

Conference on Electronics, Telecommunications and Computers – CETC 2013

An embedded processor-based front end architecture for the Daq system of a Kinetic Inductance Detector

P. Branchini^{a,b}, A. Budano^b, L. Capasso^b and D. Marchetti^{a,b}

^a *Mathematics and Physics Department, University "Roma 3", Rome, Italy*

^b *INFN, Section "Roma Tre", Rome, Italy*

Abstract

Detecting cosmic microwave background radiation anisotropies calls for extreme precision measurement of photon energy in the range of 70 to 900 GHz. Kinetic Inductance Detectors (KIDs) are able to reduce the effects of the radiative noise. In this paper we describe the Front-End electronics architecture we adopted for the Data Acquisition System of a Kinetic Inductance Detector.

© 2014 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license

(<http://creativecommons.org/licenses/by-nc-nd/3.0/>).

Peer-review under responsibility of ISEL – Instituto Superior de Engenharia de Lisboa, Lisbon, PORTUGAL.

Keywords: Embedded Processor; Microblaze; FPGA; Kinetic Inductance Detector; Signal Processing.

Introduction

Detectors commonly used in microwave cosmology, such as spider web bolometer, have reached the radiative noise limit. Therefore, decreasing the detector's noise level does not improve the quality of measurements. If the noise has a statistic origin, a large number of pixels in the array can be exploited to reduce the noise by the familiar factor $1/\sqrt{N}$ [1]. Nevertheless, big arrays of spider web bolometers are not a viable solution, since they would need many connections; moreover, cooling the detector and placing temperature sensors over each pixel would be very difficult. Recently, Kinetic Inductance Detectors (KIDs) [2] produced using silicon technique have been shown to be useful to pursue this strategy.

1.1. Kinetic Inductance Detectors

KIDs are superconductive bolometers in which electrons are organized in Cooper pairs. Any incident photon can break a pair and generate two quasi-particles that would change the inductance of the detector. Measuring the variation of the resonant properties of the KID allows to measure the energy of the incident radiation [3].

* Corresponding author. Tel.: +39-06-57337203.

E-mail address: lcapasso@roma3.infn.it

An array of N KIDs [4] is composed of N pixels disposed on a rectangular matrix; each pixel is a resonator tuned on its own frequency. It is possible to choose such frequency by designing properly the electrical parameters (Resistance, Capacitance and Inductance) of the pixel itself. In Figure 1 it is sketched a 4×4 KID array. A feed-line gets close to each pixel and delivers a stimulus signal. If such stimulus is tuned to the resonant frequency of one of the pixels, the intensity of the signal is reduced and its phase is shifted. All the pixels can be stimulated by the same signal, containing all the resonant frequencies: the comb signal. In this way, when a photon hits any pixel, it excites only the component of the signal corresponding to the pixel's frequency [5].

A simple and effective way to obtain a comb signal is producing N sinusoids (one for each pixel's proper resonant frequency) and adding them up. We have used a Digital system to generate and sample sinusoids at regular interval of time.

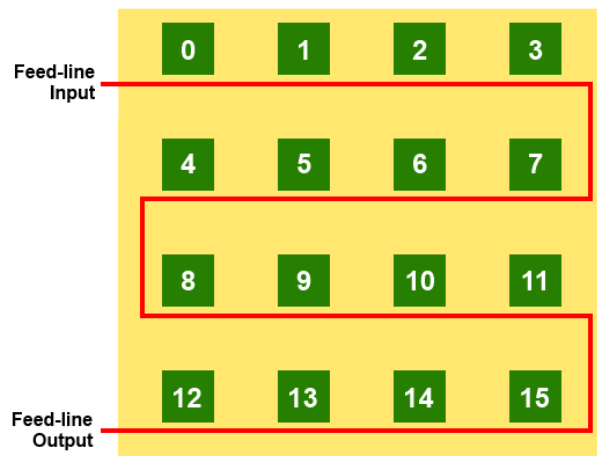


Fig. 1. Sketch of an array of KIDs.

