LabVIEW PCAS INTERFACE FOR NI CompactRIO

G. Liu*, J. Wang, K. Xuan, C. Li, NSRL, USTC, Hefei, Anhui 230029, China
K. Zheng, Y. Yang, National Instruments China, Shanghai 201203, China

Abstract

When the NI LabVIEW EPICS Server I/O Server is used to integrate NI CompactRIO devices running under VxWorks into EPICS, we notice that it only supports "VAL" field, neither alarms nor time stamps are supported. In order to overcome these drawbacks, a new LabVIEW Channel Access Portable Server (PCAS) Interface is developed, and is applied to the Hefei Light Source (HLS) cooling water monitor system. The test results in the HLS cooling water monitor system indicate that this approach can greatly improve the performance of the NI CompactRIO devices in EPICS environment.

INTRODUCTION

The Hefei Light Source (HLS) is a dedicated second generation VUV synchrotron radiation light source. It was designed and constructed two decades ago. In order to improve the performance of the HLS, especially to obtain higher brightness and more straight sections, an upgrade project was started from the end of 2009. The cooling water monitor system was reconstructed in the project. The CompactRIO (Compact Reconfigurable Input Output) product from National Instruments (NI) is adopted to the cooling water monitor system because it is designed for harsh environment, and the software is easy to duplicate and deployed.

The NI CompactRIO devices are running under VxWorks. When the NI LabVIEW EPICS Server I/O Server [1] is used to integrate the NI CompactRIO devices into EPICS, we notice that it only supports "VAL" field, neither alarms nor time stamps are supported. In order to overcome these drawbacks, a new LabVIEW Channel Access Portable Server (PCAS) [2] Interface is developed.

The LabVIEW PCAS Interface is a software library which provides channel access server ability to LabVIEW code and emulates the way how an IOC manages process variables. It is based on the PCAS which implements the underlying channel access functions. There are two ambitious aims in designing the LabVIEW PCAS interface: (1) it supports all platforms where LabVIEW exists, including Windows, Linux and VxWorks; (2) it supply common records with common fields, similar usage pattern as IOC.

DEVELOPMENT

The LabVIEW PCAS Interface employs the Channel Access Portable Server distributed with EPICS base. The portable server comprises of server library and server tool.

*gfliu@ustc.edu.cn

The server library refers to the software that lies beneath the C++ class-interface to the portable server. The developer only needs to know how to use the interface in order to create a server tool. A server tool is a specific application written by a developer using this interface.

There is no record in the portable server, all PVs are standalone data. In order to provide users an IOC-like interface, we encapsulate PVs in the format of <record name>.<field name>, e.g. Temp.VAL.

Figure 1 is the flowchart of a server application, including the VI interfaces and the background thread. The VI interfaces are functions we provide to users. The background thread implements all the supported CA server interfaces called by CA clients. The records store all data used by the server application. In the background thread, the Channel Access is handled by the portable server. We implement the callbacks required by underlying components.

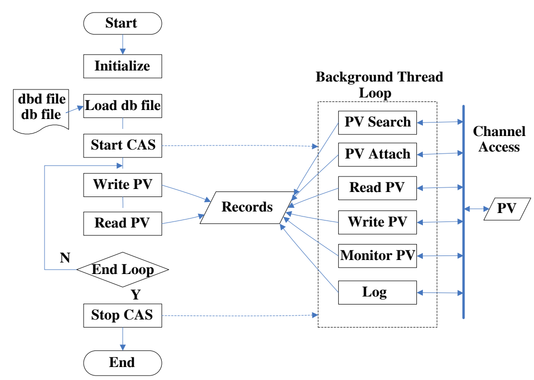


Figure 1: Flowchart of the LabVIEW PCAS Interface.

There are many record types defined by EPICS base, but we only support some common records with common fields up to now in order to limit the work load. Table 1 shows the supported record types and field. Here, "all" refers to the record type ai/ao, bi/bo, stringin/stringout and waveform.

