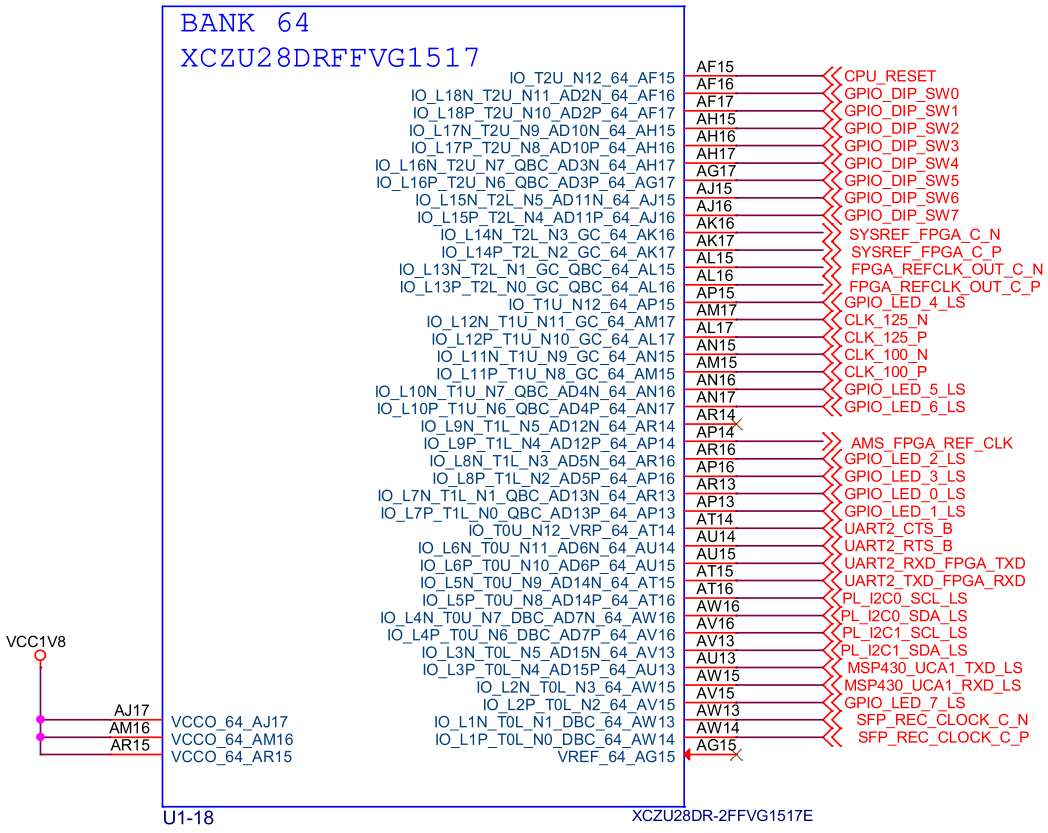
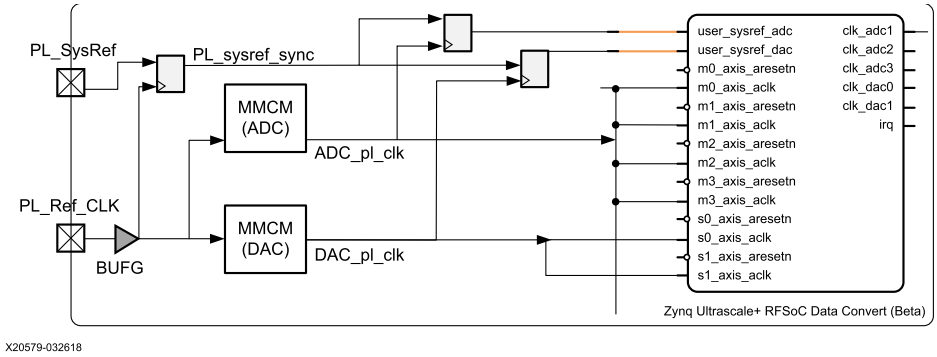
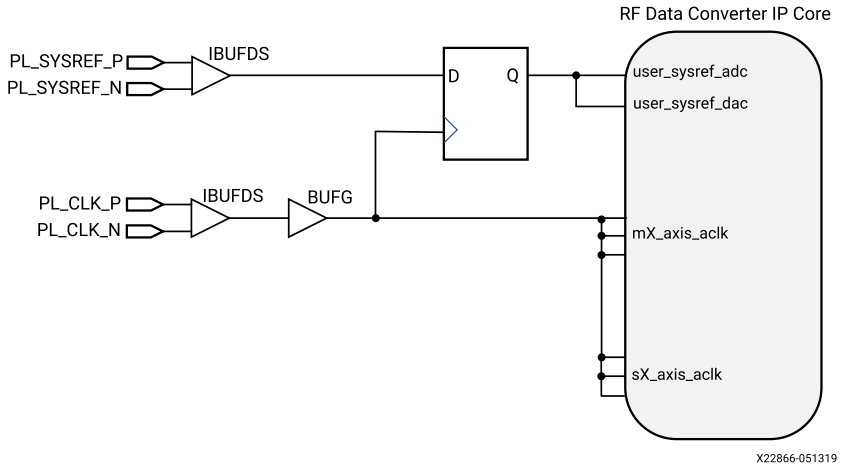


