



ICEPP
The University of Tokyo



高輝度LHC-ATLAS実験に向けた TGC検出器フロントエンドにおける エレクトロニクス制御装置の開発

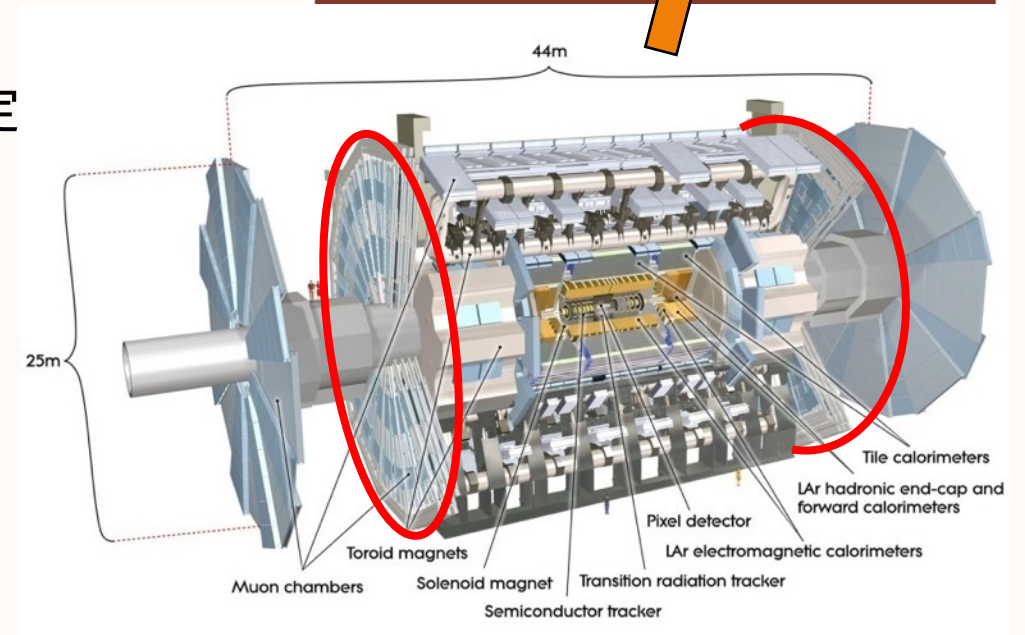
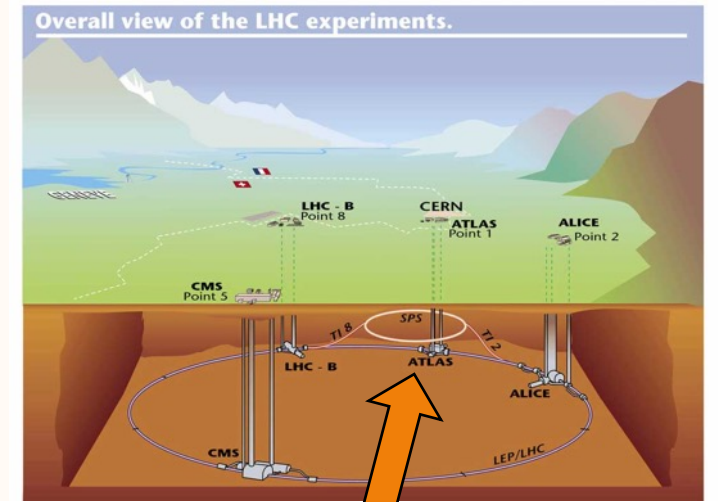
東京大学 ICEPP
石野研究室 田中碧人

高輝度LHC と TGC検出器

- LHC加速器
CERNの陽子陽子衝突型円形加速器
40MHzの陽子同士の衝突
- LHC加速器のアップグレード
瞬間最高ルミノシティを $7.5 \times 10^{34} \text{cm}^{-2} \text{s}^{-1}$ (約3倍)
に向上させる“高輝度LHC”が2027年から運転開始予定

- ATLAS検出器のアップグレード
高輝度LHCの運転開始に合わせて
ATLAS検出器の大部分を一新する予定

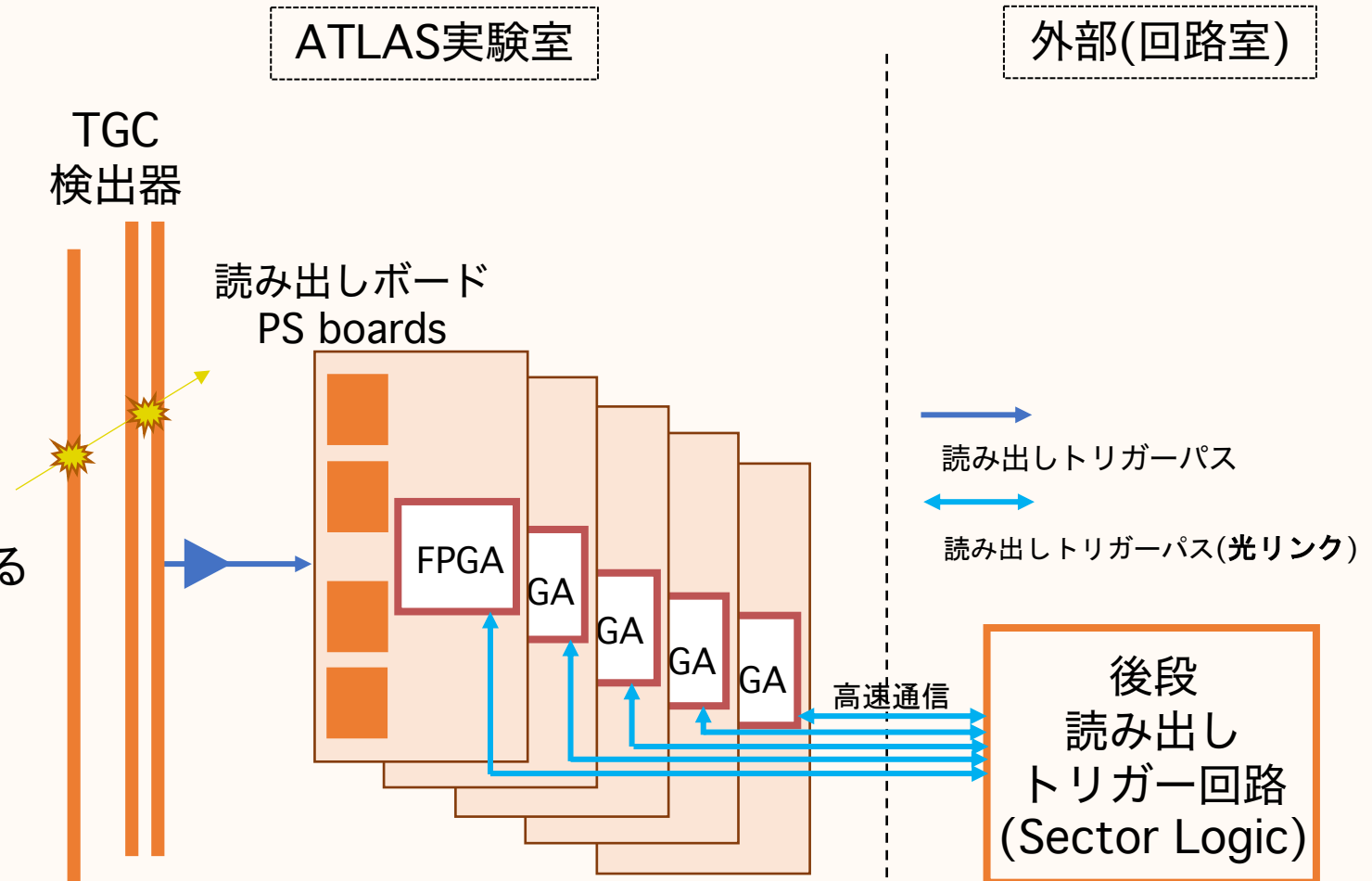
Thin Gap Chamber (TGC) 検出器の
読み出し・トリガーエレクトロニクスも一新



ATLAS検出器と
TGC検出器(ミュオン検出器)

高輝度LHCでのTGC検出器フロントエンドエレクトロニクスの特徴

- ・ PS boardから全hit信号を後段のトリガー回路へ光リンクで送る
- ・ トリガー処理などは後段が一手に引き受ける
- ・ 後段と高速通信を行うため、PS boardではFPGAを搭載する



高輝度LHC運転時の状況

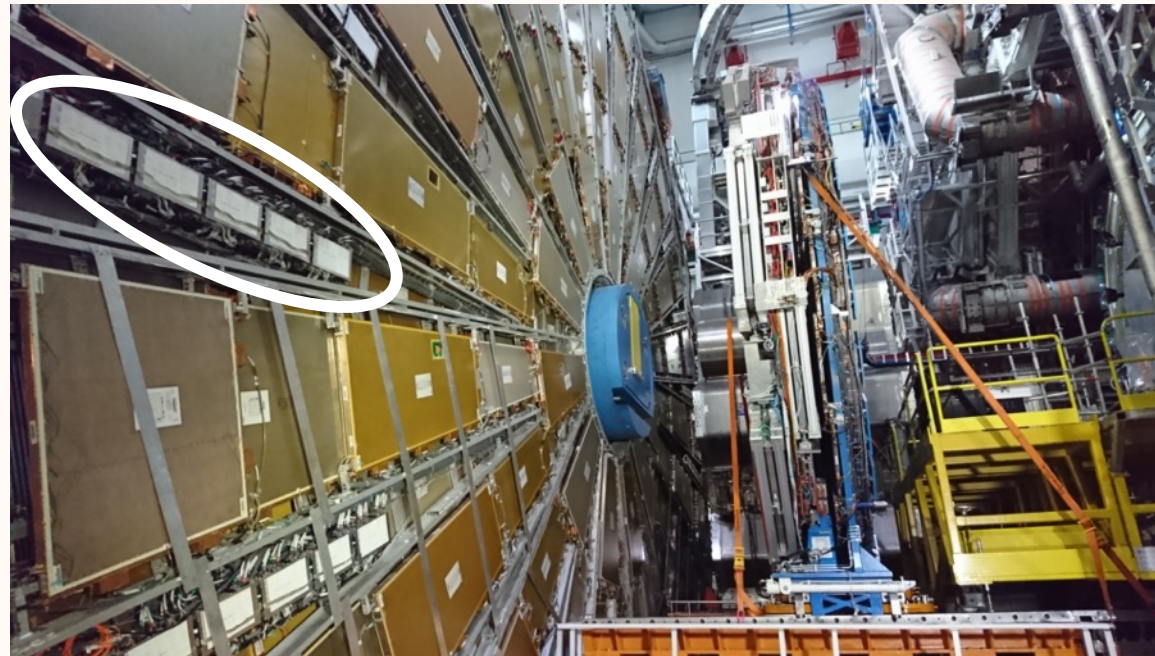
ATLAS実験室

外部(回路室)

