ITER Fast controller EPICS support

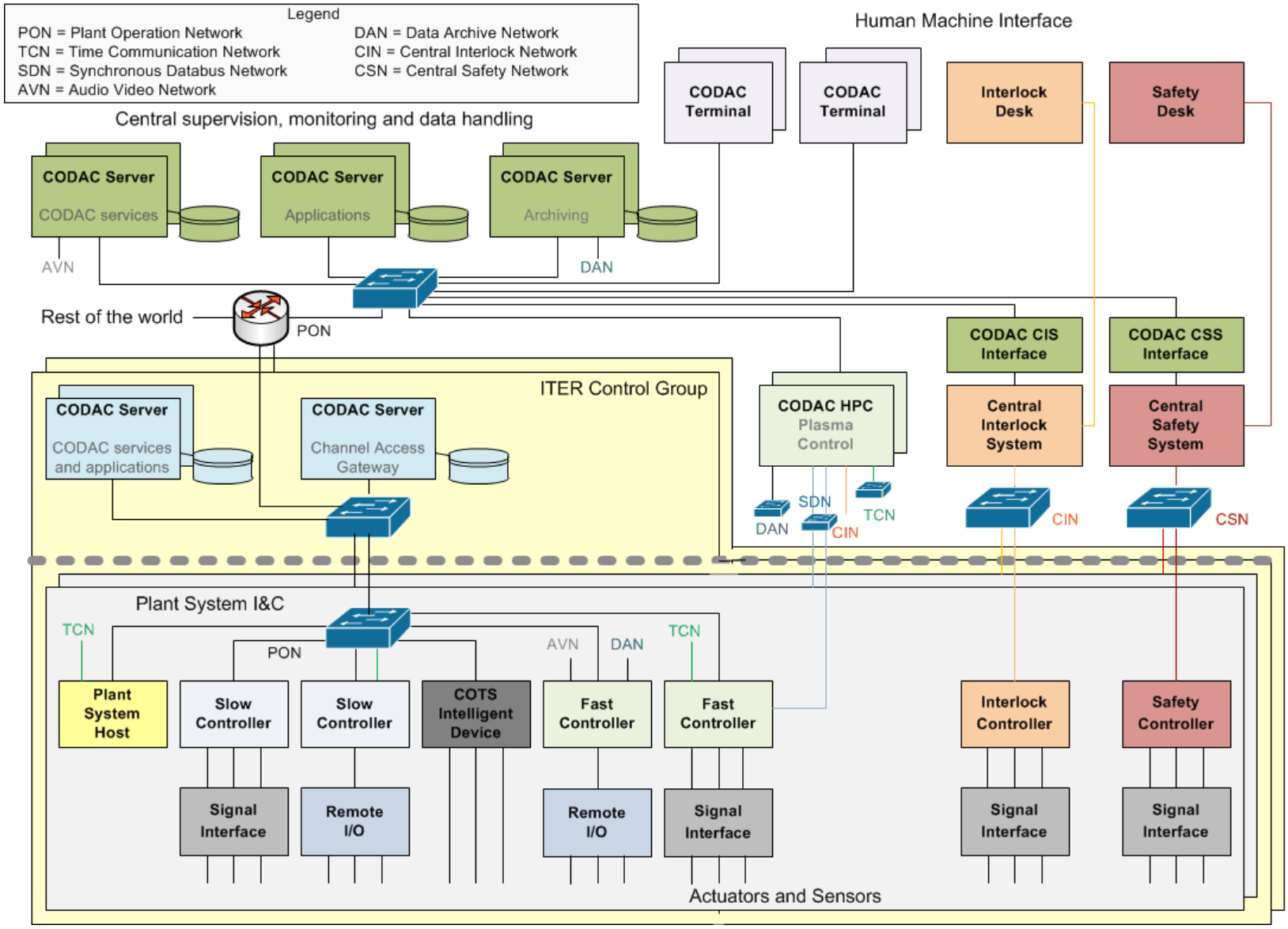
Vishnu Patel
ITER Organization
Control System Division

Content

Goal for ITER CODAC (Control, Data Access and Communication)

