F3RP61 boot up with EPICS IOC

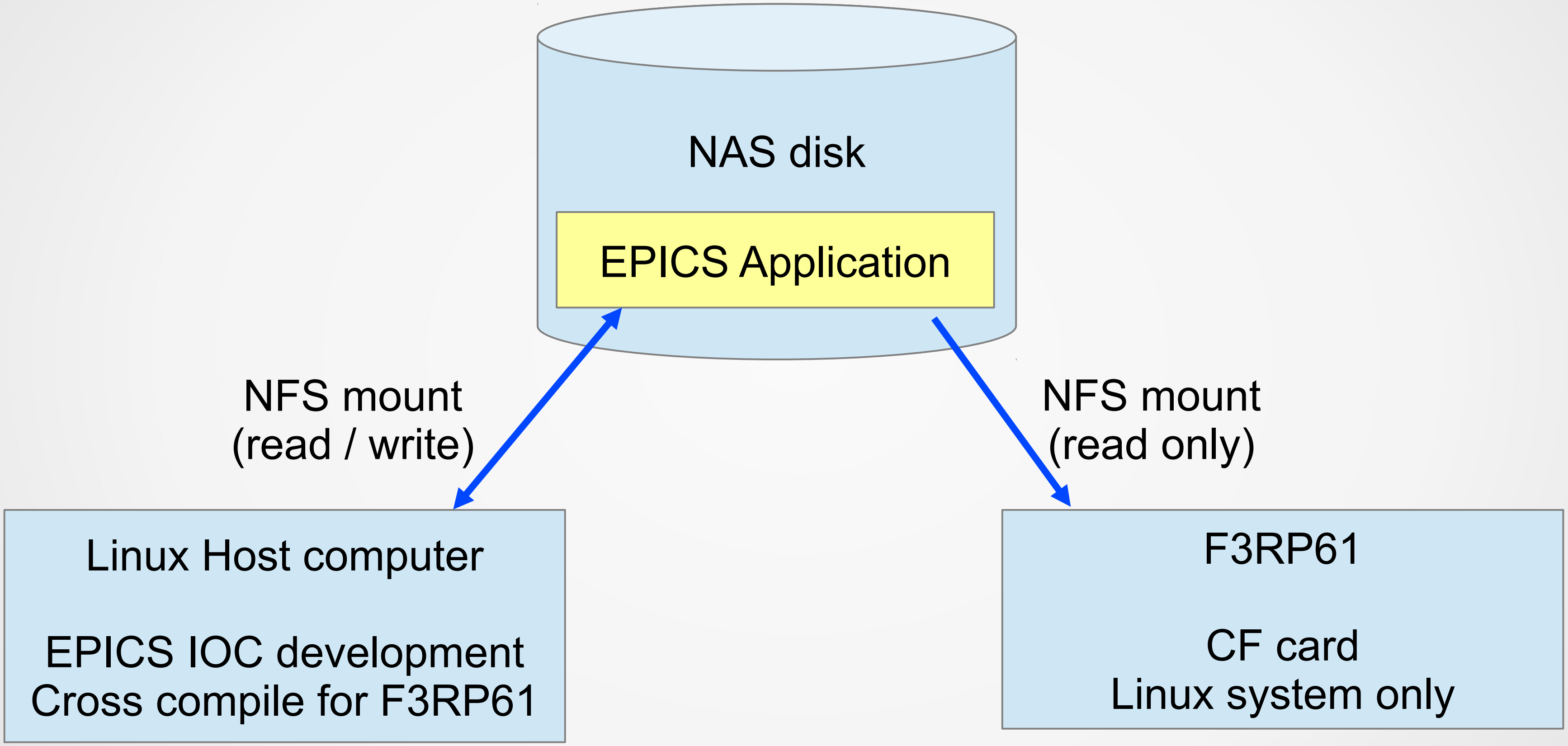
中村 卓也 (Takuya Nakamura)
三菱電機システムサービス
(Mitsubishi Electric System & Service CO., LTD.)

