

並列化VMEの紹介

理化学研究所

仁科センター

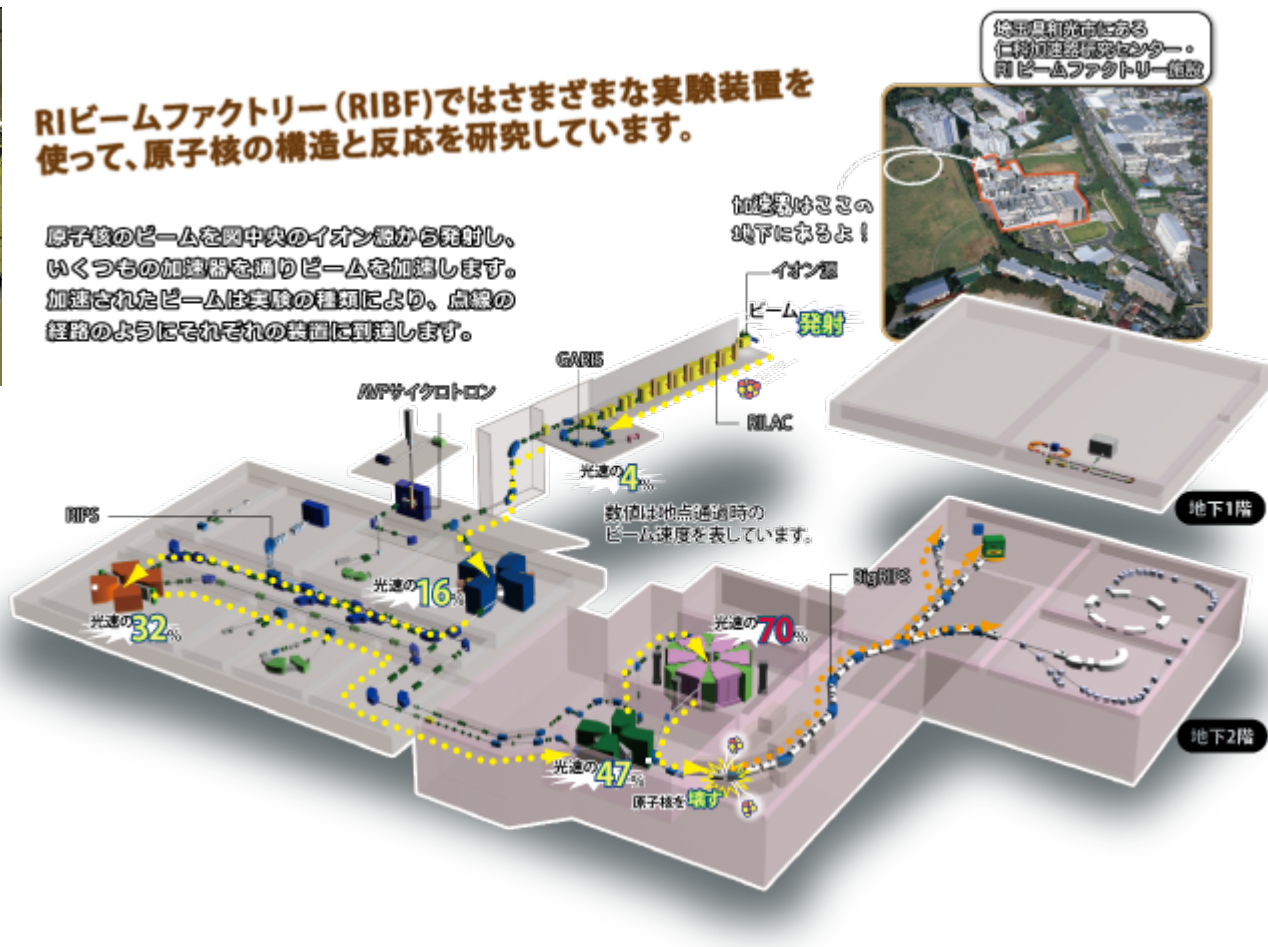
馬場 秀忠

理研RIBF (RIビーム)



RIビームファクトリー (RIBF)ではさまざまな実験装置を使って、原子核の構造と反応を研究しています。

原子核のビームを岡中央のイオン源から発射し、いくつかの加速器を通りビームを加速します。加速されたビームは実験の種類により、点線の経路のようにそれぞれの装置に到達します。



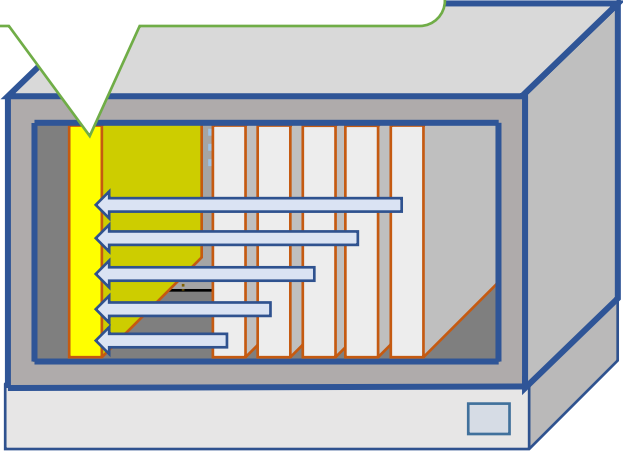
DAQの特徴

- 基本的に
 - 複数のクレートに分けて並列化（～10台）
 - ベースとなるビームライン検出器 200～300ch ADC/TDC
 - プラス実験毎の検出器
 - 長くても数ヶ月でセットアップ変わる
 - 短い時は一週間
 - コモントリガー＋コモンデッドタイム
 - イベント番号だけで同期
 - 一番遅いクレートでデッドタイム決まる
- 異なるDAQ間は
 - タイムスタンプ
 - 100MHz, 48bit

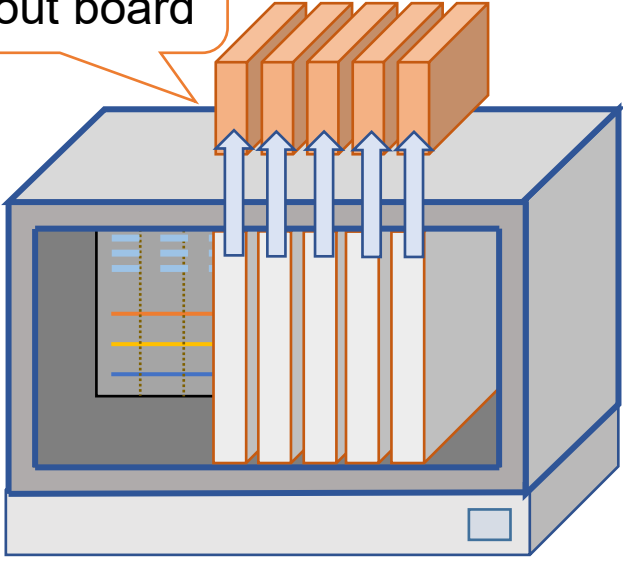
VMEが多い

- ベースとなるビームライン検出器
 - VME (+CAMAC)
 - 非マルチイベントバッファ (イベント毎読出し)
 - だんだんCAMAC減って来た。CC/NETも生産中止みたいだし。VME CPUボードもいろいろ生産中止だ。。。
 - できるだけ高速にデータを取りたい
- その他、外国製独自システムもありますが

1 CPU Board readout
all VME module data



1 module per
1 readout board



Parallelize!!

