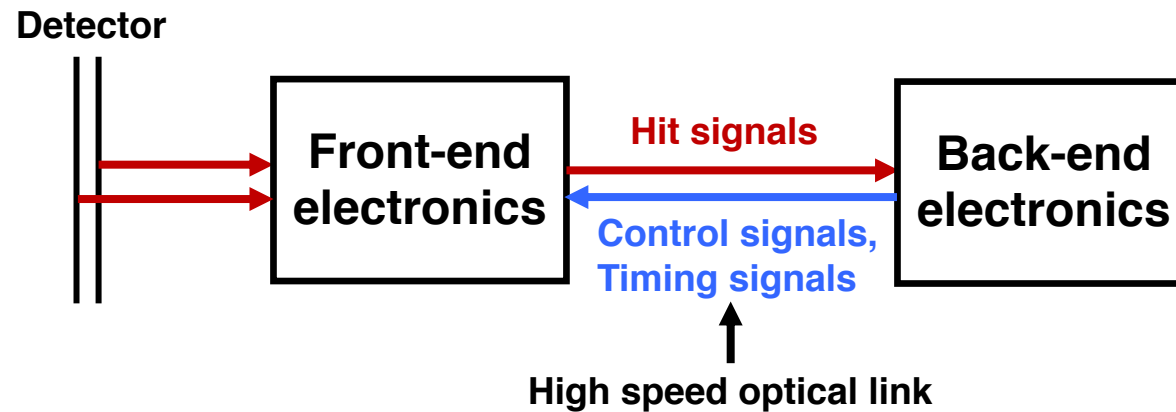

Complete design of maximally-automated self-driven control mechanism for a large scale electronics system and its application to the ATLAS Phase-II TGC system

Takumi Aoki, on behalf of the ATLAS Muon Collaboration
The University of Tokyo

Control of the large scale electronics system

- High-energy physics experiment usually involves a large-scale electronics system (for instance more than 1500 front-end electronics need to be controlled)
- “FPGA + high-speed optical links” design is one of the common choices



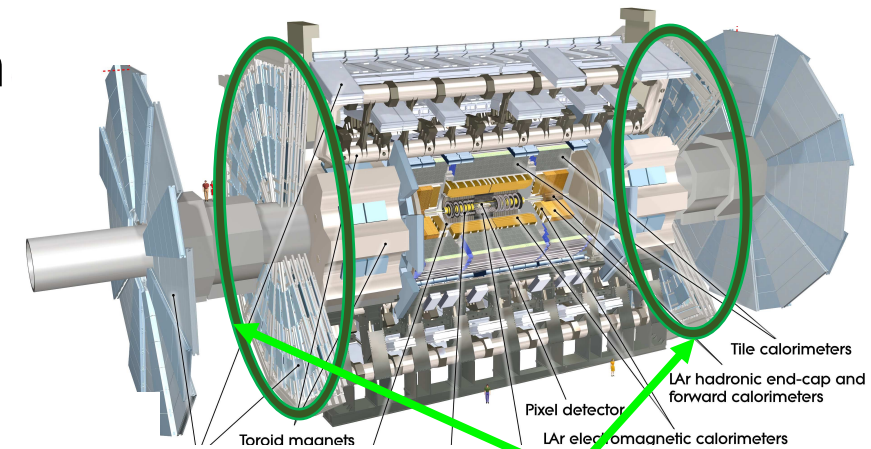
- Establishing a reliable and efficient method to control electronics system is critical
 - Also needs to establish a serial link with optical fiber without phase ambiguity
- > We have designed and implemented a maximally-automated and self-driven scheme for a system that exploits FPGAs, flash memory devices, and high-speed optical links, which can be widely applicable to HEP experiments**

ATLAS experiment in HL-LHC

- The operation of the ATLAS detector at High-Luminosity LHC (HL-LHC) will begin in 2029 for precision measurements of the Standard Model and to search for new physics with high statistics

	Peak luminosity ($\text{cm}^{-2}\text{s}^{-1}$)	First stage trigger rate (kHz)	Recording rate (kHz)
LHC (Run3)	2×10^{34}	100	> 1
HL-LHC	$5 - 7.5 \times 10^{34}$	1000	10

- Thin Gap Chamber (TGC) system generates Level-0 Muon inputs in the Endcap region ($1.05 < |\eta| < 2.4$)
- To cope with increased collision rate and Level-0 trigger rate, **the readout and triggering electronics of the Thin Gap Chamber (TGC) need to be replaced** during the Phase-II upgrade
 - Needs to establish a system reliable against radiation damage
 - Level-0 trigger : custom made hardware trigger



Thin Gap Chamber (TGC)

Multi-wire proportional chamber
320k readout channel

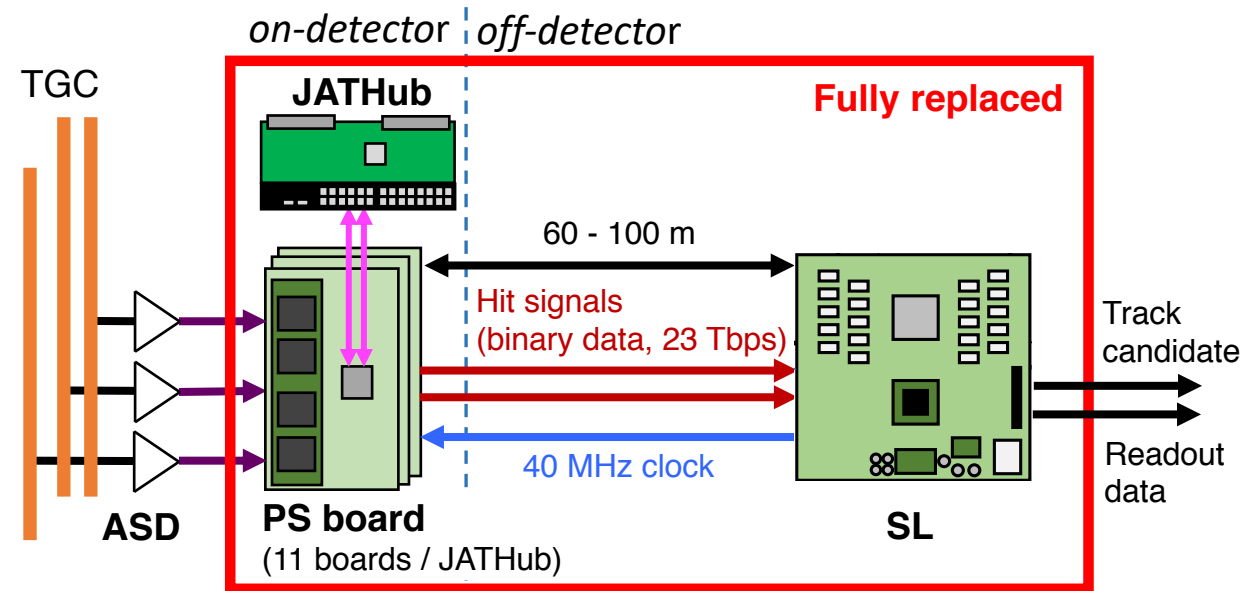
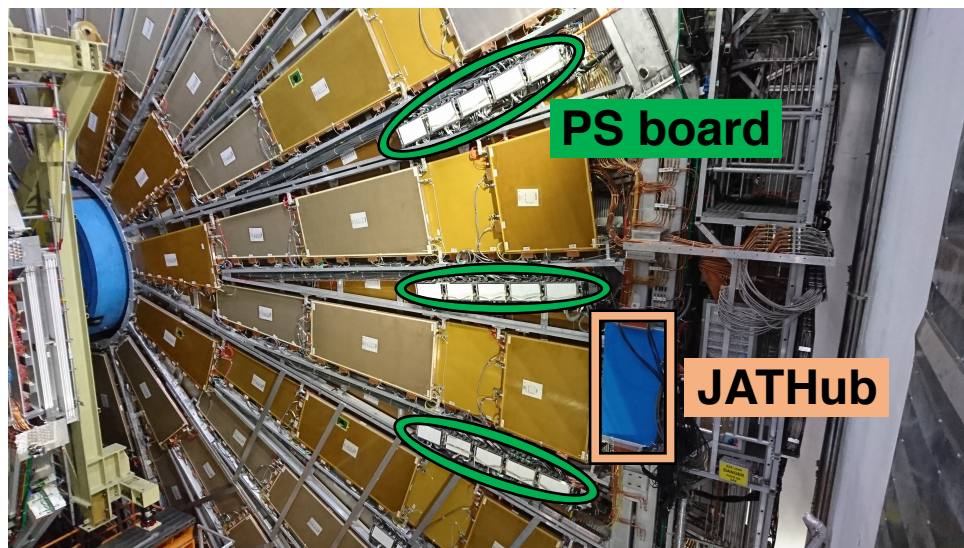
Phase-II TGC electronics overview