2013.01.16

F3RP61 configuration

- CF card and NAS disk combination.
  (NAS : Network Attached Storage)
- CF card include Linux system only.
- EPICS applications are build on NAS.
- F3RP61 mounts NAS disk. (via NFS protocol)
- EPICS IOC process is started from **“/etc/rc.local”**

KEKB Environment



NAS disk

EPICS Application

NFS mount
(read / write)

NFS mount
(read only)

Linux Host computer

EPICS IOC development
Cross compile for F3RP61

F3RP61

CF card
Linux system only

F3RP61 Network configuration (CF card)

- Network interface setting
  /etc/sysconfig/network-scripts/ifcfg-eth0

Static setting example

```
DEVICE="eth0"
BOOTPROTO=static
ONBOOT=yes
IPADDR="172.19.51.214"
NETMASK="255.255.224.0"
#NETWORK="172.19.32.0"
#BROADCAST="172.19.63.255"
#GATEWAY="172.19.32.1"
```

We choose static setting.

DHCP setting example

```
DEVICE="eth0"
BOOTPROTO=dhcp
ONBOOT=yes
```

F3RP61 File system configuration (CF card)

- File system mount setting
  /etc/fstab

"noatime" option

```
/dev/hda1              /             ext3    defaults,noatime        0 0
none                  /proc          proc    defaults        0 0
tmpfs                 /dev/shm          tmpfs   defaults        0 0
## NFS mount (read only)
saddisk3:/vol/nadata1a/proj     /mnt/proj       nfs      ro,hard,intr    0 0
saddisk3:/vol/nadata1a/users    /mnt/users      nfs      ro,hard,intr    0 0
```

- Make Symbolic links for our environment

```
% cd /
% ln -s /mnt/proj proj
% ln -s /proj/cont/cont cont
% ln -s /mnt/users users
```

KEKB Environment
/proj   : EPICS base, modules, extensions directory
/cont : EPICS IOC Application directory
/users: User's home directory (not necessary)

EPICS IOC start @ Linux boot up

- /etc/rc.local

```
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

# load m3io module
/sbin/insmod /opt/m3iodrv.ko

# load grabber module
/sbin/insmod /opt/grabber/driver/grabber.ko

touch /var/lock/subsys/local

#ioscan&

### 2013.01.17 start IOC with procServ test, MSC Takuya Nakamura
/users/takuya/epics314/epics314_f3rp61-201/iocBoot/iocTEST/startIOC_with_procServ.sh
```

EPICS IOC start @ Linux boot up (cont'd)

- startIOC_with_procServ.sh

```
### Environment setup
export BUSY_HOME=/usr/local/busybox
export EPICS_BASE=/proj/epics/R314/R31411/base
export PATH=$BUSY_HOME/bin:$BUSY_HOME/sbin:$BUSY_HOME/usr/bin:$PATH:$EPICS_BASE/bin/linux--f3rp6x/

IOCNAME="IOCCORP61TEST"
IOC_DIR="/users/takuya/epics314/epics314_f3rp61-201/iocBoot/iocTEST"

PROC_SERVE="/proj/epics/R314/R31411/extensions/bin/linux-f3rp61/procServ"

###PID_FILE="${IOCLOG_BASE}${IOCNAME}_procServ.pid"
PORT_NUM="20000"

SCRIPT="./${IOCNAME}.cmd"   ← IOCCORP61TEST.cmd

echo cd ${IOC_DIR}
echo ${PROC_SERVE} -i ^D^C --allow ${PORT_NUM} ${SCRIPT}
cd ${IOC_DIR}
${PROC_SERVE} -i ^D^C --allow ${PORT_NUM} ${SCRIPT}
```

EPICS IOC start @ Linux boot up (cont'd)

