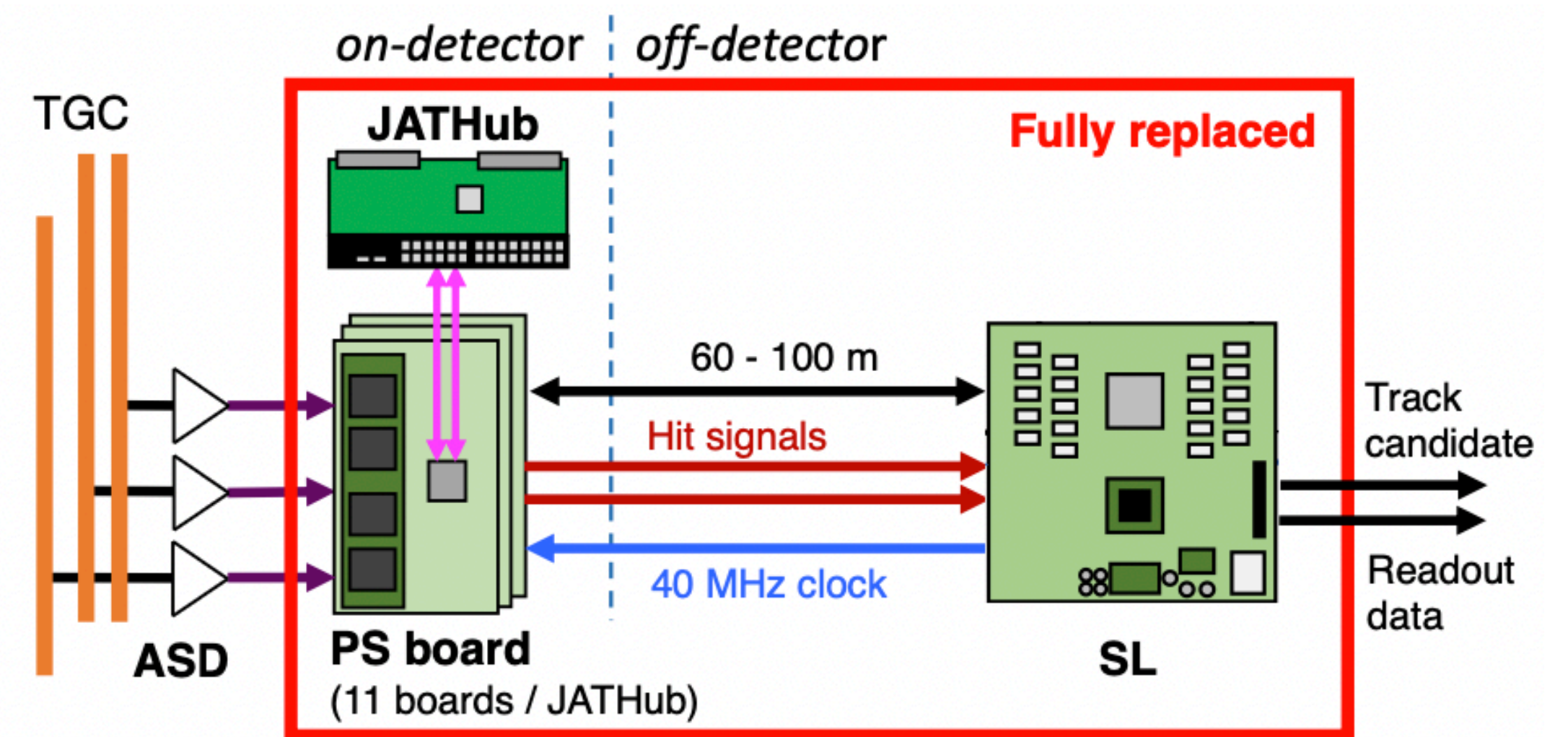
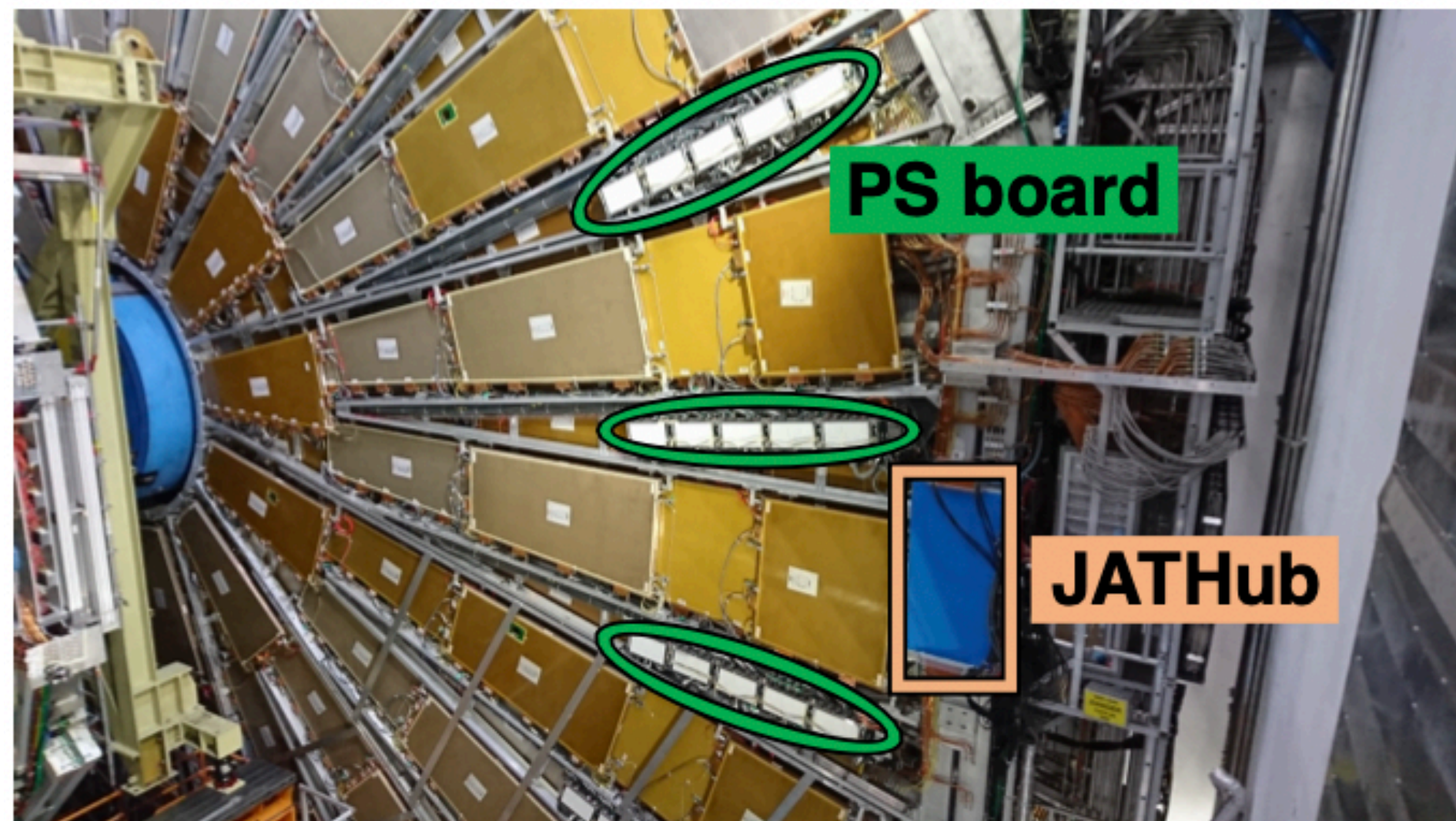

