uTCA IP Carrier Application in SLAC

Richard Dabney, Jim Chen (Hytec) , Ernest William, Till Straumann
SLAC National Accelerator Laboratory
Hytec Electronics Ltd

October 23, 2014

Contents

- Latest developments
- Integration of Ipac module with EPICS
- Issues encountered
- Conclusions and Discussion

Latest Development Hardware Components

N.A.T NAT-MCH-PHYS uTCA MCH

Concurrent Technologies Intel I7 uTCA CPU

Schroff 12-slot mTCA.4 chassis (Model 11890-035)

Vadatech uTCA PMC carrier

Struck SIS8300 uTCA digitizer

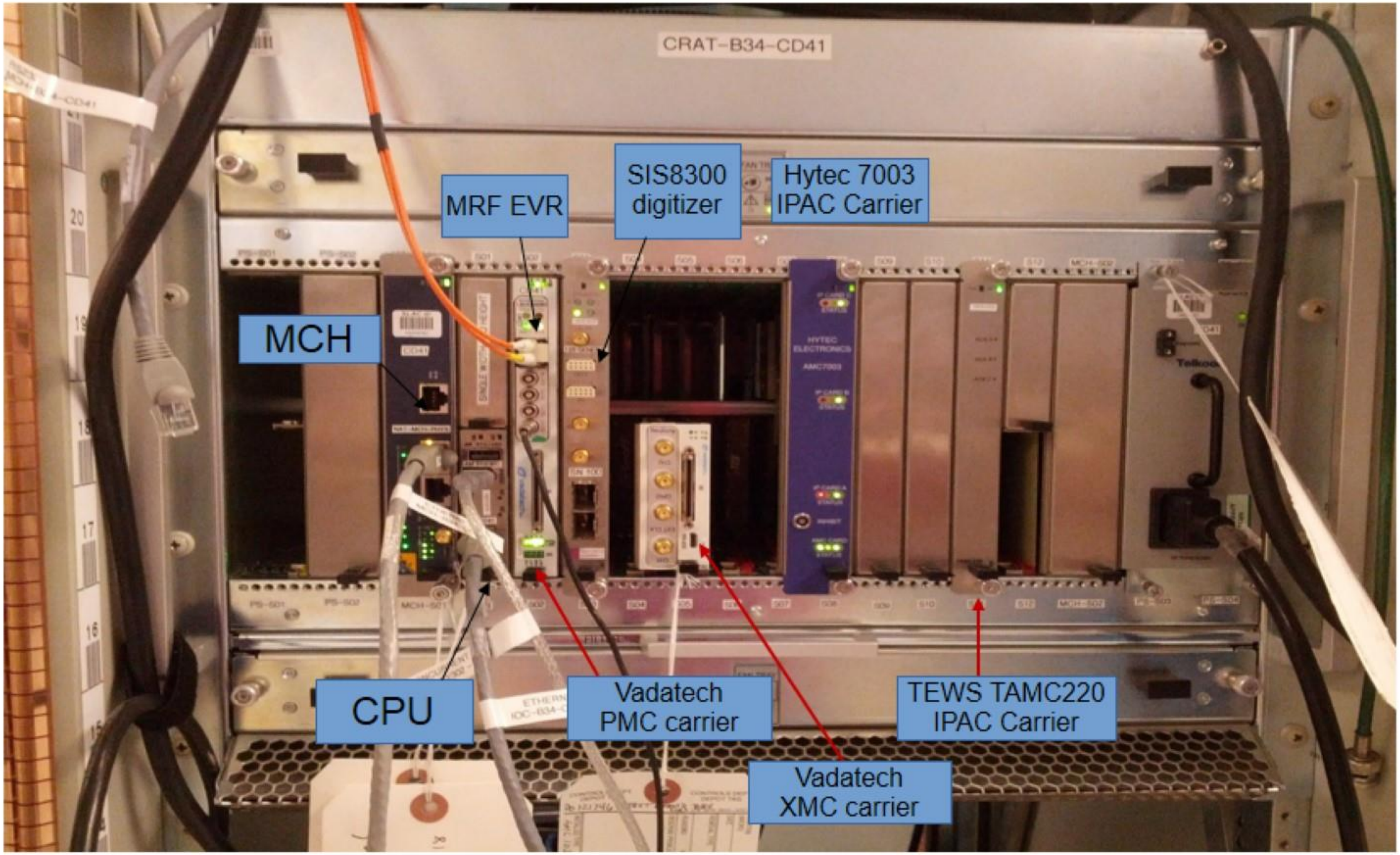
Hytec 7003 uTCA IPAC carrier w/ Hytec 7308 uTCA RTM

