Firmware & software for the AMBFTK monitoring & diagnostic: work in Pisa now and at PRIELE

Marco Piendibene - University of Pisa & INFN
Simone Donati - University of Pisa & INFN
Kostas Kordas - Aristotle University of Thessaloniki
Dimitris Dimas - Prisma Electronics (Athens)

IAPP - FTK Executive board – June 27, 2013

Outline

- Firmware for error detection (status and plan)
- Software for error detection (status and plan)
- Software for AM system simulation and tests

Firmware for error detection

Errors to be monitored into the INPUT Chips

- **Lost Synchronism** (event tags in different streams do not match)

- **Parity or CRC error** – for each link (8 pixel & 4 SCT) Parity (PA).

- **FIFO Overflow** – each FIFO full flag should produce error if set.

- **Invalid Input data** (for example invalid HIT from ROD)

- **Truncated output** (for example too many hits in input) -

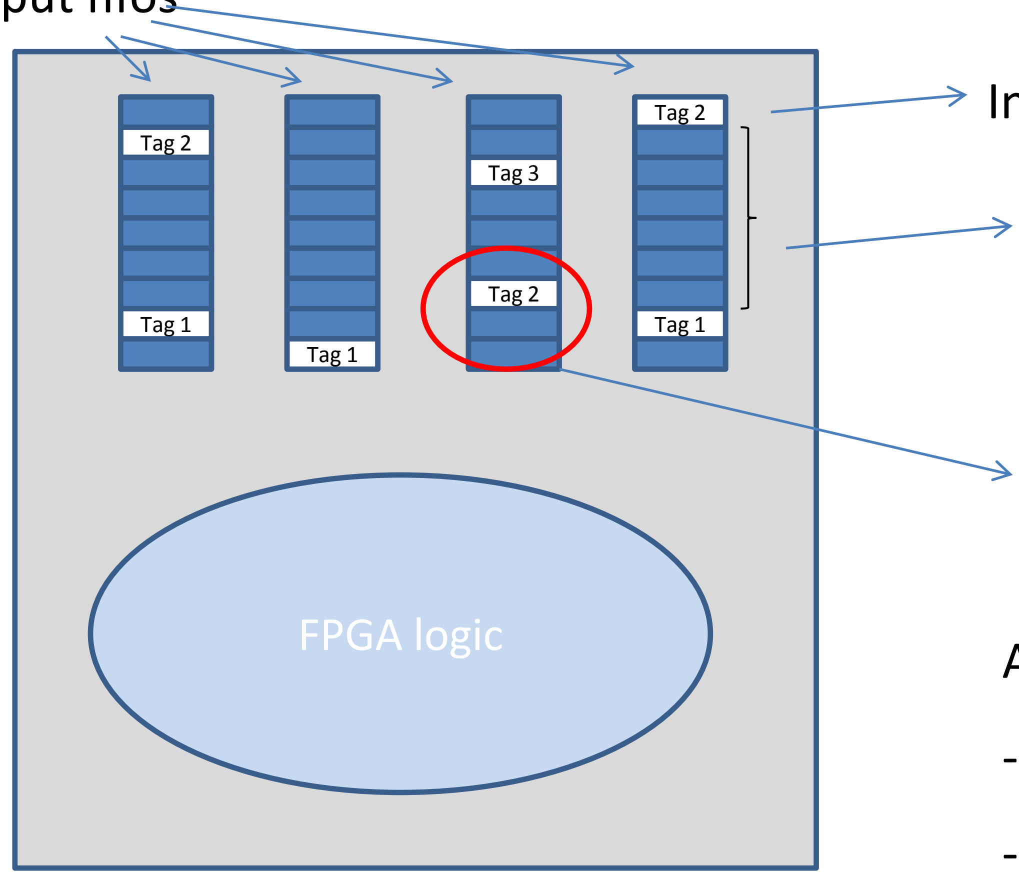
VME registers

- Error register
- For each FIFO: flag register (empty, AFull, Full)
- For each FSM: state of the machine
- Output Status (Hold flags)
- For each Spy Buffer: Spy Buffer register: Pointer, OVFL flag, status(freeze/spy).
- firmware ID.
- Input FiFos: R/W
- Input Spy Buffers memories
- Timing measurements: each chip has a counter. When Init_event is received it is started. When all the input EE event words are received it is stopped. All incoming words are written in the spy buffer with the counter content.

