EPICS-based control system for compact-ERL and iBNCT

2017/05/16
Takashi OBINA

Control/Beam Diagnostics Group of Photon Factory and PF-AR
KEK, High Energy Accelerator Research Organization
• I would like to talk about the control system of two accelerators:
  – Compact Energy Recovery Linac (cERL) at KEK Tsukuba
  – Ibaraki Boron Neutron Capture Therapy (iBNCT) at Tokai (near J-PARC)

• The reason for picking up the two accelerators are:
  – They use “EPICS”, of course
  – Both accelerators is (relatively) compact
  – Limited human resource
  – Both facilities uses similar hardware (Field Bus, console, server, etc)

  – I want to share the lessons learned with EPICS community...
1. Introduction

2. Outline of cERL at KEK

3. Outline of iBNCT at Tokai

4. Commissioning, Tuning, Operation
   - Various tuning panel
   - CSS as an operation manual (procedure)
   - Software for rapid prototype
   - Hardware example: VME-Master
Large Accelerators:

- Linac
- KEKB: HER, LER (C = 3 km)

Synchrotron Radiation Facility
- PF-Ring: 2.5 GeV (C = 187 m)
- PF-AR: 6.5 GeV (C = 640 m)

(Relatively) compact accelerator:
R&D Machine
compact ERL (∼20 MeV, C = 90 m)
What is ERL?

Keyword: Linac-based, High Average Current, Brilliant electron source
Application: Future Light Source, Electron Cooling, EUV-Lithography, etc.

(5GeV x 100mA = 500MW)

High brilliant electron gun

Super conducting cavity for Acceleration and deceleration

Beam Dump
(10MeV x 100mA = 1MW)
Parameters of the cERL

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Design</th>
<th>In operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beam energy $E$</td>
<td>35 MeV</td>
<td>20 MeV</td>
</tr>
<tr>
<td>Injector energy $E_{\text{inj}}$</td>
<td>5 MeV</td>
<td>2.9 - 6 MeV</td>
</tr>
<tr>
<td>Beam current</td>
<td>10 mA</td>
<td>1 mA</td>
</tr>
<tr>
<td>Normalized emittance [mm-mrad]</td>
<td>0.1 @7.7 pC</td>
<td>See, later page</td>
</tr>
<tr>
<td></td>
<td>1 @77 pC</td>
<td>See, later page</td>
</tr>
<tr>
<td>Repetition frequency of bunches</td>
<td>1.3 GHz</td>
<td>1.3 GHz (usual)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>162.5 MHz (for LCS)</td>
</tr>
<tr>
<td>RMS bunch length</td>
<td>1-3 ps (usual)</td>
<td>1-3 ps (usual)</td>
</tr>
<tr>
<td></td>
<td>~ 100 fs (compress.)</td>
<td></td>
</tr>
<tr>
<td>$E_{\text{acc}}$ in main linac</td>
<td>15 MV/m</td>
<td>8.2 MV/m</td>
</tr>
<tr>
<td>Gun high voltage</td>
<td>500 kV</td>
<td>390 kV</td>
</tr>
<tr>
<td>Max. heat load at 2K</td>
<td>80 W</td>
<td>100 - 80 W</td>
</tr>
</tbody>
</table>

R&D machine.
Key: Electron Gun, SC Cavity, Beam dynamics, THz, Laser-Compton scattering, etc.

Compact ERL at KEK

Circumference: ~ 90 m
In General, it is very difficult to force everyone to use one specific hardware, while the control group wants to reduce the number of support hardware. Hardware selection depends on the requirement.

If there are no special reason, we ask development team to use Yokogawa PLC based module (FA-M3 Series) as a “standard” field bus.
  - Long hardware lifetime
  - Reliability
  - Easy development: EPICS Ready!, Many experiences in KEK

- Ladder CPU for Real-time (or safety) application
- Linux CPU (F3RP61) for EPICS IOC
Multichannel Data Logger

- For temperature sensor or analog voltage
  - Yokogawa MW100
  - Chino Network Logger
    http://www.chino.co.jp/products/component/ke.html
  - Graphtec data logger
    http://www.graphtec.co.jp/site_instrument/instrument/index.html

EPICS device support or protocol files (Stream Device) for these equipment have been developed.
Magnet Power Supply

- **CAENels**
  - LiAM6005, SY3634
  - Each power supply directly attached to control network
  - ASYN + StreamDevice
- Two projectors to the wall
  - mainly for demonstration (for Guests/visitors)
- Desktop 27-inch display is mainly used for accelerator tuning.
  - 2 PCs for operation/ Beam tuning.
Software tools used

- Linux Server Machine / Windows console
- CSS for GUI, Archive/Retreival, Alarm

Thanks for Kay Kasemir for his great contribution!!