- Acromag IP330 analog input
- Acromag IP440 binary input
- Hytec 8414 analogue to digital input
- Hytec 8601 stepper controller

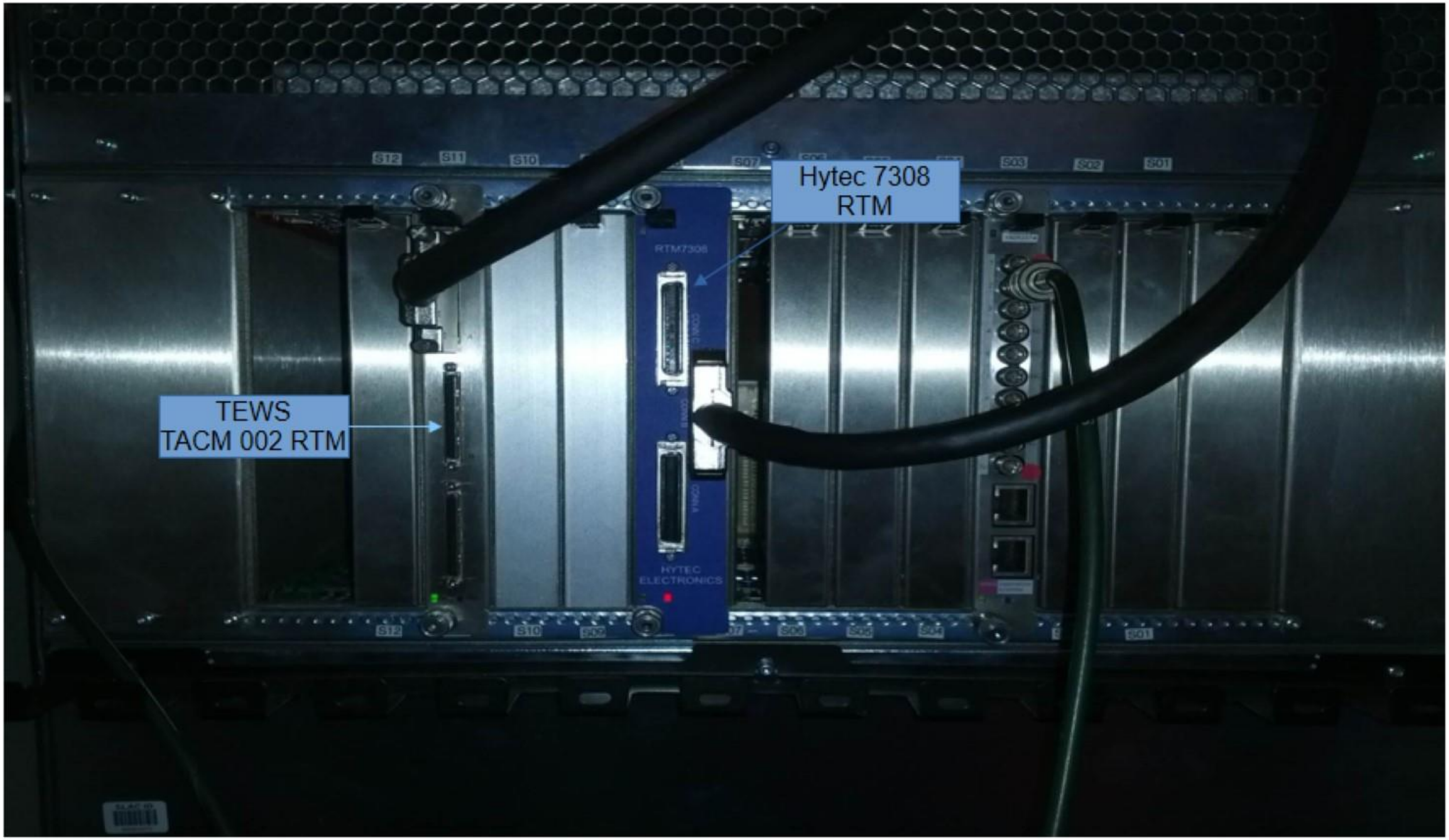
TEWS TAMC 220 uTCA IPAC carrier w/TEWS TAMC 002 uTCA RTM