60m 以上

TGC
検出器

PS board



後段回路(Sector Logic)

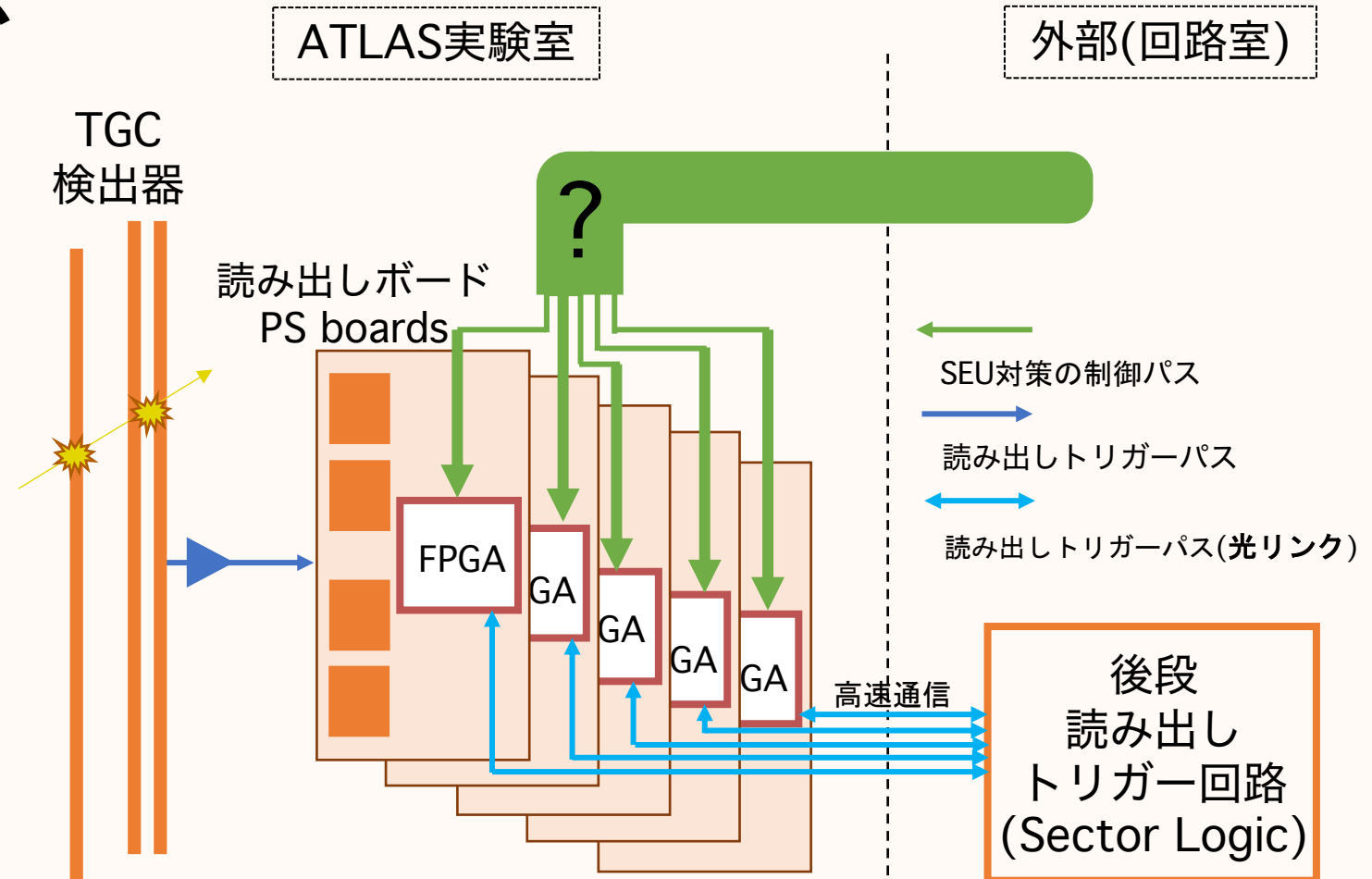


ATLAS実験室は立ち入り禁止

回路室は立ち入り可能

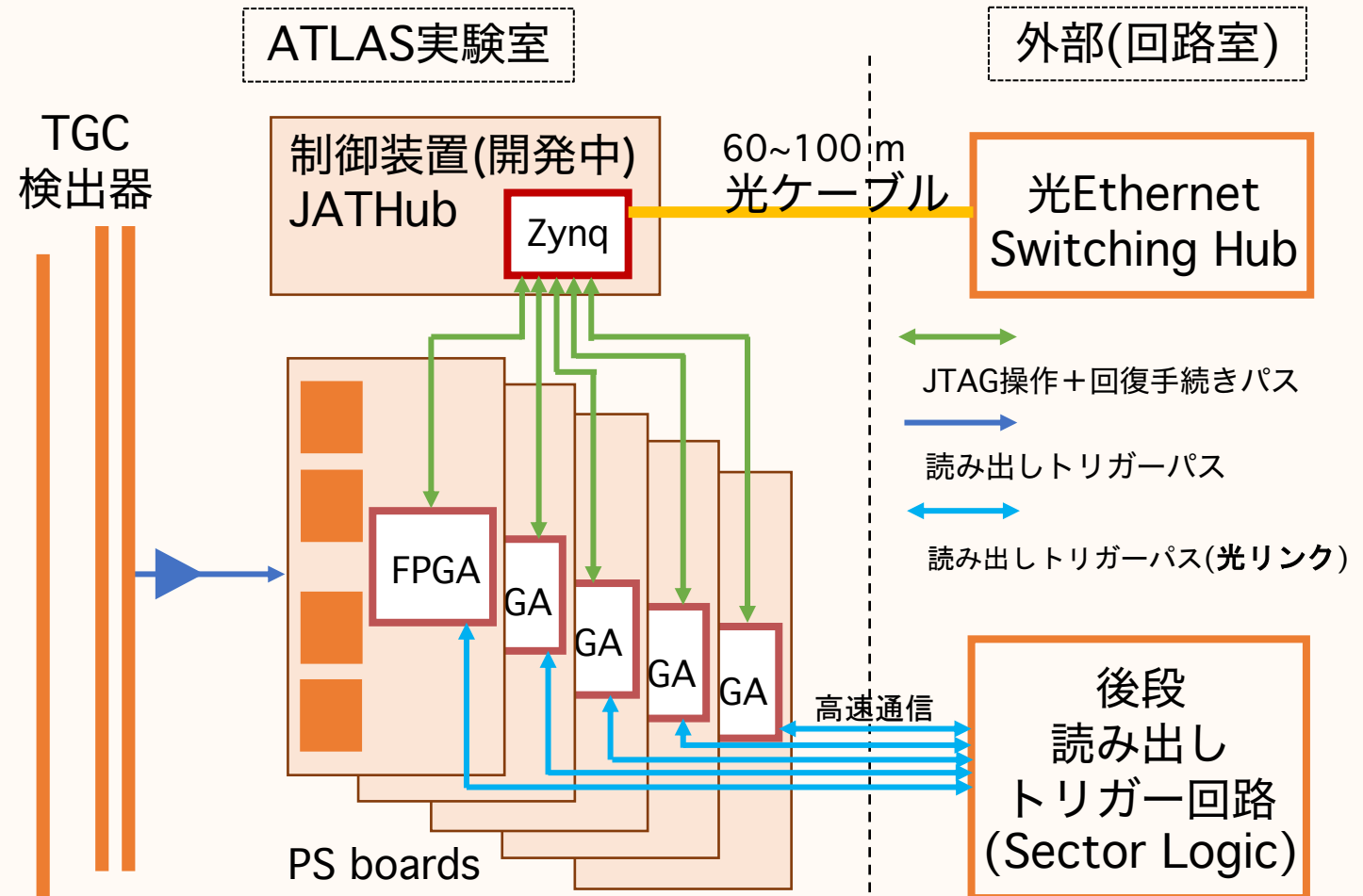
高輝度LHCでのTGCC検出器フロントエンドエレクトロニクスの問題点

- ・ FPGAは放射線環境下で使用されるため、FPGAのconfiguration memoryにてSingle Event Upset (SEU) が起き、Firmwareの異常が発生
- ・ 光リンクが切断されることがある
- ・ SEUや光リンクを監視し、FPGAの修復+光リンクの立て直しを行える制御系が必要



