SQ-CARS: A Scalable Quantum Control and Readout System

Ujjawal Singhal[§], Shantharam Kalipatnapu[§], Pradeep Kumar Gautam[§], Sourav Majumder,
Vaibhav Venkata Lakshmi Pabbisetty, Srivatsava Jandhyala, Vibhor Singh and Chetan Singh Thakur

Abstract—**Qubits are the basic building blocks of a quantum processor which require electromagnetic pulses in giga hertz frequency range and latency in nanoseconds for control and readout. In this paper, we address three main challenges associated with room temperature electronics used for controlling and measuring superconducting qubits: scalability, direct microwave synthesis, and a unified user interface. To tackle these challenges, we have developed SQ-CARS, a system based on the ZCU111 evaluation kit. SQ-CARS is designed to be scalable, configurable, and phase synchronized, providing multi-qubit control and readout capabilities. The system offers an interactive Python framework, making it user-friendly. Scalability to a larger number of qubits is achieved by deterministic synchronization of multiple channels. The system supports direct synthesis of arbitrary vector microwave pulses using the second-Nyquist zone technique, from 4 to 9 GHz. It also features on-board data processing like tunable low pass filters and configurable rotation blocks, enabling lock-in detection and low-latency active feedback for quantum experiments. All control and readout features are accessible through an on-board Python framework. To validate the performance of SQ-CARS, we conducted various time-domain measurements to characterize a superconducting transmon qubit. Our results were compared against traditional setups commonly used in similar experiments. With deterministic synchronisation of control and readout channels, and an open-source approach for programming, SQ-CARS paves the way for advanced experiments with superconducting qubits.**

Index Terms—**control-electronics, FPGA, RFSoC, superconducting qubit, quantum computing**

I. INTRODUCTION

QUANTUM processors utilize the properties of quantum parallelism and quantum interference in solving certain computational problems much faster than classical computers [1]. Qubits which are the building blocks of quantum processors have many realizations such as trapped-ions [2], semi-conducting quantum-dots [3], nitrogen-vacancy centers [4], and superconducting qubits [5], [6] etc. Among several such realizations, superconducting qubits are being aggressively pursued for scalable quantum computing platform [7], [8]. The

Ujjawal Singhal, Sourav Majumder and Vibhor Singh (Corresponding Author) are with the Department of Physics, Indian Institute of Science, Bangalore, India - 560012. (E-mail: vsingh@iisc.ac.in)

Pradeep Kumar Gautam is with Defence Research and Development Organisation, Bangalore, India and Department of Electronic Systems Engineering, Indian Institute of Science, Bangalore, India - 560012.

Shantharam Kalipatnapu, Vaibhav Venkata Lakshmi Pabbisetty, Srivatsava Jandhyala and Chetan Singh Thakur (Corresponding Author) are with Department of Electronic Systems Engineering, Indian Institute of Science, Bangalore, India - 560012. (E-mail: csthakur@iisc.ac.in)

[§]These authors contributed equally.

superconducting qubits are essentially nonlinear oscillators, which utilize the non-linearity of the Josephson inductance to form an effective quantum two-level system [9]. These systems need microwave pulses with gigahertz (GHz) frequency and latency in nanoseconds for control and readout, which are realized by the high speed electronics at room temperature [6], [10], [11].

The room temperature electronics that support the control and measurement of superconducting qubits pose three main challenges –

1) Direct synthesis of microwave signals: Traditionally for superconducting qubits, commercially available arbitrary waveform generators (AWG) based on Radio Frequency-Digital to Analog Converters (RF-DAC) with ≤ 1 GHz of analog bandwidth have been used. These are usually designed for general purpose tests and measurements. RF qubit control pulses, which are typically $4 - 8$ GHz are obtained by upconverting the AWG waveforms with analog mixers. These analog mixers come with a Local Oscillator (LO) leakage and imperfect sideband suppression [12], and their electrical properties vary with manufacturing tolerance and environmental effects. It demands a periodic calibration of the mixers to suppress unwanted image frequencies which is an overhead to the experiments [12].

2) Scalability: The decoherence of a quantum system is generally attributed to fluctuations in the device and environmental factors. However, the decoherence of the system is not just internal phenomenon but also depends on the master clock that drives the control electronics. Stable and coherent signals that drive and interact with the quantum system reduces the error significantly [13]. For a larger system, the control and readout would involve information processing to synthesize large number of control signals, estimate the state of the qubits and to provide a real-time feedback. In conventional systems, the technical details of these is often largely unavailable. As the number of qubits scales up, the use of these systems becomes challenging in terms of both cost and complexity.

3) Lack of Unified Interface: The control and readout electronics come from different vendors. In order to operate them, the user community has to deal with heterogeneous user interfaces which limits the productivity. In addition, a unified control and readout system is essential for advance experiments involving quantum feedback or error

correcting codes to minimize the feedbak latency [6].

This invites the need for a customized engineering solution that meets the requirement of the high data rates and signal processing of quantum computing community while keeping the system scalable, affordable and easy to work with. The integration of Field Programmable Gate Arrays (FPGA) with Radio Frequency -Digital to Analog Converter (RF-DAC) and Radio Frequency -Analog to Digital Converter (RF-ADC) have led to several breakthroughs such as active-reset, pulse-routing [14], faster readout [15], stabilization of Rabi-oscillations [16], quantum error correction [17], [18]. These hardware implementations have reached sufficient maturity and several commercial products from different vendors are also available [19]–[21].

Recently, the availability of RF-Dataconverters with high sampling rates in the order of gigahertz frequencies has gained interest [22]–[24]. The Xilinx Zynq Ultrascale+Radio Frequency System-On-Chip (RFSoC) [25] which is a family of devices of Field Programmable Gate Arrays (FPGA) comes with Digital to Analog Converters (DACs) and Analog to Digital Converters (ADCs) of very high sampling rates. It also includes up/down frequency converters using internal digital mixers which eliminates the need for external analog mixers. This first generation of RFSoC device XCZU28DR comes with eight high precision and low power DACs and ADCs with maximum sampling rates of 6.554 GSPS and 4.096 GSPS, respectively. These data converters are configurable and integrated with Programmable Logic (PL) resources of the RFSoC through AXI interfaces. The eight DACs are clocked by primary onboard reference Phase Locked Loop (PLL) LMK04208 and onboard RF PLL LMX2594 to generate the sample clocks [26] of the data converters. The ZCU111 evaluation board comes with a sister card XM500 on which, two DACs and two ADCs routed to High Frequency (HF) baluns with –1dB Pi pad attenuators, two DACs, and two ADCs routed to Low Frequency (LF) baluns with -3dB Pi pad attenuators, and remaining four DACs, and four ADCs routed to SMAs for use with external custom baluns and filters, all being routed to Sub Miniature Version A (SMA) connectors [26]. The baluns are primarily added to attenuate higher image frequency signals generated by DAC. The RFSoC board has been utilized to demonstrate their applicability for quantum computing in recent works [27]–[31]. However, these works do not address the issues of scalability, direct synthesis of microwave pulses, and a user-friendly interface entirely.

This work utilises a single FPGA board (ZCU111 by Xilinx) that is populated with the XCZU28DR device to develop an integrated framework to support a scalable Quantum Control system. The proposed framework, Scalable Quantum Control and Readout System (SQ-CARS) supports up to four qubits and can be easily scaled to control higher number of qubits. The main contributions of this work are -

1) Phase synchronisation of all channels using Multi-Tile Synchronisation
2) Direct digital synthesis of microwave pulses using Mix mode technique
3) Arbitrary waveform generation and lockin detection to

microwave quadratures
4) A python based programming interface to configure and control the above functionalities

The rest of the paper is organised as follows. Section II describes the system level architecture of SQ-CARS along with the design and functionalities of its individual blocks. Section III discusses the performance characterisation of SQ-CARS in which we first benchmark the performance of the Continuous Wave (CW) microwave signals at room temperature by measuring various parameters like spurious-free dynamic range (SFDR), single-sideband phase noise and the reduction in amplitudes while generating signals in multiple Nyquist zones, latency of RF-SoC pipeline, multichannel phase synchronisation and comparison of the proposed system with state-of-the-art platforms. In section IV, to show the applicability of our technique, we generate the control pulses required for the control of the superconducting qubit and carry out coherence measurements of a transmon qubit. Finally, conclusion is drawn in Section V.

II. SYSTEM ARCHITECTURE OF SQ-CARS

This section describes the overall architecture of the control and readout system, its submodules and their functionalities.

SQ-CARS is designed in modular way with both hardware (FPGA logic) and firmware easily scalable for supporting more number of qubits. The current design which is publicly available on GitHub [32] can support up to 4 qubits. This limitation is imposed by the number of DACs and ADCs available on the board. With Channel level modularity SQ-CARS can be easily scaled for control and readout of more number of qubits either by increasing the number of DACs and ADCs or by frequency multiplexing the available channels.

A. PYNQ based User Interface

The Python based framework allows us to create an abstraction layer, which masks the underlying details of the hardware implementation and offers clean and user friendly configuration interface to the physics user community. The whole architecture of the system is divided between Programmable Logic (PL) and the Processing System (PS) as shown in Figure 1. ZCU111 is based on XCR28DR SoC, which has multicore ARM processor (PS) and Programmable FPGA Logic (PL). The PS runs Python productivity for Zynq (PYNQ) framework on Linux OS and can configure and control the PL design using the Overlays.

We utilized the available RFdc python code [23], [33] and made modifications to incorporate Multi-Tile Synchronization (MTS). Python classes were created for other hardware IPs on the PL to facilitate communication between the framework and hardware design modules. A hierarchical design approach is adopted to ensure a well-structured system which would be beneficial as the number of qubits and their corresponding IPs scale up.

