ICARUS-Q: A scalable RFSoC-based control system for superconducting quantum computers

Kun Hee Park,1,∗ Yung Szen Yap,2,1,† Yuanzheng Paul Tan,3,1 Christoph Hufnagel,1 Long Hoang Nguyen,3 Karn Hwa Lau,4 Stavros Efthymiou,5 Stefano Carrazza,6,7,5 Rangga P. Budoyo,1 and Rainer Dumke1,3,†
1 Centre for Quantum Technologies, National University of Singapore, 3 Science Drive 2, Singapore 117543, Singapore
2 Faculty of Science and Centre for Sustainable Nanomaterials (CSNano), Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia
3 Division of Physics and Applied Physics, School of Physical and Mathematical Sciences, Nanyang Technological University, 21 Nanyang Link, Singapore 637371, Singapore
4 Advinno Technologies Pte. Ltd., 22, Sin Ming Lane, #05-75 Midview City 573969, Singapore
5 Quantum Research Centre, Technology Innovation Institute, Abu Dhabi, UAE
6 TIF Lab, Dipartimento di Fisica, Università degli Studi di Milano and INFN Sezione di Milano, Milan, Italy
7 Theoretical Physics Department, CERN, CH-1211 Geneva 23, Switzerland

Abstract: We present a control and measurement setup for superconducting qubits based on Xilinx 16-channel radio frequency system on chip (RFSoC) device. The proposed setup consists of four parts: multiple RFSoC FPGA boards, a setup to synchronise every DAC and ADC channel across multiple boards, a low-noise DC current supply for qubit biasing and cloud access for remotely performing experiments. We also design the setup to be free of physical mixers. The FPGA boards directly generate microwave pulses using sixteen DAC channels up to the third Nyquist zone which are directly sampled by its eight ADC channels between the fifth and the ninth zones.

I. INTRODUCTION

Superconducting qubits in the dilution refrigerator are controlled and measured with room temperature electronics. A typical superconducting qubit is designed with its transition energy in the order of a few GHz, and requires arbitrary and precise microwave generation and detection for control and measurement. As the number of qubits increases, the number of microwave channels required increases linearly. Therefore, designing a qubit control system that is scalable, compact and cost-effective, while maintaining its precision, speed, and features, is imperative.

Apart from the microwave circuits for frequency up/down-conversion, a basic qubit control system consists of digital to analog converters (DACs), and analog to digital converter (ADCs) and stable current sources; the DACs generate the microwave pulses that travel into the fridge, the ADCs digitise the analog signals that travel out from the fridge, and the current source biases the qubits. Some of the earlier microwave control systems for electron spin and superconducting qubits [1], relied on benchtop arbitrary waveform generators (AWGs) for the microwave generation [2–7]. The recent trends however are favoring field programmable gate array (FPGA) [8–15] for their higher number of channels (i.e. cost per channel), versatility, and the form factor. Typically, two DAC channels are required, per qubit, for qubit driving; additionally, one DAC channel is shared among five or more qubits for frequency-multiplexed readout schemes [16–18].

The latest family of FPGA by Xilinx known as the Zynq UltraScale+ Radio Frequency System-on-Chip (RFSoC) [19] hosts a wide-variety of features that are advantages for qubit control and measurement. To the best of our knowledge, this family of devices feature the highest number of independent DAC and ADC channels within a single chip with high sampling rates and is equipped auto-synchronisation between channels. The device also has digital up/down converters using internal complex mixers and a 48-bit numerically-controlled oscillator (NCO), and two processors. These features which are available at a fraction of the cost and size to that of other commercial off-the-shelf devices, makes the RFSoC particularly enticing for applications such as radar [20], communications [21] and quantum computing [22–24].

First announced in late 2018, the RFSoC has three generations of devices where only the first two are available at the time of writing [25]. Within the first generation of RFSoC devices, the two top devices are XCZU28DR and XCZU29DR. There primary differences between them are: the number of channels (8 versus 16 channels of DACs and ADCs) and the maximum sampling rate of the ADCs (4.096 GS/s versus 2.058 GS/s, respectively).

A single FPGA board (ZCU111 by Xilinx) that is populated with the XCZU28DR (equipped with eight 6.554 GS/s DAC channels and eight 4.096 GS/s ADC channels) has been used as a control and measurement system for superconducting quantum computers [22–24]. Here, we develop a scalable setup based on multiple synchronised XCZU29DR FPGA boards, where each board features sixteen 6.554 GS/s DAC channels and sixteen 2.058 GS/s ADC channels per board and operates without physical IQ mixers.

