



Heterogeneous Computing System Platform For High-Performance Pattern Recognition Applications

M Ali Mirzaei, [Vincent Voisin](#),

Alberto Annovi, Guillaume Baulieu, Matteo Beretta,
Giovanni Calderini, Saverio Citraro, Francesco Crescioli, Geoffrey Galbit, Valentino
Liberali, Seyed Ruhollah Shojaii, Alberto Stabile, William Tromeur, and Sebastien Viret

October 3, 2017

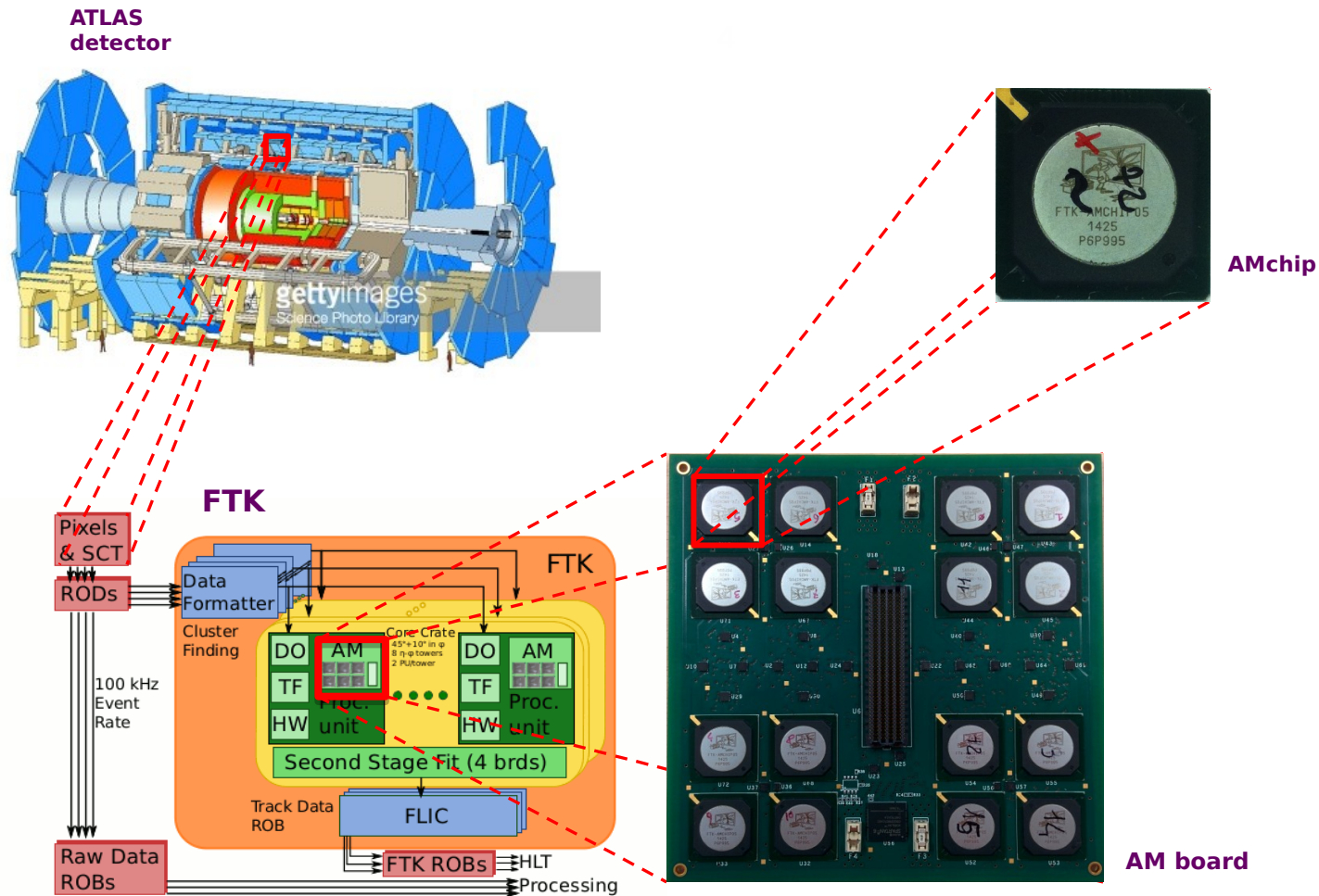
- Various Heterogeneous System exists :
 - CPU / GPGPU
 - CPU / FPGA
 - CPU / ASIC
 - CPU / FPGA /ASIC

ZYNQ + AMchip Board

- ZYNQ+AMchip = ARM CPU + FPGA + AMchip:
 - AMchip: AMchip can perform Pattern-Matching very quickly with parallel computing
 - FPGA: Main logic and interfacing blocks can be host by FPGA
 - ARM CPU: ARM is a conventional CPU which executes embedded linux