Figure B.7: Schematic of bank 64 on the ZU28DR, the RFSoC device integrated into the ZCU111 evaluation board, showing the pins corresponding to the *SYSREF_FPGA_C* and *FPGA_REFCLK_OUT* clocks.



(a)



(b)

Figure B.8: Additional schematics for the synchronisation between the PL AXI stream and user SYSREF clocks, from [8] (a) and [9] (b).

addition to the .xdc constraint file of the board design wrapper, based on the pins from the schematic in figure B.7:

```
set_property IOSTANDARD LVDS [get_ports {SYSREF_FPGA_C_clk_n[0]}]
set_property IOSTANDARD LVDS [get_ports {SYSREF_FPGA_C_clk_p[0]}]
set_property PACKAGE_PIN AK16 [get_ports {SYSREF_FPGA_C_clk_n[0]}]
```

```
set_property PACKAGE_PIN AK17 [get_ports {SYSREF_FPGA_C_clk_p[0]}]
```

Similarly to the AXI stream clock, the user SYSREF is connected through IBUFDS clock buffer and flip flop blocks before being connected to the RF data converter block module using the Vivado GUI, following the schematics in figure B.8.

Appendix C

RFSoC Initialisation

The AMD Software Development Kit (SDK) is an integrated development environment designed for developing C++ applications targeted at the embedded processors of the processing system. After generating the bitstream, SDK can be launched from within Vivado via the file menu.

After establishing a serial (COM) connection with the ZCU111 board's micro USB port, the following code snippets enable the configuration of the RFSoC on startup, linking up with the numerous hardware configurations in place.

A number of header files should be included. *xparameters.h* contains definitions of the hardware configuration, including base addresses for hardware modules. Headers as well as source code related to the RF data converter can be copied from manufacturer provided example IP application on GitHub [83], and included to enable a simplified configuration. *xil_io.h* provides interfaces for low level interaction with the hardware, including writing to and reading from memory-mapped registers.

```
1    #include "xparameters.h"
2    #include "xrfdc.h"
3    #include "xrfdc_mts.h"
4    #include "xrfdc_clk.c"
5    #include "xrfdc_clk.h"
6    #include "xil_io.h"
```

The following code initialises the driver and controller of the RF data converter:

```
#define XPS_BOARD_ZCU111
int Status;
XRFdc_Config *ConfigPtr;
XRFdc *RFdcInstPtr = &RFdcInst;
XRFdc_BlockStatus BlockStatus;
XRFdc_IPStatus myIPStatus;
XRFdc_Mixer_Settings MixerSettings = {0};
ConfigPtr = XRFdc_LookupConfig(RFDC_DEVICE_ID);
if (ConfigPtr == NULL) {
        return XRFDC_FAILURE;
}
Status = XRFdc_CfgInitialize(RFdcInstPtr, ConfigPtr);
if (Status != XRFDC_SUCCESS) {
        return XRFDC_FAILURE;
}

```

After waiting for the initialisation to complete, clock sources from the LMK04208 low-noise clock jitter cleaner and the LMX2594 RF synthesiser can be configured. Note that the configuration array for LMX2594 is contained in the example application *xrfdc_clk.c*, and omitted here for brevity. The default values for that array must be modified to enable *VCO_PHASE_SYNC*. Both this and the below array for LMK04208 are configured using the TICS Pro application.

```
usleep(1000);
unsigned int LMK04208_CKin[1][26] = {{0x00160040, 0x00143200,
 ↪  0x00143201, 0x00140322, 0xC0140023, 0x40140024, 0x80141E05,
 ↪  0x01100006, 0x01100007, 0x06010008, 0x55555549, 0x9102410A,
 ↪  0x0401100B, 0x1B0C006C, 0x2302886D, 0x0200000E, 0x8000800F,
 ↪  0xC1550410, 0x00000058, 0x02C9C419, 0x8FA8001A, 0x10001E1B,
 ↪  0x0021201C, 0x0180033D, 0x0200033E, 0x003F001F}};
LMK04208ClockConfig(1, LMK04208_CKin);
LMX2594ClockConfig(1, 2949120);
```