The Nyquist theorem sets a maximum frequency we can generate which is half of the clock sampling frequency. Each resonator has about 500 kHz of FWHM (Full Width at Half Maximum), so two resonant frequencies must be parted by a minimum of 1 MHz to be clearly distinguishable. Therefore a minimum bandwidth of 100 MHz (200 MHz clock frequency) is needed to stimulate the tones of 100 pixels. Such a task cannot be accomplished by even the most performing PC. Field Programmable Gate Arrays (FPGAs) have been used in this environment because of their intrinsic capability of performing parallel computation.

The Front End system

1.2. Hardware overview

The Front End system is based on a microprocessor environment embedded inside the FPGA. The processor is the Xilinx Microblaze (32-bit, RISC) [6] and the FPGA chip we used is a Xilinx XC7VX485T from Virtex 7 family. It is hosted on a VC707 development board, that includes a 512-MB DDR3 memory module and a PCI Express bus interface [7]. The DDR3 memory is used to store the data produced by the DAQ system, while the PCI Express allows the processor to interface an external PC to the hardware environment, via DMA.

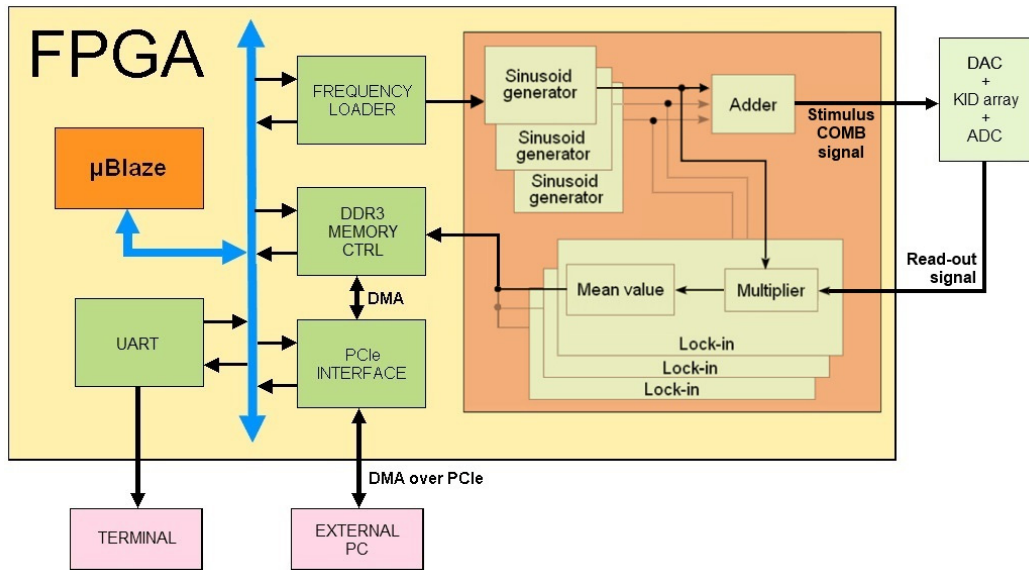


Fig. 2. The planning strategy, based on an embedded processor.

In Figure 2, a scheme of the complete project is reported. On the right side it is possible to distinguish the KID Daq electronics (sketched in light green inside a pink box), made up of 128 *Sinusoid Generators*, an *Adder* and 128 *Lock-in* components, interfaced to the external KID array via a DAC and an ADC.

On the left side, the Front End system is made up of the *Microblaze* (in Orange) connected to several devices via the *Axi_4* bus. The *Frequency Loader* is needed to transfer the frequencies to the *Sinusoid Generators*, via a custom communication protocol, that will be shown later. The *UART* controller allows the user to get information about the Daq system and to monitor the occurrence of possible errors during its activity.

The *PCI Express Interface* manages the PCI communication protocol and makes it possible to access the *DDR3* memory module via DMA, without the intervention of the processor. The external PC allows the user to send instructions to the Front End system and to download into the disk the data produced by the Daq system.

All the devices (included the processor) connected to the *Axi_4* bus are memory mapped. In the followings, the base address of any of device will be referred to as *DEVICENAME_base_addr*.