AMchip in FTK and ATLAS Experiment

AMchip in FTK and ATLAS detector



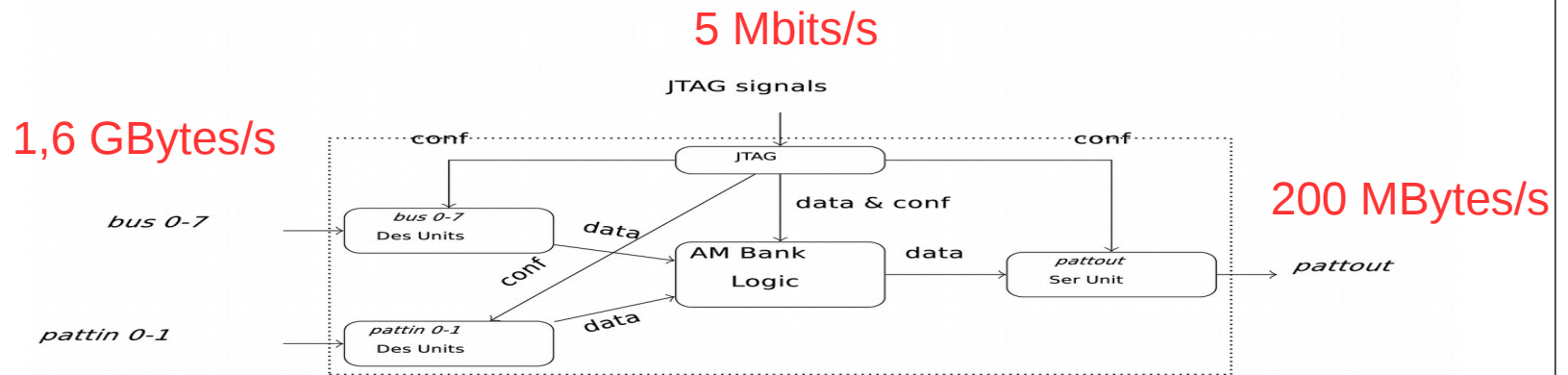
FastTrack @ ATLAS : <https://cds.cern.ch/record/1552953/>

- AMchip (Associative Memory Chip):
 - an ASIC system with fully parallel architecture designed to perform thousands of pattern matching in few clock cycles
 - An AM06chip has 128k patterns, each pattern is 16* 8 bits, it makes comparisons at 100MHz, so the chip is able to compare the input and the pattern bank at a rate of **190 TBytes/s**
 - We thought of using this chip in other pattern recognition applications to prove the efficiency of the chip in other science/engineering fields
 - AMchip bandwidth is about **1,6GBytes/s**

AMchip

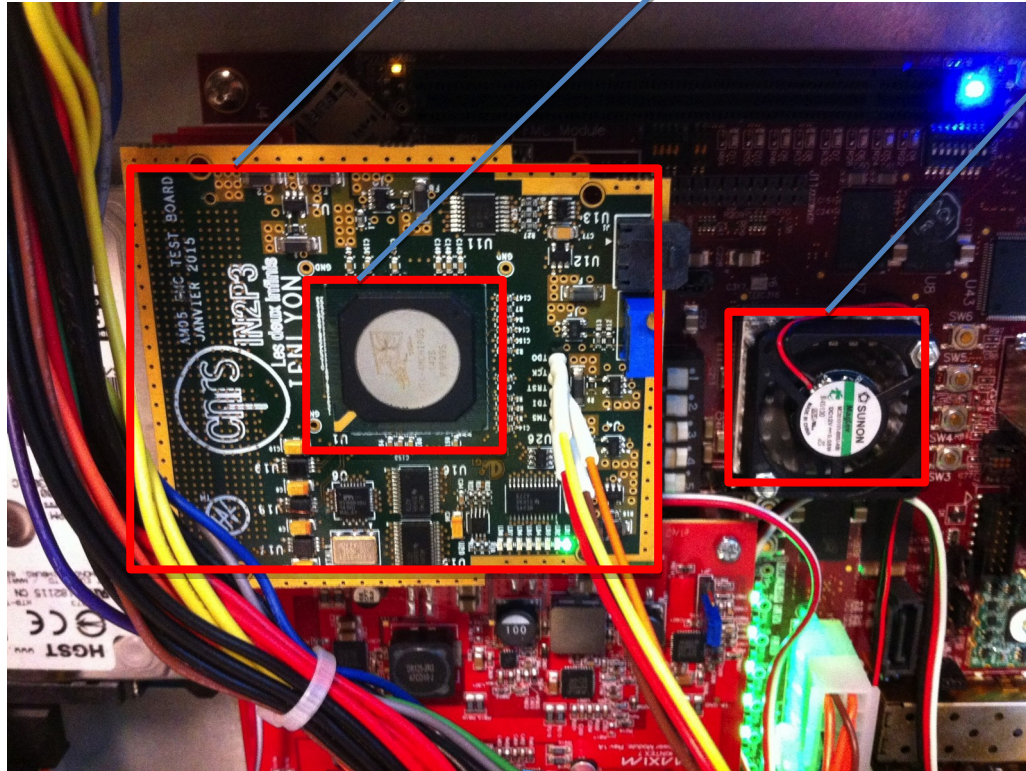
- Currently we have two version of AMchip available:
 - AMchip05 with 2k patterns
 - AMchip06 with 128k patterns
- In the future (2019-2020) we will have an AMchip with ~512k patterns
- AMchip can be chained together to form larger banks (such in FTK), but our actual hardware use a single -chip mezzanine