∗ These authors contributed equally to the work.
† rduke@ntu.edu.sg
II. IMPLEMENTATION

The developed setup in Fig. 1 consists of several parts. The FPGA board, codename Icarus-Q (Integrated Control And Readout Unit for Scalable Quantum processors), runs with an embedded Linux kernel that receives commands and transfers data in/out of the board via Ethernet (Sec. II A). In our approach, an alias in the higher Nyquist zone [13] of the DAC signal is used to address the GHz-range qubit transition and direct sampling of the high-frequency signals by the ADCs at a lower sampling rate. Multiple FPGA boards are synchronised with each other using a master oscillator and trigger signals that are synchronised to the same master oscillator (Sec. II D). For tunable superconducting qubits, the Josephson junction of the qubit is replaced by a DC-SQUID loop. This allows the qubit to be biased with a magnetic flux, which is coupled into the loop from a current carrying wire close to the loop. To support this, low-noise DC sources are integrated into the setup for biasing the qubits (Sec. II E). The FPGA boards, trigger and current sources are connected to a PC, which runs a worker program that communicates with a cloud server to remotely run experiments (Sec. II F).

The FPGA board used in this work is available commercially (HTG-ZRF16, HiTech Global) and is populated with one unit of XCZU29DR RFSoC device (see Fig. 2 for photo and Tab. 1) that comes with sixteen 14-bit DAC (6.554 GS/s) and sixteen 12-bit ADC (2.058 GS/s) differential pins. On the FPGA board, these are converted to single-ended signals using baluns (which support frequencies from 10 MHz to 8 GHz) and SSMC ports (support frequencies up to 12.4 GHz) for all of the DAC and ADC channels. Despite a rated sampling rate of 6.554 GS/s for the DACs and 2.096 GHz for the ADCs, our setup reaches a limit of 6.144 GS/s for the DACs and 1.96608 GS/s for the ADCs. This limit occurs because RFSoC FPGA is linked to the master clock and the sampling rates are multiples of the master clock (122.88 MHz).

In our setup, more DAC channels are needed than ADC channels. To allocate more of the limited FPGA block RAM memories [26] to the DACs, we reduced the number of active ADC channels to eight. The FPGA board also has both fixed memories (4 GB) and SODIMM DDR4 modules (up to 16 GB), which in the current design, are not used for direct streaming via the DAC and ADC channels.

A. FPGA Logic

This section describes the FPGA logic that enables data movement between the PC and the quantum processor (see Fig. 3). The conversions between data and signals are performed by the ADCs and DACs, which are activated by the external triggers. The HTG-ZRF16 contains a two-stage clock distribution logic for phase synchronisation of the ADCs and DACs. The FPGA also encompasses the Ethernet, microSD card and DDR memories on the HTG-ZRF16 board.

1. FPGA Logic for DAC waveforms generation

The arbitrary waveforms are generated using RFSoC DAC (see Fig. 4). The sixteen DAC channels are powered by four DAC ‘tiles’ inside the RFSoC chip. In order to ensure synchronous output of all DACs within a single board, multi-tiles synchronisation logic inside the RFSoC is utilised to calibrate the ‘tile-to-tile’ time skew.

The DAC data flow starts from loading the waveform of interests into the programmable logic (PL) DDR memory. PL DDR memory is a physical SODIMM memory module connected to the PL-FPGA. The waveforms data are then moved into AXI Stream FIFO (First-In, First-Out) of each DAC channels. This is essential to allow the DAC playback to start simultaneously for all channels. However, due to the memory capacity limit of the internal block RAMs, individual AXI Stream FIFO can store up to 65,536 samples of DAC waveform of each channel.

The DAC waveform playback supports loopback function. When enabled, it allows the waveforms to be reloaded into AXI Stream FIFO without the need of reloading the waveform data from a host computer, thus reducing the overhead time of re-arming the DAC for next DAC playback.

Once the waveforms data are loaded into AXI Stream FIFO of each channel, the system will wait for an external trigger event from the external control logic before starting the DAC waveform playback. The trigger signal from external control logic applies to all DAC channels so the output can be streamed out via the SSMC connectors simultaneously. The external control logic also supports the waveform data swap. When enabled, the waveforms of upper eight channels can be swapped to the lower eight
FIG. 1. Overall circuit diagram. Several units of FPGA boards, each with an RFSoC device (XCZU29DR by Xilinx), are used to directly generate and sample microwave signals in the GHz frequencies using its higher Nyquist zones (with the help of filters). The boards are synchronised to a master reference clock (VHF Citrine Gold, Wenzel) and triggers from a TTL pulse generator (PulseBlaster PB24-100-4K, Spincore). The triggers are also synchronised to the same clock using a D flip-flop circuit. To bias the superconducting qubits, low noise current sources are merged with microwave pulses using bias-tees, located at the mixing chamber stage. The biasing for the flip-flop clock input has been omitted for clarity. The worker PC is connected to the trigger source via PCI, and to the current sources via USB. The RFSoC boards are controlled from the PC via SSH over Ethernet through the network switch.

channels to support more advanced pulse sequences (see Sec. II C for more information).

2. FPGA Logic for ADC waveforms acquisition

In our firmware, the waveform acquisition system is powered by eight ADC channels of the RFSoC (see Fig. 5). The analog input is fed into RFSoC ADC via SSMC connectors. ADC digitise the incoming waveform continuously, but is not streamed to the AXI Stream FIFO without the external ADC trigger.

