



Australian
Synchrotron

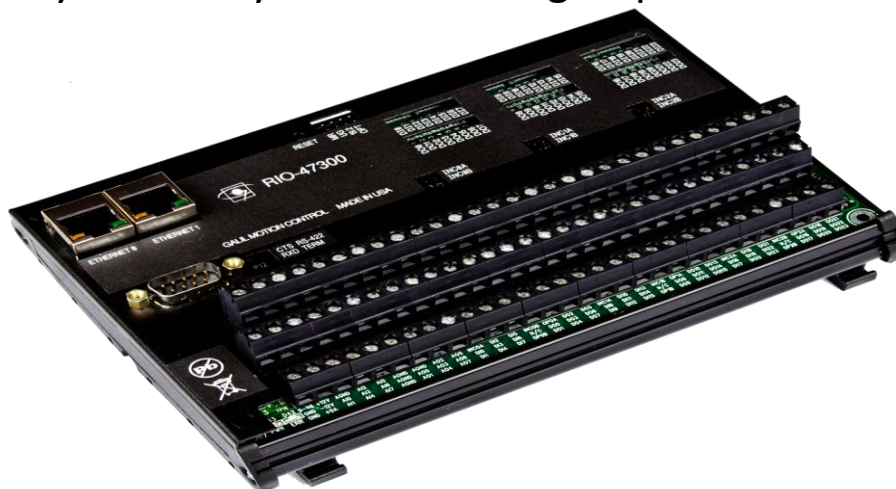
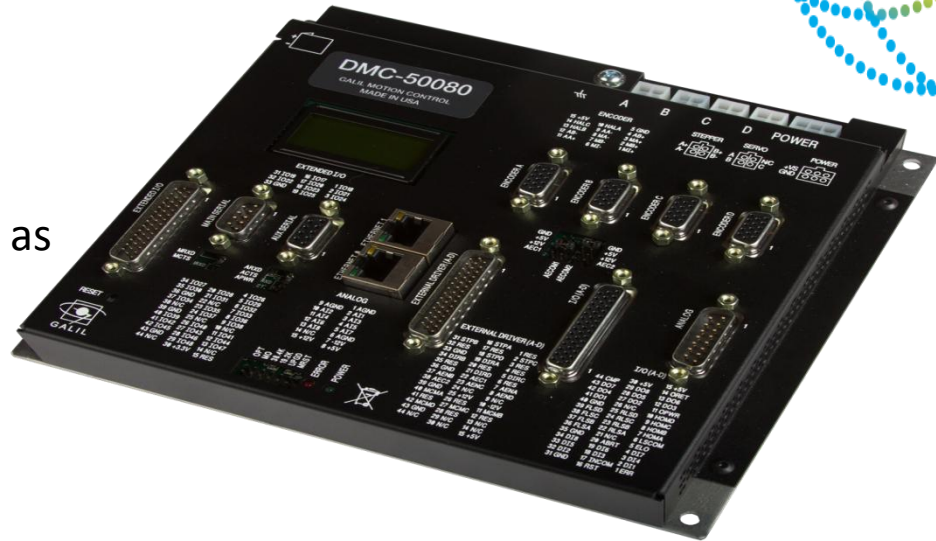
Turning bright ideas into brilliant outcomes



An EPICS solution for Galil products that can provide a comprehensive, and high performance motor and PLC control system for use at synchrotrons and other research laboratories