AMchip



Hardware of The Development Platform

AMchip board **AMchip** **Zynq FPGA**



Zynq FPGA board

AMchip configuration by JTAG

- To Configure the AMchip we need to communicate through a JTAG port
 - The JTAG communication is ensure via a simple bit banging on JTAG signals (TDI, TMS, TCK, TDO, TRST)
 - Inspired by a library in Kovan-JTAG project
<https://github.com/xobs/kovan-jtag>
- For AMchip05, it takes 2 minutes by JTAG to load the full bank (2k patterns) which is acceptable, but doesn't scale for AM06 (128k patterns)
- For AMchip06, the solution is to perform JTAG cycles in hardware (FPGA) and control at higher level in the software:
https://www.xilinx.com/support/documentation/application_notes/xapp503.pdf

- Small Scale Particule Physics Experiments such a test beam telescope
 - The AMchip pattern matching can provide real time tracking
 - Tracking could be further refined with a fit in FPGA
 - The associative memory for the self-triggered SLIM5 silicon telescope

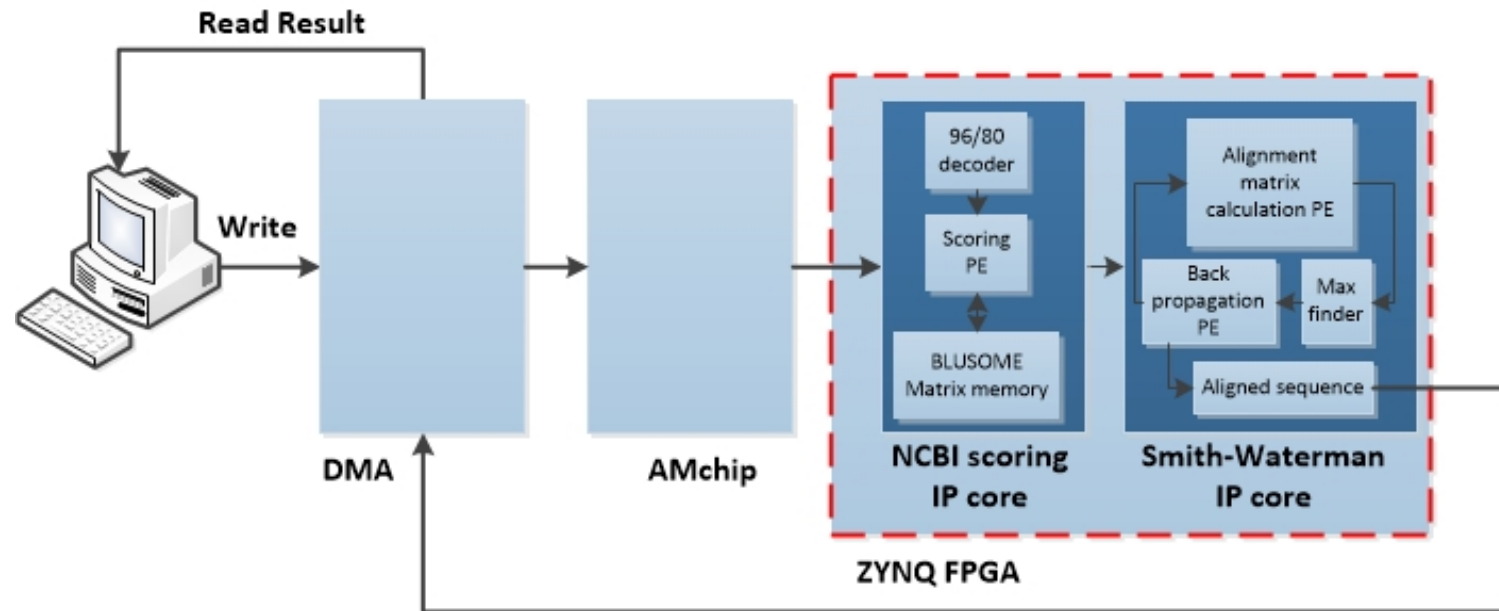
[\[1\] - G. Batignani and al., "The associative memory for the self-triggered SLIM5 silicon telescope", Nuclear Science Symposium Conference Record, 2008 .NSS '08. IEEE](#)

- Image Processing:

[\[2\] - M. M. Del Viva, G. Punzi, and D. Benedetti, "Information and perception of meaningful patterns," PLOS ONE, vol. 8, 07 2013.](#)

[\[3\] - "A Hardware Implementation of a Brain Inspired Filter for Image Processing" \(to be published in IEEE TNS\)](#)