When the external ADC trigger event occurs, 65,536 samples of digitised waveform data for each ADC channel will be stored in the FPGA AXI Stream FIFO. The data stored in AXI Stream FIFO is moved to the external DDR4 SODIMM, awaiting subsequent process/instruction from ZYNQ Processor System (PS). The data can be stored in both HEX or ASCII file format depending on the applied settings for subsequent analysis. The ADC trigger will be re-armed after the acquired data has completely the transfer to the ZYNQ Processor System.
FIG. 2. ICARUS-Q. The FPGA board (HTG-ZRF16), hosted within a casing, contains a unit of XCZU29DR RFSoC by Xilinx.

FIG. 3. Some of the features on the HTG-ZRF16 board that are used in this setup.

B. Microwave Generation & Detection

The on-board DACs output arbitrary waveform, generated from 65,536 samples at variable sampling rates, up to 6.144 GS/s. At its maximum sampling rate, this translates to about 10 µs of waveform points. The ADC also stores equal number samples but operates at 1.96608 GS/s, resulting in a waveform of about 33 µs. After triggering the DAC channels, there is a minimum delay of about 30 µs before triggering the next pulse. To further evaluate the performance of the DACs and the ADCs, we performed several tests and describe their results in the following.

1. Arbitrary Waveform Generation

In a typical quantum computer experiment, rectangular or Gaussian-shaped pulses are common but pulses with arbitrary phases and amplitudes [27, 28] are also often used. Some quantum information processing applications also requires the use of non-gate-based signals such as optimal control theory [29–31], adiabatic quantum computation, continuous variable quantum computing [32], etc. To demonstrate true arbitrary waveform generation capabilities, we tested the DAC with pink noise and compared the generated signal against the calculated waveform datapoints (see Fig. 6). The pink noise waveform was calculated using the Voss algorithm [33] and the signal was generated by the DAC at two different sampling rates: 1.96608 GS/s and 6.144 GS/s. Both waveforms were sampled using the ADC at 1.96608 GS/s.

In Fig. 6(a) and (b), the DAC samples and the ADC data are plotted in frequency domain, respectively. The frequency profiles bear a qualitative resemblance with each other across the frequency components, except below 1 MHz. We attribute this to the balun’s supported frequency range (10 MHz–8 GHz) where frequencies below 10 MHz are attenuated, akin to a high pass filter. In Fig. 6(b), when the DAC was operated at 6.144 GS/s, the DAC noise spectral reaches its Nyquist frequency at 3.072 GHz whereas the ADC noise spectral ends at 983.04 MHz. Additionally, the signal generated at 6.144 GS/s only lasts for approx. 10 µs (and the rest of ADC data are approximately zeroes), resulting in a lower magnitude in the ADC frequency domain.

2. Nyquist Zone Implementation

The Shannon-Nyquist sampling theorem states that a signal can be adequately generated or sampled at frequencies below half of the sampling rate. This frequency threshold is known as Nyquist frequency. However, generating or sampling signals in discrete-time creates aliases that are mirrored repeatedly across multiples of the Nyquist frequency [13, 34]. Each “segment” of the frequency domain is commonly referred to as Nyquist zones. With careful planning, one can utilise frequencies above the first zone without upgrading existing electronics [13, 23, 35].

However, there are implications. The voltage in time domain, \( v(t) \) described by:

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

is affected by the “reconstruction waveform”, \( r(t) \) [13]. The Fourier transform of \( v(t) \) is:

\[
V(\omega) = \left[ X(\omega) \sum_{n=-\infty}^{\infty} (\delta\omega T - 2\pi n) \right] \times R(\omega). \quad (2)
\]

where, \( R(\omega) \) is a sinc function that is determined by the DACs operational mode [36]. The RFSoC used here supports two modes: the non-return-to-zero (NRZ) mode and the mixed mode. The respective reconstruction waveforms in the Fourier space are two different sinc functions:

\[
R_{\text{NRZ}}(\omega) = T e^{-\omega T/2} \text{sinc} \left( \frac{\omega T}{2} \right) \quad (3)
\]
FIG. 4. The FPGA logic for the DAC waveforms generation. The figure shows the data flow from the processor system (PS), going through the DDR memory and into the AXI Stream FIFO before it is sent out through the sixteen DAC channels. A multi-tile synchronisation logic ensures that all channels are synchronised.

and

\[ R_{\text{mix}}(\omega) = \frac{\omega T^2}{4} e^{-i(\omega T - \pi)/2} \text{sinc}^2 \left( \frac{\omega T}{4} \right) \]  \hspace{1cm} (4) 

respectively.

Therefore, when using the ADC with its higher Nyquist zones, we expect the signal-to-noise ratio (SNR) to degrade to an extent. We investigated this by using the ADC at 1.96608 GS/s to sample various signals generated at higher frequencies (zones) which were aliased to 800 MHz within the first zone of the ADC. We compared the SNR when the signal is generated using a microwave synthesizer to that from the RFSoC DAC. The DAC decoder mode was set to normal (NRZ) mode for this test.

