

Pulse Picker Project Log

Pulse Selector in the Hutches

- [Pulse Selector in the Hutches](#)
 - [Friday Jul 5th](#)
 - [Wednesday Jul 3rd](#)
 - [Tuesday Jul 2nd](#)
 - [Monday Jul 1st](#)
 - [TODO and issues in XCS](#)
 - [Alignment without pass by open shutter position \(MCode Version\):](#)
 - [MCODE programs \(Jeff version on Jun 26 2013\):](#)
 - [Working in Progress:](#)
 - [Single Shot Mode \(MODE 1\)](#)
 - [Continuous Mode \(MODE 2\)](#)
 - [Burst Mode \(MODE 3\)](#)
 - [Settings in XCS](#)
 - [Server running the IOC](#)
 - [Startup IOC](#)
 - [How to start the Pulse Selector Python script:](#)
 - [Configuration file](#)
 - [Current configuration \(Working in progress\)](#)
 - [Screens](#)
 - [TODO:](#)
- [Setup Lab tests](#)
 - [XIP Pulse Selector: Files, modes, schematics](#)
 - [Setup Lab: EVR Settings](#)
 - [Setup Lab: Working with the EVR from pslogin](#)
 - [Setup Lab: In case the server ioc-tst-cam5 was power cycled](#)
 - [Setup Lab : In case you need to power cycle ioc-tst-cam5](#)
 - [Setup Lab : Test Screens](#)
 - [Mode Descriptions](#)
 - [Homing Sequence](#)
 - [MCode Program State Diagram](#)
 - [<insert VISIO diagram>](#)
 - [<code example>](#)
 - [Motor Interlock Scheme](#)
 - [Screenshots](#)
 - [Testing Notes](#)

Friday Jul 5th

SOLVED:

1. Fast OPEN and CLOSE
2. :RESET_FLAG to monitor the finish reset process
3. :AD (Align Done) PV Flag to monitor the pre-mode alignment feature

IMPROVED:

1. PV's for OPEN and CLOSE range check
2. New class to provide EVR default settings (used inside Princeton script)
No need anymore python Picker script to set EVR.

TODO (lower priority):

1. Provide a XCS_Python home function on command line.

ISSUES:

When IOC reboot the Pulse needed to be Homed again.
We will investigate more on Monday.

Tests all mode changes, and not found problems.
Ready to do a complete test on Monday.

Wednesday Jul 3rd

SOLVED:

1. Integrate 4 shots instead of 3 (using ccdtake(3, 1, bShutter=True)
2. missing some shots when in 2 shots mode. SOLVED (USING MODE 3.2)
3. Switch from mode anymode to anymode OK

IMPROVED:

1. PV's for OPEN and CLOSE range check
2. Pre-mode alignment subroutine.

TODO:

1. Fast OPEN and CLOSE
2. RESET_PG Flag as PV (to monitor the finish reset process)
3. ALIGN Flag as PV (monitor the pre-mode alignment feature)

Tuesday Jul 2nd

Initial XRay aligned positions:

Y position: 0.0000 mm

X position: 0.1512 mm

Test examples

Single shot 10 images

```
ccdtake(1,10, bShutter=True)
```

Burst mode 3 shots 100 images

```
ccdtake(3,100, bShutter=True)
```

Alignment Laser used to test the system

1. To insert the laser

```
In [2]: r12.movein()
```

2. To remove the laser

```
In [2]: r12.moveiout()
```

Results: Tests **doesn't** work satisfactory. Problems found in Mcode:

1. Integrate 4 shots instead of 3 (using ccdtake(3, 1, bShutter=True)) OK
2. missing some shots when in 2 shots mode. SOLVED (USING MODE 3.2)
3. Stopping the program (:RESET_PG 1) took long time than before 1-3 sec more)
Needed one variable to check if RESET_PG is completed or if is OK to change mode.
4. Switch from mode 1 to 3 still doesn't work. Example:
ccdtake(1, 1, bShutter=True) **WORKS**
5. ccdtake(3, 1, bShutter=True) **FAILS!**
Still needs to do HOMING
6. Fast OPEN and CLOSE **doesn't work.**

Monday Jul 1st

New tests with new ioc version (new MCode version)

Features:

- New MODE (MODE 3.3)
 - One EVR Trigger OPEN, Next EVR Trigger CLOSE **OK IT WORKS**
- Fast OPEN and CLOSE commands **NEED TO BE TESTED**