**高輝度LHC-ATLAS実験
前段回路の量産後試験に向けたSoCデバイスを活用した
コンパクトなDAQシステムの実装**

東京大学ICEPP 奥村研究室 M1 成川佳史

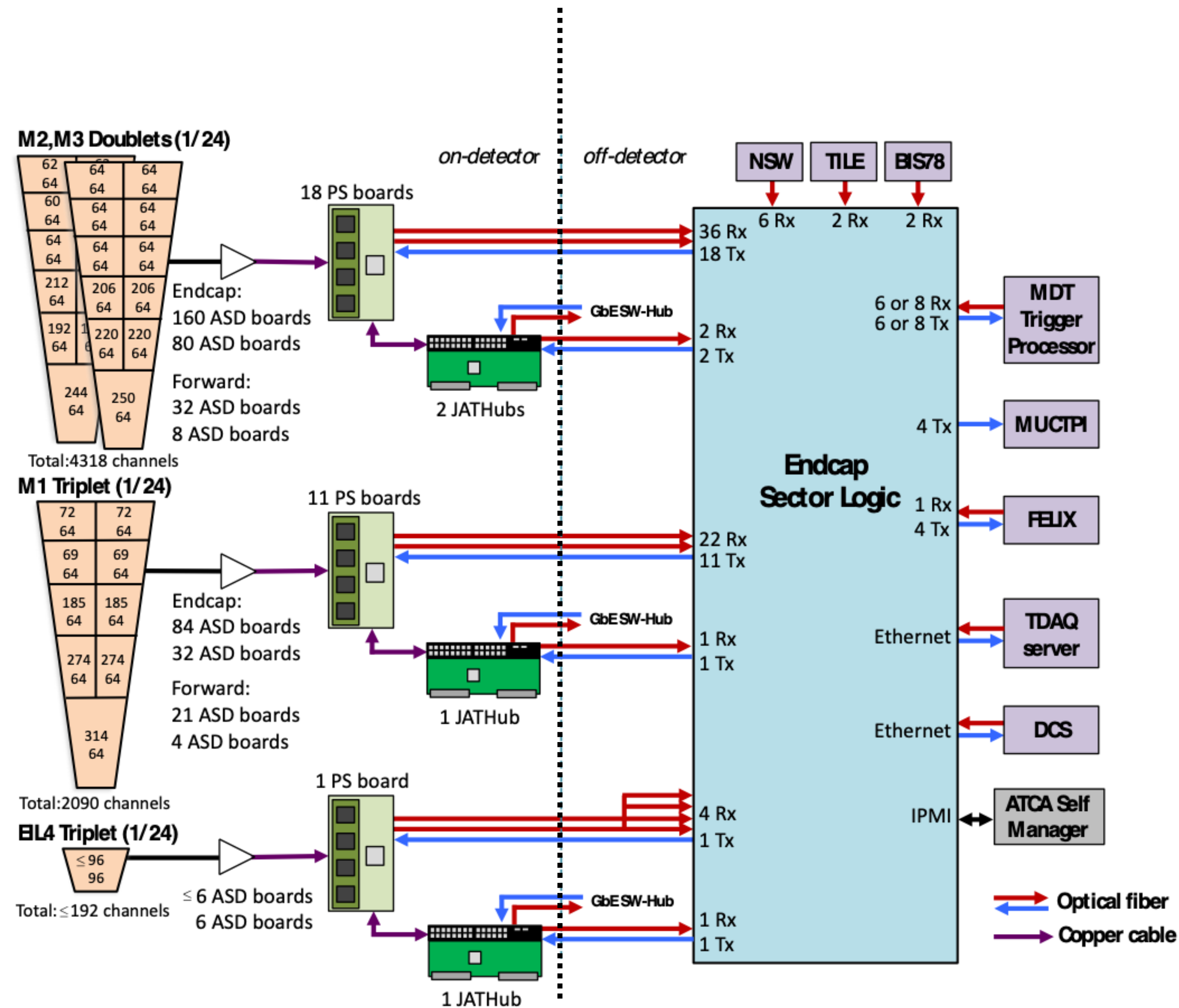
TGC検出器トリガーエレクトロニクス

- 2029年から始まるHL-LHCに向けてTGCエレクトロニクスは刷新される
 - ミューオンがTGCに入射、電流信号はASDボードに集められる
 - ASDボードで電流->電圧の変換、増幅、閾値電圧による信号識別を経てデジタル信号に変換
 - PS boardにて信号がどの陽子バンチ衝突に由来するか識別(**Bunch Crossing IDentification**) される
 - ヒットの有無を表すビットマップ(16ASD x 16ch = 256 ch)が生成され高速光通信を通じて、回路室のSLへと送られる
 - SLでは31枚のPS boardからのヒット情報をもとに飛跡再構成、運動量概算が行われる
 - JATHubモジュールはトリガーパスとは独立にPS boardの制御を行う



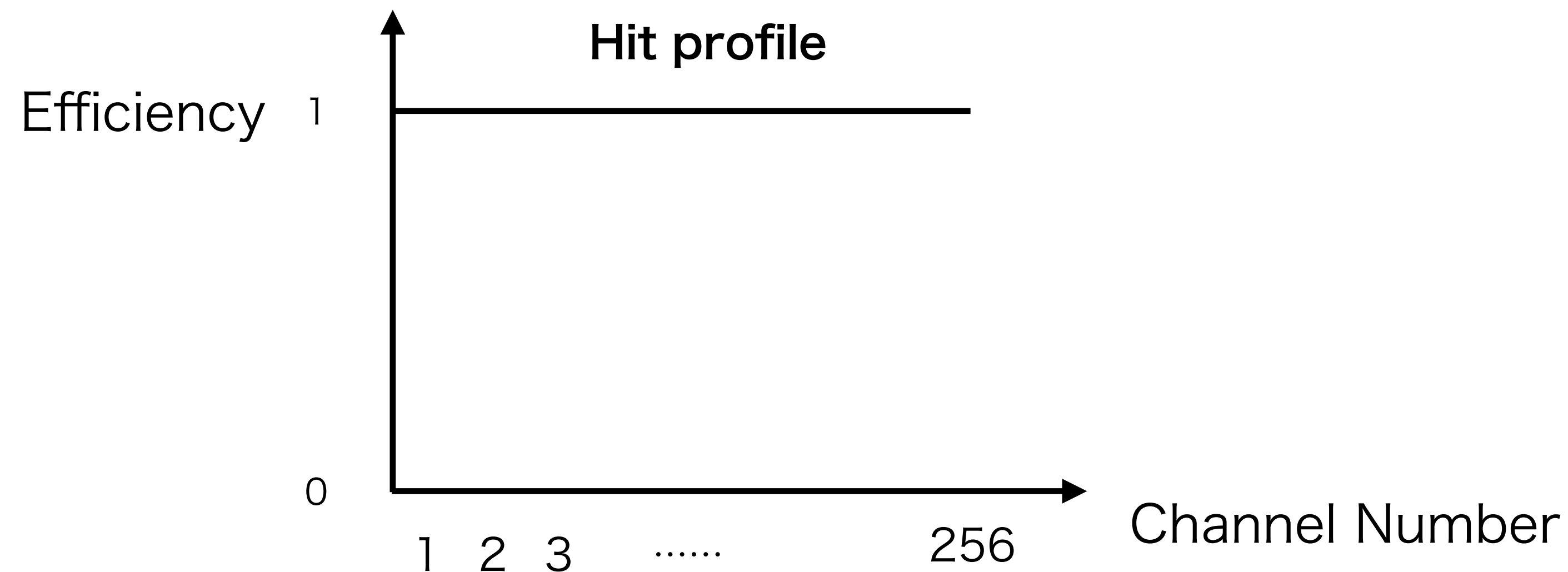
TGC検出器におけるTTCシステム

- 各ボードがLHCの陽子バンチ衝突と同期して動作するためにTTC(Timing, Trigger and Control)が分配される
 - 40MHz LHCバンチ交差クロック
 - BCR(Bunch Counter Reset)(バンチ識別番号のリセット)
 - TPT(Test Pulse Trigger)
- PS boardは光通信を通してSLからTTC信号を分配される