The power up status of the ADC and DAC tiles can be verified, shown below for the the first ADC tile:

```
    int powerup_status;
    int tile_state;
    powerup_status = myIPStatus.ADCTileStatus[0].PowerUpState;
    tile_state = myIPStatus.ADCTileStatus[0].TileState;
    printf("ADC 0 PowerUp Status: %u\n", powerup_status);
    printf("ADC 0 Tile State: %u\n", tile_state);
```

Similar status reports can be carried out for the other configurations, omitted here for brevity. Finally, the multi tile synchronisation procedure is carried out, including a report of the required adjustment:

```
    XRFdc_MultiConverter_Sync_Config ADC_Sync_Config;
    XRFdc_MultiConverter_Init(&ADC_Sync_Config, 0, 0);
    ADC_Sync_Config.Tiles = 0xf;

    u32 factor;
    for(int i=0; i<4; i++) {
        if((1<<i)&ADC_Sync_Config.Tiles) {
                XRFdc_GetDecimationFactor(RFdcInstPtr, i, 0, &factor);
                printf("ADC%d: Latency(T1) =%3d, Adjusted Delay"
                "Offset(T%d) =%3d\n", i, ADC_Sync_Config.Latency[i],
                 ↪  factor, ADC_Sync_Config.Offset[i]);
        }
    }
```

Bibliography

[1] M. Ackermann, S.K. Agarwalla, J. Alvarez-Muñiz, R.A. Batista, C.A. Argüelles, M. Bustamante et al., *High-energy and ultra-high-energy neutrinos*, 2022.

[2] M.J. Mottram, *A Search for Ultra-High Energy Neutrinos and Cosmic-Rays with ANITA-2*, Springer Berlin Heidelberg (2012), 10.1007/978-3-642-30032-5.

[3] P. Gorham, P. Allison, O. Banerjee, L. Batten, J. Beatty, K. Belov et al., *Constraints on the ultrahigh-energy cosmic neutrino flux from the fourth flight of anita*, *Physical Review D* **99** (2019) .

[4] C. Deaconu, *Searches for ultra-high energy neutrinos with anita*, 2019.

[5] Q. Abarr, P. Allison, J. Ammerman Yebra, J. Alvarez-Muñiz, J. Beatty, D. Besson et al., *The payload for ultrahigh energy observations (pueo): a white paper*, *Journal of Instrumentation* **16** (2021) P08035.

[6] A.M.D. Inc., "Ultrascale architecture dsp slice user guide (ug579)." `https://docs.amd.com/v/u/en-US/ug579-ultrascale-dsp`.

[7] A.M.D. Inc., "Zcu111 evaluation board user guide (ug1271)." `https://docs.amd.com/r/en-US/ug1271-zcu111-eval-bd`.

[8] A.M.D. Inc., "Ultrascale architecture pcb design user guide (ug583)." `https://docs.amd.com/v/u/en-US/ug583-ultrascale-pcb-design`.

[9] A.M.D. Inc., "Zynq ultrascale+ rfsoc rf data converter v2.4 gen 1/2/3 product guide." `https://docs.amd.com/v/u/2.4-English/pg269-rf-data-converter`.

[10] M.K. Gaillard, P.D. Grannis and F.J. Sciulli, *The standard model of particle physics*, *Reviews of Modern Physics* **71** (1999) S96–S111.

[11] C. Rovelli, *Notes for a brief history of quantum gravity*, 2001.

[12] G. Bertone, D. Hooper and J. Silk, *Particle dark matter: Evidence, candidates and constraints*, *Phys. Rept.* **405** (2005) 279 [hep-ph/0404175].

[13] L. Canetti, M. Drewes and M. Shaposhnikov, *Matter and antimatter in the universe*, *New Journal of Physics* **14** (2012) 095012.

[14] J.A. Formaggio, A.L.C. de Gouvêa and R.H. Robertson, *Direct measurements of neutrino mass*, *Physics Reports* **914** (2021) 1–54.