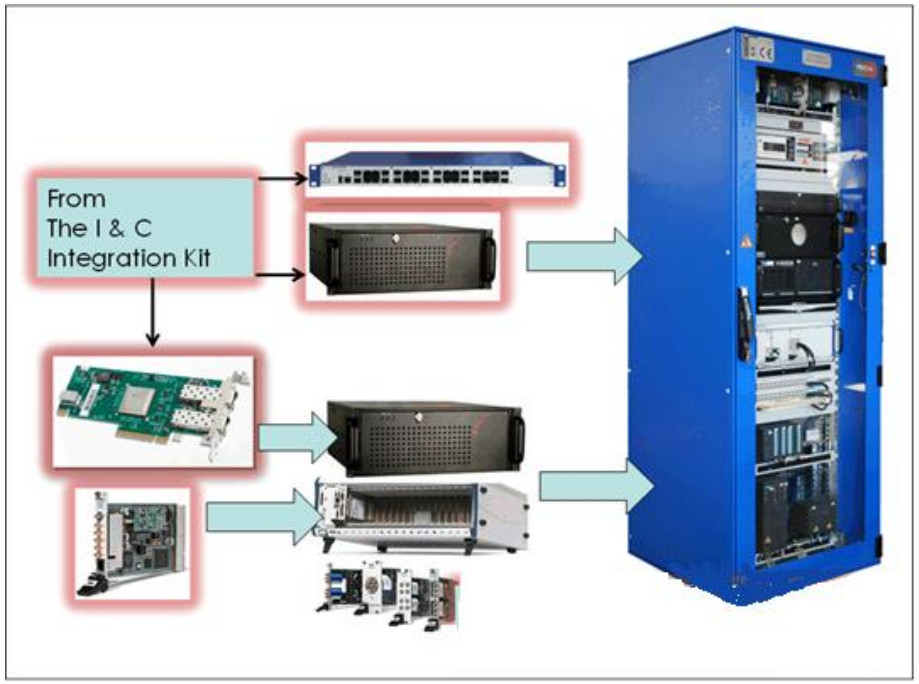
- Ensure all ITER Plant Systems Instrumentation & Control are designed, implemented and integrated such that ITER can be operated as a fully **integrated** and automated system.
- Higher reliability and availability.

Physical Architecture



What constitutes a fast controller?

- Industrial computer
- I/O chassis as required
- Interlinked with a PCI Express bus extension
- In an I&C cubicle



What constitutes a fast controller?

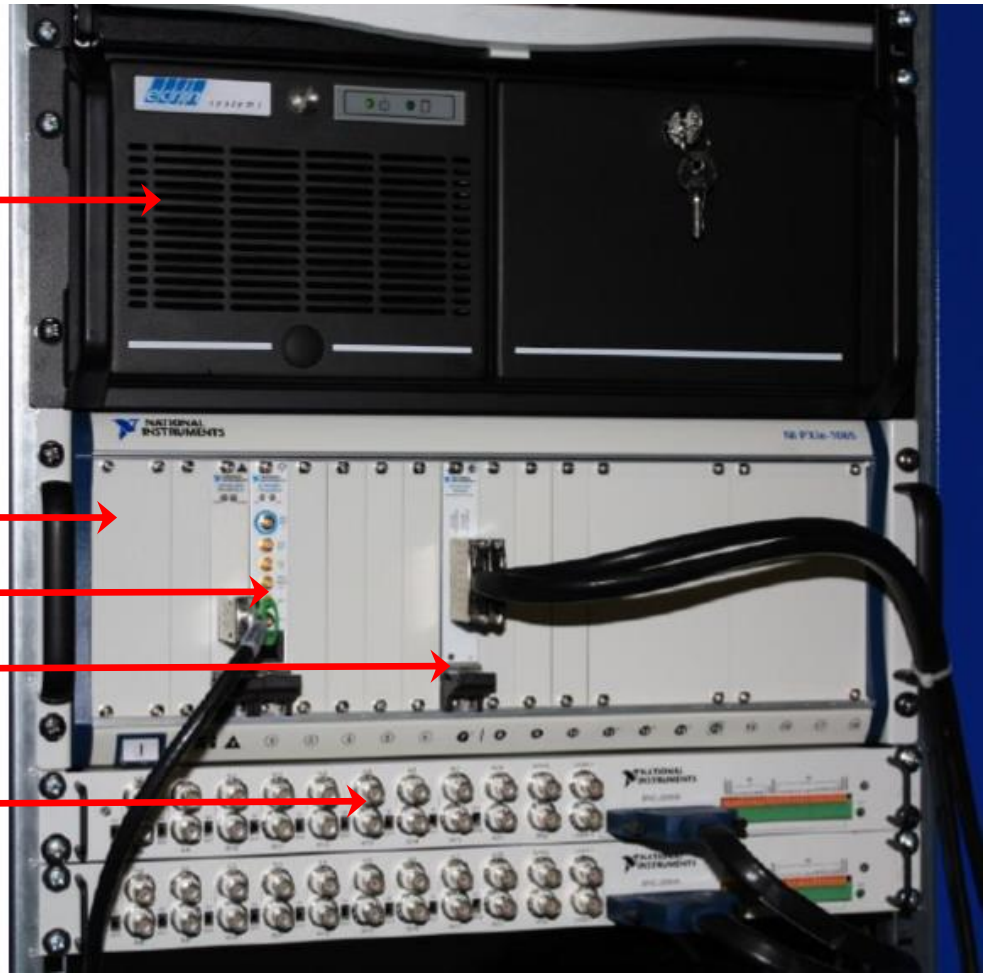
The industrial computer

I/O chassis (a PXIe rack)

