

# Fly and trajectory scans

Tim Mooney

2/27/2015

# Fly-scan choices

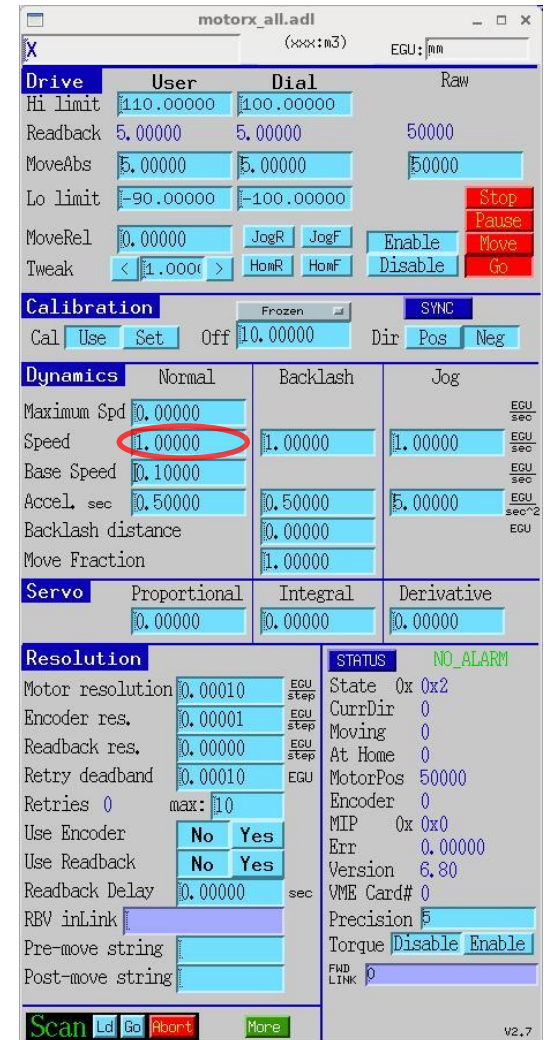
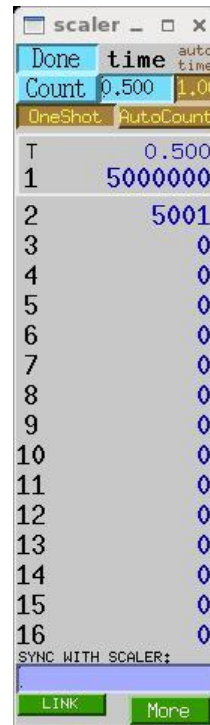
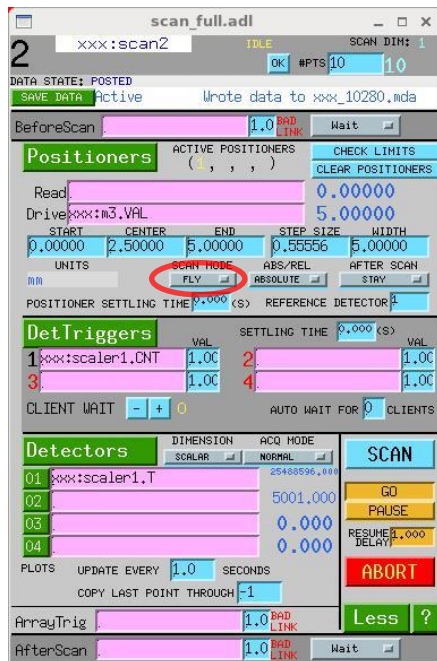
- Software fly scan
  - Data acquired while positioners move at constant speed
  - Detectors triggered by software
    - Periodically, or at user-specified time intervals
  - Positions acquired by software along with detector data
  - Few-ms dead time between data points
  - No cabling required
- Hardware fly scan
  - Data acquired while positioners move
  - Detectors triggered by pulses from positioner
    - Periodically
    - At user-specified positions
  - Positions implied or acquired by multichannel scaler
    - Arraycalc “cum” function reconstructs motor positions from scaler data
  - No dead time between points
  - Need cable from selected motor to selected detector(s)

