Review of Scientific Instruments ARTICLE scitation.org/journal/rsi

ICARUS-Q: Integrated control and readout unit for scalable quantum processors

Cite as: Rev. Sci. Instrum. 93, 104704 (2022); doi: 10.1063/5.0081232
Submitted: 8 December 2021 • Accepted: 30 August 2022 •
Published Online: 6 October 2022

Kun Hee Park,[1] Yung Szen Yap,[1,2,a)] Yuanzheng Paul Tan,[1,3] Christoph Hufnagel,[1] Long Hoang Nguyen,[3] Karn Hwa Lau,[4] Patrick Bore,[1] Stavros Efthymiou,[5] Stefano Carrazza,[5,6,7] Rangga P. Budoyo,[1] and Rainer Dumke[1,3,b)]

AFFILIATIONS

[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, No. 05-75, Midview City 573969, Singapore
[5] Quantum Research Centre, Technology Innovation Institute, Accelerator 2 Building, on Plot M12, P.O. Box 9639, Masdar City, Abu Dhabi, United Arab Emirates
[6] TIF Lab, Dipartimento di Fisica, Università Degli Studi di Milano and INFN Sezione di Milano, via Celoria 16, 20133 Milan, Italy
[7] Theoretical Physics Department, CERN, Esplanade des Particules 1, 1211 Meyrin, Switzerland

[a)] **Author to whom correspondence should be addressed:** yungszen@utm.my
[b)] rdumke@ntu.edu.sg

ABSTRACT

We present a control and measurement setup for superconducting qubits based on the Xilinx 16-channel radio-frequency system-on-chip (RFSoC) device. The proposed setup consists of four parts: multiple RFSoC boards, a setup to synchronize every digital to analog converter (DAC) and analog to digital converter (ADC) channel across multiple boards, a low-noise direct current supply for tuning the qubit frequency, and cloud access for remotely performing experiments. We also designed the setup to be free of physical mixers. The RFSoC 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), analog to digital converters (ADCs), and stable current sources; the DACs generate the microwave pulses that travel into the fridge, the ADCs digitize the analog signals that travel out of the fridge, and the current source tunes the qubit frequencies. Some of the earlier microwave control systems for electron spin and superconducting qubits[1] relied on benchtop arbitrary waveform generators (AWGs) for microwave generation.[2–7] 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 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 FPGAs 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 features the highest number of independent DAC and ADC channels within a single chip with high sampling rates and is equipped with auto-synchronization 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 of those of other commercial off-the-shelf devices, make 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 the XCZU28DR and XCZU29DR. There primary differences between them are the number of channels (8 vs 16 channels of DACs and ADCs) and the maximum sampling rate of the ADCs (4.096 vs 2.058 GS/s, respectively).

A single RFSoC 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 synchronized XCZU29DR RFSoC boards, where each board features 16 6.554 GS/s DAC channels and 16 2.058 GS/s ADC channels per board and operates without physical IQ mixers.

II. IMPLEMENTATION

The developed setup in Fig. 1 consists of several parts. The RFSoC 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. 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 (Secs. II A–II D). Multiple RFSoC boards are synchronized with each other using a master oscillator and trigger signals that are synchronized to the same master oscillator (Sec. II C). For tunable superconducting qubits, the Josephson junction of the qubit is replaced by a direct current superconducting interference device (DC-SQUID) loop. This allows the qubit to be tuned 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 tuning the qubits (Sec. II F). The RFSoC boards, triggers, 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 G).

The RFSoC board used in this work is available commercially (HTG-ZRF16, HiTech Global) and is populated with one unit of an XCZU29DR RFSoC device (see Fig. 2 for the photograph) that comes with 16 14 bit DAC (6.554 GS/s) and 16 12 bit ADC (2.058 GS/s) differential pins. On the RFSoC board, these are converted to single-ended signals using baluns (which support frequencies from 10 MHz to 8 GHz) and SSMC ports (which support frequencies up

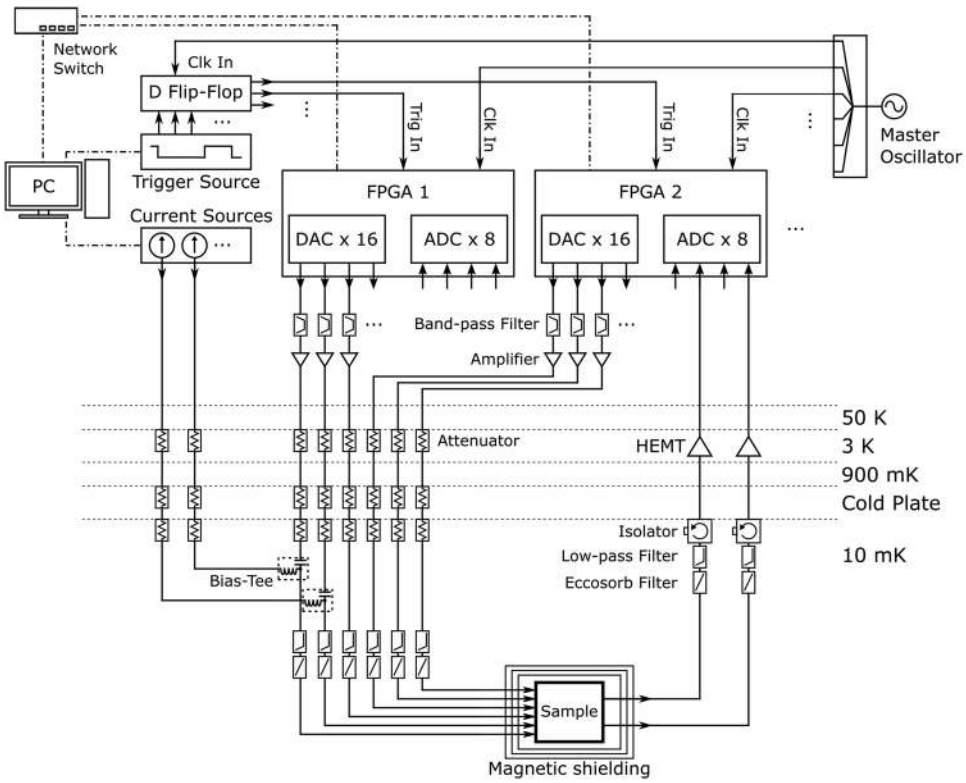

FIG. 1. Overall circuit diagram. Several units of RFSoC boards, each with an RFSoC device (XCZU29DR by Xilinx), are used to directly generate and sample microwave signals in the GHz frequencies using their higher Nyquist zones (with the help of filters). The boards are synchronized to a master reference clock (VHF Citrine Gold, Wenzel) and triggered by a programmable[26] TTL pulse generator (PulseBlaster PB24-100-4K, Spincore). The triggers are also synchronized to the same clock using a d-type flip-flop circuit. The trigger source board runs a custom firmware that takes in the master oscillator signal as the reference clock. To tune 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 PCIe and to the current sources via USB. The RFSoC boards are controlled from the PC via SSH over Ethernet through the network switch.

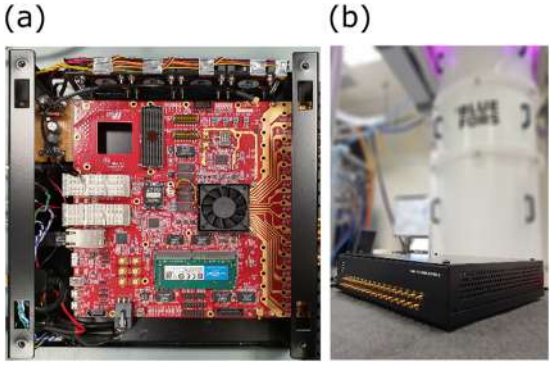

FIG. 2. ICARUS-Q. (a) The RFSoC board (HTG-ZRF16), hosted within a casing, contains a unit of XCZU29DR RFSoC by Xilinx. (b) The front panel view of the enclosed RFSoC board.

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 1.966 08 GS/s for the ADCs. This limit occurs because the 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 memory[27] to the DACs, we reduced the number of active ADC channels to eight.

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 external triggers. For the current FPGA design with 16 active DACs and eight active ADCs, we utilized around 50% of the configurable logic blocks, 47% of the total FPGA RAM (BRAM usage is around 75% without using Ultra RAMs), and almost none of the Digital Signal Processing (DSP) slices (0.12%). This should leave space for future improvements, especially in real time calculation of the acquired signals. The HTG-ZRF16 also contains a two-stage clock distribution logic for phase synchronization of the ADCs and DACs. The FPGA also encompasses the Ethernet, microSD card, and DDR memories on the HTG-ZRF16 board. In the following parts, we describe the FPGA logic for the DAC and ADC implementations.

1. FPGA logic for DAC waveforms generation

The arbitrary waveforms are generated using the RFSoC DAC (see Fig. 4). The 16 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-tile synchronization logic inside the RFSoC is utilized to calibrate the "tile-to-tile" time skew.

The DAC data flow starts by loading the waveform of interest into the programmable logic (PL) DDR memory. PL DDR memory is a physical SODIMM memory module connected to the PL-FPGA. The waveform data are then moved into the AXI Stream First-In, First-Out (FIFO) of each DAC channel. 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, each AXI Stream FIFO can store up to 65 536 samples of DAC waveform for each channel.