TGC検出器におけるテストパルス機構

- トリガーパスを担う各モジュールの機能検証にミュオンヒット信号を模したテストパルスを発行する2種類の仕掛けが備わっている
 - ASDテストパルス
 - PS board FPGAテストパルス
- テストパルスを用いたトリガーパスの評価は最終的にHit profileとしてまとめられる



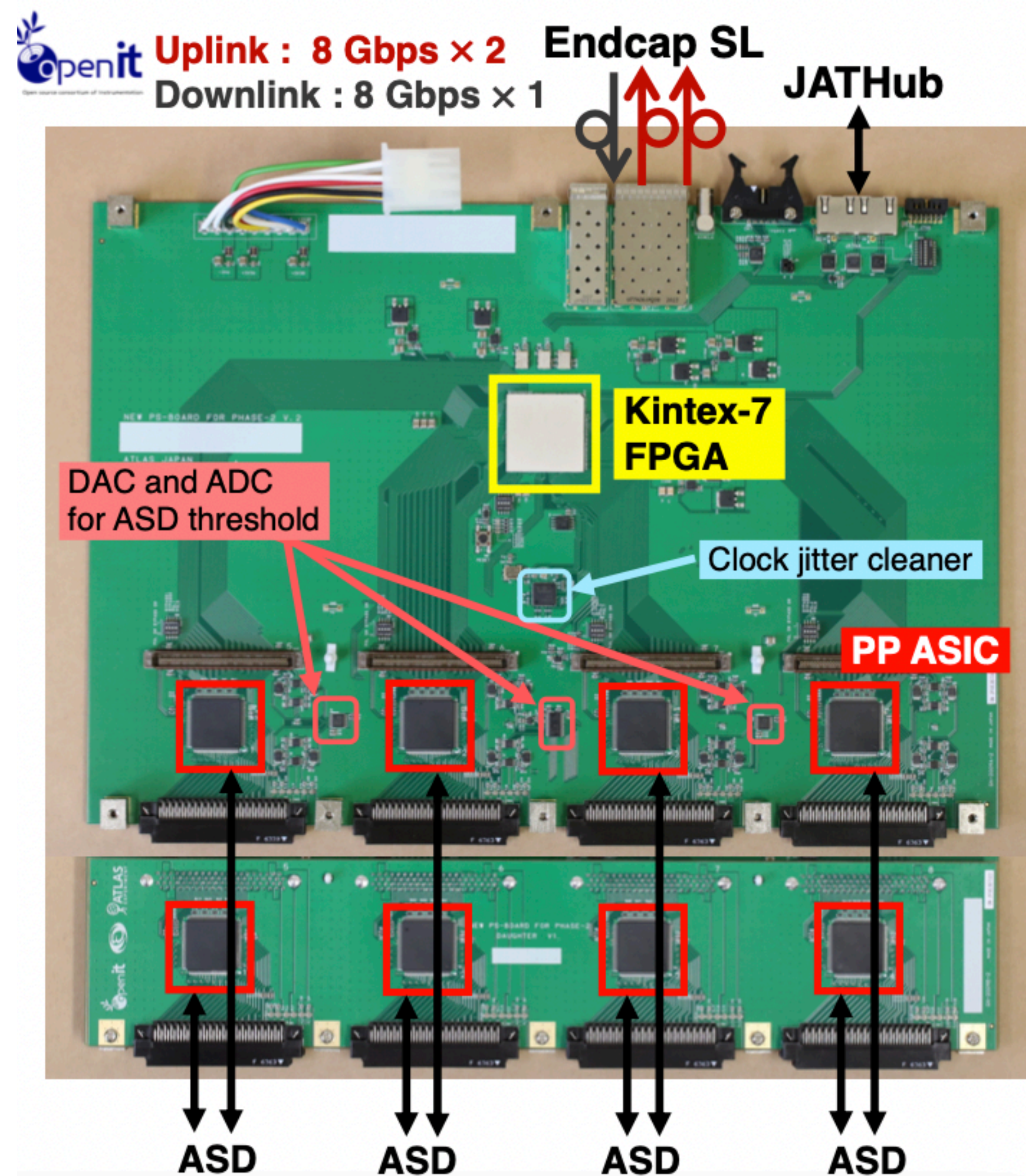
PS boardの概要

• PP ASIC

- ASDから32ch(16ch x 2)のヒット信号を受信
- 可変遅延を用いたタイミング合わせ(陽子衝突点からチェンバーまでのToFやASDまでのケーブル長の差を吸収)
- BCID
- ASDへのテストチャージの発行