TO BE IMPLEMENTED IN THE MCODE **ASAP**:

- Start position insensitive
- HOME (Index Mark) position set to zero (**DIAL**)
- START position (insensitive) relative to HOME (**USER**)
- Record to report position (Encoder) as monitor at any time

TODO and issues in XCS

=====

in xcs-control:

caput XCS:SB2:MMS:09:SET_VE 0

modify reg/d/iocCommon/sioc/ioc-xcs-trigger-ims/startup.cmd

to point to working area,

ssh ioc-xcs-mot1

kill ioc (telnet localhost...)

boot ioc from reg/d/iocCommon/sioc/ioc-xcs-trigger-ims/startup.cmd

=====

to fast open/close:

caput XCS:SB2:MMS:09:MODE_OPEN 1 (C2 should be around 655)

caget XCS:SB2:MMS:09:SE_L (to be read 4)

caput XCS:SB2:MMS:09:MODE_CLOSE 1 (C2 should be around 0 or 2x655)

caget XCS:SB2:MMS:09:SE_L (to be read 5)

=====

1. START STOP mode3.3

SET_TG 2

SET_MODE 3

2. FAST OPEN/CLOSE PV

3. START POSITION INDEPENDENT:

1. ISSUE: No counting -> trigger "burst stop" sometimes
2. FIXME: Avoid arm/disarm
3. TODO: Benchmark time needed for position correction
4. FIXME: Start position dependent on present position
5. TODO: If
position correction takes $\leq 10\text{ms}$ then do each time
6. TODO: Home offset as PV
7. TODO: Make Aperture Angle as VAR and PV (to be changed as needed)
8. TODO: Dial zero @ Encoder Home marker
User zero @ "Aligned" position (means that RESET_PG reset only users) - to be discussed with Jeff
9. TODO: Fast Open/Close PV

1. NEXT: Uses First Trigger to Open, Second Trigger to Close

The PP listen then for just one Event Code.

Example of sequence:

84 1 -> PP
85 2 -> DAQ
85 1 -> DAQ
85 1 -> DAQ
84 1 -> PP
85 1 -> DAQ

Alignment without pass by open shutter position (MCode Version):

```

LB Y3      'Realignment checks
CL Y4      'Save current speeds
CL Y5      'Assign realignment speeds
R3=N3-50   'Lower upper deadband
R4=N4+50   'Upper lower deadband
CL Y2, C2<R4 'If C2 is below this db, move to 0
CL Y1, C2>R3 'If C2 above the db, move to 180
CL Y1, N9=0 'If open, just move to 0
CL Y6      'Restore PP speeds
AD=1       'Set align done flag
H 100
RT

LB Y1      'Reset far end, C2>N3-SD
EE=1       'Enable encoder
MA 8192    'Move to 180 degrees
H          'Wait till move is done
C1=0       'Reset microsteps register
C2=0       'Reset encoder register
EE=0       'Disable encoder for trigger mode
N9=1       'Flag successful close
H 100
RT

LB Y2      'Reset near zero, C2<N4+SD
EE=1       'Enable encoder
MA 0       'Move to 0 degrees
H          'Wait till move is done
C1=0       'Reset microsteps register
C2=0       'Reset encoder register
EE=0       'Disable encoder for trigger mode
N9=1       'Flag successful close
H 100
RT

LB Y4      'Save current speeds
V1=VM
V2=VI
V3=A
V4=D
RT

LB Y5      'Assign realignment speeds
VM=8192
VI=2000
A=2000
D=2000
RT

LB Y6      'Restore speeds
VM=V1
VI=V2
A=V3
D=V4
RT

```

MCODE programs (Jeff version on Jun 26 2013):

located in:

```
[working area]... modules/pcds_motion/current/pcds_motionApp/src/ims_bootup_V3.mcode
```