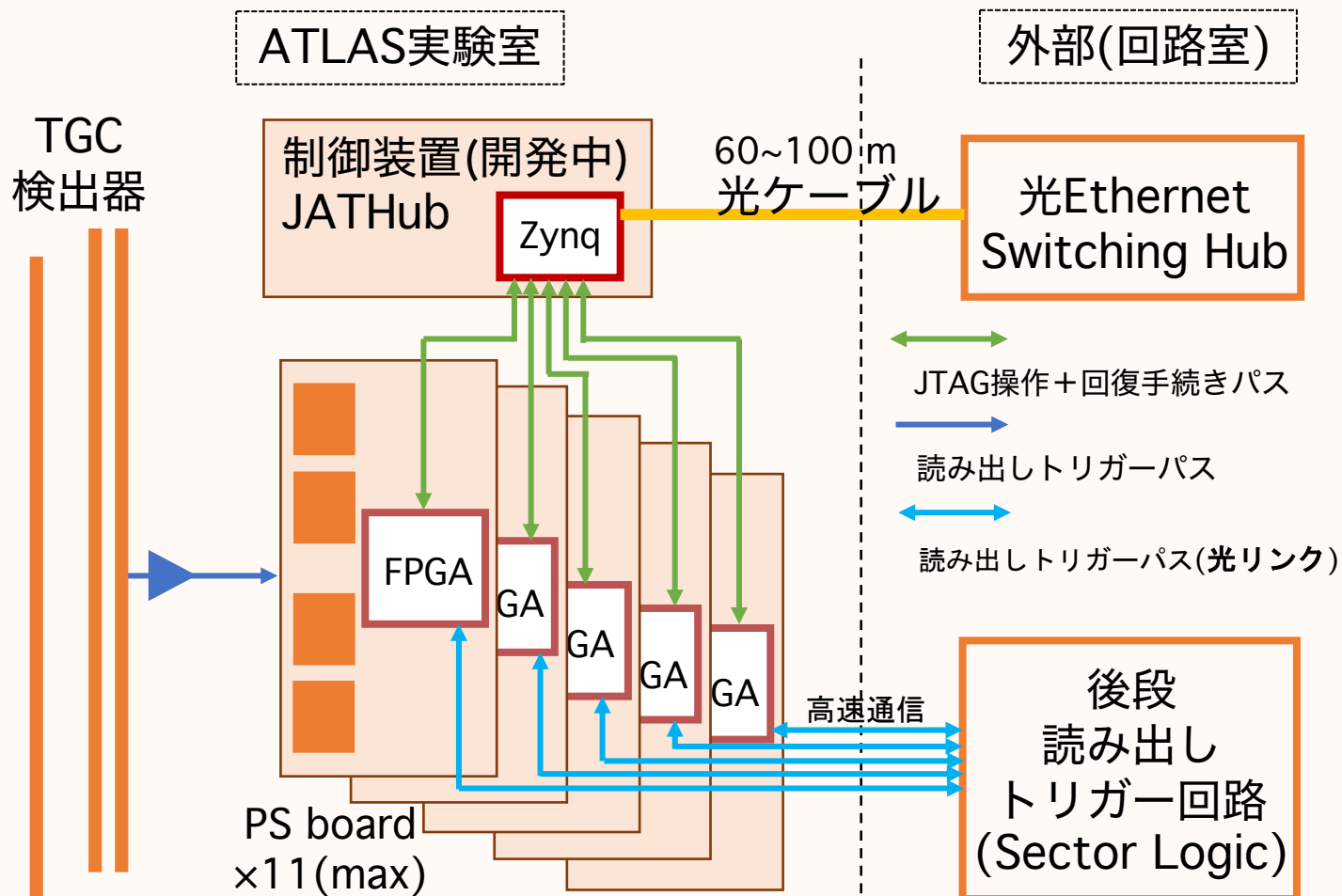
制御装置 “JATHub” の導入と開発

- ・ 複数のPS boardを監視、管理するハブモジュールを設計
- ・ 制御装置 (VMEクレー트에接続する回路) “JTAG Assistance Hub” (JATHub) を開発中
- ・ CPUとFPGAを組み合わせ高い汎用性を持つXilinx社製 Zynq SoC (System on Chip) デバイス
- ・ FPGAの修復+光リンク立て直しを行うためZynqをメインドライバーとして周辺機器を自由に操作する設計



JATHubによるエレクトロニクス制御の仕組み

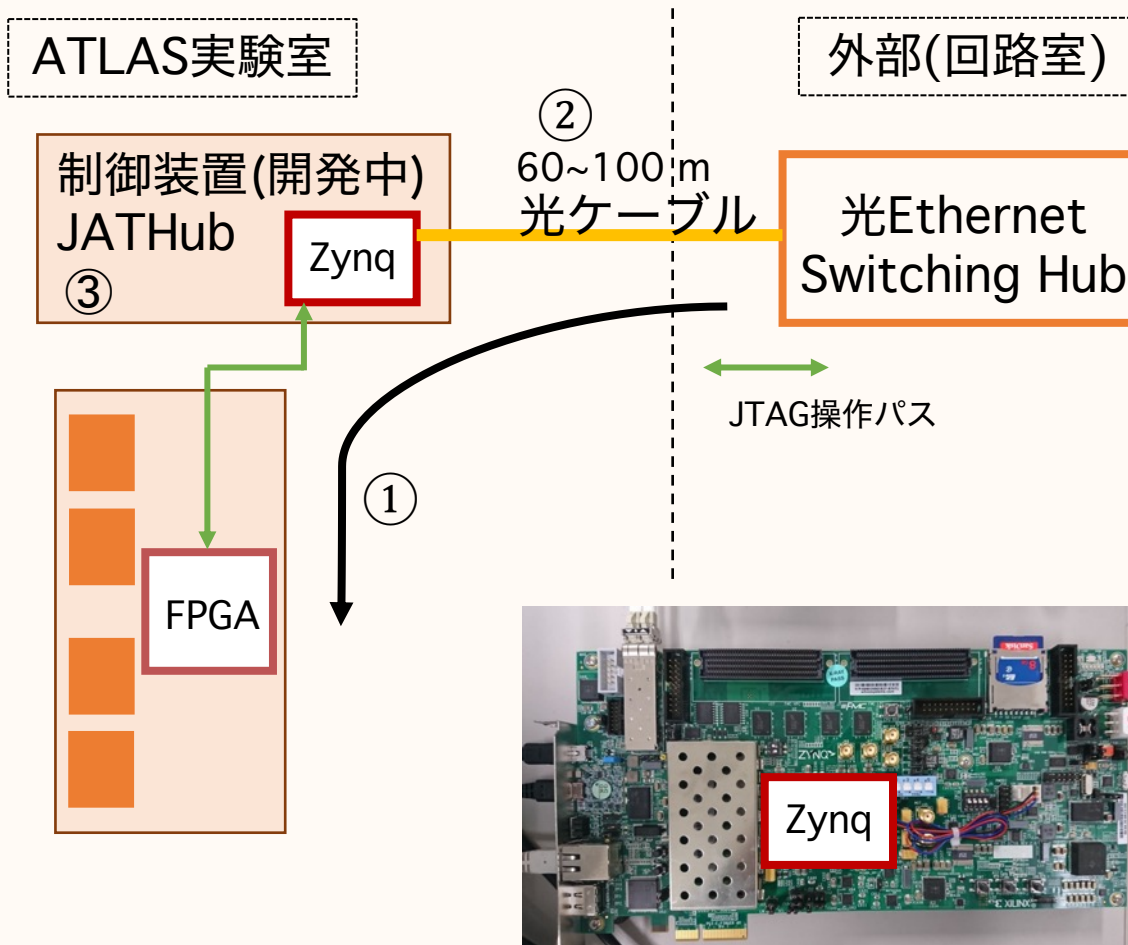
- ・ Zynqを介して外部からPS boardをJTAG操作
1/12 Sectorにつき
58枚のPS boardを6枚のJATHubが担当
- ・ 外部とZynqが光イーサネット通信
- ・ SEUによる救難信号を送ってきたPS boardへ
Zynqが自動で回復信号を返信
- ・ JATHubのSEU対策として、
同様に、近隣のJATHubと相互で回復手続き
- ・ JATHubのBootの冗長性を保つために
複数のBootファイルをZynqが自由に選択



評価ボードによる機能のデモンストレーション

JATHubに実装する予定の以下の3つの機能をZC706評価ボードでデモンストレーションする

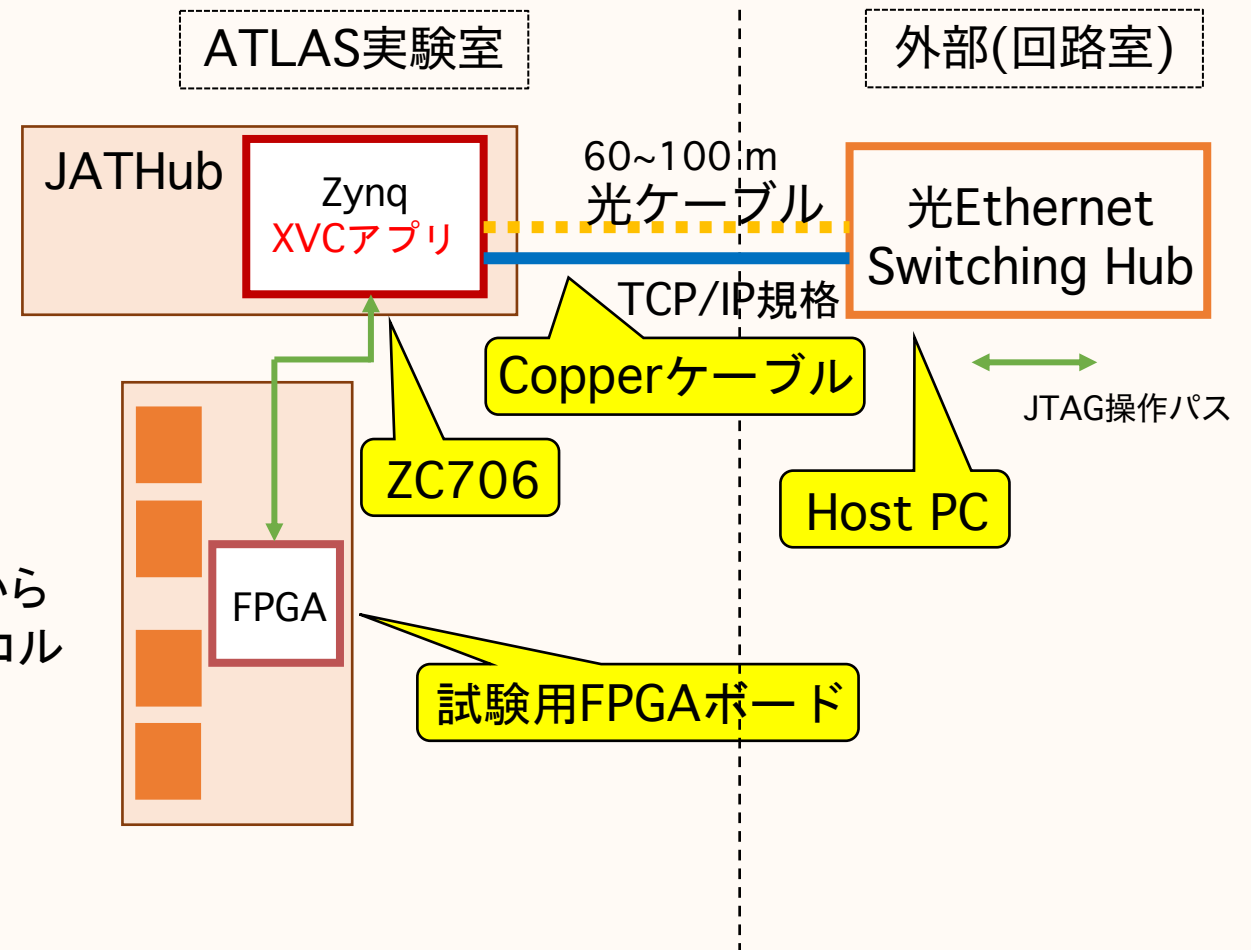
- ・ 試験①: XVC
Zynqを介してネットワーク経由で外部からFPGAをJTAG操作できるようにする
- ・ 試験②: 光Ethernet
Copperケーブルでなく、光ケーブルによりSFP経由でZynqが光Ethernet通信できるようにする
- ・ 試験③: 放射線損傷に対する冗長性のあるBoot



