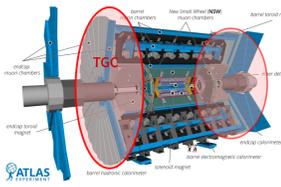


HL-LHC ATLAS TGC 前段回路の QAQC におけるデータ管理システムの開発

ICEPP, U. Tokyo, M1 大坪 航
ICEPP シンポジウム, 2025-02-18 火

1 TGC エレクトロニクス

- HL-LHC: 陽子陽子衝突型加速器 LHC の高輝度化 (- 2041)
- TGC(Thin Gap Chamber): ATLAS 実験のエンドキャップ領域 ($1.05 < |\eta| < 2.4$)のトリガー用ミュオン検出器
- HL-LHC の TGC エレクトロニクスは 4 種類のボードで構成される
 - バックエンド(off-detector): Sector Logic(SL) (× 24 セクター × 2 サイド)
 - フロントエンド(on-detector): PS Board, JATHub, TAM

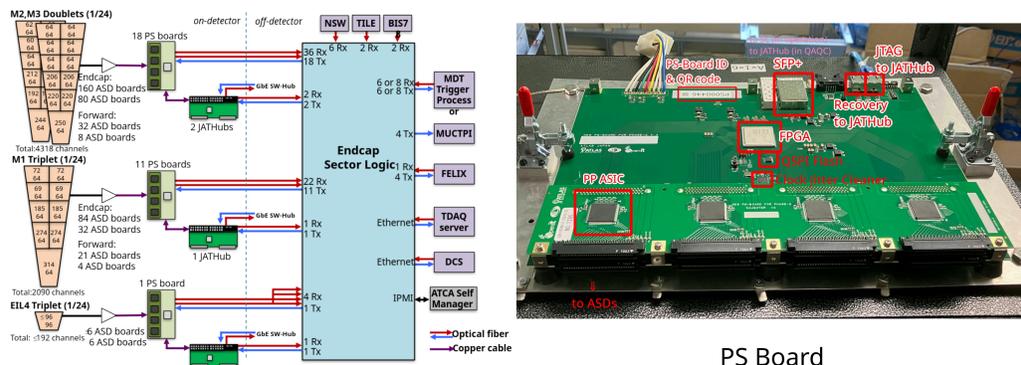


- ASD: 信号の増幅、デジタル化
- PS Board: 可変遅延による信号の同期とバンチ交差の同定
- JATHub: PS Board/TAM の制御、モニター
- Sector Logic: トリガーロジックと読出ロジックの実装、システム全体の監視・制御

1.A PS Board

主な部品とその機能

8 Patch Panel(PP) ASICs	<ul style="list-style-type: none"> 信号の同期(設定した可変遅延をかける) LHC クロックとの同期(バンチ同定)
FPGA (Kintex-7)	<ul style="list-style-type: none"> PP ASIC からのデータを後段回路(SL)に送信する ボードの制御
クロックジッタークリーナー	<ul style="list-style-type: none"> LHC クロックから低ジッターなクロックを生成し、PP ASIC や FPGA に供給する
2× SFP+ トランシーバー	<ul style="list-style-type: none"> 後段回路との通信(SL) データ/ボードの状態の送信、制御信号の受信
2× RJ45 端子	<ul style="list-style-type: none"> JATHub から JTAG 線によって FPGA と QSPI を制御する JATHub に繋がる Recovery/Monitor 線(LVDS) <ul style="list-style-type: none"> 放射線損傷時の救難信号の送受信と FPGA のリコンフィギュレーション クロック位相測定のための信号線
QSPI フラッシュメモリ	<ul style="list-style-type: none"> 不揮発メモリでの FPGA のファームウェアと PP ASIC のパラメータの保持
ADC, DAC	<ul style="list-style-type: none"> ASD の閾値電圧の供給と監視



PS Board

2 PS Board QAQC

- 今年度(7月-2月)量産を実施・完了(目標値 1480 枚を達成)
- 量産と並行して行われる QAQC(品質試験)において上記の機能が機能するかを試験する

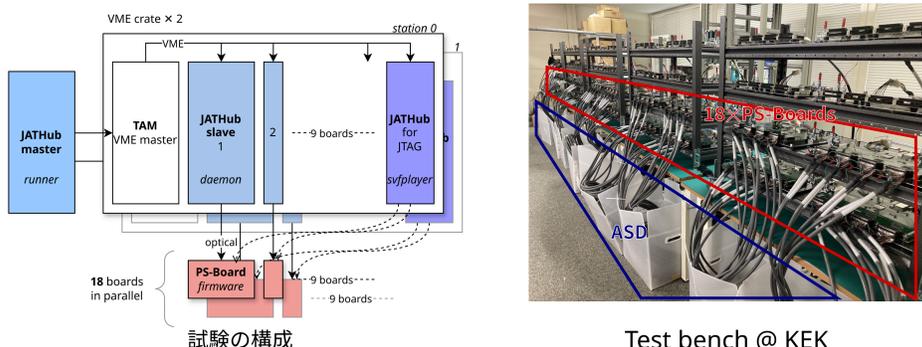


試験項目	試験で対象となる部品
1 ファームウェア書き込み	JTAG 線、QSPI フラッシュメモリ
2 パラメータ書き込みと分配	QSPI, PP ASIC, Clock Jitter Cleaner, ADC, DAC
3 リカバリー試験	RJ45 線(回復パス)
4 電圧モニタリング	DAC, ADC, xADC(FPGA 内部)
5 クロック位相測定	
6 ASD テストパルス試験	PP ASIC, FPGA, SFP+