[15] S. Bilenky, *Neutrinos: Majorana or dirac?*, 2020.

[16] D. Aguillard, T. Albahri, D. Allspach, A. Anisenkov, K. Badgley, S. Baeßler et al., *Measurement of the positive muon anomalous magnetic moment to 0.20 ppm*, *Physical Review Letters* **131** (2023) .

[17] Muon g-2 collaboration, *The Muon g-2 experiment at Fermilab*, *EPJ Web Conf.* **137** (2017) 08001 [1701.02807].

[18] T. Aoyama et al., *The anomalous magnetic moment of the muon in the Standard Model*, *Phys. Rept.* **887** (2020) 1 [2006.04822].

[19] S. Borsanyi et al., *Leading hadronic contribution to the muon magnetic moment from lattice QCD*, *Nature* **593** (2021) 51 [2002.12347].

[20] P. Gorham, B. Rotter, P. Allison, O. Banerjee, L. Batten, J. Beatty et al., *Observation of an unusual upward-going cosmic-ray-like event in the third flight of anita*, *Physical Review Letters* **121** (2018) .

[21] M.G. Aartsen, M. Ackermann, J. Adams, J.A. Aguilar, M. Ahlers, M. Ahrens et al., *A search for icecube events in the direction of anita neutrino candidates*, *The Astrophysical Journal* **892** (2020) 53.

[22] I.M. Shoemaker, A. Kusenko, P. Kuipers Munneke, A. Romero-Wolf, D.M. Schroeder and M.J. Siegert, *Reflections on the anomalous anita events: the antarctic subsurface as a possible explanation*, *Annals of Glaciology* **61** (2020) 92–98.

[23] K.D. de Vries and S. Prohira, *Coherent transition radiation from the geomagnetically induced current in cosmic-ray air showers: Implications for the anomalous events observed by anita*, *Physical Review Letters* **123** (2019) .

[24] N. Arkani-Hamed, S. Dimopoulos and G.R. Dvali, *The Hierarchy problem and new dimensions at a millimeter*, *Phys. Lett. B* **429** (1998) 263 [hep-ph/9803315].

[25] J.E. Kim and G. Carosi, *Axions and the Strong CP Problem*, *Rev. Mod. Phys.* **82** (2010) 557 [0807.3125].

[26] A. Hayrapetyan, A. Tumasyan, W. Adam, J. Andrejkovic, B. Arnold, H. Bergauer et al., *Development of the cms detector for the cern lhc run 3*, *Journal of Instrumentation* **19** (2024) P05064.

[27] I. Zurbano Fernandez et al., *High-Luminosity Large Hadron Collider (HL-LHC): Technical design report*, .

[28] G. Bernardi, E. Brost, D. Denisov, G. Landsberg, M. Aleksa, D. d'Enterria et al., *The future circular collider: a summary for the us 2021 snowmass process*, 2022.

[29] P. Bambade, T. Barklow, T. Behnke, M. Berggren, J. Brau, P. Burrows et al., *The international linear collider: A global project*, 2019.

[30] CLICdp, CLIC collaboration, *The Compact Linear Collider (CLIC) - 2018 Summary Report*, 1812.06018.

[31] LIGO Scientific, Virgo collaboration, *Observation of Gravitational Waves from a Binary Black Hole Merger*, *Phys. Rev. Lett.* **116** (2016) 061102 [1602.03837].

[32] *Evidence for high-energy extraterrestrial neutrinos at the icecube detector*, *Science* **342** (2013) .

[33] ANITA, SalSA, IceRay collaboration, *Radio detection of ultra high energy neutrinos*, *Nucl. Instrum. Meth. A* **623** (2010) 413.

[34] D. Bergman, *Combined Fit of the Spectrum and Composition from Telescope Array*, *PoS* **ICRC2019** (2019) 190.

[35] T. Kashti and E. Waxman, *Astrophysical neutrinos: Flavor ratios depend on energy*, *Physical Review Letters* **95** (2005) .

[36] N. Song, S.W. Li, C.A. Argüelles, M. Bustamante and A.C. Vincent, *The future of high-energy astrophysical neutrino flavor measurements*, *Journal of Cosmology and Astroparticle Physics* **2021** (2021) 054.

[37] C.A. Argüelles, T. Katori and J. Salvado, *Effect of new physics in astrophysical neutrino flavor*, *Physical Review Letters* **115** (2015) .