The 800 MHz signal from the RFSoC DAC, in the first Nyquist zone of the ADC sampling rate, is measured to have SNR \( \approx 2 \times 10^3 \) (see Fig. 7). The 7.06432 GHz signal, the 8th Nyquist zone alias of the 800 MHz, is in the similar range of the typical qubit transition frequency. At this frequency, the SNR is measured to be around \( 4 \times 10^2 \), which is around 5 times lower than when using 800 MHz.
FIG. 6. Pink noise generated by the DAC at (a) 1.96608 GS/s and (b) 6.144 GS/s. For both runs, the generated waveform was sampled with the ADC at 1.96608 GS/s. All plots stop at their respective Nyquist frequencies, which is one half of the sampling frequency.

FIG. 7. SNR measurements with the ADC, sampling microwave signals of higher Nyquist zones (with respect to the ADC sampling rate), from the DAC on RFSoC and a microwave synthesizer (QuickSyn). The signal in the first zone is at 800 MHz, and the ones in the higher zones are its aliases. The SNR was calculated in frequency domain using the signal peak amplitude and the average noise within a bandwidth of 100 MHz.

3. Power-Frequency Dependence

Our approach to use the higher Nyquist zone involves a power dependency as defined in Eq. 2. Since superconducting qubits are controlled and readout in the frequency domain at unique frequencies, we decided to investigate its output power-frequency dependence, particularly in between the DAC Nyquist zones. We measured the DACs output power using a spectrum analyzer from 4.5 GHz to 10 GHz, which is around the typical range for superconducting qubits and their readout resonators [37, 38]. The results are presented in Fig. 8 where Eqs. 3 and 4 were fitted to the measured DAC output power for the normal (NRZ) and the mix modes respectively.

For the normal (NRZ) mode, the output power dipped at 6.144 GHz as expected. Between 7–10 GHz, the power averaged at $-24.1 \pm 2.4$ dBm, which improved slightly between 7–9 GHz (in terms of standard deviation) to $-23.1 \pm 1.8$ dBm. Although the power variation is higher compared to microwave synthesizers or high-end benchtop AWGs, these error margin are not expected to pose significant problems since the qubits would be characterised/calibrated periodically at fixed frequencies. For the mix-mode, the expected power dip takes place at 12.288 GHz (double the sampling rate). The average power for this mode was at $-20.4 \pm 5.7$ dBm between 7–10 GHz, and $-16.9 \pm 2.3$ dBm between 7–9 GHz.

FIG. 8. Output power from 5.4 to 10 GHz, corresponding to parts of the first three Nyquist zones. The zones are delineated by the dashed lines.

The output power was observed to deviate downwards from the fitted plots at higher frequencies. We attribute this observation to the on-board balun’s supported frequency range (10 MHz–8 GHz), beyond which some attenuation is to be expected, similarly to the power spectra in Sec. II B 1—except acting as a low pass filter here.

C. Feedback Control

Having the ability to switch the waveform in real-time (nanosecond-scale) allows for the possibility to correct the qubit state in the midst of running quantum circuits. The RFSoC integrates waveform switching based on hardware trigger signal. Upon receiving the switching trigger, the outputs of the DAC channels in the upper gang (0 through 8) switches to those in the lower gang (9 through 16) within a few nanoseconds (see Fig. 9). By implementing fast readout system which signals the switching trigger to activate, correction pulses can be sent to the appropriate channels according to the qubit state.
D. Multi-channel and Multi-board Operation

In order to scale up the number of DAC and ADC channels used to control and measure the qubits, the channels need to be able to output the waveforms at a synchronised timing and phase. There are two kinds of synchronisation we ought to achieve: (1) intra-board inter-channel synchronisation, and (2) inter-board synchronisation.

Within a RFSoC board, the inter-channel synchronisation is achieved through the multi-tile synchronisation (MTS) logic in the firmware, which utilises the on-board phase-locked loop (PLL) to lock the channel outputs to the external reference clock. The inter-board synchronisation, on the other hand, is achieved by carefully distributing the single master oscillator to all of the boards such that they have the same reference clock signal for synchronisation. To generate and sample the waveforms via the DACs and the ADCs in a synchronised fashion, hardware triggers are implemented. The trigger signals are synchronised to the reference clock signal using a d-type flip-flop, such that the DACs (ADCs) output (sample) the waveform at a consistent and predictable timing upon the reception of the trigger signal.

Using the MTS logic and the trigger signal through the d-type flip-flop, the DAC outputs from two RFSoC boards were synchronised (see Fig. 11). The oscillations in the trigger signal are caused by the leakage of the clock signal through the flip-flop, but did not affect the trigger reception by the RFSoC boards. The slight delay on Board B is caused by the difference in the lengths of the trigger distribution paths - which can be easily corrected by exactly matching the cable lengths (or by introducing delays in the software).

