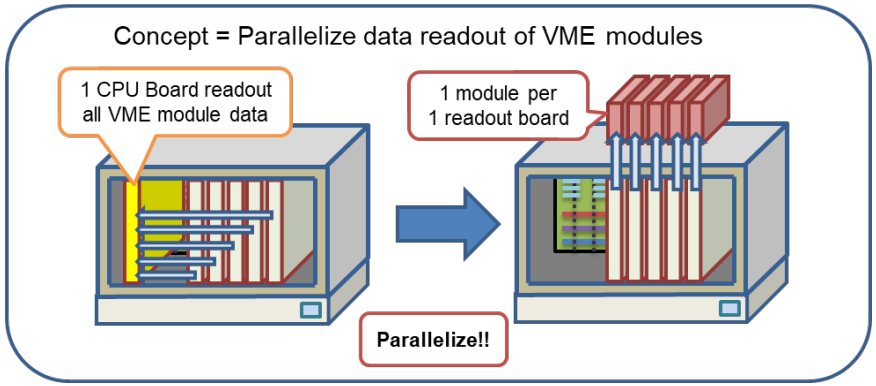
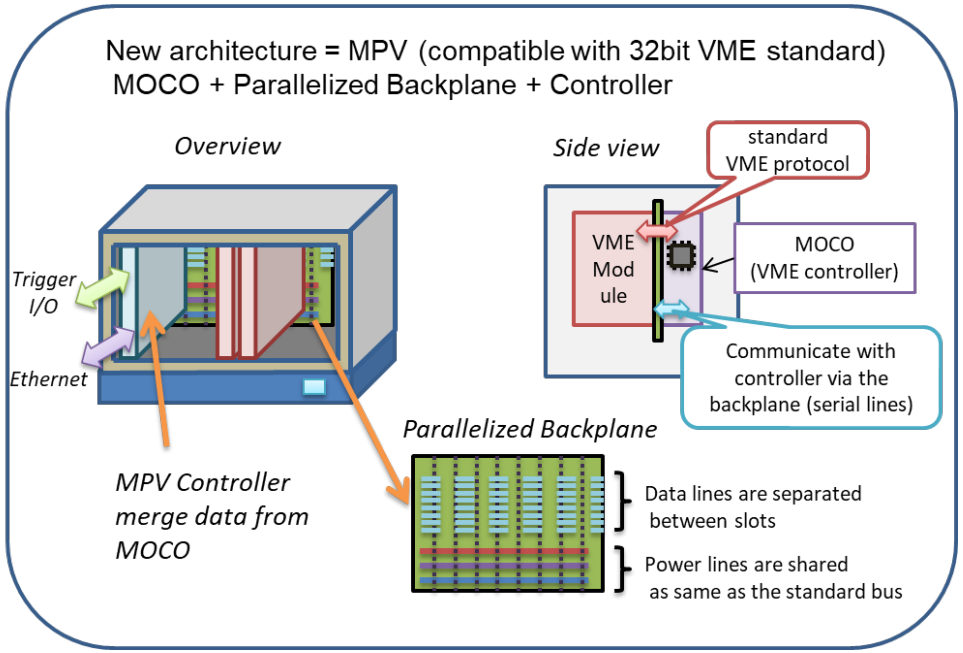
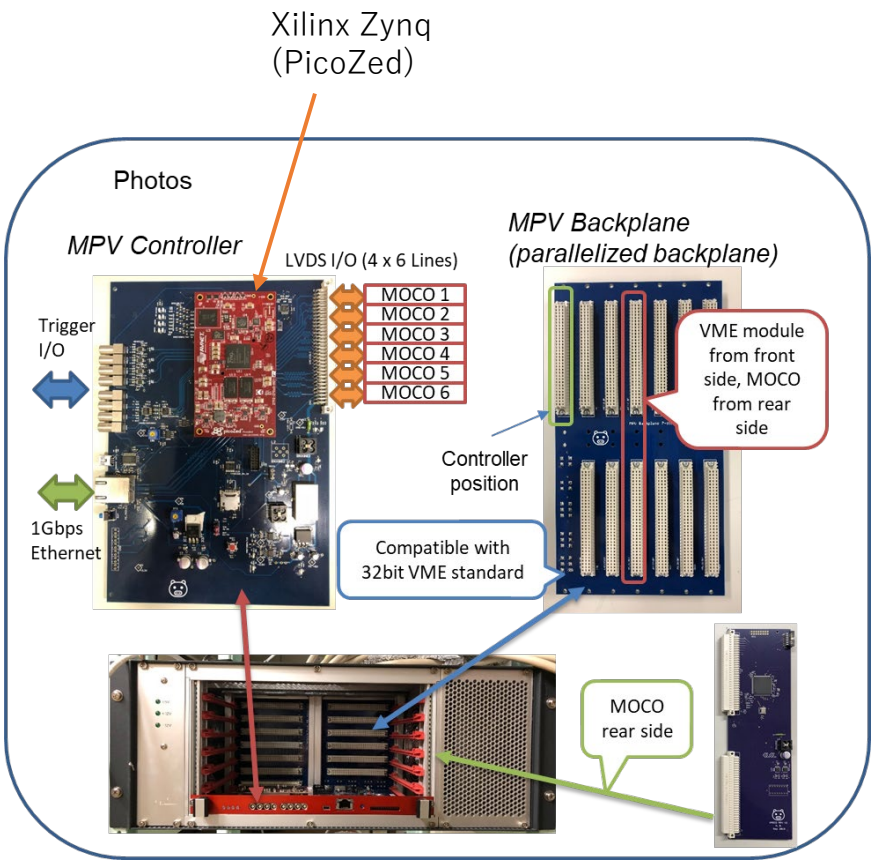
MPV – Parallel Readout Architecture for the VME data acquisition system

