# Embedding Online Test & Monitoring Features in Real Time Hardware Systems

**I. Mandjavidzé**

*IRFU, CEA Saclay, 91191, Gif-sur-Yvette, France*

**T. Romanteau**

*LLR, École polytechnique, CNRS/IN2P3, 91128, Palaiseau, France*

irfu
cea saclay

LLR

# Context

## TCCs & SRP in the CMS ECAL Readout System



TCC – Trigger Concentrator Card
SRP – Selective Read-out Processor
DCC – Data Concentrator Card

**TCC & SRP Readout Paths**
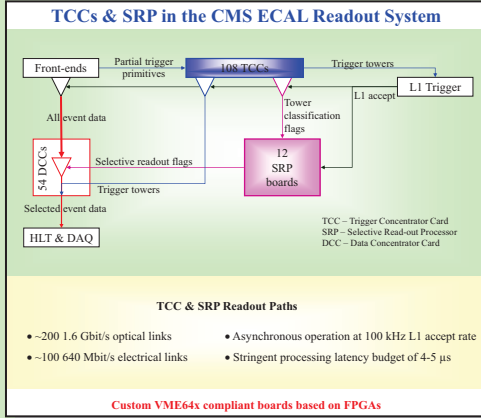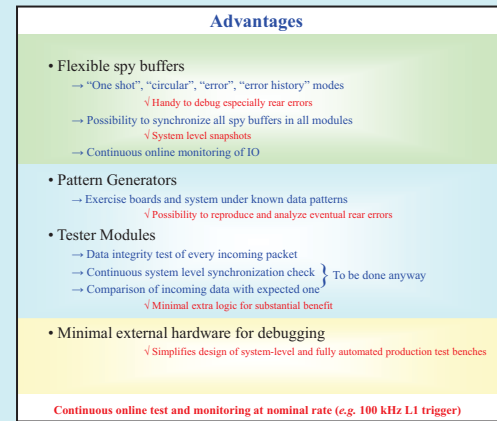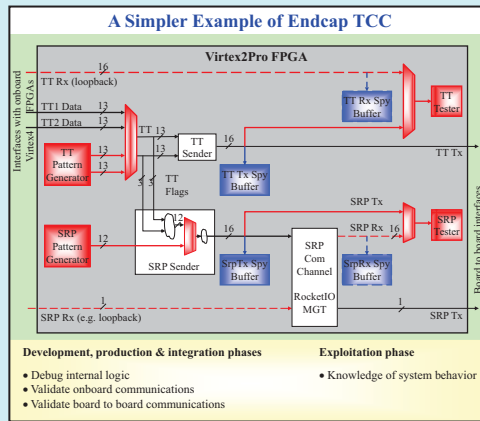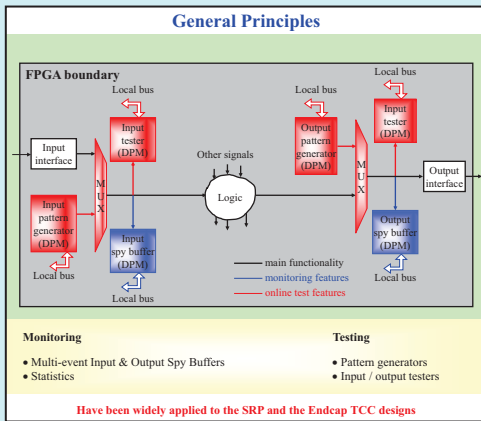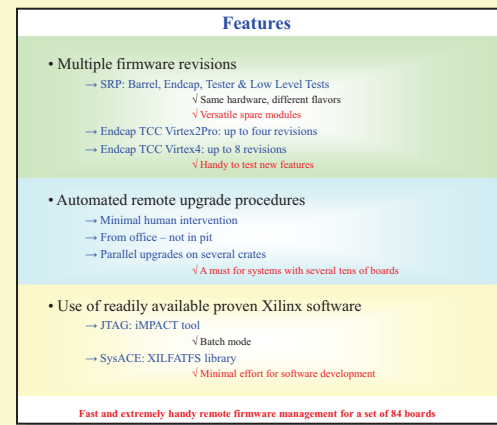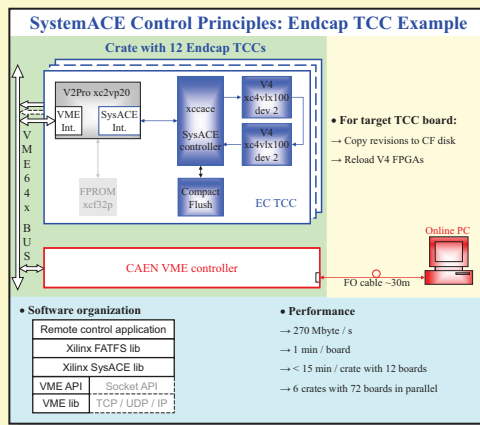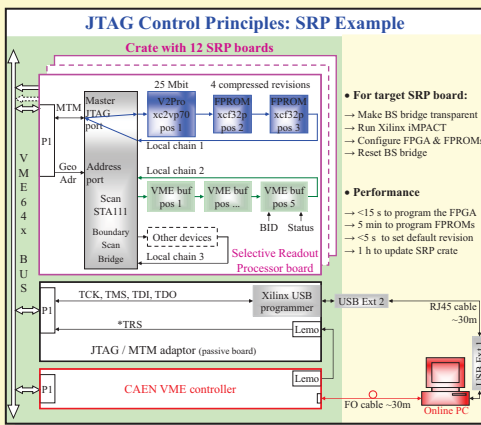
- ~200 1.6 Gbit/s optical links
- ~100 640 Mbit/s electrical links
- Asynchronous operation at 100 kHz L1 accept rate
- Stringent processing latency budget of 4-5 µs