• Front-end

- **ASD** (Amplifier - Shaper - Discriminator) (~ 23K boards)
- **PS board** (Primary Processor board) (1434 boards)
- **JATHub** (JTAG Assistance Hub) (148 boards)

• Back-end

- **Endcap SL** (Endcap Sector Logic) (48 boards)



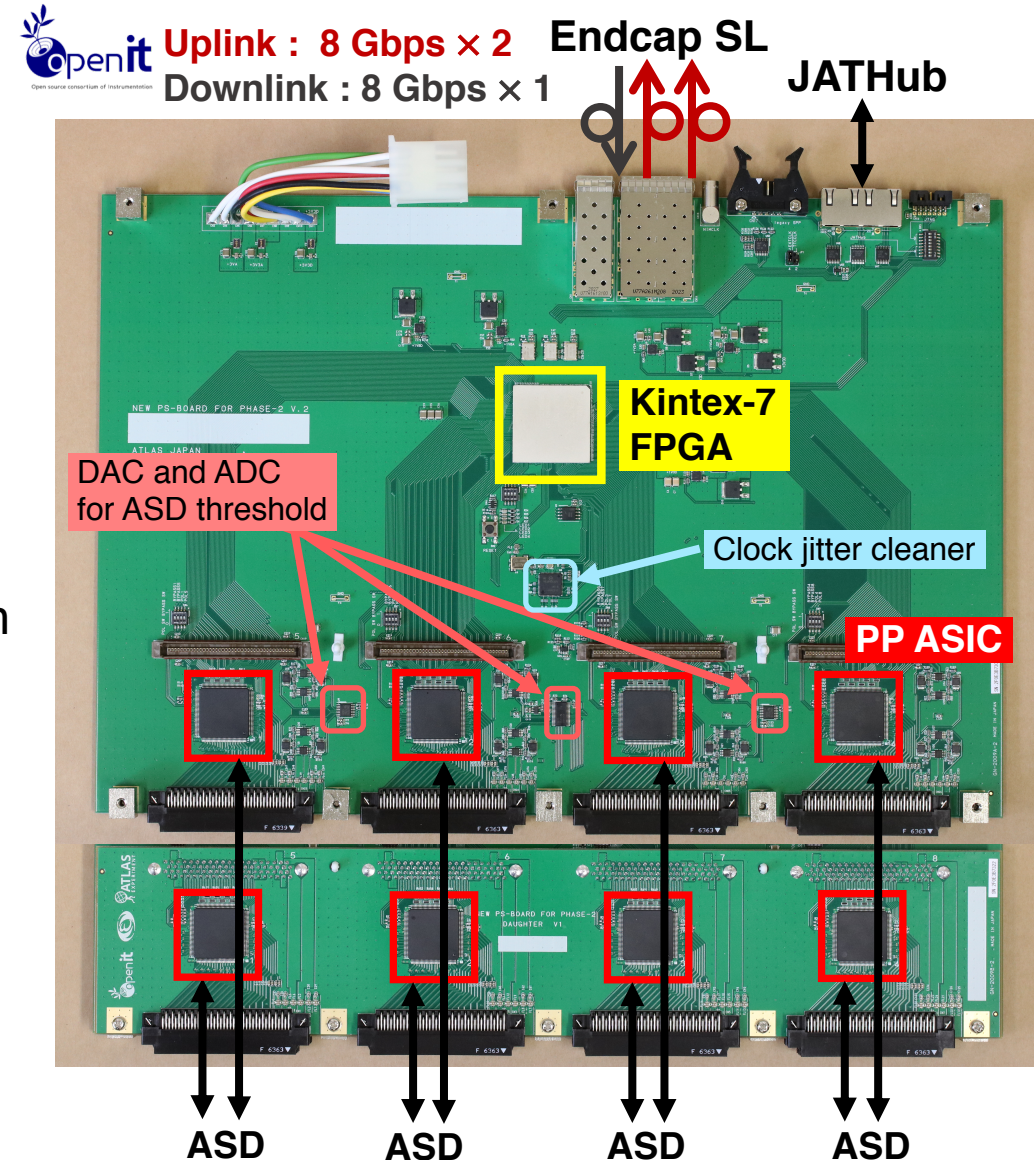
PS board and PP ASIC

• PP ASIC

- Aligns signal timing exploiting variable delay
 - Signal delay variation due to different muon ToF, ASD signal cable length
- Performs Bunch Crossing Identification (BCID)