同じZynq SoCデバイス搭載
ZC706評価ボード

[試験1] Xilinx Virtual Cable (XVC) の特徴

- ・ 本来、FPGAを操作するにはVivadoの入っているPCから直接JTAGケーブルで繋ぐ
- ・ XVCとは、TCP/IP規格でZynq SoCにアクセスし、Zynqから別のFPGAにJTAG規格でアクセスするプロトコル
- ・ Zynqとネットワークで繋がれば、FPGAを遠距離から操作可能
- ・ 東大ICEPPの杉崎と共同で、デモを行い成功した



XVCでFPGAの操作成功

- 一つのZynqボード(ZC706)に対して一つのFPGAボードを操作したが、今後、JATHubが複数のFPGAを操作できるXVCアプリを開発する予定

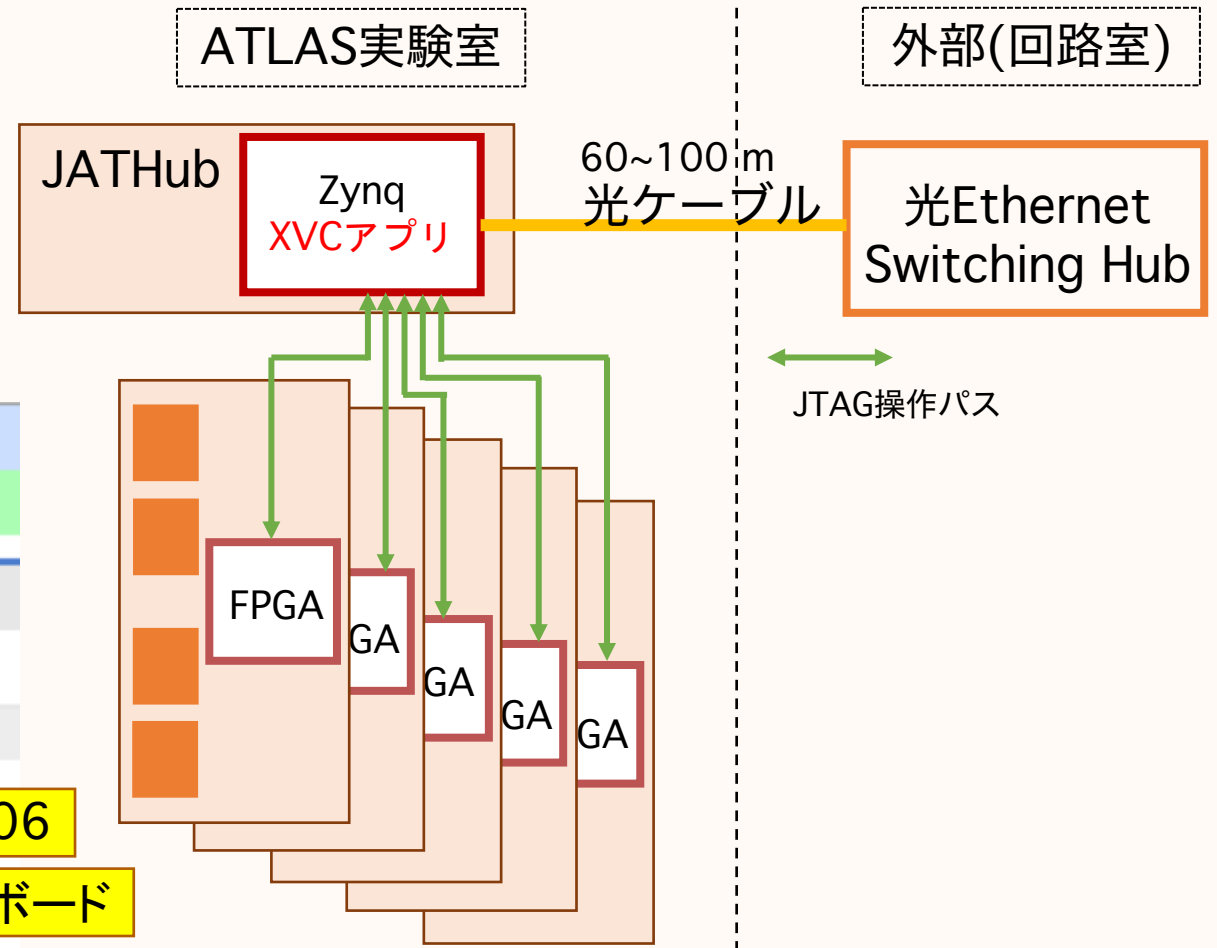
HARDWARE MANAGER - localhost/xilinx_tcf/Xilinx/133.11.25.196:2542

There are no debug cores. Program device Refresh device

Hardware

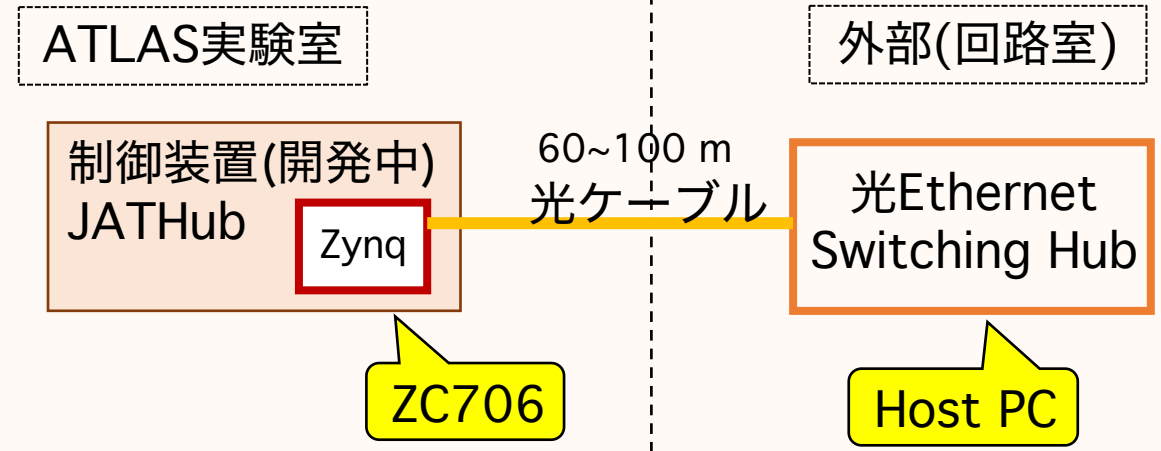
Name	Status
localhost (1)	Connected
xilinx_tcf/Xilinx/133.11.25.196:2542 (1)	Open
xc7k325t_0 (1)	Programmed

XADC (System Monitor)



[試験2] 光Ethernet の特徴

- ・ 光ケーブルは長距離伝送が可能
- ・ Zynqが、RJ45とCopperケーブルではなく、SFPと光ケーブルを使用してEthernet通信を行うこと



Zynq

I²C操作

ジッタークリーナー

発振器

CPU

MAC
Media Access Controller

GMII規格

GMII規格
でMACと会話ができる

FPGA

GMII変換

PHYに相当

PHY Chip

125MHz

1000Base-X規格

GTX

SFP
connector

SFP

RJ45

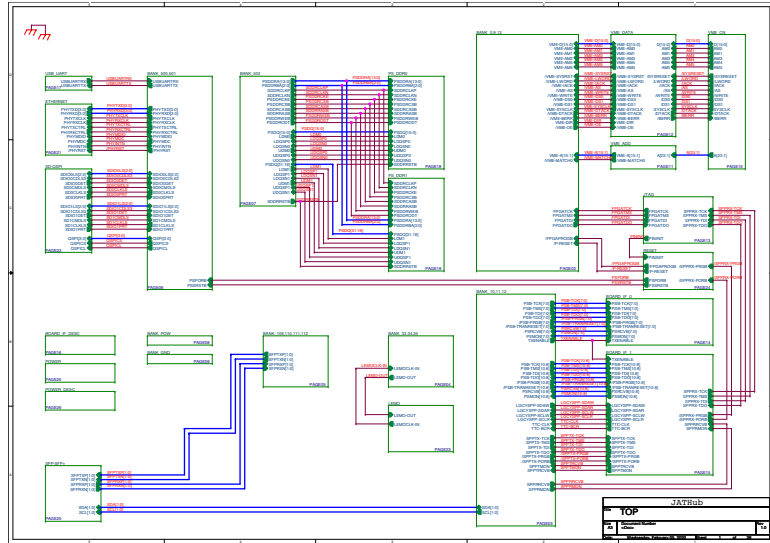
光ケーブル

(1000base-X規格)