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

Once the waveform data are loaded into the 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

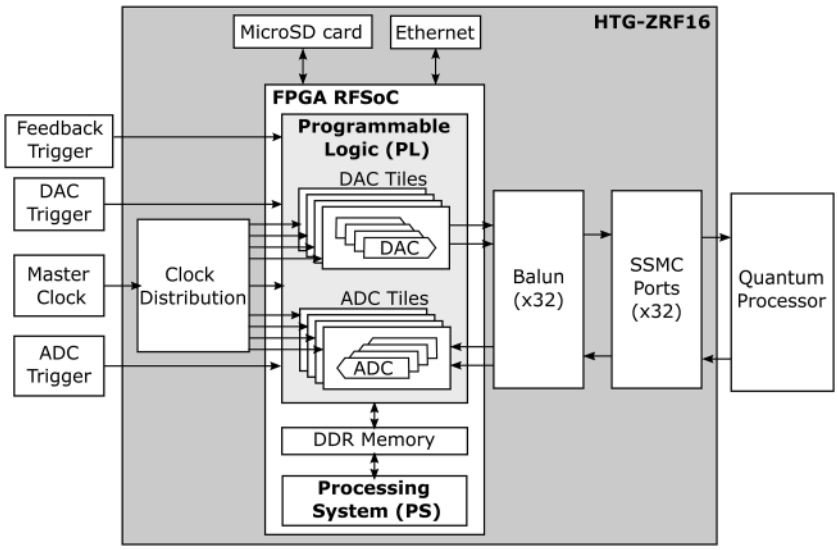

FIG. 3. Some of the features on the HTG-ZRF16 board are used in this setup. The board takes in multiple triggers and a clock signal as a reference for the FPGA logic. The clock distribution subsystem distributes it to the FPGA as well as the DACs and ADCs tiles. The DACs and ADCs tiles generate differential microwave signals, which are converted to single-ended signals and transmitted out via SSMC ports. The board also supports communication via microSD card and Ethernet.

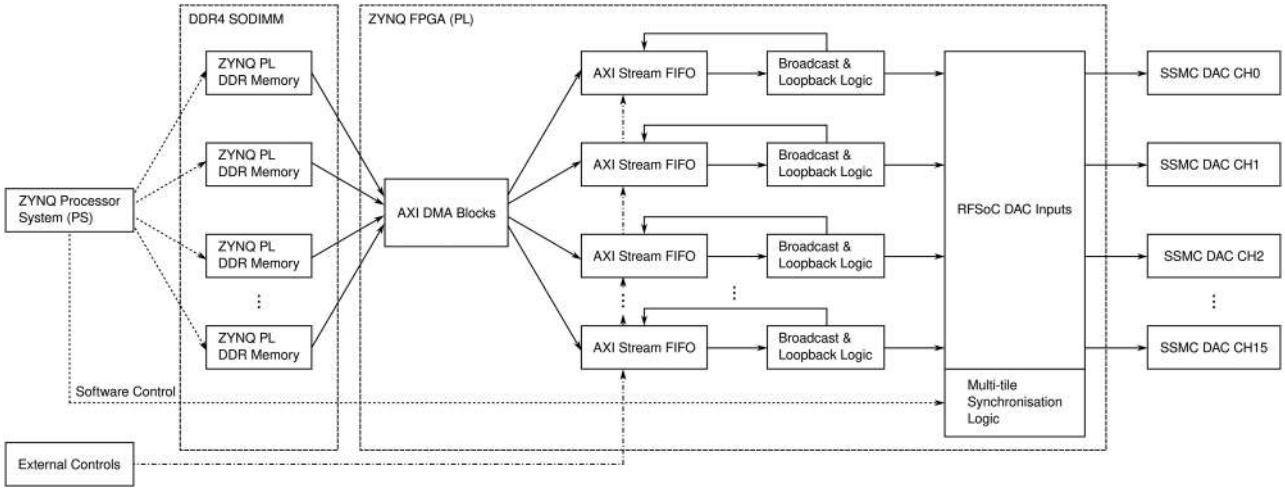

FIG. 4. The FPGA logic for the DAC waveforms generation. The figure shows the data flow from the processor system (PS), going through the DDR memory and into the AXI Stream FIFO before it is sent out through the sixteen DAC channels. A multi-tile synchronization logic ensures that all channels are synchronized.

the waveform data swap. When enabled, the waveforms of the upper eight channels can be swapped to the lower eight channels to support more advanced pulse sequences (see Sec. II D for more information).

2. FPGA logic for ADC waveforms acquisition

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

When the external ADC trigger event occurs, 65 536 samples of digitized waveform data for each ADC channel will be stored in the FPGA AXI Stream FIFO. The data stored in the AXI Stream FIFO is moved to the external DDR4 SODIMM, awaiting subsequent process/instruction from the ZYNQ Processor System (PS). The data can be stored in both HEX and ASCII file formats depending on the applied settings for subsequent analysis. The ADC trigger will be re-armed after the acquired data have completely the transferred to the ZYNQ Processor System.

B. Microwave generation and detection

The RFSoC has gained some degree of attention and its performance has been tested by several groups.[13,28] The on-board DACs output arbitrary waveforms, generated from 65 536 samples at

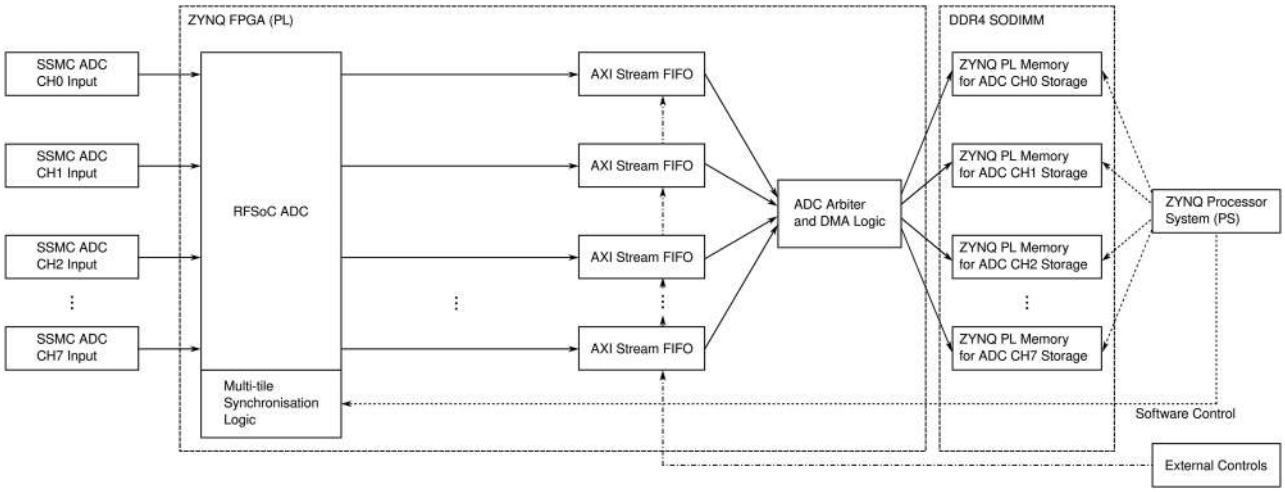

FIG. 5. The FPGA logic for the ADC waveforms acquisition. The data from the eight ADC channels streamed to the AXI Stream FIFO get moved into the DDR memory via the ADC arbiter and DMA logic, only when a trigger is received. The ADC arbiter and DMA logic then transfer the data to the programmable logic (PL) memory for storage when an instruction is received from the processor system (PS). A multi-tile synchronization logic ensures that all channels are synchronized.

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 an equal number of samples but operates at 1.966 08 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[29,30] are also often used. Some quantum information processing applications also require the use of non-gate-based signals such as optimal control theory,[31–33] adiabatic quantum computation, continuous variable quantum computing,[34] etc. In our setup, the modulated pulses are numerically designed and generated up to 3.072 GHz (one-half of the DAC's maximum sampling rate). 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,[35] and the signal was generated by the DAC at two different sampling rates: 1.966 08 GS/s and 6.144 GS/s. Both waveforms were sampled using the ADC at 1.966 08 GS/s.

In Fig. 6(a), the DAC samples and the ADC data are plotted in the frequency domain, respectively. The frequency profiles bear a qualitative resemblance to 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.

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 the Nyquist frequency. However, generating or sampling signals in discrete-time creates aliases that are mirrored repeatedly across multiples of the Nyquist frequency.[13,36] Each "segment" of the frequency domain is commonly referred to as the Nyquist zones. With careful planning, one can utilize frequencies above the first zone without upgrading existing electronics.[13,23,37]

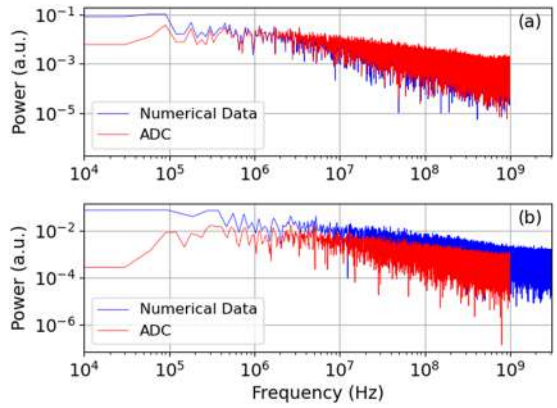

FIG. 6. Pink noise generated by the DAC at (a) 1.966 08 GS/s and (b) 6.144 GS/s. The pink noise waveform was calculated using the Voss algorithm,[35] and the DACs were fed back directly into the ADCs without any filters for sampling. For both runs, the generated waveform was sampled with the ADC at 1.966 08 GS/s. All plots stop at their respective Nyquist frequencies, which is one-half of the sampling frequency.