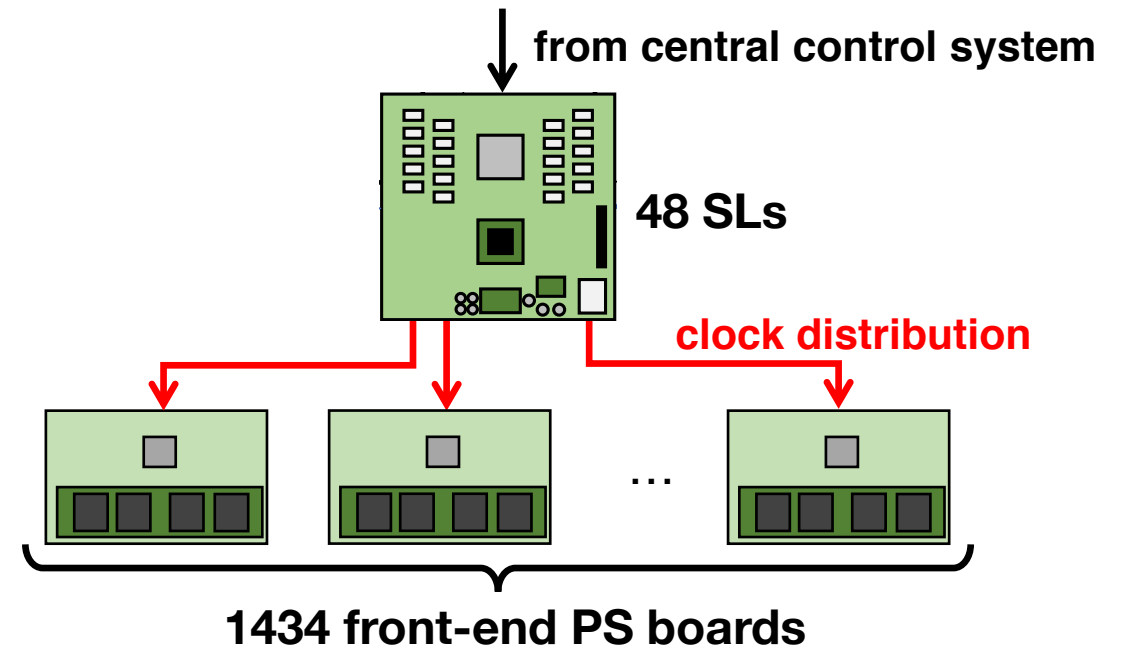
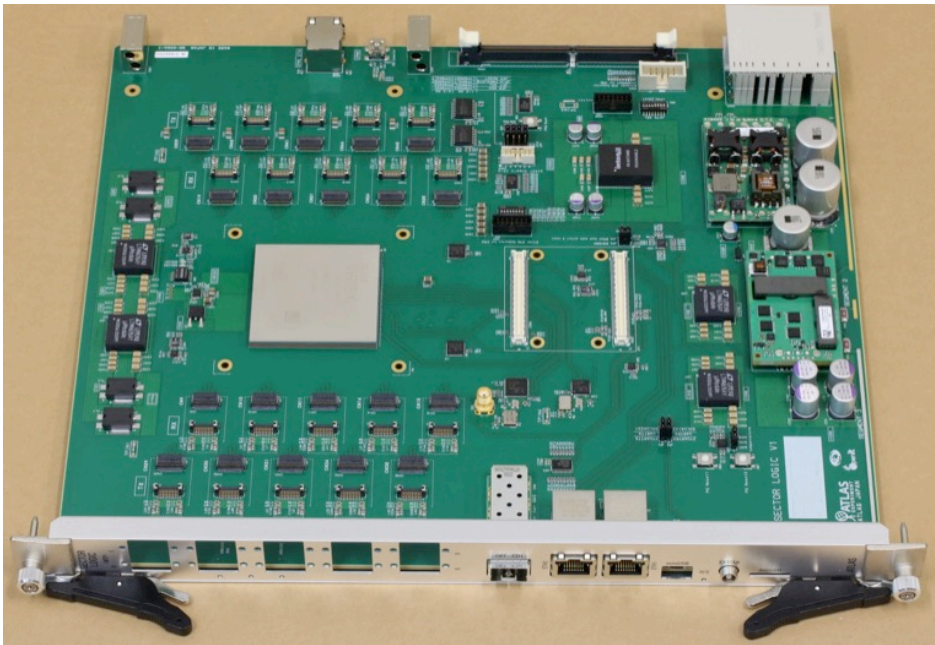
• PS board FPGA

- Receives 40 MHz clock from Endcap SL with fixed latency (clock on data)
 - Clock phase delay variation due to different fiber length between SL and PS board
- Transfers a fixed-length hit bitmap regardless of whether there is a hit or not to Endcap SL via optical link
- Controls thresholds of the discriminator of ASD with DAC, and monitors the thresholds by ADC
 - Optimized threshold value different for each chamber



Endcap Sector Logic (SL)

- Performs muon track reconstruction and estimates p_T using hit signal from TGC
- Readouts hit data for each triggered event from the L0 buffer via optical links
- **Receives 40 MHz clock from the back-end with fixed latency and distributes 40 MHz clock with fixed latency to front-end PS boards**
- Distributes timing & control signals to PS boards
- Controls FPGAs and ASICs on PS boards from Zynq Ultrascale+ MPSoC



Dedicated parameters for PS board, SL

- **PS board (Total ~ 2200 bit per board)**

- **PP ASIC**

- Signal delay (16 bits per ASD)
- BCID gate width (6 bits per ASD)
- Signal mask (16 bits per ASD)

- **PS board FPGA**

- 40 MHz clock delay (12 bit)
- Delay of the timing & control signals
- ASD threshold voltage value supplied by DAC (16 bit per ASD)
- ASD threshold voltage sign supplied by DAC (1 bit per ASD)

- **SL (Total ~ 800 bit per board)**

- Signal delay (25ns step) (16 bit per PS board)
- L0 buffer depth (16 bit)
- RX latch edge select (62 bit)
- Delay of the timing & control signals

Dedicated parameters for PS board, SL

- PS board (Total ~ 2200 bit per board)

- PP ASIC

- Signal delay (16 bits per ASD)
- BCID gate width (6 bits per ASD)
- Signal mask (16 bits per ASD)

- PS board FPGA

- 40 MHz clock delay (12 bit)
- Delay of the timing & control signals
- ASD threshold voltage value supplied by DAC (16 bit per ASD)
- ASD threshold voltage sign supplied by DAC (1 bit per ASD)

- SL (Total ~ 800 bit per board)

- Signal delay (25ns step) (16 bit per PS board)
- L0 buffer depth (16 bit)
- RX latch edge select (62 bit)
- Delay of the timing & control signals

Static parameters

Dynamic parameters

Autonomous Control Mechanism (ACM)

- ACM realizes the advancement and automation of the electronics control system

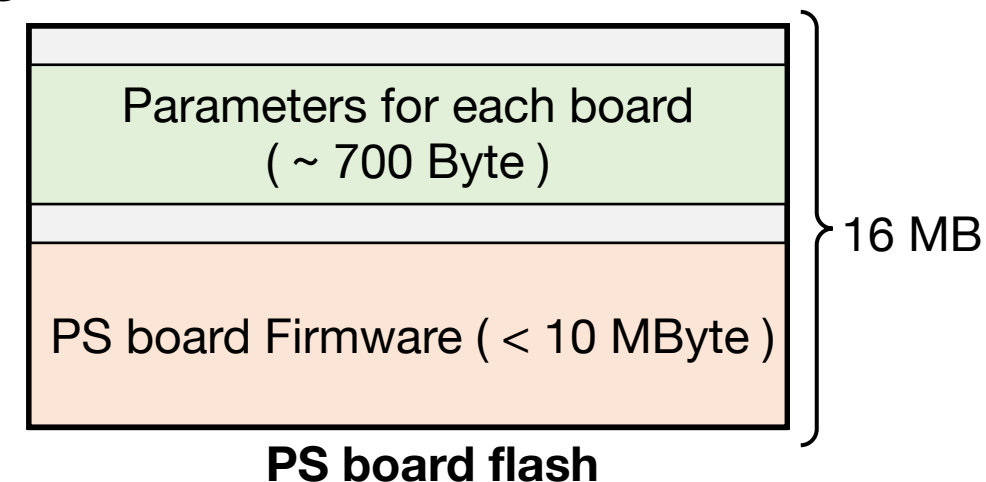
- **ACM on each electronics recognizes the situation by itself**

e.g. FPGA power-up, FPGA reconfiguration, serial link lost,
soft reset signal, transceiver reset signal, reconfiguration signal etc.

and takes the necessary actions automatically

- e.g. (1) Initialize serial link (re-establish clock with fixed latency manner)
(2) Set Element Register
(3) Initialize readout state machine, including buffer clear

- Parameters necessary to configure the elements **are stored in the non-volatile memory on each board**, allowing **electronics to automatically configure itself**



Concept of the ACM

- **Automatic FPGA reconfiguration**

- When the front-end PS board receives a reconfiguration signal or when the PS board power is turned on,
 - (1) Autonomous Control Mechanism **automatically recognizes the situation,**
 - (2) **configures the FPGA,**
 - (3) **executes the procedure to re-establish the link with fixed latency,**
 - (4) **sets the parameters which is different board by board**

(PS board will be fully ready for data acquisition)