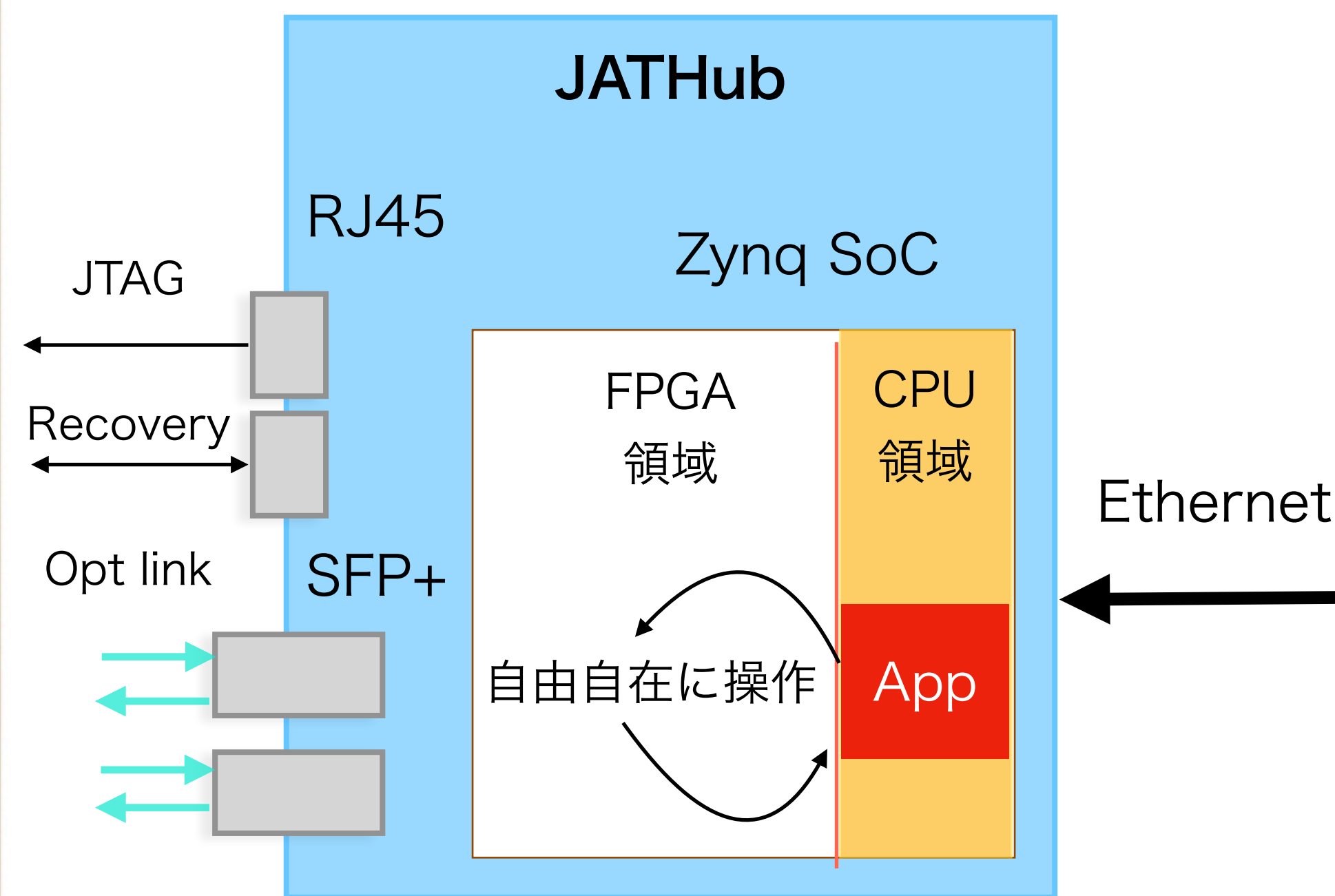
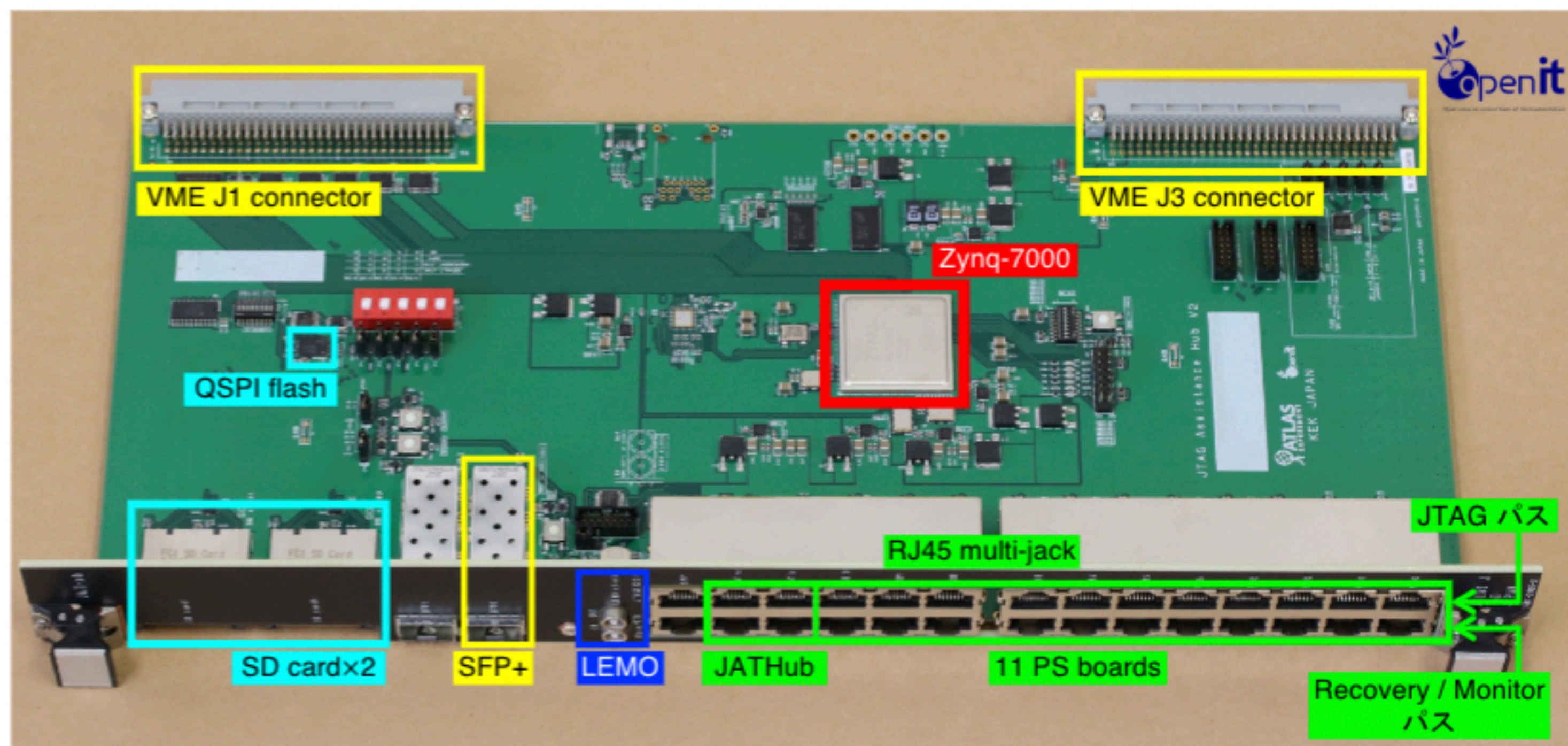
• PS board FPGA

- SLから高速光通信を用いてTTC信号を受け取る
- PP ASICからヒット信号を集め256bit(8PP ASIC x 32bit)のヒットビットマップをSLに送信



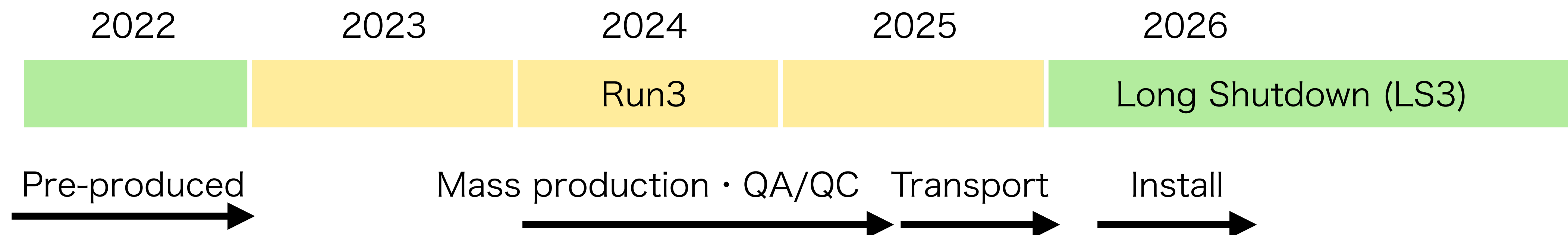
JATHubモジュールの概要

- Zynq SoCを中心に光シリアル通信、Ethernetを搭載したハイスペックボード
 - Zynq SoC(System-on-a-chip)：CPUとFPGAが一体となったチップ。On-boardのソフトウェアを起点にFPGAの操作やFPGAからのデータ受信が可能=>手軽に高度な処理を実現
 - 光シリアル通信：8Gbpsの高速シリアル通信=>大量のデータの送受信