System Architecture For Genomics Sequence Analysis



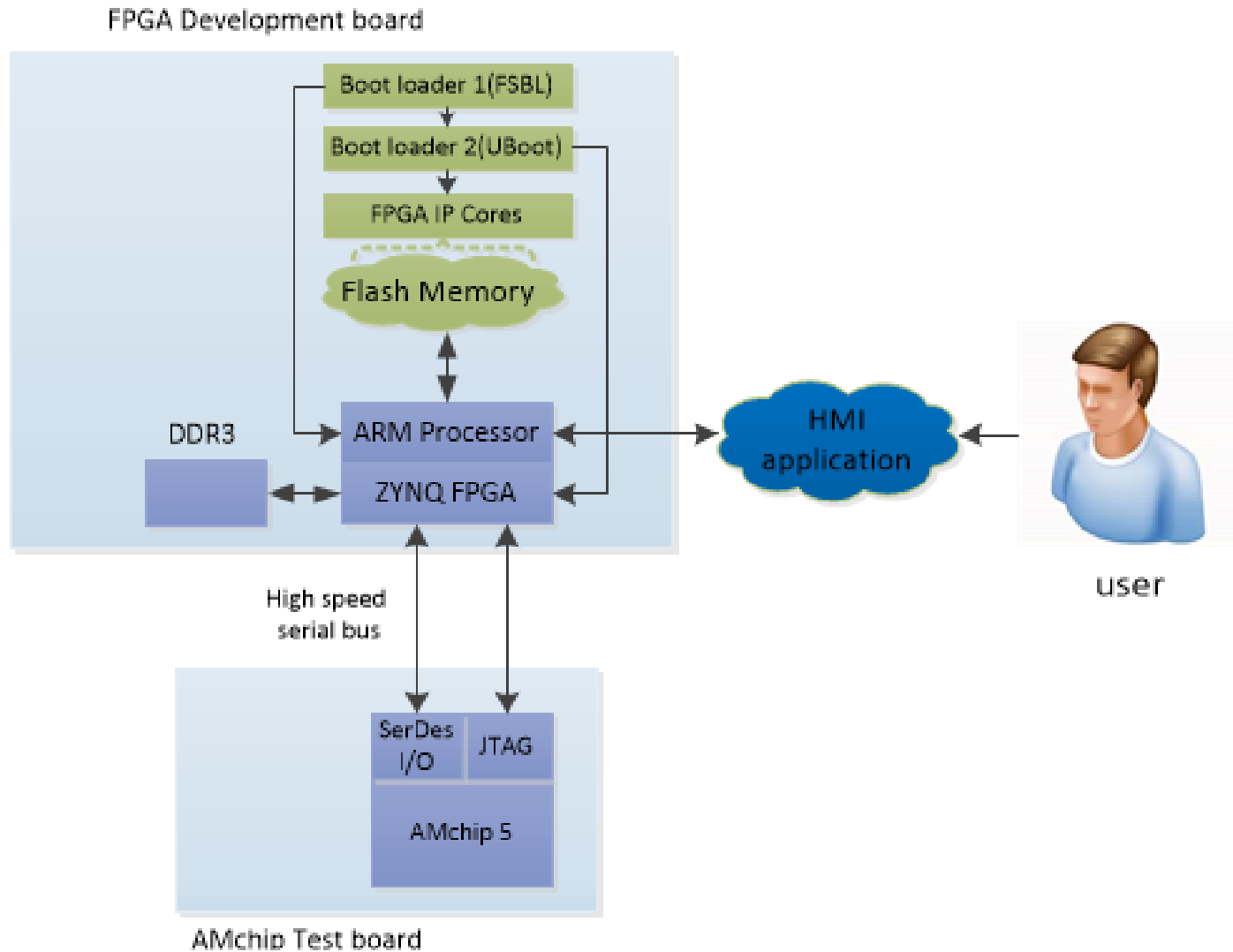
[4] M. Ali Mirzaei, Francesco Crescioli and al.,
 “A Novel Associative Memory Based Architecture for Sequence Alignment”
 HiCOMB 2016

- Bootloaders
 - First State BootLoader + U-Boot
 - <https://github.com/Xilinx/u-boot-xlnx>