Figure 2 illustrates the block diagram of the framework running on the PS and its logical connection with the SQ-CARS hardware design running on the PL. The communication between the PS and PL can be divided into two parts:
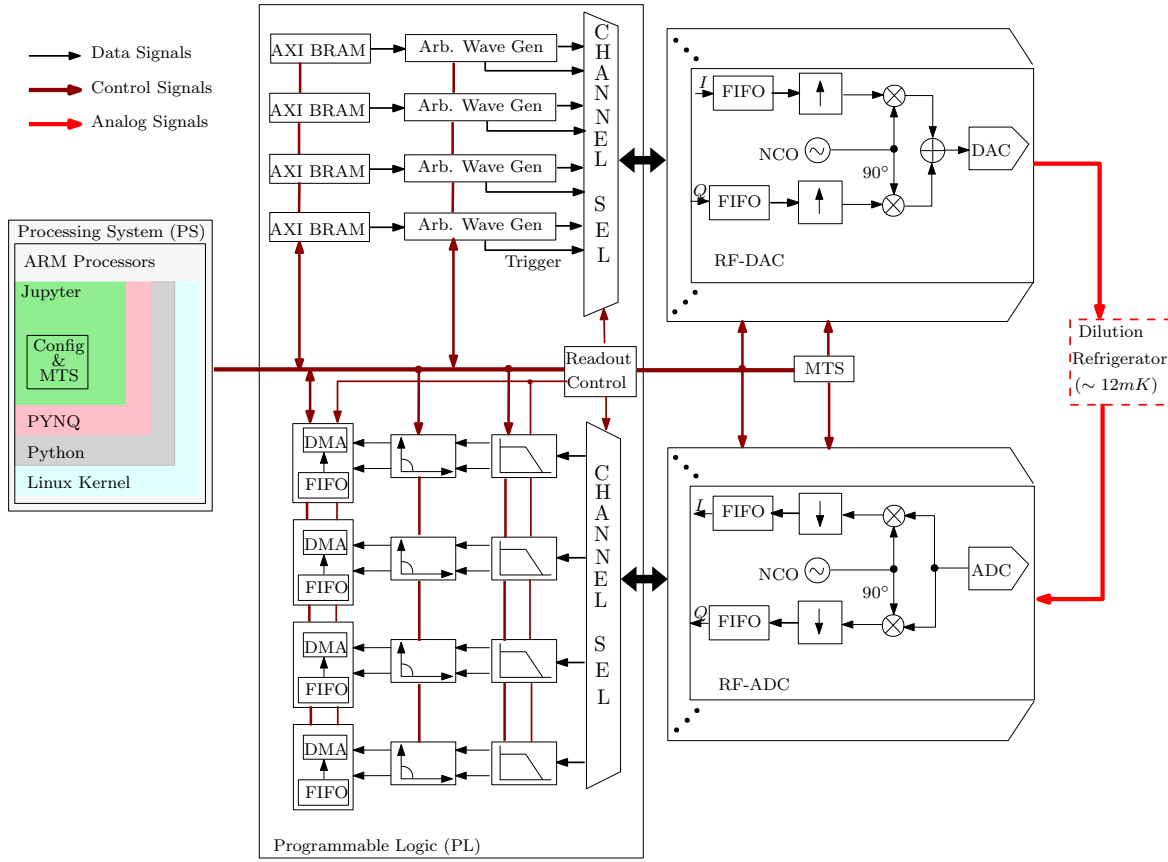
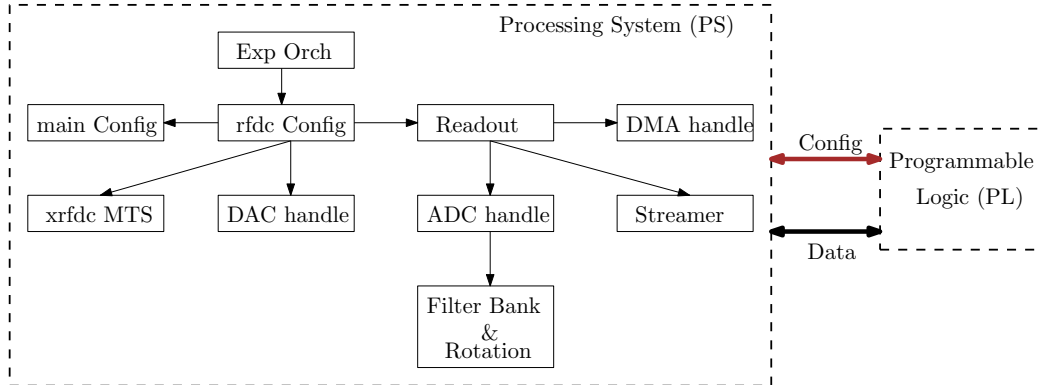Fig. 1: System Architecture of 4-Qubit SQ-CARS

Fig. 2: The Class hierarchy of SQ-CARS Python Framework

the Configuration & Sync channel, and the data channel. Both channels are implemented using AXI interfaces available between the PS and PL. The Configuration & Sync channel employs lightweight communication using Memory Mapped Input Output (MMIO) PYNQ library for read/write operations. The data channel utilizes the AXI Direct Memory Access (DMA) driver to collect data from the PL. The modular framework can be easily adapted for other versions of the RFSoC board and can be initialized and used as a Jupyter notebook or a Python script. The Experiment Orchestrator (Exp Orch) shown in Figure 2 receives the experimental parameters and controls the overall flow of the experiment. The mainConfig class holds the system configuration, such as the bitstream

file, frequency settings, amplitude settings, remote host and port information, trigger settings, and more. The rfdcConfig class is responsible for configuring the RF data converters. It creates handles for the DAC and ADC channels, assigns Block RAM (BRAM) addresses, and sets up the RF data converter for operation. This class also includes methods for initializing and running the Multi-Tile Synchronization (MTS) process. The DAC and Readout classes handle the Signal Generation (AWG) and Readout pipeline respectively. They set NCO frequencies and phases, configure the DMA, and manage the data streaming to a remote host. The Experiment Orchestrator initiates the experiment by loading the waveform into the corresponding DAC's BRAM. Once the experiment

```python
#all timing parameters are in nano sec
exp_config = {
  "exp_type": "T1",
  "continuous": 1,
  "qubit_freq": 4690.2968955,
  "readout_freq": 5962.36,
  "mode": 0,
  "repetition_rate": 300000,
  "time_between_pulses": 1000,
  "initial_amp": 10,
  "trigger_delay": 0,
  "trigger_width": 4000,
  "amplitude_factor": 30,
  "amplitude_steps": 70,
  "gaussian_sigma": 400,
  "gaussian_pulse_duration": 900,
  "outer_loop_count": 1,
  "inner_loop_count": 10000,
  "inner_loop_step": 0,
  "data_fetch_time": 500000,
  "loopback": 0,
  "wave_type": "gaussian"
}
```

Listing 1: Listing showing the Configuration of Experiment by the user

is started, no further communication between the framework and PL is required, which reduces the experiment duty cycle.

Listing 1 shows snippet of the code used for taking experiment parameter input from the user. Snippet shown in Listing 2 shows the initialization process by providing the handle to user for modifying various parameters on the fly according to the requirement of the experiment.

B. Arbitrary Waveform Generation and Control

The control side of superconducting quantum system involves generation of microwave pulses of various shapes and duration depending on the experiment at hand. DACs of this board are packaged into 2 tiles (Tile-0 and Tile-1), each tile containing 4 DACs, thus providing total 8 channels. Tile-0 Channels are used to generate control microwave pulses, while Tile-1 DACs are used to generate corresponding readout pulses. Tile-1 DACs can also be used independently to generate control pulses, extending the design's capability to control upto 8-qubits.

Each of the DACs can be controlled independently by the *Arbitrary Wave Generator* (AWG) block which is shown in Figure 3. The parameters that control the shape, amplitude, duration etc. of the microwave signals are controlled by this block. It can generate arbitrary waveform by reading continuous samples from local memory of PL (BRAM). This allows for real-time play of the samples and provides control to the user for changing the shape and duration of the waveform. These samples are written from Python framework along with other experimental parameters like mode, loop and time delays to the BRAM. The available local memory in PL is 4.75MB of BRAM. Each sample needs two bytes of storage. When all the eight DAC channels are used, a maximum storage of 311K samples can be allotted for each channel. The current

```python
thisConfig = SQ_CARS.mainConfig(SQ_CARS.config
    )
rfdc_handle = rfdcConfig(thisConfig)
# Example of setting mixer frequency of DAC
for i in rfdc_handle.dac_channels:
    # Generation of wave
    w1_I = gen_wave(thisConfig.exp_config[
        "gaussian_pulse_duration"], thisConfig.
        exp_config["gaussian_sigma"])

    # Setting NCO Frequency
    rfdc_handle.dac[i].set_nco_freq(thisConfig
        .exp_config["qubit_freq"])
for i in rfdc_handle.adc_channels:
    # Setting up DMA and streamer
    rfdc_handle.readout[i]._dma.init_dma()
    rfdc_handle.readout[i].init_streamer()

    # Example of passing the value to IP
        blocks
    rfdc_handle.readout[i]._adc_pipeline.
        _filter.set_iir_params(0.53) #Filter
        cutoff freq in MHz
    rfdc_handle.readout[i]._adc_pipeline.
        _filter.set_iir_order(1) # Filter
        order

# Call to MTS routine and starting the
    experiment
rfdc_handle.run_MTS()
rfdc_handle.run_exp()

for i in rfdc_handle.adc_channels:
    jobs.new('rfdc_handle.readout[i].
        dma_streamer_thread()')
    rfdc_handle.readout[i].set_readout_update
        (1)
    jobs.status()
```

Listing 2: Listing to change the experimental parameters in Python framework

design supports 64K samples per channel, which is equivalent to $80\ \mu s$ of playtime at the current FPGA clock frequency of 192 MHz, which is a sufficient number for most experimental purposes. The signal generator has the capability to play the flat part or time between pulses of the waveform utilizing an internal counter, without actually storing the respective portions of it in BRAM. This effectively allows us to play pulses of longer duration more than $80\ \mu s$ in experiments like Time Rabi and Ramsey experiment.

The *mode* allows to play continuous waveform or a fixed number of pulses depending on the experiment requirement, thus catering to a wide spectrum of quantum experiments. The *loop* parameters help to run an experiment multiple times repeatedly to collect as many measurements as required. The *Controller* also controls the generation of a trigger which facilitates for the timely capture on the readout side. The *timing* parameters decide the delay of the trigger and width of the arbitrary wave pulses which are fully programmable. The sample data read from BRAM is sent to *Data Splitter* and the amplitude of each sample is scaled by the *Scaler* block. These scaled samples are further merged and the interleaved I and Q
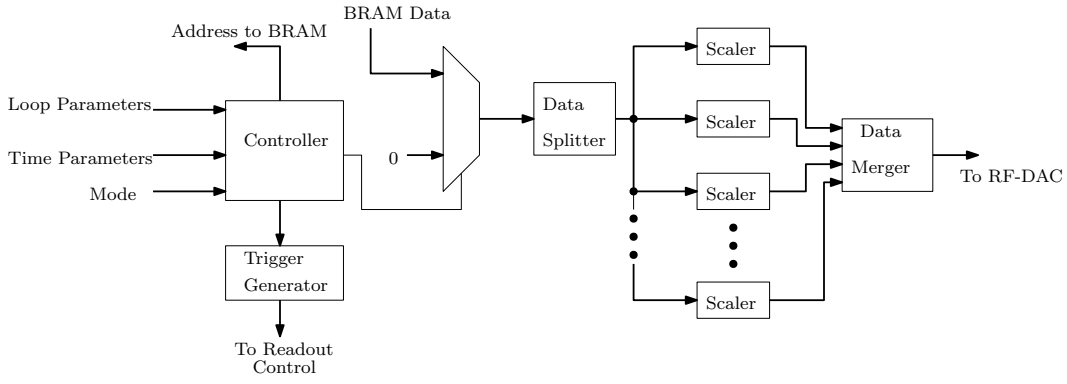
Fig. 3: Block Diagram of Arbitrary Waveform Generation and Control

samples are fed to the RF-DAC. The *Channel Select* allows us to select whether control or readout pulses be played through DACs.