- 2 種類の試験がある(1 回試験および 100 回試験)
- 問題のなかった PS Board はアルミケースへのパッキングのためにメーカーに引き渡す

2.A 試験ベンチ

- 十分な並列数を確保するための最適なシステムを設計
- JATHub を PS Board のモニター/コントロールのホストとして用いる
 - SL をエミュレート(どちらも Linux とテスト用アプリを実行できる MPSoC を載せている)
 - 1 台の JATHub は 1 台の PS Board を扱える(SFP のポート数の制約)
- 18 台の PS Board が並列で試験される
 - 18 台の JATHub を制御するために、1 台のマスター JATHub を導入した

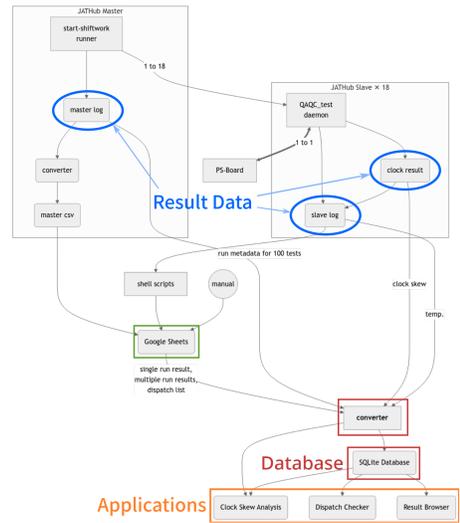


Test bench @ KEK

3 データ管理システム

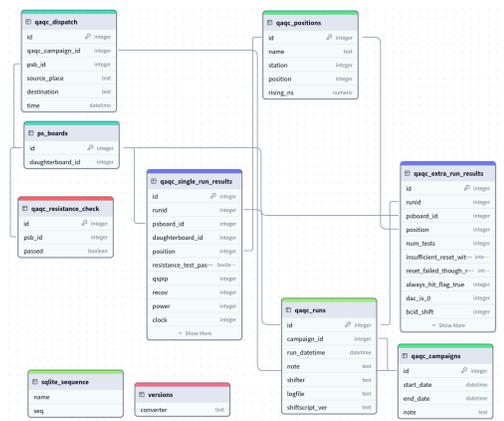
3.A 試験システムで得られるデータ

- 試験結果は JATHub のマスターおよびスレーブ JATHub の両方で得られる(右図)
 - マスターで動く runner とスレーブで動く daemon に分かれていることが原因
- これらのデータを管理するためのデータ管理システムを開発した
- 初期は Google スプレッドシート上での管理を行っていた
 - 手でデータの入力や検索を行う
 - ヒューマンエラーによるミスがあり、改善する必要性が指摘された
- リレーショナルデータベースを使用したシステムを開発
 - 複数のソースからのデータを統合し、単一のデータベースに整理する
 - より構造化され、他システムに組み込み可能になった



3.B データベースの構造

- フォーマット: SQLite (リレーショナルデータベース)
- テーブル + 制約
 - キー列は互いに重複できない
 - 列は他のテーブルのキー列を参照できる
- テーブルごとにデータを直交化できる



3.C 変換ソフトウェア

- 素早く開発するために Julia で実装
- GitLab で管理(CI や統合テストも整備)
 - 定期的なテストの自動実行により、意図しない Google スプレッドシートの書き換えを検知できた
- ログの解析を行うパーサーやキャッシュも提供

