Application with RT-patched EPICS for real-time monitoring

October 23, 2012

Woongryol Lee
Contents

I. Introduction
   A. Motivation
   B. Background

II. Implementation
   A. System configuration
   B. Operation flow and optimization

III. Test results

IV. conclusion and future work
Introduction

Motivation

◆ Supervisory monitoring system to catch up the RT data
  ✓ Reflective memory card for the feedback control
  ✓ Analysis and new design about RT-network for the next fast control system
  ✓ Inspection tool for our feedback control interface

◆ Fast feedback control system in KSTAR
  ✓ Two kinds of RTOS in KSTAR
    • VxWorks : Local control system of magnetic power supply
    • Customized Linux : Plasma control system
  ✓ Improvement for the long pulse operation
    • All outbound interface of PCS was disconnected at RT-mode
    • Give a reconfigurable mechanism to PCS during plasma discharging time

◆ ITER standard RTOS
  ✓ We are in progress of CODAC Technologies evaluation project in KSTAR
Introduction

● Background

◆ Approaches to increase real time performance
  ✔ Application binding
    • restricting certain CPUs to running designated application processes
  ✔ Interrupt binding
    • designating specific CPUs handle device interrupts
  ✔ Memory pinning
    • designating that physical memory be exclusively allocated to dedicated processes.
  ✔ Scheduler priority control
    • ability to designate process priorities at a fine grained level

◆ Red Hat™ real-time OS
  ✔ MRG Realtime, TUNA

◆ Real-Time Support in EPICS
  ✔ Use POSIX priority scheduling by enabling special option
  ✔ Set CPU affinity using “epics-affinity-patch.txt”
    • From ITER and Cosylab.
System configuration

Simple layout

- Magnetic power supply
  - PF1
  - PF2
  - PF3U
  - PF3L
  - PF6U
  - PF6L
  - PF7
  - IVC
  - RMP

- Reflective Memory interface (Real time network)
- Fiber interface (Timing network)
- Time Synchronization System (TSS)
- Central Control System (CCS)
- NIC
- RFM board
- LTU

Reconfigured LTU

Real time monitoring system

Application of RT-patched EPICS for real-time monitoring, October 23, 2012
System configuration

● **Hardware features**

◆ **Host controller (red box)**
  ✓ General rack mountable PC
  ✓ Intel® Core™ i7-3930K, 3.2GHz, 12MB cache
  ✓ 6 cores, no Turbo Boost, no Hyper-Threading
  ✓ DDR3 16GB
  ✓ 150GB SSD

◆ **Local Timing Unit v.2 (green box)**
  ✓ Synchronized with Central Timing Unit (CTU) via fiber optics, 2Gbps
  ✓ Provide 10KHz clock to the event generator
  ✓ Generate reference clock for time measurement
  ✓ Installed on ATCA platform with PMC extension board
System configuration

**Hardware features**

◆ **Event generator (blue box)**
  ✓ Customized board using FPGA flexibility
  ✓ LTU v.1 was reconfigured event generator
  ✓ PMC true form-factor, Virtex-4 FX20
  ✓ 10 digital output for time checking

◆ **Reflective Memory board (red box)**
  ✓ High speed, easy to use fiber-optic network (2.12 Gbaud serially)
  ✓ Fiber network transfer rate 43 MByte/s to 170 MByte/s
  ✓ Star topology in KSTAR
  ✓ Main interface for feedback control between PCS and MPS
System configuration

● **Software features**
  
  ◆ **Host controller**
    - Red Hat Enterprise Linux 6.2 (2.6.32) x86_64
    - MRG Realtime – 2.6.33.9-rt31.74.el6rt.x86_64
      - Disable all unnecessary service except network.
      - But system need more tuning

  ◆ **Application features**
    - Recent stable released EPICS (3.14.12.2)
    - Apply the real-time support patch
    - Built on KSTAR standard software framework
      - Organized under driver/device support routine
      - Sequence for synchronized operation
      - State notification mechanism which is commonly used in DAQ and control system
      - Standard template for fast development
    - Use ram-disk for fast archiving: 2.5 GB/s
**KSTAR sequence synchronized operation**

**Software framework**
- It has been developed since 2009.
- Now we consider adding a real-time control feature

---

**Operation flow**

- **Software framework**
  - Developed since 2009.
  - Consider adding a real-time control feature.