E. Low Noise DC Biasing (Circuit)

For driving the bias circuit current, needed to set the idle frequency of the tunable qubits, we have developed a low-noise bipolar current source that can be controlled via software from the main computer. In the design process we adopted the following considerations. (i) Ultra-low noise: as any noise in the current is directly affecting the coherence properties of the qubit, the current noise of the supply should be as low as possible. (ii) Ultra-low current drift: any drift in the current directly alters the qubit properties and should therefore be suppressed. (iii) Current bandwidth: depending on the design, the change of one flux quantum through the SQUID loop typically corresponds to a change in current of sub-milliampere to a few milliampere. The current range of the source needs to be able to generate a change of at least one flux quantum. In addition the source should be bipolar. (iv) Automation: the source should be addressable via a standard protocol like USB or Ethernet in order to integrate
it into the software workflow.

Fig. 12 shows the basic design of our current source. The current controller is embedded on a PCB board together with a microcontroller and a DAC that sets the current value. Each PCB hosts 4 current controllers, which are connected to the experiment via SMA cables. In order to set a current value on a certain current controller, the host PC sends the corresponding DAC value and channel number to the microcontroller, which programs the DAC via a serial interface. For the microcontroller we chose a Teensy 4.1 (PJRC.com LLC), which contains an Ethernet interface. Through the Ethernet interface we are able to control multiple PCBs from our host PC.

![Fig. 12. Working principle of the DC current source. A host PC sends current values data to a microcontroller via an Ethernet interface. Subsequently the microcontroller changes the setpoint of the current control circuit via reprogramming the DAC.](image)

1. Current Controller Circuit

In the following we give a description of the current controller circuit. In the design, we followed the paper by Ref. [39] with some modifications. Fig. 13 shows a simplified schematic of the setup. For clarity, details like supply voltages, decoupling capacitors and connectors have been omitted.

At the heart of the current controller is an operational amplifier that acts as a current source and is configured as an integrator circuit. We chose to use an OPA547 (Texas Instruments Inc.) for this purpose as it can supply large currents of up to 500 mA and has a low input noise density of 70 nV/√Hz at 1 kHz. The current output of the amplifier is stabilised using a sense resistor, $R_{\text{sense}}$, that converts the current to a voltage $V_{\text{sense}}$. Subtracting $V_{\text{sense}}$ from the reference voltage $V_{\text{ref}}$ creates a feedback signal that is used for integral control of the current. In this case, the amplifier itself is acting as the integrator.

It is important that each part in the circuit has low noise and drift. For sensing the current we chose a metal foil resistor (Vishay Foil Resistor) with a resistance of 500 Ω and a temperature coefficient of ±2 ppm/°C. To acquire $V_{\text{sense}}$, we first buffer the high-side and low-side of the sense resistor independently using zero-drift operational amplifiers (ADA4522-2, Analog Devices Inc.).

The buffered signals are then fed into a precision operational amplifier (AD8422, Analog Devices Inc.) to obtain $V_{\text{sense}}$. For the subsequent determination of the feedback signal another AD8422 chip is used to obtain $V_{\text{sense}} - V_{\text{set}}$.

![Fig. 13. Current control circuit. The output current of an operational amplifier is stabilised by integral feedback. The setpoint of the current control circuit via reprogramming of the DAC. As this current source should be bipolar, we are operating the DAC in its bipolar mode, which allows us to set the output voltage from $-V_{\text{ref}}$ to $+V_{\text{ref}}$, where $V_{\text{ref}}$ is the reference voltage supplied to the DAC. It is crucial that $V_{\text{ref}}$ has low noise and low drift, as it directly influences the output current of the source. To generate $V_{\text{ref}}$ we are using a ADR4520 (Analog Devices Inc.) reference voltage, which has a temperature coefficient of ±2 ppm/°C and a peak-to-peak noise of 1 µV in the frequency range of 0.1–10 Hz. The output voltage of the ADR4520 chip is 2.048 V, which results in an output range of $V_{\text{set}}$ from $-2.048$ V to $+2.048$ V.

In our supply design, the current range is given by the sense resistor and the maximum absolute value of $V_{\text{set}}$. For our choice we get a current range of $\pm V_{\text{ref}}/R_{\text{sense}} = \pm 4.096$ mA, which is sufficient for our experiments. We are supplying the current controlling amplifier with a voltage of ±12 V. This results in a compliance voltage of the supply of 12 V – $V_{\text{ref}} \approx 9.9$ V.

2. Current Supply Performance

To demonstrate the noise and stability of the current source, we performed measurements of the current amplitude noise and probed the long-term behaviour by measuring the Allan deviation of the current source. Moreover, we tested the current source in experiment by tuning the frequency of a superconducting qubit.

We evaluated the current noise and Allan deviation by DC coupling 1 mA of current to the 50 Ω input of a 12-bit digital oscilloscope (LeCroy, Waverunner 610Zi). To
determine the current noise, the measured voltage was digitised with a sampling rate of 2 MS/s and a total measurement time of 30 s. The full amplitude range of the oscilloscope was set to 80 mV, which corresponded to a current resolution of 390 nA. From the time series data we calculated the current amplitude noise using Fast Fourier Transform (FFT). Fig. 14 shows a typical current amplitude noise spectrum. It can be seen that the broadband noise of the supply is below $3 \times 10^{-6} \text{A/} \sqrt{\text{Hz}}$. Taking the mean value of the amplitude noise in the band of 0.1 Hz–10 kHz and multiplying it by the square root of the same range, we obtain a total noise value of $5.45 \times 10^{-6} \text{A}_{\text{rms}}$ (0.1 Hz–10 kHz).