For controlling the qubits, we digitally generate shaped sine wave pulses directly in the GHz frequency range and use them in their respective Nyquist zones, which also naturally preserves the phase coherence of the qubit. As such, this method does not require a local oscillator or its modulation.

However, there are implications for using this approach. The voltage in the time domain, $v(t)$, described by

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

is affected by the "reconstruction waveform," $r(t)$.[13] Here, $x(t)$ is the analytical function we sample, $\delta(t - kT)$ is the Dirac delta function, and T is the sampling period. The Fourier transform of $v(t)$ is

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

where $X(\omega)$ is the Fourier transform of $x(t)$, and $R(\omega)$ is a sinc function that is determined by the DACs operational mode.[38] 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

$$R_{\mathrm{NRZ}}(\omega) = Te^{-i\omega T/2}\mathrm{sinc}\left(\frac{\omega T}{2}\right) \tag{3}$$

and

$$R_{\mathrm{mix}}(\omega) = \frac{\omega T^2}{4} e^{-i(\omega T - \pi)/2}\mathrm{sinc}^2\left(\frac{\omega T}{4}\right), \tag{4}$$

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 1.966 08 GS/s to sample various signals generated at higher frequencies (zones), which were aliased to 800 MHz within the first zone of the ADC. 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 7.064 32 GHz signal, the eighth Nyquist zone alias of 800 MHz, is in the similar range of the typical qubit transition frequency. At this frequency, the SNR is measured to be around 4×10^2, which is around five times lower than when using 800 MHz.

3. Power-frequency dependence

Our approach to using the higher Nyquist zone involves a power dependency as defined in Eq. (2). Since the superconducting

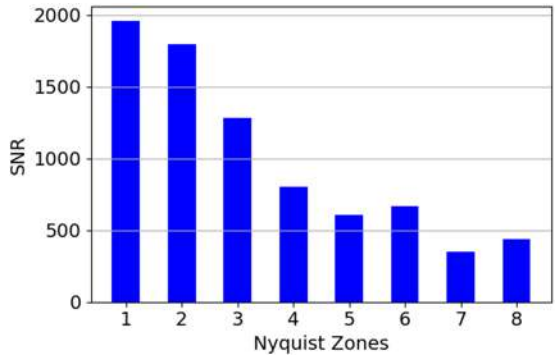

FIG. 7. SNR measurements of DAC output at different ADC Nyquist zones. The signal in the first zone is at 800 MHz, and the ones in the higher zones are its aliases. The DAC output is passed through filters to only select the alias frequency that corresponds to each ADC Nyquist zone. For example, The DAC outputs a 4732.16 MHz signal (using its second zone) and is filtered with a 2.7–6.0 GHz bandpass filter; it is sampled in the ADC's at 800 MHz (using its fifth zone). The SNRs were calculated in the frequency domain using the signal peak amplitude and the average noise within a bandwidth of 100 MHz.

qubits are controlled and read out in the frequency domain at unique frequencies, we decided to investigate their output power-frequency dependence, particularly in between the DAC Nyquist zones. We measured the DACs' output power using a spectrum analyzer from 4.5 to 10 GHz, which is around the typical range for superconducting qubits and their readout resonators.[39,40] 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 and 10 GHz, the power averaged at -24.1 ± 2.4 dBm, which improved slightly between 7 and 9 GHz (in terms of standard deviation) to -23.1 ± 1.8 dBm. Although the power variation is higher compared to microwave synthesizers or high-end benchtop AWGs, these error margins are not expected to pose significant problems since the qubits would be characterized/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 ± 5.7 dBm between 7 and 10 GHz, and -16.9 ± 2.3 dBm between 7 and 9 GHz.

The output power was observed to deviate downward from the fitted plots at higher frequencies. We attribute this observation to the onboard 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. 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 synchronized timing and phase. There are two kinds of synchronization we ought to achieve: (1) intra-board inter-channel synchronization and (2) inter-board synchronization.

Within a RFSoC board, the inter-channel synchronization is achieved through the multi-tile synchronization (MTS) logic in the firmware, which utilizes the on-board phase-locked loop (PLL) to lock the channel outputs to the external reference clock. The inter-board synchronization, 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 synchronization. To generate and sample the waveforms via the DACs and the ADCs in a synchronized fashion, hardware triggers are implemented. To prevent metastability of sampling the external trigger by the RFSoC, the trigger signals are first synchronized to the master clock signal using an external d-type flip-flop (see Fig. 9). The synchronized triggers are then distributed to multiple RFSoC boards. This ensures that every RFSoC board samples the trigger at the same clock cycle.[41]

With the master clock, MTS, and synchronized triggers, phase coherence between pulses was preserved for all channels across with multiple boards without the use of any local oscillator.

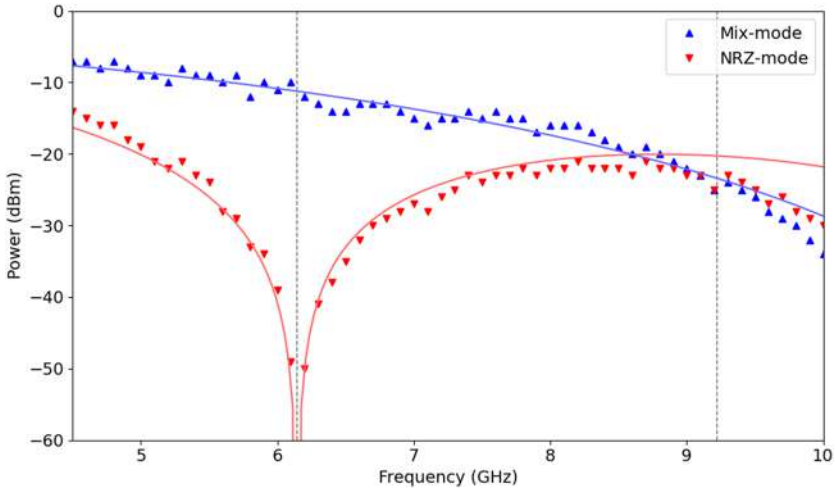

FIG. 8. Output power of the RFSoC DAC in pulse mode from 4.5 to 10 GHz, corresponding to parts of the first three Nyquist zones. The zones are delineated by the dashed lines.

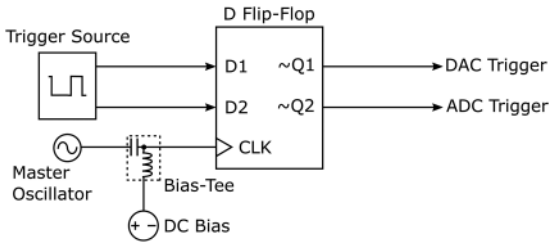

FIG. 9. Schematic of the d-type flip-flop circuit. The two independent falling-edge trigger signals for the DAC output and the ADC sampling (Q1 and Q2) are provided by the trigger source (D1 and D2) and are synchronized to a master oscillator using a dual-channel d-type flip-flop. An appropriate DC bias is supplied to the clock input of the flip-flop, so that the AC clock signal from the oscillator moves through the on/off levels.

Using the MTS logic and the trigger signal through the d-type flip-flop, the DAC outputs from two RFSoC boards were synchronized (see Fig. 10). 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).

D. 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 a hardware trigger signal. Upon receiving the switching trigger, the outputs of the DAC channels in the upper memory banks (0 through 7) switch to those in the lower memory banks (8 through 15) within a few nanoseconds (see Fig. 11).

E. Measurements with superconducting qubit

The control system was used for experiments with a non-tunable superconducting qubit to demonstrate its capabilities. For the following qubit measurements, the low noise bias circuit described in Sec. II F was not used. We also upgraded the trigger source (PulseBlaster PB12-100-K-PCIe) to receive the master clock directly without the need to use the flip-flop circuit. The qubit is a transmon with a single Josephson junction on a silicon substrate, placed in a 3D aluminum cavity. Two DAC channels, for driving the qubit and the cavity, respectively, were combined through a microwave combiner and fed through the input port of the cavity. The readout port of the cavity was connected to an ADC channel.

1. Cavity power sweep

We first demonstrate the ability of the system to perform qubit readout by driving a readout signal through the cavity and

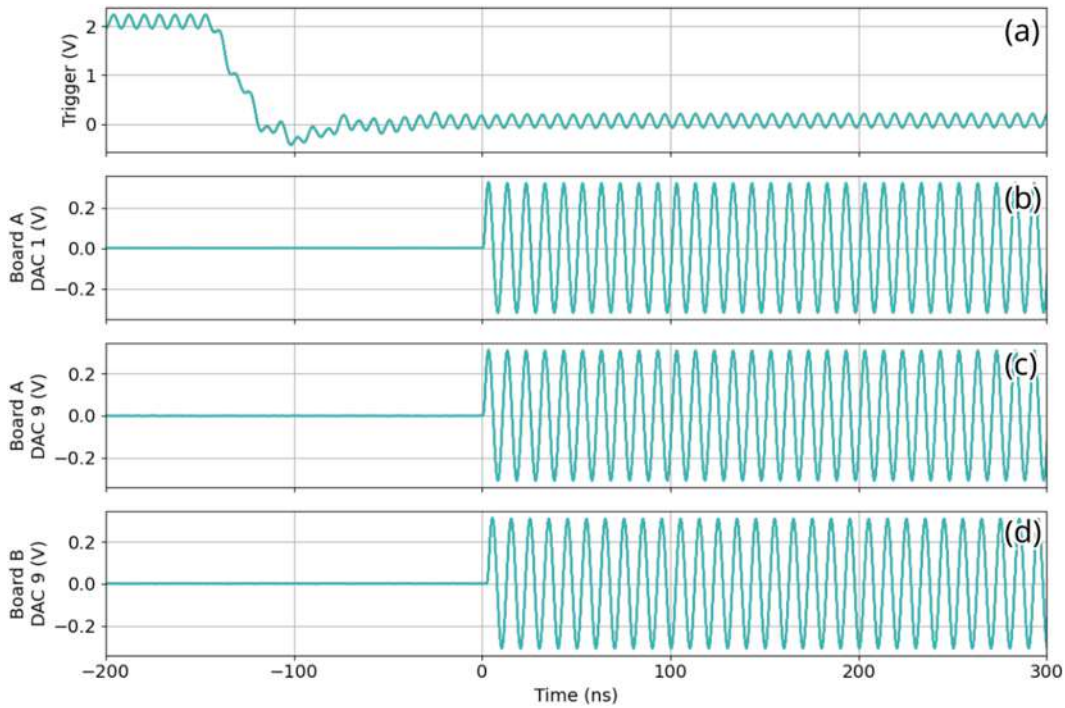

FIG. 10. Synchronization of the DAC channels from multiple boards. (a) Using a synchronized to the trigger signal, ten runs of the DAC were captured. (b)–(d) The DAC outputs from two RFSoC boards (Boards A and B) were connected to an oscilloscope and synchronization between channels of the same board and between multiple boards were demonstrated.