Working in Progress:

```

initial condition:
Motor is homing from outside:
  C2 = 0 C1= 0
  MN = ?
  MF = XXXX steps (known) forward
  MB = XXXX steps (known) backward
  MO = XXXX steps (known) open position

Idle routine:
looking for where to go... (mod 2, mod 3, mod 4, fastopen, fastclose)

checkset subroutine
check my pos:
  read C2
  MN=MB, C2=MF +/- DELTA (dead band)
  RT

  MN=MF, C2=MB +/- DELTA (dead band)
  RT

  (MA MF and MN=MF), C2=MO +/- DELTA (dead band)
  RT

  if I m in startup+:
    set next move to startup- (MB)
    return subroutine
  if I m in startup-:
    set next move to startup+ (MF)
    return subroutine
  if I m open (MO):
    MA to startup+ (MF)
    set next move to startup- (MB)

mod 2
call chekset
waiting for trigger
...

mod 3
call chekset
waiting for trigger
...

mod 4
call chekset
waiting for trigger
...

fastopen
MA to open pos

fastclose
if in open pos
MA to startup+

```

Single Shot Mode (MODE 1)

```

' Program Fragment: MODE_1 (Single Shot)

Mode 1.0 - Set GO to 1 to arm trigger. Single sweep from trigger.

```

Continuous Mode (MODE 2)

```
' Program Fragment: MODE_2 (Continuous)

Mode 2.0 - Immediately armed. Continuous sweep from trigger.
```

Burst Mode (MODE 3)

```
' Program Fragment: MODE_3 (Burst mode)

Before mode select, select sub-mode TG:
Mode 3.1 - TG=0, Single Burst: Set GO to 1, then only single burst sequence will execute

Mode 3.2 - TG=1, Continuous Burst: Execute burst sequence

Mode 3.3 - TG=2, Open/Close Burst: Single pulse to open, single pulse to close
```

Settings in XCS

Server running the IOC

```
ssh ioc-xcs-mot1
```

Startup IOC

```
su <authorized_user>
ssh ioc-xcs-mot1
sudo /reg/d/iocCommon/sioc/ioc-xcs-trigger-ims/startup.cmd
telnet localhost 30999
```

How to start the Pulse Selector Python script:

```
/reg/neh/home1/paiser/working/ioc/xcs/xip_pp/current/pyscripts/src/run_pp.sh
```

Configuration file

```
~/pp_xcs/pvlist.lst
```

Current configuration (Working in progress)

```
# -----
# Pulse Selector Description File
# -----
# Syntax:
# <TYPE>, <PVNAME|IOCNAME|SCRIPT>, <DESC> # some_more_comments
# Where:
# <Type>      : "SEQ" -> Sequencer
#               "EVR" -> EVR associated to sequencer
#               "PPM" -> Pulse Selector
#               "SVR" -> Motor and Pulse Selector Server
#               "IOC" -> Motor and Pulse Selector software IOC
#               "SPP" -> EDM screens for motion
#               "SEV" -> EDM screens for evr
# <PVNAME>    : PV base name
# <IOCNAME>    : Server name associated to Pulse Selector PVs
# <DESC>       : User description
# Notes:
# PVNAME or IOCNAME are not case sensitive.
# Line can be commented out by starting with '#' character.
# -----
SEQ, XPP:R35:IOC:SEQ,      XPP DAQ Sequencer                # FIXME To check
PLY, IOC:IN20:EV01,       XPP DAQ Sequencer play mode        # FIXME to check
EVR, XCS:R42:EVR:01,      VME EVR located in XCS Rack 42      # OK
IOC, ioc-xcs-trigger-ims, IOC running pulse selector         # OK
SPP, ppm_gui.sh,          EDM screen startup shell script for PP motor # OK
PPM, XCS:SB2:MMS:09,       Pulse Selector Motor              # OK
YTR, XCS:SB2:MMS:21,       Y translation motor                # OK
XTR, XCS:SB2:MMS:08,       X translation motor                # OK
SEV, evr_gui.sh,          EDM screen startup shell script for EVR # OK
SVR, ioc-xcs-mot1,        server running IOC for pulse selector # OK
```

Screens

PP and Motor GUIs :

```
/reg/neh/home1/paiser/working/ioc/xcs/xip_pp/current/pyscripts/ppm_gui.sh XCS:SB2:MMS:09 XCS:SB2:MMS:21 XCS:SB2:MMS:08
```

TODO:

```
# FIXME Configuration file default location
# TODO Autosave default rampup values
# TODO Fast close and open functions
# TODO Test Sequencer
```