The *RF-DAC* consists of a First In First Out (FIFO), an interpolation filter, a mixer, and a DAC as shown in Figure 1. The scaled sample values from the *Arbitrary Wave Generator* are fed into AXI Stream (AXIS) FIFO of the corresponding DAC channel. At higher sampling rates of Data Converters, the data streaming clock of the DAC cannot be pushed in the orders of their sampling rates. So a digital upsampling is necessary. This function on the DAC side is realized by the Interpolation Filters. The interpolation rate can be chosen among 1x, 2x, 4x, and 8x.

Algorithm 1 shows the pseudo code of the control and readout that suits the needs of the user in characterizing a superconducting qubit. This algorithm is implemented as the entirety of the hardware architecture.

Using this algorithm, basic characterizing experiments such as the measurement of energy relaxation time, dephasing, Rabi-oscillations in time-domain or power domain *etc.* can be configured easily. It takes arguments such as Ramsey, Time-Rabi, T_1, Power-Rabi, ϵ_A, ϵ_T, ϵ_t, $Experiments$ and N_{iter} as the inputs. N_{iter} indicates the the number of RF pulses that needs to be sent to the dilution refrigerator. The parameters ϵ_A, ϵ_T, and ϵ_t represent the increments in power, trigger delay and time between the pulses respectively. Depending on the kind of experiment that needs to be performed, these parameters are initialised and updated accordingly as shown in Algorithm 1.

C. Multi-Nyquist zone operation

According to Nyquist criteria, the sampling rate limits the frequency that can be faithfully reconstructed to be less than half the sampling rate. However, in practice, when a signal is sampled, its images appear at higher frequencies. Each Band of the spectrum with width $\frac{F_S}{2}$ is termed as Nyquist zone (NZ). For example, the range from DC to $\frac{F_S}{2}$ is termed as first-Nyquist zone, $\frac{F_S}{2}$ to F_S is called second-Nyquist zone and so on [34]. The images above the first-Nyquist zone can be utilized according to the need. The output voltage of DAC

Algorithm 1: Pseudo code of Control and Readout

Input: $Ramsey$, $Time\_Rabi$, T_1, $Power - Rabi$, ϵ_A,ϵ_T,ϵ_t, $Experiments$, N_{iter}
Initialize: $n = 0$; $N = 0$;
if $Power\_Rabi$ **then**
  | $\Delta A = \epsilon_A$; $\Delta T = 0$; $\Delta t = 0$;
else if $T1$ **then**
  | $\Delta A = 0$; $\Delta T = \epsilon_T$; $\Delta t = 0$;
else if $(Ramsey) \vee (Time\_Rabi)$ **then**
  | $\Delta A = 0$; $\Delta T = 0$; $\Delta t = \epsilon_t$;
else
  | $\Delta A = \Delta T = \Delta t = 0$;
end

1 **while** $(N < Experiments)$ **do**
2   **while** $(n < N_{iter})$ **do**
3   Play Control Pulses;
4   Capture Readout Data on Trigger;
5   Process the Captured Data;
6     $n = n + 1$;
7   **end**
8   $A = A + \Delta A$;
9   $T = T + \Delta T$;
10   $t = t + \Delta t$;
11   $N = N + 1$;
12 **end**

---

can be represented [35] as,

$$v(t) = \left[x(t) \sum_{k=-\infty}^{\infty} \delta(t - kT) \right] * r(t), \qquad (1)$$

where $x(t)$ is the desired waveform whose samples are being generated, $r(t)$ is the reconstruction waveform, and $T = \frac{1}{F_S}$. Taking Fourier transform on both sides of the above equation, we get

$$V(\omega) = \left[X(\omega) * \sum_{n=-\infty}^{\infty} \delta(\omega T - 2\pi n) \right] R(\omega), \qquad (2)$$

where $V(\omega)$, $X(\omega)$ and $R(\omega)$ represent the Fourier transform of $v(t)$, $x(t)$ and $r(t)$, respectively. From equation 1 and 2, it becomes evident that the sampled signal is passed through a system having transfer function $R(\omega)$ and when a signal
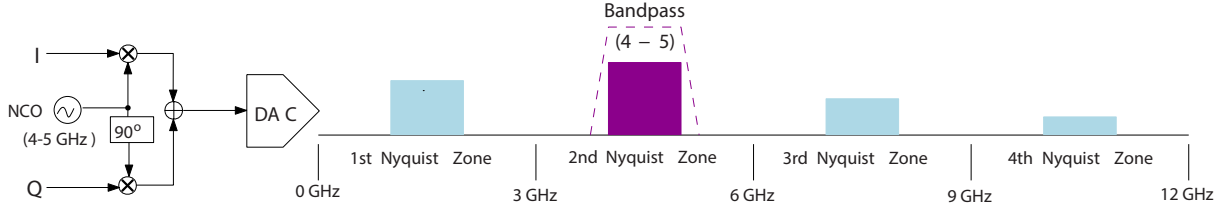
Fig. 4: Up conversion scheme on RFSoC using the on-chip IQ mixer and a numerically controlled oscillator. Using the maximum sampling rate F_S = 6.554 GHz, and mix-mode of operation of DAC, the signal/image can be pushed in the second Nyquist zone which can be conditioned using a Mini-circuit VBFZ-3590-S+(15542) band-pass filter.

is sampled we get its copies in higher frequency ranges. Therefore, the output signal strength of the copies in different range of frequencies gets affected according to the response of $R(\omega)$.

The RFSoC supports two modes of operation: Normal mode or Non-Return-to-Zero (NRZ) mode and Mix mode or Return-to-Complement (RTC) mode which determines $R(\omega)$. In the NRZ mode of operation the DAC uses a fixed level reconstruction waveform during one clock cycle. It has high output power in the first Nyquist zone, but low output power in the second Nyquist and beyond. RTC mode or the mixed mode, it outputs the sample for the first half of the clock period and then inverts the sample for the second half of the clock period. The resulting frequency response shows high power in the second Nyquist zone and attenuation in the first Nyquist zone. The output power goes to zero at DC, and $2F_S$. This mode provides highest power for the second Nyquist zone applications. Mathematically, the reconstruction waveforms can be written as,

$$r(t) = \begin{cases} u(t) - u(t-T), & \text{NRZ mode} \\ u(t) - 2u(t-T/2) + u(t-T), & \text{RTC mode} \end{cases} \tag{3}$$

$$R(\omega) = \begin{cases} Te^{-i\omega T/2} \, \mathrm{sinc}\left(\frac{\omega T}{2}\right), & \text{NRZ mode} \\ Tie^{-i\omega T/2} \, \mathrm{sinc}\left(\frac{\omega T}{4}\right) \sin\left(\frac{\omega T}{4}\right), & \text{RTC mode} \end{cases} \tag{4}$$

The maximum sampling rate of DACs in RFSoC is 6.554 GSPS. Since the frequencies necessary for the control side of qubit are typically in the range of $4-8$ GHz, the desired RF band for signal synthesis fall in the second Nyquist zone. It can be accessed by operating in mix mode while maintaining highest power. The major advantage of this approach is that RF signal is generated using onboard NCOs and a digital IQ-mixer. Therefore, it allows a full vector control of amplitude, frequency and phase on the fly. To suppress the images in other Nyquist zones and signal conditioning, we use standard coaxial band-pass filters. In principle, these filters can be incorporated on a custom daughter board. A schematic of the flow and different Nyquist zones are shown in Figure 4.

D. Readout

On the readout side, the signals from the dilution refrigerator are fed to RF-ADCs. The board has four tiles, each

tile containing two ADCs. Among the eight ADCs in total, four ADCs are differential ended and four ADCs are single ended. The design supports four independent readout lines. The user has the flexibility to chose between a single-ended or differential ended ADC for each readout channel which is facilitated by *Channel Select*. It also allows us to lock ADC to any particular DAC channel for capturing the readout based on the internal trigger. The ADCs are designed to directly measure the signals in higher Nyquist zones. The zone of operation is indicated to the digital calibration engine of RFSoC to ensure optimal performance of the ADC. The input samples are provided by the parallel digital interface of the high-speed ADC. The incoming signal is down-mixed using the internal NCO of RF-ADC. Upon down-mixing and decimation, the signal is low-pass filtered (LPF) and smoothed using a moving-average (MA) filter. The NCO frequency, cut-off frequency of the LPF can be configured using the Python framework based on experiment. The in-phase (I) and quadrature (Q) samples received after the filter are fed to Rotation block which performs,

$$\begin{pmatrix} I' \\ Q' \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} I \\ Q \end{pmatrix} \tag{5}$$

Such a rotation block, equivalent to the auto-alignment of phase in a lock-in measurement, is a helpful feature while performing qubit readout using single quadrature. The angle of rotation can be changed by the user using the Python framework. This estimation can be used to determine the qubit state or to provide an active feedback to the control side to further fine-tune the superconducting system.

The filter and rotation blocks can also be bypassed to capture the raw data. The resultant samples are fed into AXI Stream FIFO for transfer to PS using AXI-DMA block. The AXI Stream handshake signals like *tvalid* and *tlast* are generated using Readout Control Block, depending on the internal trigger received from Signal Generation Block. Stream to Memory Mapped (S2MM) port of AXI-DMA engine is connected to the PS via AXI Slave interface on ZYNQ SoC. The DMA engine is configured in Scatter Gather (SG) mode to facilitate use of cyclic buffer and eliminate need for PS to continuously provide descriptors to the DMA engine. To fetch the descriptors, a separate SG port of DMA is used. A custom wrapper is created in PYNQ to support DMA in Scatter Gather mode. Both descriptors and data buffers for DMA are kept in DDR memory attached to PS. Each readout channel is

equipped with its own separate DMA channel, providing fine grained control on data capture on readout side. The readout data received on PS is transferred to remote PC using the PS GEM-3 Ethernet which is either saved in file or played using python scripts for initial startup of the experiment.

E. Multi-Tile Synchronisation

To control and measure multiple qubits, the output waveforms of multiple ADC and DAC channels requires to be synchronised both in time and phase. This can be achieved by an external 10 MHz clock which is fed to LMK04208 to synchronize the timebase of the FPGA board with other instruments and Multi Tile Synchronisation (MTS). The RF-ADC and RF-DAC constitute dual clock FIFOs. The data converters in a single tile share the clocking and data infrastructure, therefore the FIFO latency within a tile remains the same. However, the latency of the FIFOs can vary from one tile to another. MTS enables to achieve the relative multi-tile alignment. A single master clock generates all clock signals required for RF data converters and the programmable logic. A custom python wrapper for C-drivers is developed to configure and invoke MTS.

III. Performance Characterisation of SQ-CARS

This section describes various measurements and characterisation done with 4-qubit SQ-CARS design on ZCU111 to benchmark the performance of the proposed system with the state-of-the-art traditional setup. For all the measurements discussed in this section, the DACs and ADCs are operated at sampling rates of 6.144 GSPS and 3.840 GSPS respectively with their NCOs operating in Mix/Normal mode as per the requirement.