Lost sync error detection (example of error)

Lost sync error: we have to check that all the EE word have the same event tag

Input fifos



Input End event word (EE)

Event data

Lost sync !

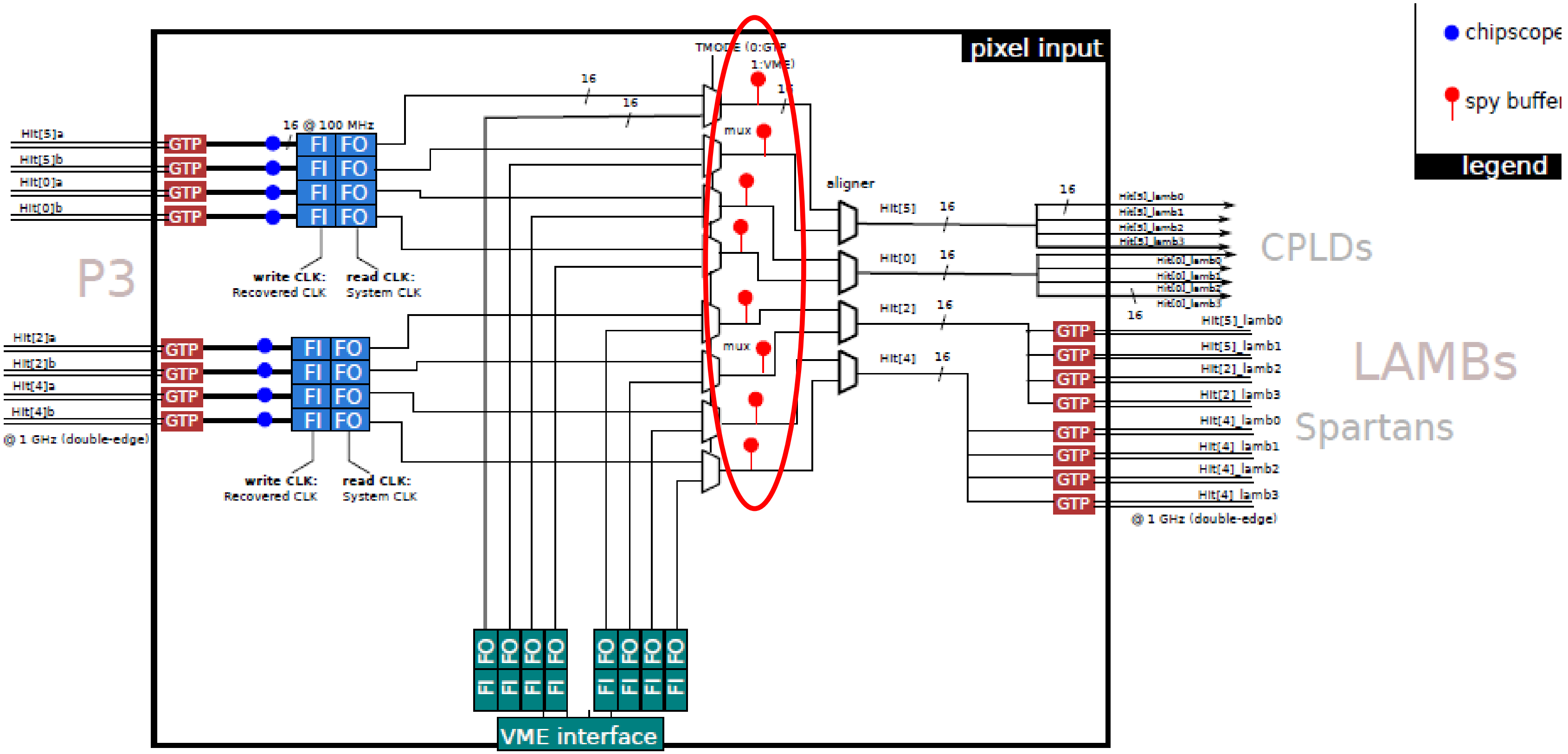
About errors:

-Critical errors (lost sync)

-Warnings (parity errors)

We have to propagate error and warning in the output EE word

Spybuffers: essential tools for debug & monitoring



Spybuffer: makes a copy of a portion of data (without interfere on the running dataflow) for debugging and monitor purposes

- readable by VME

Firmware for debugging: status

Status: Dimitris (@UNIPI) has written the code for LOST SYNC ERROR detection.