Table 1: Record Types and Fields Supported by the LabVIEW PCAS Interface

Record Type	**Fields**
all	VAL/STAT/SEVR
ai/ao	HIHI/LOLO/HIGH/LOW HHSV/LLSV/HSV/LSV HYST/ADEL/MDEL
bi/bo	ZSV/OSV/COSV
waveform	NELM
ai/ao/waveform	HOPR/LOPR/PREC/EGU

ISBN 978-3-95450-146-5

USAGE

As mentioned earlier, the LabVIEW PCAS Interface is a software library. This library consists of several LabVIEW VIs and shared object libraries. The LabVIEW VIs are platform independent and the shared object libraries are compiled for each supported platform. They can be called by VIs through the "Call Library Node".

Figure 2 shows how the LabVIEW PCAS Interface is used. The server application first creates a CA server with a given server name. A .dbd file which includes all the database definitions, and a .db file to configure records and PVs that it handles are loaded into the server. The "Run CAS" VI is used to start the server application. The CA client can access the server application after it is started. During the running time, the "Write PV" and "Read PV" VIs are used to write the PV value or get the PV value respectively. If a PV is changed by the server application or by a client with 'caput', at least one monitor is set on that PV, and an event is posted by the server application. The "Stop CAS" VI is used to stop the server application.

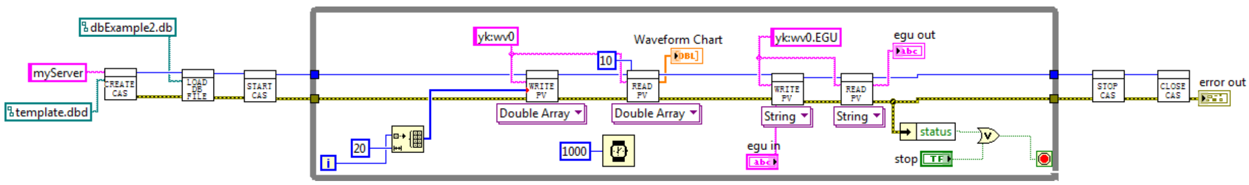


Figure 2: Usage of the LabVIEW PCAS Interface through LabVIEW code.

APPLICATION

The signals to be monitored in the HLS cooling water monitor system are the cooling water temperature, magnet temperature, cooling water pressure and cooling water flow. There are totally about 1000 signals, and these signals are distributed over the whole light source area. The CompactRIO products from National Instruments are adopted. The hardware structure of the cooling water monitor system is shown in Figure 3.

The CompactRIO is a real-time embedded industrial controller made by National Instruments for industrial control systems. It is a combination of a real-time controller, reconfigurable IO Modules, a FPGA module and an Ethernet expansion chassis. The National Instruments CompactRIO devices used for the cooling water monitor system are NI cRIO 9073, NI 9217 and NI 9208.

The NI cRIO 9073 is an 8-slot chassis with a 266MHz real-time processor, a 2M gate reconfigurable FPGA and a 10/100BASE-TX Ethernet port. The embedded processor runs the WindRiver VxWorks real-time operating system. The NI 9217 is a module which can detect 4 channels 3/4-wire RTD signals. The NI 9208 is a module which can detect 16 channels current signals with the range +/-21.5mA.

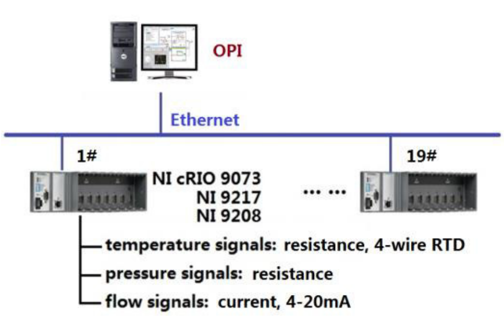


Figure 3: Hardware structure of the HLS Cooling Water Monitor System.

A LabVIEW program package is developed, and installed in each NI cRIO 9073 with a configuration file. In this way it is easy to operate, maintain and deploy in a large-scale. Figure 4 shows the software list downloaded to each NI cRIO 9073. As shown in Figure 5, the configuration file includes the information of all chassis and the installed modules. The configuration file can be generated step by step with a LabVIEW VI for all online NI cRIO 9073 chassis, or it can be edited with any text editors. The configuration file will be checked automatically by the software when the NI cRIO 9073 restarts. The USER1 LED will flash when any conflict or mismatch occurs.