- **Automatic transceiver reset**

- When the front-end PS board receives a transceiver reset or when the serial links with SL is lost,
 - (1) Autonomous Control Mechanism **automatically recognizes the situation**
 - (2) **executes the procedure to re-establish the link with fixed latency**

-> **ACM** makes it possible to do these actions **in parallel** in each of the electronics

Parameter setting by ACM (PS board)

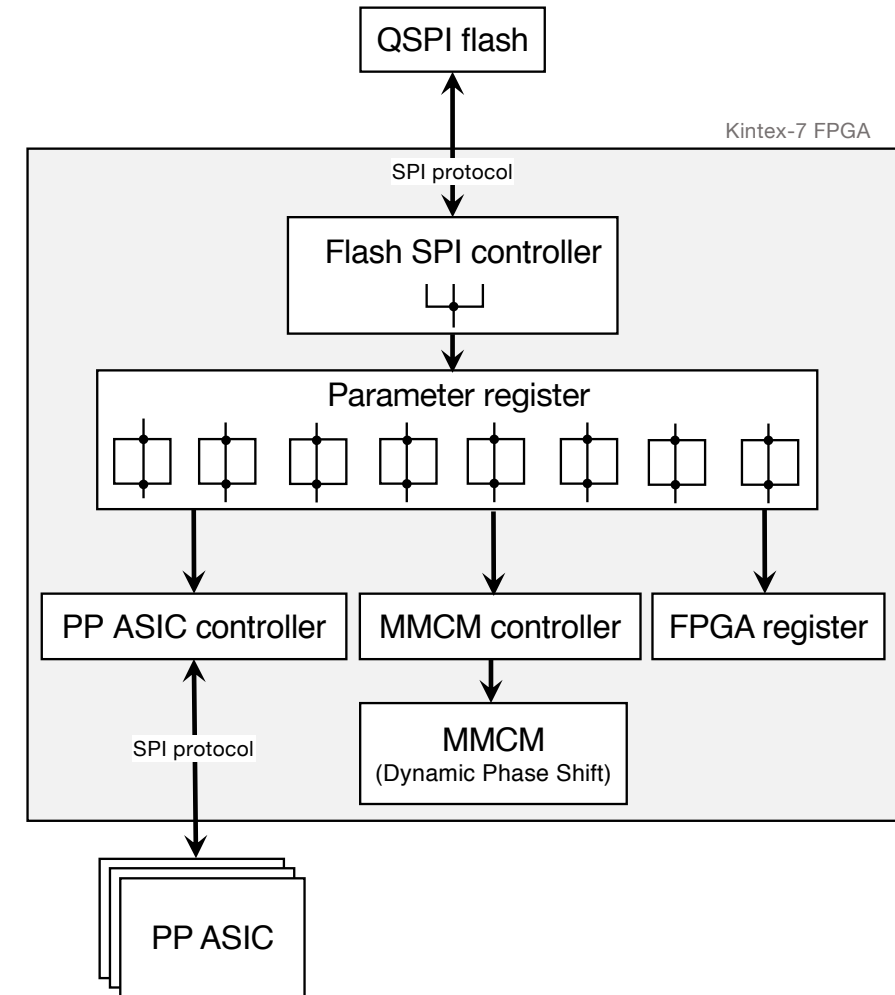
(0) Parameters that need to be written to the non-volatile memory (QSPI flash) of each front-end PS board is first written from the back-end SL MPSoC

(1) **Recognizes that the power is turned on or reconfigured** then the “Flash SPI controller” reads the parameters from the non-volatile memory and store it to the “Parameter register”

(2) “Parameter register” **stores parameters in triplicate to build a robust mechanism for SEUs**

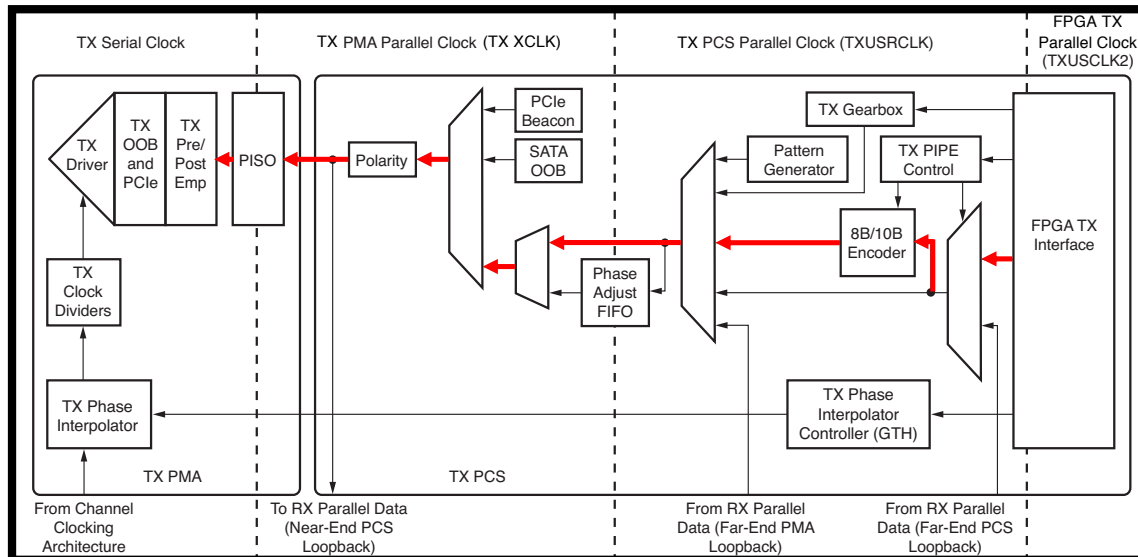
(3) Based on the stored parameters, sets the FPGA registers, PP ASIC parameters, and adjusts the phase of the 40 MHz clock

-> **The time required to configure electronics can be drastically reduced compared to current system**