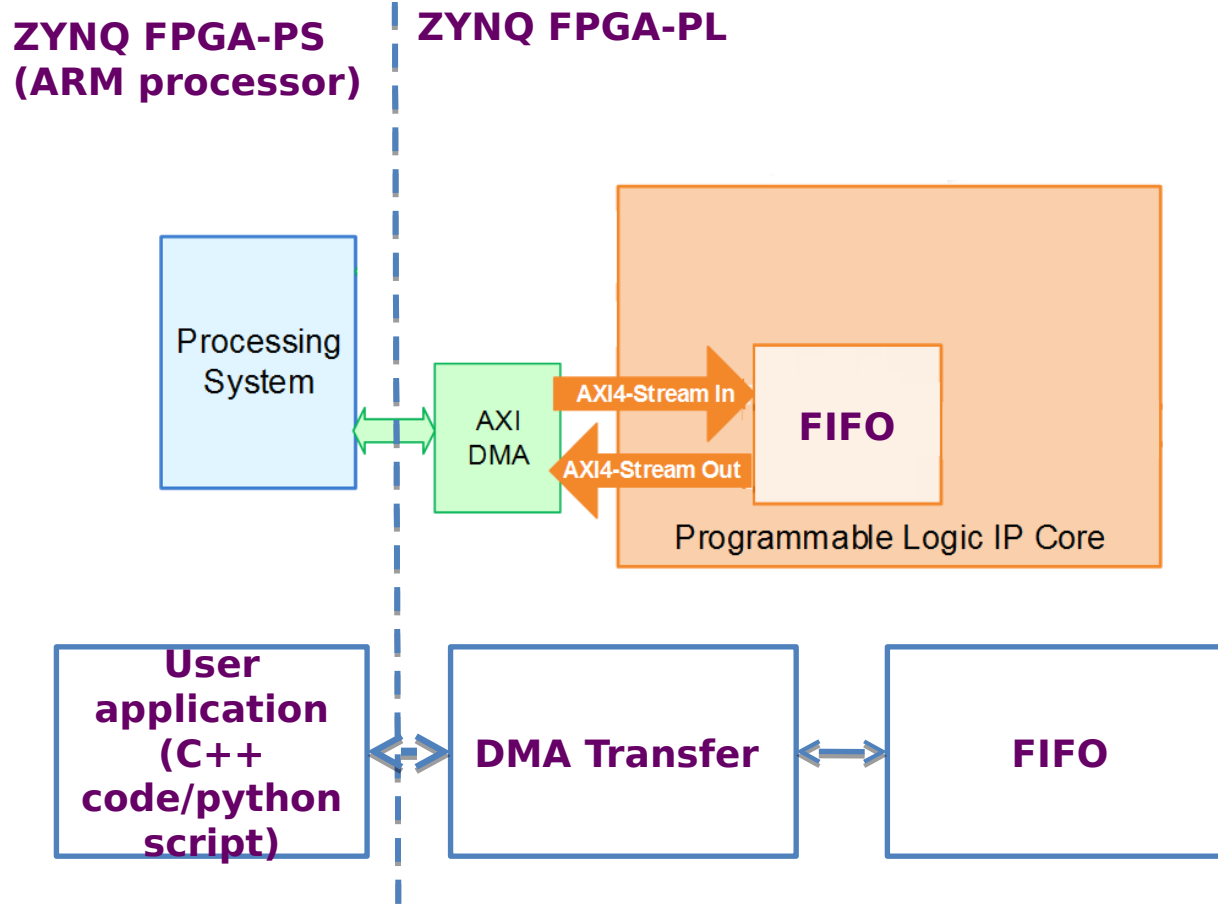
- The Linux Kernel
 - Very recent Kernel: Linux v4.4.x
 - <https://github.com/Xilinx/linux-xlnx>

- The Root File System
 - Based on Ubuntu Core 16.04.02
 - <http://cdimage.ubuntu.com/ubuntu-base/releases/16.04.2/release/ubuntu-base-16.04-core-armhf.tar.gz>