- Acromag IP231 analog output
- Acromag IP445 binary output

Current Development Glance



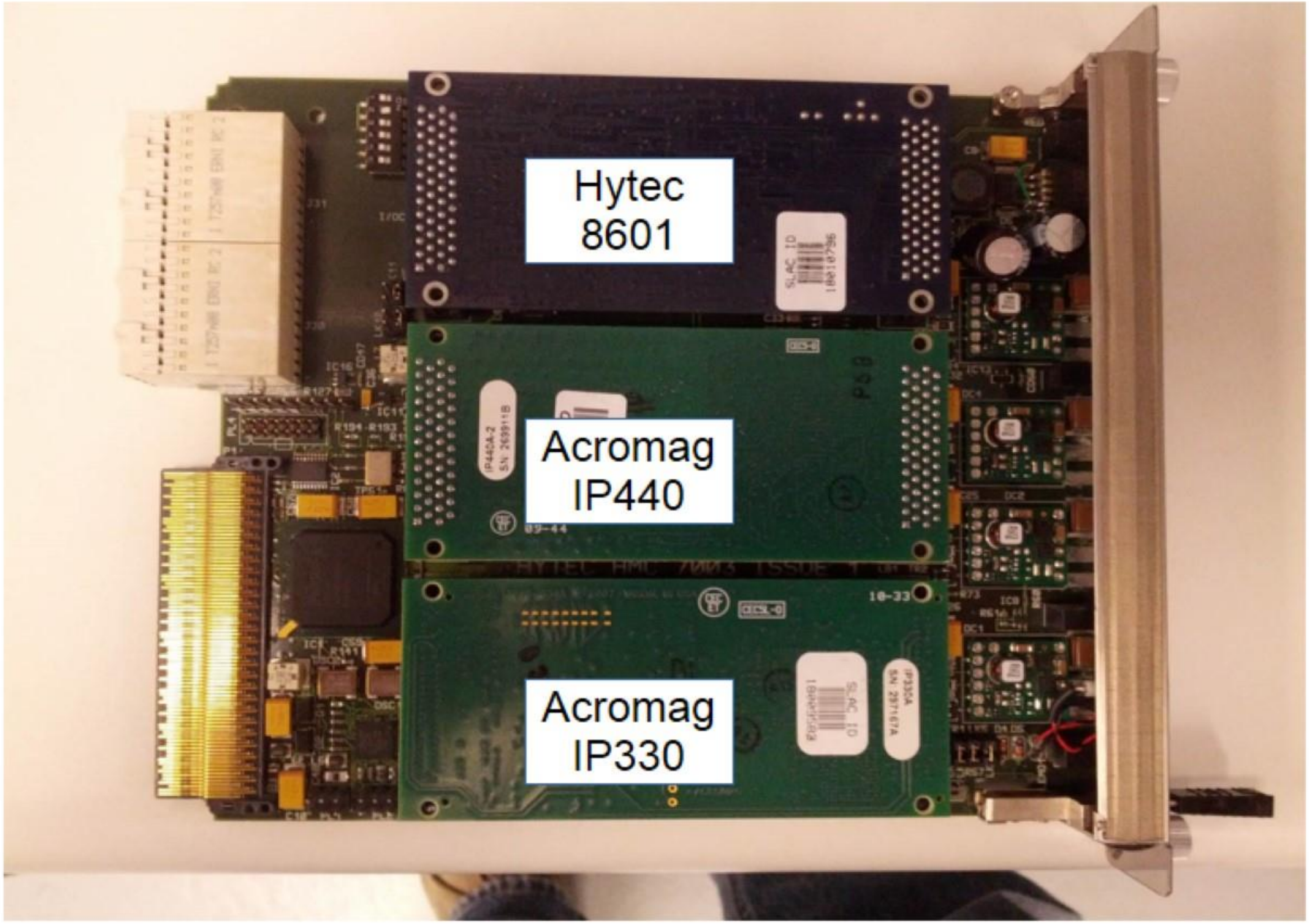
uTCA Crate Rear View