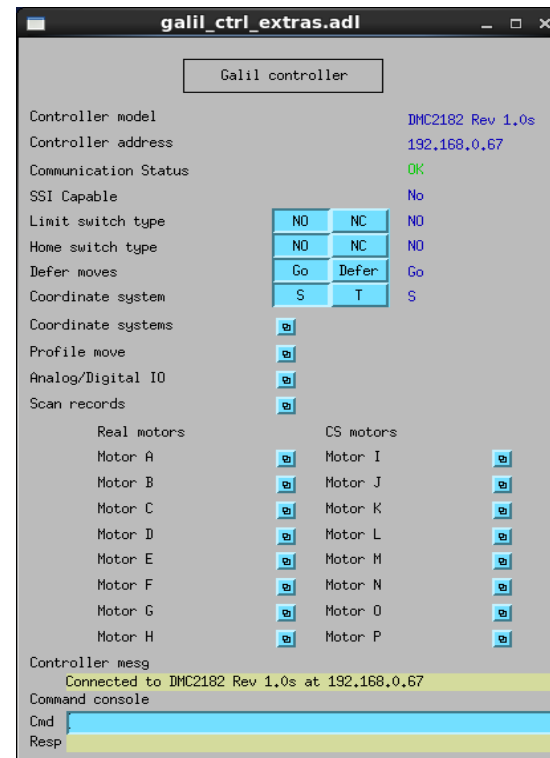
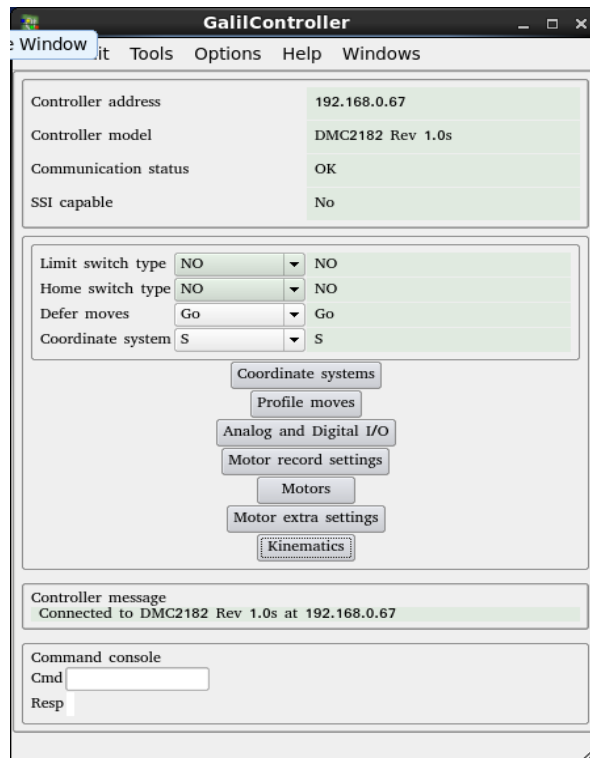
EPICS Galil Driver 3-1

- Replacement for Galil 1-5
- ASYN model 3 architecture
- Connects to all Galil controllers
- Uses EPICS motor record
- Adds a litany of new features such as
 - Auto amplifier on/off
 - Auto brake on/off
 - Profile motion
 - Coordinate system motors
- Simple and intuitive to use
- Provides high performance and flexibility
- The easy way to meet your demanding requirements



Bundled software

- Example IOC
- Autosave request files
- QE framework and MEDM screens
- QE screens are based on Qt and provide for a more intuitive experience
 - Especially true for the more complex functions (eg Kinematics definition)



Communications

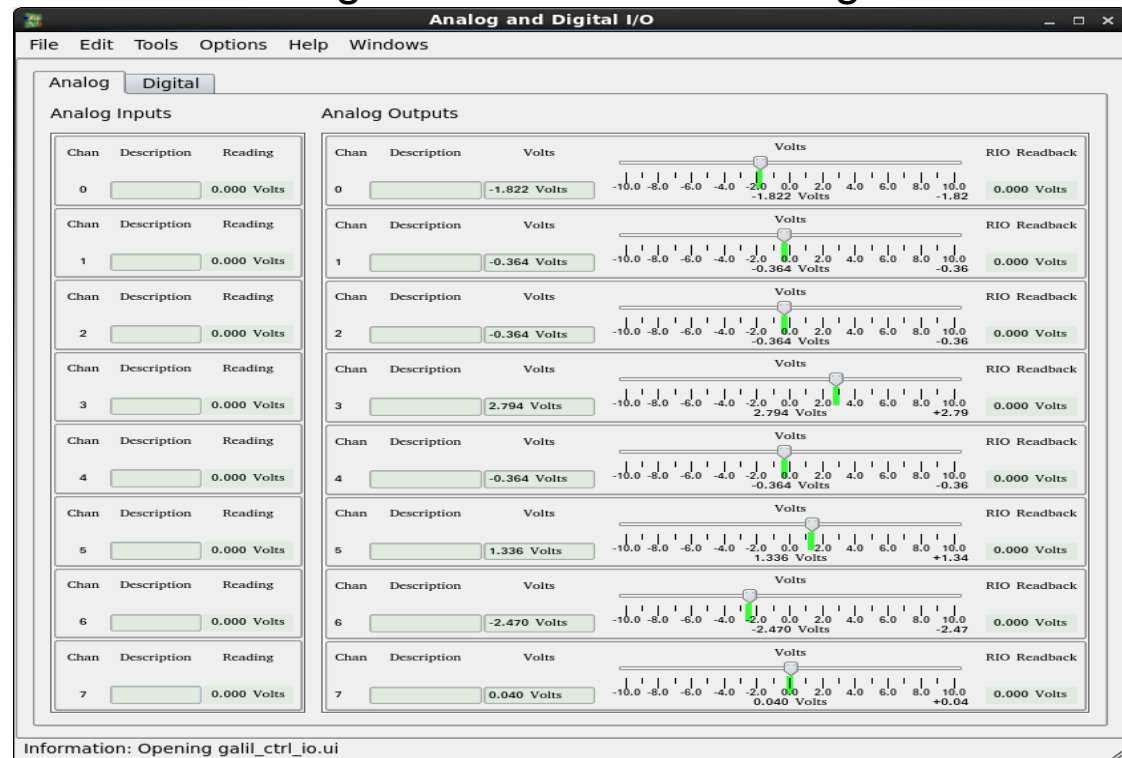
- ASYN based communications
- Two connections per controller
 - TCP Synchronous command/response
 - UDP Asynchronous stream
- Serial also supported
- Time between UDP packets
 - RIO PLC – 2ms
 - DMC-2xxx-8ms
 - DMC-4xxx-2ms
- Sophisticated connection management
- Manage controller code