In order to evaluate the stability of the current source, we performed a long-term measurement of the current, using a sampling rate of 500 S/s. The full amplitude range of the oscilloscope was set to 40 mV, which corresponded to a current resolution of 195 nA. From the time series data we calculated the fractional overlapping Allan deviation. Fig. 15 (blue line) shows the Allan deviation of the current for a 18 hours long measurement. It can be seen that the bias stability of the source is about $4 \times 10^{-4}$ at an averaging time of 500 s. For comparison we also plot the slope of a white noise source (dotted-dashed line). For smaller averaging times, the source of the Allan deviation of the current is consistent with white noise.

FIG. 14. High frequency amplitude noise of the DC current source. The total noise in the band of 0.1 Hz to 10 kHz is $1.3 \times 10^{-7} \text{A}_{\text{rms}}$.

FIG. 15. Stability measurement of the current source. Depicted (solid line) is the fractional overlapping Allan deviation of a 18 hours long-term current measurement. The bias stability is about $4 \times 10^{-4}$ at 500s averaging time. Comparison to a white noise slope (dotted-dashed line) shows that for shorter averaging time the deviation is caused by white noise.

FIG. 16. Transmission spectrum of a cavity dispersively coupled to a biased superconducting transmon qubit. Shown are the spectra for bias values from -2.0 mA to 2.0 mA. In this range the frequency of the qubit is going through 3 periodic changes.

After characterising the noise performance of the current source, we tested its capabilities of biasing a superconducting circuit. In this case we measured the transmission of a cavity, which was dispersively coupled to a tunable superconducting transmon qubit. Due to the dispersive coupling, a shift of the qubit frequency directly reflects in the shift of the cavity resonance frequency. We measured the transmission spectrum of the cavity for various bias currents of the qubit. Fig. 16 shows a 2D map of the cavity transmission over a current range of -2.0 mA to +2.0 mA. It can be seen that the frequency of the resonator shifts with applied bias current as expected [40]. Over the measured current range we observe roughly 3 periods of qubit frequency oscillation. This shows that the current source is suitable for biasing a superconducting qubit linearly and with low noise.

F. Software Control & Cloud Access

In this section we provide an overview of the layout implemented for software control of the RFSoC systems and the cloud-based execution scheduler of pulse experiments.

1. Software Control

Following the layout presented in Section II, the RFSoC board used in this project has commands for board
configuration, external clock locking, channel synchronisation and starting the FPGA.

These commands are executed by a central computer (worker) through the use of the Secure Shell (SSH) protocol. We define a Python class IcarusQ that handles the transfer of user-defined waveforms into the RFSoC, processing of ADC data from the device and execution of the board commands. This class is the interface between experimenter and the RFSoC and provides remote control of the instrument. As this class is written in Python, it can be used alongside other Python instrumentation packages such as PyVISA [41], QCoDeS [42] and Python IVI [43].

The worker communicates with other instruments such as the trigger source for the RFSoC boards, via respective connections. Through these connections, the worker is able to control the triggering for the DACs and the ADCs. On the worker we define a Python function IcarusQ-Executor that runs a pulse experiment. This function takes in user-defined trigger timings, number of repetitions and pulse sequence and sets up the trigger control and the RFSoC. Then, it starts both devices and runs the pulse sequence. On completion, it returns the FPGA ADC data as the results of the experiment.

2. Cloud Access

As the input and output of IcarusQ-Executor is well defined, we can expose it as a cloud service to run experiments remotely. In Fig. 17, we explain our implementation of hosting the remote experiment execution on a cloud platform. Through this approach, we have the Flask server [44] and Redis [45] database act as a broker between the user and the worker. Neither party directly communicates with each other and the API follows a strict format. Hence, we are able to create a secure environment to execute our experiments remotely.

Finally, the currently layout will be interfaced to the Qibo framework [47] in order to automate the process of circuit execution.

III. CONCLUSION

Our setup for superconducting qubit control and measurement was designed with the following goals in mind: to be scalable, to minimise the number of microwave components/instruments to be managed and calibrated, and to have remote access capabilities for the experimenter. To these ends, we opted for the Xilinx RFSoC device with sixteen channels of DAC and ADC with high sampling rate to support direct generation and detection of microwave signals at 5–8 GHz. The approach avoids using physical IQ mixers and eliminates the periodic calibration associated with the mixers. We also designed a circuit that synchronised several FPGA boards together using a master clock and triggers for the DAC and ADC channels. The remote access to the experiment was implemented with an API to a database on a server, where the user can submit abstracted experimental parameters and retrieve the results.