Setup Lab tests

- [Pulse Selector in the Hutches](#)
 - [Friday Jul 5th](#)
 - [Wednesday Jul 3rd](#)
 - [Tuesday Jul 2nd](#)
 - [Monday Jul 1st](#)
 - [TODO and issues in XCS](#)
 - [Alignment without pass by open shutter position \(MCode Version\):](#)
 - [MCODE programs \(Jeff version on Jun 26 2013\):](#)
 - [Working in Progress:](#)
 - [Single Shot Mode \(MODE 1\)](#)
 - [Continuous Mode \(MODE 2\)](#)
 - [Burst Mode \(MODE 3\)](#)
 - [Settings in XCS](#)
 - [Server running the IOC](#)
 - [Startup IOC](#)
 - [How to start the Pulse Selector Python script:](#)
 - [Configuration file](#)

- Current configuration (Working in progress)
 - Screens
 - TODO:
- Setup Lab tests
 - XIP Pulse Selector: Files, modes, schematics
 - Setup Lab: EVR Settings
 - Setup Lab: Working with the EVR from pslogin
 - Setup Lab: In case the server ioc-tst-cam5 was power cycled
 - Setup Lab : In case you need to power cycle ioc-tst-cam5
 - Setup Lab : Test Screens
 - Mode Descriptions
 - Homing Sequence
 - MCode Program State Diagram
 - <insert VISIO diagram>
 - <code example>
 - Motor Interlock Scheme
 - Screenshots
 - Testing Notes

XIP Pulse Selector: Files, modes, schematics

Setup Lab: EVR Settings

Setup Lab: Working with the EVR from pslogin

1. in pslogin type (you need to have your securekeys):

```
ssh ioc-tst-cam5 /reg/neh/home1/paiser/bin/evr.sh EVR=TST:CAM:05:EVR
```

2. Then configure EVR as you need manually...

Setup Lab: In case the server ioc-tst-cam5 was power cycled

1. Connect to the server:

```
ssh ioc-tst-cam5
```

2. Run the startup command that contains EVR ioc in:

```
sudo /reg/d/iocCommon/sioc/ioc-tst-cam5/startup.cmd
```

3. You should see with psproc:

```
[paiser@ioc-xrt-xcscam04 ~ 12:18:20] psproc
PID  USER-ID  SIOC          COMMAND  HOSTNAME          PORT
4851  tstioc    caRepeater    procServ ioc-xrt-xcscam04  30000
5147  tstioc    ioc-tst-cam5  procServ ioc-xrt-xcscam04  30001
5150  tstioc    ioc-tst-cam5  procServ ioc-xrt-xcscam04  40000
```

Setup Lab : In case you need to power cycle ioc-tst-cam5

```
ipmitool -I lanplus -U ADMIN -P <you_should_know> -H ioc-tst-cam5-ipmi power status
ipmitool -I lanplus -U ADMIN -P <you_should_know> -H ioc-tst-cam5-ipmi power reset
```

Where: <you_should_know> is the standard ipmi password that you_should_know...

Setup Lab : Test Screens

Python homing script:

```
/reg/neh/home1/jsludvik/test-python/xip-home.py
```


XIP main GUI:
/reg/neh/home1/jsludvik/svn/trunk/ioc/xpp/xip_pp/current/motionScreens/xip_gui.sh

XPP motor GUIs:

```
/reg/g/pcds/package/epics/3.14/modules/pcds_motion/R2.3.4/launch-motor.sh XPP:TST:MMS:01  
/reg/g/pcds/package/epics/3.14/modules/pcds_motion/R2.3.4/launch-motor.sh XPP:TST:MMS:02  
/reg/g/pcds/package/epics/3.14/modules/pcds_motion/R2.3.4/launch-motor.sh XPP:TST:MMS:03
```

Startup.cmd:

```
/reg/d/iocCommon/sioc/ioc-xpp-trigger-ims/startup.cmd
```

Mode Descriptions

Mode 1: Single pulse
Mode 2: Continuous trigger

Mode 1 and 2 should be the same program, as the Seq will generate the pulses to the motor, and open/close operation shouldn't make any difference whether it is 1 or more. The motor sees a trigger and moves.

This mode is accessed by <PV>:RUN_MODE2

Mode 3: The motor opens on a pulse, counts N pulses and closes on the Nth pulse. The pre-trigger and close trigger should be factored in by the upper layer software to determine the proper N value to put into the motor. All the motor does is, open on N=1, count, then close on N=N.