4 データベースの応用事例

4.A “出荷”確認

- データベースを参照し、ボードが試験をパスしたかを判定
 - ボードに貼り付けてある QR コードを読み取って検索
 - 各試験キャンペーンの最後で使用
- より短時間 & 高精度での検査が可能に

```

1. Enter the PS-Board ID with QR reader.
2. Prints "OK" if the board passed all the tests.

single-run results =
  258 DataFrame
  Row  id      runid  psboard_id  daughterboard_id  position  resistance_test_passed  qspi  power  clock  adsp
  Int64  Int64  Int64      Int64             Int64     Union{Missing, Int64}  Int64  Int64  Int64  Int64
1  1378  508  1169  1169             508       12       1       1       1       1
2  1508  388  1169  1169             missing    17       1       1       1       1
select extra_results, Not(id, run.tests, (dac.is_B, (icid.fail.B00, (low.afficiency))) =
  822 DataFrame
  Row  id      runid  psboard_id  daughterboard_id  position  insufficient_reset_with_30  reset_failed_though_reconfig_done  always_bit_flag_true  dac_is_B  icid_fail_B00
  Int64  Int64  Int64      Int64             Int64     Union{Missing, Int64}  Union{Missing, Int64}  Union{Missing, Int64}  Union{Missing, Int64}  Union{Missing, Int64}
1  359  1169  1169  1169             missing    missing    missing    missing    missing    missing
Please determine [v/all] y

2.b. Prints the test results and ask for manual decision when the board had any problematic result.

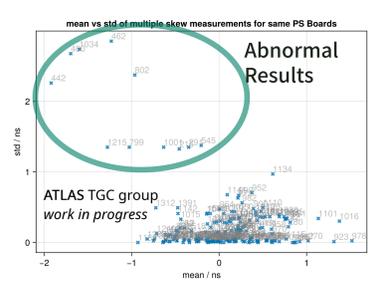
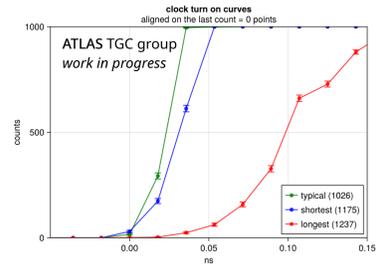
single-run results =
  258 DataFrame
  Row  id      runid  psboard_id  daughterboard_id  position  resistance_test_passed  qspi  power  clock  adsp
  Int64  Int64  Int64      Int64             Int64     Union{Missing, Int64}  Int64  Int64  Int64  Int64
1  1378  508  1169  1169             508       12       1       1       1       1
2  1508  388  1169  1169             missing    17       1       1       1       1
select extra_results, Not(id, run.tests, (dac.is_B, (icid.fail.B00, (low.afficiency))) =
  822 DataFrame
  Row  id      runid  psboard_id  daughterboard_id  position  insufficient_reset_with_30  reset_failed_though_reconfig_done  always_bit_flag_true  dac_is_B  icid_fail_B00
  Int64  Int64  Int64      Int64             Int64     Union{Missing, Int64}  Union{Missing, Int64}  Union{Missing, Int64}  Union{Missing, Int64}  Union{Missing, Int64}
1  359  1169  1169  1169             missing    missing    missing    missing    missing    missing
Please determine [v/all] y

3. Print all of passed board IDs at the end

Finished
PS Board ID: P51170
Info: results
  258 DataFrame
  Row  id      runid  psboard_id  daughterboard_id  position  resistance_test_passed  qspi  power  clock  adsp
  Int64  Int64  Int64      Int64             Int64     Union{Missing, Int64}  Int64  Int64  Int64  Int64
1  1378  508  1169  1169             508       12       1       1       1       1
2  1508  388  1169  1169             missing    17       1       1       1       1
Info: results
  822 DataFrame
  Row  id      runid  psboard_id  daughterboard_id  position  insufficient_reset_with_30  reset_failed_though_reconfig_done  always_bit_flag_true  dac_is_B  icid_fail_B00
  Int64  Int64  Int64      Int64             Int64     Union{Missing, Int64}  Union{Missing, Int64}  Union{Missing, Int64}  Union{Missing, Int64}  Union{Missing, Int64}
1  359  1169  1169  1169             missing    missing    missing    missing    missing    missing
Please determine [v/all] y
    
```

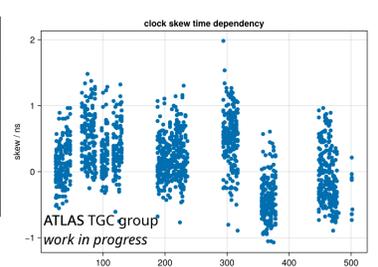
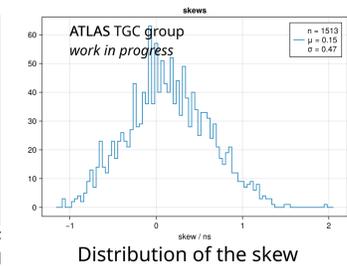
4.B データベースを活用した解析

- e.g., PS Board ごとのクロック位相のスキュー
- 生データ: PS Board で再構成されたクロックの JATHub によるスキャン
- 手順:
 - 場所依存性(JATHub + ケーブルの成分)を取り除く
 - データベースを参照する: (PS Board ID, runid) → position
 - 同じ PS Board で場所を変えて測定した結果をリファレンスとする
 - 複数回試験した PS Board に対しては、複数回の測定の整合性を確認
 - 分散は小さいことが期待される
 - もし大きければ、結果を確認する
 - ここで得られた結果を適宜試験ソフトの判定基準に反映
- 結果:
 - 試験のランごとの分布の違いが見つかった 今後の課題



測定結果のファイル例(一部)

12.928571428571429	0.000000
12.946428571428571	0.000000
12.964285714285714	0.000000
12.982142857142858	5.000000
13.000000000000000	219.000000
13.017857142857142	803.000000
13.035714285714286	990.000000
13.053571428571429	1000.000000
13.071428571428571	1000.000000
13.089285714285714	1000.000000



クロックの波形を表すデータから立ち上がりの回数や位相や立ち上がりにかかる時間を取得する。