A. Magnitude Response of DAC

To measure the power dependency in NRZ and RTC modes, the DAC output response is being characterized by enabling the NCO whose frequency is changed on the fly and the corresponding magnitude is recorded. To compensate for the loss in magnitude, an inverse sinc filter is applied in both modes of operation. The recorded DAC output response is shown in Figure 8(a). The DAC output response in NRZ mode follows a sinc function. It is evident from the Figure 8 that the signal power is maximum in the second and third Nyquist zone when operating in the RTC mode.

B. Multi Channel Phase Synchronisation

The phase synchronisation between the control and readout channels is critical in a multi-qubit system. QICK [28] has reported achieving inter-channel phase synchronisation using the tprocessor. However, their approach does not consider the variable delay introduced by the RF-SoC pipeline, due to NCOs and FIFOs. To address this limitation, our work SQ-CARS, utilises MTS with on-board NCOs enabled, to generate and capture phase synchronised signals, demonstrating inter-channel synchronisation from generation by DACs to capture on ADCs.

To test the multi channel phase synchronisation on control side, we generate a 1 MHz sinusoid by two DACs on ZCU111. These signals are being observed on an oscilloscope for jitter measurement. The channel to channel jitter of two DACs is measured by repeatedly sampling the jitter 4000 times with a time interval of 100 μs . The histogram of jitter is shown in Figure 6. The standard deviation of channel to channel jitter is $\sim 0.6\ ps$ which is better than $\sim 5\ ps$ and $< 1\ ps$ as reported in Ref [29] and Ref [31], respectively. Low standard deviation of jitter shows stable channel synchronisation to support high-precision synchronized qubit operations.
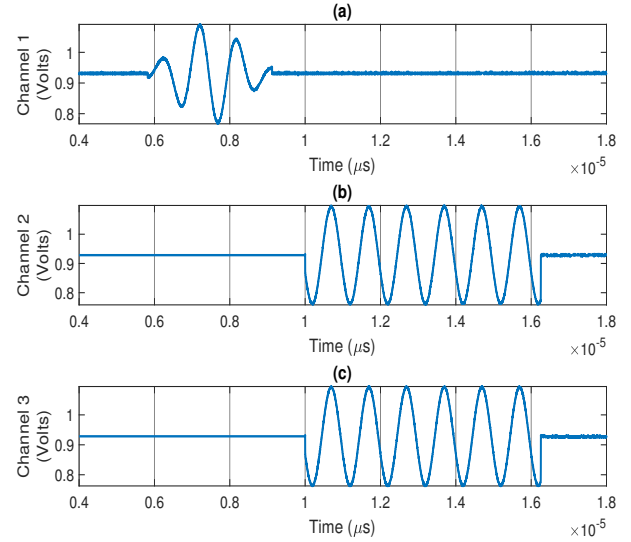

Fig. 5: Phase Synchronisation across various DAC Channels (a) Gaussian control pulse on Channel 1 (b) Readout pulse on Channel 2 (c) Readout pulse on Channel 3. The readout pulse starts with a delay after the control pulse. It can be seen that the readout pulses on Channels $2 - 3$ are in synchronisation.
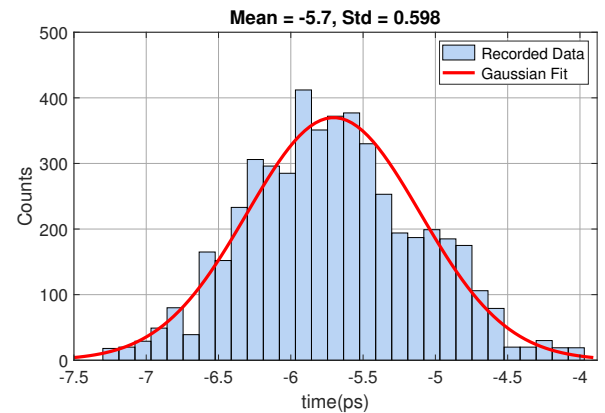

Fig. 6: Histogram of channel to channel jitter of two DACs

In order to demonstrate the multi channel phase synchronised capture on readout side, a 1 MHz sinusoid generated by DACs as earlier (which are already phase synchronised) is

fed to ADCs. The resultant signal at the output of the ADC, the I and Q channels are captured. and found to be in phase synchronisation. Figure 7 shows the output of I-channels of two such ADCs.
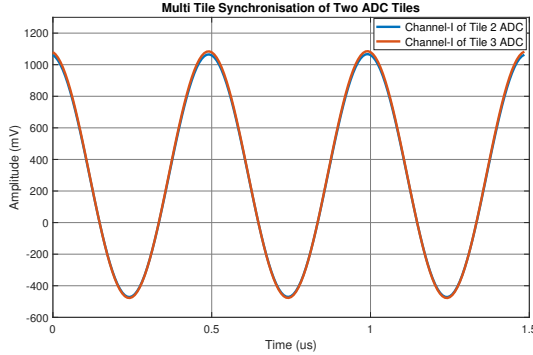

Fig. 7: Multi-Channel Phase Synchronisation of two ADC channels

C. Latency

Latency is an important metric for qubit readout, active feedback, and error-correction protocols. The RFSoC ADC and DAC pipelines have a series of modules of FIFO, Interpolation/Decimation filters, Mixers as shown in Figure 1. These modules can be used or bypassed as per the requirement. The latency changes as we bypass or utilise these modules. Latency measurement was done using loopback between DAC and ADC and observing the markers in Integrated Logic analyzer (ILA) running at 192 MHz clock. The interpolation rate of DAC is 8x and decimation rate of ADC is set to 4x, the mixer is enabled and the DAC and ADC channels are synchronised using MTS. The round-trip latency including DAC, ADC and the digital AXI interfaces associated with the data converters is measured to be 48 cycles equalling 250 ns with the design operating at 192 MHz. This number is comparable to the latency measurement reported without MTS by Quantum Instrumentation Control Kit (QICK) [28]. Each half of the measured latency is contributed by RF-DAC and RF-ADC pipelines.

D. Noise Characterisation

Another important benchmarking metric for continuous mode of operation is the single-sideband phase noise. A frequency tone (carrier) is generated, and at various offset frequencies from the carrier, the power is measured in a specified bandwidth of 1 Hz.

We use a signal analyzer (Rohde and Schwartz FSV-13) to perform the phase noise measurements. Figure 8(b) shows the measurement of SSB phase noise plotted for different offset frequencies around the carrier frequency of 4.5 GHz generated from the XCZU28DR device. The measured phase noise of -102 dBc/Hz at 4.5 GHz carrier frequency at offset 10 kHz is comparable to the phase noise performance of standard test and measurement RF Signal generators [36]–[40].
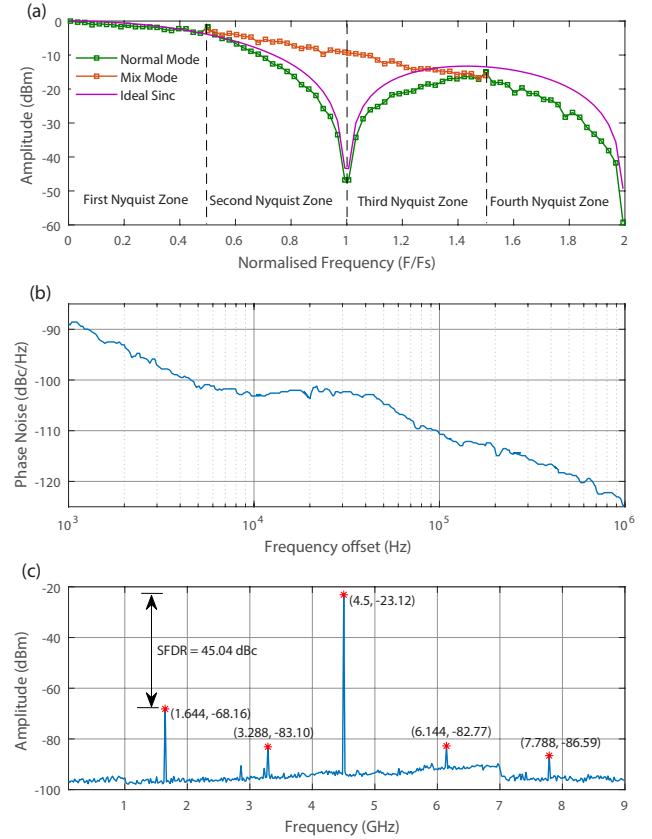

Fig. 8: RF performance evaluation of XCZU28DR: (a) amplitude response of the DAC across various Nyquist zones in normal mode (NRZ) and mix mode (RTC mode). A plot of sinc function in included for comparison. (b) Single sideband (SSB) phase poise of XCZU28DR at carrier frequency of 4.5 GHz. (c) A large span spectrum for the measurement of SFDR at 4.5 GHz carrier tone.

While generation of signal using multi-zone Nyquist technique expands the scope of frequency domain capabilities of a DAC, the images generated in other zones need to be carefully suppressed to achieve a practically useful spurious-free dynamic range (SFDR) [41]. Figure 8(c) shows the various spurs for a carrier tone of 4.5 GHz while using a standard bandpass-filter minicircuit VBFZ-3590-S+(15542). With this general purpose filter, we achieve a SFDR of nearly 45 dB. Our focus in this study has been on the generation of control signals for the superconducting qubits in the frequency range of 4-4.5 GHz. It is important to mention here that this value is currently limited by the choice of the filter, and can be further improved by using tunable cavity or switchable filter banks [42], [43].