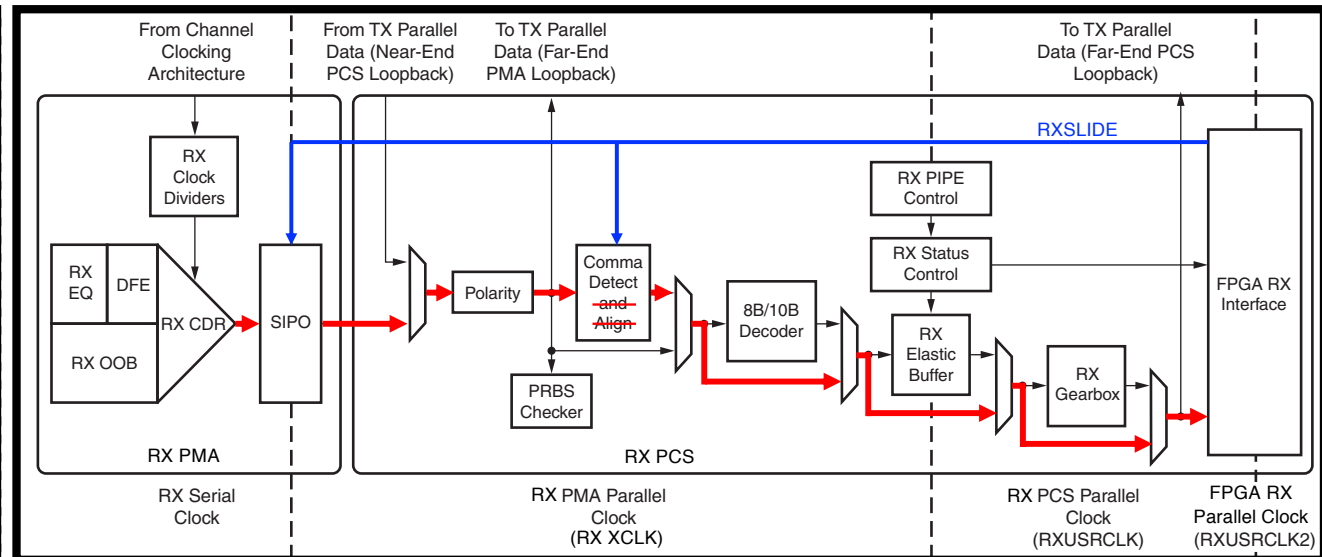


Serial link establishment by ACM

- The following modifications are required to realize clock distribution in fixed latency
 - Clock domain unification in GT transceiver (TX, RX) } Configuration of the GT transceiver
 - Bypass buffers in GT transceiver (TX, RX) }
 - Phase adjustment of recovered clock (RX) -> Performed by ACM**
- ACM performs automatically the multiple steps required to recover the clock in fixed latency and (re-)establish the serial link in the correct order (e.g. TX / RX reset assertion / deassertion, clock phase adjustment, dynamic phase shift)



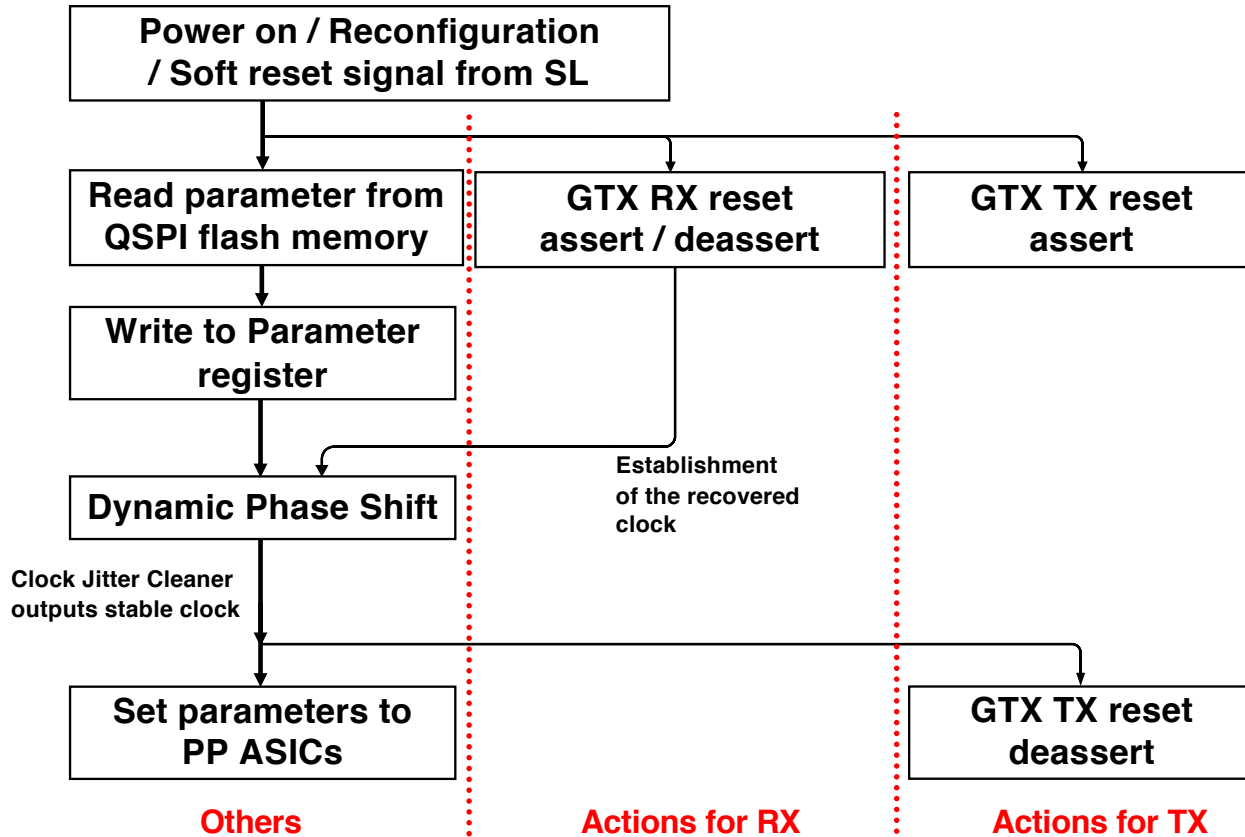
TX



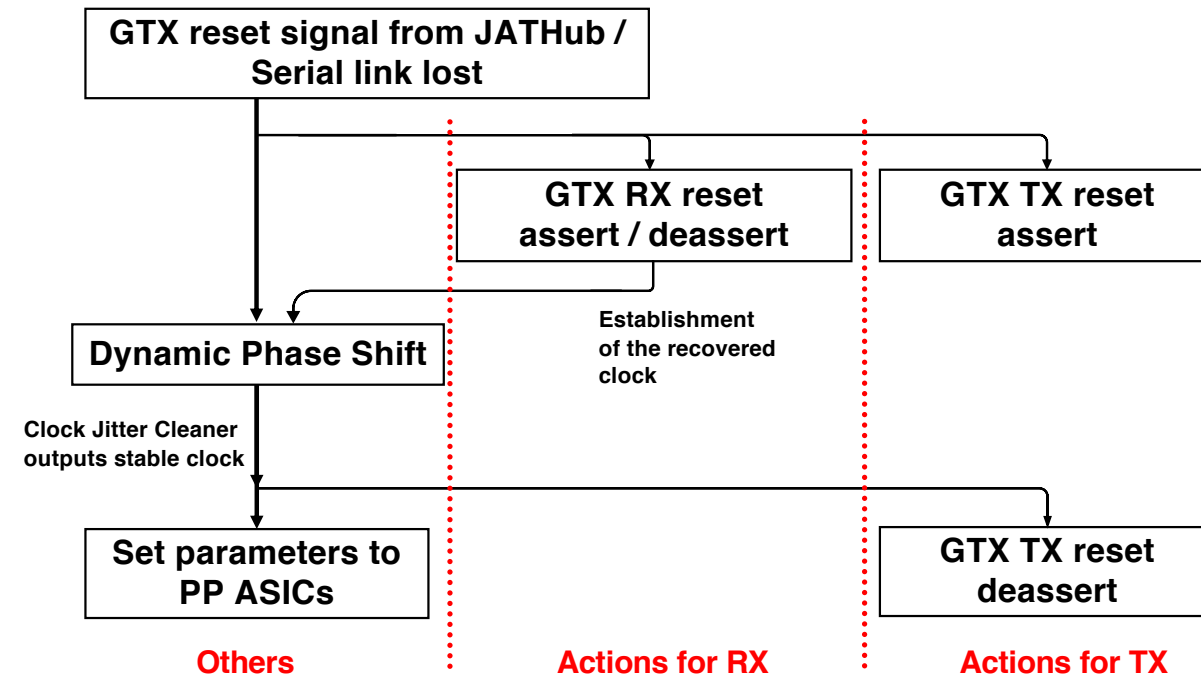
RX

Overall procedure for ACM

Automatic FPGA reconfiguration



Automatic transceiver reset



- The overall procedure of the ACM was constructed so that **Dynamic Phase Shift is performed only after the establishment of the recovered clock is completed** and **data transmission is performed only after the reset on the TX side is completed**

