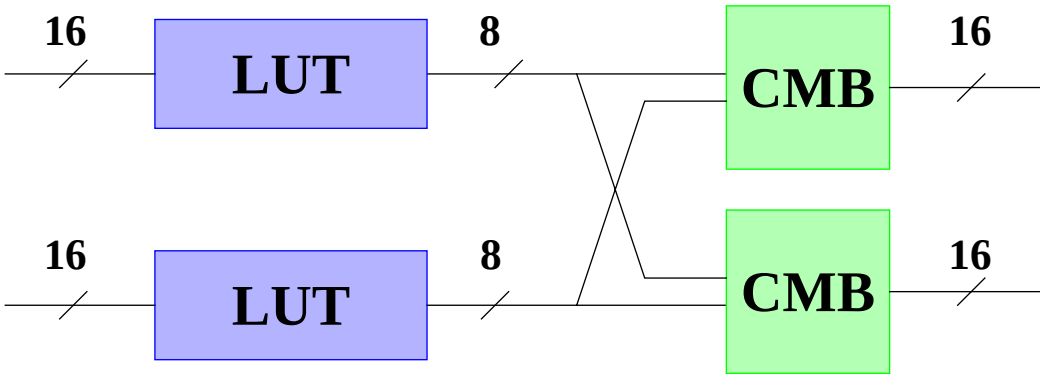
Integration of the Central Trigger Processor Demonstrator and Final Design of the Central Trigger Processor

- **Central Trigger Processor Demonstrator**
  - Specification
  - Hardware
  - Software
- **Central Trigger Processor**
  - Final Design
  - Modules
  - Interface to TTC Partition

CTPD - Specification

→ EDMS ATL-DA-ES-0005:

- trigger inputs: **32** bits ⇒ trigger items: **32** bits



- synchronization: BC or !BC; alignment: 1 to 24 BC
- test memory: 1 MWords or random generator (32 independant bits)
- monitoring: event sampling, scalers, temperature/voltage
- no ROD functionality!

• Modules:

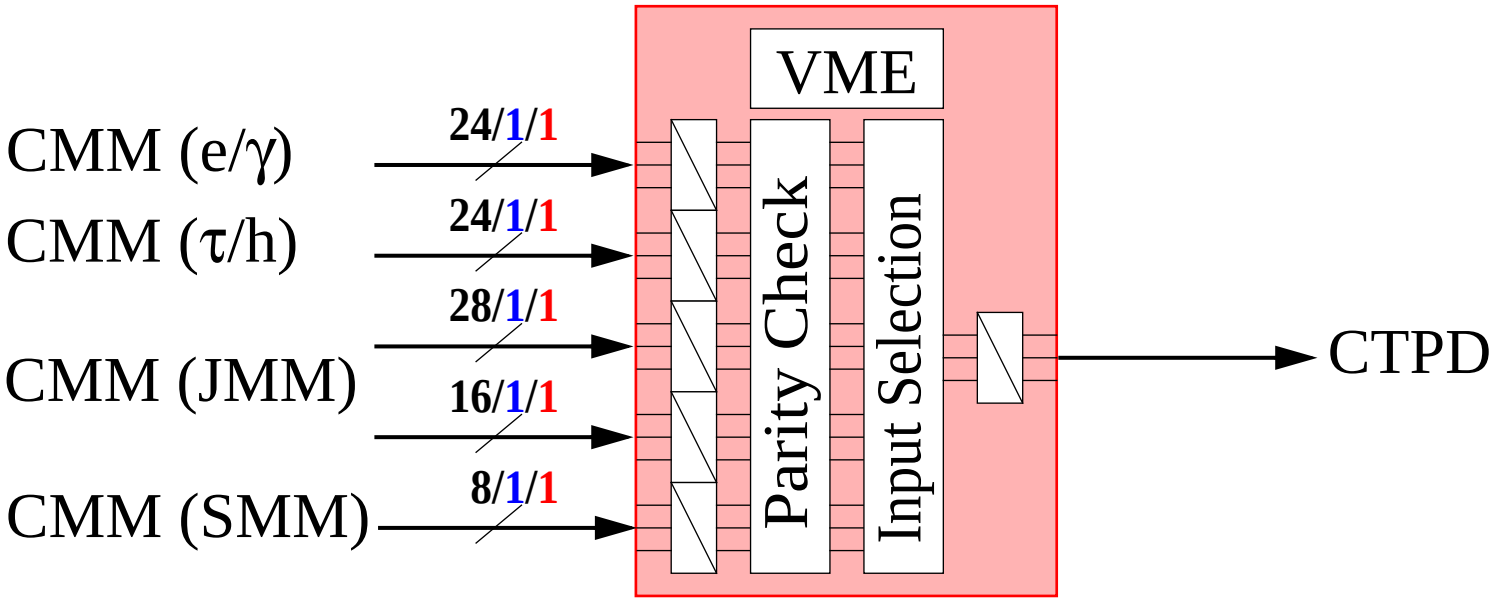
→ 1996, 2 modules 9U VME(32):

module #0: Altera Max7000 on sockets

module #1: Altera Max7000S in-situ programmable

CTPD - Patch Panel

→ Y. Ermoline



⇒ total input signals:

108 trigger inputs + 5 clock + 5 parity

- internal groups (with clock fan-out):

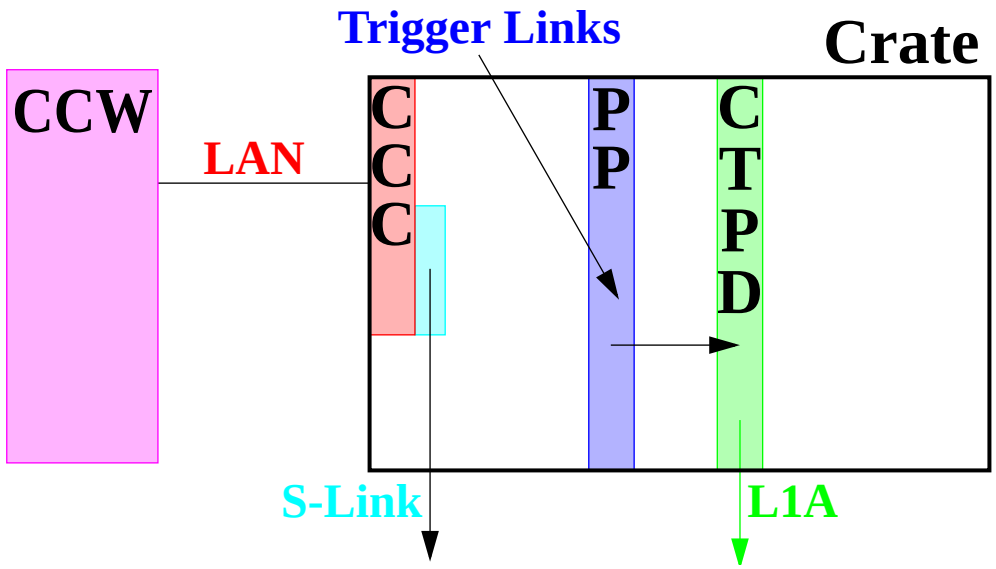
$25 \times$ (**4 trigger inputs + 1 clock)**

- output selection to CTPD:

$8 \times$ (**4 trigger inputs + 1 clock)**

→ status: PCB ready mid March 2002, mounting end April 2002

CTPD - Hardware



Crate	CTP Crate	9U VME(32) Crate	HD **???**
CCC	CTP Crate Controller	Concurrent VP PMC/P33 + S-Link Source Card (+ adapter) + S-Link Destination Card (+ adapter)	CERN CERN CERN
PP	Patch Panel	Patch Panel + cable **PP** → **CTPD** + external signals: BC, BCR, ...; TTCvi	Y. Ermoline CERN HD, UK **???**
CTPD	CTP Demonstrator	module #1	CERN
CCW	CTP Crate Workstation	PC (portable) → boot service for **CCC**	CERN, HD **???**

CTPD - Software

- ROD Emulation:
  - read from CTPD monitoring FIFO; format event data; send to S-Link
  - ROD Crate DAQ (?) or ROS software
  - first tests (simple reading) indicate: 10 to 50 kHz event rate

- Run Controller:
  - to be developed (Online software)
    - → will include control of patch panel
    - → configuration data mostly in local files with ad-hoc format

- Trigger Menu Handler:
  - to be adapted from existing trigger menu handler
    - → **(EDMS ATL-DA-ER-0002)**

  what flexibility of trigger is menu required?
  ⇒ can we put everything in LUTs?