- IOCCORP61TEST.cmd

```
#!../../bin/linux-f3rp61/TEST

## You may have to change TEST to something else
## everywhere it appears in this file

< envPaths

cd ${TOP}

## Register all support components
dbLoadDatabase("dbd/TEST.dbd")
TEST_registerRecordDeviceDriver(pdbbase)

## Load record instances
#dbLoadTemplate "db/userHost.substitutions"
dbLoadRecords("db/dbExample2.db","user=ET_takuya, no=1, scan=1 second")
dbLoadRecords("db/dbExample3.db","user=ET_takuya, no=2, scan=1 second")
dbLoadRecords("db/dbExample4.db","user=ET_takuya, scan=1 second")
dbLoadRecords("db/f3rp61_example.db","user=ET_takuya, no=1, scan=1 second")

# pcMonitor
dbLoadRecords("db/pcMonitor.template", "BOOTPC=CO_IOC:CORP61TEST")

cd ${TOP}/iocBoot/${IOC}
iocInit()
```

procServ

- EPICS Extension software
- ProcServ is like 'screen' command with telnet port.

procServ Web page (sourceforge)

http://sourceforge.net/projects/procserv/

- procServ is a wrapper that starts an arbitrary command (e.g. an EPICS soft IOC) as a child process in the background, connecting its standard input and output to a TCP port for telnet access. It supports logging, child restart (manual or automatic),...

Description from procServ Web page.

procServ options

```
% ./procServ --help
Usage: ./procServ [options] <port> <command args ... >    (-h for help)
<port>               use telnet <port> for command connections
<command args ...>   command line to start child process
Options:
    --allow            allow control connections from anywhere
    --autorestartcmd   command to toggle auto restart flag (^ for ctrl)
    --coresize <n>     sets maximum core size for child to <n>
 -c --chdir <dir>      change directory to <dir> before starting child
 -d --debug            enable debug mode (keeps child in foreground)
 -h --help             print this message
    --holdoff <n>      set holdoff time between child restarts
 -i --ignore <str>     ignore all chars in <str> (^ for ctrl)
 -k --killcmd <str>    command to kill (reboot) the child (^ for ctrl)
    --killsig <n>      signal to send to child when killing
 -l --logport <n>      allow log connections through telnet port <n>
 -L --logfile <file>   write log to <file>
 -n --name <str>       set child's name (defaults to command line)
    --noautorestart    do not restart child on exit by default
 -p --pidfile <str>    name of PID file (for server PID)
 -q --quiet            suppress informational output (server)
    --restrict         restrict log connections to localhost
    --timefmt <str>    set time format (strftime) to <str>
 -V --version          print program version
 -w --wait             wait for telnet cmd to manually start child
```

EPICS IOC start with procServ (again)

- startIOC_with_procServ.sh

```
### Environment setup
export BUSY_HOME=/usr/local/busybox
export EPICS_BASE=/proj/epics/R314/R31411/base
export PATH=$BUSY_HOME/bin:$BUSY_HOME/sbin:$BUSY_HOME/usr/bin:$PATH:$EPICS_BASE/bin/linux-f3rp6x/

IOCNAME="IOCCORP61TEST"
IOC_DIR="/users/takuya/epics314/epics314_f3rp61-201/iocBoot/iocTEST"

PROC_SERVE="/proj/epics/R314/R31411/extensions/bin/linux-f3rp61/procServ"

###PID_FILE="${IOCLOG_BASE}${IOCNAME}_procServ.pid"
PORT_NUM="20000"

SCRIPT="./${IOCNAME}.cmd"

echo cd ${IOC_DIR}
echo ${PROC_SERVE} -i ^D^C --allow ${PORT_NUM} ${SCRIPT}
cd ${IOC_DIR}
${PROC_SERVE} -i ^D^C --allow ${PORT_NUM} ${SCRIPT}
```

Execute command :
IOCCORP61TEST.cmd

procServ port number : 20000 (in this example)

allow option: procServ allow connections from anywhere.

Ignore key option : Ignore Ctrl-C and Ctrl-D key input.

Thank you.

谢谢

ありがとうございました