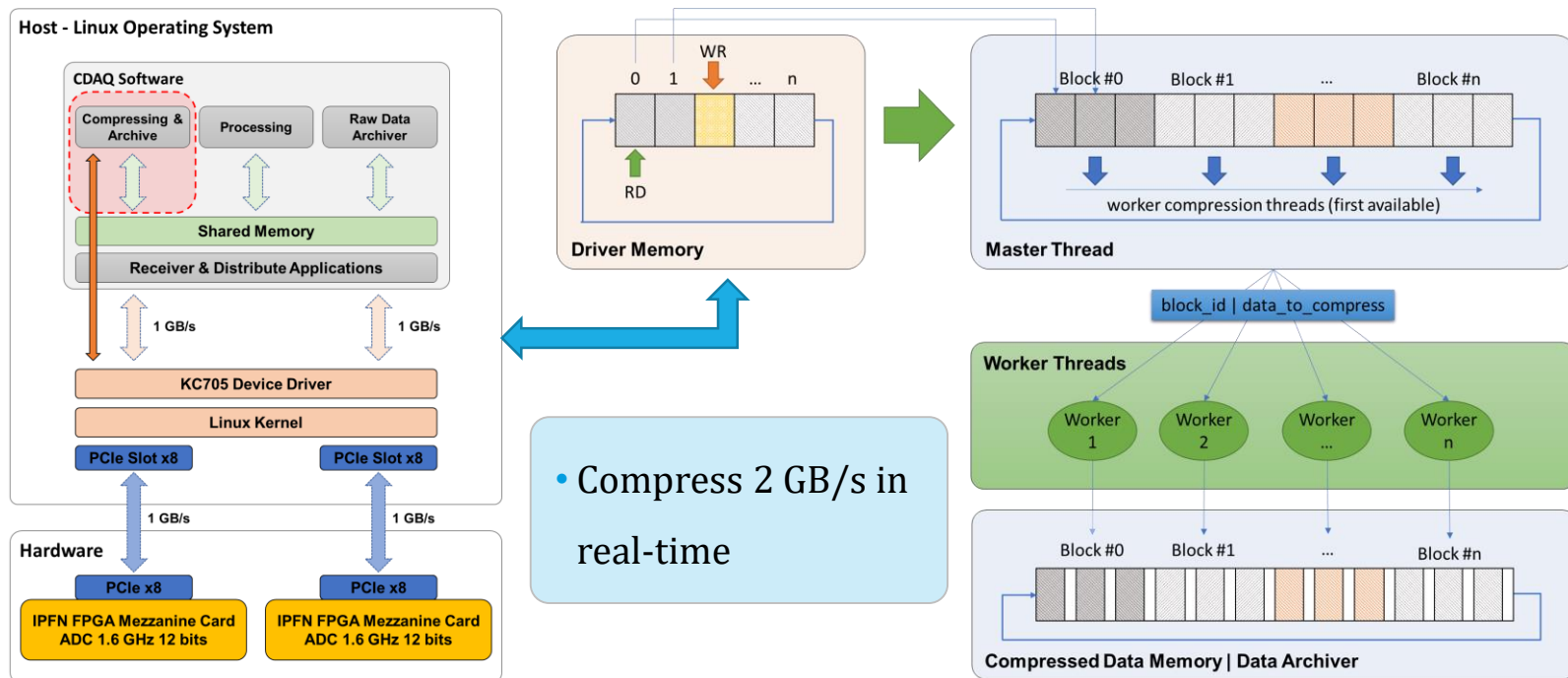



Real-time data compression for data acquisition applied to the ITER radial neutron camera



Motivation and Background

System Architecture


Real-time data compression for data acquisition systems applied to the ITER Radial Neutron Camera

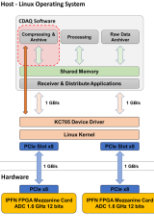
B. Santos¹, N. Cruz¹, A. Fernandes¹, P.F. Carvalho¹, J. Sousa¹, B. Gonçalves¹, M. Riva², F. Pollastrone², C. Centioli³, D. Marocco², B. Esposito², C.M.B. Correia³, B. Brichard⁴ and R.C. Pereira¹

¹Instituto de Plasmas e Fusão Nuclear, Instituto Superior Técnico, Universidade de Lisboa, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal
²ENEA C. R. Frascati, Dipartimento FSN, via E. Fermi 45, 00044 Frascati (Roma), Italy
³LIBHy-LIC, Department of Physics, University of Coimbra, P-3004-516 Coimbra, Portugal
⁴Fusion for Energy, Barcelona, Spain.

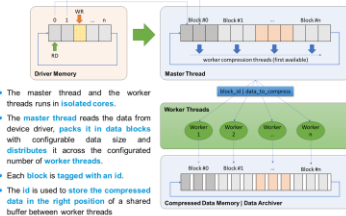
Introduction

- To achieve the aim of the ITER Radial Neutron Camera Diagnostic, the data acquisition prototype must be compliant with a sustained 2 MHz peak event for each channel with 128 samples of 16 bits per event. The data is acquired and processed using an IPFN FPGA Mezzanine Card (FMC-ADC-1600) with 2 digitizer channels of 12-bit resolution sampling up to 1.6 GSamples/s in a PCIe evaluation board from Xilinx (KC705) installed in the host PC.
- The acquired data in the event-based data-path is streamed to the host through the PCIe x8 Direct Memory Access (DMA) and the maximum data throughput per channel is ~0.5 GB/s of raw data (event based), ~1 GB/s per digitizer and up to ~1.6 GB/s in continuous mode.
- The prototype architecture comprises one host PC with two KC705 modules and four channels, producing up to ~2 GB/s in event mode and up to ~3.2 GB/s in continuous mode.
- Real-time data compression to reduce data throughput from the host to ITER databases was evaluated using the LZX lossless compression algorithm, which provides compression speed up to 400 MB/s per core.
- This contribution presents the architecture, implementation and test of the parallel real-time data compression system running in multiple isolated CPU cores. The average space saving and the performance results for long pulse acquisitions up to 30 minutes, using different data block sizes and different number of CPUs, are also presented.