Readout

~ 100us

(OS Latency ~ 20us)

Readout

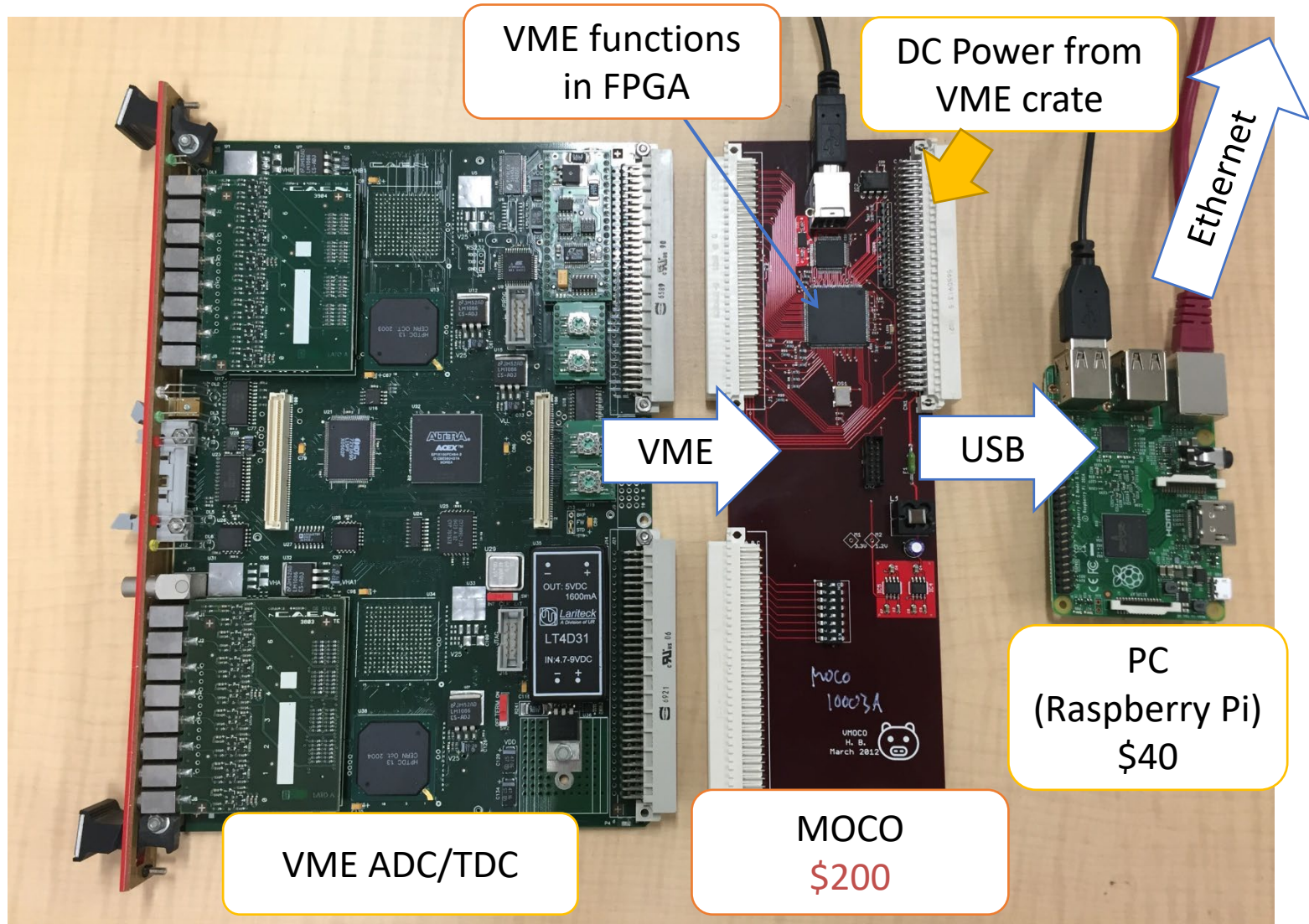
<< 20us

(no OS latency)

Crate Parallel

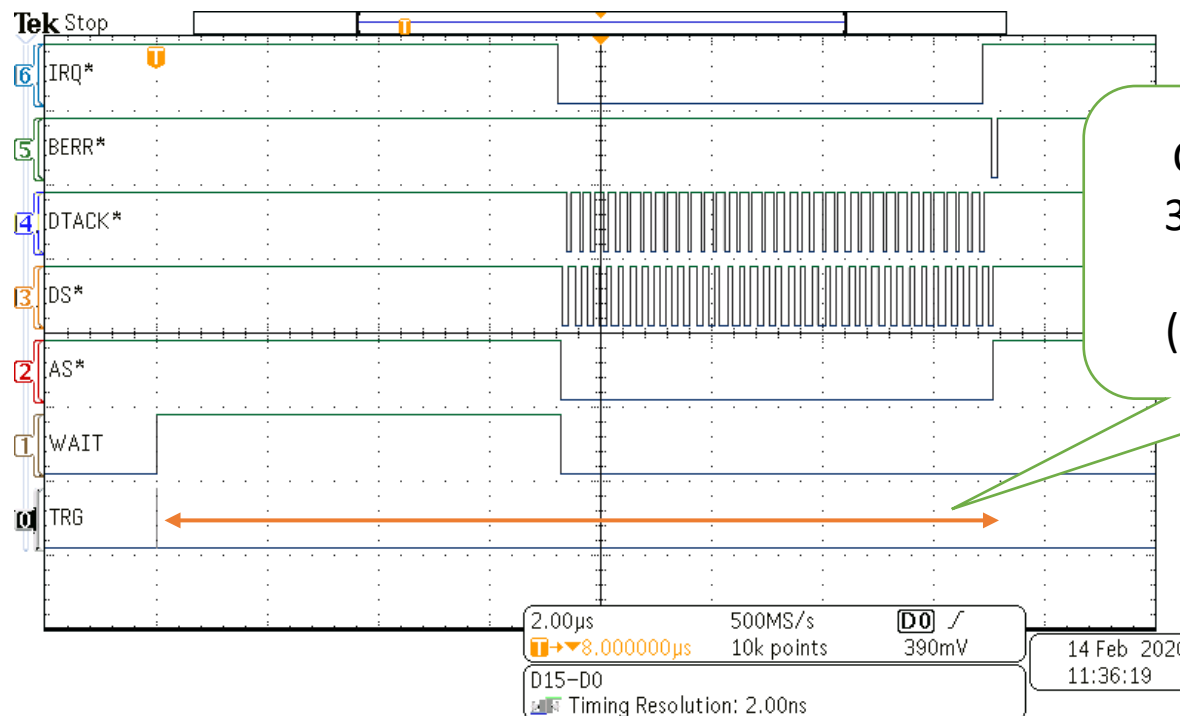
Module Parallel

MOCO (Mountable Controller) 2011~



特徴

- FPGAにVMEマスタを実装
 - 1台のモジュールに1台VMEコントローラ
 - 読出しを完全に並列化
 - VME規格の速度は超えない（典型的に $0.2\mu\text{s} \times N$ ）
 - IRQ応答は $<1\text{clock}$ (20ns)



現状のMOCOで出来る事

- A32, A24, A16
- D32, D16
- Write (Single)
- Read (Single, BLT)
 - BERR 検出
- IRQ検出

