FPGA code for the data acquisition and real-time processing prototype of the ITER Radial Neutron Camera

A. Fernandes\textsuperscript{1*}, N. Cruz\textsuperscript{1}, B. Santos\textsuperscript{1}, P.F. Carvalho\textsuperscript{1}, J. Sousa\textsuperscript{1}, B. Gonçalves\textsuperscript{1}, M. Riva\textsuperscript{2}, F. Pollastrone\textsuperscript{2}, C. Cetioli\textsuperscript{2}, D. Marocco\textsuperscript{2}, B. Esposito\textsuperscript{2}, C.M.B.A. Correia\textsuperscript{3}, and R.C. Pereira\textsuperscript{1}

\*anaf@ipfn.tecnico.ulisboa.pt

\textsuperscript{1} Instituto de Plasmas e Fusão Nuclear, Instituto Superior Técnico, Universidade de Lisboa, 1049-001 Lisboa, Portugal.

\textsuperscript{2} ENEA C. R. Frascati, Dipartimento FSN, via E. Fermi 45, 00044 Frascati (Roma), Italy.

\textsuperscript{3} LIBPhys-UC, Department of Physics, University of Coimbra, P-3004 516 Coimbra, Portugal

Acknowledgements
The work leading to this publication has been funded partially by Fusion for Energy under the Contract F4E-FPA-327. IST activities also received financial support from “Fundação para a Ciência e Tecnologia” through project UID/FIS/50010/2013. This publication reflects the views only of the author, and Fusion for Energy cannot be held responsible for any use which may be made of the information contained therein.
**Motivation**

ITER Radial Neutron Camera (RNC) diagnostic main goal: measure in Real-Time (RT) the plasma neutron emissivity profile at high peak count rates for a time duration up to 500 s.

Unprecedented high performance conditions expected → set of activities selected, focused on the development of high priority prototypes, capable to deliver answers to critical issues before the final RNC design.

FPGA code for the front-end electronics prototype aims to acquire, process and store in RT the neutron and gamma pulses from the detectors located in collimated LOS viewing a plasma poloidal section.

*IPFN task*: design, development and testing of the common data-path and dedicated FPGA-side algorithms for RNC front end electronics prototype.
RNC Front End Electronics Prototype

**Host PC:**
Intel(R) Core(TM) i7-5930K CPU @ 3.50GHz;
Scientific Linux 7;
kernel: 3.10-rt

**Digitizer:**
Xilinx evaluation board (**KC705**);
FPGA Mezzanine Card (**FMC-AD2-1600**) with 2 digitizer channels of 12-bit resolution @ 1600 MHz.
RNC FPGA CODE

Xilinx Vivado tool
2015.4/ 2017.4
HDL implementation
(Verilog)
Data Processing – Filter / Bypass

Digital Trapezoidal based Shaper (DTS) -> improvements in SNR; baseline restoring capabilities…
Data Processing – Event detection

*Digital Pulse Processing in Nuclear Physics, CAEN, WP2081, Rev. 3, 2011*
Data Processing – Events Storage (1/2)

- Synthetic input signal with several superimposed pulses (128 samples; PTRG 16 samples; THR. Level)

- Event data

- Event = n x PWIDTH

- TS

- LSB

- MSB

- EVENT

- cnt – decreasing counter

- yes

- no

- cnt = PWIDTH

- cnt = PWIDTH - 4

- cnt = 1
RT Processing – PSD (1/2)

Implementation based on DTS\(^1\):

- The FPGA n/γ PSD code receives data from filter module (DTS filtered data);
- PSD determined by the relation factor between the maximum (peak) of filtered pulses and its integration value – charge integration (CI)

\[ Q \text{ - word 1} \ [63:0] \]

<table>
<thead>
<tr>
<th>63</th>
<th>48</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>reserved</td>
<td>TimeStamp</td>
<td></td>
</tr>
</tbody>
</table>

\[ Q \text{ - word 2} \ [63:0] \]