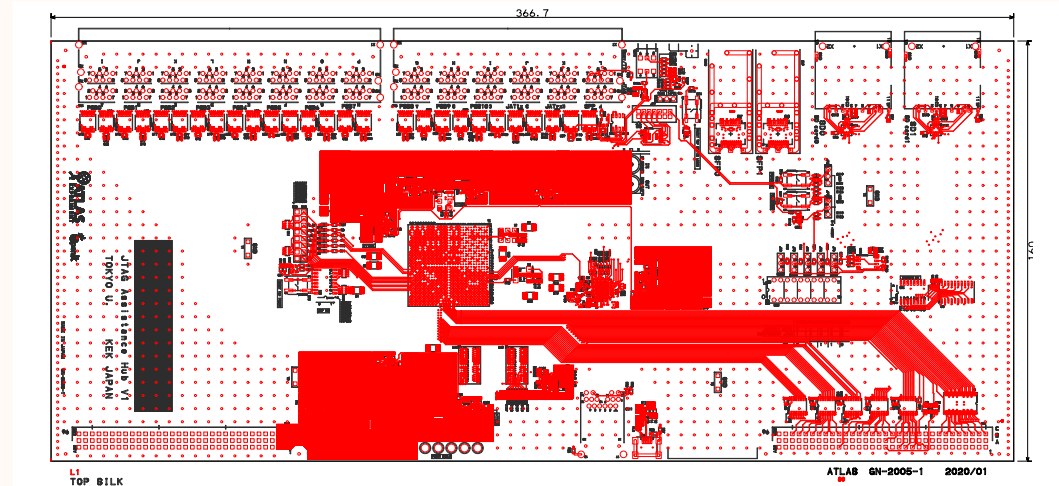
Copperケーブル

(100/1000base-T規格)

JATHubの制作状況



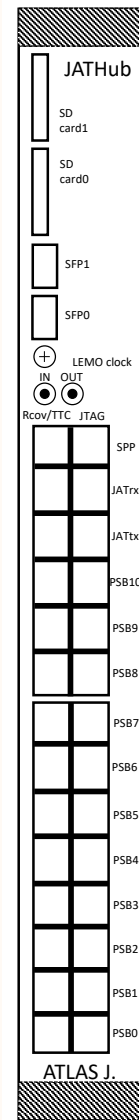
JATHub試作機回路図



JATHub試作機レイアウト配線図面(almost done)

<予定>

- ・ 実機の製造
- ・ JATHub試作機が今年の春までに完成の予定



フロントパネルのデザイン

まとめと展望

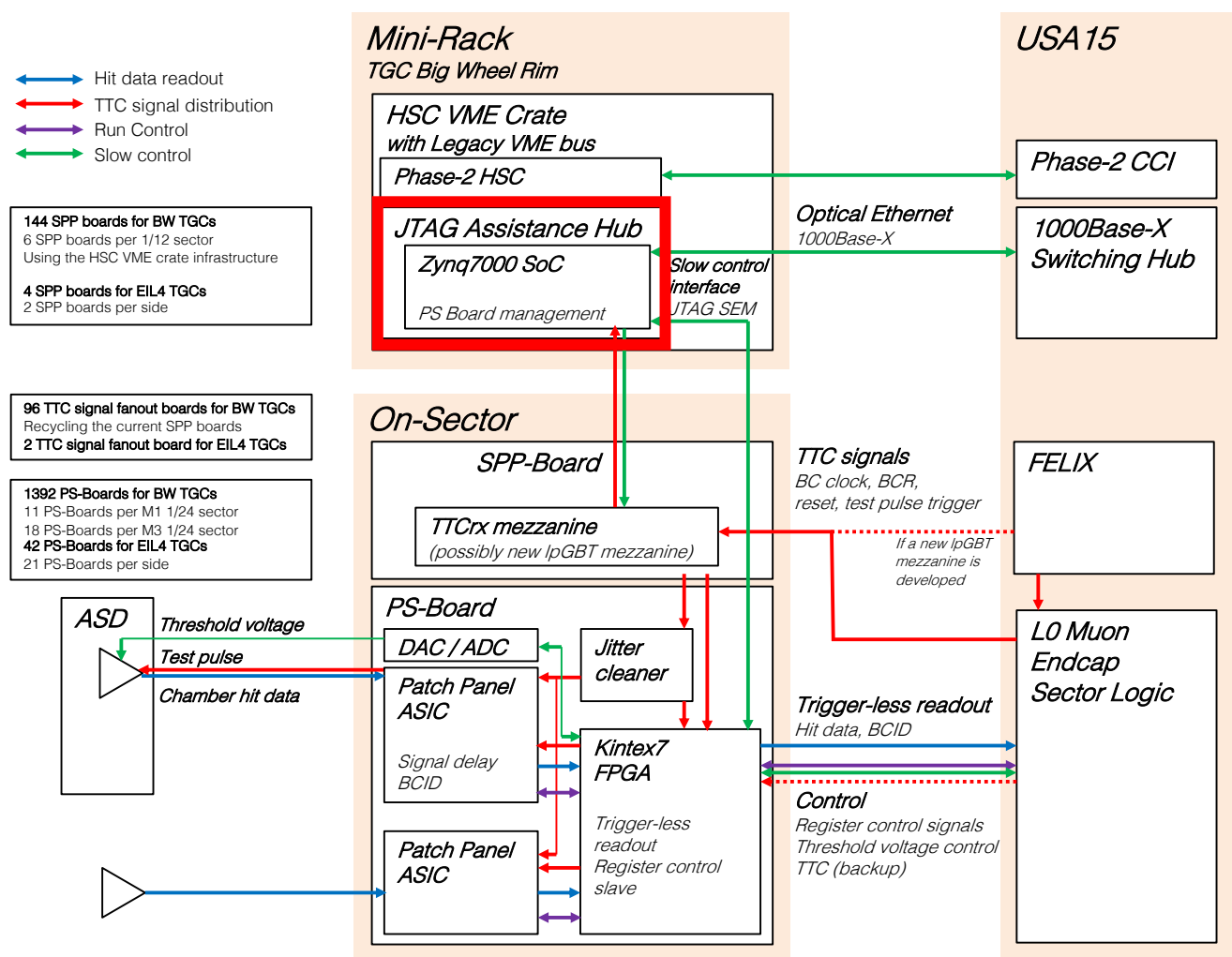
- ・ PS boardのFPGAのSEU対策に、複数のFPGAへのバイパス制御系の中心として機能するJATHubモジュールを設計
- ・ 制御装置JATHubの各機能のデモンストレーションは着々と進行中
- ・ JATHub試作機の実機は今年の春までに完成

<今後の展望>

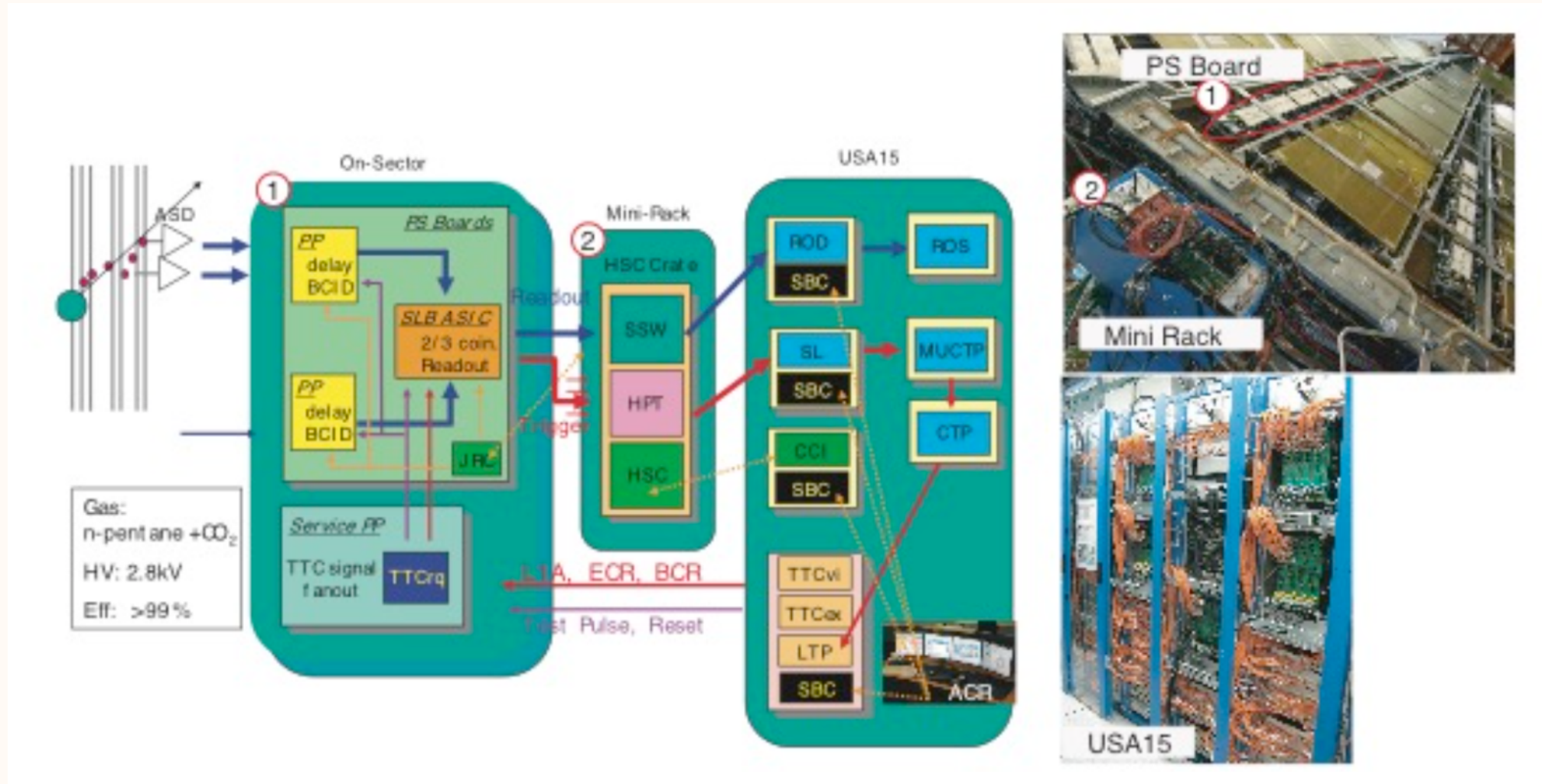
- ・ XVCの機構で複数のFPGAを操作できるかZC706でデモンストレーション
- ・ XVCのネットワークを光Ethernet通信で行えるかZC706でデモンストレーション
- ・ 冗長性のあるBootをZC706でデモンストレーション
- ・ JATHub試作機の動作試験項目を検討し、実機完成に備える
- ・ Test Benchを作り、実機とPS boardとのシステム機能テストを行う