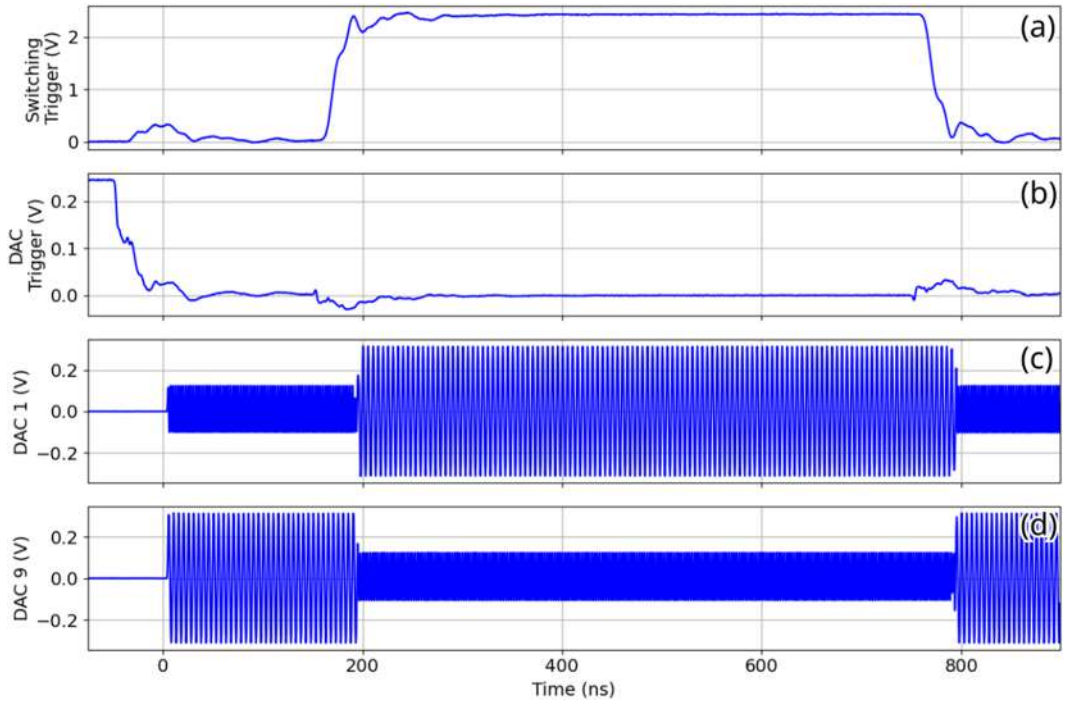

FIG. 11. DAC playback switching with hardware trigger signals. (a) The switching trigger sent to the RFSoC. (b) The DAC trigger sent to the RFSoC. (c) and (d) Upon receiving the switching trigger, the upper DAC memory banks (DACs 0–7) switch with the lower banks (DACs 8–15). The delay from the trigger to completion of switching of the channels took ~20 ± 5 ns.

measuring the response as shown in Fig. 12. A range of readout frequencies from 5.117 to 5.127 GHz was swept to test the frequency precision of the device. The sampling rate of the DAC was set at 5.898 24 GS/s and the ADC was set at 1.966 08 GS/s. These sampling rates were chosen based on the cavity and qubit frequencies and the corresponding output powers at different Nyquist zones. For this experiment, the DAC is operating at the second Nyquist zone and the ADC at the fifth Nyquist zone.

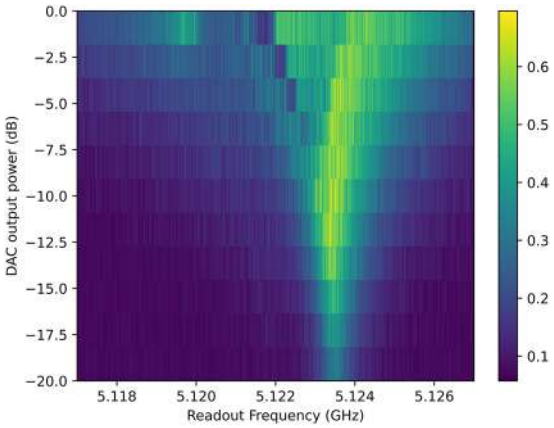

FIG. 12. Cavity transmission response. We plot the amplitude of the Fourier transform of the readout signal from the ADC as a function of DAC power and cavity driving frequency. The cavity was driven for 10 μs near its resonance frequency over 20 dB of power with a rest period of 200 ms. At each point, the amplitude was averaged over 100 shots. The dressed cavity frequency is at 5.123 75 GHz and from the data, we are able to resolve the frequency at −12 dB. The bare frequency of 5.12 GHz is slightly visible at the highest power of the instrument.

Furthermore, we test the voltage control of the instrument by varying the drive power to resolve the dispersive shift of the cavity without the use of RF attenuators. While the maximum driving power of the device was unable to fully resolve the bare frequency of the cavity, we were able to resolve the dispersive shift of 3.8 MHz and the dressed cavity frequency at −12 dB.

2. Rabi spectroscopy

Next, we move on to demonstrating qubit control with the RFSoC. We do this with Rabi spectroscopy, varying the length of the qubit driving pulse and measuring the cavity response (see Fig. 13). The qubit was driven at its resonance frequency of 3.357 GHz in the second Nyquist zone on DAC channel 1, tile 1, and measured with a readout pulse of length 5 μs on DAC channel 13, tile 4. We also compare this to a separate experimental setup, AWG (described in Ref. 42).

From the oscillation, we are able to drive and resolve the 3 ns step rotations of the qubit. Hence, we demonstrate the ability of the RFSoC to perform and measure arbitrary rotations of the qubit and synchronize DACs on separate tiles.

3. Ramsey spectroscopy

We further test the phase stability of the system over half of the available DAC playback time through a Ramsey experiment

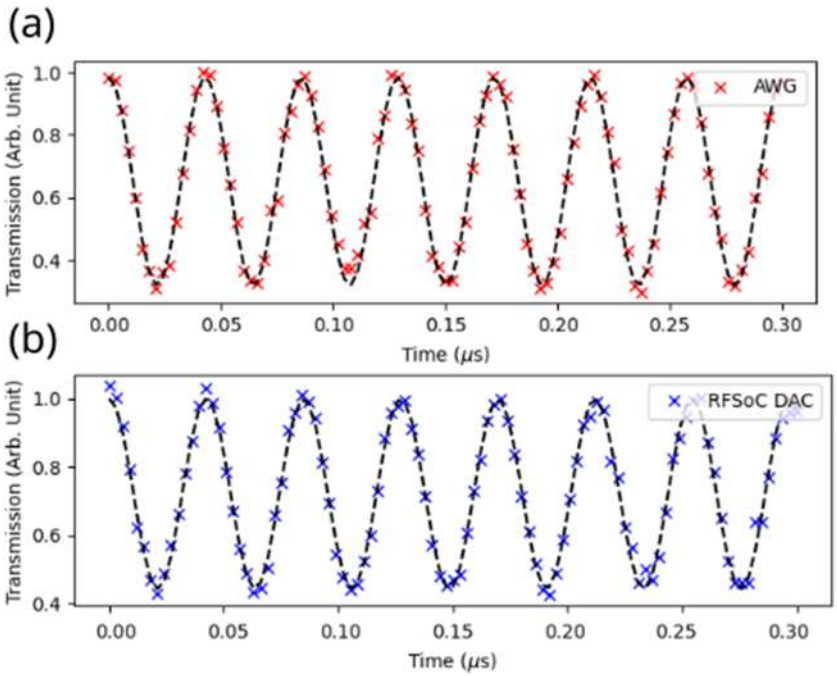

FIG. 13. Rabi oscillation of the qubit using (a) AWG and (b) RFSoC DAC, respectively. The qubit was driven with pulse durations up to 300 ns in 3 ns intervals. Each point was averaged over 1000 measurements. Through a sine wave fit, we approximate the Rabi oscillation frequency f_{Rabi} as 23.3 MHz for the AWG and 23.5 MHz for the RFSoC. We also obtain the π-pulse duration t_π of 21.5 ns for the AWG and 21.2 ns for the RFSoC.

(see Fig. 14). We apply two $\pi/2$-pulses separated by flight time τ before readout. From the Ramsey fringes, we do not observe significant deviations compared to AWG in 5 μs of free evolution time. Hence, the system appears to be stable during full playback.