PS board QA/QC試験へのアイデア

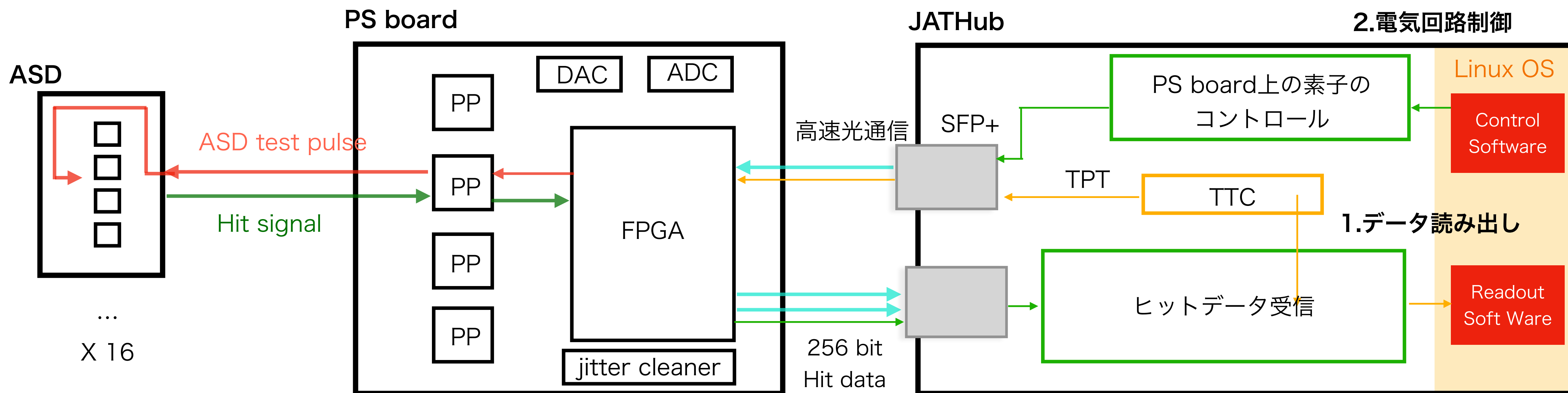
- PS boardはこれまでに設計、開発、プレ量産が完了し2024年度から1500枚に及ぶボードの量産・QA/QC試験が始まる



- QA/QC試験(品質保証試験)としてテストパルス機構が有用
 - しかし、SLは開発中
 - SLの駆動には大掛かりなセットアップが必要
- Zynq SoCと高速光通信を備えたJATHubを活用してコンパクトなDAQ(Data Acquisition)システムを開発
 - (1)データ読み出し (2)電気回路制御 (3)タイミング信号の分配 を一台で！

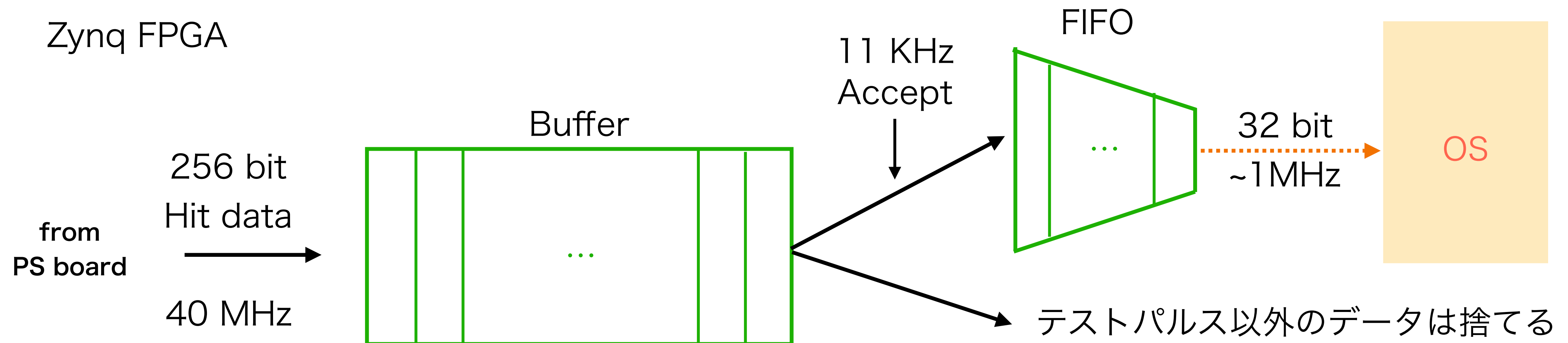
テストパルス試験に向けたコンパクトDAQシステムの実装

1. **データ読み出し機能**：高速光通信を用いた256bit ヒットデータ受信、Zynq上のCPUから読み出し
2. **電気回路制御**：PS board上の素子(FPGA, PP, ...)をZynq上のソフトウェアで制御
3. **タイミング信号分配**：40MHz clockをJATHub内部のOn-board oscillatorから生成、その他のTTC信号(BCR,TPT)と合わせて高速光通信を用いて分配



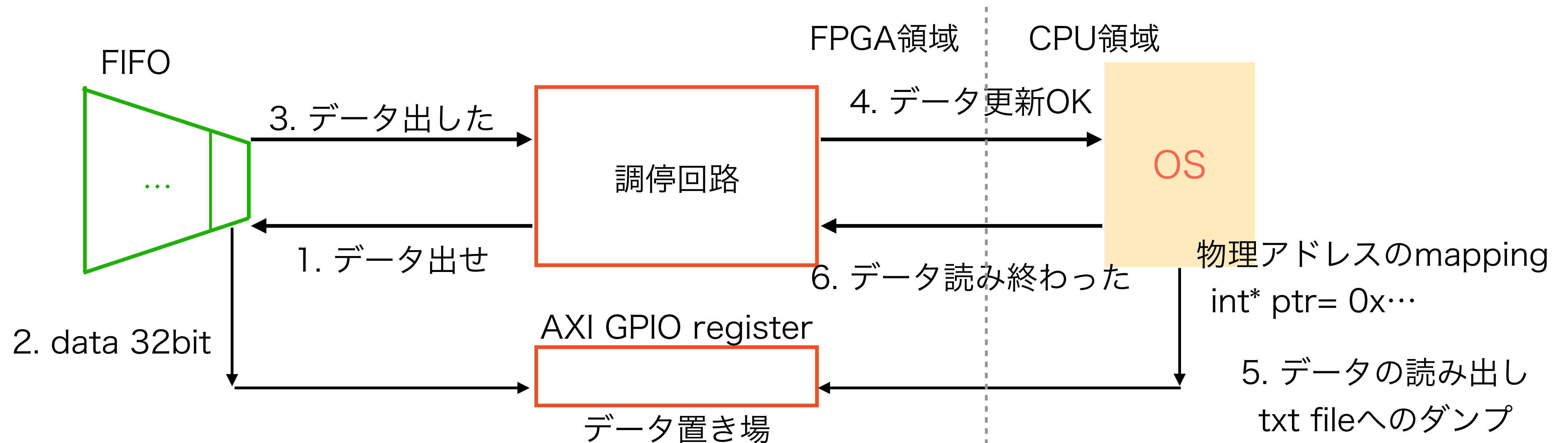
データ読み出し機能実装(1)

- CPUによるデータ読み出しのレイテンシーは1MHzが限界
- テストパルスを1MHzより低いレイテンシー(11KHz)で発行
- 40MHzで受信するヒットデータの中からテストパルスのみを抜き出す
 - TPTを発行してからテストパルスが返ってくるまでの時間をあらかじめ計測(ケーブル長が主なパラメータ)
 - 40MHzで受信する全データを深さ一定のBufferに保存
 - テストパルスのタイミングに合わせてAccept signalを発行しFIFO(First In First Out)メモリーに入れる



データ読み出し機能実装(2)

