Journal of Low Temperature Physics
https://doi.org/10.1007/s10909-022-02858-x

Online Demodulation and Trigger for Flux-ramp Modulated SQUID Signals

N. Karcher[1] · T. Muscheid[1] · T. Wolber[1] · D. Richter[2] · C. Enss[2] · S. Kempf[3] · O. Sander[1]

Received: 22 October 2021 / Accepted: 24 August 2022

Abstract

Due to the periodic characteristics of SQUIDs, a suitable linearization technique is required for SQUID-based readout. Flux-ramp modulation is a common linearization technique and is typically applied for the readout of a microwave SQUID multiplexer as well as since recently also for dc-SQUIDs. Flux-ramp modulation requires another stage in the signal processing chain to demodulate the SQUID output signal before further processing. For cryogenic microcalorimeters, the signal contains events that are given by a fast exponentially rising and slowly exponentially decaying pulses shape. The events shall be detected by a trigger engine and recorded by a storage logic. Since the data rate can be decreased significantly by demodulation and event detection, it is desirable to do both steps on the deployed fast FPGA logic during measurement before passing the data to a general-purpose processor. In this contribution, we show the implementation of efficient multi-channel flux-ramp demodulation computed at run-time on a SoC-FPGA. Furthermore, a concept and implementation for an online trigger and buffer mechanism with its theoretical trigger loss rates depending on buffer size is presented. Both FPGA modules can be operated with up to 500 MHz clock frequency and can efficiently process 32 channels. Correct functionality and data reduction capability of the modules are demonstrated in measurements utilizing magnetic microcalorimeter irradiated with an Iron-55 source for event generation and read out by a microwave SQUID multiplexer.

Keywords Magnetic microcalorimeters · Software-defined radio · Frequency-division multiplexing · Microwave SQUID multiplexer · Flux-ramp modulation · Event trigger

✉ N. Karcher
karcher@kit.edu

[1] Institute for Data Processing and Electronics, Karlsruhe Institute of Technology, Karlsruhe, Germany

[2] Kirchhoff-Institute for Physics, Heidelberg University, Heidelberg, Germany

[3] Institute of Micro- and Nanoelectronic Systems, Karlsruhe Institute of Technology, Karlsruhe, Germany

Published online: 26 September 2022

1 Introduction

Cryogenic microcalorimeters based on paramagnetic or superconducting temperature sensors achieve excellent energy resolutions at low temperatures and enable groundbreaking experiments in various fields of science [1, 2].

However, system complexity with single-channel readout techniques merely scales linearly with the number of channels and increases the parasitic thermal load on the experimental platform at millikelvin temperatures for large arrays. For this reason, frequency-division multiplexed systems based on rf-SQUIDS [3, 4] or dc-SQUIDS [5] are used. Since the SQUID transfer functions for rf-/dc-SQUIDs are periodic, sine-like and nonlinearly dependent on the magnetic flux, the so-called flux-ramp modulation can be used for linearization with both methods [6]– in the latter, to enable multiplexing at the same time. Via an additional modulation coil, a sawtooth-shaped flux-ramp signal (period τ_{ST}) with an amplitude of several flux quanta is induced in the SQUID. An additional flux from the sensor ($\tau_{\mathrm{sig}} \gg \tau_{\mathrm{ST}}$) acts as a quasi-static flux offset within the time frame of one flux-ramp similar to a time offset of the flux-ramp and therefore results in a phase offset of the output signal. The sensor signal can be recovered from the phase offset via demodulation. Figure 1 shows how the flux-ramp modulation is combined with the multiplexing methods.

In the multiplexed readout, there is a large discrepancy between the data rate arising at the input of the AD converters (order GBs^{-1}) and the total data rate of the finally acquired signals (MBs^{-1}) [7]. Two essential steps for reducing data rates are the demodulation of the flux-ramp, where undersampling occurs; and a triggering on events so that the idle trace can be discarded. Corresponding firmware modules have been implemented for our application, the Electron Capture in Holmium-163[8] experiment and are presented in the following.