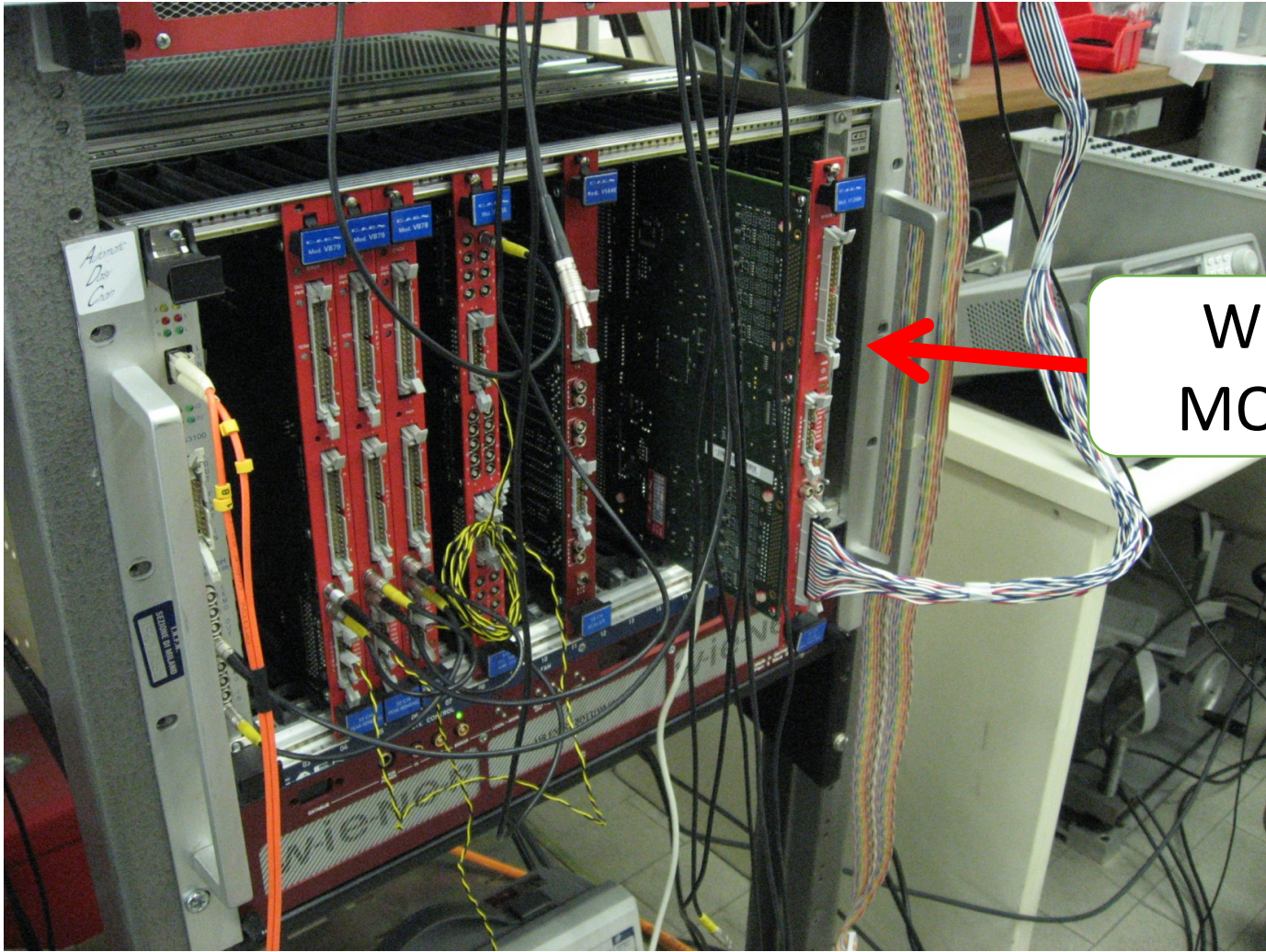
いろいろ問題

上からUSBケーブル

トリガーとビシーのケーブル
がとにかくさしにくい



基板むき出しなのでショートする

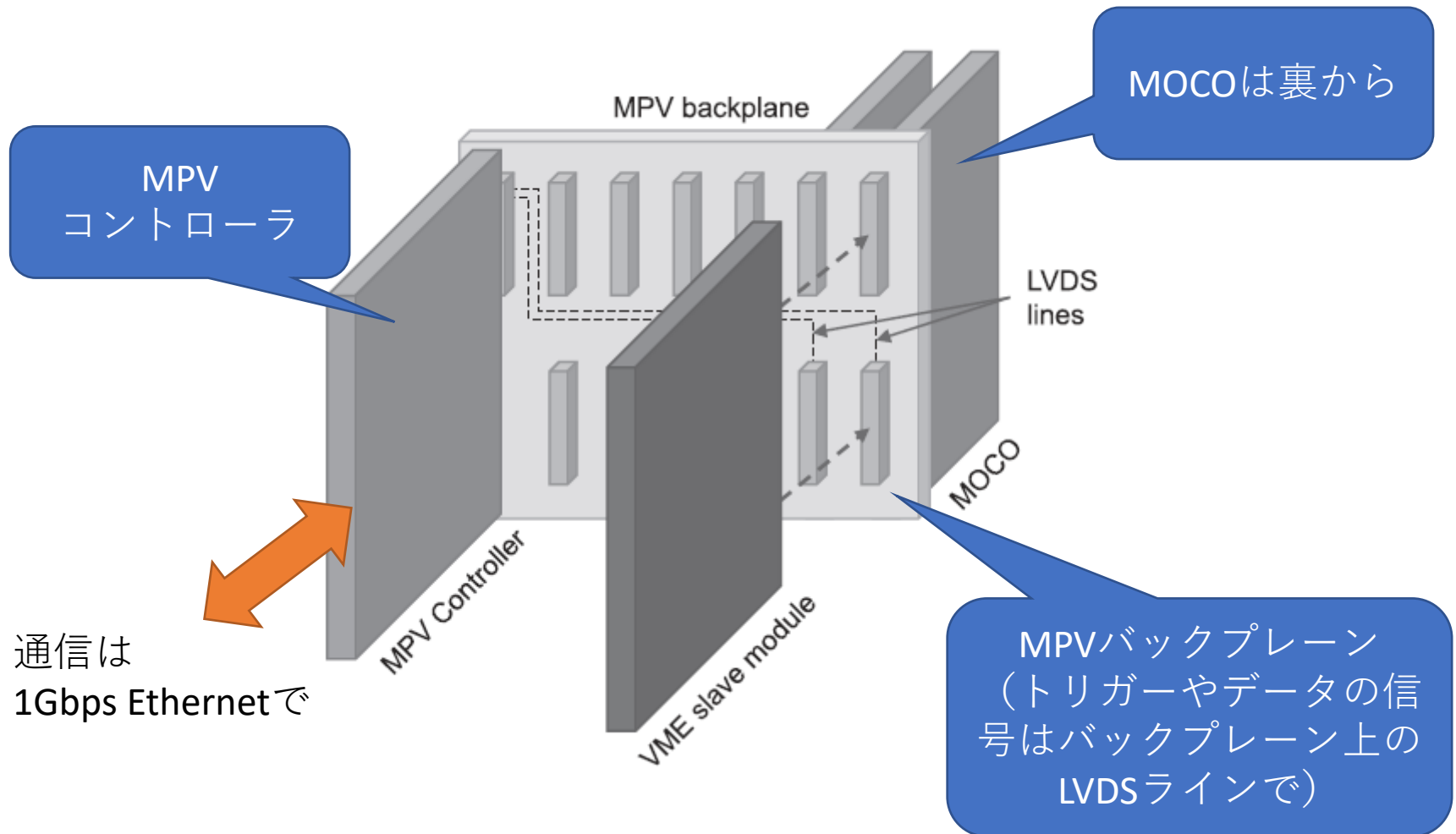


ちゃんとやろう

- クレートから飛び出さない
- USBやめる
 - Ethernetからデータ取れるように
- トリガーとビジーのためにケーブルささない
- MOCOはほとんどデザイン変えない