- EPISS 3.14
- CSS KEK version
  - http://www-linac.kek.jp/cont/epics/css/
  - Tutorial, documents, etc
  - Version 3.2.16 for cERL

Control System Studio (CSS) at KEK

Control System Studio (CSS) is an Eclipse-based collection of tools started at DESY and is now actively extended in the collaborating institutes.

Several different solutions have been employed in the past for DCS. Static displays can be replaced by CSS. Several other tools like duct as well. Some other Python-based scripting programs may be replaced.

The Rich Client Platform (RCP) provided by Eclipse enables unified common look-and-feel and shared control service libraries.

CSS/KEK Download site.

CSS download site at KEK was prepared by Dr. K. Kasemir in June 2011. It was studied by K. Furukawa of Cosylab in September 2011, February, August 2012, and June 2013, and by T. Michikawa in December 2013.
We want to share know-how in Japanese: EPICS Users JP wiki

- For advanced researcher/programmer: send e-mail to tech-talk!!
- Mailing List (in Japanese) ... not so active like tech-talk
  - epics-users@ml.post.kek.jp
Machine Status Panel

- Figure
Thanks for the Java environment, we can use Japanese on the panel.
- nice feature for operator (not good for scientists from foreign countries)
Almost 9,000 PVs are stored in archive

### cERL Archive Engine

<table>
<thead>
<tr>
<th>Summary</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>3.2.1.201505261345</td>
</tr>
<tr>
<td>Description</td>
<td>cERL</td>
</tr>
<tr>
<td>HTTP Server</td>
<td></td>
</tr>
<tr>
<td>State</td>
<td>RUNNING</td>
</tr>
<tr>
<td>Start Time</td>
<td>2015/08/25 15:16:58.164828722</td>
</tr>
<tr>
<td>Uptime</td>
<td>1.00 days</td>
</tr>
<tr>
<td>Workspace</td>
<td>/css/tmp/engcERL/</td>
</tr>
<tr>
<td>Groups</td>
<td>17</td>
</tr>
<tr>
<td>Channels</td>
<td>6900</td>
</tr>
<tr>
<td>Disconnected</td>
<td>1760</td>
</tr>
<tr>
<td>Batch Size</td>
<td>500 samples</td>
</tr>
<tr>
<td>Write Period</td>
<td>30 sec</td>
</tr>
<tr>
<td>Write State</td>
<td>OK</td>
</tr>
<tr>
<td>Last Written</td>
<td>2015/08/26 15:22:33.160811891</td>
</tr>
<tr>
<td>Write Count</td>
<td>26095 samples</td>
</tr>
<tr>
<td>Write Duration</td>
<td>3.2 sec</td>
</tr>
<tr>
<td>Idle Time</td>
<td>100.0 %</td>
</tr>
<tr>
<td>Memory</td>
<td>505.0 MB of 3555.5 MB used (14.2 %)</td>
</tr>
</tbody>
</table>

### Disk usage:
- 2012 (1.7 TB)
- 2013 (3.5 TB)
- 2014 (4.5 TB)
- 2015 (6.1 TB)
- 2016 (3.0 TB)
1. Introduction

2. Outline of cERL at KEK

3. Outline of iBNCT at Tokai

4. Commissioning, Tuning, Operation
   - Various tuning panel
   - CSS as an operation manual (procedure)
   - Software for rapid prototype
   - Hardware example: VME-Master
What is iBNCT?

• Location

About 1 hour by the shuttle-bus (via Highway)
- iBNCT Location: near the entrance of J-PARC
What is iBNCT?

- Ibaraki Boron Neutron Capture Therapy

1. **Administer boron-containing drug:**
   a boron-containing drug that selectively accumulates in cancer cells is used.

2. **Neutron irradiation:**
   The affected site is irradiated with an energy-adjusted neutron beam.

Courtesy Dr. Kumada, University of Tsukuba
Principle (cont.)

3 Neutrons react with boron:

emitted alpha beam and lithium particles destroy cancer cells.