---

**Machine / Experiment Network**

**Timing Network**

- Shot sequence start
- Get shot number
- Get blip time
- Arming
- Wait for trigger
- In-progress
- Post-processing
- Data transfer
- Internal task stop
- Shot sequence stop
- Time flow
- Prepare file saving
- Perform local system defendant functions

---

Application of RT-patched EPICS for real-time monitoring, October 23, 2012
Operation flow

- Application work flow
  - Unit task per each input pulse

1. Interrupt response time
2. Response time in user space
3. Application working time
# Main function

## Archiving the real time data

- Text file contains these information

<table>
<thead>
<tr>
<th>Event counter</th>
<th>Blip based</th>
<th>Self counter</th>
<th>PCS fault code</th>
<th>Force fast INTL</th>
<th>RMP P/S BI, MI, TI</th>
</tr>
</thead>
<tbody>
<tr>
<td>RT counter</td>
<td>CHK counter</td>
<td>Current Time</td>
<td>PCS</td>
<td>All PFs</td>
<td>IVC</td>
</tr>
</tbody>
</table>

- IC. Over current
- IP error
- P/S fault
- External fault
- PF over voltage
- PF voltage
- PF over current
- PF error
- Ne error
- IP minimum
Main function

- **RTMON' graphic user interface**
  - Developed by using QT based KWT
  - Standard layout for Operation mode
  - LTU configuration
System tuning

• Application binding
  ◆ Set CPU affinity
    ✓ Boot option \textit{isolcpus}=1-5
    ✓ epicsThreadSetCPUAffinity( pthreadInfo, "5")
    ✓ epicsThreadSetPosixPriority(pthreadInfo, 92, "SCHED_FIFO")

<table>
<thead>
<tr>
<th>NAME</th>
<th>EPICS ID</th>
<th>PTHREAD ID</th>
<th>LWP ID</th>
<th>OSI P</th>
<th>OSI P</th>
<th>STATE</th>
<th>POLICY</th>
<th>AFFINITY</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>main</em></td>
<td>0x1a83190</td>
<td>0</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>OK</td>
<td></td>
<td>SCHED_FIFO</td>
</tr>
<tr>
<td>errlog</td>
<td>0x1a8cd50</td>
<td>140184245008128</td>
<td>6498</td>
<td>10</td>
<td>10</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>RTMON_ctrl</td>
<td>0x1a941b0</td>
<td>14018421767616</td>
<td>6490</td>
<td>10</td>
<td>10</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>RTMON_StopEvt</td>
<td>0x1a94480</td>
<td>14018421754448</td>
<td>6491</td>
<td>10</td>
<td>10</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>RTCORE_ctrl</td>
<td>0xb139b0d0</td>
<td>14018421749280</td>
<td>6492</td>
<td>10</td>
<td>10</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>RMCHK_ctrl</td>
<td>0xb152660</td>
<td>14018421741112</td>
<td>6493</td>
<td>10</td>
<td>10</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>Taskwd</td>
<td>0xb157660</td>
<td>14018421738944</td>
<td>6494</td>
<td>10</td>
<td>10</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>timerQueue</td>
<td>0xb154620</td>
<td>14018421703776</td>
<td>6495</td>
<td>76</td>
<td>89</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>cbLow</td>
<td>0xb157190</td>
<td>14018421673756</td>
<td>6496</td>
<td>59</td>
<td>50</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>cbMedium</td>
<td>0xb16c20</td>
<td>140184216299152</td>
<td>6497</td>
<td>44</td>
<td>43</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>cbHigh</td>
<td>0xb16ce10</td>
<td>140184215682768</td>
<td>6498</td>
<td>71</td>
<td>70</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>dbCaLink</td>
<td>0xb16d2d0</td>
<td>1401842152384</td>
<td>6499</td>
<td>50</td>
<td>50</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>KSTAR_ShotSeq</td>
<td>0xb16d560</td>
<td>140184214624099</td>
<td>6500</td>
<td>96</td>
<td>89</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>TimerQueue</td>
<td>0xb17790</td>
<td>8018338</td>
<td>14018421389736</td>
<td>6501</td>
<td>92</td>
<td>91</td>
<td>OK</td>
<td>SCHED_FIFO</td>
</tr>
<tr>
<td>CAC-UDP</td>
<td>0xb17790</td>
<td>8018338</td>
<td>14018421248096</td>
<td>6502</td>
<td>94</td>
<td>93</td>
<td>OK</td>
<td>SCHED_FIFO</td>
</tr>
<tr>
<td>ipToAsciiProxy</td>
<td>0xb17790</td>
<td>8018338</td>
<td>14018421435776</td>
<td>6503</td>
<td>10</td>
<td>10</td>
<td>OK</td>
<td>SCHED_FIFO</td>
</tr>
<tr>
<td>CAC-UDP-send</td>
<td>0xb17790</td>
<td>8018338</td>
<td>140184210454272</td>
<td>6504</td>
<td>92</td>
<td>91</td>
<td>OK</td>
<td>SCHED_FIFO</td>
</tr>
<tr>
<td>CAC-UDP-rev</td>
<td>0xb17790</td>
<td>8018338</td>
<td>140184210982656</td>
<td>6505</td>
<td>88</td>
<td>87</td>
<td>OK</td>
<td>SCHED_FIFO</td>
</tr>
<tr>
<td>CAC-UDP-send</td>
<td>0xb17790</td>
<td>8018338</td>
<td>140184210913264</td>
<td>6506</td>
<td>92</td>
<td>91</td>
<td>OK</td>
<td>SCHED_FIFO</td>
</tr>
<tr>
<td>CAC-UDP-rev</td>
<td>0xb17790</td>
<td>8018338</td>
<td>140184210563264</td>
<td>6507</td>
<td>92</td>
<td>91</td>
<td>OK</td>
<td>SCHED_FIFO</td>
</tr>
<tr>
<td>RMCHK_DAQ</td>
<td>0xb169650</td>
<td>140184206853552</td>
<td>6508</td>
<td>93</td>
<td>92</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>RTCORE_RT</td>
<td>0xb16a170</td>
<td>140184206317312</td>
<td>6509</td>
<td>93</td>
<td>92</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>scanfduce</td>
<td>0xb169266</td>
<td>014183353334112</td>
<td>6510</td>
<td>70</td>
<td>59</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
<tr>
<td>scan10</td>
<td>0xb18270</td>
<td>140184206051072</td>
<td>6511</td>
<td>69</td>
<td>59</td>
<td>OK</td>
<td>SCHED_FIFO</td>
<td></td>
</tr>
</tbody>
</table>
**System tuning**