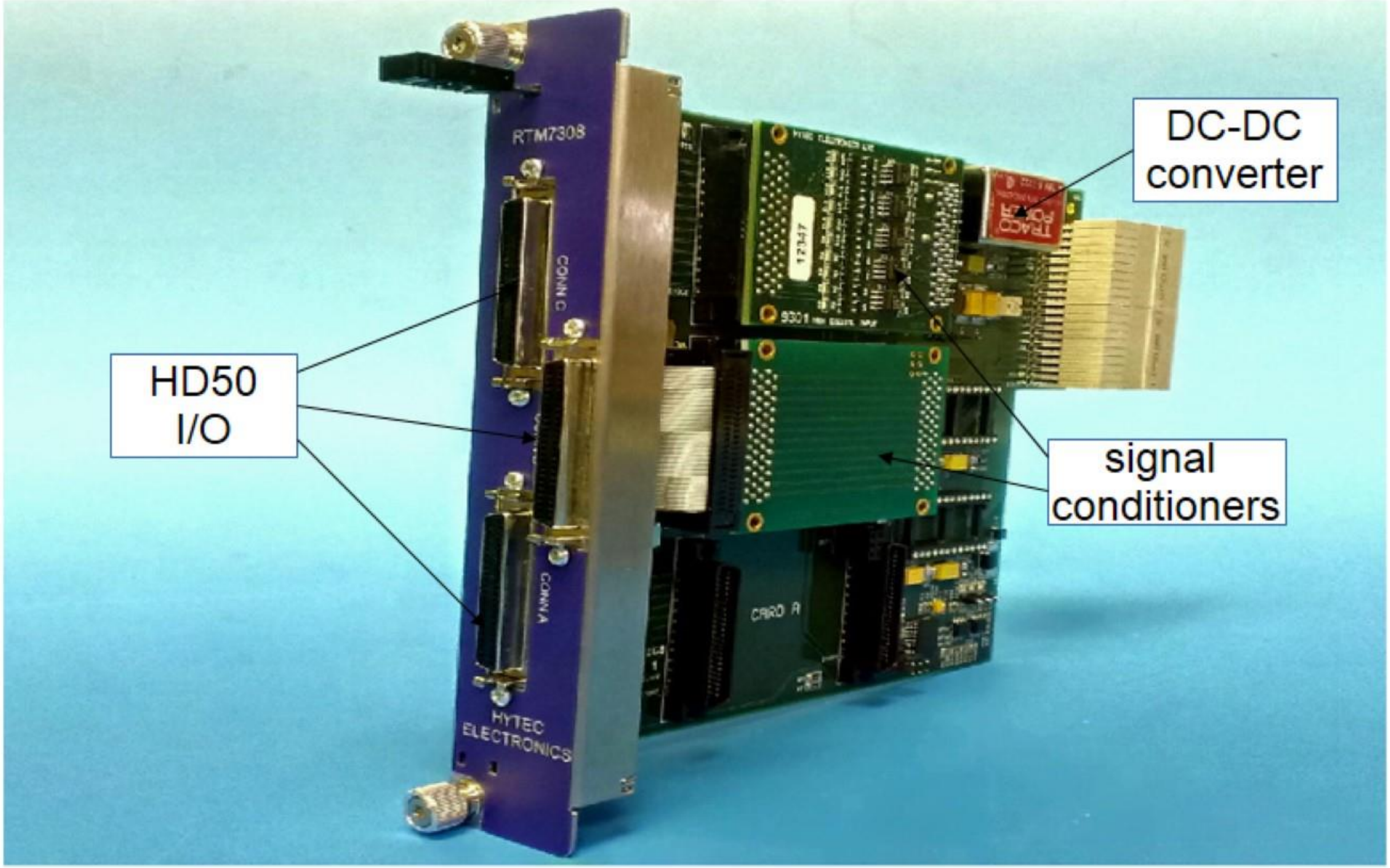
Hytec 7308 RTM

TEWS TACM 002 RTM

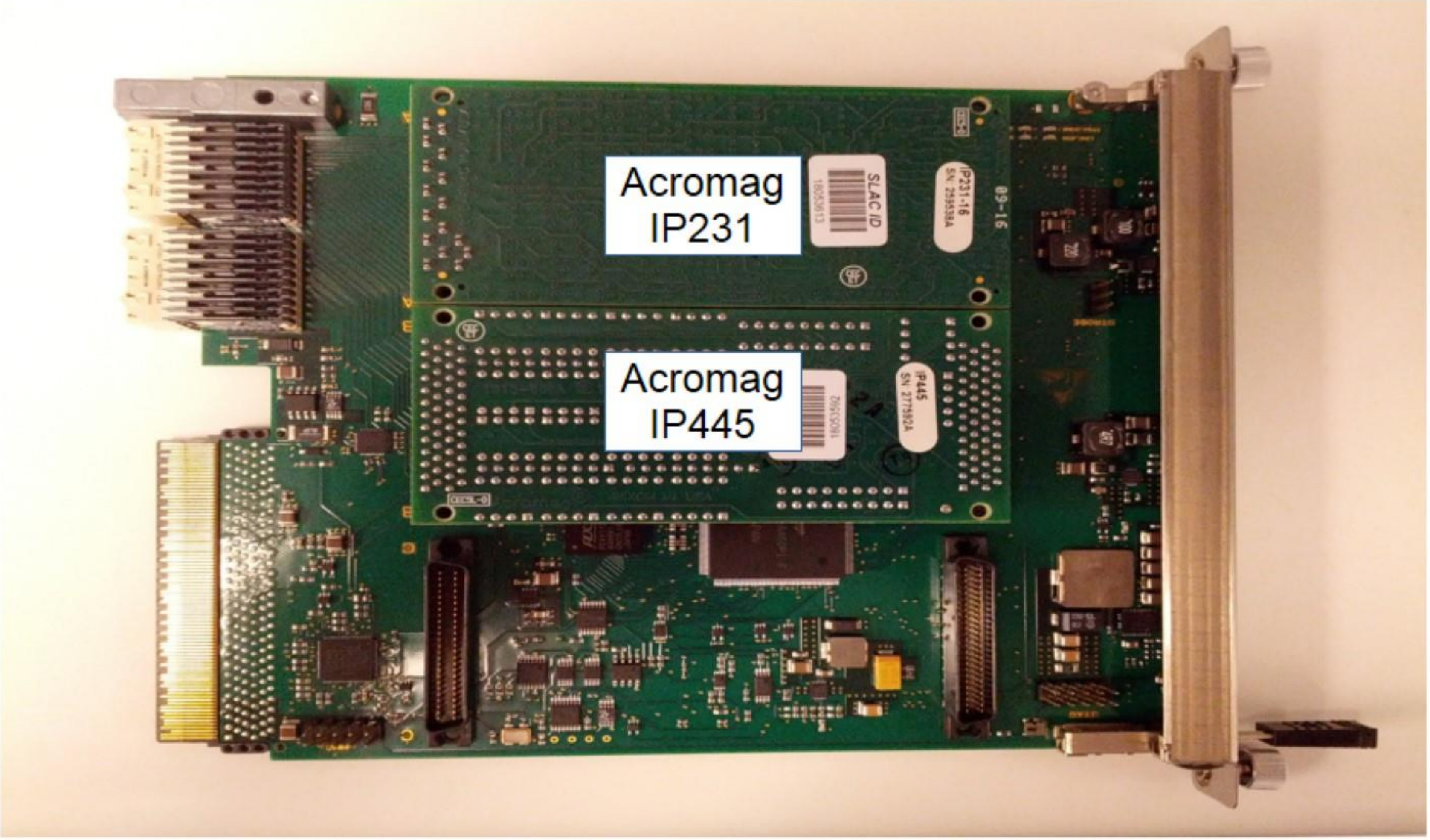
Hytec 7003 uTCA.4 IP Carrier



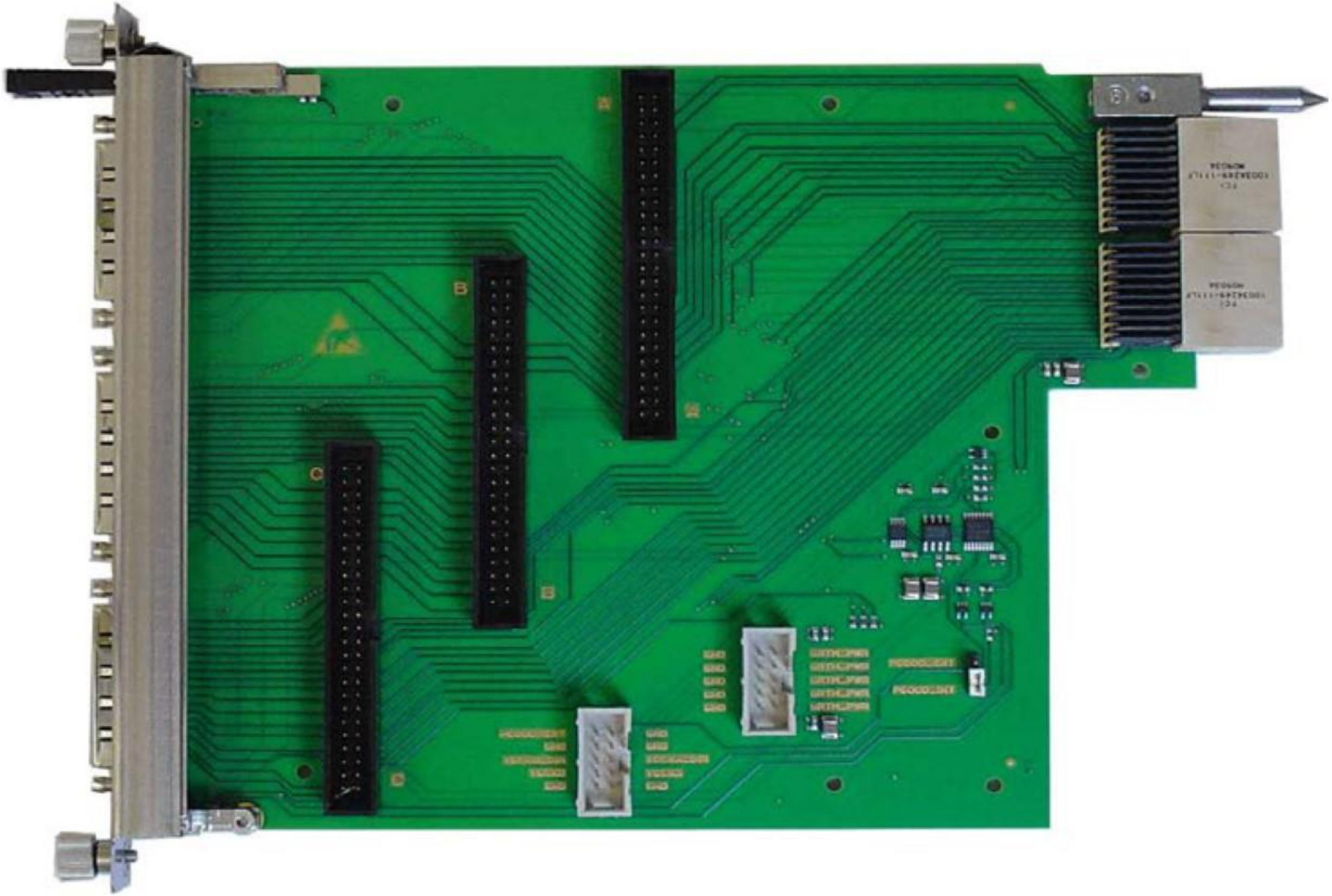
Hytec 7308 uTCA.4 RTM



TEWS TAMC220 IPAC Carrier



TEWS TAMC002 RTM



EPICS IOC -- Software Components

LinuxRT 3.2.13

EPICS 3.14.12

Standard SLAC MRF EVR driver

sis8300 driver from Struck with modifications (SLAC). The digitizer fits the generic transient record (GTR) framework which is very good.