System Architecture



- The Host Computer runs Scientific Linux 7 as Operating System with kernel 3.10-r1 and LZX version 1.5.
- Intel® Core™ i7-5930K@3.50 GHz (12 cores)
- 256 GB SSD and 64 GB of RAM
- To test the compression branch maximum performance, in the presented tests, the compression application is directly connected to the device driver.
- The device driver implements an internal circular buffer to store temporarily the data transferred from the hardware, until it is read from the consumer applications.
- The shared memory layer is only used when several clients need the data at same time.
- Introducing the shared memory layer, the compression performance must not be compromised.



- The master thread and the worker threads runs in isolated cores.
- The master thread reads the data from device driver, packs it in data blocks with configurable data size and distributes it across the configured number of worker threads.
- Each block is tagged with an id.
- The id is used to store the compressed data in the right position of a shared buffer between worker threads.
- The worker threads can store the compressed data in a memory buffer or deliver it through the network directly to the data archiver.

Results

- The minimum number of needed cores per acquisition rate (with no data loss) is presented in the table below.
- The device driver and the master thread implement algorithms to check the data loss.
- The input signal was a pulse type signal (gamma distribution) from a waveform generator with different pulse width using a 10 MB block size during 60 minutes.

Channels	Rate (MB/s)	Time Loss (Number of CPUs)					CPU Used (Number of Cores)		Compression		Space Saving	Data Rate
		1	2	3	4	5	Worker Thread	Master Thread	Speed (MB/s)	Ratio		
1	128	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	1024.00	100.00%	1.00	1.00
1	128	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	1024.00	100.00%	1.00	1.00
1	128	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	1024.00	100.00%	1.00	1.00
2	256	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	1024.00	100.00%	1.00	1.00
2	256	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	1024.00	100.00%	1.00	1.00
2	256	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	1024.00	100.00%	1.00	1.00
2	256	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	1024.00	100.00%	1.00	1.00

- The figures below present the CPU and memory usage during 3 runs of 10 seconds duration with 1024 MB/s of data acquisition rate. The left snapshot presents the state of the system during an acquisition from one board and the right from two boards simultaneously (2x128 MB/s per board).



- The chart in the right presents the relation between space saving and pulse width for signals using one and two ADCs.



- The table below presents the test of different data block sizes with same input signal and configurations (1 Channel, Pulse Width 128, Data Rate 512 MB/s).




Block Size (MB)	CPU Utilization		Compression		Space Saving
	Worker Thread	Master Thread	Speed (MB/s)	Ratio	
1	80.00%	0.00%	200.00	100.00%	0.00%
10	80.00%	0.00%	200.00	100.00%	0.00%
100	80.00%	0.00%	200.00	100.00%	0.00%
200	80.00%	0.00%	200.00	100.00%	0.00%

Conclusions

- The presented architecture is scalable and adjustable. The number of worker threads can be configured to comply with different algorithms and data throughput.
- The stress test shows a stable solution during 60 minutes acquisitions with acquisition rates at 1.5 GB/s, using 6 worker threads in parallel.
- The system was also tested in Fedora Linux 27, kernel 4.16 with similar results.
- It was identified a relation between space saving and pulse width.
- The data block size did not improve the space saving, however the standard deviation of the compression speed during the pulse is reduced.
- To compress 1 GB/s from one board, a minimum of 5 cores is needed (1 master and 4 worker threads).
- Based on the tests with individual boards:
 - The system needs 14 cores to compress 1 GB/s from two boards in real-time (2 cores for operating system, 2 cores for the device driver, 2 cores for master thread and 8 cores for the worker threads).
- The preliminary tests with two boards simultaneously showed a performance decreasing in the minimum number of CPUs (With 512 MB/s of data rate acquisition the system needs 3 cores, instead of 2 with a single board). Tests with two hardware modules acquiring simultaneously are scheduled to a future task.

Main References:
 [1] B. Santos et al., Real-time performance for the performance of the ITER Radial Neutron Camera, Fusion Eng. Des. 103 (2017) 1001-1006.
 [2] R.C. Pereira et al., Real-time data acquisition for the ITER Radial Neutron Camera using a generic operating system, Fusion Eng. Des. 103 (2017) 1001-1006.
 [3] LZX website: <http://lzx.sourceforge.net>

Corresponding author: bsantos@ipfn.tecnico.ulisboa.pt

P1.504, June 13th 2018 | Related Contributions: 0024 (June 13th 2018 09:00) and 0912 (June 13th 2018 16:10)

21st Real Time Conference, June 9th - 10th, Colonial Williamsburg, Virginia, United States

Results

- Stable solution
- 60 minutes at 1.5 GB/s
- 6 worker threads in parallel.
- Space Saving ≈ 35%

Conclusions

- Two boards simultaneously
- 14 cores to compress 2 GB/s

B. Santos (Presented by N. Cruz)