2 Flux-ramp Demodulation

The FPGA firmware for microwave SQUID-multiplexed signals initially requires band-pass filters and amplitude demodulation for channel separation [9, 10]. After a poly-phase channelizer and a digital down conversion a decimated, complex-valued envelope remains. By calculating the absolute value of the signal, the amplitude of the resonator-SQUID modulated signal can be obtained. From this point on, the processing of both multiplexing methods is similar, as the real-valued dc-SQUID signal for the flux-ramp-based multiplexing method is directly sampled by the AD converter. A major difference is that with the flux-ramp-based multiplexing method a channel contains modulated signals of multiple SQUIDs with a larger bandwidth, whereas the channel of microwave-multiplexed sensors contains a single modulated signal. If the frequency of the periodic ramp signal for SQUID modulation is known (f_r), the output signal can be approximately trimmed to a natural number of periods ($o_{\mathrm{beg,end}}$) over the course of one ramp period. By means of sine and cosine transformation, the input signal $s(n)$ is mapped by a correlation to the corresponding Fourier

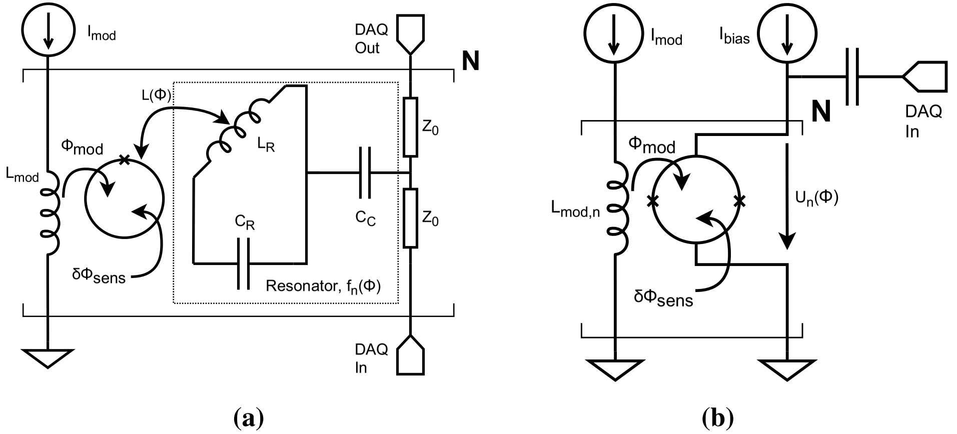

Fig. 1 Flux-ramp modulation for FDM readout concepts employing **a** rf- and **b** dc-SQUIDs. Microwave SQUID multiplexer-based readout **a** A magnetic flux shifts the resonance frequency of a microwave resonator, by changing the inductance $L(\Phi)$ of an coupled rf-SQUID. A flux modulation (I_{mod},Φ_{mod}) is applied on the SQUID through L_{mod} and forms a periodic change of the resonance frequency. An additional flux from the sensor ($\delta\Phi_{\mathrm{sens}}$) results in a phase shift of this waveform [3]. Dc-SQUID-based readout **b** A magnetic flux change results in a voltage change across current biased dc-SQUID. A flux-ramp is applied through $L_{\mathrm{n,mod}}$ on the SQUID (I_{mod},Φ_{mod}) and results in a periodic voltage change over the SQUID. An additional flux $\delta\Phi_{\mathrm{sens}}$ adds a phase to this periodic shape [5]

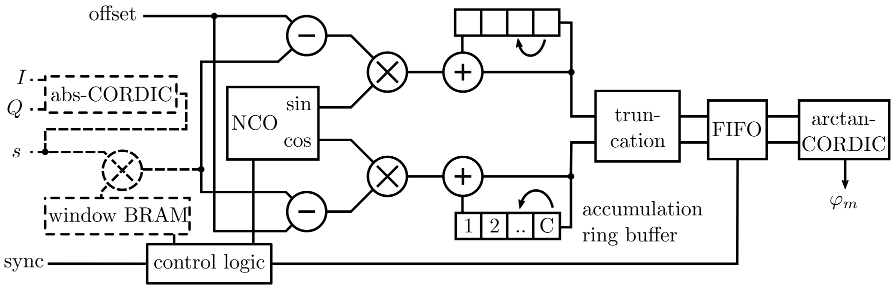

Fig. 2 FPGA Firmware Block Diagram of the flux-ramp demodulation module. The module can be used for microwave SQUID multiplex with the complex input $I\&Q$ or for dc-SQUID multiplex with real input s. Each processing cycle a different channel is calculated where C is the total amount of channels