Low level programming

- Sophisticated code generator
 - Behaviour when limits activated, home program
 - Provided home program is feature rich and features can be enabled/disabled (eg. Use encoder index, jog after home to).
- Most cases require no code at all
- Supports construction of code from template
 - Header, axis1, axis2, .., footer
- Supports complete custom code and will manage it's delivery to controller

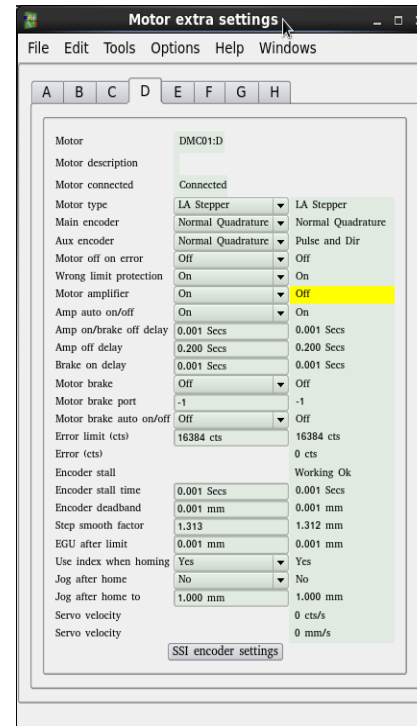
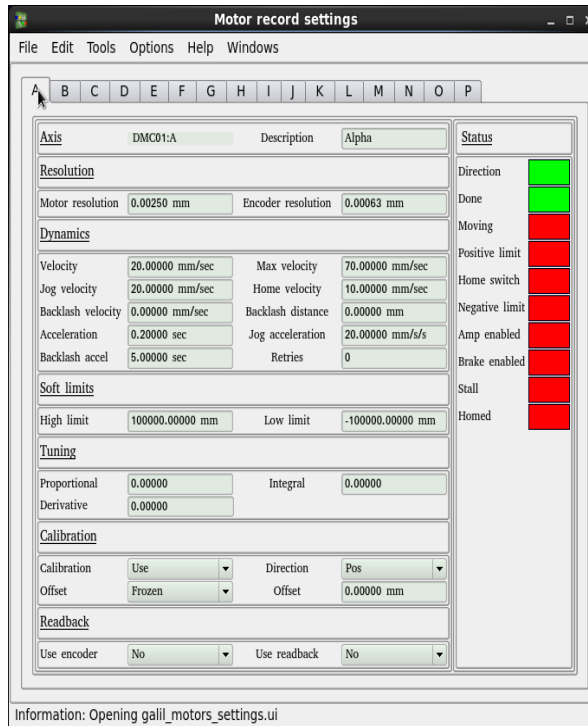
RIO PLC control

- Cost effective option for general control < \$500 USD
- Dual ethernet
- Provides access to digital and analog IO
- Example IOC provides access to 24 Digital In/Out and 8 Analogs
 - ADC 16 bit option
- 2ms UDP updates