Backup

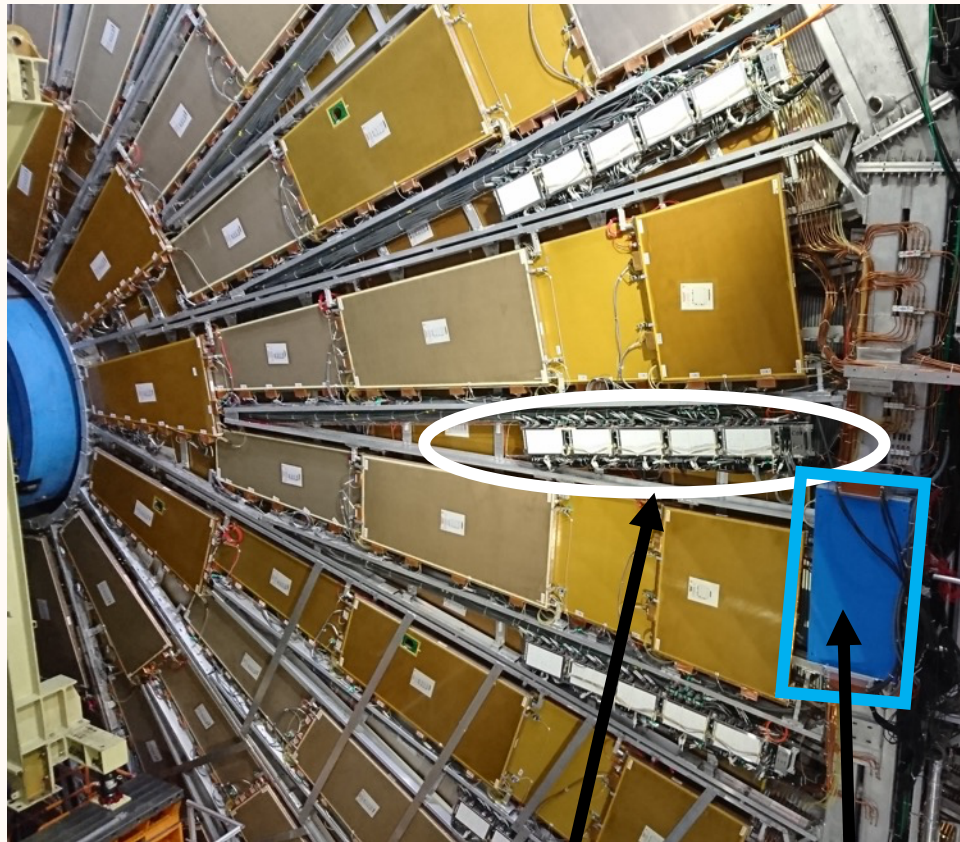
JATHub周辺のパス詳細



現行のTGCフロントエンドのパス



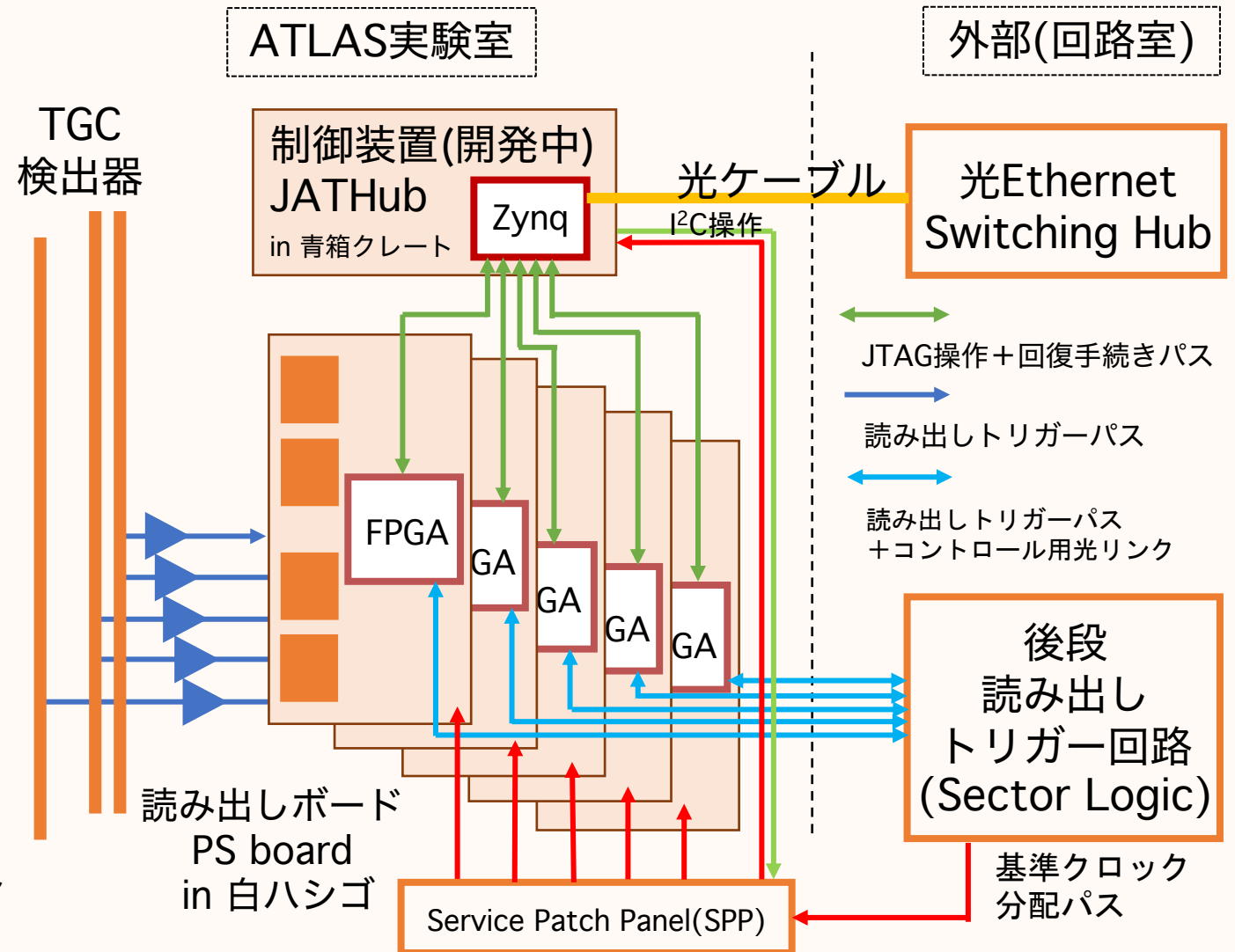
JATHubとPS boardの位置関係



TGC検出器

白ハシゴ

青箱クレート



Zynq とは

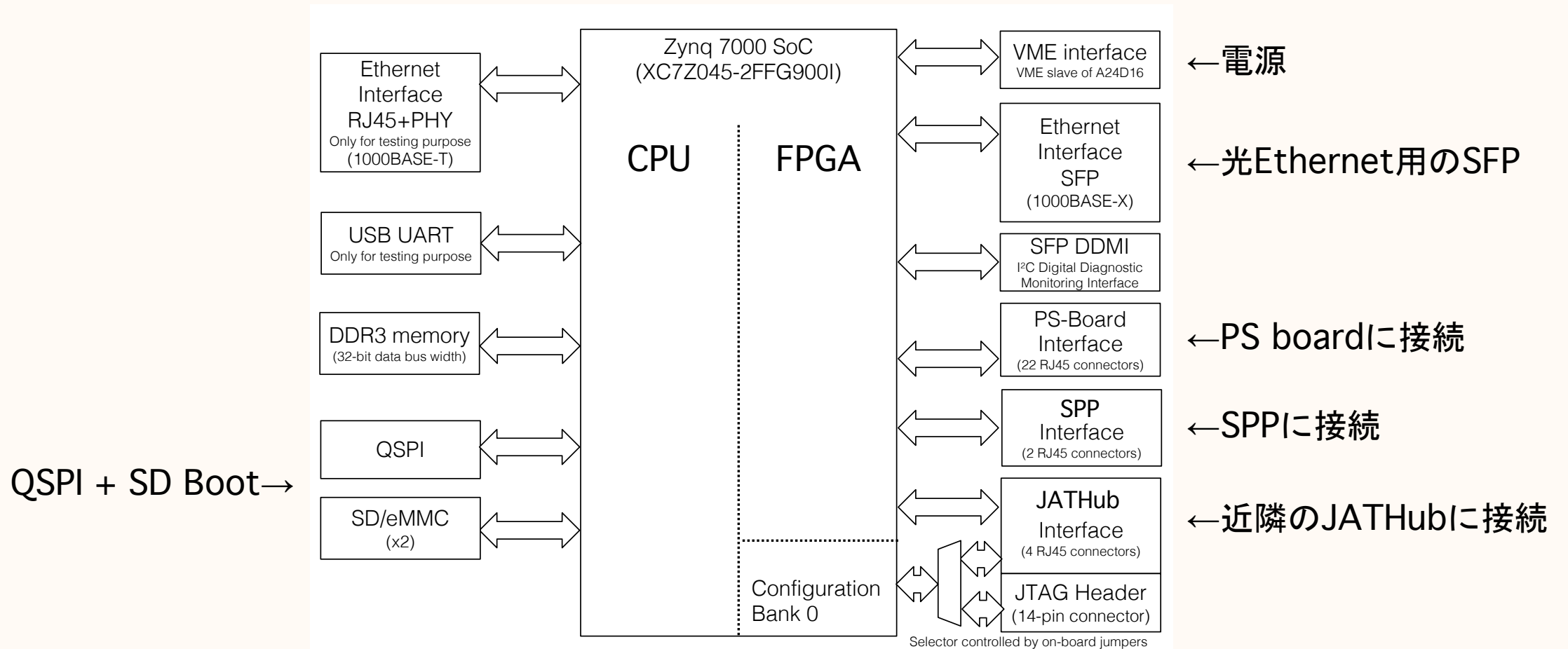
- ・ Zynq : CPU + FPGAを組み合わせたXilinx社製のSoC (System on Chip)デバイス
- ・ CPU : ネットワークによるモニタリングや制御、Softwareによるデータ処理