series coefficients. Eventually the phase φ_m for each ramp index m can be obtained using the arc-tangent [6]:

$$\varphi_m = \arctan\left(\frac{\sum_{n=o_{\text{beg}}}^{(N-1)-o_{\text{end}}} s(n+mN)\cdot\cos(2\pi\frac{f_r}{f_s}n)}{\sum_{n=o_{\text{beg}}}^{(N-1)-o_{\text{end}}} s(n+mN)\cdot\sin(2\pi\frac{f_r}{f_s}n)}\right), \tag{1}$$

where N is the length of the ramp in samples and f_s the sample rate. This implies a data reduction down to the flux-ramp frequency, which is around 125 kHz in our case.

The implementation of the demodulation algorithm is shown in Fig. 2. For resource efficiency, the demodulation is performed in an interleaved, time division multiplex (TDM) fashion, where each clock cycle a different channel is processed. For the microwave SQUID multiplex setup, a clock frequency of 500 MHz is used to process 32 channels at a sampling frequency of 15.625 MHz. At the beginning, the absolute value of the input signals is formed by a pipelined[1] CORDIC IP core from Xilinx®. Optionally, the input signal can be pre-filtered by a user-defined window in order to achieve better linearization. The window-samples are stored in a BRAM block[2]. The sine and cosine values for the correlation are generated using a multi-channel numerically controlled oscillator (NCO) with direct-digital synthesis (16 bit address and amplitude width). The computation of the correlation is performed within two DSP elements (DSP48E2) (see footnote 2). Here, the *pre-adder* is used to remove a remaining DC component of the signal. Then the difference is multiplied by the sine or cosine value and added to the internal accumulator. The accumulator and offset values are stored in a ring buffer that shifts for each TDM channel. Start and end of the accumulation is controlled by a state machine. When the correlation is complete, the accumulator values leave the ring buffer and are scaled. The scaling unit takes both accumulator values and determines from these the most significant bit of the correlation results and truncates both values accordingly. Afterwards,

[1] *pipelined*: one processing cycle per sample, *sequential*: multiple cycles per sample

[2] DSP: Digital signal processors; LUT: Lookup tables; BRAM: Block random access memory

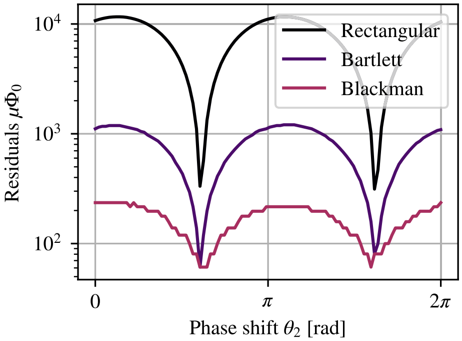

Fig. 3 Simulation of the demodulation module with two sine functions as flux-ramp with 40 and 44.4 periods (1000 samples). The figure shows the demodulation error of the first ramp for three window functions if the second signal is phase-shifted over 2π. Already with the low-cost Barlett window isolation can be improved, in this case one order of magnitude

the values are temporarily stored in a FIFO buffer and forwarded to a sequential[1] CORDIC IP core, which calculates the quotient and arc-tangent, resulting in the phase data of the channel (compare Fig. 5). Since the correlation period must be aligned to the flux-ramp, the ramp generator passes a synchronization pulse to the demodulator. This resets the NCO and state machine for accumulation. The flux-ramp demodulation for 32 channels with a abs-CORDIC, clocked with 500 MHz requires 4 DSPs, 5243 LUTs and 8 BRAM units on a Xilinx® Zynq Ultrascale+ device (see footnote 2).

The increased bandwidth for dc-SQUID-flux-ramp multiplexing method [5] demands a higher signal processing sampling rate. After a decimation stage four channels are processed within the module with 125 MHz sampling rate. The individual coupling factors of the SQUIDs lead to different modulation frequencies per channel, which makes the definition of a common correlation period difficult or even impossible. If the period can only be adjusted for one channel, spectral leakage of other channels occurs. This can be mitigated by applying a window function over the correlation period (see Fig. 3). Utilizing the windowing mechanism requires one additional DSP and flux-ramp period dependend amount of BRAM units. The total amount of resources for a four-channel module with a maximum ramp length of 1024 samples is: 3 DSP, 2218 LUTs and 9 BRAM.