Asyn-based Hytec IPAC carrier driver supplied by Hytec

- Uses small kernel driver supplied by Hytec

Asyn-based TEWS IPAC carrier driver written by Richard Dabney

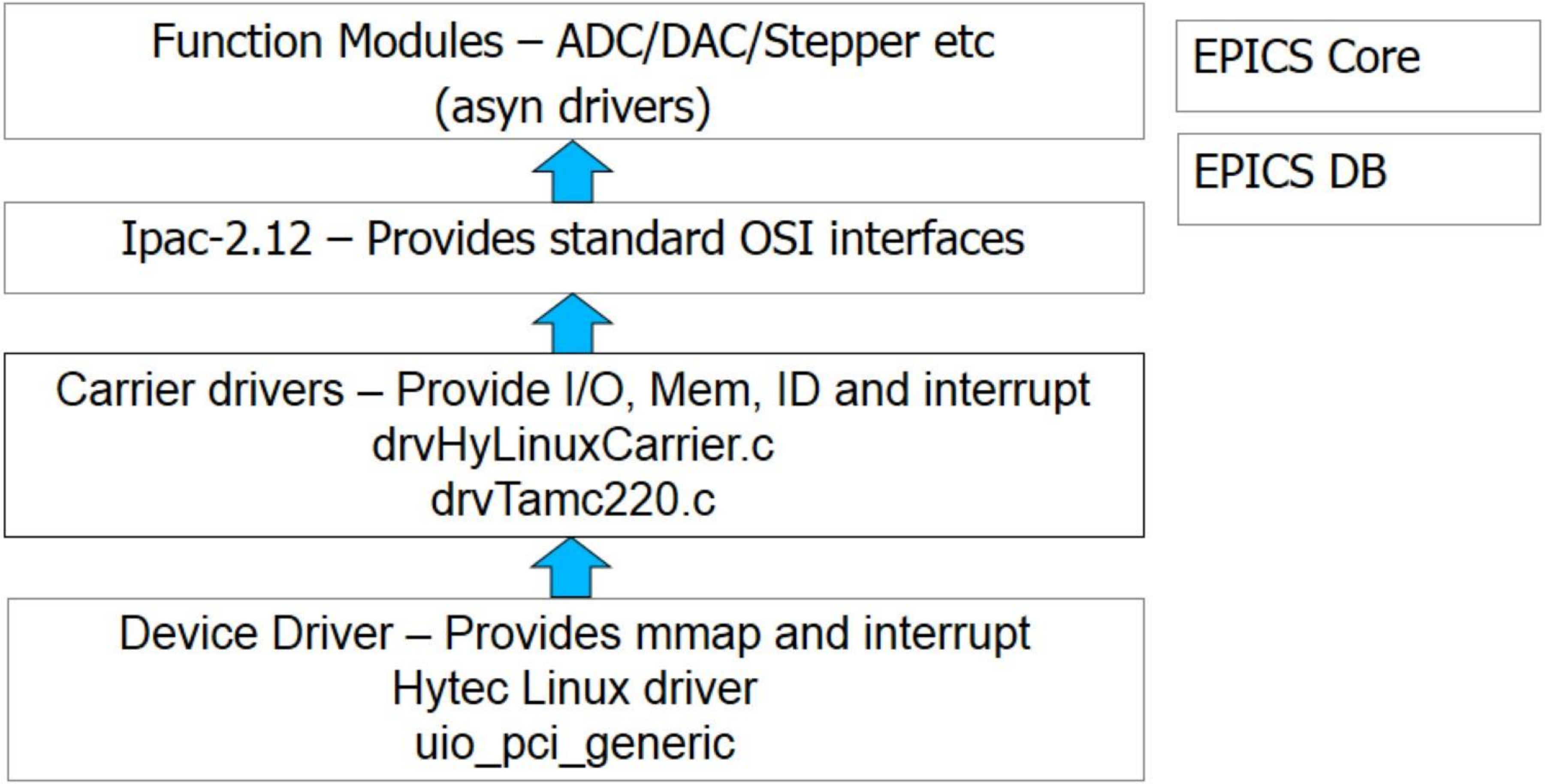
- Uses Linux driver uio_pci_generic

Asyn-based Acromag IPAC modules from SLAC CVS MAIN_TRUNK -- These drivers come originate from the EPICS community (via Mark Rivers)

Hytec 8601 IPAC stepper driver from the standard motor record package

Hytec 8414 IPAC ADC asyn driver from Hytec

Carrier Module Integration – Ipac + Carrier Drivers



Function Modules – ADC/DAC/Stepper etc
(asyn drivers)

Ipac-2.12 – Provides standard OSI interfaces

Carrier drivers – Provide I/O, Mem, ID and interrupt
drvHyLinuxCarrier.c
drvTamc220.c

Device Driver – Provides mmap and interrupt
Hytec Linux driver
uio_pci_generic

EPICS Core

EPICS DB

Timing Module – MRF EVR + Vadatech PMC carrier

- Timing module uses MRF PMC EVR, the same module we use for VME system
- The carrier is a Vadatech PMC carrier for uTCA.4
- This provides four triggers from the front panel
- SLAC designed a breakout chassis that attaches to the Vadatach front panel SCSI connector on the front panel to provide more triggers.
- The software as mentioned before uses SLAC MRF EVR driver
- Also, we want to try MRF uTCA.4 native AMC module and RTM and the Hytec uTCA.4 PMC/XMC carrier with RTM designed for timing triggers.

Issues Encountered. Some for Discussion

- Byte access -- Hytec 7003 carrier initially had no support for Byte access which is needed for IP cards like Acromag. Solved by firmware upgrade