Performance evaluation of ACM on PS board

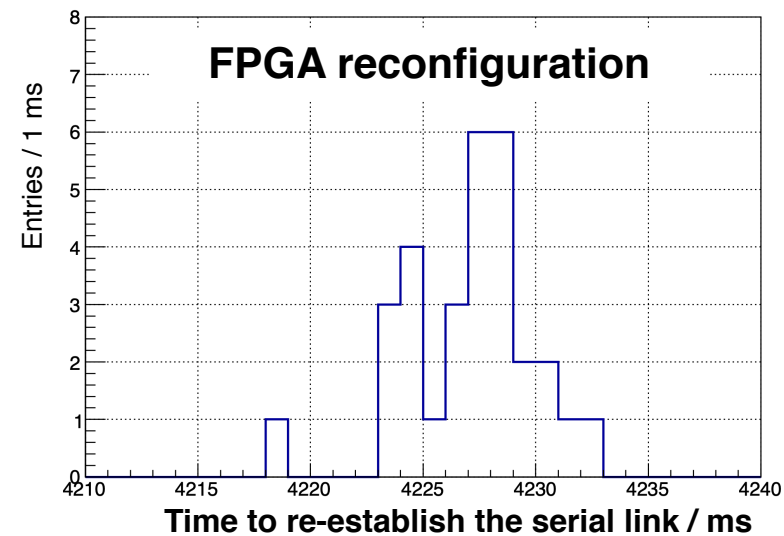
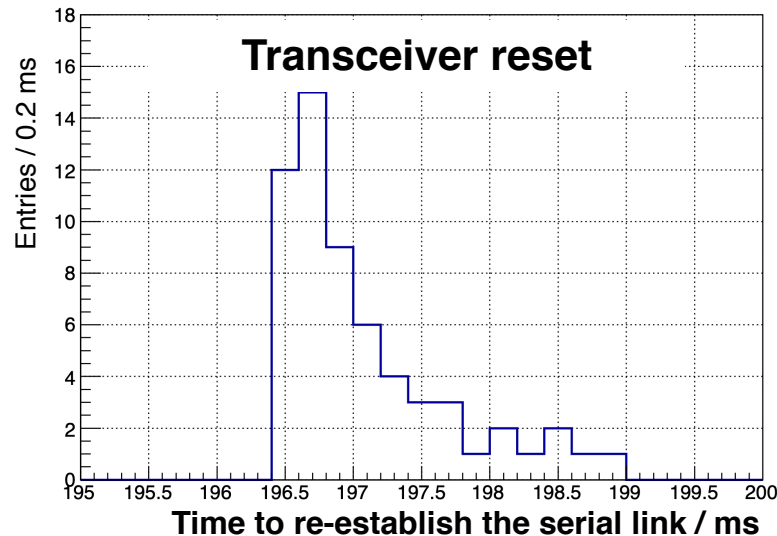
- **Automatic transceiver reset**

- Time it takes for ACM to re-establish the serial link with SL after the transceiver reset signal
- 196.5 ~ 199 ms (99% of the time is spent waiting for the jitter cleaner to output a stable clock)

- **Automatic FPGA reconfiguration**

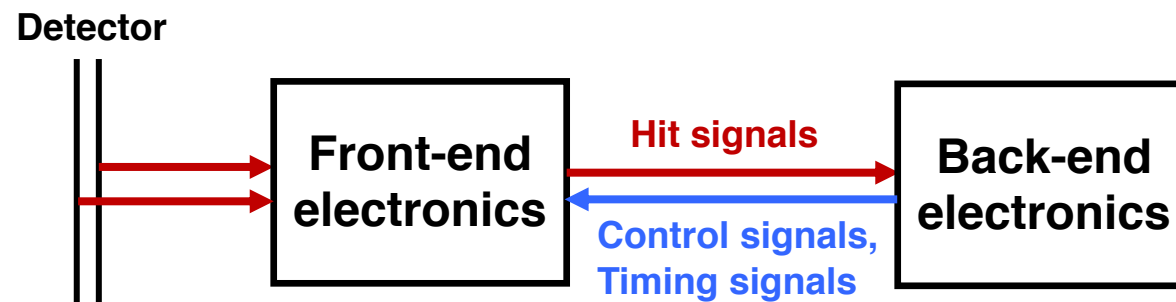
- Time it takes for the FPGA to be configured and ACM to re-establish the serial link with SL after the reconfiguration signal
- 4218 ~ 4233 ms (95% of the time is spent waiting for the FPGA to be configured)

-> **By introducing the Autonomous Control Mechanism, the time required for electronics to be ready can be shortened to < 1/60 compared to the current system**



Operational model realized with ACM

- Once the system is powered on, FPGAs on each board recognize the situation on its own and **automatically make everything work properly**
- All front-end electronics are **automatically configured to optimized parameters** e.g. signal delay parameters, threshold voltage, clock phase etc. with a single reset broadcast or power on
- Electronics automatically recover without signals from the central system, **minimizing the possible downtime** in case of errors during the data taking



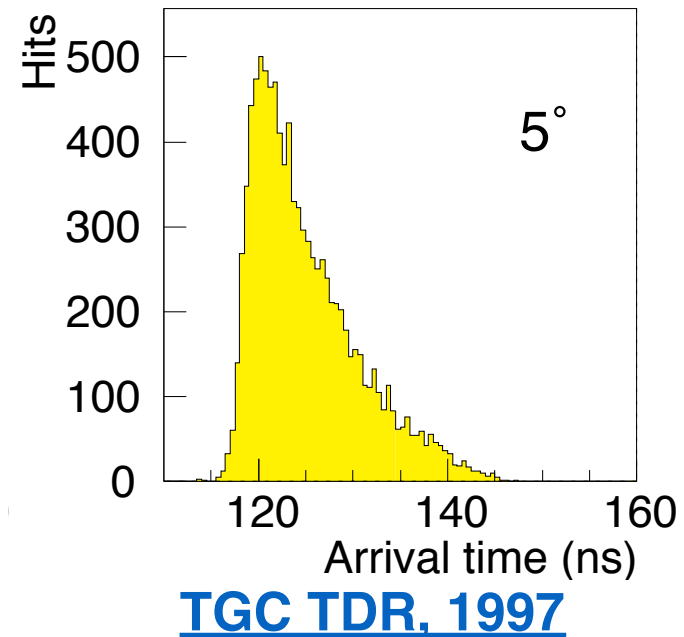
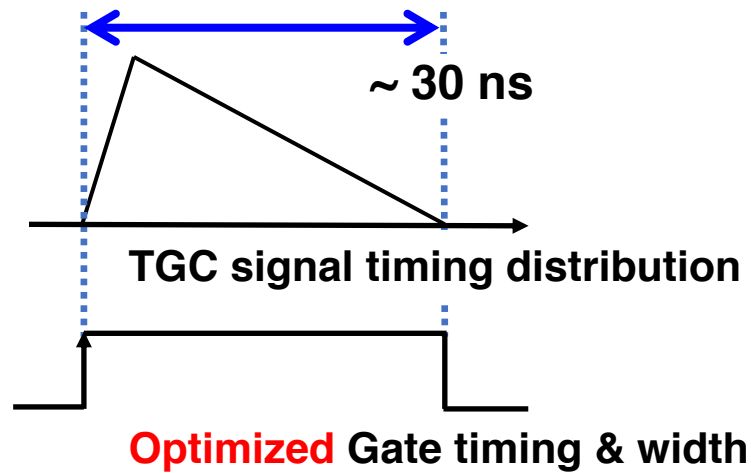
Summary