The ADCs are designed to directly measure the signals in higher Nyquist zones. The zone of operation is indicated to the digital calibration engine of RFSoC to ensure optimal performance of the ADC. A 6 GHz signal at -23 dBm is fed to ADC in mix-mode operating at a sampling rate of 4.096 GSPS. The resultant image would appear at $6000 - 4096 = 1904$ MHz. The frequency response of the signal captured by ADC is shown in Figure 9.
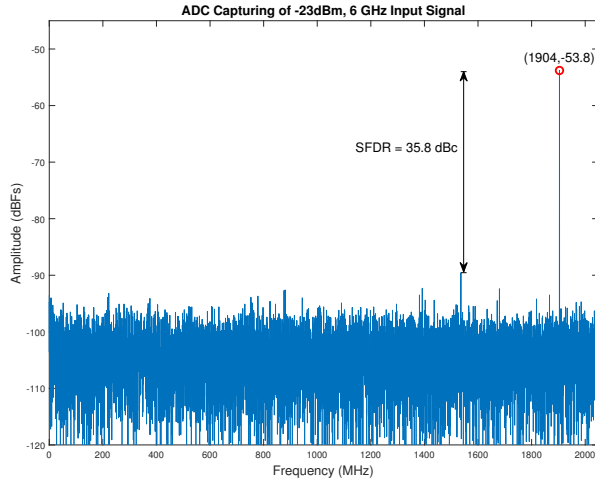
Fig. 9: Spectrum of 6 GHz signal acquired by ADC in mix mode of operation

To measure the linearity of ADC response in the Mix-mode, a signal is fed with varying input power for frequencies between $6-7$ GHz and the resultant amplitude (dBFs) is recorded. The measured amplitude at the ADC shows a good linear behaviour with respect to the input power as shown in Figure 10.
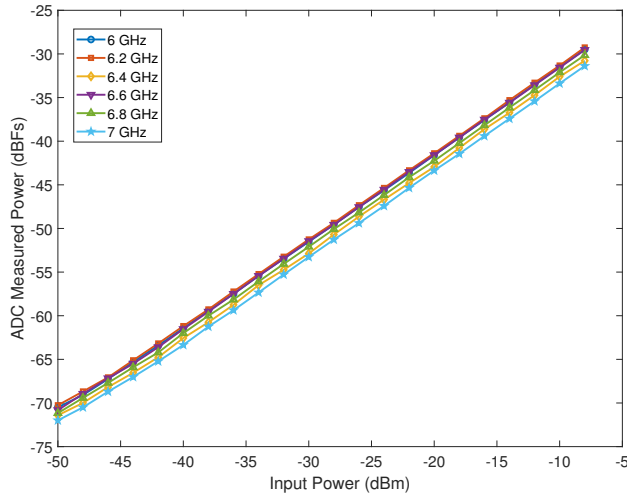

Fig. 10: Linearity of ADC from 6 GHz to 7 GHz

E. Comparison with State-of-the-Art

The proposed architecture has been implemented using Vivado 2020.2 and the configuration parameters of the experiment like the name of the experiment, pulse duration, time between pulses, etc., are being passed using SQ-CARS Python framework running on PYNQ v2.7. Table I show a comparison of the proposed SQ-CARS with the state-of-the-art Quantum control platforms.

The major challenges in the design of an integrated control and readout system are the scalability, direct synthesis and capture of microwave signals, on-board data processing and an unified user interface. The existing approaches, such as QubiC [27], QICK [28], and Yang et al. [29], do not provide a comprehensive end-to-end solution for addressing all the challenges aforementioned. These approaches rely on discrete components and custom clock distribution modules at the baseband level, resulting in increased complexity and challenges in unified control. QICK does not make use of the on-board NCOs for generation and capture of microwave frequencies. These works use external analog mixers to up-convert the baseband signals. Their additional utilization of external analog mixers for RF conversion introduces problems such as leakage, periodic calibration etc. Although QICK achieves inter-channel phase synchronization using a custom state machine, it fails to address the variable delay caused by the RF-SoC pipeline due to NCOs and FIFOs and does not support MTS. ICARUS-Q [30] performs MTS but again like QICK, it does not utilize the on-board NCOs. This limits on-the-fly synchronized control of frequency and phase across different channels and leaves the variable delay caused by the RFSoC pipeline unaddressed.

To overcome the limitations of the previous works, our work, SQ-CARS combines MTS with on-board NCOs to directly synthesize, control and capture phase-synchronized signals, demonstrating inter-channel synchronization among all data converters. We also developed a Python-based framework, by creating an abstraction layer that conceals the complexities of hardware implementation. It offers a user-friendly configuration interface to the physics user community, simplifying their interaction with the underlying hardware. Leveraging the existing RFdc Python code, we enhanced it by incorporating Multi-Tile Synchronization (MTS) functionality which enables on-the-fly synchronized control of frequency and phase of different channels. In addition, we developed Python classes for other hardware IPs within the programmable logic (PL), enabling seamless communication between the framework and hardware design modules. This streamlined approach enhances flexibility and ease of use for researchers and engineers working with the framework. To enable low-latency active feedback, on-board processing of acquired readout signals is essential. SQ-CARS incorporates on-board tunable filtering, rotation, and averaging blocks, which are absent in ICARUS-Q, Yang et al., and QubiC.

In Table II, we compare the resource utilization of SQ-CARS with existing platforms. Except ICARUS-Q, no other work has reported resource utilization. Hence, it is written 'Not Reported' indicating the absence of the report of the corresponding parameter in the literature. SQ-CARS achieves waveform playtime of 80us, a crucial experimental parameter, by leveraging on-board NCOs. This is significantly higher than ICARUS-Q, which only offers a playtime of 10us while utilizing similar BRAM resources per channel. Another main contribution lies in the on-board information processing pipeline employed by SQ-CARS, which is absent in ICARUS-Q. SQ-CARS consumes 46.38% of available LUTs and 42.96% of available BRAM resources which leaves ample

TABLE I: Comparison of SQ-CARS with various state-of-the-art Quantum Control System platforms

			Qubic [27]	QICK [28]	Yang et.al [29]	ICARUS-Q [30]	Presto [31]	SQ-CARS
Platform			VC707	ZCU111	Kintex 7	HTG-ZRF16	ZCU208/ ZCU216	ZCU111
Features	Sampling Rate (GSPS)	DAC	1.25	6.144	2	6.144	10	6.144
		ADC	1	4.096	1	1.96608	5	4.096
	Multi Channel Sync		No	using state machine, No MTS	using clock distribution	using MTS	using MTS	using MTS
	User Interface		Python	Python	NA	Python	Python	Python
	Microwave Synthesis & Capturing	Mix-Mode	No	No	No	Yes	Yes	Yes
		On-board NCOs	No	No	NA	No	Yes	Yes
	Information Processing		No	Filtering	No	No	Template Matching, Feedback	Tunable Filtering, Rotation, Averaging

TABLE II: Resource Utilization Comparison

	Device	Slice LUTs		Flip Flops		BRAMs		DSPs		Information Processing
		Available	Consumed	Available	Consumed	Available	Consumed	Available	Consumed	
Qubic [27]	XC7VX485T	303600	Not Reported	607200	Not Reported	1030	Not Reported	2800	Not Reported	No
QICK [28]	XCZU28DR	425280	Not Reported	850560	Not Reported	1080	Not Reported	4272	Not Reported	Filtering
Yang et.al. [29]	XCKU060	331680	Not Reported	663360	Not Reported	1080	Not Reported	2760	Not Reported	No
ICARUS-Q [30]	XCZU29DR	425280	212640 (50%)	850560	Not Reported	1080	810 (75%)	4272	6 (0.14%)	No
Presto [31]	XCZU48DR	425280	Not Reported	850560	Not Reported	1080	Not Reported	4272	Not Reported	Template Matching, Feedback
SQ-CARS	XCZU28DR	425280	197271 (46.38%)	850560	194580 (22.87%)	1080	464 (42.96%)	4272	2244 (52.52)	Filtering, Rotation, Averaging

room for expanding the system's capacity to accommodate a larger number of qubits, ensuring scalability.

The Power consumption of SQ-CARS has been generated using Vivado 2020.2 enabling the power optimisation feature. SQ-CARS consumes consumes 1.456 W of Static power and 20.353 W of Dynamic power, out of which only 15% is due to Logic. The rest of the power is majorly due to hard IPs like RF-DACs, RF-ADCs etc. These power numbers are unreported by any of the previous works.

In conclusion, SQ-CARS offers a comprehensive end-to-end solution, with the set of features including direct microwave synthesis and capture, long-term inter-channel phase synchronization, a scalable and flexible architecture, a digital pipeline for information processing, and a user-friendly unified interface, all on a single platform. These features, except for those found in Presto [31], enable SQ-CARS stand apart from the previous approaches.

IV. COHERENCE MEASUREMENTS OF A SUPERCONDUCTING QUBIT

To benchmark the performance of our technique for the direct generation of control signal pulses, we use a superconducting transmon qubit coupled to a 3D waveguide copper cavity. A fixed-frequency transmon [44] qubit was fabricated using standard lithography processes on an intrinsic silicon wafer. The whole qubit-cavity setup was mounted to a dilution refrigerator's base flange (10 mK) with various attenuators and filters on the input microwave lines. To benchmark the performance of RFSoC, we test its performance with two different transmon qubit devices named D1 and D2. A schematic of the full measurement setup is shown in Fig. 11. By carrying out cavity-qubit spectroscopy, we find the cavity frequency $\omega_c/2\pi = 5.995$ GHz, frequency of qubits D1 and D2 are approximately 4.2 and 4.7 GHz, respectively. We then characterize the performance of the RFSoC by generating the microwave pulses to carry out time domain characterization of qubits. To demonstrate the applicability of the directly synthesized control pulses, we carry out time-domain coherence measurements of the superconducting qubit. Generally, the time-domain experiments are done in two steps. First the calibration of microwave pulses is carried out. It may then followed by some basic characterization experiment.

A. Readout and control

To generate the microwave control pulses for the qubit, we use RFSoC DAC. As the qubit frequencies fall in the second-Nyquist zone, we use a wide-band Pasternack (PE 15A1002) amplifier to boost the signal amplitude to overcome the attenuation before sending it into the fridge. The baseband readout signal was generated using UHF lock-in amplifier from Zurich Instruments with AWG option. We used a home-built frequency up-converter and down-converter setup to generate the microwave readout pulse and to demodulate the readout signal coming from the fridge. We then determine the in-phase and quadrature components of the readout pulse and the subsequent qubit state determination. To address the issue of cross-platform triggering, we generated a trigger from ZCU111 analog output channel and used it to arm the sequencer on UHF-AWG. This straightforward handshaking allows us to correct any trigger latency by advancing the trigger generation on ZCU111. Moreover, it allows defining simple "loops" on ZCU111 so that the entire experiment can be controlled from the board. It is important to mention that, in principle, the fast ADC channels available on ZCU111 can be used for qubit readout and the entire control and readout part can be done on the same board.
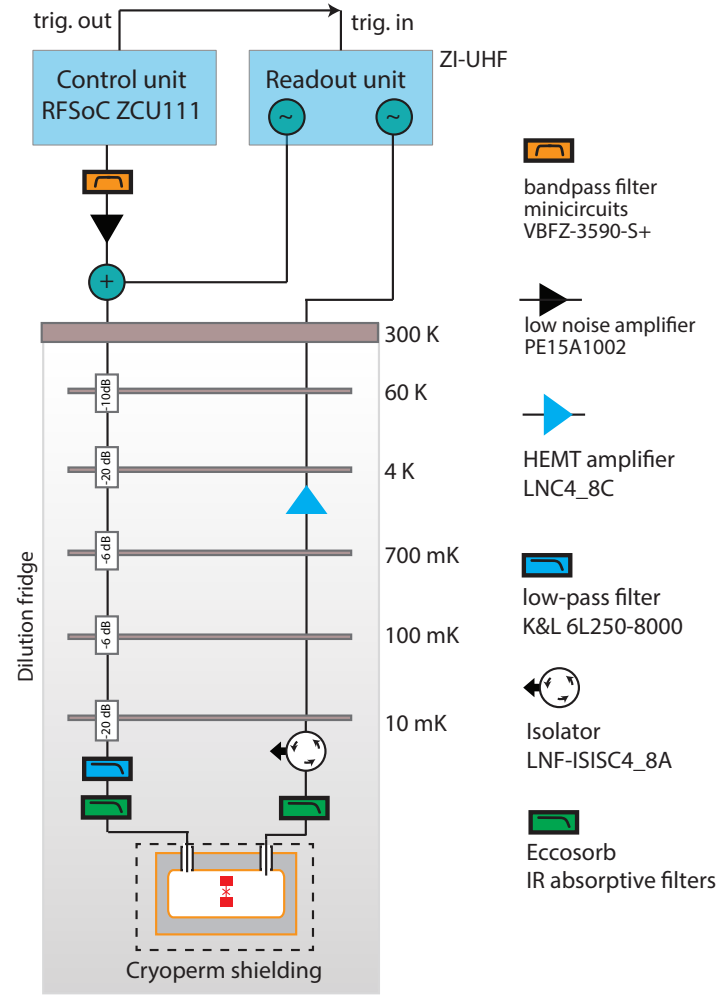
Fig. 11: Experimental measurement setup of superconducting qubit devices : The input lines are used for qubit state manipulation and to send cavity probe signals. First, the control signal was boosted up by using a Pasternack low-noise amplifier, and then it was added together with the readout signal. Both the signals then traveled to the device through the series of attenuators, tubular low pass filter, and homemade IR filter. The output signal from the cavity is then amplified by a HEMT amplifier mounted on the 4K stage and then the cavity quadratures were measured directly using the ZI-UHF lock-in amplifier.