- ・ FPGA : Firmwareによる高速なデータ処理、汎用I/Oによる柔軟な周辺機器との接続

→ ZynqはCPUとFPGAの長所を兼ね備え、高い性能と柔軟性と拡張性をもつデバイス

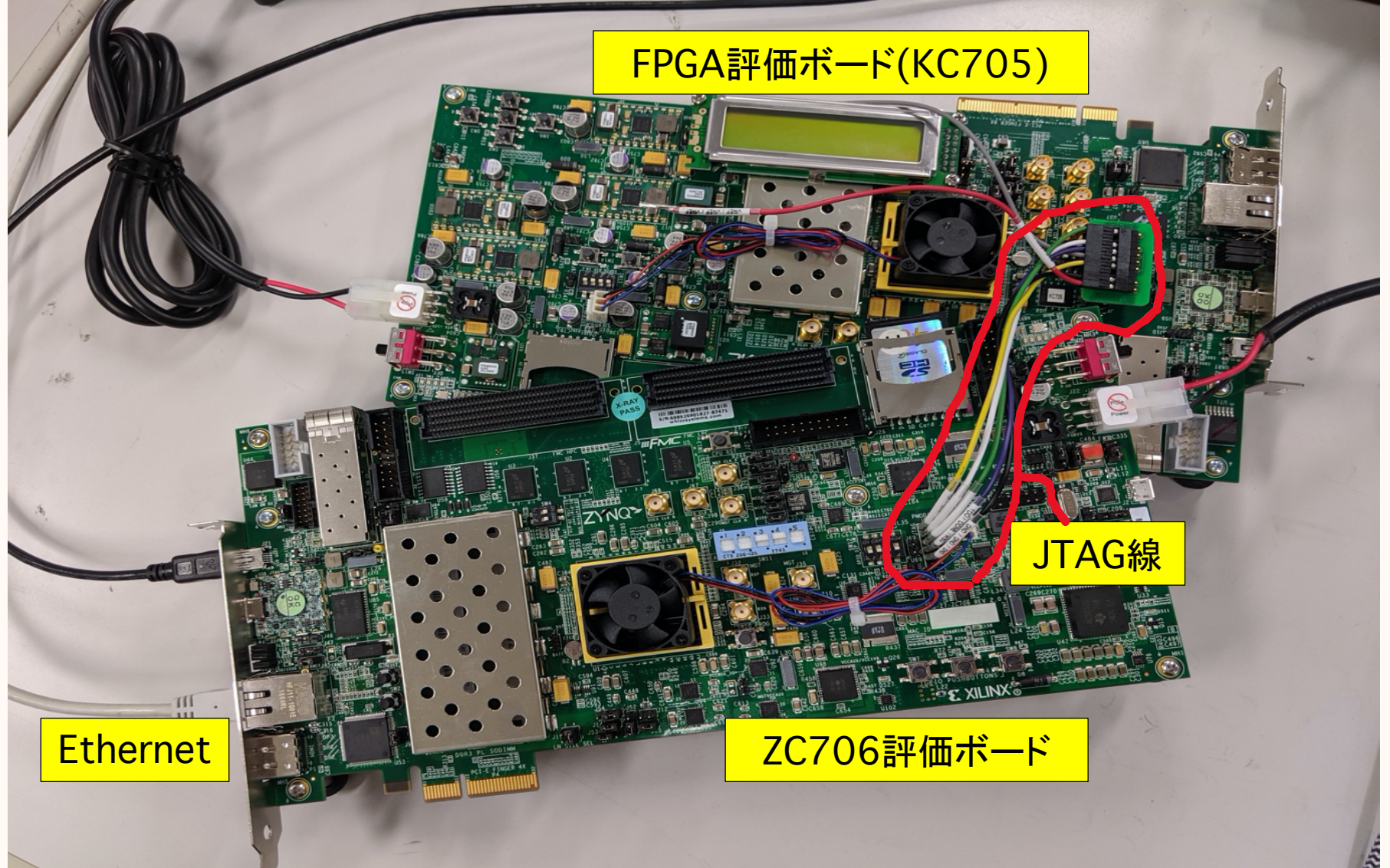
フロントエンドにおけるデータ処理とモニタリングを行うのに適しており
放射線耐性も持つので、高エネルギー実験での実用性がとても高い

- ・ Zynqに関するkeywordとして、自動運転 (Advanced Driver-Assistance Systems (ADAS))
やIoT、Edge Computingが挙げられる