- **Maximally-automated and self-driven scheme for a large scale electronics system**
 - Establishing a reliable and efficient method to control electronics system is critical
 - Autonomous Control Mechanism realizes the advancement and automation of the electronics control system
 1. ACM will **automatically recognize the system's situation** by monitoring the status of the FPGA device itself and external components to decide a minimally-required configuration procedure without communication with the central control system
 2. ACM firmware will **run the configuration sequence by itself**, including the transceiver initialization in a fixed latency manner
 3. ACM will **retrieve all the individual parameters stored in external non-volatile memory devices**
- > **ACM can be widely applied to electronics systems with FPGAs, flash memory, and high-speed optical links**

Backup

40 MHz clock phase matching at PS boards

- Hit signals from the TGC detector have a width of about 20 ~ 30 ns in the signal arrival time distribution to the PP ASIC
 - In order to perform BCID without missing any hit signals, while picking up as little background as possible that is out of sync with the proton crossing timing, it is necessary to optimize the timing and width of the gate that accepts the hit signal
- > To align signal timings and minimize gate widths, the phases of the 40 MHz clocks on all 1434 front-end PS boards must be aligned with sufficient precision $\mathcal{O}(100\text{ ps})$**



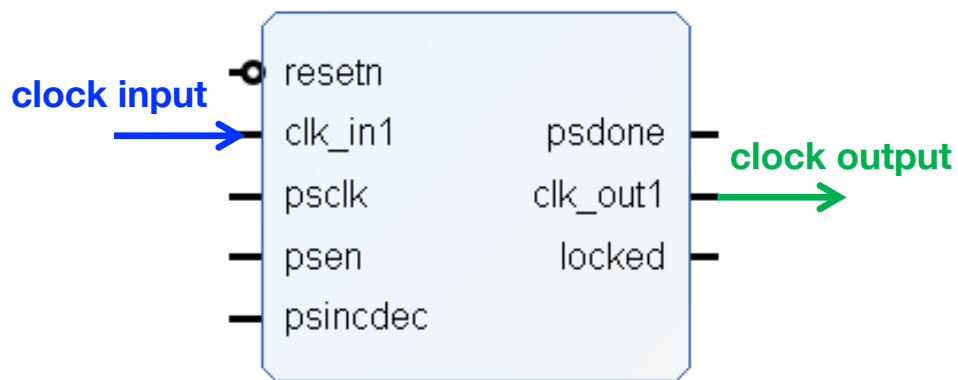
How to match the clock phase

- **Clock distribution with fixed latency**

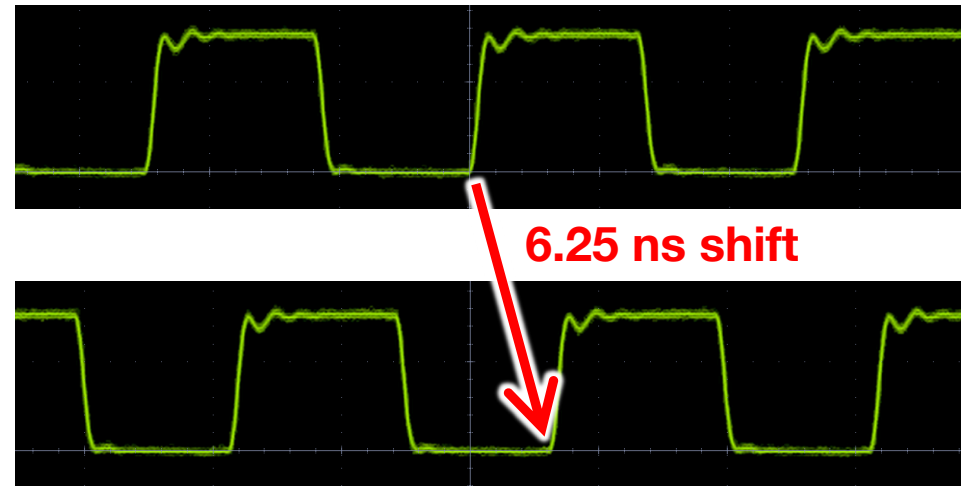
- “Fixed latency clock distribution” where the phase of the reconfigured clock on the PS board does not change upon reset or reconfiguration of the SL or PS board

- **Clock phase adjustment**

- Lengths of the fibers between SL and PS boards are not all the same
-> It is necessary to adjust the phase on PS boards appropriately to align the clock phases between all PS boards with sufficient precision $\mathcal{O}(100\text{ ps})$
- By using the "Dynamic Phase Shift" feature in the Clocking Wizard IP core from Xilinx, the phase of a 40 MHz clock can be adjusted remotely in steps of 1/56 ns

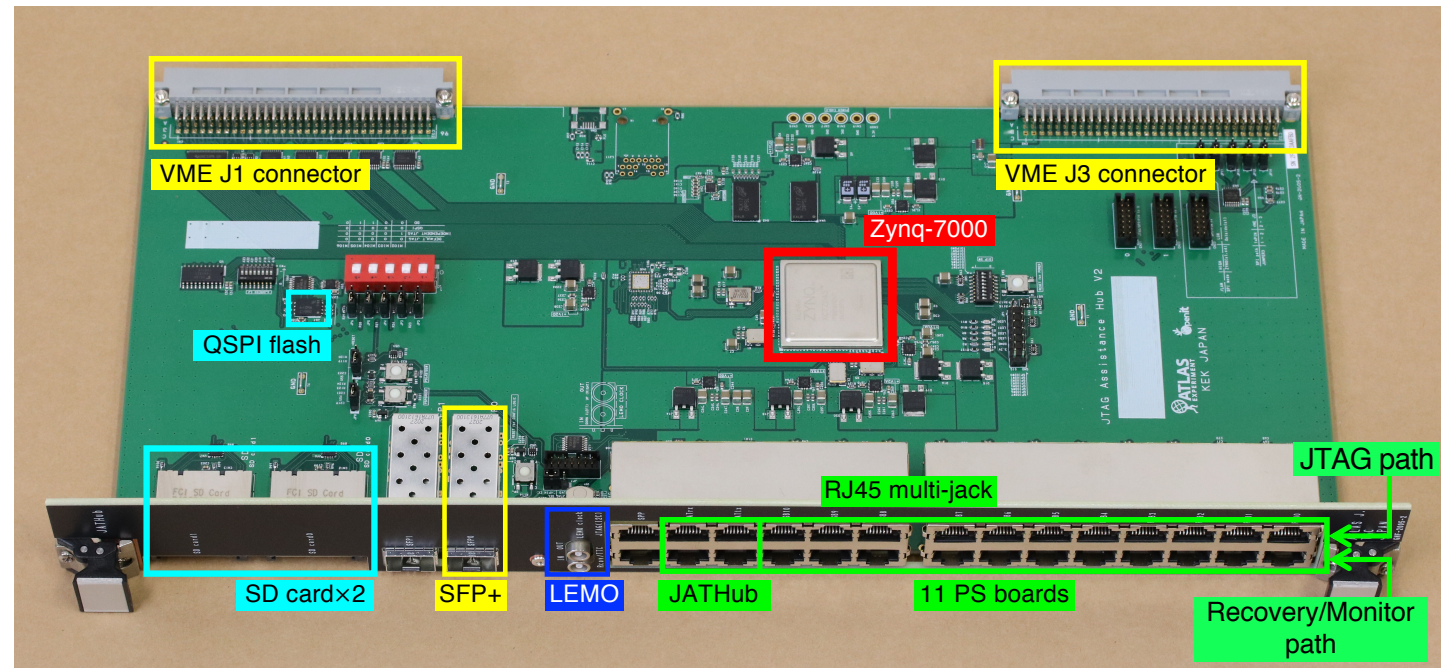
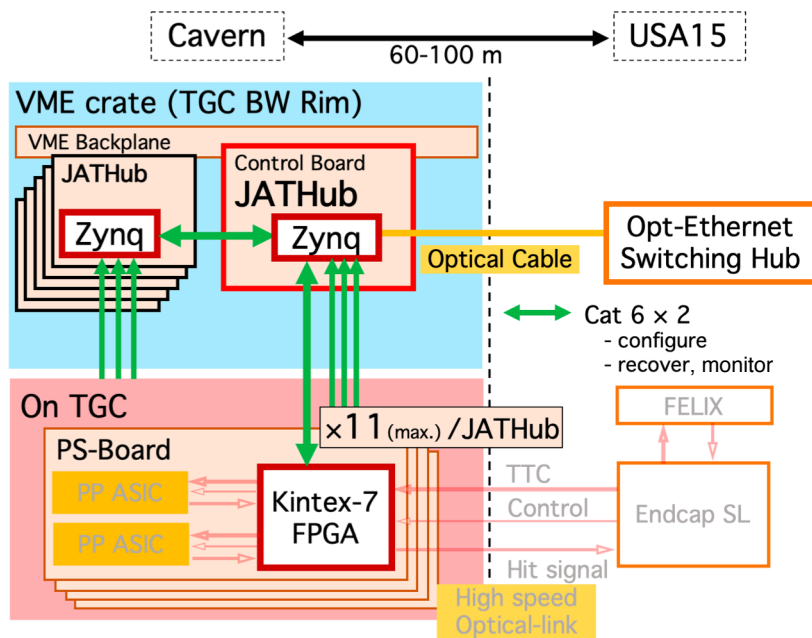