CTP - Final Design - I

- Constraints:
  - start in spring 2002
  - build on experience with existing CTPD
    and Preliminary Design Review April 1999
  - provide final functionality $\Rightarrow$ Final Design Review in spring 2003

- Features:
  - trigger inputs: **160** bits, trigger items: **96** bits
  - trigger input $\rightarrow$ L1A: 4 BC latency
  - fully functional ROD for DAQ (event) and Level-2 (RoI)
  - beam monitoring
  - trigger input capture and play-back
  - fan-out to sub-detector TTC partitions

CTP - Final Design - II

→ ***First Ideas:***

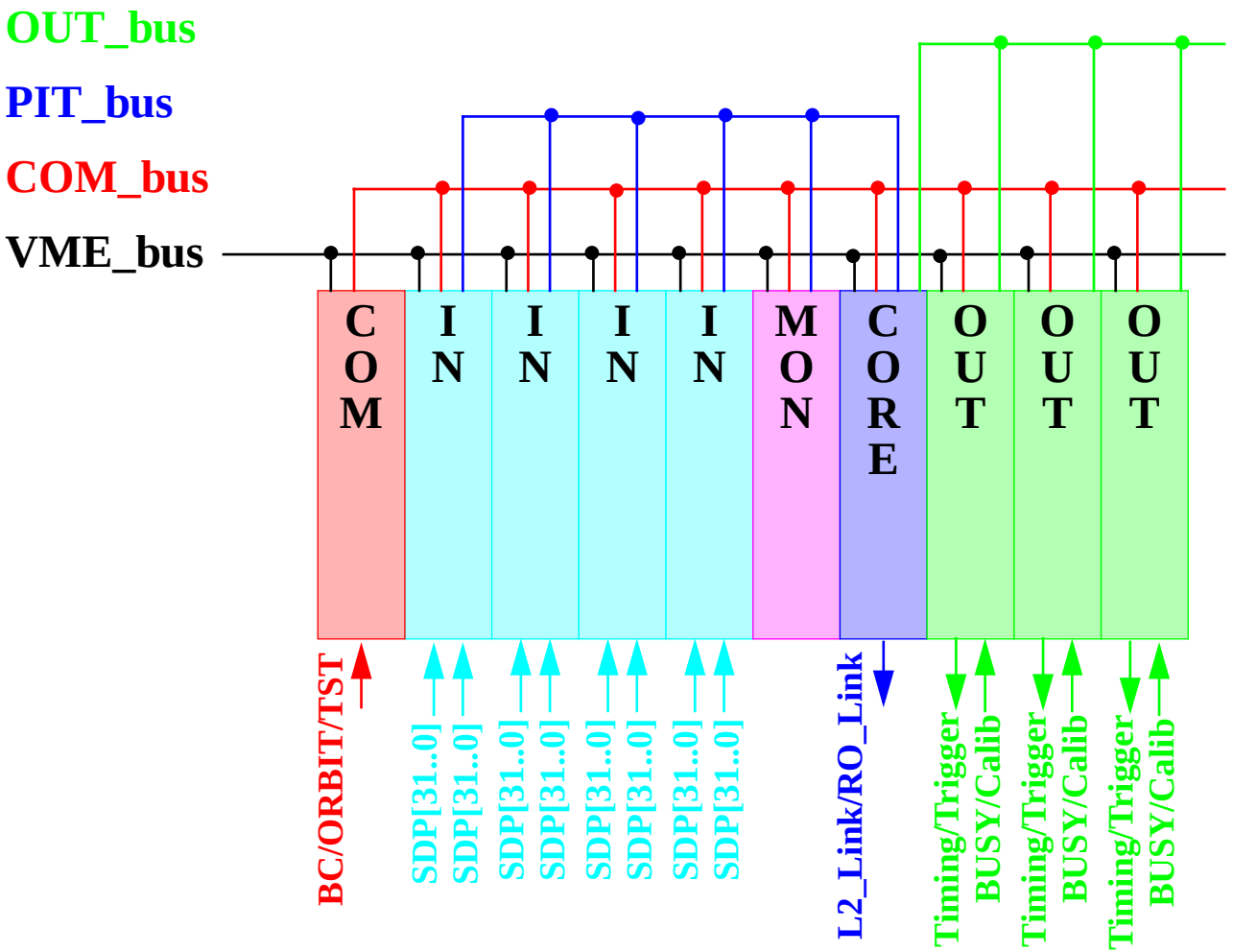
- CTP Modules:
  - → split functionality into different modules:
    - input module(s): receive trigger inputs from trigger processors
    - core module: implement trigger menu (+ ROD)
    - output module(s): fan-out to sub-detectors

    + monitoring module: beam monitoring
    + common module: common timing signals

- CTP Backplane:
  - → dedicated segmented backplane (max 300 signals)

    + VME64x to all modules

CTP - Overview



CTP - Modules - I

• Common Module:

- receive timing signals from LHC
- provide timing and trigger signals common to CTP modules

→ **COM_bus:**

BC, ORBIT, ECR, BUSY, TST (?)