**Custom VME64x compliant boards based on FPGAs**

## Selective Readout Processor



**12 VME64x compliant identical boards**
**240 optical communication channels**
- 36 barrel TCCs
- 72 endcap TCCs } Inputs
- 54 DCCs — Outputs
- 78 intra SRP — Bidirectional

**1700-pin Virtex2Pro FPGA from Xilinx**
- 20 multi-gigabit transceivers
- 2 Embedded PowerPC
- 328 18-kbit dual port memories
- Various hard and soft IPs

**Four 12-channel parallel optic modules**
**Three firmware flavors**
- Barrel, Endcap, Tester

**What are the means to validate a system with the substantial number of IO interfaces ?**

## Common Considerations

**Use loopback capability**
Validate logic and physical layers

**Transform the board to a tester**
Emulate neighbor SRPs:
Generate TTs patterns
Validate expected TTs
Emulate DCC:
Validate expected SR flags
Emulate TCC:
Generate TTs patterns

*e.g. production test bench*
*e.g. system level development test bench*

**When appropriate adopt SoC design: standalone operation capability**



# Principles & Applications

## General Principles



— main functionality
— monitoring features
— online test features

**Monitoring**
- Multi-event Input & Output Spy Buffers
- Statistics

**Testing**
- Pattern generators
- Input / output testers

**Have been widely applied to the SRP and the Endcap TCC designs**

## A Simpler Example of Endcap TCC



**Development, production & integration phases**
- Debug internal logic
- Validate onboard communications
- Validate board to board communications

**Exploitation phase**
- Knowledge of system behavior

## Advantages

- **Flexible spy buffers**
  → "One shot", "circular", "error", "error history" modes
    √ Handy to debug especially rear errors
  → Possibility to synchronize all spy buffers in all modules
    √ System level snapshots
  → Continuous online monitoring of IO
- **Pattern Generators**
  → Exercise boards and system under known data patterns
    √ Possibility to reproduce and analyze eventual rear errors
- **Tester Modules**
  → Data integrity test of every incoming packet
  → Continuous system level synchronization check } To be done anyway
  → Comparison of incoming data with expected one
    √ Minimal extra logic for substantial benefit
- **Minimal external hardware for debugging**
  √ Simplifies design of system-level and fully automated production test benches

**Continuous online test and monitoring at nominal rate (*e.g.* 100 kHz L1 trigger)**

# Remote Firmware Management

## JTAG Control Principles: SRP Example



**For target SRP board:**
→ Make BS bridge transparent
→ Run Xilinx iMPACT
→ Configure FPGA & FPROMs
→ Reset BS bridge

**Performance**
→ <15 s to program the FPGA
→ 5 min to program FPROMs
→ <5 s to set default revision
→ 1 h to update SRP crate

## SystemACE Control Principles: Endcap TCC Example



**For target TCC board:**
→ Copy revisions to CF disk
→ Reload V4 FPGAs

**Software organization**

| Remote control application | |
|---|---|
| Xilinx FATFS lib | |
| Xilinx SysACE lib | |
| VME API | Socket API |
| VME lib | TCP / UDP / IP |

**Performance**
→ 270 Mbyte / s
→ 1 min / board
→ < 15 min / crate with 12 boards
→ 6 crates with 72 boards in parallel

## Features

- **Multiple firmware revisions**
  → SRP: Barrel, Endcap, Tester & Low Level Tests
    √ Same hardware, different flavors
    √ Versatile spare modules
  → Endcap TCC Virtex2Pro: up to four revisions
  → Endcap TCC Virtex4: up to 8 revisions
    √ Handy to test new features
- **Automated remote upgrade procedures**
  → Minimal human intervention
  → From office – not in pit
  → Parallel upgrades on several crates
    √ A must for systems with several tens of boards
- **Use of readily available proven Xilinx software**
  → JTAG: iMPACT tool
    √ Batch mode
  → SysACE: XILFATFS library
    √ Minimal effort for software development

**Fast and extremely handy remote firmware management for a set of 84 boards**

---

*Online test and monitoring logic, that closely couples with functional features, and remote firmware control greatly simplify development, production, commissioning and maintenance of large scale FPGA-based real time systems*