4 Cancer cells are destroyed:

these particles only travel a distance of one cell width (about 10μm), allowing for cell-level treatment.

\[ n + B(10) = Li + \alpha + \gamma \]
Today, I would like to talk about control related topics of Ibaraki BNCT. Machine layout: Ion Source + RFQ + DTL + (Transfer Line ) + Be Target
accelerator layout

50 keV Ion Source

3 MeV RFQ

5 MeV DTL

8 MeV Beam Transport Line

Be Target
• 3 MeV RFQ + 5 MeV DTL
Control room

- 2 PCs for Operation and beam tuning. Large (wall-mount) display for status.
Control System

• Requirement for the control system is “Reliable System”

• Accelerator control system is developed by Cosylab.
• First beam is reported in the Cosylab newsletter
  – T. Nakamoto and T. Zagar
• Excellent work done by the company
  – No major trouble in the basic control system

• I need to follow-up some software tools such as
  – Beam Loss monitors
  – Utilities for beam tuning
  – Software to share information : wiki, NAS, etc
Fieldbus

- **Yokogawa PLC**
  - most of the accelerator equipment are controlled by PLC.
  - Ladder CPU + WideField (development environment)

- **Yokogawa SL1000**
  - CT, BPM, Loss Monitor, etc
  - VXI-11 protocol

- **EVG/EVR (mrf) for timing system**
GUI Example

- Magnet status/direct set (Cosylab)
**Archive Engine**

<table>
<thead>
<tr>
<th>Summary</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Version</strong></td>
<td>3.1.1.201305071351</td>
</tr>
<tr>
<td><strong>Description</strong></td>
<td>ibnct</td>
</tr>
<tr>
<td><strong>HTTP Server</strong></td>
<td>:4812</td>
</tr>
<tr>
<td><strong>State</strong></td>
<td>RUNNING</td>
</tr>
<tr>
<td><strong>Start Time</strong></td>
<td>2017/01/10 09:55:33</td>
</tr>
<tr>
<td><strong>Uptime</strong></td>
<td>123.07 days</td>
</tr>
<tr>
<td><strong>Workspace</strong></td>
<td>/workspace/</td>
</tr>
<tr>
<td><strong>Groups</strong></td>
<td>16</td>
</tr>
<tr>
<td><strong>Channels</strong></td>
<td>1615</td>
</tr>
<tr>
<td><strong>Disconnected</strong></td>
<td>16</td>
</tr>
<tr>
<td><strong>Batch Size</strong></td>
<td>500 samples</td>
</tr>
<tr>
<td><strong>Write Period</strong></td>
<td>30 sec</td>
</tr>
<tr>
<td><strong>Write State</strong></td>
<td>OK</td>
</tr>
<tr>
<td><strong>Last Written</strong></td>
<td>2017/05/13 11:37:06</td>
</tr>
<tr>
<td><strong>Write Count</strong></td>
<td>2914 samples</td>
</tr>
<tr>
<td><strong>Write Duration</strong></td>
<td>0.3 sec</td>
</tr>
<tr>
<td><strong>Idle Time</strong></td>
<td>100.0 %</td>
</tr>
<tr>
<td><strong>Memory</strong></td>
<td>4095.0 MB of 4095.0 MB used (100.0 %)</td>
</tr>
</tbody>
</table>

no disconn. channels during operation
Optics tuning (reduce beam loss)

iBNCT passed the radiation facility safety inspection in December 2016.

Transmission rate after DTL $\rightarrow$ near 100% (CT resolution)
1. Introduction

2. Outline of cERL at KEK

3. Outline of iBNCT at Tokai

4. Commissioning, Tuning, Operation (for cERL and iBNCT)
   - Example of tuning panel
   - CSS as an operation manual (procedure)
   - Software for rapid prototype
   - Hardware example: VME-Master
High Level Application: Software for beam tuning

- In case you need accelerator optics knowledge: Use SAD

KEKB Optics (Tune) Panel

PF-AR Injection bump panel

Other accelerator laboratories may use elegant, matlab/AT, etc.
Basic instruments control panel

<table>
<thead>
<tr>
<th>Q</th>
<th>K set</th>
<th>I set</th>
<th>I mon</th>
</tr>
</thead>
<tbody>
<tr>
<td>QMAG01</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
<tr>
<td>QMAG02</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
<tr>
<td>QMAG03</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
<tr>
<td>QMAG04</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
<tr>
<td>QMAG05</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
<tr>
<td>QMAG06</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
<tr>
<td>QMAG07</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
<tr>
<td>QMAG08</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>H</th>
<th>K set</th>
<th>I set</th>
<th>I mon</th>
</tr>
</thead>
<tbody>
<tr>
<td>ZHBMAG01</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
<tr>
<td>ZHBMAG02</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
<tr>
<td>ZHBMAG03</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
<tr>
<td>ZHBMAG04</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
<tr>
<td>ZHBMAG05</td>
<td>0.050</td>
<td>0.050</td>
<td>0.000</td>
</tr>
</tbody>
</table>