[38] G.A. Askar'yan, *Excess negative charge of an electron-photon shower and its coherent radio emission*, *Zh. Eksp. Teor. Fiz.* **41** (1961) 616.

[39] J. Alvarez-Muñiz, A. Romero-Wolf and E. Zas, *Practical and accurate calculations of askaryan radiation*, *Physical Review D* **84** (2011) .

[40] O.B. Bigas, O. Deligny, K. Payet and V. Van Elewyck, *UHE tau neutrino flux regeneration while skimming the Earth*, *Phys. Rev. D* **78** (2008) 063002 [0806.2126].

[41] P. Allison, S. Archambault, J. Beatty, D. Besson, A. Bishop, C. Chen et al., *Low-threshold ultrahigh-energy neutrino search with the askaryan radio array*, *Physical Review D* **105** (2022) .

[42] A. Anker, S. Barwick, H. Bernhoff, D. Besson, N. Bingefors, G. Gaswint et al., *Targeting ultra-high energy neutrinos with the arianna experiment*, *Advances in Space Research* **64** (2019) 2595–2609.

[43] F. Halzen and S.R. Klein, *Invited review article: Icecube: An instrument for neutrino astronomy*, *Review of Scientific Instruments* **81** (2010) .

[44] M.G. Aartsen, R. Abbasi, M. Ackermann, J. Adams, J.A. Aguilar, M. Ahlers et al., *Icecube-gen2: the window to the extreme universe*, *Journal of Physics G: Nuclear and Particle Physics* **48** (2021) 060501.

[45] J. Aguilar, P. Allison, J. Beatty, H. Bernhoff, D. Besson, N. Bingefors et al., *Design and sensitivity of the radio neutrino observatory in greenland (rno-g)*, *Journal of Instrumentation* **16** (2021) P03025.

[46] L.H. Crockett, D. Northcote and R.W. Stewart, *Software Defined Radio with Zynq Ultrascale+ RFSoC* (Jan., 2023).

[47] Cummings, Austin, Krizmanic, John and Wissel, Stephanie, *Modeling neutrino and background signals for the payload for ultrahigh energy observations (pueo) experiment*, *EPJ Web Conf.* **283** (2023) 06018.

[48] L. Cremonesi, A. Connolly, P. Allison, O. Banerjee, L. Batten, J. Beatty et al., *The simulation of the sensitivity of the antarctic impulsive transient antenna (anita) to askaryan radiation from cosmogenic neutrinos interacting in the antarctic ice*, *Journal of Instrumentation* **14** (2019) P08011–P08011.

[49] J. Alvarez-Muñiz, R.A. Vázquez and E. Zas, *Calculation methods for radio pulses from high energy showers*, *Physical Review D* **62** (2000) .

[50] J. Alvarez-Muñiz, E. Marqués, R.A. Vázquez and E. Zas, *Coherent radio pulses from showers in different media: A unified parametrization*, *Physical Review D* **74** (2006) .

[51] S. Barwick, D. Besson, P. Gorham and D. Saltzberg, *South Polar in situ radio-frequency ice attenuation*, *J. Glaciol.* **51** (2005) 231.

[52] J.W. Goodman, *Statistical Optics*, Wiley (2015).

[53] R.N. Bracewell, *Fourier Transform and Its Applications*, McGraw-Hill (2000).

[54] A. Collaboration, "Anita simulation framework." `https://github.com/anitaNeutrino/anitaSim`.

[55] A.M.D. Inc., "Vivado overview." `https://www.xilinx.com/products/design-tools/vivado.html`.

[56] A.M.D. Inc., "Software development kit (sdk)." `https://www.xilinx.com/products/design-tools/legacy-tools/sdk.html`.

[57] A.M.D. Inc., "Vitis unified software platform." `https://www.xilinx.com/products/design-tools/vitis.html`.

[58] A.M.D. Inc., "Integrated logic analyzer (ila)." `https://www.xilinx.com/products/intellectual-property/ila.html`.

[59] A.M.D. Inc., "Tcl scripting in vivado." `https://docs.amd.com/r/en-US/ug894-vivado-tcl-scripting`.

[60] A.M.D. Inc., "Pynq — python productivity to amd adaptive computing platforms." https://docs.amd.com/r/en-US/ug894-vivado-tcl-scripting.

[61] C. Xie, *Improving Radio Frequency Detectors using High Performance Programmable Logic*, *PoS* **ICRC2021** (2021) 1028.