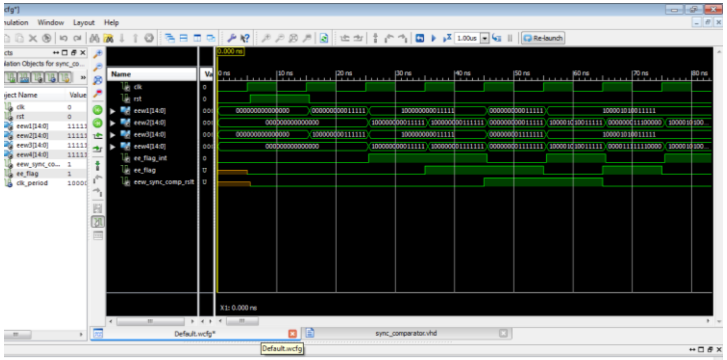
Test bench written and OK.

next steps:

- try to implement the LOST SYNC code in the AMBFTK actual firmware to test the functionality (July 2013)
- Write the code for other kind of errors and spybuffers (July 2013)

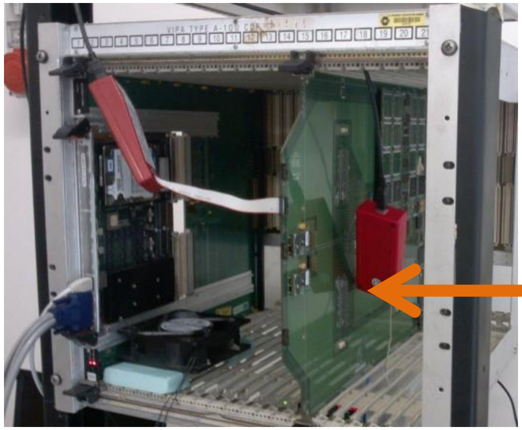
Coffe time! ☺

Simulation time

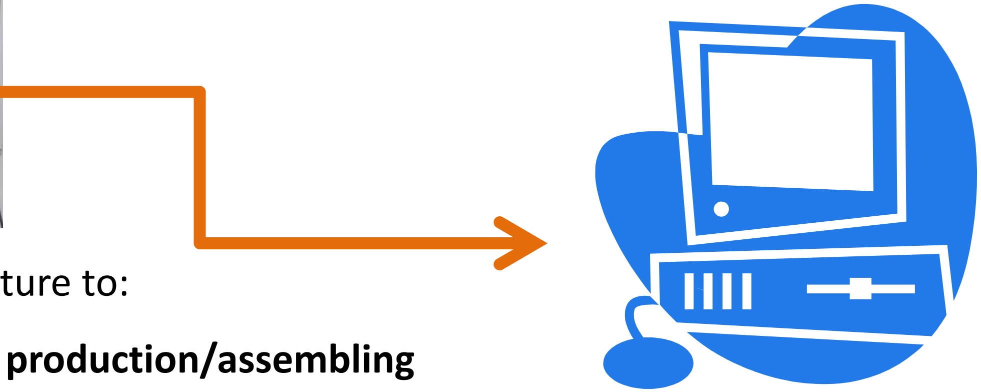


Software for error detection

Software



Actual AMBFTK board in the VME crate



We need a software infrastructure to:

-**Debug the single board after production/assembling**

-Read error information from the boards

-Debug the dataflow (read/check the spybuffers)

-Control and configure the boards

-**Monitoring the running system**

-Configure the system

-Verify hardware vs simulation

-Plot interesting information (timing measurement, errors..)

Software

Status: Simone and Kostas (@CAEN) are studying the standalone debug programs to test the single board

They are studying also the infrastructure of CDF experiment (Spymon, SVTmon) to monitoring the system and dataflow

next steps:

- Simone and Marco (@PRIELE – August 2013) will start to interface firmware and software in order to start the debug the first prototypes of AMBSLP (see Saverio presentation) in Prisma during summer
- Simone and Kostas will integrate in the ATLAS environment the tools to monitoring the system

Software for AM system simulation and tests

Software for standalone AM system simulation and tests

Status: we have already a software for standalone simulation of the associative memory, but needs to be upgraded ad improved to be compatible with the new system (AMBSLP, LAMBSLP, AMCHIP05)

- For example, 16 output spybuffers instead of 1 to be dumped and compared with the AM simulation

next steps:

- Simone will help **Daniel** (university of Modena & INFN Perugia) to accomplish this task (next 2 months)

Conclusion

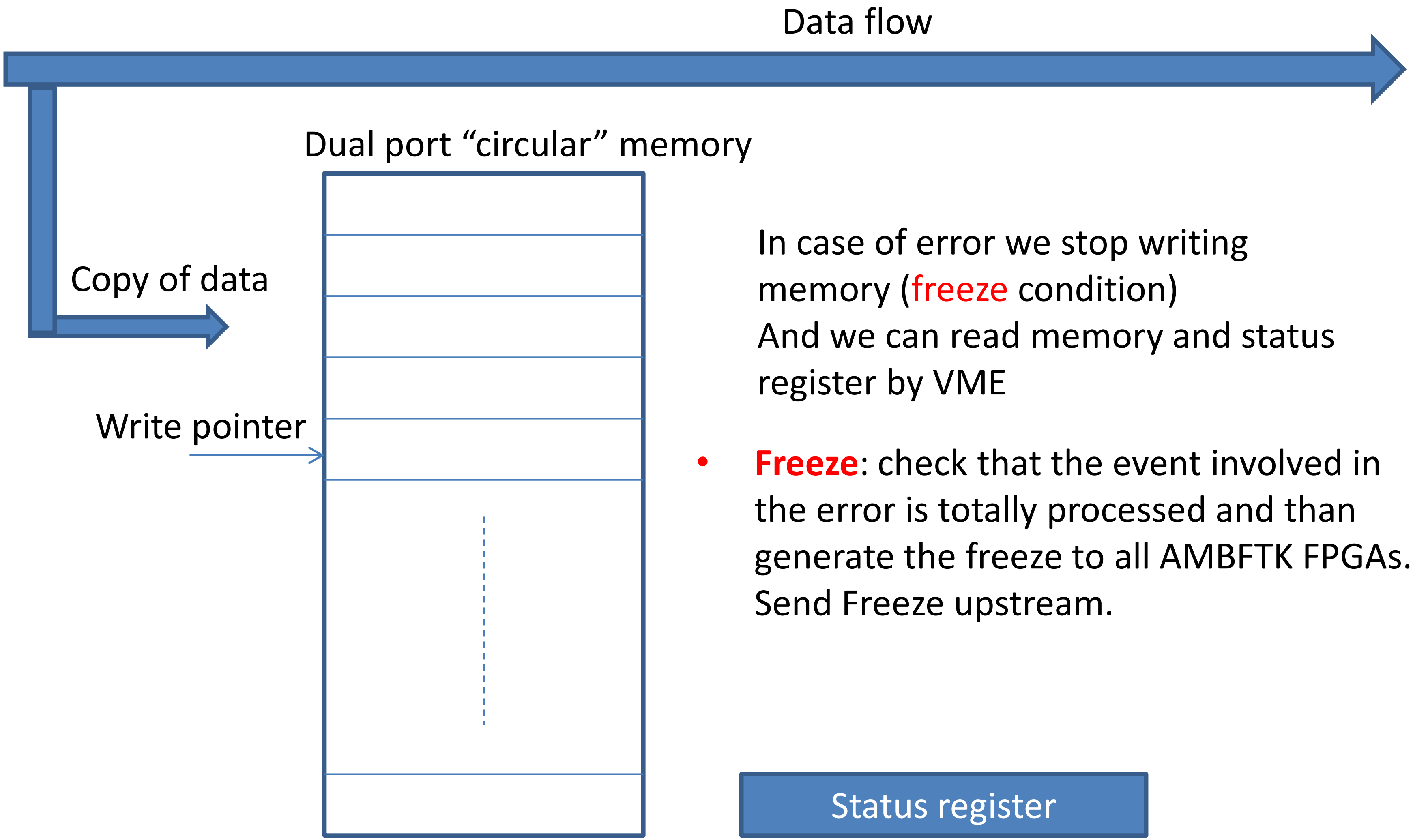
We are in good shape but a lot of things to do!

PRIELE – AUTH – UNIPI meeting

(Χαλαρά!)

Backup

Spybuffers



In case of error we stop writing memory (freeze condition)
And we can read memory and status register by VME

- **Freeze**: check that the event involved in the error is totally processed and than generate the freeze to all AMBFTK FPGAs. Send Freeze upstream.