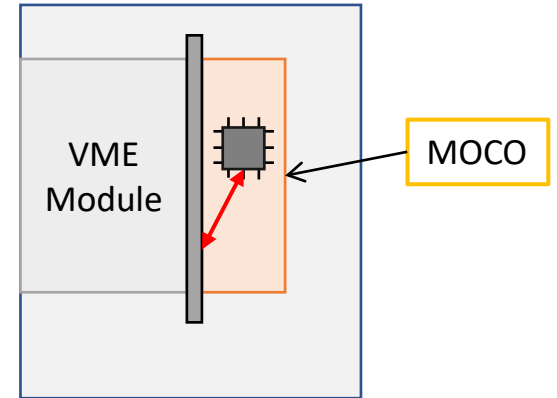
2019年～

MPV : MOCO with Parallelized VME

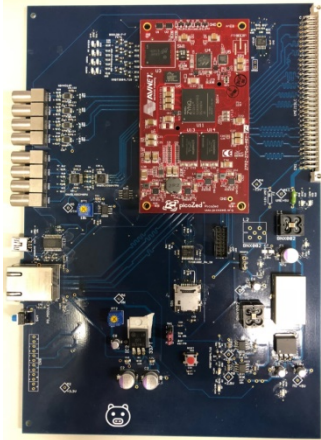


写真

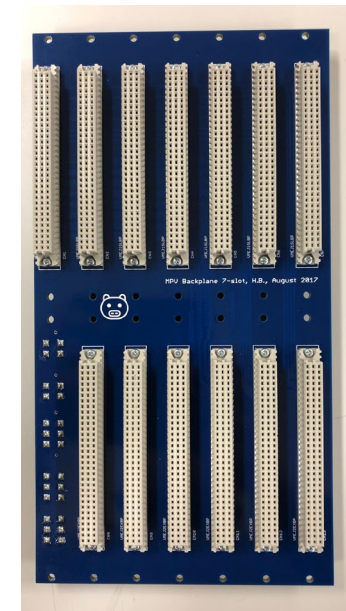
Side view



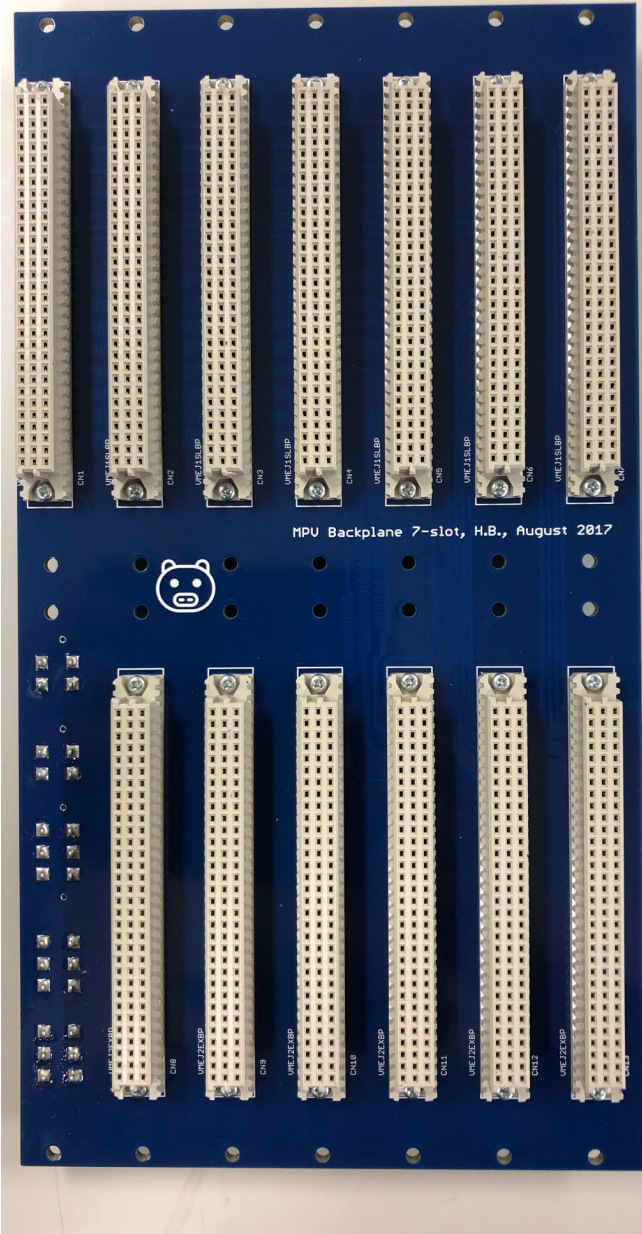
MPV Controller



MPV Backplane

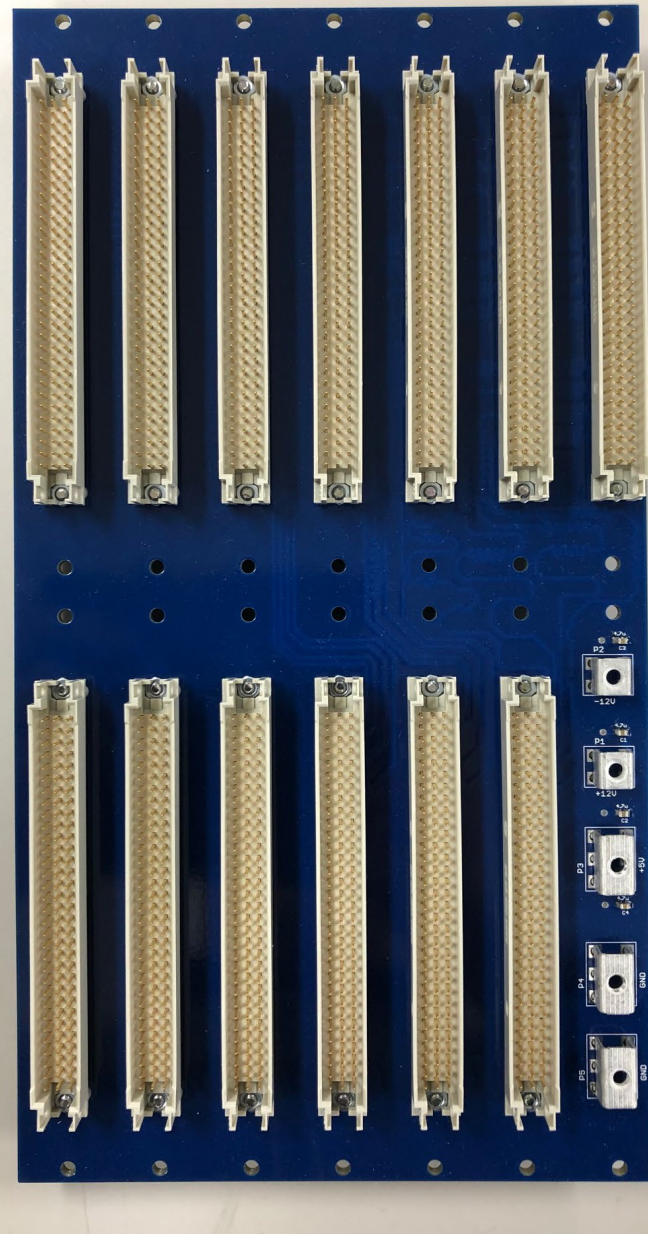


表

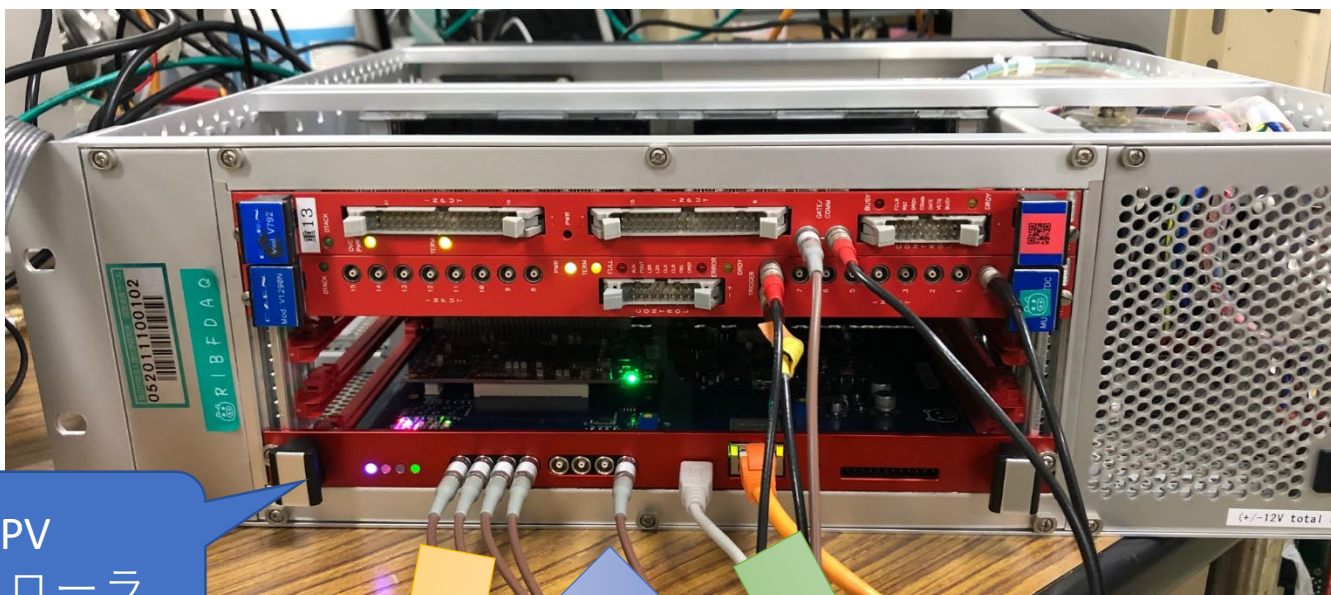


並列化VME 馬場

裏



写真



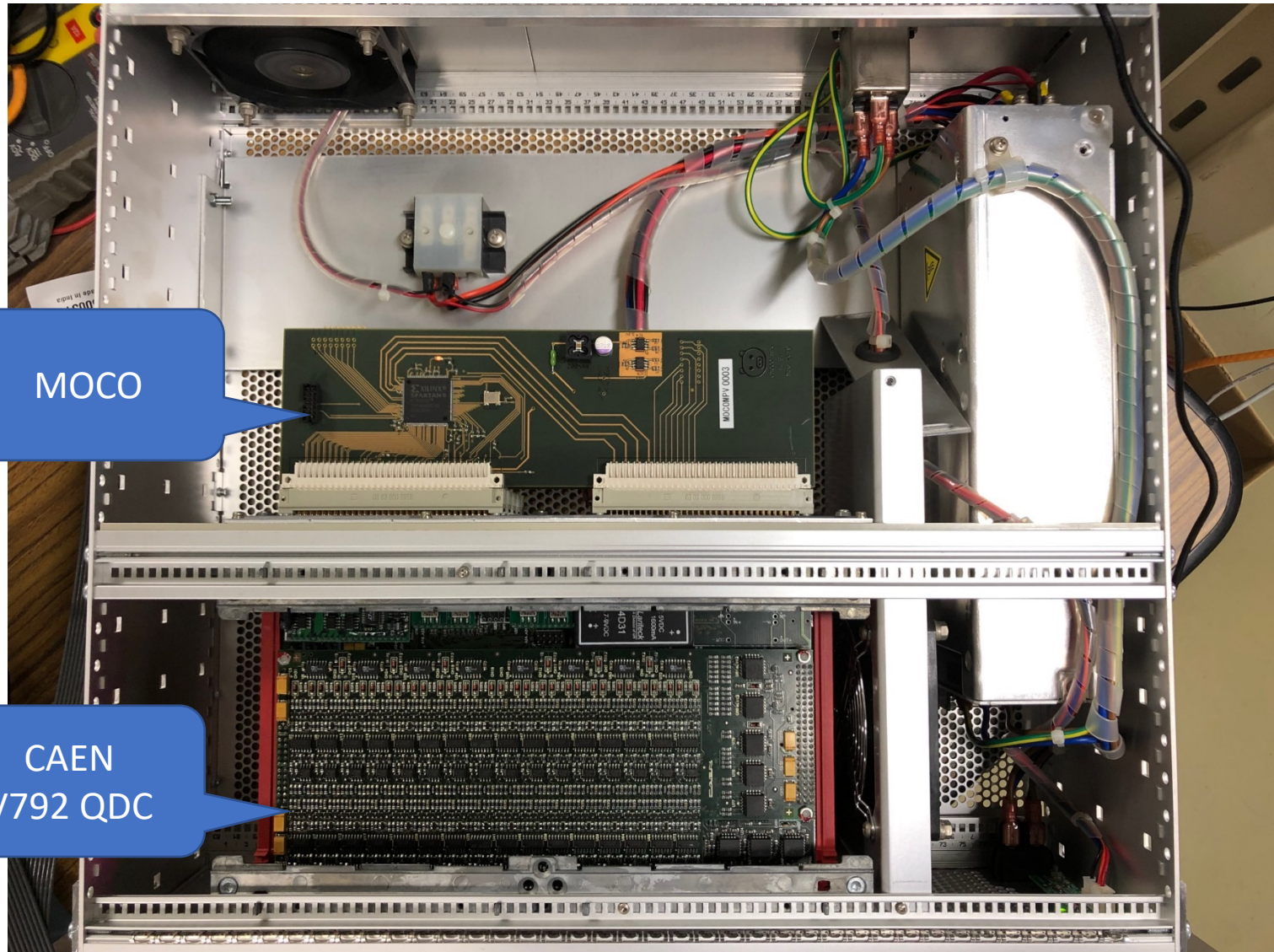
MPV
コントローラ

Busy

Trigger

Data

上から見た写真



MOCO

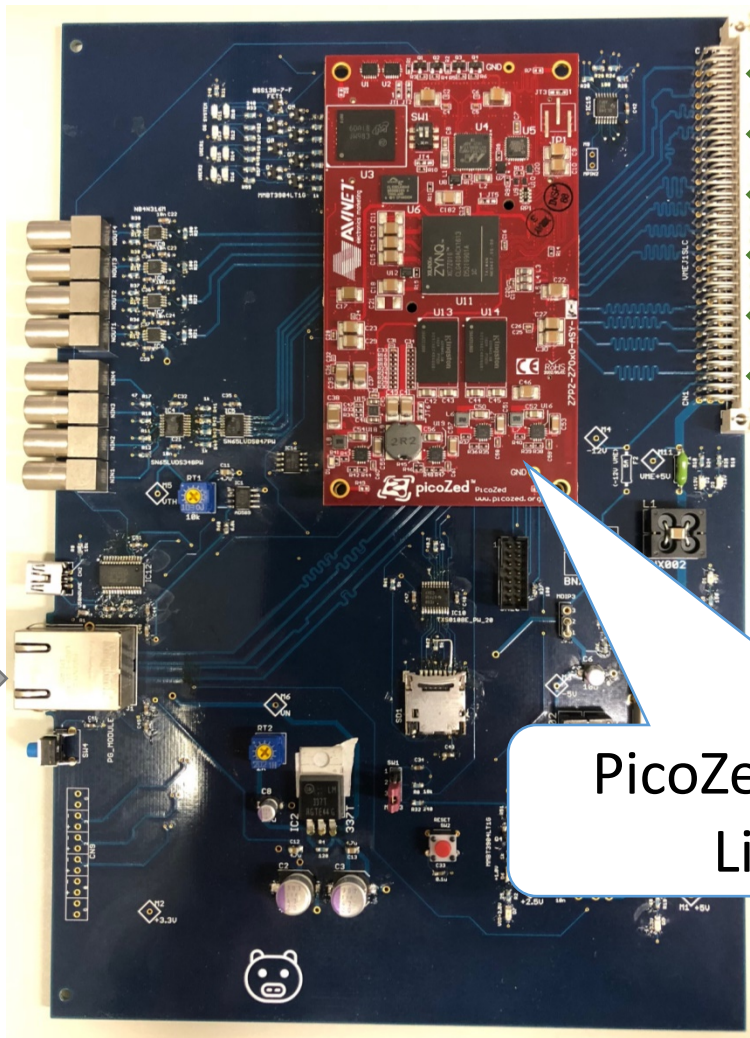
CAEN
V792 QDC

MPV Controller

NIM I/O
Trigger,
Busy, etc



1Gbps
Ethernet



LVDS I/O (4 x 6 Lines)



MOCO 1

MOCO 2

MOCO 3