Hardware of The Development Platform



Full-chain high-speed link for data communication

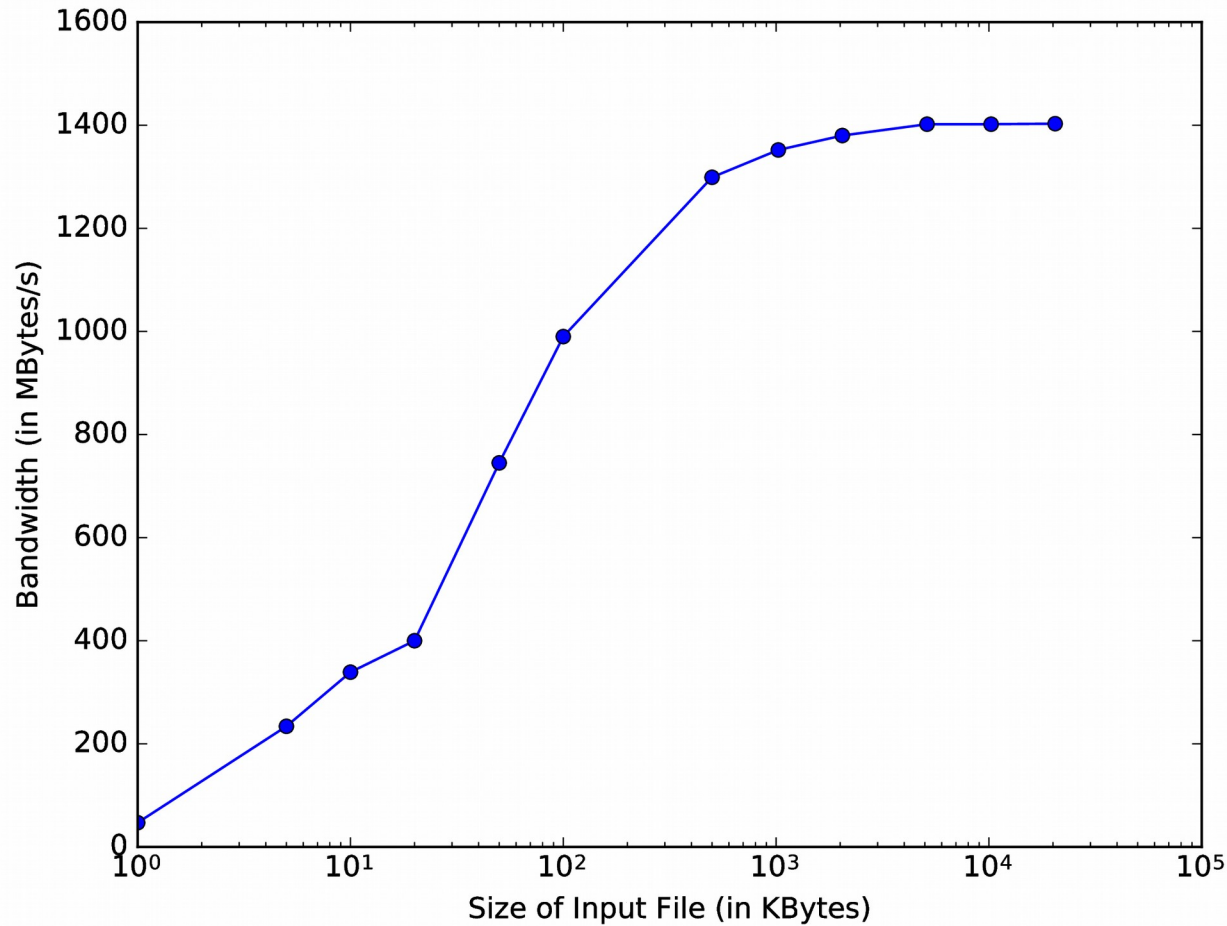


The libgannet solution

- A complete framework to create DMA interface
 - Programmable Logic
 - Softwares (linux driver + user-space library to handle DMA)
- <https://gitlab.com/SmartAcoustics/libgannet>
- Python wrappers were created to make the development easier

DMA Bandwidth

Max = 1.4 GBytes/s



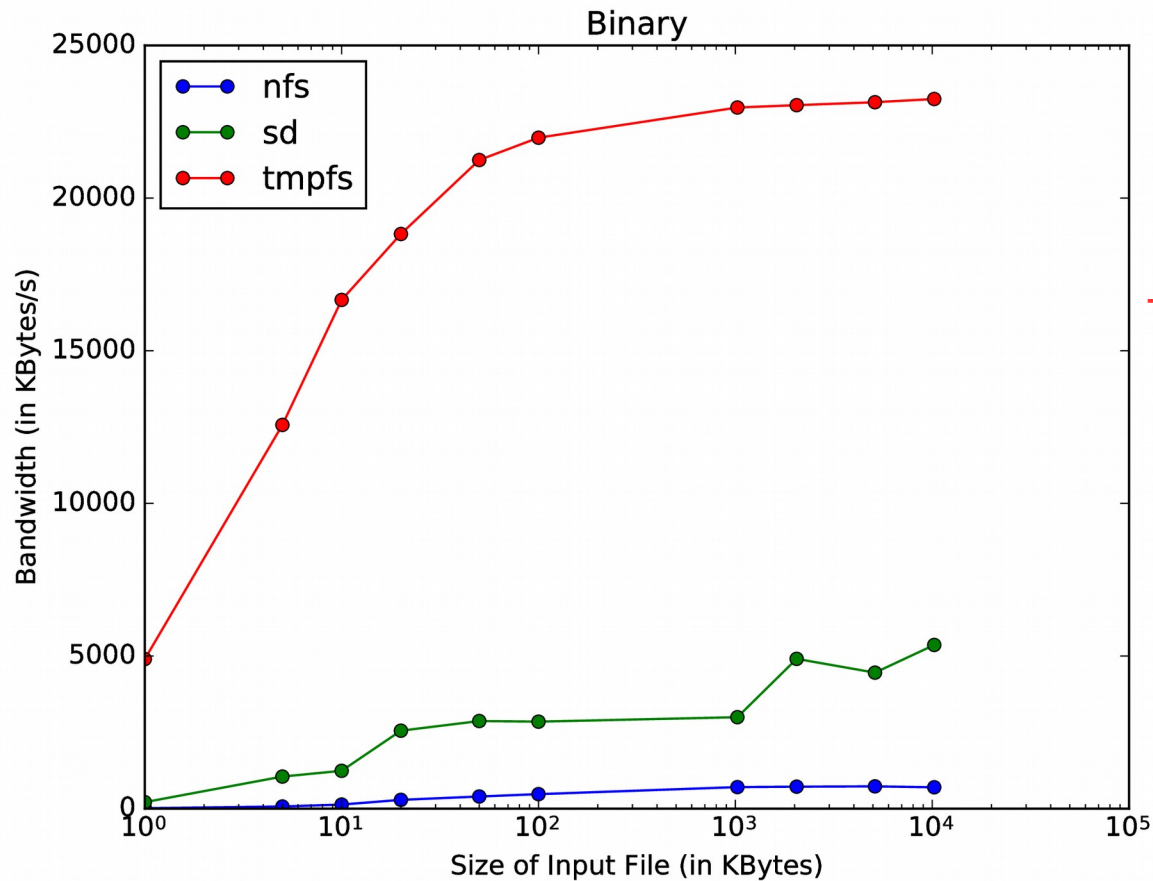
“Full-Chain” bandwidth

- Bandwidth evaluation:
 - “Full-Chain”:
 - Data are read from a file and put in memory
 - Data in memory are send to PL (here a FIFO) through DMA
 - Data are read back to Memory from FIFO
 - Data in memory are saved in another file