3 Event Detection

The signal processing chain before the event detection processes the channels in an interleaved way (Time-Division-Multiplex). Once an event has been detected, the samples of this specific channel are extracted from the data stream and temporarily stored in an assigned memory slot. Eventually, the data packet is transferred by a DMA into a larger DDR memory. For efficiency reasons, it is desirable to keep the BRAM memory as small as possible. We assume a constant decay rate, with Poisson distributed events. Ideally, to capture all events, each channel is equipped with one memory slot and an event must be instantaneously fetched from the back-end. If less memory slots are provided, a loss of data might happen. A buffer overflow occurs

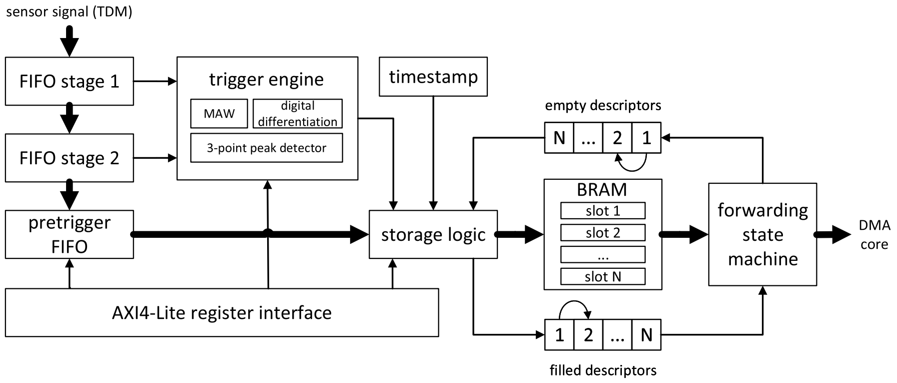

Fig. 4 FPGA firmware block diagram of the event detection module

in situations with simultaneous events on more channels than slots provided. While the decay rate is known, the buffer size can be optimized such that only a reasonable amount of events is discarded. For the probability P_b that an event is discarded, the Erlang-B formula from queuing theory can be used. P_b for an event rate $E = \lambda \tau_c$ (λ: events per second, τ_c: length of an event) and a number of limited resources or memory locations N is defined as:

$$P_b = B(E, N) = \frac{E^N}{N!} \Big/ \sum_{i=0}^{N} \frac{E^i}{i!}. \tag{2}$$

For the targeted event rate in the ECHo-experiment of 20 Bq of a length of 3.5 ms on active 20 channels only 5 slots must be instantiated in order to capture almost 99% of the events, comparable to sensors quantum efficiency. This is 75% less RAM than a full population. Although this model neglects time for data forwarding, Monte-Carlo simulations suggest that the effect is not significant.

An overview of the event detection with its functional units is displayed in Fig. 4. The sensor data stream first passes the trigger filter. It is implemented with two recursive moving average window (MAW) filters, each containing a shift register, a subtractor and an accumulator. The output of the filters is combined by another subtractor, that calculates the trigger input signal for the following 3-point trigger [11]. The trigger fires if the absolute value reaches the highest point, and it is above a predefined threshold value. In the used sensor chips one readout channel contains information of two pixels, resulting in events with positive and negative polarity. Therefore, the trigger compares both positive and negative threshold. Samples before the trigger time are buffered by a pre-trigger buffer, which is implemented by a synchronous FIFO buffer with variable length.

