Open Source Motion Control

Based on the Etherlab open source EtherCAT master (www.etherlab.org)

Anders Sandström
Torsten Bögershausen
ESS Motion Control and Automation Group

Outline

EtherCAT fieldbus

Architecture

Axis Object

Trajectory Object

Monitoring Object

Basic Kinematics

Opportunities / Challenges

Summary

Acknowledgments

therCAT Fieldbus

therCAT = **Ether**net for **C**ontrol **A**utomation **T**echnology

)pen fieldbus standard originally developed by Beckhoff GmbH

Iaintained by EtherCAT Technology Group (www.EtherCAT.org).

ardware requirements:

- –**Master: standard computer hardware (NIC)**
- –**Slaves: dedicated hardware, EtherCAT Slave Controller (ESC)**

Iasters: Several commercial and **open source** masters available

laves: Several 100 manufacturers of slaves (drives, I/O, sensors, robots)

opologies: Line, Star, Ring

Iedia: Cat 5 cable, plastic fiber, glass fiber

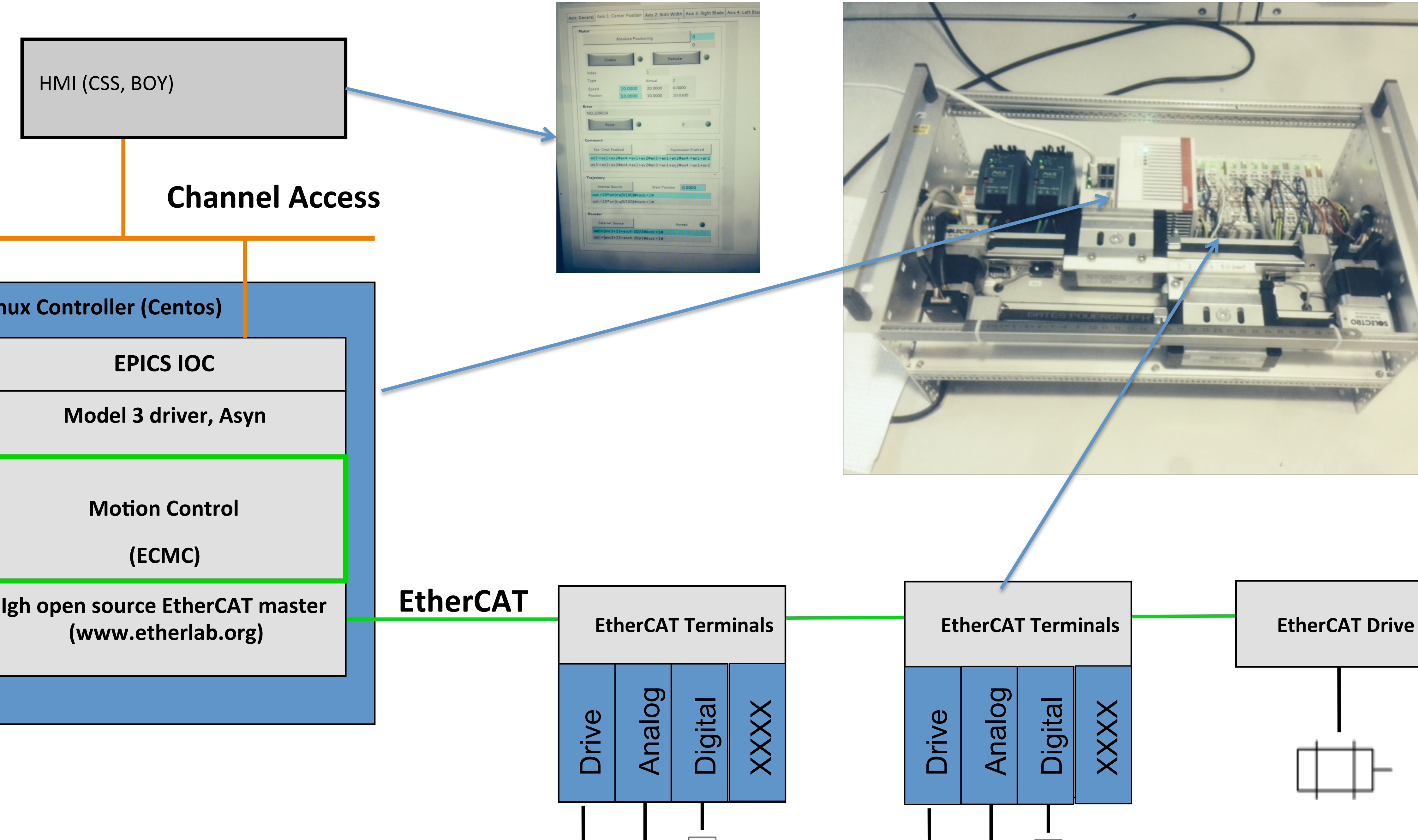
upports Distributed Clock (DC) in slaves