• Input Module:

- receive trigger inputs: 2×32 bit **(SDP = Sub-Detector Pattern)**
- phase measurement, synchronization, parity checking, alignment
- bi-directional FIFO for injecting/capturing trigger inputs
- scalers for counting of (decoded) trigger inputs
- selection and routing of trigger inputs to **PIT_bus**

→ *use switching matrices*

→ **PIT_bus:**

PIT[0..159] **(PIT = Pattern In Time)**

CTP - Modules - II

• Monitoring Module:

implement beam monitoring:

histogram continuously hits per trigger input and per bunch crossing

$\Rightarrow$ 32 bit $\times$ 160 $\times$ 3564 = 2.2 MByte

$\rightarrow$ *use FPGAs with large embedded memory*

• Core Module:

- implement trigger menu:

160 trigger inputs, **96** trigger items

$\rightarrow$ *use look-up tables and/or combinatorial devices*

- implement dead-time algorithms

- send Level-1 trigger decision to **OUT_bus**

- send event data $\rightarrow$ ROS (**RO_Link**) , and RoI data $\rightarrow$ RoIB (**L2_Link**)

$\rightarrow$ **OUT_bus:**

L1A, Pre-pulse, Trigger Type

$\rightarrow$ **RO_Link, L2_Link:**

S-Link: event data and RoI data

CTP - Modules - III

• Output Module:

implement CTP-to-Detector Interface

send **Timing/Trigger** signals, receive **BUSY/Calib** signals

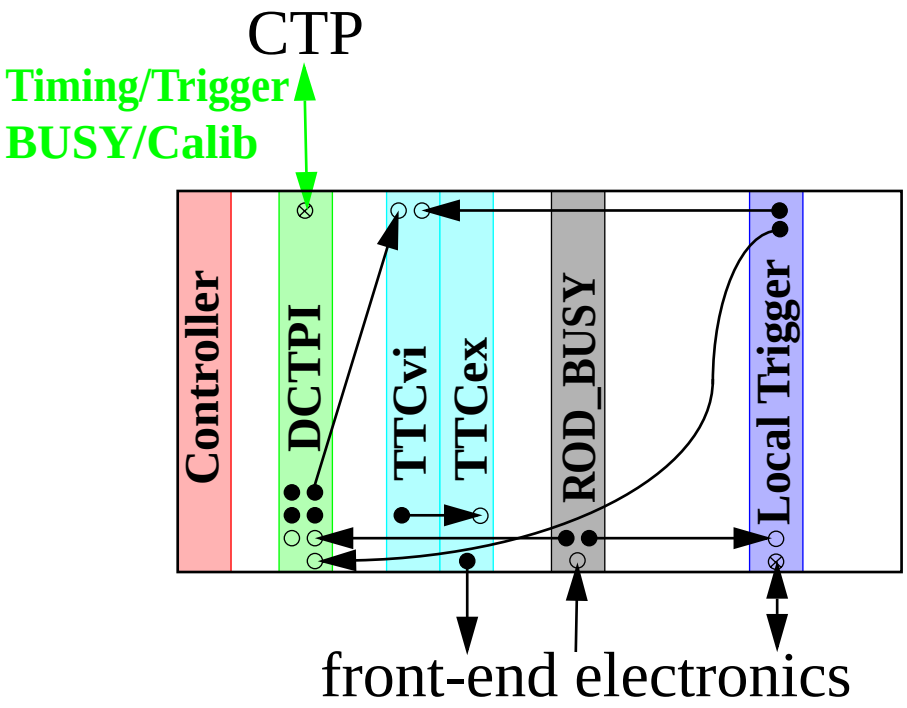
Timing/Trigger **CTP → TTC Partition**		***Busy/Calib*** **CTP ← TTC Partition**	
BC	1 bit	calibration	3 bit
ORBIT	1 bit	BUSY	1 bit
L1A	1 bit		
ECR	1 bit		
Trigger Type	8 bit		
Pre-pulse	1 bit		

→ one cable between CTP and TTC partition

→ several TTC partitions of the same sub-detector can be daisy-chained

CTP - Interface to TTC Partition

→ TTC Partition:



- **Detector-to-CTP Interface (DCTPI)** designed by CERN Level-1
- **ROD_BUSY Module** designed by CERN Level-1
- **TTCvi + TTCex/TTCtx/TTCmi/TTCmx** from CERN EP/ESS
- **Local Trigger Logic** provided by sub-detectors;

  generates sub-detector specific calibration triggers; can be NIM logic