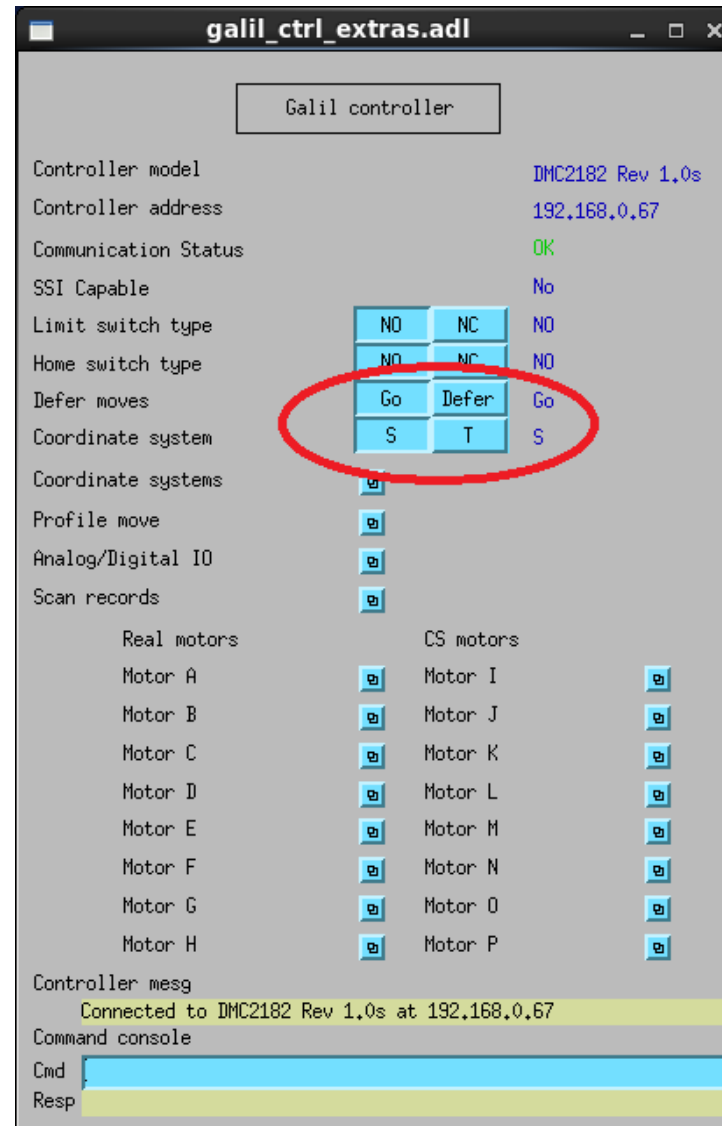
Standard motor control

- Motor record used for both real, and CS motors
- Standard motor records settings are available, but not all apply to CS motors (eg. use encoder if present)
- Additional features for real motors:
 - Wrong limit protection
 - Encoder stall definition

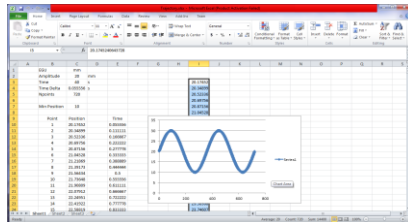
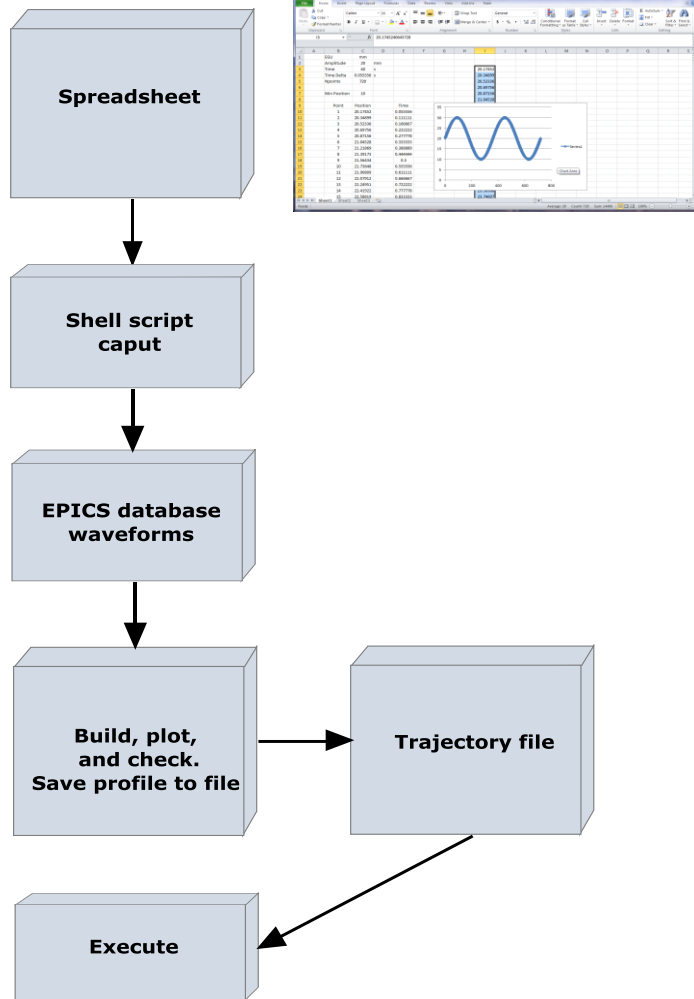