Furthermore, we perform a Ramsey experiment with the feedback control (see Fig. 15) to swap the waveform memory banks as described in Sec. II D. The switching of the memory banks is carried out in between the two $\pi/2$-pulses of the Ramsey sequence. Similarly, we do not observe significant deviations when using the feedback control for the Ramsey experiment.

4. Quantum state tomography

Finally, using t_π and $t_{\pi/2} = t_\pi/2$ obtained from the results of Rabi spectroscopy, we perform Quantum State Tomography (QST)

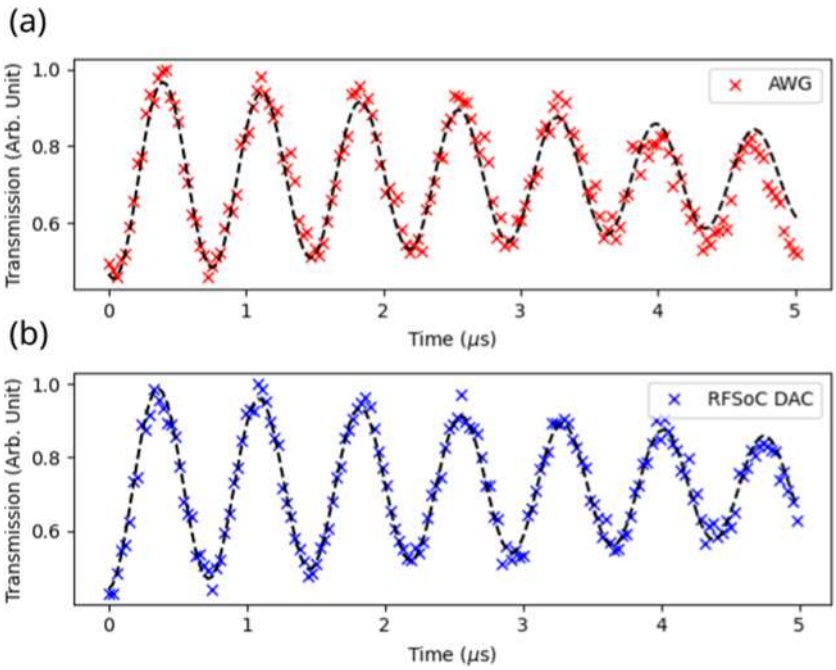

FIG. 14. Ramsey fringes. We vary τ up to 5 μs and apply a frequency detuning of 1 MHz using (a) the AWG and (b) the RFSoC. Each point was averaged over 1000 measurements. The data are fitted to a sine wave with an exponential decay, yielding an oscillation frequency of 1.367 MHz and a dephasing time of 6.66 μs for the RFSoC and similar values of 1.38 MHz and 6.52 μs for the AWG, respectively.

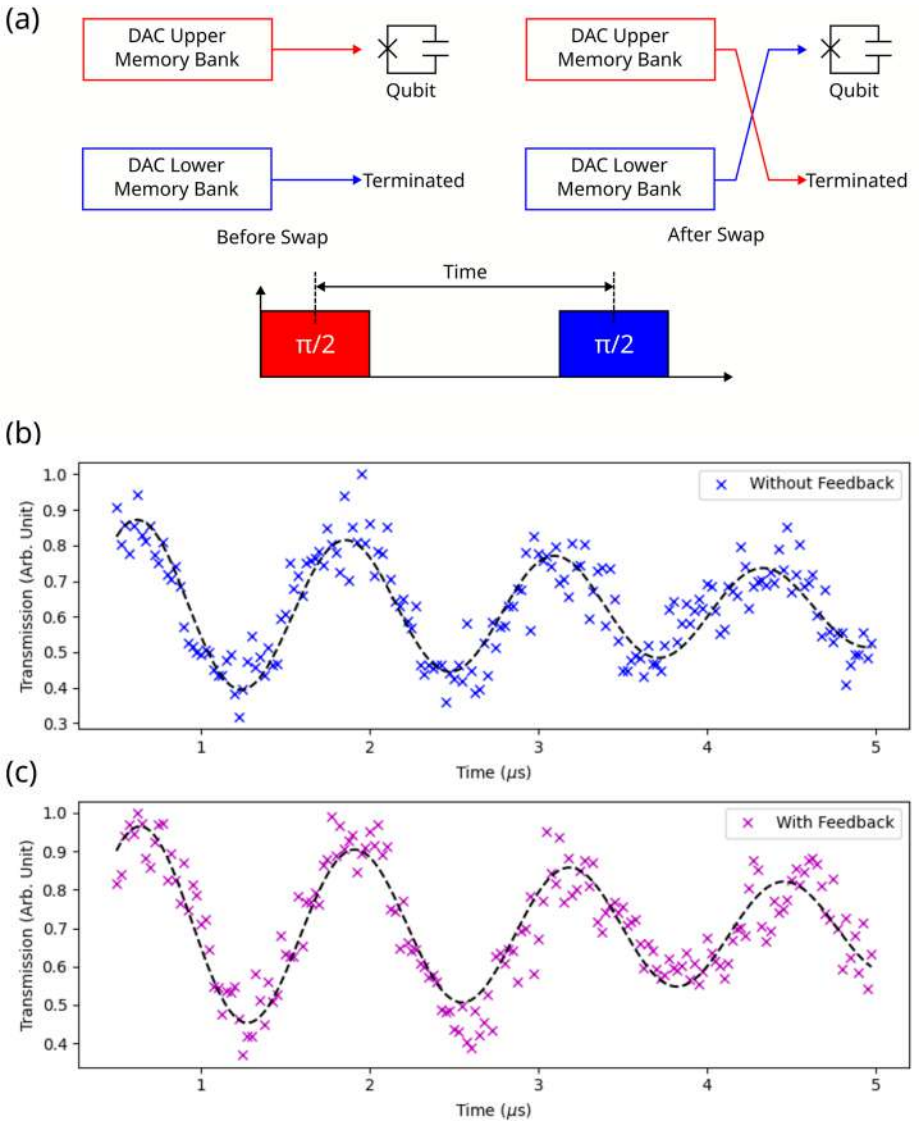

FIG. 15. Ramsey fringes performed with feedback control (i.e., swapping of the DAC memory banks). (a) To test the effects of the feedback, we carried out a Ramsey experiment where the first $\pi/2$-pulse originated from the original memory bank and the second $\pi/2$-pulse was generated from the swapped memory bank. The Ramsey fringes (b) without and (c) with feedback control. Each point was averaged over 1000 measurements. The fits yield dephasing times of 4.84 and 5.08 μs, respectively. This experiment was performed at a different cooldown as compared to Fig. 14.

with Maximum Likelihood Estimate (MLE) using $I, X, X/2, Y/2$ as pre-rotation gates for the $\frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle)$ state to evaluate the state fidelity (see Fig. 16).

With this measurement, we demonstrate the ability of the ICARUS-Q platform to perform qubit control and measurement. We have also described the implementation of a single-qubit algorithm on the RFSoC in the Appendix.

F. 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 the current of sub-milliampere to a few milliamperes. 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.

Figure 17 shows the basic design of our current source. The current controller is embedded on a printed circuit board (PCB) together with a microcontroller and a DAC that sets the current value. Each PCB hosts four 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

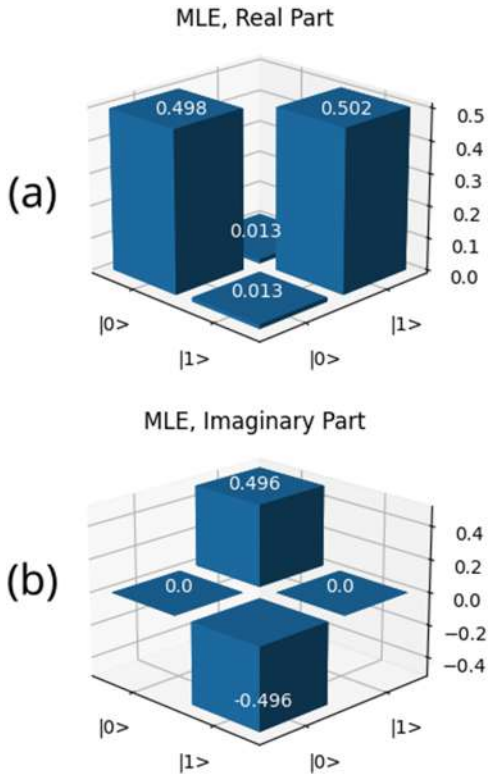

FIG. 16. Quantum State Tomography for $\frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle)$. The figure shows the (a) real and (b) imaginary parts of the Maximum Likelihood Estimate (MLE) fit. A $\pi/2$-pulse is applied before prerotation. The state fidelity of MLE is 99.8%.

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.

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. 43 with some modifications. Figure 18 shows a simplified schematic of the setup. For clarity, details such as 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 $\mathrm{nV}/\sqrt{\mathrm{Hz}}$ at 1 kHz. The current output of the amplifier is stabilized using a sense resistor, R_{sense}, that converts the current to a voltage, V_{sense}. Subtracting V_{sense} from a given set voltage, V_{set}, generates 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_{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_{sense}. For the subsequent determination of the feedback signal, another AD8422 chip is used to obtain $V_{\mathrm{sense}} - V_{\mathrm{set}}$.