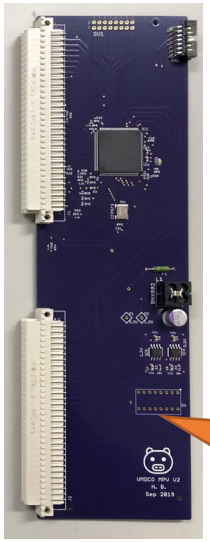
H.Baba, T. Isobe, T. Ichihara, T. Onishi, K. Yoshida, Y. Watanabe, S. Ota[A], H. Shimizu[A], S. Shimoura[A], S.Takeuchi[A], D. Nishimura[B], J. Zenihiro[C], A. Tokiyasu[D], R Yokoyama[E]
RIKEN Nishina Center, CNS University of Tokyo[A], Tokyo City University[B], Kyoto University[C], Tohoku University[D], Tennessee University[E]



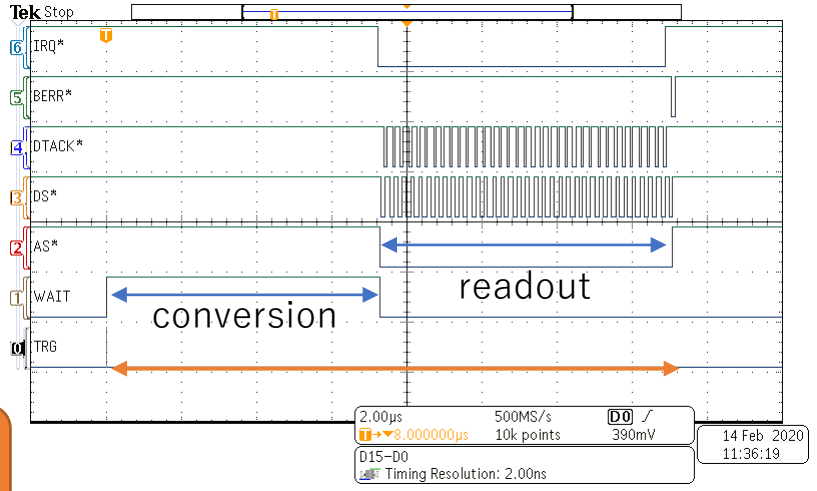




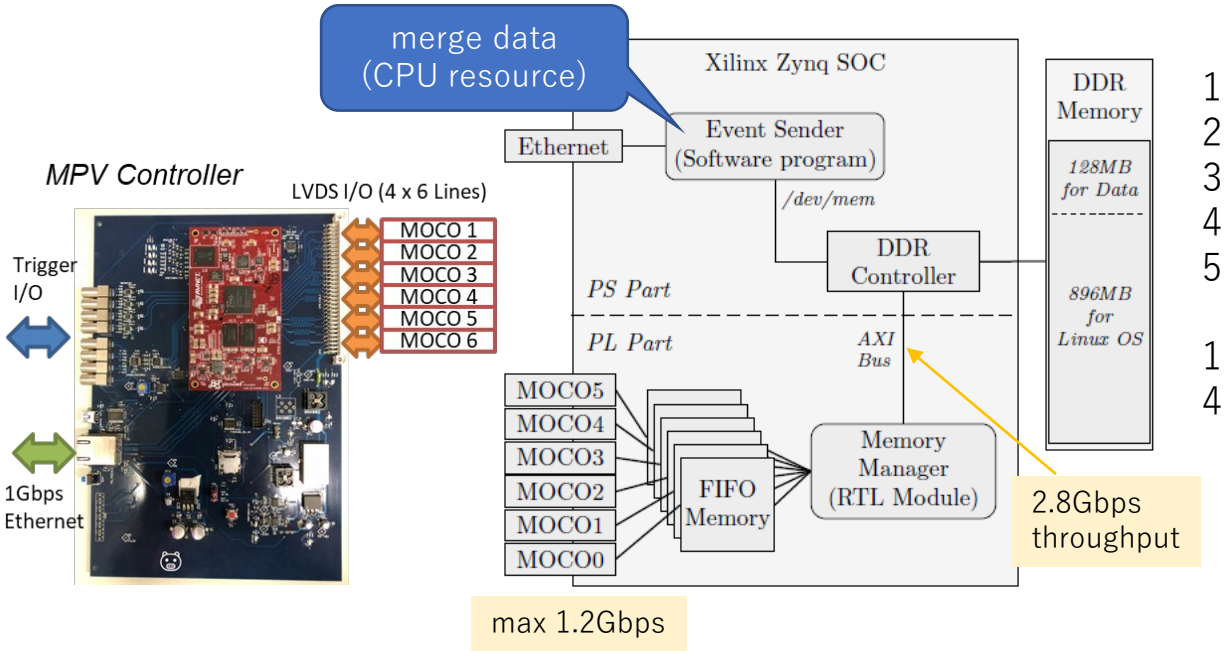
Dataway



- CAEN V792 34word readout
- **15us** dead time
  - conversion 7.4us
  - readout 7.6us
- interrupt latency < 20ns



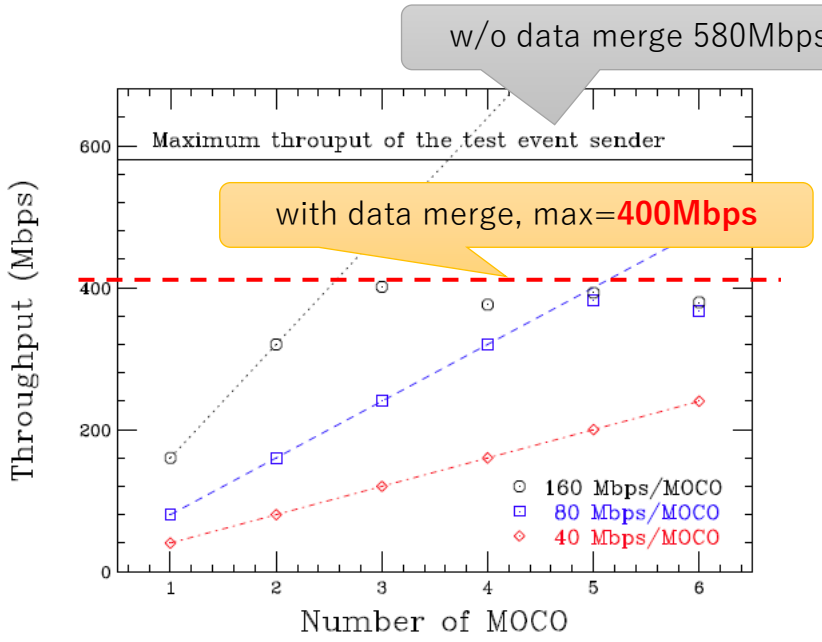
MOCO
cost-efficient VME master (FPGA)



1. FPGA : Receive data from MOCO
2. FPGA : store data into FIFO
3. FPGA : copy data to DDR
4. CPU : merge data from DDR
5. CPU : send merged data to server

1-3 = Real Time
4-5 = depends on CPU resource

Performance measurements



- 10kHz periodic trigger
- 40, 80, 160 Mbps dummy data / MOCO
- Maximum data throughput = 400Mbps
  - w/o data merge = 580Mbps

- Random trigger
- Compared with VMIVME (CPU board)
- readout data from x1 or x2 V792 QDCs
- VMIVME (CPU board)
  - dead time depends on the number of modules
  - OS's interrupt latency, 20us
- MPV (Parallel readout)
  - the same dead time for x1, x2 V792
  - no interrupt latency