TCN timing card NI PXI-6683H

A/D I/O card NI PXI-6259

Signal terminal block (patch panel)



Standard I/O Boards

N.I. PXI-6682/PXI-6683H : Synchronization and timing (IEEE1588-2008 / TCN)

N.I. PXI-6259: multi-function data acquisition

16bits analog input channels (16 Diff. /32 Single Ended) (1.25MS/s)
16bits analog output channels (4) (2.86 MS/s)
Digital input/output channels (48)

N.I. PXIe-6368: High Performance multi-function data acquisition

16bits analog input channels (16 Diff.) (Simultaneous, 2MS/s)
16bits analog output channels (4) (3.3 MS/s)
Digital input/output channels (48)

N.I. PXI-6528 : Digital I/O

24 optically isolated input channels
24 solid-state relay output channels

N.I. PXIe-7961R : FlexRIO

Virtex-5 SX50T, 8160 FPGA Slices, 16 DMA Channels

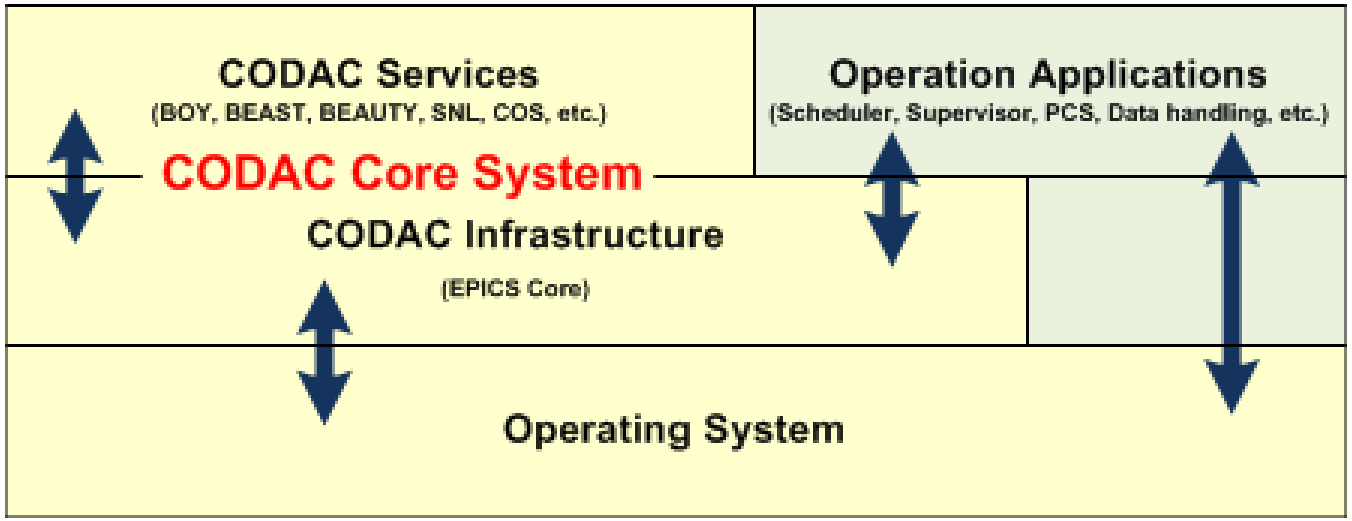
N.I. PXIe-7966R : FlexRIO

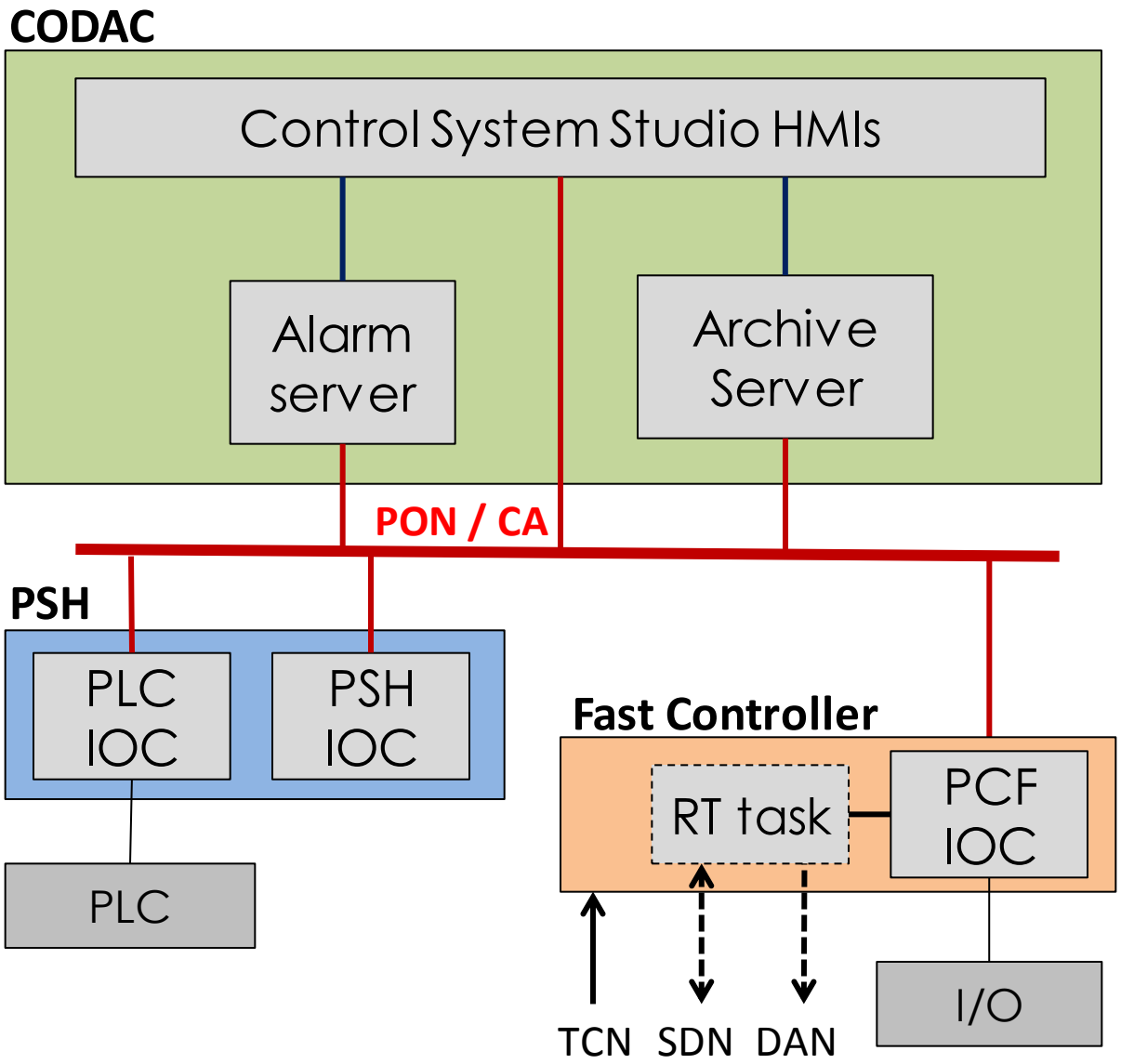
Virtex-5 SX95T , 14720 FPGA Slices, 512MB DRAM, 16 DMA Channels 8160

The Supported boards are listed in the ITER Catalog of I&C products - Fast Controllers (345X28).