"Full-Chain" bandwidth

- Two kinds of file format
 - Data File is in a binary format
 - Data File is in a JSON format
- Three types of file systems
 - NFS (slow)
 - SDCard
 - tmpfs (useful to discover bottlenecks)

With files in binary format

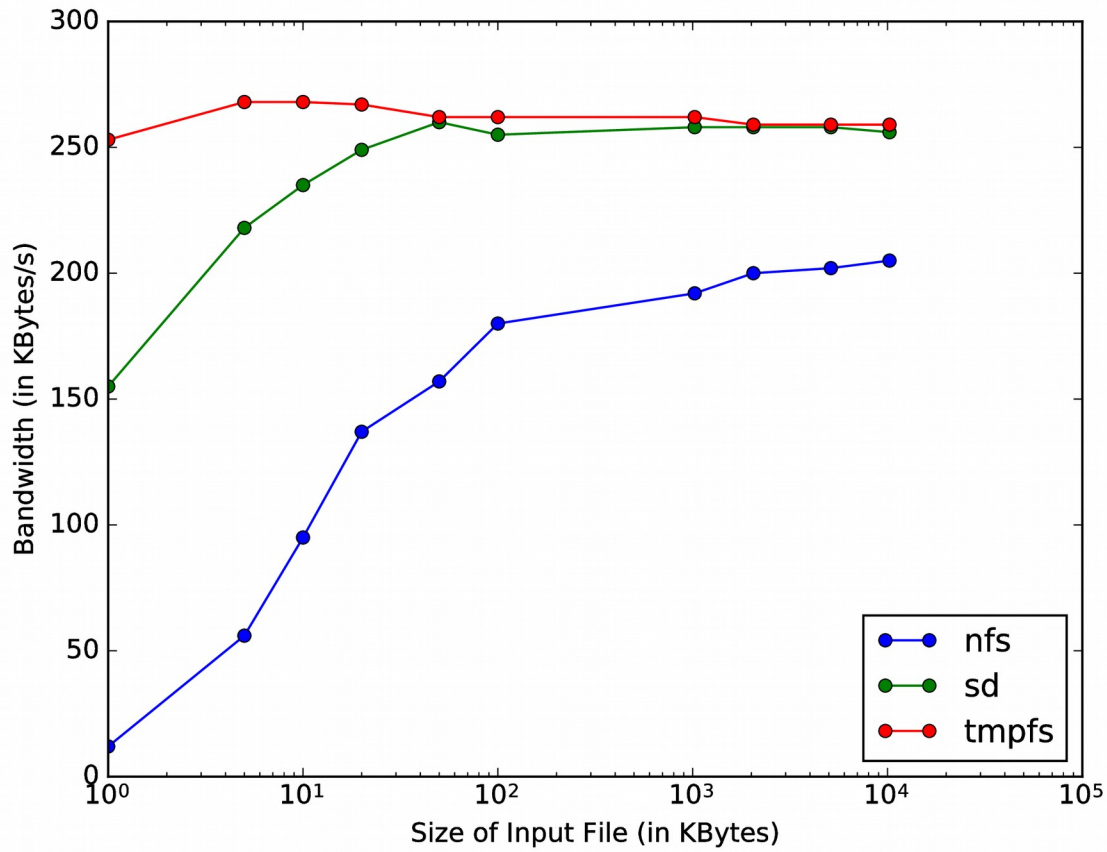


NFS = 730 KBytes/s

SD = 5.36 MBytes/s

TMPFS = 23.25 MBytes/s

With files in JSON format



NFS = 205 KBytes/s

SD = 256 KBytes/s

TMPFS = 259 KBytes/s

- The DMA bandwidth is **1,4GBytes / s** !
- With Binary File Format as Input/Output, the “full bandwidth” is limited to **23MBytes/s**
- With JSON File Format as Input/Output; the “full bandwidth” is limited to **260KBytes/s**
 - JSON parsing is a CPU intensive task
 - But JSON is a flexible format, suitable in a variety of applications
 - So we’ll investigate other methods, like store it in a database (such MongoDB)
- Full chain bandwidth is still low for a read, calculate and write operation when storing data locally

- A Heterogeneous Computing System Platform for Pattern Recognition has been presented
- A high-speed data communication infrastructure between Linux UserSpace, FPGA and AMchip has been presented
- Proposed approach reach up to 1,4GBytes/s communication speed between UserSpace Memory and FPGA
- “Full chain bandwidth” is still low, we are working to improve it.

- All developments are published through the following links
 - Embedded Linux Platform:
<https://gitlab.in2p3.fr/zynq-am/linux>
 - Softwares Development:
<https://gitlab.in2p3.fr/zynq-am/software>
 - Firmwares:
<https://gitlab.in2p3.fr/zynq-am/firmwares>
- If someone is interested in evaluating the platform :
 - The Zynq Development kit is off-the-shell
 - Software and Firmware are available
 - Contact us for the AMchip board, we could provide it after discussion with the different partners

- Thanks You !