Direct set with K or I

Direct

Diff

Monitor value
The image shows a computer interface with a table and a comment field. The comment reads: "new optics for 7.7 pC, final". The author is Miyajima and Honda, and the date is 2014/06/20 23:20:28. The table lists various records with columns for Snap Val, Current, Current-Snap, Snap Moni, Monitor, and Monitor-Snap. The table includes data for different records such as CERL:MAG:ZH04:IDIR, CERL:MAG:ZV04:IDIR, etc. The interface also includes buttons for save and restore.
X-Y plot OPI : (example : RF Phase scan)

- General-purpose plot tool
  - main part is written CSS python script
  - disadvantage : difficult to move newer version of CSS! → should be implemented in software sequencer or other IOC
CSS as operation manual?

- Operator (non-programmer, non-accelerator Physicist) can create panels.
- I surprised they start to create “operator manual” using CSS.
- Using “Japanese” is mandatory for them
Table of Contents

1. Introduction

2. Outline of cERL at KEK

3. Outline of iBNCT at Tokai

4. Commissioning, Tuning, Operation
   – Various tuning panel
   – CSS as an operation manual (procedure)
   – Software for rapid prototype
   – Hardware example : VME-Master
IPython (Jupyter) Notebook

- Suitable for equipment control that does not need Accelerator optics.
- IPython core is running on server machine, client use web browser only.
- Intensively used in cERL and iBNCT
Rapid prototyping with IPython Notebook

- ex: RF conditioning and DTL tuner (slow) feedback control for iBNCT
- Need to adjust input voltage and pulse height, repetition rate, etc.
  - monitoring tuner position, RF frequency, power, ... and many other parameters.
  - Some patterns have been tried at the beginning.
  - “Quick and Dirty” approach required
- IPython Notebook has nice feature such as
  - easy to understand (script).
  - can execute a part (block) or whole script
- After the parameters are fixed, the script is migrated to EPICS sequencer, then create a CSS panel.
- Notebook is used like a “requirement definition document” + “Prototype”.
Misc. Hardware : VME-Master

- Commercial Product of “BeeBeans Technologies” Co.
  - [http://www.bbtech.co.jp/](http://www.bbtech.co.jp/) (KEK Venture Company)
- SiTCP (Silicon TCP in FPGA) for communication. No operating system.

We can use VME boards like a network-attached I/O module. Standard “Stream Device/Asyn” for communication.
Outline of operation procedure

- Please refer to the manual for details.

**Packet Format**

- **Start Address**: VME Address
- **Access Length**: data length
- **MODE**: VME access mode
  - read/write, A16/24/32, etc
- **ID#**: for verify CMD packet and ACK packet
- **CRC8**: for verify packet
• read pulse counter

```plaintext
# Example protocol file for VME-Master SiTCP
# REPIC 100MHz OCTAL CALER
#
addr   = 0x00 0x10 0x06 0x00;  # board base address 0x100600
leng   = 0x00 0x00 0x00 0x04;  # data length 4 byte
mode   = 0x05 0x40;            # mode
mode_r = 0x05 0x48;
id     = 0x01;
#
getCounter {
    out $addr $leng $mode   $id "%<crc8a>";
    in  $addr $leng $mode_r $id $crc_r "%4D";
}
```

Default CRC8 checksum uses different initial value from SiTCP format. We defined a new checksum pseudo-converter.
Comments on VME-Master

• VME-Master has been used for cERL and iBNCT
• Very good for small experiment because...
  – We can utilize many old VME boards
  – No need to setup development environment. Just use a socket communication.

• Latest version can support VME bus interrupt.
• Fast enough for non-realtime application

• We plan to use the VME-Master board to replace magnet power-supply controller for KEK-PF electron storage ring in coming summer.
  – Present: Linux CPU (IOC, non-realtime) + VME Bus-Bridge
  – Total 10 VME chassis
Summary

- Introduction of two accelerator control system
  - cERL
  - iBNCT
- Overview of control room, field bus, other hardware
- Software for accelerator commissioning, tuning, operation
  - EPICS Application
  - CSS GUI
  - IPython notebook
- Some hardware (example: VME-Master)
- Other software/hardware (excluded from today’s talk)
  - wiki for internal information sharing
  - Status display (CATV-like)
  - HipChat
  - Beaglebone Black as ioc
  - Yokogawa F3-HA12 module (12 channel 16 bit ADC)
  - .... and more