[62] S. Prohira, K. de Vries, P. Allison, J. Beatty, D. Besson, A. Connolly et al., *The radar echo telescope for cosmic rays: Pathfinder experiment for a next-generation neutrino observatory*, *Physical Review D* **104** (2021) .

[63] A.M.D. Inc., "Zynq ultrascale+ rfsoc rf data converter evaluation tool (zcu111) user guide(ug1287)." https://docs.amd.com/v/u/en-US/ug1287-zcu111-rfsoc-eval-tool.

[64] A.M.D. Inc., "Zynq ultrascale+ rfsoc zcu111 evaluation kit." https://www.xilinx.com/products/boards-and-kits/zcu111.html.

[65] IceCube collaboration, *The IceCube Neutrino Observatory: Instrumentation and Online Systems*, *JINST* **12** (2017) P03012 [1612.05093].

[66] IceCube-Gen2 collaboration, *IceCube-Gen2: the window to the extreme Universe*, *J. Phys. G* **48** (2021) 060501 [2008.04323].

[67] ANTARES collaboration, *ANTARES: the first undersea neutrino telescope*, *Nucl. Instrum. Meth. A* **656** (2011) 11 [1104.1607].

[68] KM3NeT collaboration, *The KM3NeT deep-sea neutrino telescope*, *Nucl. Instrum. Meth. A* **766** (2014) 83 [1408.1392].

[69] M. Agostini, M. Böhmer, J. Bosma, K. Clark, M. Danninger, C. Fruck et al., *The pacific ocean neutrino experiment*, *Nature Astronomy* **4** (2020) 913–915.

[70] A. Collaboration, P. Allison, S. Archambault, R. Bard, J.J. Beatty, M. Beheler-Amass et al., *Recent results from the askaryan radio array*, 2019.

[71] ARIANNA collaboration, *Targeting ultra-high energy neutrinos with the ARIANNA experiment*, *Adv. Space Res.* **64** (2019) 2595 [1903.01609].

[72] RNO-G collaboration, *Design and Sensitivity of the Radio Neutrino Observatory in Greenland (RNO-G)*, *JINST* **16** (2021) P03025 [2010.12279].

[73] Pierre Auger collaboration, *The Pierre Auger Cosmic Ray Observatory*, *Nucl. Instrum. Meth. A* **798** (2015) 172 [1502.01323].

[74] GRAND collaboration, *The Giant Radio Array for Neutrino Detection (GRAND): Science and Design*, *Sci. China Phys. Mech. Astron.* **63** (2020) 219501 [1810.09994].

[75] TAROGE, Arianna collaboration, *TAROGE-M: radio antenna array on antarctic high mountain for detecting near-horizontal ultra-high energy air showers*, *JCAP* **11** (2022) 022 [2207.10616].

[76] A.M. Brown, M. Bagheri, M. Doro, E. Gazda, D. Kieda, C. Lin et al., *Trinity: An Imaging Air Cherenkov Telescope to Search for Ultra-High-Energy Neutrinos*, in *37th International Cosmic Ray Conference*, 9, 2021 [2109.03125].

[77] M. Sasaki and T. Kifune, *Ashra Neutrino Telescope Array (NTA): Combined Imaging Observation of Astroparticles — For Clear Identification of Cosmic Accelerators and Fundamental Physics Using Cosmic Beams —*, *JPS Conf. Proc.* **15** (2017) 011013.

[78] A. Olinto, J. Krizmanic, J. Adams, R. Aloisio, L. Anchordoqui, A. Anzalone et al., *The poemma (probe of extreme multi-messenger astrophysics) observatory*, *Journal of Cosmology and Astroparticle Physics* **2021** (2021) 007.

[79] H. Saheb and S. Haider, *Scalable high speed serial interface for data converters: Using the jesd204b industry standard*, in *2014 9th International Design and Test Symposium (IDT)*, pp. 6–11, 2014, DOI.

[80] C.-W. Wu and P. Cappello, *Application-specific cad of vlsi second-order sections*, *IEEE Transactions on Acoustics, Speech, and Signal Processing* **36** (1988) 813.

[81] T. Instruments, "Texas instruments clocks and synthesizers (tics) pro software." https://www.ti.com/tool/TICSPRO-SW.

[82] A.M.D. Inc., "Zynq ultrascale+ rfsoc zcu111 evaluation kit." https://www.xilinx.com/products/boards-and-kits/zcu111.html.

[83] A.M.D. Inc., "Xilinx github." https://github.com/Xilinx.