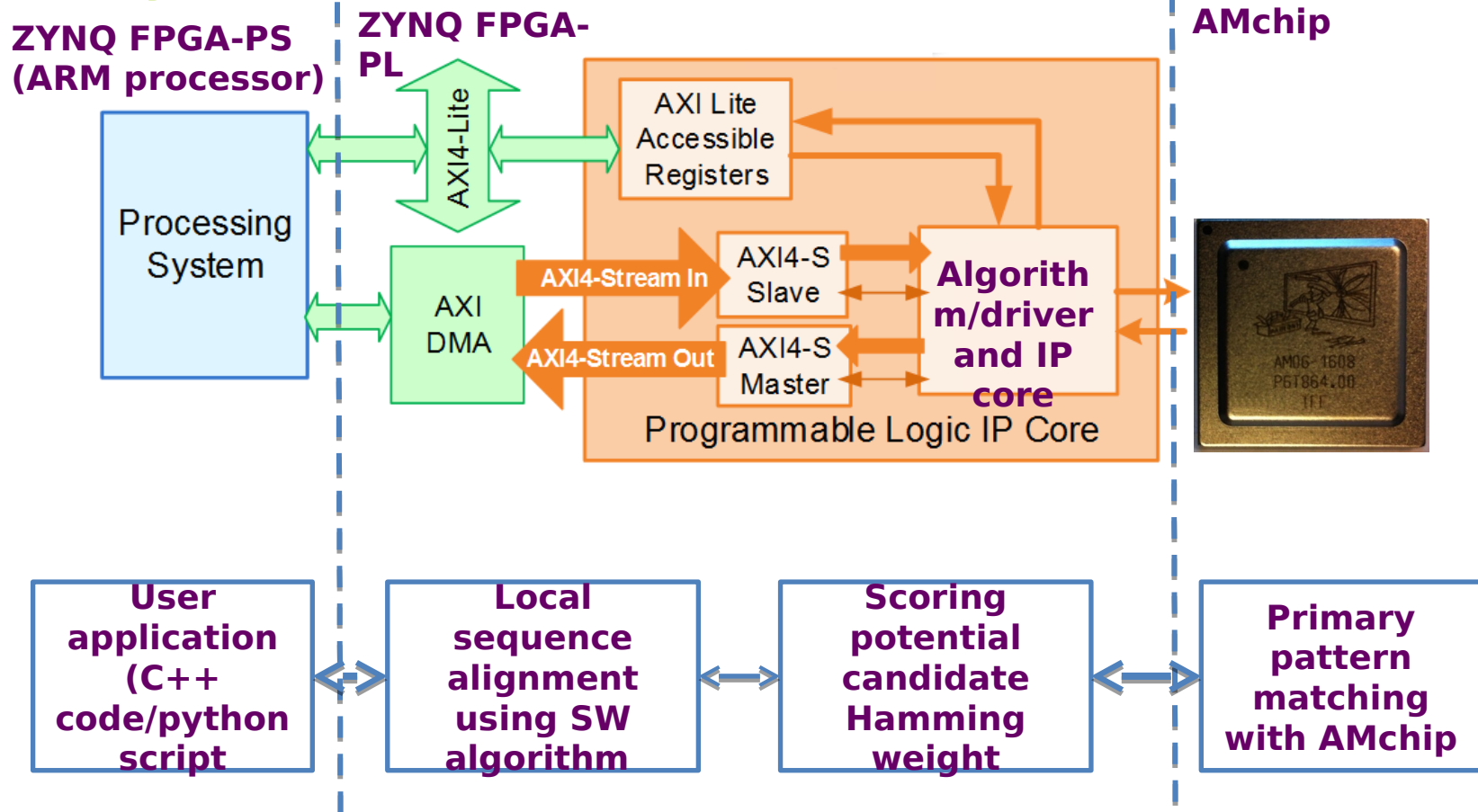
- Project received founding from the French ANR project FastTrack:

[ANR-13-BS05-0011 FastTrack](#)

- The authors would like to thank AMchip design team at LPNHE, INFN and IPNL

- BACKUP SLIDES

System Architecture For Genomics Sequence Analysis



Content:

1. Heterogeneous Computing System
2. Our Heterogeneous Computing
 - ZYNQ + AMchip board
3. AMchip Applications
 - AMchip configuration
4. Examples Applications
 - Small Scale Physics Experiments
 - Computer Vision, Image Processing
 - System Architecture for Genomics Sequences Analysis
5. Embedded Linux System
 - FSBL+Uboot
 - Kernel
 - Root FileSystem
6. DMA Evaluation
 - DMA Principle
 - Schematics for DMA evaluation
 - DMA Software
 - DMA performance
7. “Full-Chain” Bandwidth Evaluation
 - Binary File
 - JSON File
 - Results
8. Conclusions
9. Future works
10. Acknowledgments

Heterogeneous Computing System Platform For high-Performance Pattern Recognition Applications

M. Vincent Voisin
CNRS, Software Engineer
LPNHE,
ATLAS group, AMchip team
Contact:

vvoisin@lpnhe.in2p3.fr

28