The *DDR3* is divided into two address spaces:

- the first one is made up of 512 4-bytes registers and it is located within the range [*DDR_base_addr*, *DDR_base_addr* + 2047]. It contains the data produced by the Daq system;
- also the second address space is made up of 512 4-bytes locations and its address range is [*DDR_base_addr* + 2048, *DDR_base_addr* + 4095]. The first 32-bit register is the *PC Status Register* (PCSR) and it contains flags for the Front End system: in particular, the *New Frequencies Available* (NFA) flag is asserted when a new frequency set has been loaded into the *DDR3* memory and it is ready to be transferred to the *Sinusoid Generators*. The second register is the *System Status Register* (SSR) and it contains information about the Daq and the Front End systems. The remaining 510 registers contain the frequencies to be sent to the *Sinusoid Generators*.

1.3. The algorithm

If a frequency update is needed, the external PC writes the frequencies to be sent to the Sinusoid Generators into the second address space of the *DDR3* memory. It then sets the NFA flag.

The flow chart in Figure 3 schematizes the Front End system routine.

After the start-up (reset and initializations), the processor monitors the NFA flag and checks if any new data are available from the Daq system. If the NFA flag has been set, the Frequency loader transfers the 128 new frequencies to the *Sinusoid Generators* and it then triggers the Daq system.

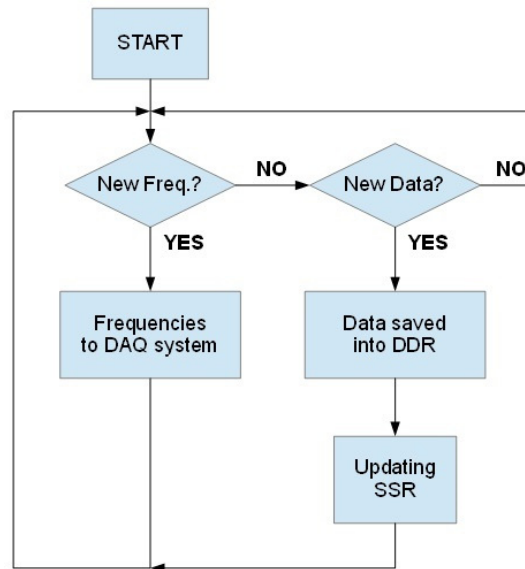


Fig. 3. Flow chart of the Front End system procedure.

At the end of the Daq data processing, the Microblaze saves the produced data into the *DDR3* memory and sets the “Data Available” code into the SSR. Eventually, the PC starts a DMA transfer and downloads the data into the PC disk.

1.4. The communication protocol to transfer the frequencies

During the start-up, the processor loads the frequencies into a buffer directly connected to the *Sinusoid Generators*. The transfer procedure is repeated every time a new set of frequencies is needed.

The peripheral that manages such communication is the *Frequency Loader*. It consists of two registers (the *Data Register* and the *Address Register*), a *Data Strobe* to validate each frequency transferred and a *Frequencies OK* flag to trigger the *Sinusoid Generators* at the end of the transfer.

In Figure 4 it is reported a sketch of the protocol.

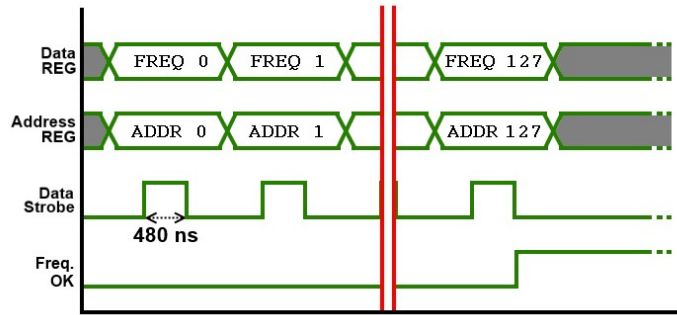


Fig. 4. Timing diagram of the Custom Transfer Protocol.