B. Rabi spectroscopy

It involves measuring the excited state qubit population while varying the amplitude/length of the Gaussian pulse resonant with the qubit frequency, the measurement colloquially known as power/time Rabi measurement. We implemented this experiment on device D1, by inserting a rectangular pulse with a variable number of samples at the center of the X_π Gaussian pulse. We use 260 ns long Gaussian pulses for these measurements with a standard deviation of 65 ns. The duration of the qubit control pulse can be adjusted by adjusting the number of sample points in the rectangular pulse. The top part of Fig. 12(a) panel shows the pulse sequence. Any trigger latency between ZCU11 and the readout setup (UHF) is adjusted by adjusting the trigger delay. The bottom panel shows the coherent time oscillations as the duration of the rectangular portion of the control pulse is increased. The measurement of Rabi oscillations in time and power domain allows to calibrate the microwave pulses necessary for qubit operation.

C. Relaxation time measurement

After getting the calibrated X_π from the above measurement, it is straightforward to carry out energy relaxation time measurement. The setup is same as the Rabi oscillation measurement setup with a minor change in the pulse sequence. Here we first send a calibrated X_π pulse and then wait for time t, before performing a readout as shown in Fig. 12(b). Due to on-the-fly configuration of the delay time, the entire measurement can be controlled by ZCU111 unit. For a given wait time, we generate 50,000 X_π pulses at a repetition rate of 5 kHz and determine the excited state population. Fig. 12(c) shows the result from T_1 measurements of D1 device. The value of T_1 was extracted by fitting the following equation
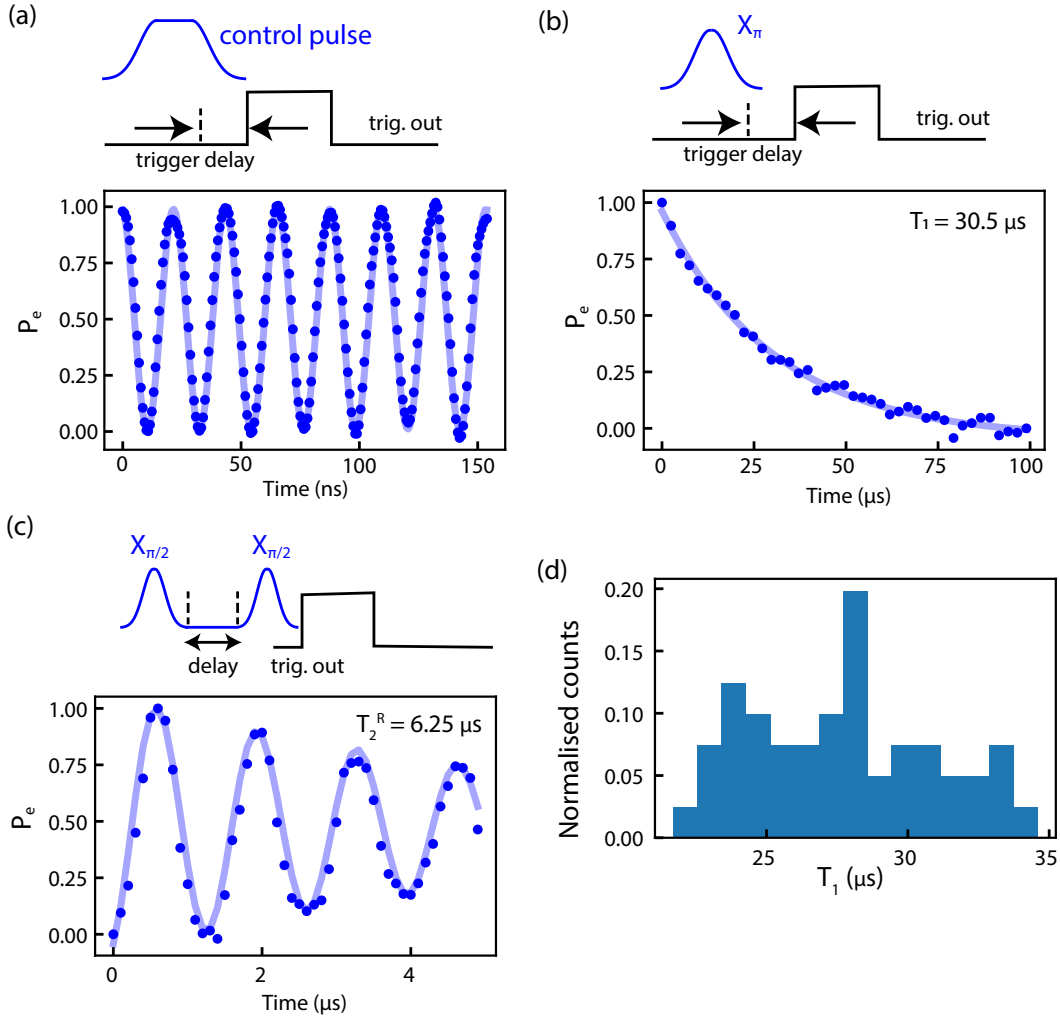
Fig. 12: Time-domain control of a transmon qubit (D1) using ZCU111. Each measured data point in these panels is a result from 50,000 shots. (a) Rabi oscillations between the qubit's ground $|0\rangle$ and first excited state $|1\rangle$ as a the duration of control pulse is varied. (b) Measurement of the energy relaxation time T_1. The blue line indicates an exponential fit yielding $T_1 = 30.5$ μs. (c) Measurement of Ramsey dephasing time (T_2^R). The top part of the panel shows the pulse sequence. From the fit indicated by solid line, we extract Ramsey dephasing time $T_2^R = 6.25$ μs and detuning $\Delta = 0.8$ MHz. (d) Histogram showing the statistical variation between 47 T_1 measurements taken over a period of 24 hours.

$A + Be^{-t/T_1}$. The results from the T_1 experiment along with the fits, yielded a T_1 of approximately 30.5 μs.

D. Ramsey spectroscopy

Next, we carry out the Ramsey experiment, a measurement of dephasing rate. The protocol requires the generation of two $X_{\pi/2}$ pulses with a variable time-delay between them, and it is followed by the qubit measurement. A schematic of the pulse sequence and result from the Ramsey fringe experiment is shown in Fig. 12(c). As the experiment can be completely controlled by the ZCU111, the procedure remains similar to the T_1 measurements. Here the trigger for the readout unit is aligned with the end of the second $X_{\pi/2}$ pulse and a delay between the two $X_{\pi/2}$ pulses is varied. By fitting the measured curve of the Ramsey fringes using $A + B\cos(2\pi\Delta t + \phi)e^{-t/T_2^R}$, one can obtain the values for T_2^R and detuning Δ, respectively. In this context, the detuning

represents the deviation of the microwave pulse frequency from the qubit frequency. For the device D1, we measured T_2^R of 6.25 μs and detuning of 0.8 MHz. It is important to point out here that since both the $X_{\pi/2}$ pulses are generated by the internal digital IQ-mixer and common NCO, a well-defined phase relation holds between the two pulses. Such functionality makes it very easy to implement where X- and Y- qubit rotations can be made by controlling the signals on I and Q inputs of the digital mixer.

E. Standard AWG vs RFSoC

To compare the performance of RFSoC and standard test and measurement instruments, we compare the results of qubit characterization, T_1 and T_2^R, on the same device during the same cooldown run. The conventional measurements were carried out by generating the drive pulses using the single-sideband modulation technique and a vector signal generator.
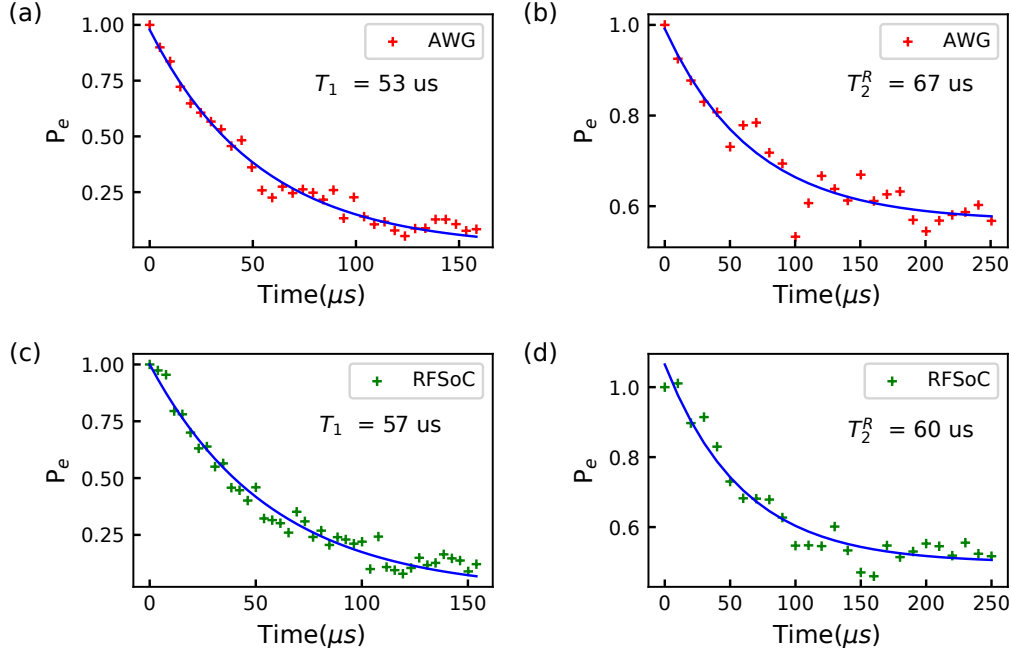
Fig. 13: Comparison of the measurements of T_1 and T_2^R from a transmon qubit (D2) using traditional AWG and RFSoC. Panel (a, b) show T_1 and T_2^R measurement obtained from traditional AWG. The solid line indicates an exponential fit yielding $T_1 = 53$ μs and $T_2^R = 67$ μs. Panel (c, d) show the measurement of the T_1 and T_2^R using RFSoC DAC. The solid line indicates an exponential fit yielding $T_1 = 57$ μs and $T_2^R = 60$ μs. For Ramsey measurements, we drive the qubit on resonance and thus it does not show any beating oscillations. Each measured data point in these panels is a result from 50,000 shots.

