Handling Serial and GPIB Devices at Jefferson Lab

Pavel Chevtsov
- Application Management System

- Extensions to Handle GPIB and Serial Devices

- Conclusions
Application Management System at Jefferson Lab
Control Applications

Applications with EPICS Databases

Device Support Only Applications
Control Applications

Development Area

Production Area
Controls Group

November 19-22

EPICS Collaboration

Jefferson Lab

$(DVL)$

Appl1

Ver_1

Ver_2

Ver_K

Appl2

ApplN

$(PROD)$

Appl1

Ver_1

Ver_2

Ver_K

Appl2

ApplN

src

Makefile(s)

Appl1_code.c (c++)

O.mv162  O.mv167

O.mv177  O.mv2700

vxcmd

startup.appl1

startup.appl1init

mv162

mv167

mv177

mv2700
IOC basic directory:

$(IOCS)

- ioc_1
- ioc_2
- ioc_M

startup config/config
links: applV applO startup.appl startup.applinit
The links are created by the linkmaker script:

linkmaker.pl  -i ioc_name –f config_file

config_file:

---

ioc_name:
arch, mv2700
...

serialDrv,  313g2   ->  4-2
gpibGSlib, 313f     ->  2-2
dnetDrv, 313a       ->  1-1

serialDrvV       ->
$(PROD)/serialDrv/4-2
serialDrvO       ->
$(PROD)/serialDrv/4-2/mv2700
startup.serialDrv  ->
serialDrvV/vxcmd/startup.serialDrv
To install the version K of the control application appl on the IOC_ABC, all you have to do is:
In the $(DVL)/Ver_K/src directory:
- Compile your code (do not forget to prepare startup.appl and startup.applinit files in vxcmd)
- Run “make install”. It will copy the object code, startup.appl and startup.applinit files into the $(PROD)/Ver_K directory. Now they are available for the IOC_ABC

In the $(IOCS)/IOC_ABC directory:
- Add the information about your appl into the config/config file
- Run linkmaker script. This will create the links to applV, applO, startup.appl, startup.applinit
- Add the info about your appl into the startup file
Reboot IOC_ABC

Enjoy your new application on the IOC !!!
The extensions to the Application Management System to simplify handling communication hardware
Serial (RS-232) and GPIB devices.

- Common Serial Driver
- GPIB Support Library
- PLC Support Library

All this software is activated with the use of a very limited number of library calls.
initCommHardware(carrier_board_type, parameters)

serialPortConfig(carrier, slot, port, baud, parity, stop, bits, flow, intNumb)
Extension 1
$(IOCS)/DATA/CommHardware

ioc_name.commconfig     ioc_name.gpibconfig

ioc_name.serialconfig   ioc_name.plcconfig
# iocCOMM.commconfig
#
#-------------------
initCommHardware("IPAC", &vipc616, "0x0000")
initCommHardware("IPAC", &vipc616, "0x6000")
iosSERTIAL.serialconfig

# Carrier 0, Slot "A" that is slot number 1, port 0, 9600 baud,  
# no parity, 1 stop bit, 7 bits for a word, no flow control,  
# intVec number is 0x42  
# Carrier 0, Slot "B" that is slot number 1, port 1, 19200 baud,  
# no parity, 2 stop bits, 8 bits for a word, no flow control,  
# intVec number is 0x44

serialPortConfig(0, 0, 0, 9600, 'N', 1, 7, 'N', 0x42)
serialPortConfig(0, 1, 1, 19200, 'N', 2, 8, 'N', 0x44)
The registration of every GPIB IP module which is handled by the IOC is done with the next two GPIB IP driver and device support library calls (the order is important !!!):

initGpibGsLib( CarrierNumber, SlotNumber, IntVec )
GsGpibLinkConfig( LinkNumber, CarrierNumber, SlotNumber )

initGpibGsLib(0, 1, 0x47)
GsGpibLinkConfig(10, 0, 1)
initGpibGsLib(0, 2, 0x49)
GsGpibLinkConfig(11, 0, 1)
# iocBD1.plcconfig
#
# ******************************************************
#
# Create our plc information entry
# ---------------------------------------------
#
# Carrier 0, Slot "A" that is slot number 0, port 0,
# for the PLC HADUMP
# Carrier 0, Slot "A" that is slot number 0, port 1,
# for the PLC HCDUMP
#
createDnSerialPLC("HADUMP", 0, 0, 0, 1)
createDnSerialPLC("HCDUMP", 0, 0, 1, 1)
Extension 2
# APP: serialDrvLib
# serialDrvLib_loaded=1

# load module for serialDrvLib
ld<serialDrvO/serialDrvLib

Smart VxWorks Shell Script
IOC_ABC

<startup.serialDrv

Loads and Activates software

$(IOCS)/DATA/CommHardware

IOC_ABC.serialconfig
To connect a new serial or GPIB control device to IOC_ABC, all that you have to do is:
Into $(IOCS)/DATA/CommHardware directory:

- add the information about the data communication hardware into IOC_ABC.commconfig file
- register your new communication channel into the proper IOC_ABC. (gpibconfig, serialconfig, plcconfig) file

Into the $(IOCS)/IOC_ABC directory:

- make all necessary references to the used device support libraries into the startup file
Connect your control device to the data communication hardware (use the proper connection cables!)

Create your database that uses serial or GPIB communication interface and make the references to it for the IOC_ABC with the use of the Application Management System Tools
Reboot IOC_ABC

Enjoy your new application on the IOC !!!
Benefits:

- The information about all control devices and the used data communication hardware is kept in one standard place. This makes it easy to support existing and add new hardware components throughout the whole control system.

- Each device support library has only one `startup.appl` file. This file makes all the work on loading the device support software into the IOC.
All this works if:

- The device support software is reliable and provides troubleshooting mechanisms

- You have a Control Device Information System with
  - documentation on the device control software
  - documentation on the data communication hardware
  - examples on the use of this software and hardware
“Information-Control Software for Handling Serial Devices in an EPICS Environment”, ICALEPCS-2001

“PLC Support Software at Jefferson Lab”, PCaPAC-2002

Extension 3

Control Device Information System
THANK YOU