<table>
<thead>
<tr>
<th>63</th>
<th>56</th>
<th>24</th>
<th>31</th>
<th>21</th>
<th>18</th>
<th>15</th>
<th>12</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>rsv</td>
<td>Cl</td>
<td>rsv</td>
<td>PU</td>
<td>N/γ/L</td>
<td>rsv</td>
<td>Peak</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Other PSD methods might be feasible to implement\(^2,3\)

\(^1\) R.C. Pereira, Neutron/Gamma discrimination code based on trapezoidal filter, Fusion Engineering and Design (submitted)


RT Processing – PSD (2/2)

**Processed data from PSD @ FPGA**

**Setup:**
- **Synthetic data** based on theoretical particle shape of two combined CAEN channels
- **Ch. 1 – gamma; Ch. 2 - neutron**

Poisson distribution (~ 10 \% of pileup) applied to in each channel

- **PU detected @ FPGA**
- **n/γ PSD from FPGA**
- **Low sep. slope**
- **High sep. slope**
Packet with neutron (N) & gamma (γ) Pulse Height Spectra (PHS) + Counts, to be sent periodically to host:

<table>
<thead>
<tr>
<th>N_bin_1</th>
<th>N_bin_0</th>
<th>γ_bin_1</th>
<th>γ_bin_0</th>
<th>γ_bin_3</th>
<th>γ_bin_2</th>
</tr>
</thead>
<tbody>
<tr>
<td>…</td>
<td>…</td>
<td>…</td>
<td>…</td>
<td>…</td>
<td>…</td>
</tr>
<tr>
<td>N_bin_(n-1)</td>
<td>n_bin_(n-2)</td>
<td>γ_bin_(n-1)</td>
<td>γ_bin_(n-2)</td>
<td>γ_total</td>
<td>n_total</td>
</tr>
</tbody>
</table>

- LED
- Single
- Pileup
- Total

- n tot wind DT
- n tot wind DD
- γ tot wind DT
- γ tot wind DD

bins with N counts

bins with γ counts

\[ n – total \text{ spectra bins}^{**} \]

* no-calibrated spectra

**power of 2 (512, 1024, 2048 …)
RT Processing – PHS (2/2)

Setup:
- 100 ms acquisition of two CAEN channels
- 500kev/s each channel

FPGA neutron/gamma PHS

Ch1: gamma shaped pulses

Ch2: neutron shaped pulses
- Negotiates **data-paths between RX, TX and other FPGA blocks**;
- DMAs management and completion to requests **state machine**
- **3 DMAs** available - two DMAs for data transfer; 1 DMA for status reg.
System Control

- **Standard Hardware API (SHAPI)** – Guideline for designing hardware access APIs for xTCA systems, from PICMG® xTCA for Physics

  - Synchronization tasks between the board and the host done through **shared configuration registers**, located in the **host shared memory PCIe configuration space (BAR)**

![Diagram showing System Control](image)

- **Device Registers**
  - **Device reg.** (HW ID, dev. FW ID, version, capabilities)
  - **Fixed registers** (e.g. Vendor ID, Device ID …) settled in `shapi_include.v` (Global Include)

- **Modules Registers**
  - Standard module 1 reg. + FMC dedicated reg. (DMA addresses; acq. and conf. parameters; status …)
  - Not used
Conclusions

✓ The Front End Electronics **FPGA code** designed and successfully implemented in the RNC prototype.

FPGA code **includes**:

- **Signal conditioning**;
- **RT processing** – filtering, event and pileup detection, event storage, PSD and PHA;
- **Data streaming** - two DMAs for real-time data streaming (event-based / PSD or PHS processed data); one DMA for status.

✓ Code tested with **synthetic data** in laboratory:

- maximum throughput: **1600 MB/s** (maximum allowed for 2 channels @ 400 MHz – continuous acquisition);

- **PSD relation factors** from FPGA able to **distinguish neutron /gamma** → further improvements for PU detection identified (overcome the DTS smoothing effect);

- **PHS successfully tested** → might be difficult to operate in experiments with high fluctuation of the separation slopes.