32nd URSI GASS, Montreal, 19-26 August 2017

Design of a LabVIEW-Based Polyphase Filter Bank Spectrometer for Radio Astronomy using FlexRIO FPGA Technology and CUDA-enabled GPU

Rodrigo G. Freundt* (1) and Jorge A. Heraud(1)
(1) Pontificia Universidad Católica del Perú, Lima, Peru (Institute for Radio Astronomy)

Abstract

The spectrometer is the most important back-end in single antenna radio astronomy observations. The state-of-the-art designs for this type of instruments propose to reduce the effects of spectral leakage by using the Polyphase Filter Bank (PFB) technique and to achieve wideband and high resolution by using digital, reconfigurable, and high-performance computing hardware, such as commercial-available FPGA and GPU. We herein describe the development of a prototype PFB spectrometer using an integrated hardware and software development environment from National Instruments (NI), along with FlexRIO FPGA technology for acquisition and CUDA-enabled GPU card for intensive processing. The results show that the proposed design, configured as a 4-tap PFB with 2^{22} channels and 200MHz instantaneous bandwidth, is 95 times faster than a CPU implementation. It can be concluded that the hardware and software design approach used to prototype the spectrometer in this paper must be further studied within radio astronomy applications.

1. Introduction

A radio telescope collects radiation from radio sources millions of light years away, transforming those radio waves into voltage signals at the output of the front-end. The digital back-end processes those signals in order to obtain valuable information about certain source parameters. In single dish radio astronomy, the spectrometer is the most important back-end since it allows analyzing the frequency components of astronomical signals, such as narrowband spectral line emissions and wideband synchrotron radiation.

The Polyphase Filter Banks (PFB) technique is broadly used in radio astronomy contexts in order to overcome some disadvantages of the straightforward application of the FFT algorithm. One side effect, known as spectral leakage, can cause that a strong radio frequency interference (RFI) in an output channel can mask fainter astronomical signals of interest in nearby channels (Figure 1). This technique reduces spectral leakage by improving the channel frequency response (Figure 2) through the addition of extra time samples and decimation of the FFT output [1], which could be implemented in a computationally efficient manner by placing a decimating polyphase FIR structure before the FFT [2].

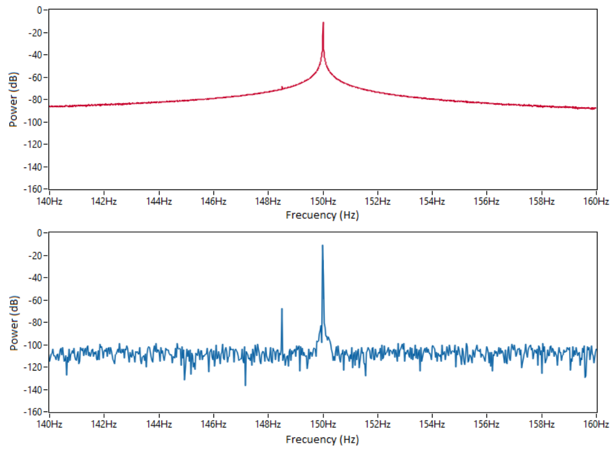

Figure 1. Simulation in LabVIEW of PFB technique benefits in radio astronomy. In the direct FFT case (red), the simulated astronomic signal of interest is masked by a strong RFI, while in the PFB case (blue), spectrum is less affected by spectral leakage and thus better resolved.

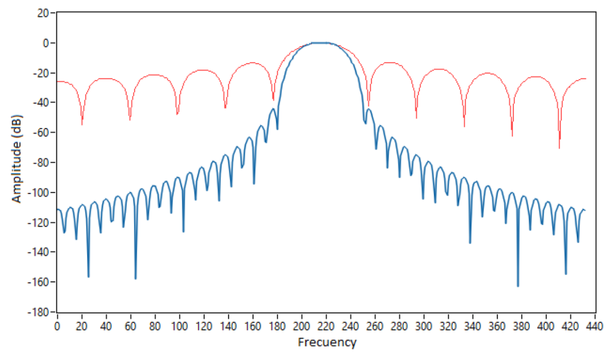

Figure 2. Single channel frequency response comparison of PFB (blue) and direct FFT (red).