To deliver each frequency, the processor firstly loads it and its address respectively in the Data Register and the Address Register; then it asserts the Data Strobe for 480 ns: during this time, a state machine saves the received frequency into the buffer connected to the Sinusoid Generators. The Microblaze then deasserts the Data Strobe and loads the next data to be transferred into the Data Register and the Address Register. The procedure is repeated as long as the last of the 128 words has been transferred. At the end, the Frequency Loader asserts the Transfer Done signal and the Sinusoid Generators start working with the new frequency set.

Simulations

To test our project, we have developed a gate-level timing simulation. The test bench we have written for the simulator emulates the reset signals for both the KID logic circuit and the microprocessor. Moreover it saves into a log file the data produced by the KIDs. In Figures 5 and 6 several snapshots of the simulation have been reported.

Figure 5 (a) shows an overview of all the simulation we have made. It is possible to distinguish the complete communication of the frequency set to the KID detectors (activity on the *Frequencies_to_KID* bus) and, when the transfer completes at 220 μ s, the assertion of the signals *Frequencies OK* and *Write Enable*.

Figure 5 (b) shows the first frequencies sent to the KIDs by the processor. In the simulation, we have unified the *Data Register*, the *Address Register* and the *Data Strobe* signals into a single waveform: *Frequencies_to_KID*. Here they follow the waveform bit-fields reserved to the above buses:

- Bit from 0 to 19: *Data Register* signal;
- Bit from 20 to 27: *Address Register* signal;
- Bit 31: *Data Strobe* signal.

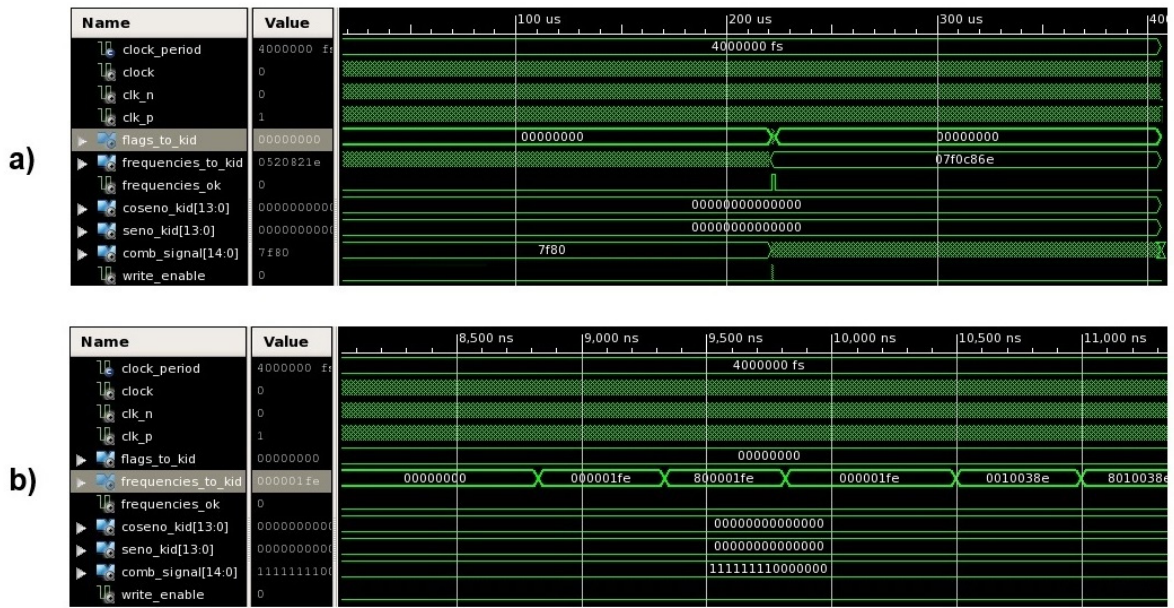


Fig. 5. (a) Overview of the whole simulation; (b) first frequencies sent to the KID detectors.



Fig. 6. (a) First frequencies sent to the KID detectors; (b) zoom view of the *Write Enable* assertion.