Software Architecture

CODAC Core System
Operation Applications





CODAC Terminal
- Operator Interface (OPI)
- Alarm views
- Data plots

CODAC Server
- Alarm handling
- Archiving

PSH:
- I&C monitoring
- PLCs Gateway
- I&C coordination

Fast Controller:
- I/O interface
- HPN interfaces: TCN, SDN, DAN
- RT control

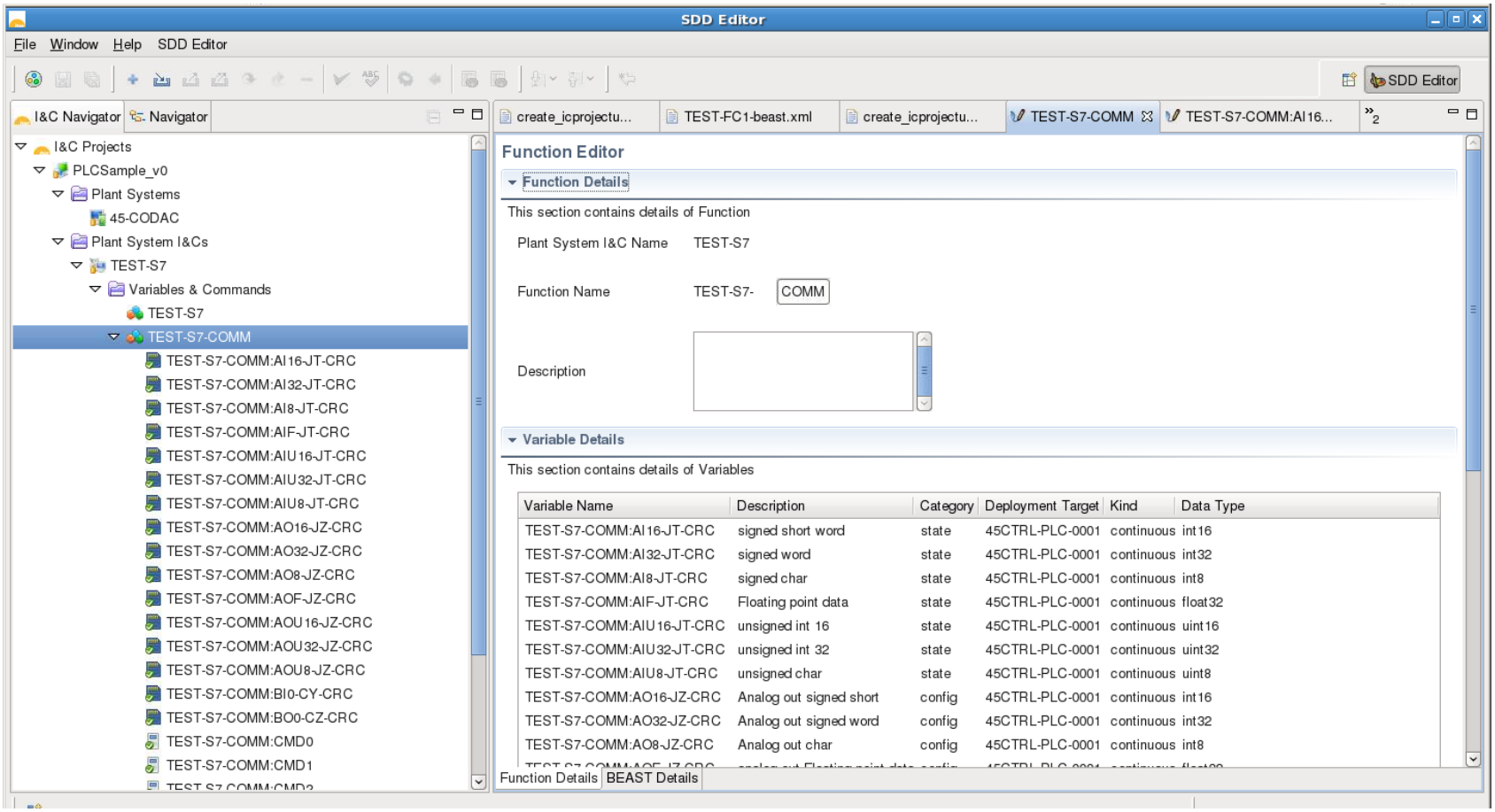
The Operating System(RHEL) & EPICS in CCS

- ITER selected **Red Hat Enterprise Linux** for the x86-64 architecture (RHEL x86_64) & open source control system EPICS
  - CCS 4.3 -> RHEL 6.3 -> EPICS 3.14 , asynDriver 4.21
  - CCS 5.0 -> RHEL 6.5 -> EPICS 3.15 , asynDriver 4.23 (Feb 2015)
- For fast feedback control a real-time enabled version of RHEL, **RHEL MRG-R**, is deployed.
  - Optimized kernel for real-time applications
  - More deterministic latencies / reduced jitter

Standard Software Support

- CODAC Core System includes the software for ITER standard I/O boards
  - Linux driver
  - EPICS integration (EPICS device support)
  - SDD integration (configuration via SDD tools)

Self-Description Data Definition (SDD) Editor



EPICS device support

Basic EPICS Device Driver

. PXI-6259: multi-function data acquisition.

asynDriver based Device Driver

PXI-6682 & PXI-6683h : Synchronization and timing.
PXIe-6368: High Performance multi-function data acquisition.
PXI-6528 : Digital I/O.

Future device Support

FlexRIO boards (PXIe-7961R/PXIe-7966R).
cRIO.

PXI 6259 DAQ Board

PXI 6259 DAQ Board

AI Channels

Channel : 32 Single-Ended Channels
/16 Differential Channels
Sample rate : 1.25Ms/s single channels
1Ms/s multi channels (all)

AO Channels

Channel : 4
Update rate :
1.25Ms/s per each channel (4 Ch)
2.86MS/s (1 Ch)

Digital I/O

Bidirectional Channel : 48

PXI 6259 Board functions

- General
  - ✓ Device Information
  - ✓ Software version
- DAQ mode Support
  - ✓ Software timed acquisition
  - ✓ Hardware timed acquisition
    - ➢ Finite sample mode
    - ➢ Continuous mode
- Clock Settings
  - ✓ Sampling rate
  - ✓ Clock Source
  - ✓ Clock divider
- Triggering
  - ✓ Set trigger
    - ➢ Software
    - ➢ Hardware (Start, Stop & Pause Trigger)
  - ✓ Pre & Post samples