On the other hand, FPGA technology has been widely used in digital signal processing instruments because of its capacity to develop parallel algorithms for use in physically parallel hardware. However, as design complexity rises, the development of this hardware at Register Transfer Level (RTL) becomes much more complicated and needs experience in digital hardware design and months of development time. Nowadays, engineers and scientists prefer to employ High Level Synthesis (HLS) languages and tools, which make programming easier and more efficient; therefore, powerful applications can be written, modified, and debugged easily with little or no hardware design experience [3].

For instance, the CASPER[1] community has spread out open source, platform-independent, DSP libraries that can be reused and scaled, along with modular, connectible, upgradable hardware components [4]. These software libraries are based on Simulink drag-and-drop components that enable researchers to rapidly design and deploy radio astronomy instrumentation using, e.g., ROACH boards [5].

2. The LabVIEW and CUDA Approach

A commercial available alternative to design and prototype digital instruments is to use the LabVIEW dataflow-oriented graphic programming language in order to design circuitry functionality and employ the compiler to translate it into a FPGA-executable circuit. LabVIEW is a development environment, owner of National Instruments, which makes use of a graphic programming language called "G". Programs developed with this platform are known as Virtual Instruments (VI). LabVIEW FPGA is an extension of LabVIEW that uses the compilation tools of Xilinx in order to translate the graphic code into FPGA code.

On the other hand, the Graphics Processing Unit (GPU) has become an integral part of astronomical instrumentation, enabling high-performance online data reduction and accelerated online signal processing [6]. The most common general-purpose GPU programming platform is Compute Unified Device Architecture (CUDA). CUDA lets developers access the GPU computational resources using an extended C/C++ language to invoke routines that run in parallel on GPU cores. Because of high-leveled software design and availability of CUDA optimized libraries (such as cuFFT library), this approach further reduces the design cycle and development time for spectrometers.

3. Spectrometer Processing Flow Overview

This work makes use of National Instrument's FlexRIO hardware, specifically, the NI-7966R Virtex-5 FPGA module and the NI-5792R digital receiver module with 200MHz instantaneous bandwidth and a 14 bits ADC, both installed in a PXIe-1085 PXI express chassis (Figure 3). The complex samples from the receiver are acquired by the FPGA and sent thought the PXIe bus to the chassis controller via DMA transfers. Samples are again queued to a server class computer using a LabVIEW FIFO-based network communication called Network Streams, which provides a lossless, high throughput dataflow. An NVIDIA Tesla K40c accelerator is installed in the PCI express bus of the server computer. The LabVIEW GPU Analysis Toolkit was used to take advantage of the computational resources available in the GPU within the framework of a LabVIEW virtual instrument.

Figure 3. Control room at the Institute for Radio Astronomy. The FlexRIO receiver module indicated by the red rectangle is directly connected to the FPGA module, which cannot be observed because it is installed inside the PXIe chassis. The NIXSYS server (64 cores, 64 GB of RAM) is indicated in green.

The whole pack of hardware devices already mentioned (along with a cesium standard atomic clock for VLBI and a RAID recording system, both observed in Figure 3) were acquired by INRAS thanks to the *"Equipamiento Científico para Laboratorios"* contest organized by FINCyT (Innovate-Perú), under the project: *"Equipamiento Científico para Radiociencia"*, with agreement number: 127-ECL-2014. The Tesla K40c GPU was donated by NVIDIA.

A data flow diagram is shown in Figure 4. The acquisition stage is composed of LabVIEW designs that run in FPGA and PXIe controller in order to acquire samples from the receiver module. IQ imbalance correction, decimation (if needed) and fine tuning are done in the FPGA.