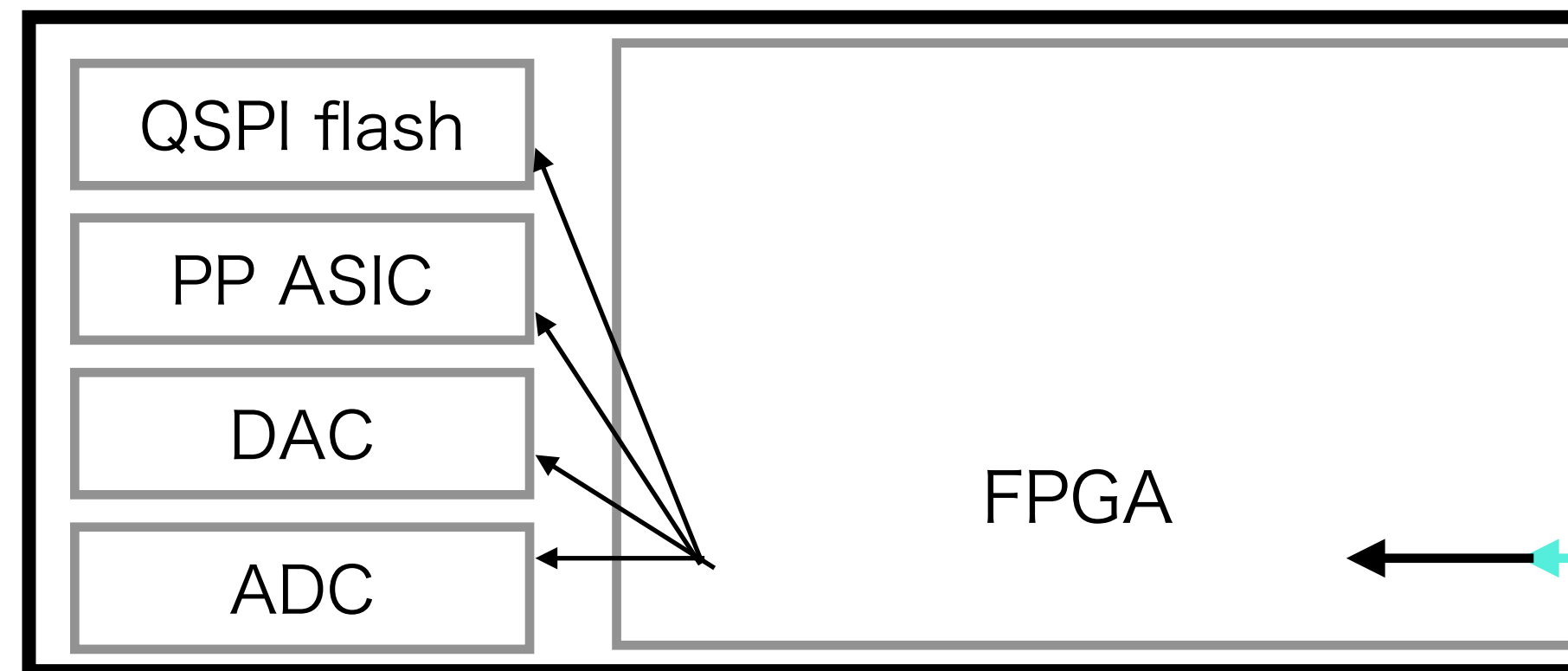
- FIFOメモリーからCPUへのデータ読み出しはAXI GPIO register(CPUからもFPGAからもアクセスできるレジスタ)を利用
- データの読み飛ばし、重複を防ぐため調停回路を作成
- FIFOによるレジスタの値の更新とCPUからの値の読み出しが必ず交互に実行される



電気回路制御機能

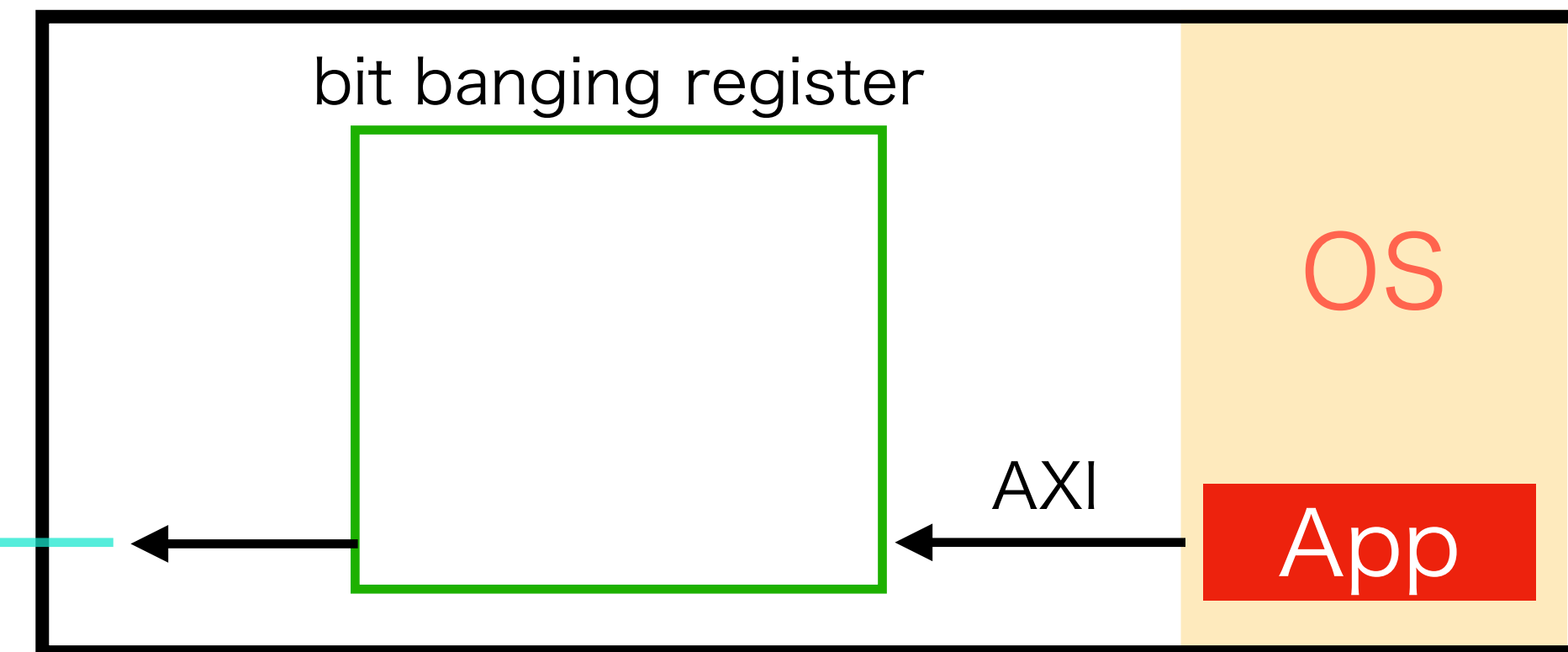
- Zynq上のソフトウェアからPS board上の素子(FPGA, PP ASIC, DAC...)を制御
- 光通信を介してJATHubのOSから各素子のレジスタへ直接Bit banging
 - AXI GPIOを利用してCPUからJATHub FPGA上のbit banging用registerを操作 cf) VME
 - JATHub上のbit banging registerの値は光通信を通じてPS board FPGA、各素子へと伝搬