PXI 6259 Board functions

- DIO configuration
- Analog Output configuration.
- General purpose counter
- Signal routing

PXI 6259 Templates

- Board level EPICS Template
  - ❖ EPICS PV Prefix :- $(CBS1)-$(CBS2)-HWCF:6259-$(BOARDTYPEIDX)-

Function	PV Suffix	Record type	Description
num_samples	NS	longout	Set Number of sample to read
sample_rate	SR	longout	Set sampling rate of AI
BoardStatus	BS	mbbi	Read Board status
switch_acquisition	ACQ	bo	Start/Stop data acquisition
daq_mode	MODE	bo	Set Continuous or Finite AI acquisition mode
pretrig_samples	PRET	longout	Set Pre-trigger sample for AI acquisition
posttrig_samples	POST	longout	Set Post-trigger sample for AI acquisition
ai_conf	CONF	bo	Load/Reset AI configuration

- Channel level EPICS Template.
  - ❖ EPICS PV Prefix :- $(CBS1)-$(CBS2)-HWCF:6259-$(BOARDTYPEIDX)-AI$(CHIDX)-

Function	PV Suffix	Record type	Description
samples_read	SR	longin	Read Analog input of selected channel
channel_type	CT	mbbo	Set Channel Type (NRSE/RSE/DIFFERENTIAL

Timing Boards
(PXI-6682 & PXI-6683H)

Timing Boards Functions

- Receive ITER time
- Synchronization of system time with Board time
- Generate Future time events, Pulse and Clocks
  - Front panel connector
  - PXI backplane lines
  - Synchronized event generation
- Signal routing
- Clock routing
- Accurate Time-stamping of event
  - External events (Front panel connector & Backplane lines)
- Pulse sequence Programming on the selected terminal.

Timing Boards Templates

- Clock Routing
  - EPICS PV Prefix :- $(CBS1)-$(CBS2)-HWCF:$(SHORTBOARDTYPE)-$(BOARDTYPEIDX)-
  -
  -

Function	PV Suffix	Record type	Description
clockSource	BKPCMON	mbbi	Read Backplane clock source
clockSource	BKPSRC	mbbo	Connect the selected source clock to the Backplane clock
clockSource	CLKOMON	mbbi	Read Clock-out clock source
clockSource	CLKOSRC	mbbo	Connect the selected source clock to the to the Clock-out
clockSource	TMKCMON	mbbi	Read Time keeper clock source
clockSource	TMCSRC	mbbo	Connect the selected source clock to the to the Time keeper clock
checkFteLevels	LVL_ERRS	longin	Check FTE level sequence error
deviceSerialNumber	SERIAL	loingin	Get device serial number
deviceStatus	STAT	mbbi	Get device status
syncStatus	SYNC	mbbi	Get device synchronization status
resetCard	RESET	bo	Reset the Device

Timing Boards Templates

- Device Terminals
  - EPICS PV Prefix :- $(CBS1)-$(CBS2)-HWCF:$(SHORTBOARDTYPE)-$(BOARDTYPEIDX)T$(CHIDX)-
  - Pulse Generation Records
  - Clock Generation Records

Function	PV Suffix	Record type	Description
clockEnable	CLOCKE	bo	Enable or disable clock generation
clockStatus	CLOCKM	bi	Returns whether clock is enabled
clockStartSec	CLOCKSS	ao	Set second part of the clock start time
clockStartNsec	CLOCKSN	longout	Set nano second part of the clock start time
clockEndSec	CLOCKES	ao	Set second part of the clock end time
clockEndNsec	CLOCKEN	longout	Set nano second part of the clock end time
clockPeriod	CLOCKP	ao	Set Clock period
clockDuty	CLOCKD	ao	Set Clock duty cycle
clockGenerate	CLOCK	bo	Generate clock with configured time

Timing Boards Templates

- Device Terminals
  - EPICS PV Prefix :- $(CBS1)-$(CBS2)-HWCF:$(SHORTBOARDTYPE)-$(BOARDTYPEIDX)T$(CHIDX)-
  - Pulse Generation Records
  - Clock Generation Records

Function	PV Suffix	Record type	Description
clockEnable	CLOCKE	bo	Enable or disable clock generation
clockStatus	CLOCKM	bi	Returns whether clock is enabled
clockStartSec	CLOCKSS	ao	Set second part of the clock start time
clockStartNsec	CLOCKSN	longout	Set nano second part of the clock start time
clockEndSec	CLOCKES	ao	Set second part of the clock end time
clockEndNsec	CLOCKEN	longout	Set nano second part of the clock end time
clockPeriod	CLOCKP	ao	Set Clock period
clockDuty	CLOCKD	ao	Set Clock duty cycle
clockGenerate	CLOCK	bo	Generate clock with configured time