MOCO 4

MOCO 5

MOCO 6

トリガーやデータはバックプレーン経由で

PicoZed 7010 (Xilinx Zynq)
Linux動いている

内部でWebサーバーが走っている

MPV Controller Index (mpv06)

index.cgi / 190919

Refresh

Flash memory = Read Only

Write OK Read Only

MPV Status

```
*** MPV Status ***
Firmware : 1.25
MOCO Status :
MOCO0 : Enable Ready TestOK
MOCO1 : Enable Ready TestOK
MOCO2 : Enable Ready TestOK
MOCO3 : Enable Ready TestOK
MOCO4 : Enable Ready TestOK
MOCO5 : Enable Ready TestOK
Trigger Select : in0
Busy Select : moco0 moco1 moco2 moco3 moco4 moco5
Level output : 0x0000
NIM Out 0 : ungated
NIM Out 1 : gated
NIM Out 2 : clk10k
NIM Out 3 : eob
```

EFN

Process EFN Set

babildes 66 Set

babies

- MOCO0 SegmentID : 0-3-1-8-0 [modify](#) (C32)
- MOCO1 SegmentID : 0-3-3-8-0 [modify](#) (C32)
- MOCO2 SegmentID : 0-3-4-8-0 [modify](#) (C32)
- MOCO3 SegmentID : 0-3-5-8-0 [modify](#) (C32)
- MOCO4 SegmentID : 0-3-6-8-0 [modify](#) (C32)
- MOCO5 SegmentID : 0-3-2-21-0 [modify](#) (V7XX)

babildes mode

[change setting](#)

MOCO Setting Index (MOCO 5)

moco.cgi / 190919

[back to index](#)

Refresh

Flash memory = Read Only

Module

SegmentID : 0-3-2-21-0 (V7XX)

Current New

Device BIGRIPS 0

Focal F3 3

Detector PLAQ 2

Module V7XX 21

EF type Normal 0

Update

[See SegID Web Page](#)