PS board



SPI, I2C プロトコルに従って
Bit banging

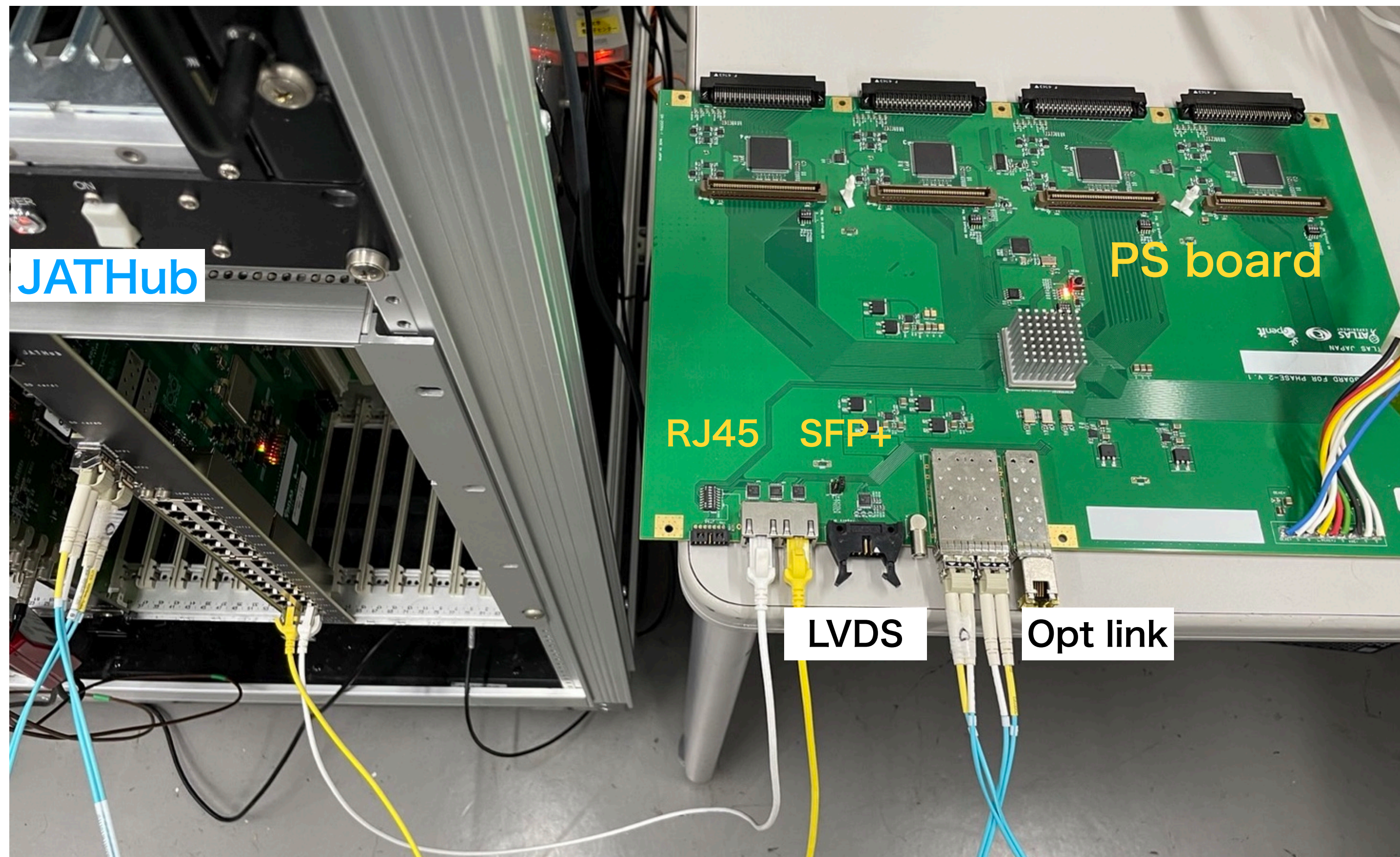
JATHub



自動で光通信に乗せて
PS board FPGAへ

コンパクトDAQシステム機能検証 @東京大学

- コンパクトDAQシステムの検証のため、PS board FPGAテストパルス試験を実施した
- JATHubからPS boardに~11KHzでTPTを発行し、PS board FPGAから発せられたヒット信号をCPUから連続的に読み出すことができることを確かめる



検証結果

CPUから読み出された3BC分のデータ

• CPUからヒット信号を連続的に読み出すことに成功

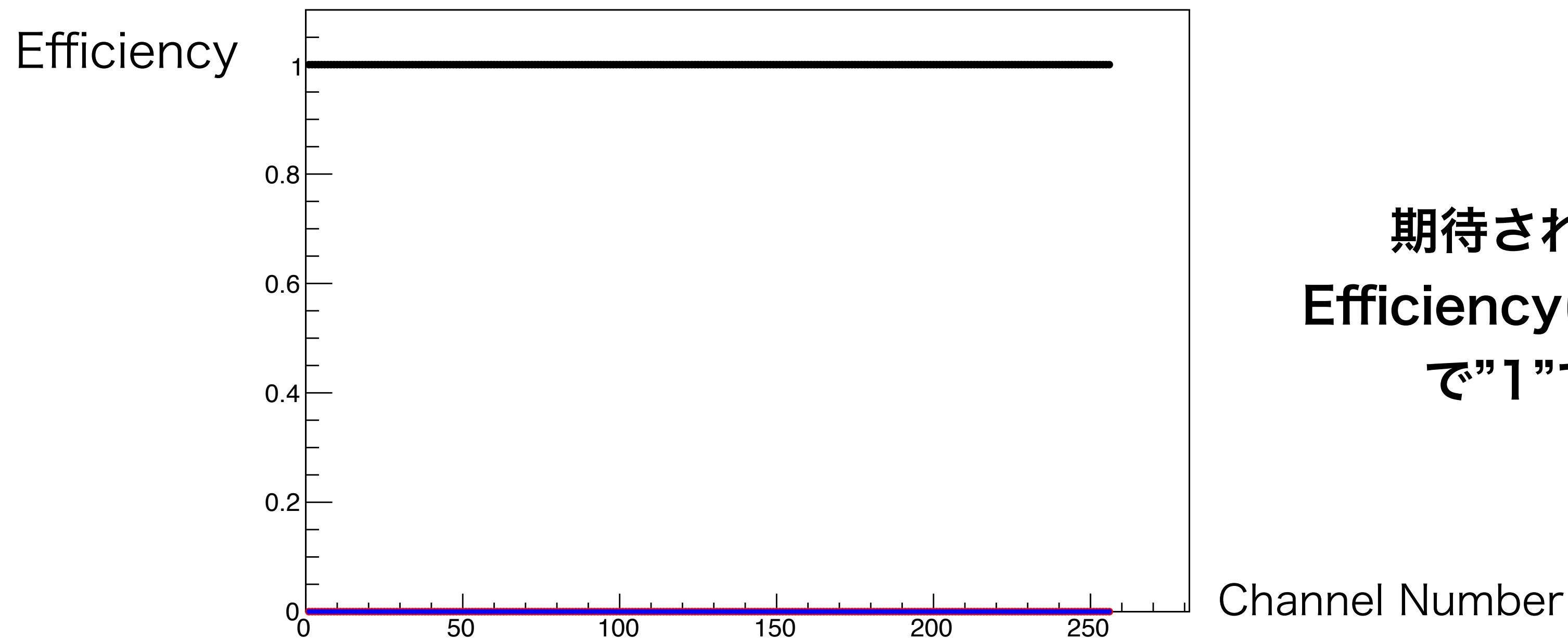
1. データ読み出し：テストパルスデータの選択的取り出し、固定レイテンシーでのDAQ
2. 電気回路制御：JATHub CPUを起点にしたPS board各素子の制御
3. タイミング信号分配：On-board oscillatorを用いた40MHz clockの生成、TTC信号の分配

```
0xb0d00000
0x0000001f
0x00810777
0x0001f004
0x00000000
0x00000000
0x00000000
0x00000000
0x00000000
0x0000f004
0x00000000
0x00000000
0x00000000
0x00000000
0x00000000
0x0101f005
0xffffffff
0xffffffff
0xffffffff
0xffffffff
0x0100f005
0xffffffff
0xffffffff
0xffffffff
0x0201f006
0x00000000
0x00000000
0x00000000
0x00000000
0x00000000
0x0200f006
0x00000000
0x00000000
0x00000000
0x00000000
0x00000000
0x00000000
0x00000000
0x001ee0d0
0xb0d00000
```