For generating the set voltage V_{set}, we are using an AD5063 (Analog Devices, Inc.) DAC, which has a resolution of 16 bits and is programmable via a serial interface. 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 $-V_{\mathrm{ref}}$ to $+V_{\mathrm{ref}}$, where V_{ref} is the reference voltage supplied to the DAC. It is crucial that V_{ref} has low noise and low drift, as it directly influences the output current of the source. To generate V_{ref}, we are using an ADR4520 (Analog Devices, Inc.) voltage reference that 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_{set} from −2.048 to +2.048 V.

In our supply design, the current range is given by the sense resistor and the maximum absolute value of V_{set}. For our choice, we get a current range of $\pm V_{\mathrm{ref}}/R_{\mathrm{sense}} = \pm 4.096$ 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 − $V_{\mathrm{ref}} \approx 9.9$ 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 behavior by measuring the Allan deviation of the

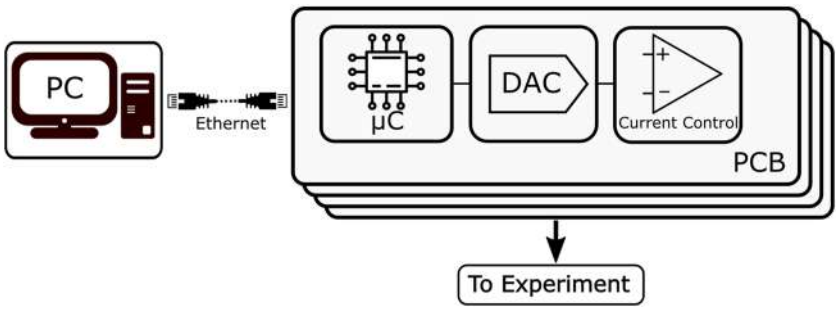

FIG. 17. Working principle of the DC current source. A host PC sends current value data to a microcontroller via an Ethernet interface. Subsequently, the microcontroller changes the setpoint of the current control circuit via reprogramming the DAC.

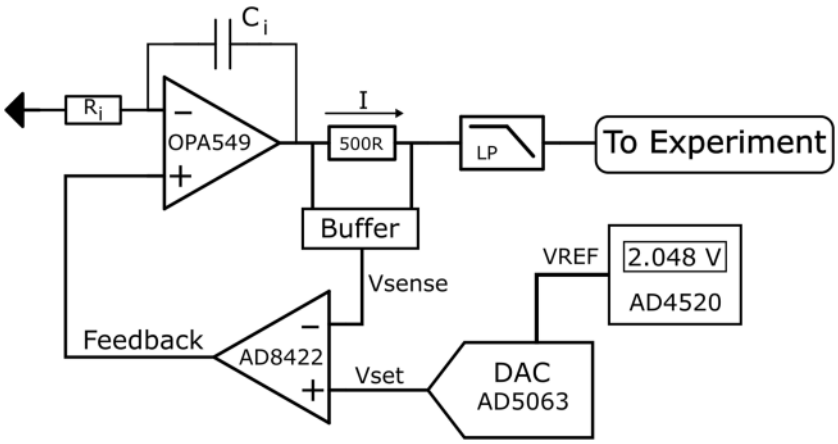

FIG. 18. Current control circuit. The output current of an operational amplifier is stabilized by integral feedback. The set-point is generated by a DAC, which is referenced to a precise voltage reference. Details like supply voltages, decoupling capacitors, and connectors have been omitted for clarity.

current source. Moreover, we tested the current source in an 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 digitized 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 of noise using the Fast Fourier Transform (FFT). Figure 19 shows a typical current amplitude noise spectrum. It can be seen that the broadband noise of the supply is below 3×10^{-6} 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 5.45×10^{-6} A_{rms} (0.1 Hz–10 kHz). Comparing Fig. 19 to that in Ref. 43, our setup is most likely limited by the noise floor of the oscilloscope. This, however, represents the upper limit of our current source noise density.

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. Figure 20 (blue line) shows the Allan deviation of the current for an 18-h long measurement. It can be seen that the bias stability of the source is about 4×10^{-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.

After characterizing the noise performance of the current source, we tested its capabilities for tuning 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 is directly reflected in the shift of the cavity resonance frequency. We measured the transmission spectrum of the cavity for various bias currents applied to the qubit. Figure 21 shows a 2D map of the cavity transmission over a current range of −2.0 to +2.0 mA. It can be seen that the frequency of the resonator shifts with applied bias current as expected.[44] Over the measured current range, we observe

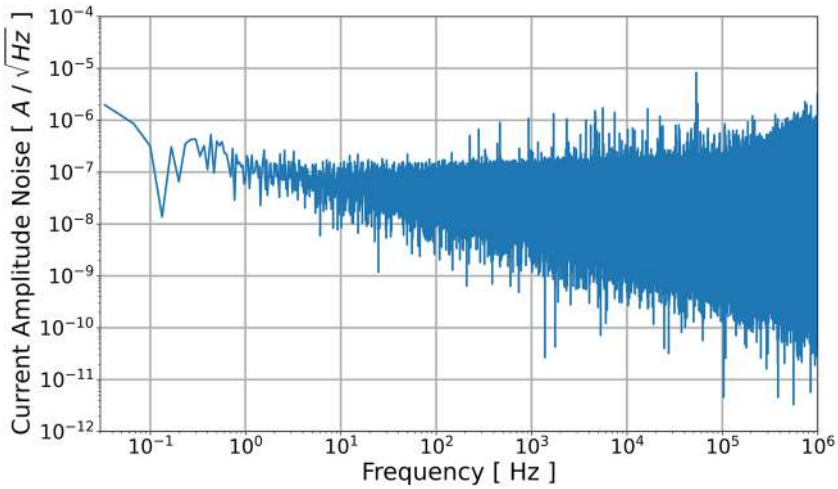

FIG. 19. High-frequency amplitude noise of the DC current source. The total noise in the band of 0.1 Hz to 10 kHz is 1.3×10^{-7} A_{rms}.

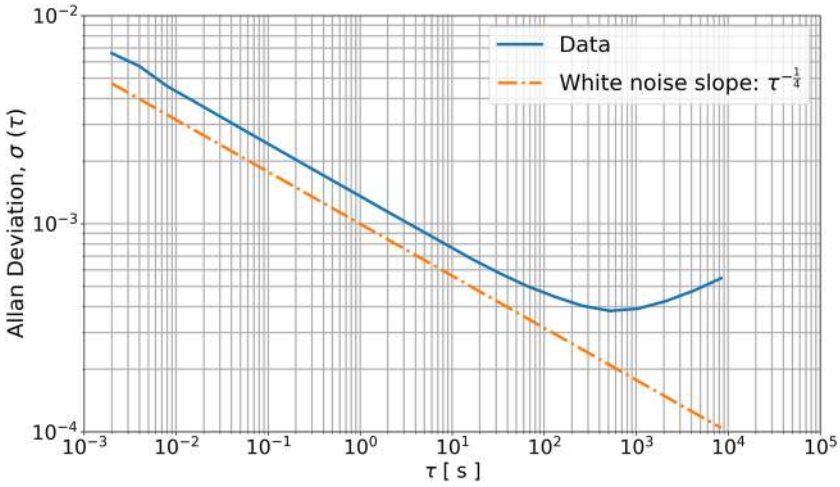

FIG. 20. Stability measurement of the current source. Depicted (solid line) is the fractional overlapping Allan deviation of an 18-h long-term current measurement. The bias stability is about 4×10^{-4} at 500 s of averaging time. Comparison to a white noise slope (dotted-dashed line) shows that for shorter averaging times, the deviation is caused by white noise.

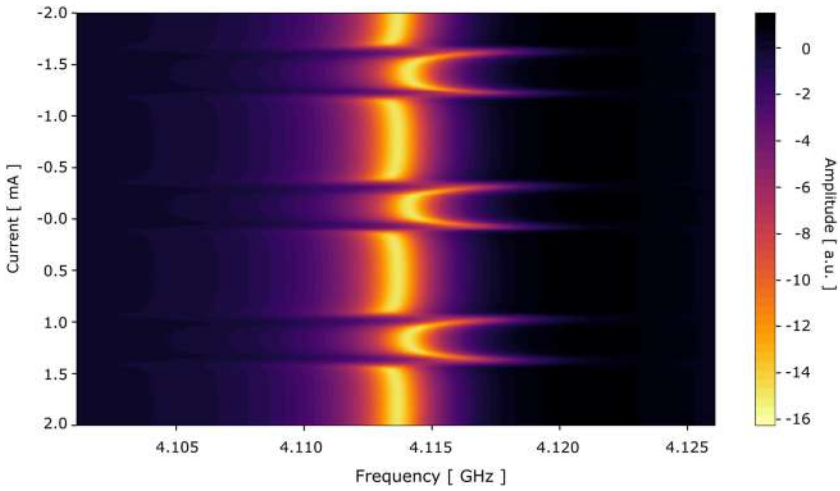

FIG. 21. Transmission spectrum of a cavity dispersively coupled to a tunable superconducting transmon qubit. Shown are the spectra for current values from −2.0 to 2.0 mA. In this range, the frequency of the qubit is going through three periodic changes. A continuous wave using a vector network analyzer was used for the readout drive.

roughly three periods of qubit frequency oscillation. Measurement of the qubit decoherence as a function of bias current is outside the scope of this work.