Ad-hoc coordinated motion

- Deferred moves facility
 - Select deferred
 - Select coordinate system
 - Move all motors
 - Select go
- Motion coordinated by hardware



Profile moves



galil_profileMove.adl

Galilprofile

Profile points: 1441 Current: 1440

Time mode: Fixed Plot time: [More]

Fixed time per point: 0.050

Output compare 1 Axis: OFF OFF Servo only

Start position: 1.000 (EGU) Then every: 1.000 (EGU)

Output compare 2 Axis: OFF OFF Servo only

Start position: 1.000 (EGU) Then every: 1.000 (EGU)

Trajectory file: TrajectoryScan.trj

Message: Output compare 2 turned off

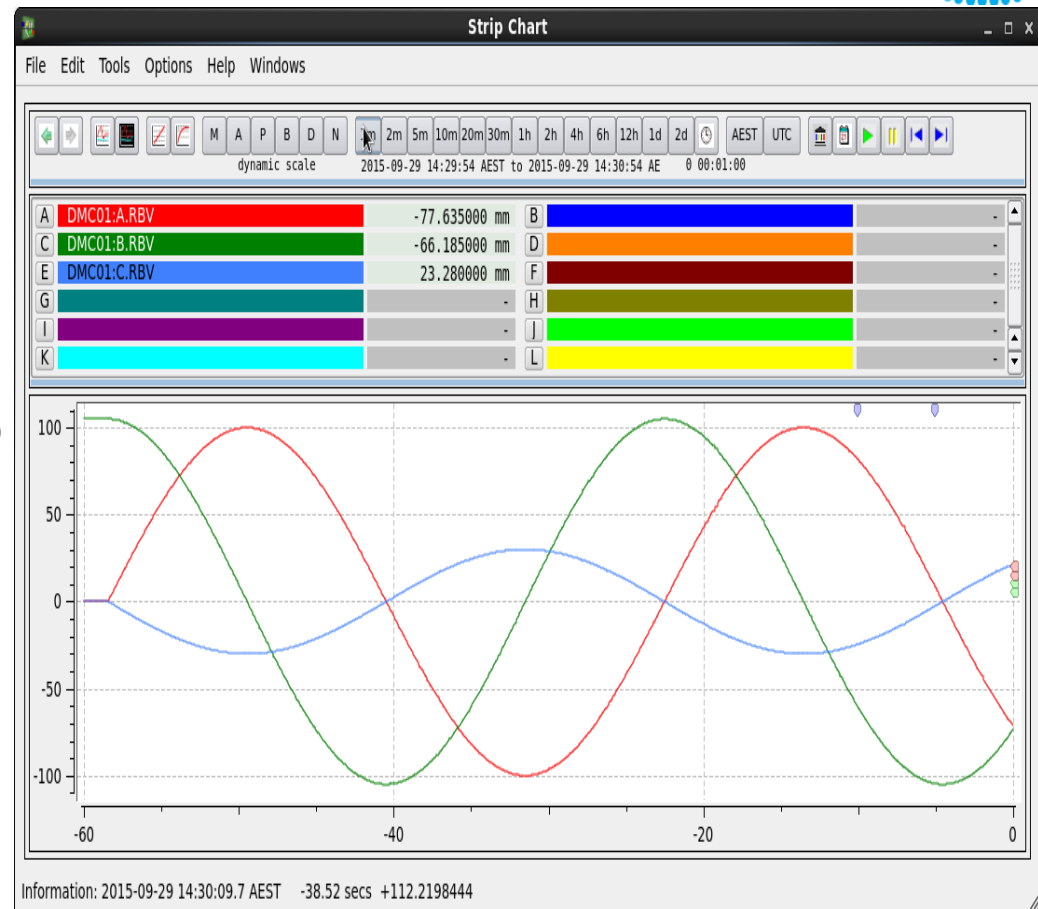
	Move axis?	Move mode	Current Pos.	Plots
Alpha	Yes	Absolute	0.00000	[More]
Beta	Yes	Absolute	105.00000	[More]
	Yes	Absolute	0.00000	[More]
	No	Relative	0.00000	[More]
	No	Relative	0.00000	[More]
	No	Relative	8.95000	[More]
	No	Relative	0.00000	[More]
	No	Relative	0.00000	[More]

Command	State	Status
Build	Build	Done Success
Execute	Execute	Done Success Abort!