In order to monitor the status of the NI cRIO 9073, two time related PVs are added to monitor the current time and startup time respectively.

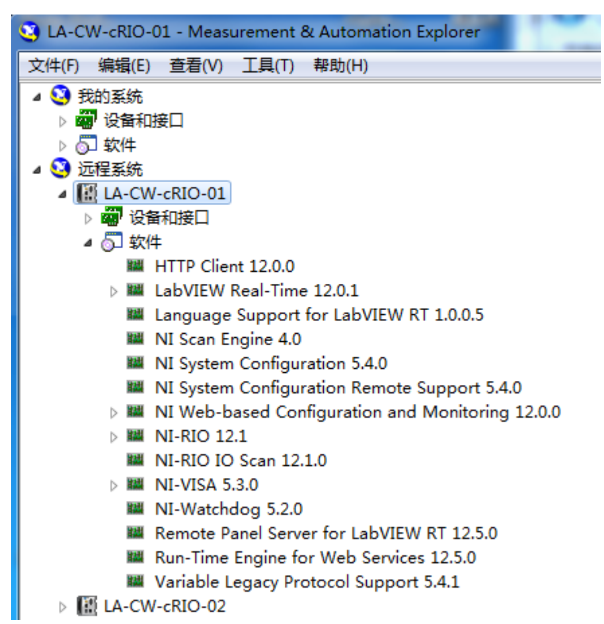
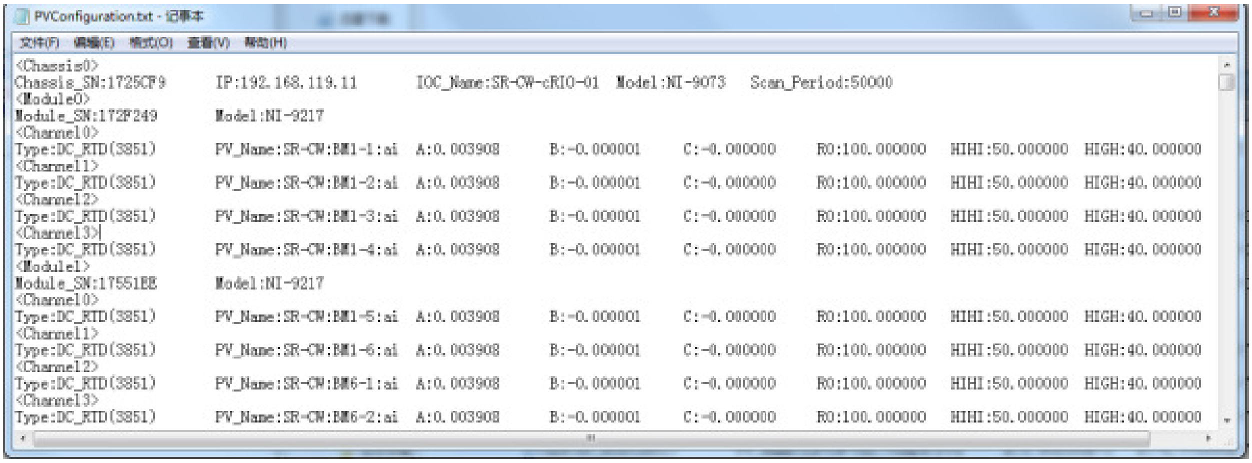


Figure 4: Software downloaded to the NI cRIO 9073.