G. Software control and 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 Sec. II, the RFSoC board used in this project has commands for board configuration, external clock locking, channel synchronization, 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 the 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`,[45] `QCoDeS`,[46] and `Python IVI`.[47]

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` are well defined, we can expose it as a cloud service to run experiments remotely. In Fig. 22, we explain our implementation of hosting the remote experiment execution on a cloud platform. With this approach, we have the Flask server[48] and Redis[49] database act as

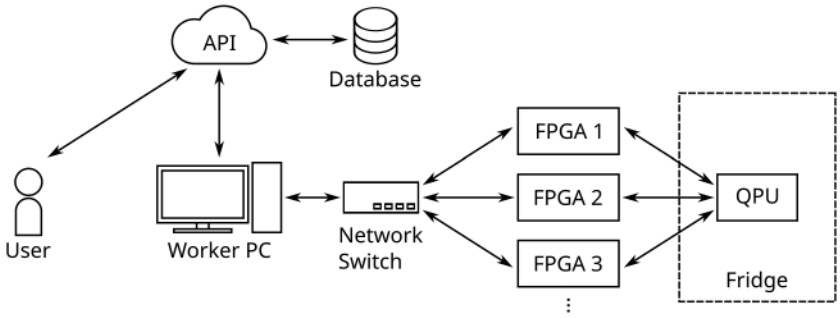

FIG. 22. A schematic overview of the cloud access layout deployed in this work. Our implementation of cloud-based experimental control. A user submits the trigger timings, number of repetitions, and the pulse sequence to a Flask[48] server hosted on an Amazon Web Services (AWS) EC2 client.[50] We provide an Application Programming Interface (API) on the Flask server for the user to submit and retrieve job results. The job is then stored on a Redis[49] database. On the worker, we have a Python program `IcarusQ-Scheduler` that queries the database for new jobs and runs them with `IcarusQ-Executor`. The results are then uploaded onto the Redis database for the user to retrieve.

a broker between the user and the worker. Neither party directly communicates with each other, and the Application Programming Interface (API) follows a strict format. Hence, we are able to create a secure environment to execute our experiments remotely.

Finally, the current layout will be interfaced to the `Qibo` framework[51] 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 minimize 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 16 channels of DAC and ADC with a 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 synchronized several RFSoC boards together using a master clock and triggers for the DAC and ADC channels. A low-noise DC source was also developed to tune the qubit based on Ref. 43. 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 utilizing the onboard mixer as an alternative waveform generation and sampling method. At the moment, the readout signals are processed by the master PC, which can be an issue when dealing with a significantly larger number of qubits. To distribute the data processing load, a future consideration is to use the DSP slices on every RFSoC to process its respective ADC data. By combining the feedback control feature and the DSP slices as a fast readout system, it may be possible to signal the switching trigger to activate and the correction pulses can be sent to the appropriate channels according to the qubit state.[10] The current setup has a relatively low duty cycle for the experiments, which is mostly due to the output data file generation and transferring it out of the RFSoC to the master PC. Improving the experiment duty cycle is currently a work in progress. Additionally, to further support a larger number of samples, we will work on memory optimization and will also consider faster DDR RAMs (such as QUAD channel RAMs) for direct streaming to/from the channels.

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[52] for improved SNR and fewer connections to room temperature electronics. This would provide an alternative implementation of cryogenic control electronics based on commercially available devices, in contrast to other possible technologies such as Josephson arbitrary waveform synthesizer[53] or single-flux technology (SFQ) pulse drivers.[54] 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, in part, by the Air Force Research Laboratory (AFRL) under Contract No. FA9550-19-S-0003.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Kun Hee Park: Conceptualization (equal); Data curation (equal); Formal analysis (equal); Investigation (equal); Methodology (equal); Software (equal); Visualization (equal); Writing – original draft (lead); Writing – review & editing (lead). **Yung Szen Yap**: Conceptualization (equal); Data curation (equal); Formal analysis (equal); Investigation (equal); Methodology (equal); Software (equal); Visualization (equal); Writing – original draft (lead); Writing – review & editing (lead). **Yuanzheng Paul Tan**: Conceptualization (equal); Data curation (equal); Formal analysis (equal); Investigation (equal);

Methodology (equal); Software (equal); Visualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Christoph Hufnagel**: Conceptualization (equal); Formal analysis (equal); Investigation (equal); Methodology (equal); Software (equal); Visualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Long Hoang Nguyen**: Resources (equal); Software (equal); Visualization (equal). **Karn Hwa Lau**: Software (equal); Writing – original draft (equal). **Patrick Bore**: Data curation (equal); Formal analysis (equal); Investigation (equal); Software (equal); Writing – review & editing (equal). **Stavros Efthymiou**: Writing – original draft (equal). **Stefano Carrazza**: Writing – original draft (equal). **Rangga P. Budoyo**: Resources (equal). **Rainer Dumke**: Conceptualization (lead); Funding acquisition (lead); Methodology (lead); Project administration (lead); Supervision (lead); Writing – original draft (equal); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

APPENDIX: GAUSS SUM FACTORIZATION

Gauss sum factorization is an algorithm using qubit superposition to factorize large numbers.[55] While the current implementation for superconducting qubits does not yield an advantage over classical factorization algorithms, this algorithm can be used to demonstrate qubit control.

Using the π-pulse and $\pi/2$-pulse from the earlier sections, we apply the factorization algorithm (Fig. 23) using the ICARUS-Q RFSoC for $N = 263\,193$ and $M = 15$.

We inspect the discernibility $\mathcal{D}$ between factors and non-factors. The greater the value of $\mathcal{D}$, the easier it is to identify actual factors over non-factors. $\mathcal{D}$ is determined by the difference between the probabilities for the worst performing factor and the non-factor closest to 1. In an earlier work,[42] we calculated the upper limit of $\mathcal{D}$ as 0.67. In this experiment, we have obtained a value of $\mathcal{D}$ as 0.195 (see Fig. 24) compared to the previously reported experimental value of 0.4. The difference between these two experiments is the qubits and the number of measurements per data point, which causes the different discernibility. Hence, we have successfully demonstrated the execution of the factorization algorithm on the ICARUS-Q platform.

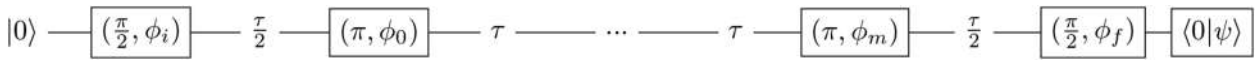

FIG. 23. Pulse sequence for Gauss sum factorization. A $\pi/2$-pulse phase shifted by $\pi/2$ is used to rotate the qubit to the superposition state. Next, a train of m π−pulses is applied to the qubit. The phase of the first π-pulse is 0 and that of the k-th pulse is $(-1)^k\pi(2k-1)N/I$ where N is the number to factorize and I is the trial factor. Finally, a $\pi/2$-pulse phase shifted by $\pi/2$ is used to rotate the system back to the computational basis. If I is a factor of N, the phase will be an integer number of π and the π-pulse train will be in phase, causing the final $\pi/2$ rotation to rotate the system to $|1\rangle$. Otherwise, an arbitrary rotation of the system will occur. The sequence is repeated for m from 1 to M, where M is the total number of pulses to be used, and the results are averaged by M. This is to reduce the impact of arbitrary non-factor rotations that may end up near $|1\rangle$. A delay τ is applied between pulses.

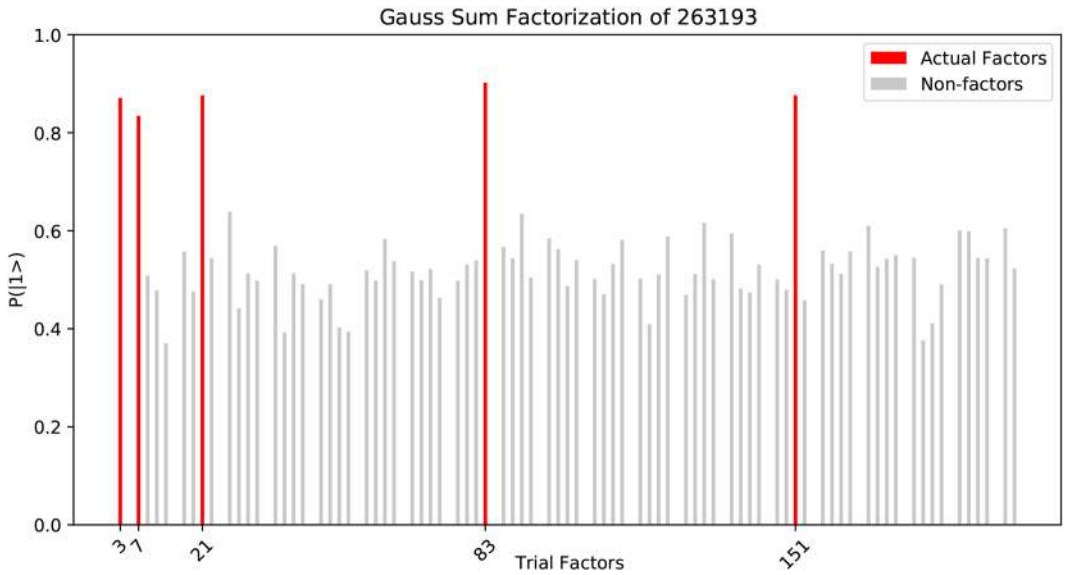

FIG. 24. Excited state probability for each trial factor up to 200 in the factorization of 263 193. We apply the preprocessing technique in Ref. 42 to skip over multiples of 2 and 5, as they provide poor contrast compared to the actual factors. The expected factors in this regime are 3, 7, 21, 83, 151 (highlighted in red). We can see that the major peaks near 1 correspond to the factors, allowing us to identify factors and composite factors. The lowest excited state probability for the factors is 0.834, while the highest excited state probability of the non-factors is 0.639 at $I = 27$. The behavior of this non-factor is discussed in more detail in Ref. 42.