# Hardware fly choices

- Constant speed (from now on, “*Hardware fly scan*”)
  - Positioner moves at constant speed
  - Positioner can be moved by motor record
  - Specify StartPos, EndPos, NPTS, and Speed
  - May be able to specify data-acquisition positions
    - Requires supported motor or external hardware
- Trajectory (from now on, “*Trajectory scan*”)
  - Motor moves along specified trajectory
  - Use trajectory controls
  - Specify trajectory positions and times
  - Specify data-acquisition positions
    - Details depend on controller
  - Requires supported motor

# Software fly scan

- Requirements:
  - positioner speed can be set
  - position updates periodically or on demand
  - scan manager (e.g., sscan record)
- User interface:
  - Differences from step scan



# Software fly scan

- Remarks:
  - Positioner speed and detector-dwell time must be set
  - Imprecise synchronization between data and *recorded* positions
  - Limited to  $<\sim 10$  Hz
- PVs: (values in green: user's choice)

## Same as for step scan:

$\$(scan).ACQT = SCALAR$

$\$(scan).P1PV = \$(motor).VAL$

$\$(scan).R1PV = \$(motor).RBV$

$\$(scan).T1PV = \$(scaler).CNT$

$\$(scan).DnnPV = scaler.T$

$\$(scan).P1SP = 0$

$\$(scan).P1EP = 5$

$\$(scan).NPTS = 10$

$\$(scaler).TP = .5$

## Modified for software fly scan:

$\$(scan).P1SM = FLY$

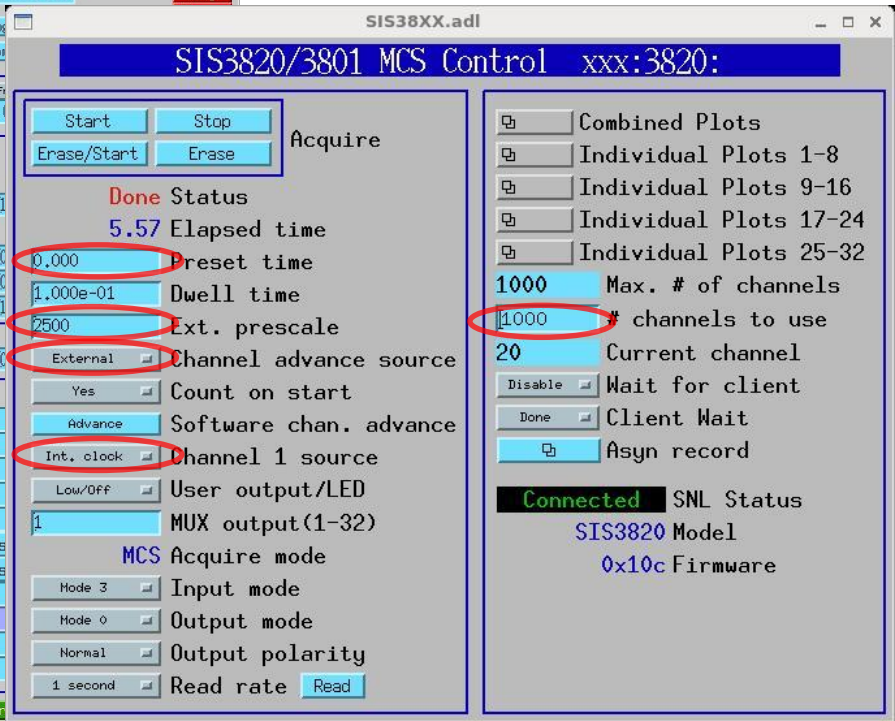
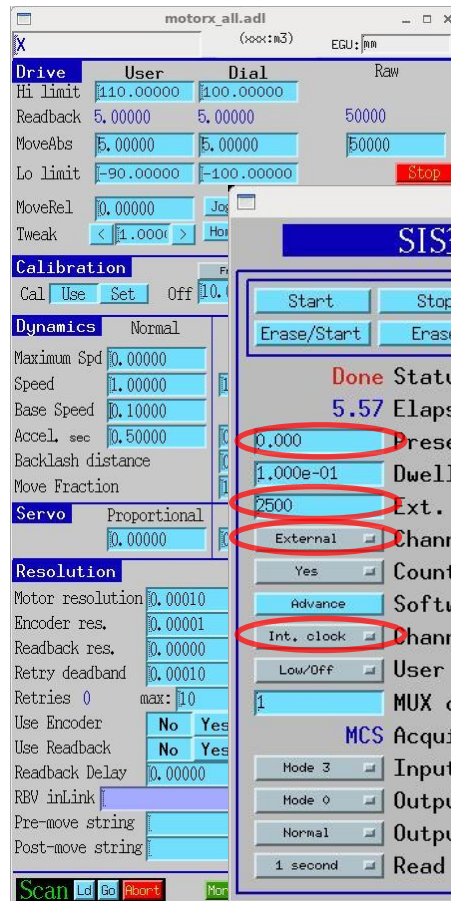
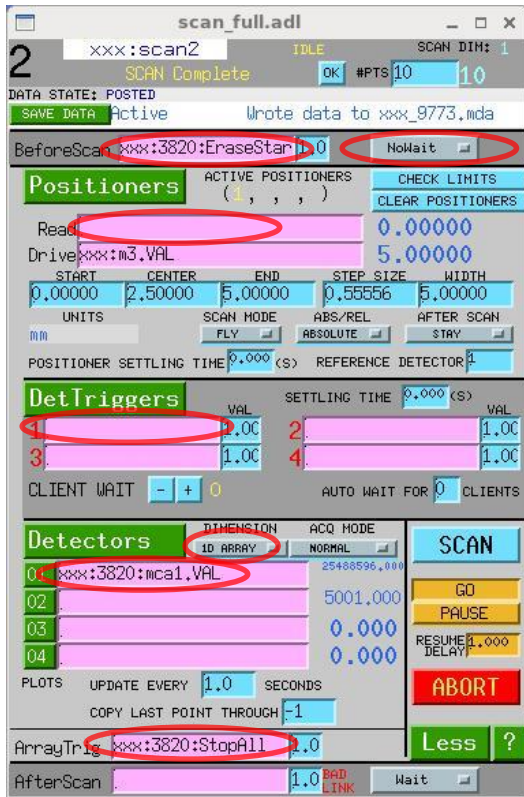
$\$(motor).VELO = 1.0$

# Hardware fly scan requirements

- Positioner can output pulses during nontrajectory move
  - Any stepper motor (with external divide-by-N)
  - Aerotech Ensemble with EnsemblePSOFly database
  - Probably other servo motors can do this
- Hardware-triggered detector can cache or stream scan data
  - MCS (*Struck* multichannel scaler)
  - XIA DXP
  - Some cameras
- Data-storage client
  - sscan record (saveData)
  - spec
  - areaDetector plugin

# Hardware fly scan

- User interface:
  - Differences from software fly scan:



# Hardware fly scan

- Remarks:
  - positioner speed must be set
  - detector must be prepared and started before motor moves

- PVs:

**Same as for software fly scan:**

`$(scan).P1PV = $(motor).VAL`

`$(scan).P1SM = FLY`

`$(scan).P1SP = 0`

`$(scan).P1EP = 5`

`$(scan).NPTS = 10`

`$(motor).VELO = 1.0`

**Modified for hardware fly scan:**

`$(scan).ACQT = 1D ARRAY`

`$(scan).BSPV = $(mcs):EraseStart`

`$(scan).BSWAIT = NoWait`

`$(scan).A1PV = $(mcs):StopAll`

`$(mcs):PresetReal = 0`

`$(mcs):ChannelAdvance = External`

`$(mcs):Channel1Source = Int. clock`

`$(scan).D01PV = $(mcs):mca1.VAL`

`$(mcs):Prescale = 2500`

`$(mcs):CountOnStart = Yes`

`$(mcs):NuseAll = 1000`

`$(scan).R1PV = not used`

`$(scan).T1PV = not used`



# Trajectory scan requirements

- Controller can move motor along trajectory
  - Newport MM4005 or XPS
  - In motor R6-9, Aerotech Ensemble or Pro-Dex (OMS) MAXv
- Controller can generate position-synchronized pulses
- Position-table generator
  - E.g., spec, arraycalc, python, etc.
- Hardware-triggered detector can cache or stream scan data
  - Same as hardware fly scan
- Data-storage client
  - Same as hardware fly scan

# Trajectory scan

- User interface:
  - Differences from hardware fly scan:

scan\_full.adl

DATA STATE: POSTED

SAVE DATA Active Wrote data to xxx\_9903.mda

BeforeScan prepForTraj 1.0 BAD LINK Wait

**Positioners** ACTIVE POSITIONERS ( , , ) CHECK LIMITS CLEAR POSITIONERS

Read: 0.00000  
Drive: 5.00000

START	CENTER	END	STEP SIZE	WIDTH
0.00000	0.00000	0.00000	0.00000	0.00000

POSITIONER SETTLING TIME 0.000 (S) REFERENCE DETECTOR 1

**DetTriggers** VAL SETTling TIME 0.000 (S) VAL

1	2	3	4
xxx:traj1:Execute 1.0C			

CLIENT WAIT - + 0 AUTO WAIT FOR 0 CLIENTS

**Detectors** DIMENSION ACQ MODE

ID ARRAY	1D ARRAY	NORMAL
01	xxx:3820:mca1.VAL	25488596.000
02		5001.000
03		0.000
04		0.000

PLOTS UPDATE EVERY 1.0 SECONDS COPY LAST POINT THROUGH -1

ArrayTrig prepData 1.0 BAD LINK

AfterScan 1.0 BAD LINK Wait

SCAN GO PAUSE RESUME DELAY 1.000 ABORT Less ?

SIS38XX.adl

SIS3820/3801 MCS Control xxx:

Start Stop Acquire  
Erase/Start Erase

Done Status  
5.57 Elapsed time

0.000 Preset time  
1.000e-01 Dwell time

1 Ext. prescale

External Channel advance source  
No Count on start  
Advance Software chan. advance  
Int. clock Channel 1 source  
Low/Off User output/LED  
1 MUX output(1-32)

MCS Acquire mode  
Mode 3 Input mode  
Mode 0 Output mode  
Normal Output polarity  
1 second Read rate Read

Combi  
Indiv  
Indiv  
Indiv  
Indiv

1000 Max  
1000 # c  
20 Cur  
Disable Wait  
Done Client Wait  
Asyn record

Connected SNL Status  
SIS3820 Model  
0x10c Firmware

MAX\_trajectoryScan.adl

traj1

# trajectory elements 2  
Trajectory definition Absolute  
# output pulses 200 Actual 0  
Range of pulses: Start 1 End 2  
Time mode Total  
Total time 10,000 ELAPSED 0.000 Plot  
Execution time scale 1.00  
Acceleration time 1.000 More

Move?	Current Pos	Start Pos	Plots
Z0 Yes	1.0000	0.5500	
ZI No	0.0000	0.0000	
X No	0.0000	0.0000	
MLLV:mYD No	0.0000	0.0000	
MLLV:mYU No	0.0000	0.0000	
MLLV:mRZ No	0.0000	0.0000	
MLLV:mZ No	0.0000	0.0000	
MLLV:mX No	0.0000	0.0000	

Command State Status

Build	Build	Done	Success
Build message	Done		
Simulate/Real	Real		
Execute	Execute	Done	Undefined
Execute message			
Abort	Abort!		
Readback	Readback	Done	Undefined
Read message			

# Trajectory scan

- Remarks:
  - Trajectory must be loaded
  - Detector must be prepared and started before motor moves
- PVs:

## Same as for hardware fly scan:

`$(scan).ACQT = 1D ARRAY`  
`$(mcs):PresetReal = 0`  
`$(mcs):ChannelAdvance = External`  
`$(mcs):Channel1Source = Int. clock`  
  
`$(scan).NPTS = 10`  
`$(scan).D01PV = $(mcs):mca1.VAL`  
`$(mcs):NuseAll = 1000`  
  
`$(scan).R1PV = not used`

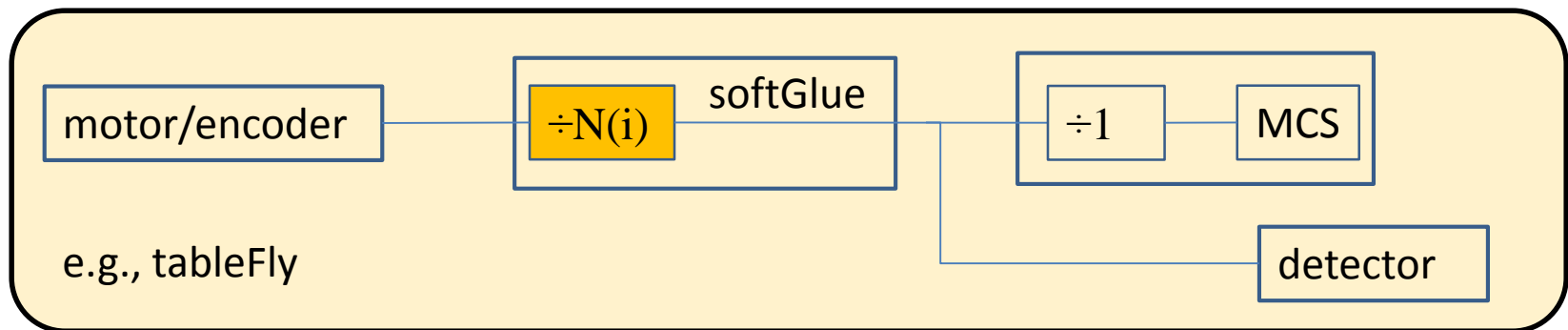
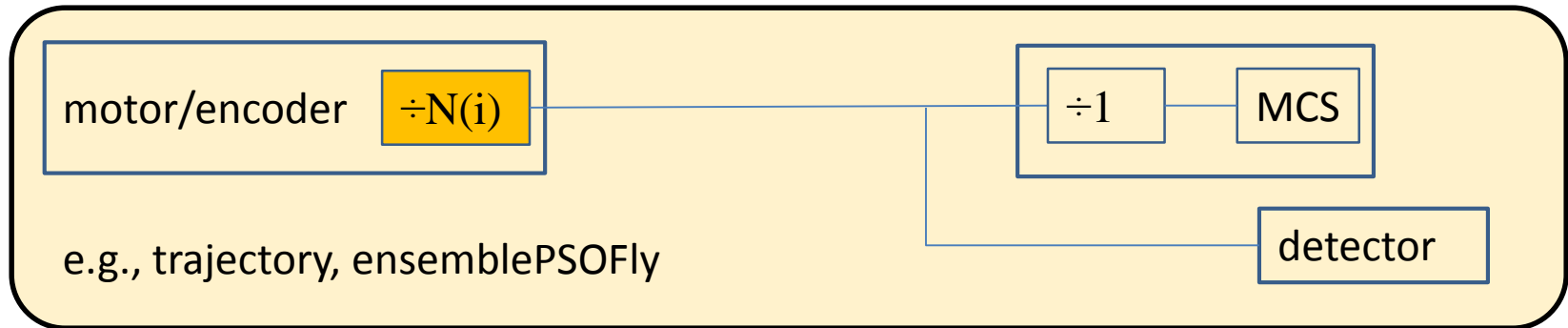
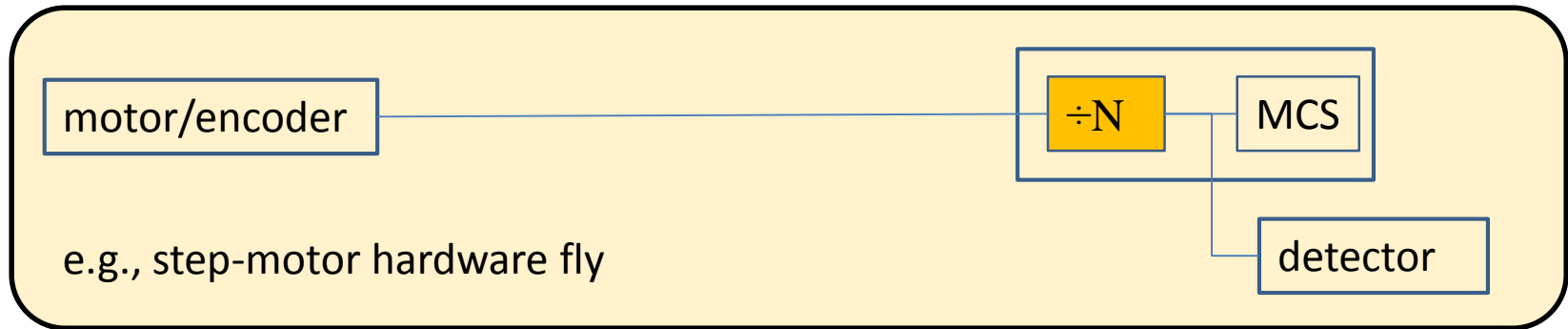
## Modified for trajectory scan:

`$(scan).T1PV = $(traj):Execute`  
`$(mcs):Prescale = 1`  
`$(scan).BSWAIT = Wait`  
  
`$(scan).BSPV = prepForTraj`  
`$(scan).A1PV = prepData`  
`$(mcs):CountOnStart = No`  
`$(traj):* = many choices`  
  
`$(scan).P1PV = not used`  
`$(motor).VELO = not used`  
`$(scan).P1SP = not used`  
`$(scan).P1EP = not used`  
`$(scan).P1SM = not used`

# Trajectory definition

- Number of trajectory elements
- Array of positions for each motor
  - Ensemble: only one motor
- Array of times
  - Can be specified as total time
- Number of output pulses, start/end element
  - MM4005: pulses evenly spaced in distance along trajectory
  - XPS: pulses evenly spaced in time
  - Ensemble: pulses evenly spaced in distance, or at trajectory points
    - Under development: at user-specified positions
  - For MAXv: pulses only at trajectory points
- Absolute/Relative/Hybrid position mode
  - Currently, Ensemble and MAXv don't support Hybrid mode
- MAXv has timing problems in very slow motion

# Detector-trigger options



# Examples

- 1ide hexFly (hard fly scan)
  - EnsemblePSOfly.db with evenly spaced data-gate signals
- 2bmb, 32idc tomography fly (hard fly scan)
  - EnsemblePSOfly.db with evenly spaced data-trigger signals
- 2bmb interlace fly (hard fly scan)
  - EnsemblePSOfly.db with user-specified data-trigger signals
    - tableFly.db uses softGlue to generate triggers from motor pulses
    - interlaceFly.db programs tableFly
  - Acquire at ~100 Hz for ~30 minutes
- 15idd USAXS fly (trajectory scan)
  - Ensemble and MAXv trajectories with user-specified data-trigger signals
  - Rotation stage (Ensemble) executes exponential trajectory
  - Translation stages (MAXv) execute commensurate trajectories
- Gradient mirror deposition system (trajectory motion)
  - Ensemble trajectory
  - support installed, but not connected to higher-level software

# Plans, possibilities

- Database support for “automatic” soft fly scans
- Database/softGlue support for “automatic” hard fly scans
- Ensemble multiple-motor trajectory support
- ID/monochromator fly scan
- Relax ensemblePSOFly motor-config constraints
- Implement *Hybrid* mode for Ensemble/MAXv trajectory
- Implement Ensemble and MAXv trajectory in model-3 driver