JATHubのZynq周りの設計

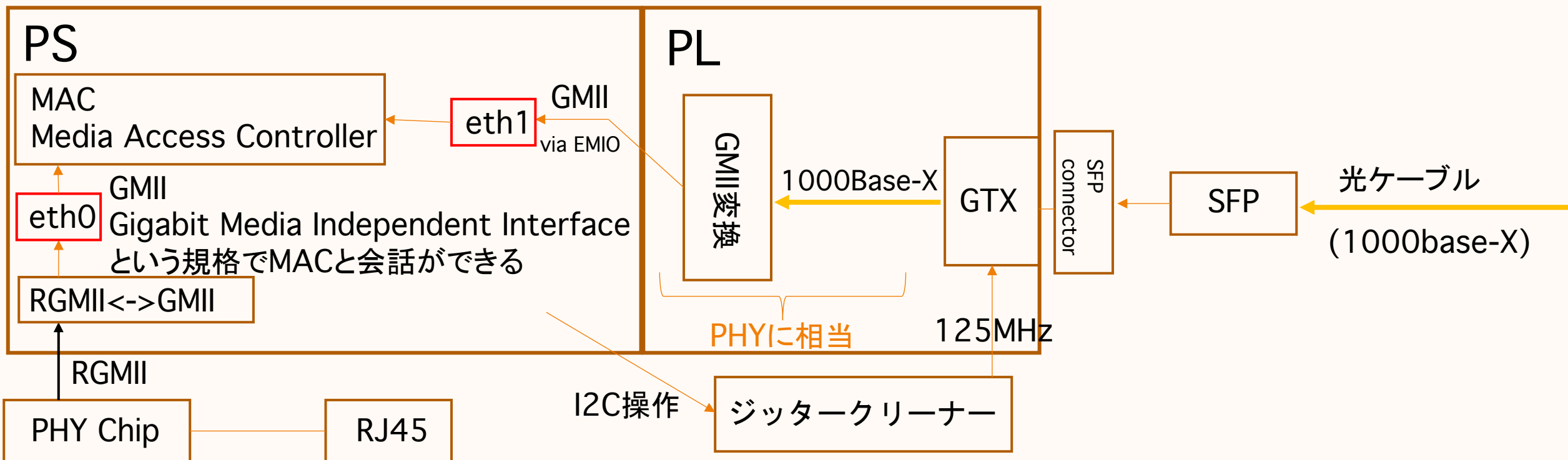


XVC配線



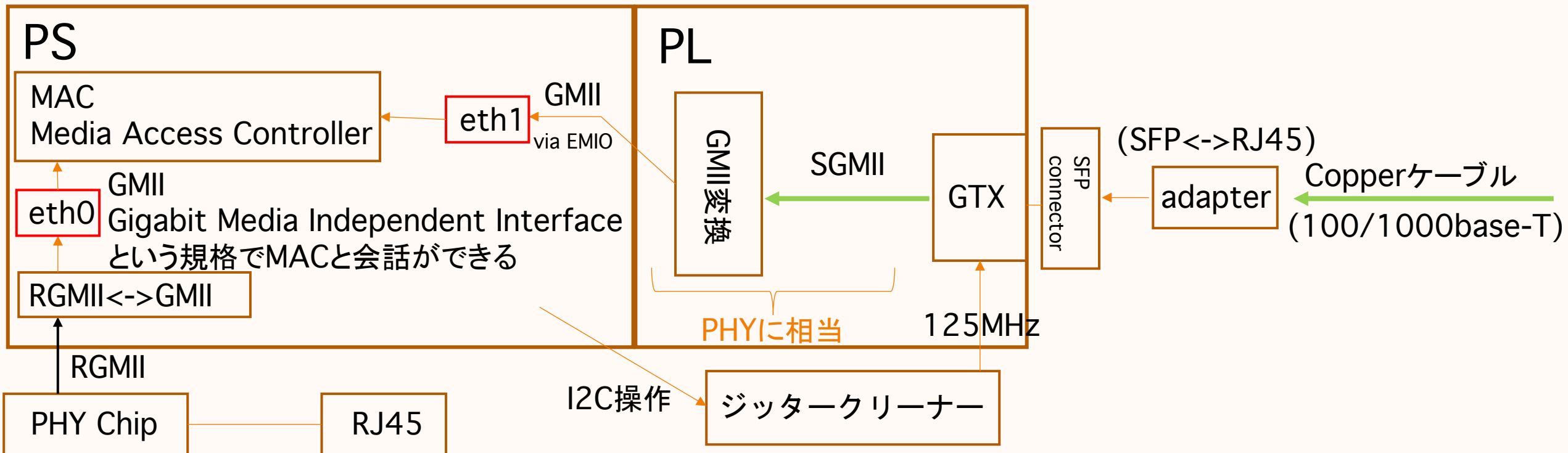
Opt-Ethernetを自作Bootファイルで開通

- Optical Cableによる1000BASE-X規格のOpt-Ethernet on ZC706 Evaluation Board



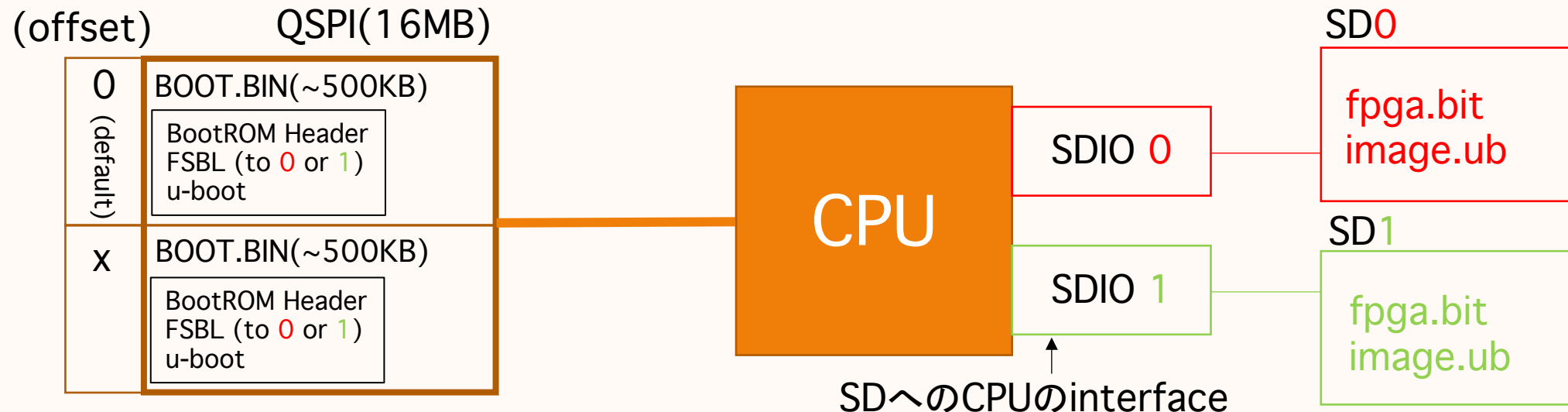
Opt-Ethernetを自作Bootファイルで開通

- Copper CableによるSGMII規格のOpt-Ethernet on ZC706 Evaluation Board



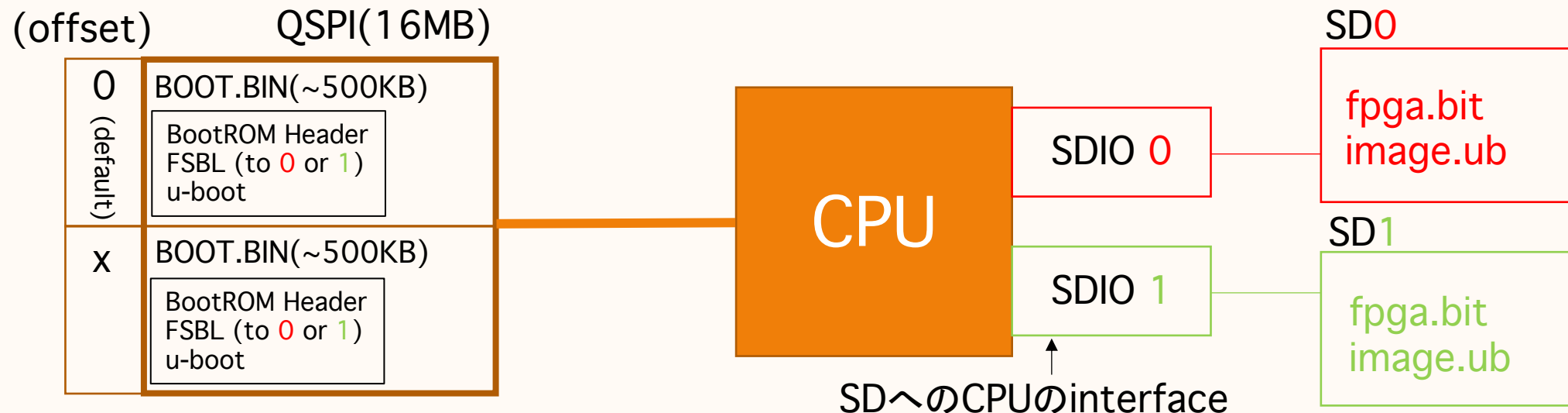
[課題3] QSPI + SD Boot の仕組み

- ・ ZynqのBootイメージは
 - *Initial Boot package (BOOT.BIN) : BootROM Header, First, Stage Boot Loader (FSBL), Second Stage Boot Loader (u-boot)
 - *Other Boot Images : FPGA Firmware image (fpga.bit), Linux Kernel image (image.ub)で構成されている
- ・ QSPIに複数のBOOT.BIN(それぞれoffset値付き)を入れ、それぞれのSDにfpga.bit, image.ubを入れる
- ・ 電源が入るとCPUがHeader(default:offset=0)を探し、同封されたFSBLを読み始める
- ・ FSBLはSDIO 0 or 1 を指定しているので、CPUは指定されたSD内のimage.ubを読む
- ・ 最後にCPUはu-bootと fpga.bitを読む → 予備のBootイメージを読み込める仕組みとなる

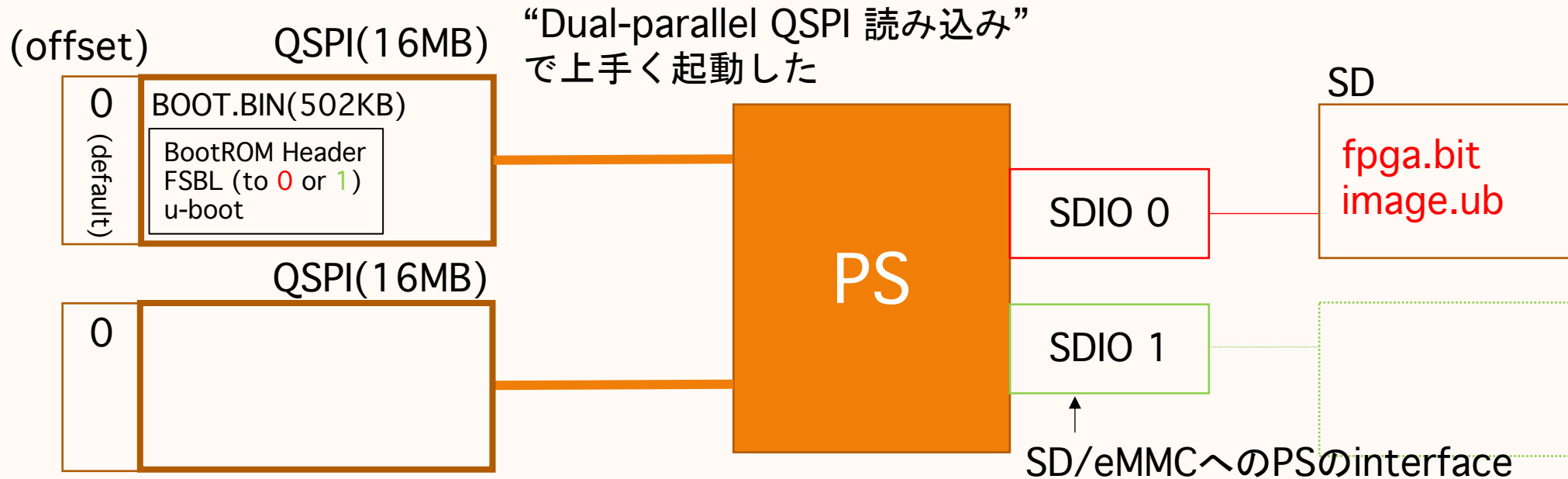


QSPI + SD Bootによる冗長性

- ・ SDIO先のデバイス(SD)もしくはその中身に問題が発生した場合 (ZC706にSD1が無いのでデモ不可)
 - QSPI内にあるFSBLのSDIO選択の設定を編集し、QSPIにFSBLを上書き
 - すると、CPUは、正常に機能するもう一つのSDIO先のデバイスで予備のイメージを読み込む
- ・ QSPIもしくはその中身(BOOT.BIN)に問題が発生した場合 (ZC706でデモ可能な機能)
 - CPUが自動的に“BootROM Header検索”をかける
 - offset≠0 のBootROM Headerにある予備のFSBLを自動で読み込み始める



QSPI+SD Boot on ZC706で実際にできた事



<わかったこと>

- ・ QSPIに焼くBOOT.BINに、少なくともFSBLとu-bootをパッケージすれば正常に起動
- ・ image.ubは今まで通り正常に読み込まれたが、fpgaのビットファイル in SDは読み込まれず、u-boot内の初期fpgaビットファイルが読み込まれた
- ・ 起動後にSDカード内のfpgaファイルはLinuxのコマンドで読み込めた
→Opt-EthernetのPHY領域を認識した