Initialize parameters

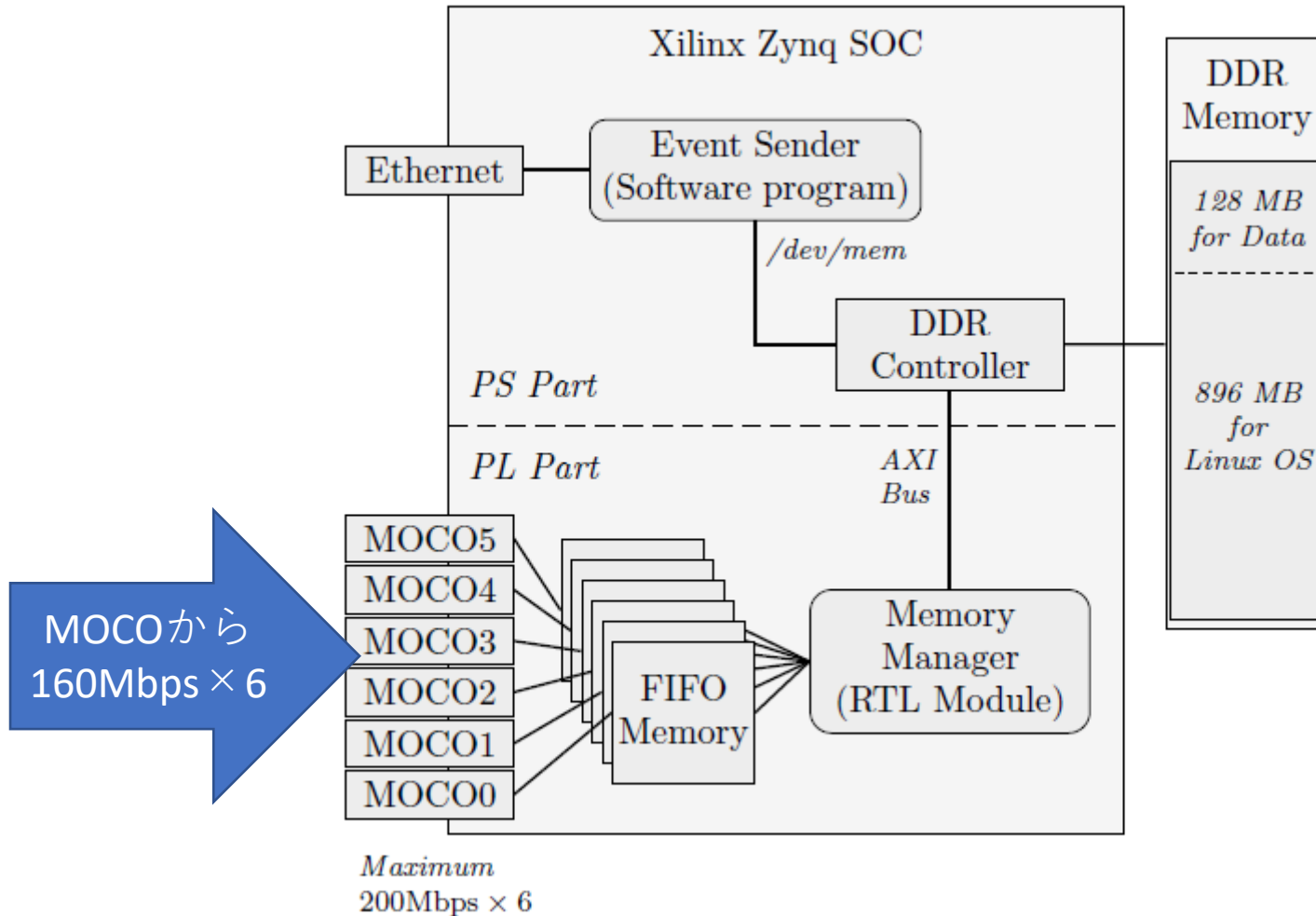
Set

formv7xx.sh / 190903

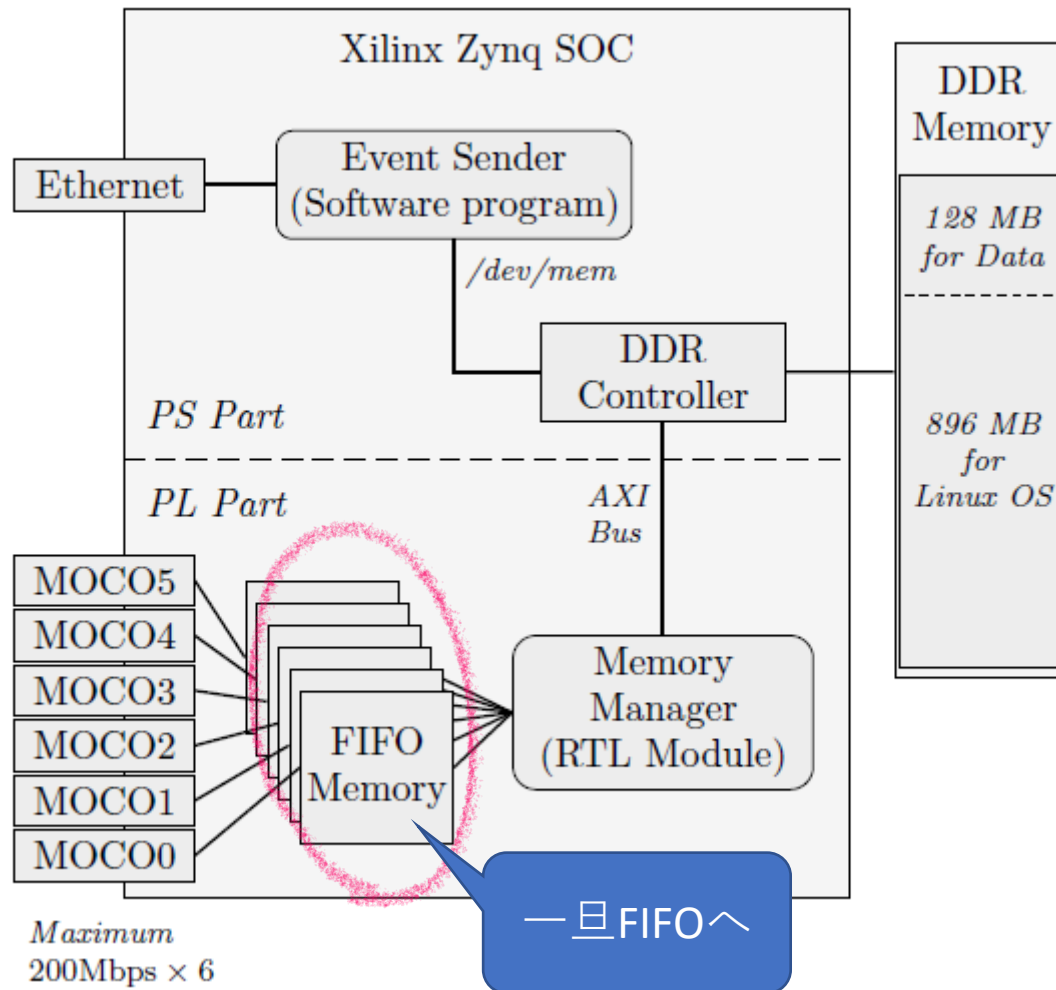
| Parameter | Set | Current |
|----------------------------|----------------------------------|---------|
| Geometry | <input type="text" value="0"/> | 0 |
| IPED/FSR | <input type="text" value="250"/> | 250 |
| V785 case, should be blank | | |
| Threshold (ch0) | <input type="text" value="0"/> | 0 |
| Threshold (ch1) | <input type="text" value="0"/> | 0 |
| Threshold (ch2) | <input type="text" value="0"/> | 0 |
| Threshold (ch3) | <input type="text" value="0"/> | 0 |
| Threshold (ch4) | <input type="text" value="0"/> | 0 |