The event data is stored in a descriptor-based buffer, that also hands it over from the signal processing clock domain to the DMA logic clock, if required. The trigger state machine has a ring buffer with the meta data for the every input channel in the TDM. As soon as the trigger condition is met, a timestamp is stored in the channel data and a descriptor is fetched from the free descriptor FIFO buffer. The memory area defined in

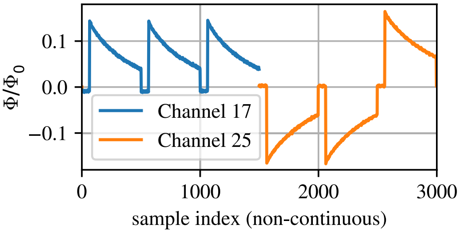

Fig. 5 Photon absorption events of a ^{55}Fe-source. Data stream of two µMux channels after frequency demultiplex, flux-ramp demodulation and event detection module with 500 stored samples per event. The x-axis represents scattered time intervals around the trigger times, with a sample time of 8 µs

the descriptor is filled with the event data for a given event length. The upper boundary of the event length is a synthesis-time parameter. As a trade-off between accuracy and resource-utilization, we limited it to 1024 samples. In this available window, samples before and after the trigger edge are saved. The long exponential decay is truncated due to the reduced signal-to-noise ratio in the tail. If the trigger is fired again during saving the data, the event is marked as *Pile-Up*. In the end the descriptor is pushed to the filled descriptor FIFO buffer. The buffer is implemented by an asynchronous two-port BRAM for the data and two FIFO shift registers for the descriptors with a clock domain crossing and handshaking. The descriptors consist of the memory address, memory length, and event meta data, such as the timestamp, trigger value and pile-up-marking. On the DMA clock domain side, the data evacuation is controlled by a state machine. This checks each clock cycle for a new descriptor shift register. If present, the machine first passes the metadata to the data stream following the event data. After the transfer is complete, the descriptor is marked as empty and is returned into the shift register for free descriptors. The resulting data stream is the sparse phase data with a header as prefix (compare Fig. 5). The data reduction depends on the event rate and length. For the given parameters the reduction lies in the range of 93%. The event detection module with five slots (N=5, rounded up to N=8), a four samples MAW, a pre-trigger FIFO of 256 samples and 32 TDM channels occupy 3 DSP, 1764 LUT and 14 BRAM units. Fully equipped, the system is designed to store and forward pulses of 800 pixels with 10 Bq each, leading to a total of 8000 events per second. The theoretical limit of transfer rate is given by the clock rate of the DMA. With 215 MHz and 1024 samples per packet, over $2{\cdot}10^5$ pulses could be transferred per second.

4 Summary

We developed an online flux-ramp demodulation and event detection, with which individual events can be extracted from a continuous data stream of flux-ramp modulated signals. The modules evaluate the acquired sensor data at the time of measurement, decimating the sensor signal down to the flux-ramp frequency and further reducing the data by a event rate-dependent factor through triggering. This corresponds to a data reduction in the order of 10^3 for our application. By estimating the blocking probability through the Erlang-B formula, the amount of BRAM needed in the trigger can be greatly reduced, by 75% in our case. We furthermore proposed a method to suppress spectral leakage in dc-SQUID-flux-ramp multiplexed channels using window functions. The method could also improve noise characteristics and spectral leakage in µMux-systems with flux-ramp modulation.

Acknowledgements This work was performed in the framework of the DFG research unit FOR2202 (funding under Grant No. En299/7-2). Nick Karcher acknowledges the support by the Doctoral School *Karlsruhe School of Elementary and Astroparticle Physics: Science and Technology*

Funding Open Access funding enabled and organized by Projekt DEAL.

Data Availability The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

References

1. S. Kempf et al., J. Low Temp. Phys. **193**(3–4), 365–379 (2018)
2. J.N. Ullom, D.A. Bennett, J. Low Temp. Phys. **28**(8), 084003 (2015)
3. J.A.B. Mates et al., Appl. Phys. Lett. **92**(2), 023514 (2008)
4. S. Kempf et al., AIP Adv. **7**(1), 015007 (2017)
5. D. Richter et al., Appl. Phys. Lett. **118**(12), 122601 (2021)
6. J.A.B. Mates et al., J. Low Temp. Phys. **167**(5–6), 707–712 (2012)
7. O. Sander et al., IEEE Trans. Nucl. Sci. **66**(7), 1204–1209 (2019)
8. L. Gastaldo et al., EPJ ST **226**(8), 1623–1694 (2017)
9. J.D. Gard et al., J. Low Temp. Phys. **193**(3), 485–497 (2018)
10. N. Karcher et al., J. Low Temp. Phys. **200**(5–6), 261–268 (2020)
11. J. Bystricky et al., IEEE Trans. Nucl. Sci. **51**(3), 351–355 (2004)