In the near-term future, a few enhancements to the capabilities are being considered. In particular, the use of more features on the RFSoC device such as the utilising the on-board mixer as an alternative waveform generation and sampling method. With all of the important features integrated into a single chip, one can consider the idea of migrating the RFSoC device from room temperature into the dilution refrigerator [48] for better SNR and less connectivity to room temperature electronics. To further support larger number of samples, we will work on memory optimisation and will also consider faster DDR RAMs (such as QUAD channel RAMs) for direct streaming to/from the channels. Next, further developments are to take place in the next generations of RFSoC upon their availability.

ACKNOWLEDGMENTS

This work was supported by the National Research Foundation Singapore, the Ministry of Education Singapore, Defence Science and Technology Agency Singapore, DSO National Laboratories Singapore and the Air Force Research Laboratory US.
[1] The control systems for electron spin qubits with electron paramagnetic resonance is nearly identical to that for superconducting qubits [4, 8].

[2] M. Tseitlin, R. W. Quine, G. A. Rinard, S. S. Eaton, and The control systems for electron spin qubits with electron paramagnetic resonance spectrometer using an arbitrary waveform generator for quantum control experiments at millikelvin temperatures, Review of Scientific Instruments 86, 063110 (2015).

[3] D. Riste, S. Poletto, M.-Z. Huang, A. Bruno, V. Vesterinen, O.-P. Saira, and L. DiCarlo, Detecting bit-flip errors in a logical qubit using stabilizer measurements, Nature communications 6, 1 (2015).

[4] Y. S. Yap, Y. Tabuchi, M. Negoro, A. Kagawa, and M. Kitagawa, A Ku band pulsed electron paramagnetic resonance spectrometer based on a phase-locked loop system, Journal of Magnetic Resonance 235, 95 (2013).

[5] C. A. Ryan, B. R. Johnson, D. Ristè, B. Donovan, and T. A. Ohki, Hardware for dynamic quantum computing, Review of Scientific Instruments 88, 104703 (2017).

[6] L. Stefanazzi, K. Treptow, N. Wilcer, C. Stoughton, S. Montella, C. Bradford, G. Cancelo, S. Saxena, H. Arnoldi, S. Sussman, et al., The QICK (Quantum Instrumentation Control Kit): Readout and control for qubits and detectors, arXiv preprint arXiv:2105.00557 (2021).

[7] M. Esposito, A. Ranadive, L. Planat, S. Leger, C. Naud, J. Aumentado, F. Lecocq, and N. Roch, Observation of two-mode squeezing in a traveling wave parametric amplifier (2021), arXiv:2111.03696 [quant-ph].

[8] F. Motzoi, J. M. Gambetta, P. Rebentrost, and F. K. Wilhelm, Simple pulses for elimination of leakage in weakly nonlinear qubits, Physical review letters 103, 110501 (2009).

[9] S. E. Economou and E. Barnes, Analytical approach to swift nonleaky entangling gates in superconducting qubits, Physical Review B 91, 161405 (2015).

[10] N. Khaneja, R. Brockett, and S. J. Glaser, Time optimal control, Journal of Magnetic Resonance 230, 27 (2013).

[11] A. Wagner, B. Donovan, D. Ristè, and T. A. Ohki, High-fidelity control of superconducting qubits using direct microwave synthesis in higher Nyquist zones, IEEE Transactions on Quantum Engineering 1, 1 (2020).

[12] Y. Xu, G. Huang, J. Balewski, R. Naka, A. Morvan, B. Mitchell, K. Nowrouzi, D. I. Santiago, and I. Sididi, QubiC: An Open-Source FPGA-Based Control and Measurement System for Superconducting Quantum Information Processors, IEEE Transactions on Quantum Engineering 2, 1 (2021).

[13] Y. Yang, Z. Shen, X. Zhu, Z. Wang, G. Zhang, J. Zhou, C. Deng, S. Liu, and Q. An, FPGA-based electronic system for the control and readout of superconducting qubit systems, arXiv preprint arXiv:2110.07965 (2021).

[14] T. Lanting, H.-M. Cho, J. Clarke, W. Holzapfel, A. T. Lee, M. Lueker, P. Richards, M. A. Dobbs, H. Spieler, and A. Smith, Frequency-domain multiplexed readout of transition-edge sensor arrays with a superconducting quantum interference device, Applied Physics Letters 86, 112511 (2005).

[15] Y. Chen, D. Sank, P. O’Malley, T. White, R. Barends, B. Chiaro, J. Kelly, E. Lucero, M. Marianioni, A. Megrant, et al., Multiplexed dispersive readout of superconducting phase qubits, Applied Physics Letters 101, 182601 (2012).

[16] J. Hornibrook, J. Colless, A. Mahoney, X. Crot, S. Blanvillain, H. Lu, A. Gossard, and D. Reilly, Frequency multiplexing for readout of spin qubits, Applied Physics Letters 104, 103108 (2014).

[17] See Xilinx website for more information.

[18] R. Fagan, F. C. Robey, and L. Miller, Phased array radar cost reduction through the use of commercial RF systems on a chip, in 2018 IEEE Radar Conference (RadarConf) (2018) pp. 0935–0939.