Message: Profile completed successfully

Profile moves

- Available on 21x3 model and above
- From spreadsheet to delivery takes only minutes
- Tools provided to analyse the trajectory and ensure it's within constraints
- Relative and absolute modes
- Position compare can be used to trigger external detectors



Coordinate system (CS) motors

- 8 coordinate system motors per controller. Axis I to P.
- 10 variables for use in kinematics
- Coordinate system motors contain kinematic methods
 - Parse equations
 - Forward transform
 - Reverse transform
- Kinematics and variables all database adjustable whilst IOC running
- Multi-mode coordinated motion easily implemented
- Nesting allowed

Kinematics

File Edit Tools Options Help Windows

Kinematic variable values

Q=		=	0.00000
R=		=	0.00000
S=		=	0.00000
T=		=	0.00000
U=		=	0.00000
V=		=	0.00000
W=		=	0.00000
X=		=	0.00000
Y=		=	0.00000
Z=		=	0.00000

Forward transforms

CSMotor forward transform equations (readbacks)
CS Motor = Description = real motors (eg. I=B-A)

DMC01:I	=	I Forward transform	=	(A+B)/2
DMC01:J	=	J Forward transform	=	B-A
DMC01:K	=	K Forward transform	=	(C+D)/2
DMC01:L	=	L Forward transform	=	D-C
DMC01:M	=	M Forward transform	=	I+K
DMC01:N	=	N Forward transform	=	J+L
DMC01:O	=	O Forward transform	=	
DMC01:P	=	P Forward transform	=	

Reverse transforms

CSMotor reverse transform equations (new setpoints)
Real motor = Description = CSMotors (eg. A=I-J/2)

I J K L M N O P

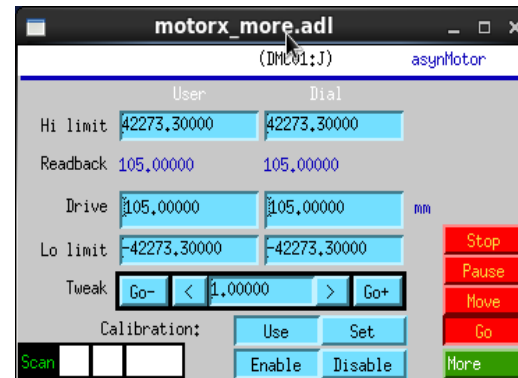
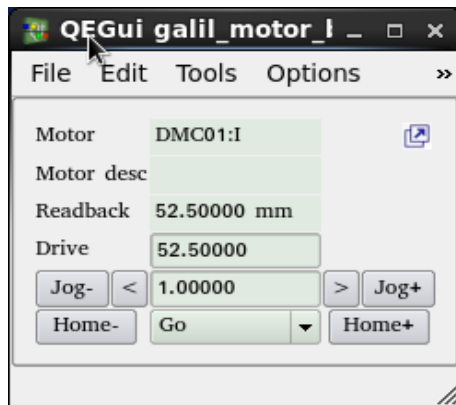
DMC01:A	=	O Reverse transform A	=	
DMC01:B	=	O Reverse transform B	=	
DMC01:C	=	O Reverse transform C	=	
DMC01:D	=	O Reverse transform D	=	
DMC01:E	=	O Reverse transform E	=	
DMC01:F	=	O Reverse transform F	=	
DMC01:G	=	O Reverse transform G	=	
DMC01:H	=	O Reverse transform H	=	

Controller message
Connected to DMC2182 Rev 1.0s at 192.168.0.67

Information: Opening galil_kinematics.ui

Coordinate system (CS) motors

- Mix closed-loop and open loop motors
- Mix different gear ratios, and resolution
- Limits reporting
- Underlying kinematics model could be used on any controller that supports coordinating groups of motors



Conclusion

- Download <http://motorapp.github.io/Galil-3-0/>
- Demo at Motion Solutions Australia booth
- Questions?