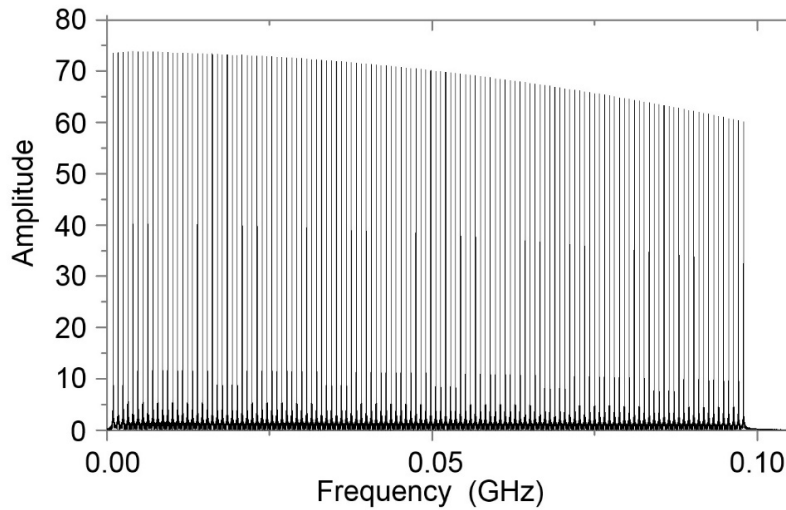


Fig. 7. Spectrum of the KID Stimulus signal.

As described in the previous paragraph, the processor firstly sets the frequency and its address, then it asserts the *Data Strobe* signal for 480 ns. In the Figure 5 (b) it's clear that the first frequency sent is 1FE (hex notation) and the corresponding address is 0, while the second frequency is 38E and its address is 1. Each frequency is transferred in about 1.7 μ s.

Figure 6 (a) and 6 (b) show the last frequency transferred to the KID (Frequency = 86E; Address = 127 in hex notation) and a zoom view around the *Write Enable* signal respectively. It is possible to see the assertion of the *Frequencies OK* signal and the corresponding assertion of the *Write Enable* signal after a clock cycle. This last is provided from the electronics managing the *lock-in* components and it triggers the elaboration of the comb signal travelling on the feed-line toward the KID array. In fact, after a while, it is possible to see activity on the comb signal waveform.

In Figure 7 it is reported the frequency spectrum of the KID stimulus signal, obtained by doing a Fast Fourier Transform of the comb signal created by the *Sinusoid* Generators and the *adder*.

Conclusions

We have developed a new strategy to test and collect data from a KID array. The system is based on a soft-core processor embedded in a Xilinx FPGA (Virtex 7). It is interfaced to a PC via a PCI bus and can be monitored via a terminal which exploits an UART interface. Our system has been fully simulated and it is now ready to be implemented on a real telescope KID array.

References

- [1] Sergeev AV, Mitin VV, Karasik BS. Ultrasensitive hot-electron kinetic-inductance detectors operating well below the superconducting transition. *Applied Physics Letters*, Volume 80, Number 5. 4 February 2002.
- [2] Calvo M. Development of Kinetic Inductance Detector for the study of the Cosmic Microwave Background Polarization. Ph.D. thesis, "Sapienza", University of Rome, IT (2008).
- [3] Baselmans J. Kinetic Inductance Detectors. *Journal of Low Temperature Physics*, May 2012, Volume 167, Issue 3-4, pp 292-304.
- [4] Doyle S. Lumped Element Kinetic Inductance Detector. Ph.D. thesis, University of Cardiff, UK (2008).
- [5] Bourrion O, Bideaud A, Benoit A, Cruciani A, Macias-Perez JF, Monfardini A, Roesch M, Swenson L, Vescovi C. Electronics and data acquisition demonstrator for a kinetic inductance camera. 2011 JINST 6 P06012. DOI: 10.1088/1748-0221/6/06/P06012.
- [6] <http://www.xilinx.com/tools/microblaze.htm>
- [7] <http://www.xilinx.com/products/boards-and-kits/EK-V7-VC707-G.htm>