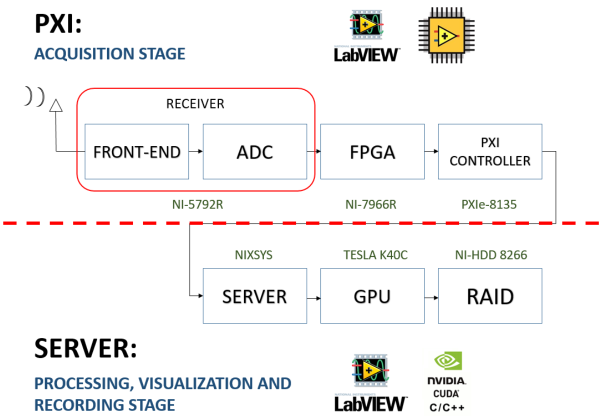

Figure 4. Dataflow diagram of the spectrometer. Time samples are downloaded to the GPU memory where all processing is done. The integrated spectrum is then

---
[1] The Collaboration for Astronomy Signal Processing and Electronics Research: https://casper.berkeley.edu/

uploaded from device memory to host memory for visualization and recording.

As mentioned before, in order to satisfy the huge computing demand, a commercial GPU with CUDA capability has been used. Because of the dataflow-oriented programming and high-level libraries, LabVIEW is ideal to accelerate the GPU design process. In order to make good use of GPU's computational resources to accelerate the application, a dynamic library (DLL) written in CUDA C was used to call customized CUDA kernels using wrapped functions within LabVIEW environment via GPU Analysis Toolkit for LabVIEW.

A throughput analysis of the system is shown in Figure 5. At full rate (200MS/s), and considering four bytes per complex sample, we get 800 MB/s of data throughput at FPGA output. It is clear that there is no bottleneck among devices. The 10GbE link between PXI controller and server computer is established by a 10GbE module for PXIe and a PCIe 10GbE Network Interface.

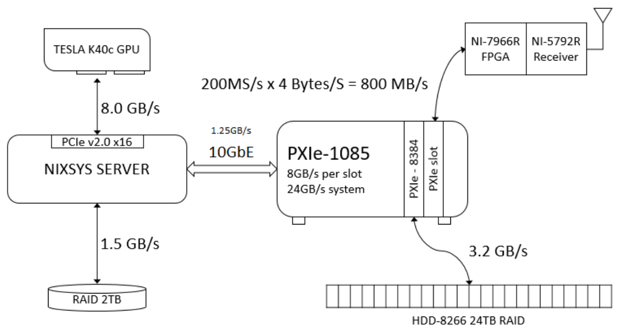

Figure 5. Throughput analysis of the spectrometer.

4. Benchmarking Results

In order to validate the design outlined before, a CPU based implementation of the PFB spectrometer, configured as a 4-tap PFB, was programmed using LabVIEW, and an execution time vs. number of channels graphic was plotted (Figure 6).

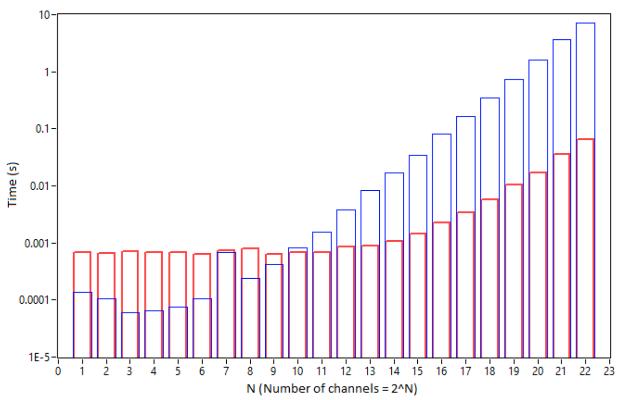

Figure 6. Execution time vs. number of channels for CPU (blue) and GPU (red) implementations.

For instance, the average execution time for 2^{22} spectral channels is 0.7 seconds, while the time needed to acquire the 2^{24} complex samples at 200 MS/s is 0.8 seconds. That means that the designed spectrometer would process at full rate in a real-time, bottleneck-free manner.