The baseband drive signals were generated using a commercially available AWG. In the Fig. 13, we show the comparative results from another device (D2) by measuring it using general test and measurement equipment and RFSOC. We do not observe any statistically significant difference between the results. Given the ease of the measurements, as no periodic calibration for sideband suppression is necessary, we also repeated the T_1 measurements of device D1 over 24 hours and recorded statistical variations in the T_1 measurements. Fig. 12 (d) shows a histogram of 47 T_1 measurements from the same device.

V. CONCLUSION

In this work, we have showcased the performance and capabilities of our integrated framework SQ-CARS, designed to cater to the demanding requirements of superconducting quantum systems. Our framework not only allows for direct generation and capture of microwave signals, but also supports real-time information processing, providing capability for active feedback within the system. What sets our framework apart is its scalability, configurability, and user-friendly nature, all achieved while keeping costs low. By providing an accessible room temperature control system, encompassing the capabilities and features detailed in this paper which is freely available on GitHub, invites further collaboration and research in the field. Furthermore, the versatility of our techniques allows for easy expansion to accommodate a larger number of channels, opening up possibilities for experimentation and

research. While our primary focus has been on the superconducting quantum devices community, the implications of our work extend beyond this domain. Researchers working with quantum computing using semiconductors, Nitrogen Vacancy (NV) centers, and trapped-ion systems will also find value and relevance in our platform.

Looking ahead, our future work will focus on further enhancements to the proposed system. To reduce overall duty cycle, we would be incorporating on-board qubit state detection and active feedback mechanisms to send corrective pulses without the intervention of the user. We will also work on implementing the capability of direct play of longer waveforms from faster DDRs, faster streaming of data by integrating Ethernet to PL.

Unlike classical processors, quantum circuits face a substantial input-output bottleneck. Each qubit in a quantum computer is individually governed by external circuitry, which introduces both noise and heat to the qubit system [45]. Currently, the control of superconducting qubits relies on the application of pulsed microwave signals. The generation and routing of these control pulses involve considerable experimental overhead, encompassing both room-temperature and cryogenic electronics hardware. This includes, but is not confined to, coherent microwave waveform generators, amplifiers, as well as coaxial lines and signal conditioning elements required to transmit these signals into the low-temperature experimental environment. While the brute-force scaling of existing technology may suffice for moderately-sized superconducting qubit systems,

the control of large-scale systems necessitates fundamentally novel approaches [46]. In this scenario, a noteworthy concept to consider is the integration of all crucial features onto a single chip, enabling the possibility of relocating the RFSoC (Radio Frequency System on a Chip) device from room temperature to the dilution refrigerator. This relocation would result in improved signal-to-noise ratio (SNR) and a reduction in the number of connections to room temperature electronics. Such a transition would offer an implementation of cryogenic control electronics for larger superconducting Qubit systems.

VI. Acknowledgment

References

[1] T. D. Ladd, F. Jelezko, R. Laflamme, Y. Nakamura, C. Monroe, and J. L. O'Brien, "Quantum computers," *nature*, vol. 464, no. 7285, pp. 45–53, Mar. 2010.

[2] S.-L. Zhu, C. Monroe, and L.-M. Duan, "Trapped Ion Quantum Computation with Transverse Phonon Modes," *Physical Review Letters*, vol. 97, no. 5, Aug. 2006. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.97.050505

[3] N. W. Hendrickx, W. I. L. Lawrie, M. Russ, F. van Riggelen, S. L. de Snoo, R. N. Schouten, A. Sammak, G. Scappucci, and M. Veldhorst, "A four-qubit germanium quantum processor," *Nature*, vol. 591, no. 7851, Mar. 2021. [Online]. Available: https://www.nature.com/articles/s41586-021-03332-6

[4] G. D. Fuchs, V. V. Dobrovitski, D. M. Toyli, F. J. Heremans, C. D. Weis, T. Schenkel, and D. D. Awschalom, "Excited-state spin coherence of a single nitrogen–vacancy centre in diamond," *Nature Physics*, vol. 6, no. 9, Sep. 2010. [Online]. Available: https://www.nature.com/articles/nphys1716

[5] M. H. Devoret and R. J. Schoelkopf, "Superconducting circuits for quantum information: An outlook," *Science*, vol. 339, no. 6124, pp. 1169–1174, mar 2013.

[6] M. Kjaergaard, M. E. Schwartz, J. Braumüller, P. Krantz, J. I.-J. Wang, S. Gustavsson, and W. D. Oliver, "Superconducting Qubits: Current State of Play," *Annual Review of Condensed Matter Physics*, vol. 11, no. 1, pp. 369–395, 2020. [Online]. Available: https://doi.org/10.1146/annurev-conmatphys-031119-050605

[7] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. S. L. Brandao, D. A. Buell, B. Burkett, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, A. Dunsworth, E. Farhi, B. Foxen, A. Fowler, C. Gidney, M. Giustina, R. Graff, K. Guerin, S. Habegger, M. P. Harrigan, M. J. Hartmann, A. Ho, M. Hoffmann, T. Huang, T. S. Humble, S. V. Isakov, E. Jeffrey, Z. Jiang, D. Kafri, K. Kechedzhi, J. Kelly, P. V. Klimov, S. Knysh, A. Korotkov, F. Kostritsa, D. Landhuis, M. Lindmark, E. Lucero, D. Lyakh, S. Mandrà, J. R. McClean, M. McEwen, A. Megrant, X. Mi, K. Michielsen, M. Mohseni, J. Mutus, O. Naaman, M. Neeley, C. Neill, M. Y. Niu, E. Ostby, A. Petukhov, J. C. Platt, C. Quintana, E. G. Rieffel, P. Roushan, N. C. Rubin, D. Sank, K. J. Satzinger, V. Smelyanskiy, K. J. Sung, M. D. Trevithick, A. Vainsencher, B. Villalonga, T. White, Z. J. Yao, P. Yeh, A. Zalcman, H. Neven, and J. M. Martinis, "Quantum supremacy using a programmable superconducting processor," *Nature*, vol. 574, no. 7779, pp. 505–510, Oct. 2019, number: 7779 Publisher: Nature Publishing Group.

[8] M. Gong, S. Wang, C. Zha, M.-C. Chen, H.-L. Huang, Y. Wu, Q. Zhu, Y. Zhao, S. Li, S. Guo, H. Qian, Y. Ye, F. Chen, C. Ying, J. Yu, D. Fan, D. Wu, H. Su, H. Deng, H. Rong, K. Zhang, S. Cao, J. Lin, Y. Xu, L. Sun, C. Guo, N. Li, F. Liang, V. M. Bastidas, K. Nemoto, W. J. Munro, Y.-H. Huo, C.-Y. Lu, C.-Z. Peng, X. Zhu, and J.-W. Pan, "Quantum walks on a programmable two-dimensional 62-qubit superconducting processor," *Science*, May 2021, publisher: American Association for the Advancement of Science. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.abg7812

[9] J. Koch, T. M. Yu, J. Gambetta, A. A. Houck, D. I. Schuster, J. Majer, A. Blais, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf, "Charge-insensitive qubit design derived from the Cooper pair box," *Physical Review A*, vol. 76, no. 4, p. 042319, Oct. 2007. [Online]. Available: http://link.aps.org/doi/10.1103/PhysRevA.76.042319

[10] P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, "A quantum engineer's guide to superconducting qubits," *Applied physics reviews*, vol. 6, no. 2, p. 021318, jun 2019.

[11] L. B. Nguyen, G. Koolstra, Y. Kim, A. Morvan, T. Chistolini, S. Singh, K. N. Nesterov, C. Jünger, L. Chen, Z. Pedramrazi, B. K. Mitchell, J. M. Kreikebaum, S. Puri, D. I. Santiago, and I. Siddiqi, "Blueprint for a high-performance fluxonium quantum processor," *PRX Quantum*, vol. 3, p. 037001, Aug 2022. [Online]. Available: https://link.aps.org/doi/10.1103/PRXQuantum.3.037001

[12] S. W. Jolin, R. Borgani, M. O. Tholén, D. Forchheimer, and D. B. Haviland, "Calibration of mixer amplitude and phase imbalance in superconducting circuits," *Review of Scientific Instruments*, vol. 91, no. 12, p. 124707, Dec. 2020, publisher: American Institute of Physics.

[13] H. Ball, W. D. Oliver, and M. J. Biercuk, "The role of master clock stability in quantum information processing," *npj Quantum Information*, vol. 2, no. 1, pp. 1–8, 2016.

[14] S. Asaad, C. Dickel, N. K. Langford, S. Poletto, A. Bruno, M. A. Rol, D. Deurloo, and L. DiCarlo, "Independent, extensible control of same-frequency superconducting qubits by selective broadcasting," *npj Quantum Information*, vol. 2, no. 1, Aug. 2016. [Online]. Available: https://www.nature.com/articles/npjqi201629

[15] J. Heinsoo, C. K. Andersen, A. Remm, S. Krinner, T. Walter, Y. Salathé, S. Gasparinetti, J.-C. Besse, A. Potočnik, A. Wallraff, and C. Eichler, "Rapid High-fidelity Multiplexed Readout of Superconducting Qubits," *Physical Review Applied*, vol. 10, no. 3, p. 034040, Sep. 2018. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevApplied.10.034040

[16] R. Vijay, C. Macklin, D. H. Slichter, S. J. Weber, K. W. Murch, R. Naik, A. N. Korotkov, and I. Siddiqi, "Stabilizing Rabi oscillations in a superconducting qubit using quantum feedback," *Nature*, vol. 490, no. 7418, Oct. 2012. [Online]. Available: https://www.nature.com/articles/nature11505

[17] N. Ofek, A. Petrenko, R. Heeres, P. Reinhold, Z. Leghtas, B. Vlastakis, Y. Liu, L. Frunzio, S. M. Girvin, L. Jiang, M. Mirrahimi, M. H. Devoret, and R. J. Schoelkopf, "Extending the lifetime of a quantum bit with error correction in superconducting circuits," *Nature*, vol. 536, no. 7617, Aug. 2016. [Online]. Available: https://www.nature.com/articles/nature18949

[18] P. Campagne-Ibarcq, S. Jezouin, N. Cottet, P. Six, L. Bretheau, F. Mallet, A. Sarlette, P. Rouchon, and B. Huard, "Using Spontaneous Emission of a Qubit as a Resource for Feedback Control," *Physical Review Letters*, vol. 117, no. 6, Aug. 2016.