This mode is accessed by <PV>:RUN_MODE3

Mode 1 was programmed like just mode 2 with a User "Trigger" enable/disable, which may or may not be useful for us. Sort of like 2 layers of enabling, versus just 1 for mode2

This mode is accessed by <PV>:RUN_MODE1

Reset Modes to start new mode:

<PV>:RESET

Homing Sequence

- 1) EPICS Motor Record: HOMF
- 2) Zero Position
- 3) Move Relative -76.25 degrees
- 4) Zero Position

MCode Program State Diagram

<insert VISIO diagram>

<code example>

Motor Interlock Scheme

Screenshots

Pulse Selector Test/Checkout Procedure (6/10/2013)

X motor calibration

- Found offset from X zero position: 5.8075 mm from + limit
- Offset for X zero position: 5.7332 mm from - limit
- Limits performed as expected, and in place before hardstops

Y motor calibration

- Offset from +Y limit to zero: 15.6689 mm
- Offset from -Y limit to zero: 7.9956 mm
- Limits performed as expected, and in place before hardstops

New Interlock:

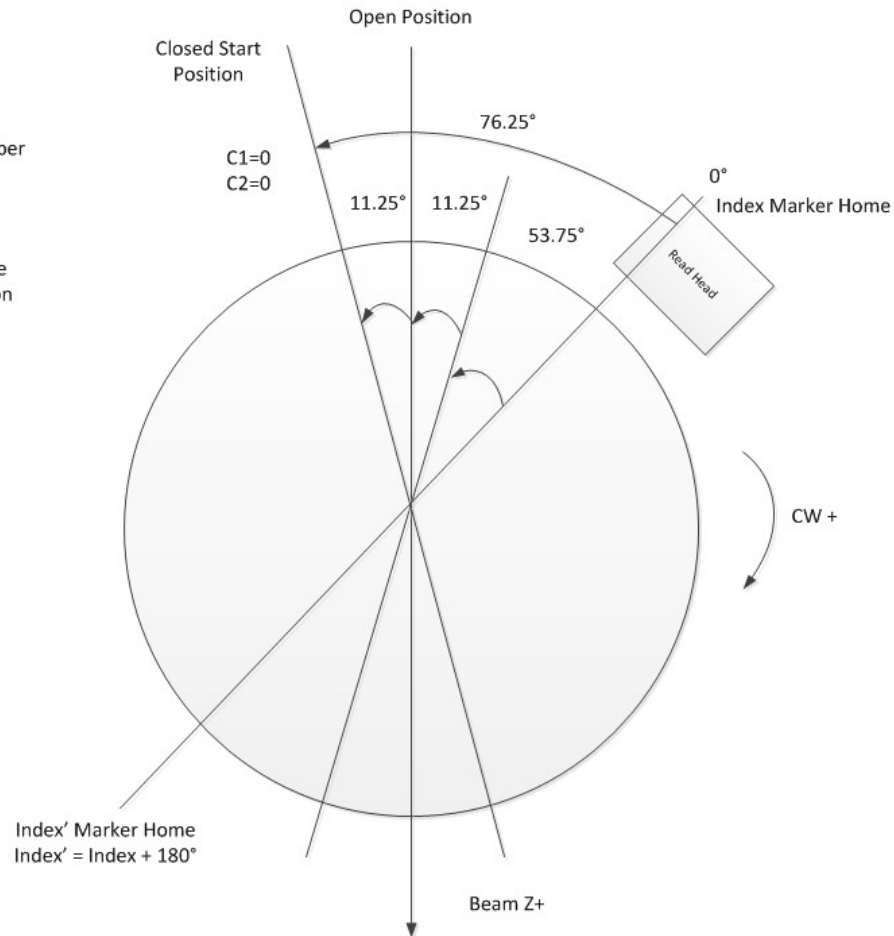
Add Y motor interlock, position ≥ 2 mm above 0 position, no X movement allowed

X-Y interlock stopped working today, need to investigate.

Pulse Selector Diagram
6/10/2013
J. Ludvik

View:
Looking Up from Chamber

3 Positions:
Index Marker Home
Closed Start Position
Open position



Testing Notes

Computational overhead:

- MR: ~2-3ms to compute a move
- Subtraction, assignment, followed by comparison/branch: ~2ms
- PR 5 character echo: ~2-3ms --> serial communication is asynchronous from the MCode
- :RESET_PG seems to be working OK (Ernesto mentioned that it used to take up to 2 seconds)