取得された256bitの
テストパルスデータ

256 bit "1"

10,000回のテストパルスに対するHit profile



期待された通り
Efficiencyは全てのch
で"1"である

まとめ

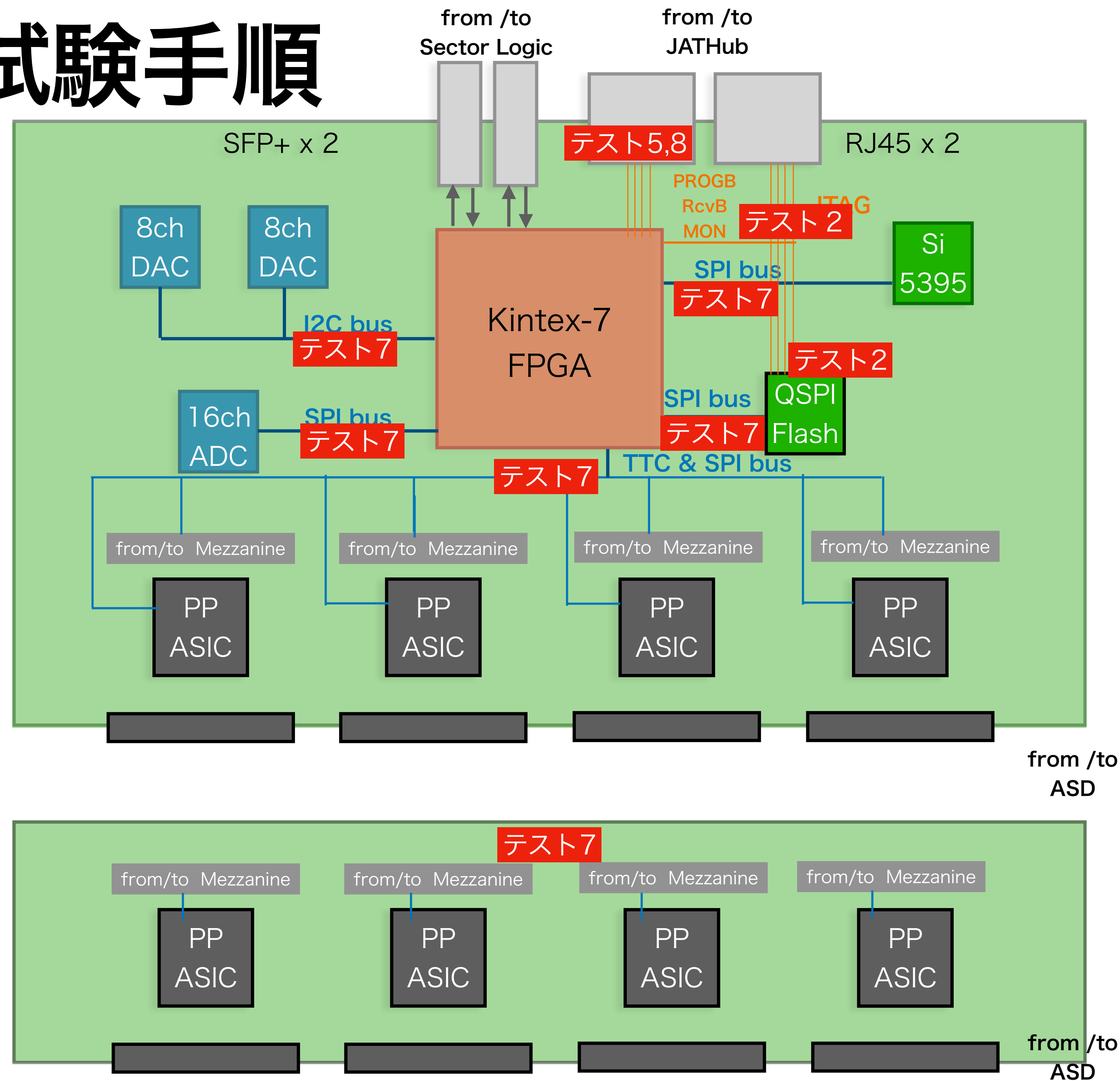
- PS board QA/QC試験に向けてZynq SoCと光通信を用いたコンパクトなDAQシステムを開発した
 - データ読み出し：高速光通信を用いたデータの受信、固定レイテンシーでのCPUからのリードアウト
 - 電気回路制御：PS board各素子をZynq上のCPU起点で制御
 - タイミング信号の分配：高速光通信によりTTC信号を分配
- FPGAテストパルスを用いた機能検証が完了
- ASDを用いたフルチェーンのテストパルス試験を予定している
- 2024年から始まるPS board QA/QC試験で実際に使用される
- 物理実験への汎用的な利用が期待できる

Back up

PS board QA/QC試験手順

試験手順案

1. 電源投入時の電流値のチェック
2. FPGAとQSPIにJTAG経由でファームウェアを焼く
3. QSPIのユーザ空間にパラメータを焼く
4. Si5395のNVMにパラメータ書き込み
5. PS boardにRecovery requestを出させ、PS boardをreboot(QSPIからファームウェア、パラメータ分配)
6. 電源まわりのモニタリング(xADC,Power good)
7. 各素子のconfigurationを読み出す。
 1. PP ASIC、DAC、ADCに関してはQSPIに書いたパラメータとコンシステントか
 2. Si5395 NVMに書いた値とコンシステントか
 3. PP ASICのRSPIb, Rsetb, PLLD, SEUの検証
8. Clock位相測定
9. ASDテストパルス試験
delay curveを書く必要あるか? →検討中
極性を反転させて2度やる。(PPASIC、DACの再設定)
10. ジャンパーピン、DIPswitchの設定
11. アルミケースに入れて出荷準備完了



JATHub - PS board間のデータフォーマット

• Phase2-SL と PS boardのフォーマットを採用

- Downlink : JATHub からPS boardを操作するためのデータ
- Uplink : PS boardからJATHubへヒットデータの送信

Downlink															
Words (32-bit)	Forth byte [31:24]					Third byte [23:16]				Second byte [15:8]			First byte [7:0]		
Word-0	0x0	Soft Reset	TPT	BCR	1'b0	4'b0	BCID [11:0]			Comma = 0xbc					
Word-1	6'b0			TX mode	4'b0	DAC Bitmap [1:0]	SCL	SDA	1'b0	CS Bitmap [10:0]	SPIRST	SIRSTB	SCLK	SDI	
Word-2	Address								Command						
Word-3	8'b0					FPGA TP pattern ID				Data					
Word-4	Footer = 0x0e0d								SL Full board ID						
Uplink 0															
Words (32-bit)	Forth byte [31:24]					Third byte [23:16]				Second byte [15:8]			First byte [7:0]		
Word-0	Monitoring data and control signal								Comma = 0xbc						
Word-1									32 bit hit data						
Word-2									32 bit hit data						
Word-3									32 bit hit data						
Word-4									32 bit hit data						
Uplink 1															
Words (32-bit)	Forth byte [31:24]					Third byte [23:16]				Second byte [15:8]			First byte [7:0]		
Word-0	Monitoring data and control signal								Comma = 0xbc						
Word-1									32 bit hit data						
Word-2									32 bit hit data						
Word-3									32 bit hit data						
Word-4									32 bit hit data						