使うモジュールを選んで、
閾値など設定

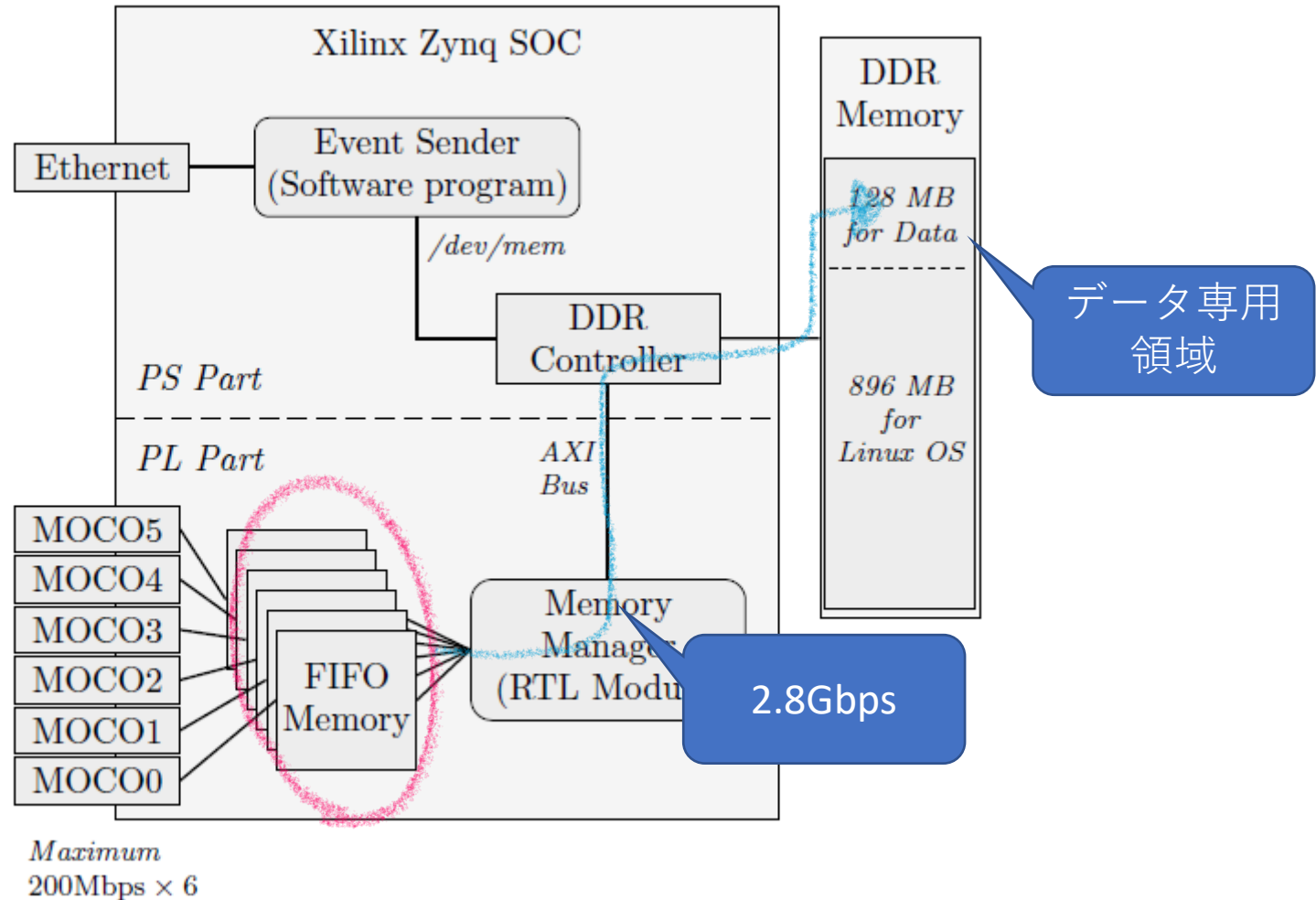
データの流れ



データの流れ



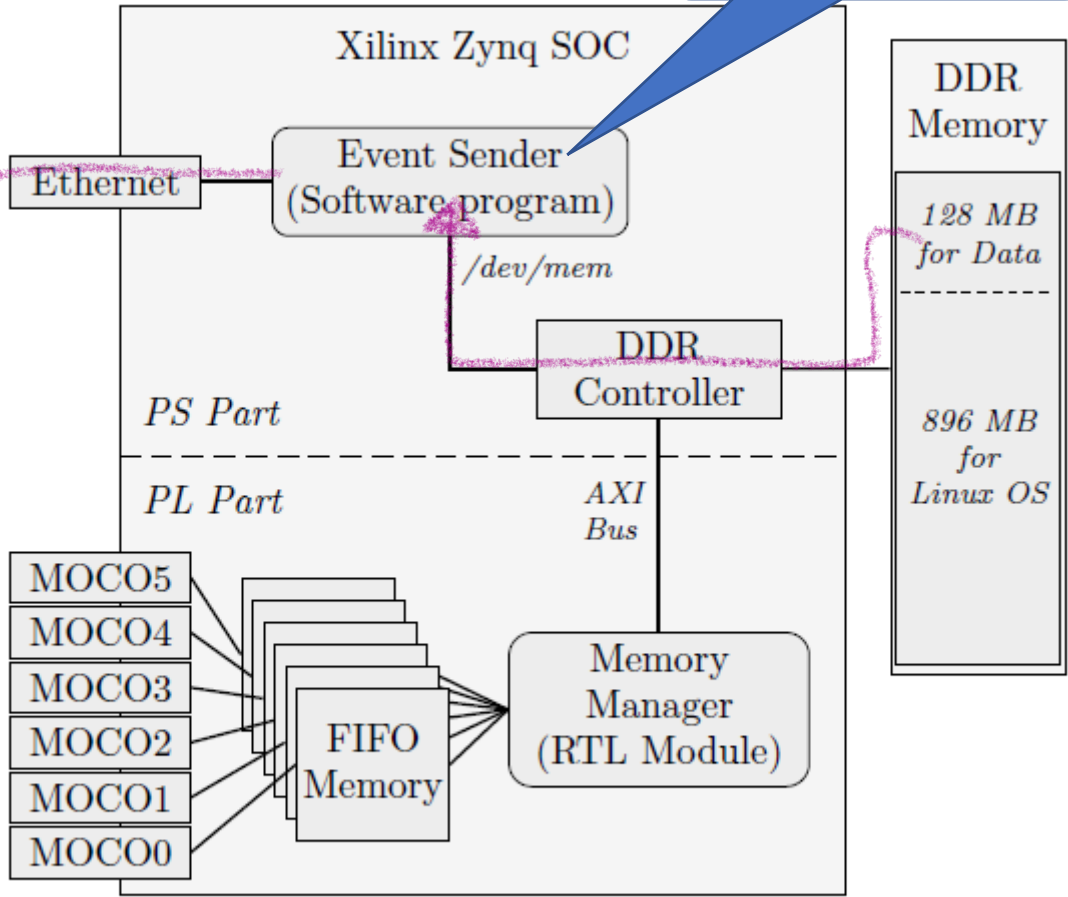
データの流れ



データの流れ

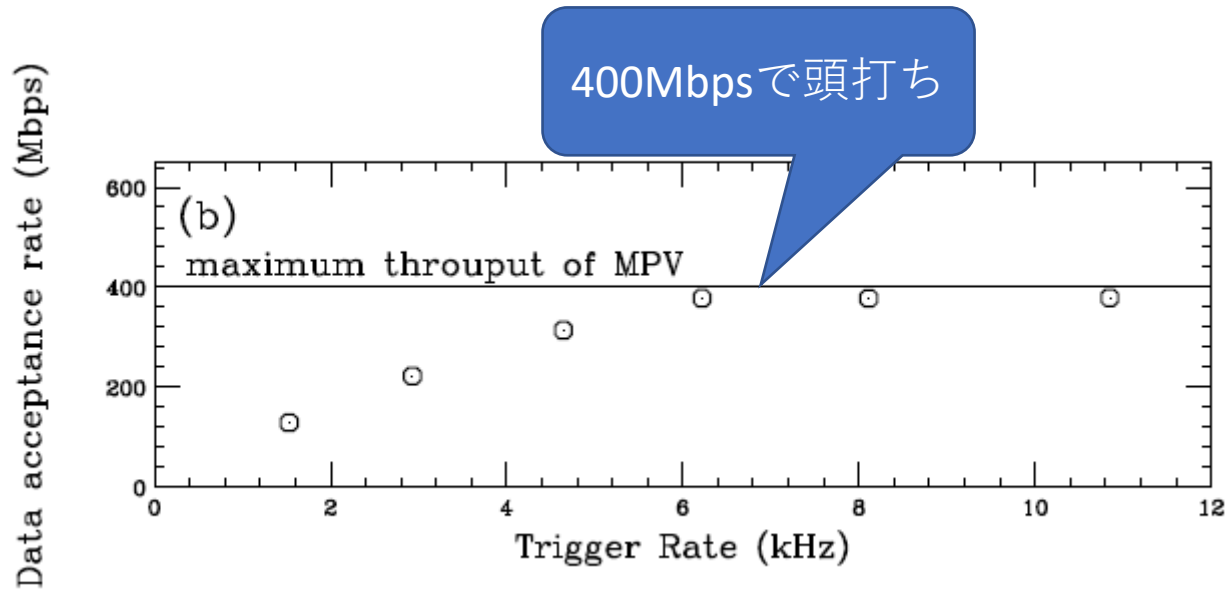
ソフトウェア (CPU) でイベント構築

サーバーへ



Maximum
200Mbps × 6

イベント構築 (CPU) が律速



iPerf3だと800Mbps
イベント構築しないと600Mbps

動作確認済みモジュール

- CAEN
 - V775, V785, V792, V1190, V1290
- Mesytec
 - MADC32, MQDC32, MTDC32
- SIS
 - SIS3820
- 二木工業
 - LUPO



いずれも最新
firmware

まとめとこれから

- 並列化VMEが出来た (MPV)
 - RIBFでは一部常駐して稼働中2019～
 - 全面的にMPVに移行したい
- 制約
 - モジュールによってはMOCOのファームウェアを専用に作らないとならない
- これから
 - 企業から買えるように
 - RIBF以外にも需要ありますか？
 - MOCOをもう少し真面目にする
 - リモートでファームウェアアップデート出来るように
 - 現状、背面からJTAGささないとならないので
 - 回路のマイナーアップデート

ブロック図