EPICS interface for pulse time

```
# Terminal status

record(ao, "$(CBS1)-$(CBS2)-HWCF:$(EVENTNAME)-O") {
  field(DESC, "Set FTE Origin time [s]")
  field(DTYP, "asynFloat64")
  field(OUT, "@asyn(S$(MODULEIDX),$(CHIDX))fteTimingOrigin,$(EVENTID)")
  field(EGU, "seconds")
  field(LOPR, "0")
  field(HOPR, "4503599627370496")
  field(VAL, "$(FTE_O_VAL)")
  field(PINI, "YES")
}

record(ao, "$(CBS1)-$(CBS2)-HWCF:$(EVENTNAME)-D") {
  field(DESC, "Set FTE Delay time [s]")
  field(DTYP, "asynFloat64")
  field(OUT, "@asyn(S$(MODULEIDX),$(CHIDX))fteTimingDelay,$(EVENTID)")
  field(EGU, "seconds")
  field(LOPR, "0")
  field(HOPR, "4503599627370496")
  field(VAL, "$(FTE_D_VAL)")
  field(PINI, "YES")
}

record(bo, "$(CBS1)-$(CBS2)-HWCF:$(EVENTNAME)-L") {
  field(DESC, "Set FTE Level")
  field(DTYP, "asynInt32")
  field(OUT, "@asyn(S$(MODULEIDX),$(CHIDX))fteTimingLevel,$(EVENTID)")
  field(ZNAM, "Low")
  field(ONAM, "High")
  field(VAL, "$(FTE_L_VAL)")
}


record(bo, "$(CBS1)-$(CBS2)-HWCF:$(EVENTNAME)-E") {
  field(DESC,  "Generate Timing FTE")
  field(DTYP, "asynInt32")
  field(OUT, "@asyn(S$(MODULEIDX),$(CHIDX))fteTimingEnable,$(EVENTID)")
  field(ZNAM, "0")
  field(ONAM, "1")
  field(VAL, "$(FTE_E_VAL)")
}
~
"nisync_timing_program.template" [readonly] 43L, 1181C
```

One timing is defined by the following PVs:

Control of the time origin (suffix: "**-O**"), expressed as a double and representing an absolute time in seconds.

Control of the time offset from the origin , called delay (suffix: "**-D**"), expressed as a double and representing a time in seconds with the resolution of a nano-second and with an absolute limit of one day.

Control of the terminal line level (suffix: "**-L**") to High or Low.

An enable/disable flag (suffix: "**-E**") for adding/removing the timing to the list of FTEs.

```
TEST-SYNC-HWCF:POWERON-O
TEST-SYNC-HWCF:POWERON-D
TEST-SYNC-HWCF:POWERON-L
TEST-SYNC-HWCF:POWERON-E
```

PXIe-6368 DAQ Board

PXIe-6368 DAQ Board

- 16 simultaneous analog inputs at 2MS/s/ch , 32MS/s total AI throughput (16 bit resolution)
- Four analog outputs, 3.33 MS/s (16 bit resolution)
- 48 digital I/O lines
  - 32 hardware-timed up to 10MHz
  - 16 PFI
- Four 32 bit counters
- Analog and digital triggering
- Acquisition mode.
  - Continuous
  - Finite sample
  - Reference triggered

PXIe-6368 DAQ Board Function

- All the Functions for PXI-6259
- Retriggerable acquisition mode
- Board level and Channel level EPICS Template.
- waveform and ai record support for Analog input.
- DIO configuration with EPICS PVs.
- Analog Output configuration with EPICS PVs.

PXIe 6368 Templates

- Board level EPICS Template
  - EPICS PV Prefix :- $(CBS1)-$(CBS2)-HWCF:6368-$(BOARDTYPEIDX)-

Function	PV Suffix	Record type	Description
change_ai_mode	CHAIMO	mbbo	Change the AI acquisition mode
reset_load_ai	LDAICF	bo	Load/Reset AI configuration
no_of_samples	AISMPL	longout	Number of AI sample to be acquire
pulse_ai	AIPU	mbbo	Generate Software pulse for AI
sampling_rate_ai	AISR	longout	Set AI sampling rate
start_stop_ai	AISTSP	bo	Start/Stop Analog acquisition
timebase_ai	AITB	longout	Set AI timebase source