[19] B. Farley, C. Erdmann, B. Vaz, J. McGrath, E. Cullen, B. Verbruggen, R. Pelliconi, D. Breathnach, P. Lim, A. Boumaalif, et al., A programmable RFSoC in 16nm FinFET technology for wideband communications, in 2017 IEEE Asian Solid-State Circuits Conference (A-SSCC) (IEEE, 2017) pp. 1–4.

[20] L. Grünhaupt, I. Takmakov, P. Winkel, L. Planat, N. Roch, W. Wernsdorfer, et al., State preparation of a fluxonium qubit with feedback from a custom FPGA-based platform, in AIP Conference Proceedings, Vol. 2241 (AIP Publishing LLC, 2020) p. 020015.

[21] L. Stefanazzi, K. Treptow, N. Wilcer, C. Stoughton, S. Montella, C. Bradford, G. Cancelo, S. Saxena, H. Arnoldi, S. Sussman, et al., The QICK (Quantum Instrumentation Control Kit): Readout and control for qubits and detectors, arXiv preprint arXiv:2110.00557 (2021).

[22] M. Esposito, A. Ranadive, L. Planat, S. Leger, D. Fraudet, V. Jouanny, O. Buisson, W. Guichard, C. Naud, J. Aumentado, F. Lecocq, and N. Roch, Observation of two-mode squeezing in a traveling wave parametric amplifier (2021), arXiv:2111.03696 [quant-ph].

[23] The main difference between the first and second generation is an extra 1 GHz in the maximum input frequency.

[24] In the product description, there is a total of 60.5 Mbits memory. This is actually a combination of 38.0 Mb of block RAM and 22.5 Mb of UltraRAMs. In our current design, we are only able to utilize the block RAM for various purposes and not solely for the ADC and DAC data.
control in spin systems, Physical Review A 63, 032308 (2001).

[30] Y. Tabuchi, M. Negoro, and M. Kitagawa, Design method of dynamical decoupling sequences integrated with optimal control theory, Physical Review A 96, 022331 (2017).

[31] M. Werninghaus, D. J. Egger, F. Roy, S. Machnes, F. K. Wilhelm, and S. Filipp, Leakage reduction in fast superconducting qubit gates via optimal control, npj Quantum Information 7, 1 (2021).

[32] C. Song, S.-B. Zheng, P. Zhang, K. Xu, L. Zhang, Q. Guo, W. Liu, D. Xu, H. Deng, K. Huang, et al., Continuous-variable geometric phase and its manipulation for quantum computation in a superconducting circuit, Nature communications 8, 1 (2017).

[33] Visit ‘Think DSP: Digital Signal Processing in Python, by Allen B. Downey’ for more information.

[34] S. S. Soliman and M. D. Srinath, Continuous and discrete signals and systems, Englewood Cliffs, 351 (1990).

[35] C. Erdmann, E. Cullen, D. Brouard, R. Pelliconi, B. Verbruggen, J. McGrath, D. Collins, M. De La Torre, P. Gay, P. Lynch, et al., 16.3 a 330mw 14b 6.8 GS/s dual-mode RF DAC in 16nm FinFET achieving- 70.8 dBc ACPR in a 20MHz channel at 5.2 GHz, in 2017 IEEE International Solid-State Circuits Conference (ISSCC) (IEEE, 2017) pp. 280–281.

[36] Zynq UltraScale+ RFSoC RF Data Converter.

[37] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, et al., Quantum supremacy using a programmable superconducting processor, Nature 574, 505 (2019).

[38] 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 6, 2 (2019).

[39] Y.-M. Yang, H.-T. Xie, W.-C. Ji, Y.-F. Wang, W.-Y. Zhang, S. Chen, and X. Jiang, Ultra-low noise and high bandwidth bipolar current driver for precise magnetic field control, Review of Scientific Instruments 90, 014701 (2019), https://doi.org/10.1063/1.5046484.

[40] 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, Phys. Rev. A 76, 042319 (2007).

[41] P. team, Pyvisa: Control your instruments with python (2021).

[42] M. quantum computing consortium, Qcodes (2021).

[43] L. foundation, A python implementation of the interchangeable virtual instrument standard. (2021).

[44] Pallets, Flask (2021).

[45] R. Ltd., Redis (2021).

[46] Amazon, Amazon web services (aws) (2021).

[47] S. Efthymiou, S. Ramos-Calderer, C. Bravo-Prieto, A. Pérez-Salinas, D. García-Martín, A. García-Saez, J. I. Latorre, and S. Carrazza, Qibo: a framework for quantum simulation with hardware acceleration (2020), arXiv:2009.01845 [quant-ph].

[48] S. Pauka, K. Das, R. Kalra, A. Moini, Y. Yang, M. Trainer, A. Bousquet, C. Cantaloube, N. Dick, G. Gardner, et al., A cryogenic cmos chip for generating control signals for multiple qubits, Nature Electronics 4, 64 (2021).