[19] Q. machines, "Realize the Quantum Possibilities of Tomorrow," https://www.quantum-machines.com, 2022, [Online; accessed 20-Feb-2022].

[20] Z. Instruments, "Zurich Instruments Website," https://www.zhinst.com/americas/en, 2022, [Online; accessed 20-Feb-2022].

[21] Keysight, "Keysight Quantum Solutions," https://www.keysight.com/us/en/solutions/emerging-technologies/quantum-solutions.html, 2022, [Online; accessed 20-Feb-2022].

[22] S. Pulipati, V. Ariyarathna, A. Dhananjay, M. E. Eltayeb, M. Mezzavilla, J. M. Jornet, S. Mandal, S. Bhardwaj, and A. Madanayake, "Xilinx rf-soc-based digital multi-beam array processors for 28/60 ghz wireless testbeds," in *2020 Moratuwa Engineering Research Conference (MER-Con)*. IEEE, 2020, pp. 254–259.

[23] J. Goldsmith, C. Ramsay, D. Northcote, K. W. Barlee, L. H. Crockett, and R. W. Stewart, "Control and visualisation of a software defined radio system on the xilinx rfsoc platform using the pynq framework," *IEEE Access*, vol. 8, pp. 129 012–129 031, 2020.

[24] F. Michalak, W. Zabołotny, Ł. Podkalicki, M. Malanowski, M. Piasecki, and K. Kulpa, "Universal rfsoc-based signal recorder for radar applications," in *2022 23rd International Radar Symposium (IRS)*. IEEE, 2022, pp. 136–140.

[25] Xilinx, "Zynq UltraScale+ RFSoC," https://www.xilinx.com/products/silicon-devices/soc/rfsoc.html, 2008, [Online; accessed 22-Jan-2022].

This article has been accepted for publication in IEEE Transactions on Instrumentation and Measurement. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TIM.2023.3305656

JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2021 15

[26] ——, "ZCU111 Evaluation Board User Guide," https://www.xilinx.com/content/dam/xilinx/support/documentation/boards_and_kits/zcu111/ug1271-zcu111-eval-bd.pdf, 2021, [Online; accessed 22-Jan-2023].

[27] Y. Xu, G. Huang, J. Balewski, R. Naik, A. Morvan, B. Mitchell, K. Nowrouzi, D. I. Santiago, and I. Siddiqi, "Qubic: An open-source fpga-based control and measurement system for superconducting quantum information processors," *IEEE Transactions on Quantum Engineering*, vol. 2, pp. 1–11, 2021.

[28] L. Stefanazzi, K. Treptow, N. Wilcer, C. Stoughton, C. Bradford, S. Uemura, S. Zorzetti, S. Montella, G. Cancelo, S. Sussman *et al.*, "The qick (quantum instrumentation control kit): Readout and control for qubits and detectors," *Review of Scientific Instruments*, vol. 93, no. 4, p. 044709, 2022.

[29] Y. Yang, Z. Shen, X. Zhu, Z. Wang, G. Zhang, J. Zhou, X. Jiang, C. Deng, and S. Liu, "Fpga-based electronic system for the control and readout of superconducting quantum processors," *Review of Scientific Instruments*, vol. 93, no. 7, p. 074701, 2022.

[30] K. H. Park, Y. S. Yap, Y. P. Tan, C. Hufnagel, L. H. Nguyen, K. H. Lau, P. Bore, S. Efthymiou, S. Carrazza, R. P. Budoyo *et al.*, "Icarus-q: Integrated control and readout unit for scalable quantum processors," *Review of Scientific Instruments*, vol. 93, no. 10, p. 104704, 2022.

[31] M. O. Tholén, R. Borgani, G. R. Di Carlo, A. Bengtsson, C. Križan, M. Kudra, G. Tancredi, J. Bylander, P. Delsing, S. Gasparinetti *et al.*, "Measurement and control of a superconducting quantum processor with a fully integrated radio-frequency system on a chip," *Review of Scientific Instruments*, vol. 93, no. 10, p. 104711, 2022.

[32] "SQ-CARS, https://github.com/NeuRonICS-Lab/Quantum-Control-Electronics," [Online; accessed 02-Feb-2023].

[33] "" pynq: Python productivity for adaptive computing platforms," https://pynq.readthedocs.io/en/latest/ (2023)."

[34] T. J. Rouphael, "Chapter 7 - uniform sampling of signals and automatic gain control," in *RF and Digital Signal Processing for Software-Defined Radio*, T. J. Rouphael, Ed. Burlington: Newnes, 2009, pp. 199–234. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780750682107000072

[35] R. Baker, *CMOS Mixed-Signal Circuit Design, Second Edition*. John Wiley & Sons, 12 2008.

[36] S. R. Systems, "SG390 Series RF Signal Generators," https://www.thinksrs.com/downloads/pdfs/manuals/SG390m.pdf, 2017, [Online; accessed 22-Jan-2022].

[37] Tektronix, "TSG4100A Series RF Signal Generators," https://download.tek.com/manual/TSG4100A-Series-RF-Signal-Generator-User-Manual-071315002.pdf, 2017, [Online; accessed 15-Feb-2022].

[38] S. Hound, "VSG60A Vector Signal Generators," https://signalhound.com/sigdownloads/VSG60/VSG60A-Product-Manual.pdf, 2020, [Online; accessed 20-Feb-2022].

[39] K. Technologies, "N5166B CXG, RF Vector Signal Generator," https://www.keysight.com/in/en/assets/7018-06626/data-sheets/5992-3959.pdf, 2021, [Online; accessed 20-Feb-2022].

[40] Z. Instruments, "SHFQA Quantum Analyzer," https://docs.zhinst.com/pdf/ziSHFQA_UserManual.pdf, 2022, [Online; accessed 22-Aug-2022].

[41] T. Collins, R. Getz, D. pu, and A. M. Wyglinski, *Software-Defined Radio for Engineers*. Artech House, 2018.

[42] A. L. Borja, J. Carbonell, V. E. Boria, J. Cascon, and D. Lippens, "A 2% Bandwidth C-Band Filter Using Cascaded Split Ring Resonators," *IEEE Antennas and Wireless Propagation Letters*, vol. 9, pp. 256–259, 2010, conference Name: IEEE Antennas and Wireless Propagation Letters.

[43] A. Devices, "HMC892ALP5E, Tunable Band-pass Filter," https://www.analog.com/en/products/hmc892a.html?doc=HMC892ALP5E.pdf#product-overview, 2018, [Online; accessed 18-Feb-2022].

[44] H. Paik, D. I. Schuster, L. S. Bishop, G. Kirchmair, G. Catelani, A. P. Sears, B. R. Johnson, M. J. Reagor, L. Frunzio, L. I. Glazman, S. M. Girvin, M. H. Devoret, and R. J. Schoelkopf, "Observation of High Coherence in Josephson Junction Qubits Measured in a Three-Dimensional Circuit QED Architecture," *Physical Review Letters*, vol. 107, no. 24, p. 240501, 2011.

[45] S. Krinner, S. Storz, P. Kurpiers, P. Magnard, J. Heinsoo, R. Keller, J. Luetolf, C. Eichler, and A. Wallraff, "Engineering cryogenic setups for 100-qubit scale superconducting circuit systems," *EPJ Quantum Technology*, vol. 6, no. 1, p. 2, 2019.

[46] D. Reilly, "Challenges in scaling-up the control interface of a quantum computer," in *2019 IEEE International Electron Devices Meeting (IEDM)*. IEEE, 2019, pp. 31–7.
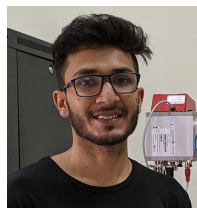
Ujjawal Singhal received the integrated B.S/M.S degree in Physics from Indian Institute of Science Education and Research, Mohali, India. He is currently pursuing the Ph.D. degree in Physics from Indian Institute of Science, Bangalore. His research interests include superconducting quantum devices.
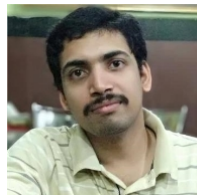
Shantharam Kalipatnapu received his Ph.D degree in Electronics and Electrical Communication Engineering from Indian Institute of Technology, Kharagpur, India, in 2021. He is currently associated as a postdoctoral researcher in NeuRonICs Lab in Indian Institute of Science, Bangalore, India. His research interests include Digital VLSI architectures for error correction codes, signal processing and control electronics for Quantum computing systems.

Pradeep Kumar Gautam received his B.Tech. degree from NIT-Jaipur in Electonics and Communication Engineering. He is currently associated with DRDO and pursuing his Ph.D. degree in Department of Electonic Systems Engineering from Indian Institute of Science, Bangalore. His research interests include System on Chip and Processor design, machine learning on edge devices.

Sourav Majumder received the B.Sc. degrees in Physics from Ramakrishna Mission Residential College, Narendrapur, India. He is currently pursuing the Ph.D. degree in Physics from Indian Institute of Science, Bengaluru, India. His research interests include superconducting quantum devices.

Vaibhav Venkata Lakshmi Pabbisetty received his B.Tech. degree from Jawaharlal Nehru Technological University, Kakinada. He is currently pursuing M.Tech in Microelectronics & VLSI from Indian Institute of Science, Bangalore. His research interests include Digital VLSI architectures for Signal processing and Quantum computing.

Srivatsava Jandhyala received the Ph.D. degree in electrical engineering from the Indian Institute of Science, Bangalore, India, in 2013. His research interests include architectures and circuits for application specific hardware accelerators, compact models for emerging electronic devices and control electronics for Quantum computing systems.

Vibhor Singh received his M.S degree in Physics from Indian Institute of Technology, Roorkee (INDIA), and Ph.D degree from Tata Institute of Fundamental Research, Mumbai (INDIA). He held a post-doctoral position at Technical University, Delft (The Netherlands) for three years. Subsequently, he joined the Department of Physics at the Indian Institute of Science, Bangalore (INDIA). His research interest lies in superconducting quantum devices, cavity optomechanical devices and hybrid qubit devices, and quantum computing.

Chetan Singh Thakur received the Ph.D. degree in neuromorphic engineering from the MARCS Research Institute, Western Sydney University (WSU), in 2016. He then worked as a Research Fellow at Johns Hopkins University. He has extensive industrial experience for six years with Texas Instruments, Singapore, as a Senior Integrated Circuit Design Engineer, designing IPs for mobile processors. He is currently an Associate Professor at the Indian Institute of Science (IISc), Bengaluru. He is also an Adjunct Faculty at the International Center for Neuromorphic Systems, WSU, Australia. His research expertise includes neuromorphic computing, mixed-signal VLSI systems, computational neuroscience, probabilistic signal processing, and machine learning. His research interest includes understanding the signal processing aspects of the brain and apply those to build novel intelligent systems. He was a recipient of several awards, such as the Young Investigator Award from Pratiksha Trust, the Early Career Research Award by Science and Engineering Research Board, India, and the Inspire Faculty Award by the Department of Science and Technology, India.