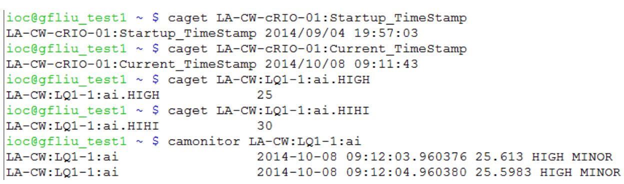
ISBN 978-3-95450-146-5



```
PVConfiguration.txt - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
<Chassis0>
Chassis_SN:1725CF9     IP:192.168.119.11      IOC_Name:SR-CW-cRIO-01  Model:NI-9073   Scan_Period:50000
<Module0>
Module_SN:172F249      Model:NI-9217
<Channel0>
Type:DC_RTD(3851)      PV_Name:SR-CW:BM1-1:ai  A:0.003908   B:-0.000001   C:-0.000000   R0:100.000000   HIHI:50.000000  HIGH:40.000000
<Channel1>
Type:DC_RTD(3851)      PV_Name:SR-CW:BM1-2:ai  A:0.003908   B:-0.000001   C:-0.000000   R0:100.000000   HIHI:50.000000  HIGH:40.000000
<Channel2>
Type:DC_RTD(3851)      PV_Name:SR-CW:BM1-3:ai  A:0.003908   B:-0.000001   C:-0.000000   R0:100.000000   HIHI:50.000000  HIGH:40.000000
<Channel3>
Type:DC_RTD(3851)      PV_Name:SR-CW:BM1-4:ai  A:0.003908   B:-0.000001   C:-0.000000   R0:100.000000   HIHI:50.000000  HIGH:40.000000
<Module1>
Module_SN:17551BE      Model:NI-9217
<Channel0>
Type:DC_RTD(3851)      PV_Name:SR-CW:BM1-5:ai  A:0.003908   B:-0.000001   C:-0.000000   R0:100.000000   HIHI:50.000000  HIGH:40.000000
<Channel1>
Type:DC_RTD(3851)      PV_Name:SR-CW:BM1-6:ai  A:0.003908   B:-0.000001   C:-0.000000   R0:100.000000   HIHI:50.000000  HIGH:40.000000
<Channel2>
Type:DC_RTD(3851)      PV_Name:SR-CW:BM6-1:ai  A:0.003908   B:-0.000001   C:-0.000000   R0:100.000000   HIHI:50.000000  HIGH:40.000000
<Channel3>
Type:DC_RTD(3851)      PV_Name:SR-CW:BM6-2:ai  A:0.003908   B:-0.000001   C:-0.000000   R0:100.000000   HIHI:50.000000  HIGH:40.000000
```

Figure 5: Configuration file.

We integrate the HLS Cooling Water Monitor System into EPICS with the LabVIEW PCAS Interface. Figure 6 shows the interactive interface with commands, alarms and time stamps. The system is stable for over one month.



```
ioc@gfliu_test1 ~ $ caget LA-CW-cRIO-01:Startup_TimeStamp
LA-CW-cRIO-01:Startup_TimeStamp 2014/09/04 19:57:03
ioc@gfliu_test1 ~ $ caget LA-CW-cRIO-01:Current_TimeStamp
LA-CW-cRIO-01:Current_TimeStamp 2014/10/08 09:11:43
ioc@gfliu_test1 ~ $ caget LA-CW:LQ1-1:ai.HIGH
LA-CW:LQ1-1:ai.HIGH            25
ioc@gfliu_test1 ~ $ caget LA-CW:LQ1-1:ai.HIHI
LA-CW:LQ1-1:ai.HIHI            30
ioc@gfliu_test1 ~ $ camonitor LA-CW:LQ1-1:ai
LA-CW:LQ1-1:ai                 2014-10-08 09:12:03.960376 25.613 HIGH MINOR
LA-CW:LQ1-1:ai                 2014-10-08 09:12:04.960380 25.5983 HIGH MINOR
```

Figure 6: Interactive interface with commands.

The common EPICS extensions tools, e.g. probe, StripTool, edm, Channel Archiver, RDB Channel Archiver, CSS BEAST, work well with this interface. However, ALH is not compatible with it. The warning message “alCaNewAlarmEvent failed” popups when ALH starts. ALH has the same behaviour when it is tested with the EPICS excas. So the incompatiblity should originate from the PCAS.

CONCLUSION

The test results in the HLS cooling water monitor system show that the operation of the LabVIEW PCAS Interface is stable and its behaviour is similar to that of a normal IOC. The LabVIEW PCAS Interface can be regarded as a good solution to integrate the NI CompactRIO devices into EPICS.

REFERENCES

[1] NI Developer Zone, "Interactively Configuring EPICS I/O Servers", http://www.ni.com/white-paper/14149/en.

[2] Philip Stanley, "Channel Access Portable Server: Reference Guide", http://www.aps.anl.gov/epics/extensions/cas/CAS_Reference.pdf.

ISBN 978-3-95450-146-5