Clocking Wizard IP core

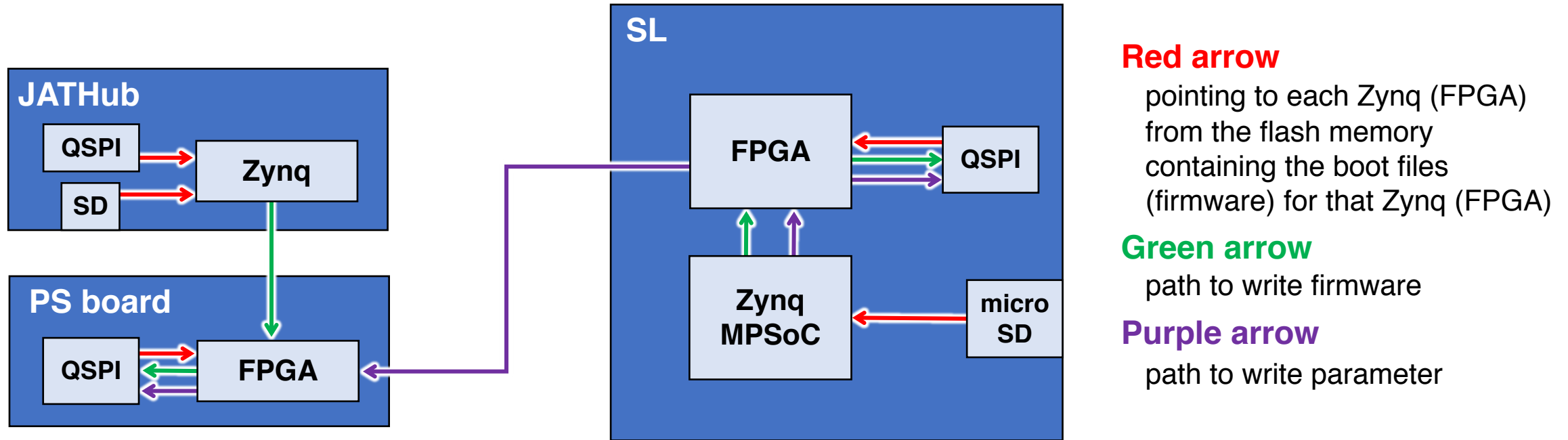


JATHub

- JTAG communication for programming, debugging FPGA on front-end electronics
- Recovery procedures for front-end electronics
 - SEM controller + recovery by JATHub on request
- Monitors clock phase of PS boards
 - To align clock phases on PS boards with O(100 ps) for high performance hit BCID

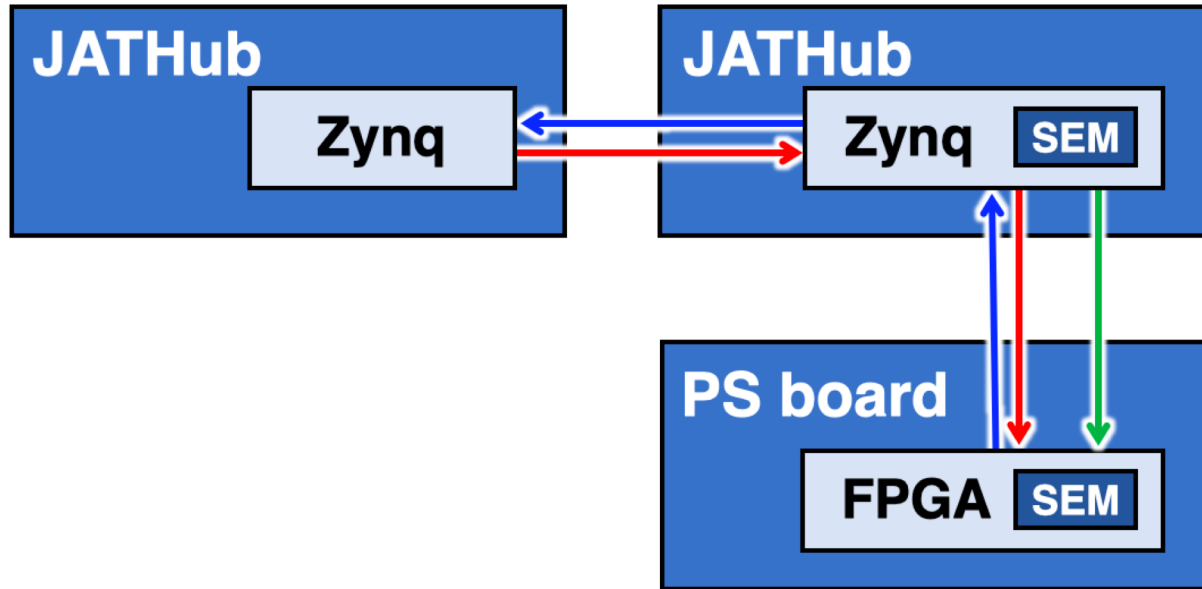


Boot sequence of the whole TGC system



- Zynq-equipped boards in the ATLAS counting room and ATLAS cavern, will allow FPGA firmware to be updated and written to flash memory remotely
- Writing parameters from SL Zynq MPSoC to each flash memory, enabling parameters to be obtained via Ethernet and parameters to be updated and written remotely

Recovery sequence of the whole TGC system



- SEM in each FPGA will recover the single bit SEU
- When a non-recoverable SEU occurs, each FPGA sends a rescue signal to the connected JATHub (**blue arrow**), and the recovery signal from the JATHub (**red arrow**) causes the FPGA to reconfigure to resolve the SEU
- To reset the serial communication line between the PS board and the SL, use the transceiver reset signal (**green arrow**) from the JATHub

- JATHub installation in the ATLAS cavern allows fully automated handling of unrecoverable SEUs and remote resetting of FPGA transceivers