- **Interrupt binding**
  - Set IRQ attribute by Tuna
    - Priority 95, Affinity 5, Policy FIFO

**Figure: System tuning**
- **Set IRQ attributes for this IRQ:**
  - **Policy:** SCHED_FIFO, **Scheduler priority:** 95, **Affinity:** 5

**Snippet:**
```
[root@RTMON ~]# cat /proc/interrupts
CPU0  CPU1  CPU2  CPU3  CPU4  CPU5
0:    142  2004  0    0    0    0
1:    0    8     0    0    0    0
8:    0    21    0    0    0    0
16:    0   20   2    0    0    0
512479144: 10-APIC-edge timer
```

**Diagram:**
- Priority 95, Affinity 5, Policy FIFO
Time measurement

• **Accuracy**

  ◆ Oscilloscope
  ✓ Model-DSO9404A, Infiniium DSO - 4 GHz, 10/20 GSa/s, 4 Ch

- To measure the consumed time, we use TTL output channel on the event generator
- Two command to make single pulse.
  
  #1 WRITE32(pRTcore→base0 + 0x4, 0x1);
  #2 WRITE32(pRTcore→base0 + 0x4, 0x0);

- Takes about 151 nanoseconds
Test results

**Captured test images**

* Interrupt response time in kernel space
  - Not use CPU affinity
  - Use CPU affinity

![Test results diagram](image-url)
Test results

● Measured time value

◆ Condition
  ✓ 10KHz interrupt in normal operation
  ✓ unit=microsecond, doing in a few minutes

<table>
<thead>
<tr>
<th>case</th>
<th>mean</th>
<th>min</th>
<th>max</th>
<th>jitter</th>
<th>Check point</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7.489</td>
<td>6.631</td>
<td>14.261</td>
<td>7.6303</td>
<td>LTU → Kernel ISR</td>
</tr>
<tr>
<td>2</td>
<td>7.162</td>
<td>6.647</td>
<td>7.842</td>
<td>1.195</td>
<td>LTU → User space return time</td>
</tr>
<tr>
<td>3</td>
<td>6.919</td>
<td>6.409</td>
<td>8.199</td>
<td>1.789</td>
<td>LTU → final action include user application</td>
</tr>
</tbody>
</table>

1. MRGR, no CPU affinity
   - 0 1 4
   - 2 3 5

2. MRGR, isolcpus=1-5, affinity=5
   - 0 1 4
   - 2 3 5

3. MRGR, isolcpus=5, affinity=5
   - 0 1 4
   - 2 3 5
Test results

**Special case**

- **Total time with heavy job**

  - Not use CPU affinity
  - Use CPU affinity

<table>
<thead>
<tr>
<th>case</th>
<th>mean</th>
<th>min</th>
<th>max</th>
<th>jitter</th>
<th>remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40.572</td>
<td>38.681</td>
<td>53.115</td>
<td>14.433</td>
<td>LTU → final action with heavy job</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>38.817</td>
<td>37.845</td>
<td>42.424</td>
<td>4.579</td>
<td></td>
</tr>
</tbody>
</table>
Conclusion and Future work

**Conclusion**

- **Achieved 10KHz stable operation**
  - file I/O, network connection (CA) are alive
  - Possible in normal or harsh condition

- **Possible 20KHz operation**
  - 20KHz (50 us) operation also possible
  - It depends on target application

- **Effect of CPU affinity**
  - Dominant effect is using RTOS (MRGR)
  - Using CPU affinity increase the system stability

**Future work**

- **Evaluation in feedback control system**
  - Develop real time feedback control system on plasma density
  - Survey of real time network performance
Thanks for your attention.