An acceleration factor plot is shown in Figure 7. For 2^{22} or 4 million spectral channel and 200MHz bandwidth, the GPU implementation performed is 95 times faster than the CPU implementation.

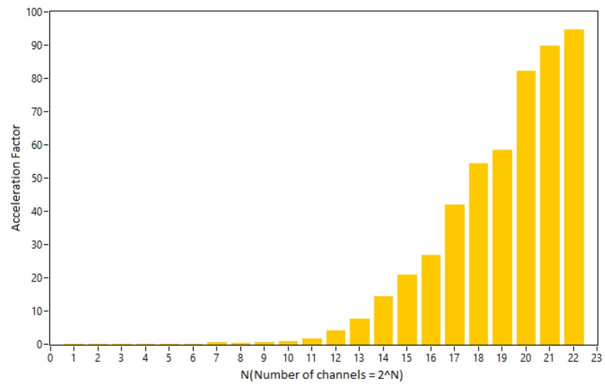

Figure 7. Acceleration factor vs. number of channels.

5. Conclusions

We can conclude that the hardware and software design approach used to prototype the spectrometer in this paper accelerates the design cycle and development time for this type of instruments. Therefore, the LabVIEW-based PFB spectrometers proved that the design approach used in this paper must be further studied within radio astronomy applications.

6. Future work

The efforts described above are part of the construction project of a 20-meter diameter radio telescope, RT-20, at the Institute for Radio Astronomy (INRAS) of the Pontificia Universidad Catolica del Peru (PUCP - Lima, Peru) [7]. Since the processing, visualization and recording stages are separated and independent of the acquisition stage, the next step is to design a front-end that combines all conversions from RF to baseband, from analog to digital, and from copper to fiber into one compact device at radio telescope's feed, possibly, using standalone USRP (Universal Software Radio Peripheral) technology to acquire complex baseband samples and to send them to the PFB spectrometer designed through a 10GbE fiber link.

7. Acknowledgements

The authors would like to thank the funding from FINCyT (*Fondo para la Innovación, la Ciencia y la Tecnología*) of the Peruvian government used in the acquisition of the hardware and software aforementioned. We also wish to thank to NVIDIA Inc. for the donation of the GPU card used in this paper, Southern Peru Copper Corporation for its donation for the 20-meter radio telescope to be built and Aceros Arequipa S.A. for the donation of 45 tons of structural steel. LabVIEW and FlexRIO are trademarks and/or registered trademarks of National Instruments.

8. References

1. C. Harris and K. Haines, "A Mathematical Review of Polyphase Filterbank Implementations for Radio Astronomy," *Publications of the Astronomical Society of Australia*, 2011, 28, 317–322.

2. Danny C. Price, "Spectrometers and Polyphase Filterbanks in Radio Astronomy," to be published in *the WSPC Handbook of Astronomical Instrumentation,* Jul 2016. arXiv 1607.03579.

3. L. Daoud, D. Zydek, and H. Selvaraj, "A Survey of High Level Synthesis Languages, Tools, and Compilers for Reconfigurable High Performance Computing," *Advances in Intelligent Systems and Computing*, 2014.

4. D. Werthimer, "The CASPER collaboration for high-performance open source digital radio astronomy instrumentation", *General Assembly and Scientific Symposium, 2011 XXXth URSI*, 2011.

5. F. Kapp, "Accelerating the Design of Radio Astronomy Instruments in South Africa and Worldwide with Simulink", *MathWorks Technical Articles and Newsletters*, 2013.

6. J. Chennamangalam et al, "A GPU-Based Wide-Band Radio Spectrometer", *Publications of the Astronomical Society of Australia*, 2014.

7. Jorge A. Heraud, Rosendo Franco, Felipe E. Carrero, Carla A. Arce, and Victor A. Centa, "RT-20, The 20-meter diameter radio telescope in Lima, Peru," *XXXI URSI General Assembly and Scientific Symposium*, pp. 16-23, 2014.