- Both carriers at the moment use Ipac + kernel device drivers to talk to the deivces
- Hytec carriers use onboard dip switch to set up 'unique' ID and EPICS ipac module & device driver for identifying individual PCIe device. This may also be achieved by devlib2, so as the driver for Tews module.
- uTCA Hop swap – capable so as Linux. Caveat: software drivers need to be carefully designed. Till has detail discussion in "Linux (3.13) Hotswap Notes".
- Endian Issue – a long discussed issue. So far, all systems appear to work fine but there are cases may fail. Could use something like ipmOut16(carrier, card, offset, value) instead of 'baseAddr' however need changes to existing drivers. Till also has detail discussion in "Ipac Carrier Driver Issues"
- The ipmCheck miss reports in some cases. This is known issue and Andrew Johnson is aware
- 64bit Linux issue – last param in ipmIntConnect in Ipac passes as int that cannot be passed with pointers. Mark Rivers has a solution by module instant table
- Higher Linux kernel distro (>2.6.34?) cannot access I/O space via WORD, but can via BYTE

Conclusions and Discussion

- The SLAC uTCA system works well so far with different modules from different manufacturers for ADCs, digital IOs, timing modules, digitizers and step motor controller etc.

- Still more work to do

- Next move for discussion –:

- We know it is hard, however, would it possible to refactor the Ipac module to suit all endian cases?
- We may merge to devlib2 for drivers. The main purpose is to have the driver support merged into the main trunk.

- **Thank You!**