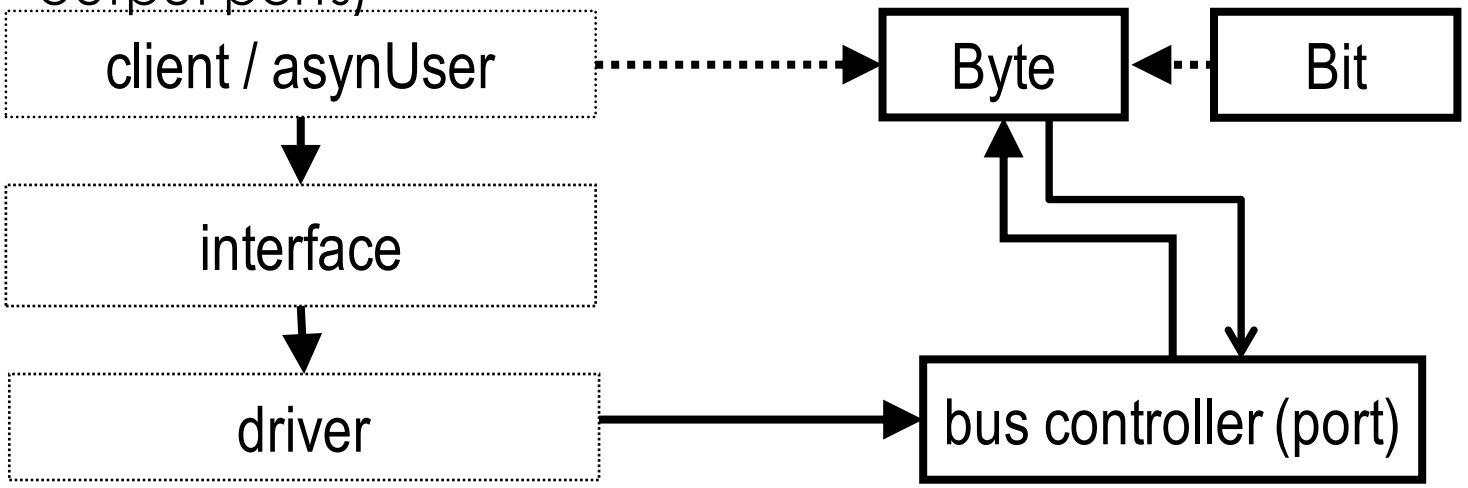
- Channel level EPICS Template.
  - EPICS PV Prefix :- $(CBS1)-$(CBS2)-HWCF:6368-$(BOARDTYPEIDX)-AI$(CHIDX)-

Function	PV Suffix	Record type	Description
Read_ai	AIW$(CHIDX)	waveform	Read AI samples of channel $(CHIDX)

PXI-6528 DIO Board

PXI-6528 DIO Board Functions

- Read Device information
- Bit and Byte write/read on hardware port (3 input and 3 output ports)



- Signal routine.
- Watchdog configuration
- Level change on watchdog
- Level change detection
- Input port filter
- Output port Power up state

PXI-6528 Templates

- EPICS PV Prefix :- $(CBS1)-$(CBS2)-HWCF:6528-$(BOARDTYPEIDX)-

Function	PV Suffix	Record type	Description
noboard	BS	bi	Give board presence
setfilterinterval	FILTINT	longout	Configure the internal filter interval
setrtsiinputroute	RTSIINPROUT	longout	Select the RTSI input port lines
setrtsioutputroute	RTSIOUTROUT	longout	Select the RTSI Output port lines
setrtsiwatchdogtrigger	RTSIWDGTRIG	longout	Configure an RTSI line to act as a hardware trigger for the watchdog timer.
setwatchdogtimeout	WDGTOUT	ao	Set watchdog timeout
setportpowerup	POWERP3	longout	Set port3 power up state
setportpowerup	POWERP4	longout	Set port4 power up state
setportpowerup	POWERP5	longout	Set port5 power up state

Management Tools

Tool	Scope	Reference
Subversion	Software source code evolution	HTTPS://SVNPUB.ITER.ORG/CODAC
Bugzilla	Issue tracker	HTTPS://BUGZILLA.ITER.ORG/CODAC
Central SDD	I&C production projects	HTTPS://SDD.ITER.ORG
CLM	Inventory control	HTTPS://PSP.ITER.ORG/PSPROFILE/PAGE/INVENTORY/

Conclusion

- Developments and distribution of EPICS Device Drivers are stable.
  - PXI-6259
  - Timing Boards (PXI-6682 & PXI-6683H
  - PXI-6528
- ALL of the requirement for PXI-6259, Timing boards (PXI-6682 & PXI-6683H) and PXI-6528 are completed
- PXIe-6368 EPICS device Driver is under progress.
- More efforts will be applied in improvement.

Thank you