REFERENCES

[1]The control systems for electron spin qubits with electron paramagnetic resonance are nearly identical to those for superconducting qubits.[4,8]

[2]M. Tseitlin, R. W. Quine, G. A. Rinard, S. S. Eaton, and G. R. Eaton, "Digital EPR with an arbitrary waveform generator and direct detection at the carrier frequency," J. Magn. Reson. **213**, 119–125 (2011).

[3]A. Doll, S. Pribitzer, R. Tschaggelar, and G. Jeschke, "Adiabatic and fast passage ultra-wideband inversion in pulsed EPR," J. Magn. Reson. **230**, 27–39 (2013).

[4]Y. S. Yap, Y. Tabuchi, M. Negoro, A. Kagawa, and M. Kitagawa, "A Ku band pulsed electron paramagnetic resonance spectrometer using an arbitrary waveform generator for quantum control experiments at millikelvin temperatures," Rev. Sci. Instrum. **86**, 063110 (2015).

[5]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," Nat. Commun. **6**, 6983 (2015).

[6]J. Raftery, A. Vrajitoarea, G. Zhang, Z. Leng, S. Srinivasan, and A. Houck, "Direct digital synthesis of microwave waveforms for quantum computing," arXiv:1703.00942 (2017).

[7]L. H. Zaw, Y. P. Tan, L. H. Nguyen, R. P. Budoyo, K. H. Park, Z. Y. Koh, A. Landra, C. Hufnagel, Y. S. Yap, T. S. Koh *et al.*, "Ghost factors in gauss sum factorization with transmon qubits," Phys. Rev. A **104**, 062606 (2021).

[8]T. Kaufmann, T. J. Keller, J. M. Franck, R. P. Barnes, S. J. Glaser, J. M. Martinis, and S. Han, "DAC-board based X-band EPR spectrometer with arbitrary waveform control," J. Magn. Reson. **235**, 95–108 (2013).

[9]C. A. Ryan, B. R. Johnson, D. Ristè, B. Donovan, and T. A. Ohki, "Hardware for dynamic quantum computing," Rev. Sci. Instrum. **88**, 104703 (2017).

[10]Y. Salathé, P. Kurpiers, T. Karg, C. Lang, C. K. Andersen, A. Akin, S. Krinner, C. Eichler, and A. Wallraff, "Low-latency digital signal processing for feedback and feedforward in quantum computing and communication," Phys. Rev. Appl. **9**, 034011 (2018).

[11]J. Lin, F.-T. Liang, Y. Xu, L.-H. Sun, C. Guo, S.-K. Liao, and C.-Z. Peng, "Scalable and customizable arbitrary waveform generator for superconducting quantum computing," AIP Adv. **9**, 115309 (2019).

[12]L. Sun, F. Liang, J. Lin, C. Guo, Y. Xu, S. Liao, and C. Peng, "Scalable self-adaptive synchronous triggering system in superconducting quantum computing," IEEE Trans. Nucl. Sci. **67**, 2148–2154 (2020).

[13]W. D. Kalfus, D. F. Lee, G. J. Ribeill, S. D. Fallek, 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 Trans. Quantum Eng. **1**, 6002612 (2020).

[14]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 Trans. Quantum Eng. **2**, 6003811 (2021).

[15]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 quantum processors," Rev. Sci. Instrum. **93**, 074701 (2022).

[16]T. M. Lanting, H.-M. Cho, J. Clarke, W. L. Holzapfel, A. T. Lee, M. Lueker, P. L. Richards, M. A. Dobbs, H. Spieler, and A. Smith, "Frequency-domain multiplexed readout of transition-edge sensor arrays with a superconducting quantum interference device," Appl. Phys. Lett. **86**, 112511 (2005).

[17]Y. Chen, D. Sank, P. O'Malley, T. White, R. Barends, B. Chiaro, J. Kelly, E. Lucero, M. Mariantoni, A. Megrant *et al.*, "Multiplexed dispersive readout of superconducting phase qubits," Appl. Phys. Lett. **101**, 182601 (2012).

[18]J. M. Hornibrook, J. I. Colless, A. C. Mahoney, X. G. Croot, S. Blanvillain, H. Lu, A. C. Gossard, and D. J. Reilly, "Frequency multiplexing for readout of spin qubits," Appl. Phys. Lett. **104**, 103108 (2014).

[19]See Xilinx website for more information.

[20]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 (RadarConf18)* (IEEE, 2018), pp. 0935–0939.

[21]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.

[22]R. Gebauer, N. Karcher, D. Gusenkova, M. Spiecker, 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," AIP Conf. Proc. **2241**, 020015 (2020).

[23]L. Stefanazzi, K. Treptow, N. Wilcer, C. Stoughton, S. Montella, C. Bradford, G. Cancelo, S. Saxena, H. Arnaldi, S. Sussman *et al.*, "The QICK (Quantum Instrumentation Control Kit): Readout and control for qubits and detectors," Rev. Sci. Instrum. **93**, 044709 (2022).

[24]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," Phys. Rev. Lett. **128**, 153603 (2022).

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

[26]SpinAPI Python wrapper for PulseBlaster programming. See SpinCore website for more information.

[27]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 able to utilize only the block RAM for various purposes and not solely for the ADC and DAC data.

[28]C. Liu, M. E. Jones, and A. C. Taylor, "Characterizing the performance of high-speed data converters for RFSoC-based radio astronomy receivers," Mon. Not. R. Astron. Soc. **501**, 5096–5104 (2021).

[29]F. Motzoi, J. M. Gambetta, P. Rebentrost, and F. K. Wilhelm, "Simple pulses for elimination of leakage in weakly nonlinear qubits," Phys. Rev. Lett. **103**, 110501 (2009).

[30]S. E. Economou and E. Barnes, "Analytical approach to swift nonleaky entangling gates in superconducting qubits," Phys. Rev. B **91**, 161405 (2015).

[31]N. Khaneja, R. Brockett, and S. J. Glaser, "Time optimal control in spin systems," Phys. Rev. A **63**, 032308 (2001).

[32]Y. Tabuchi, M. Negoro, and M. Kitagawa, "Design method of dynamical decoupling sequences integrated with optimal control theory," Phys. Rev. A **96**, 022331 (2017).

[33]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 Inf. **7**, 14 (2021).

[34]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," Nat. Commun. **8**, 1061 (2017).

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

[36]S. S. Soliman and M. D. Srinath, "Continuous and discrete signals and systems," Englewoo Cliffs, 1990, pp. 351–367.

[37]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 5.2GHz," in *2017 IEEE International Solid-State Circuits Conference (ISSCC)* (IEEE, 2017), pp. 280–281.

[38]Zynq UltraScale+ RFSoC RF Data Converter.

[39]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 *et al.*, "Quantum supremacy using a programmable superconducting processor," Nature **574**, 505–510 (2019).

[40]S. Krinner, S. Storz, P. Kurpiers, P. Magnard, J. Heinsoo, R. Keller, J. Lütolf, C. Eichler, and A. Wallraff, "Engineering cryogenic setups for 100-qubit scale superconducting circuit systems," EPJ Quantum Technol. **6**, 2 (2019).

[41]See Managing Metastability with the Intel® Quartus® Prime Software for more information.

[42]L. Zaw, Y. Tan, L. Nguyen, R. Budoyo, K. Park, Z. Koh, A. Landra, C. Hufnagel, Y. Yap, T. Koh, and R. Dumke, "Ghost factors in gauss-sum factorization with transmon qubits," Phys. Rev. A **104**, 062606 (2021).

[43]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," Rev. Sci. Instrum. **90**, 014701 (2019).

[44]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).
[45]PyVisa Team, PyVisa: Control your instruments with python, 2021.
[46]Microsoft Quantum Computing Consortium, QCoDeS, 2021.
[47]IVI Foundation, A python implementation of the interchangeable virtual instrument standard, 2021.
[48]Pallets, Flask, 2021.
[49]Redis, Ltd., Redis, 2021.
[50]Amazon, Amazon Web Services (AWS), 2021.
[51]S. Efthymiou, S. Ramos-Calderer, C. Bravo-Prieto, A. Pérez-Salinas, D. García-Martín, A. Garcia-Saez, J. I. Latorre, and S. Carrazza, "Qibo: A framework for quantum simulation with hardware acceleration," Quantum Sci. Technol. **7**, 015018 (2022).
[52]S. J. Pauka, K. Das, R. Kalra, A. Moini, Y. Yang, M. Trainer, A. Bousquet, C. Cantaloube, N. Dick, G. C. Gardner *et al.*, "A cryogenic CMOS chip for generating control signals for multiple qubits," Nat. Electron. **4**, 64–70 (2021).
[53]A. J. Sirois, M. Castellanos-Beltran, A. E. Fox, S. P. Benz, and P. F. Hopkins, "Josephson microwave sources applied to quantum information systems," IEEE Trans. Quantum Eng. **1**, 6002807 (2020).
[54]E. Leonard, M. A. Beck, J. Nelson, B. Christensen, T. Thorbeck, C. Howington, A. Opremcak, I. Pechenezhskiy, K. Dodge, N. Dupuis, M. Hutchings, J. Ku, F. Schlenker, J. Suttle, C. Wilen, S. Zhu, M. Vavilov, B. Plourde, and R. McDermott, "Digital coherent control of a superconducting qubit," Phys. Rev. Appl. **11**, 014009 (2019).
[55]M. Gilowski, T. Wendrich, T. Müller, C. Jentsch, W. Ertmer, E. M. Rasel, and W. P. Schleich, "Gauss sum factorization with cold atoms," Phys. Rev. Lett. **100**, 030201 (2008).