andwidth utilization: 80%-97% (100 Mbit/s , Ethernet, Full-Duplex)

pplications: **Motion, large or long distance systems, synchronized systems**

ycle times > 50µs

rchitecture



Index 1

Type Virtual 2

Speed 20.0000 20.0000 0.0000

Position 10.0000 10.0000 10.0300

NO_ERROR

ex3:=ex1+ex2#ex4:=ex1+ex2#en3:=en1+en2#en4:=en1+en2

ex3:=ex1+ex2#ex4:=ex1+ex2#en3:=en1+en2#en4:=en1+en2

Start Position 0.0000

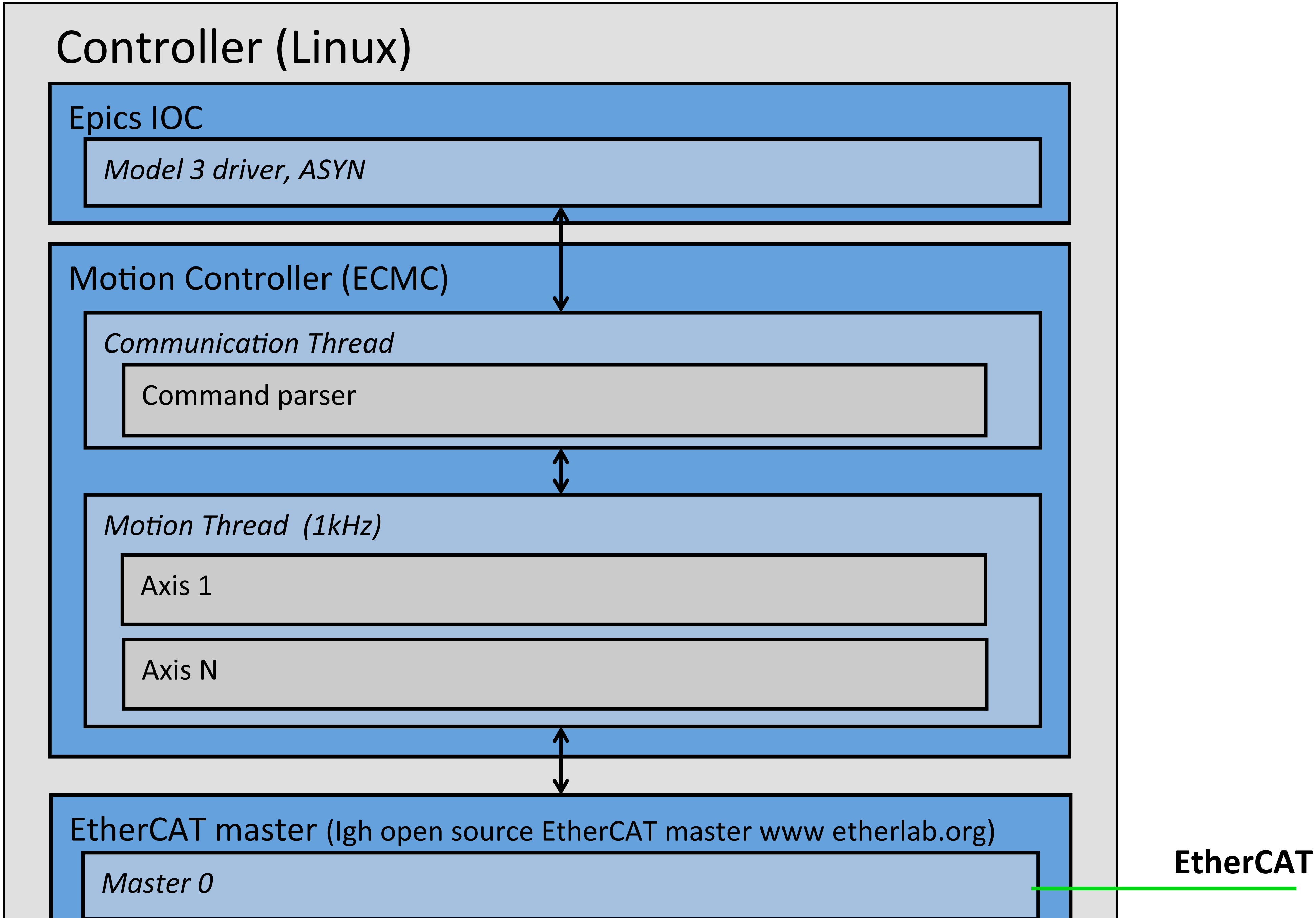
out:=10*sin(traj5/100)#ilock:=1#

out:=10*sin(traj5/100)#ilock:=1#

out:=(enc3+22+enc4-20)/2#ilock:=1#

out:=(enc3+22+enc4-20)/2#ilock:=1#

Motion Controller



Controller (Linux)

Epics IOC

Model 3 driver, ASYN

Motion Controller (ECMC)

Communication Thread

Command parser

Motion Thread (1kHz)

Axis 1

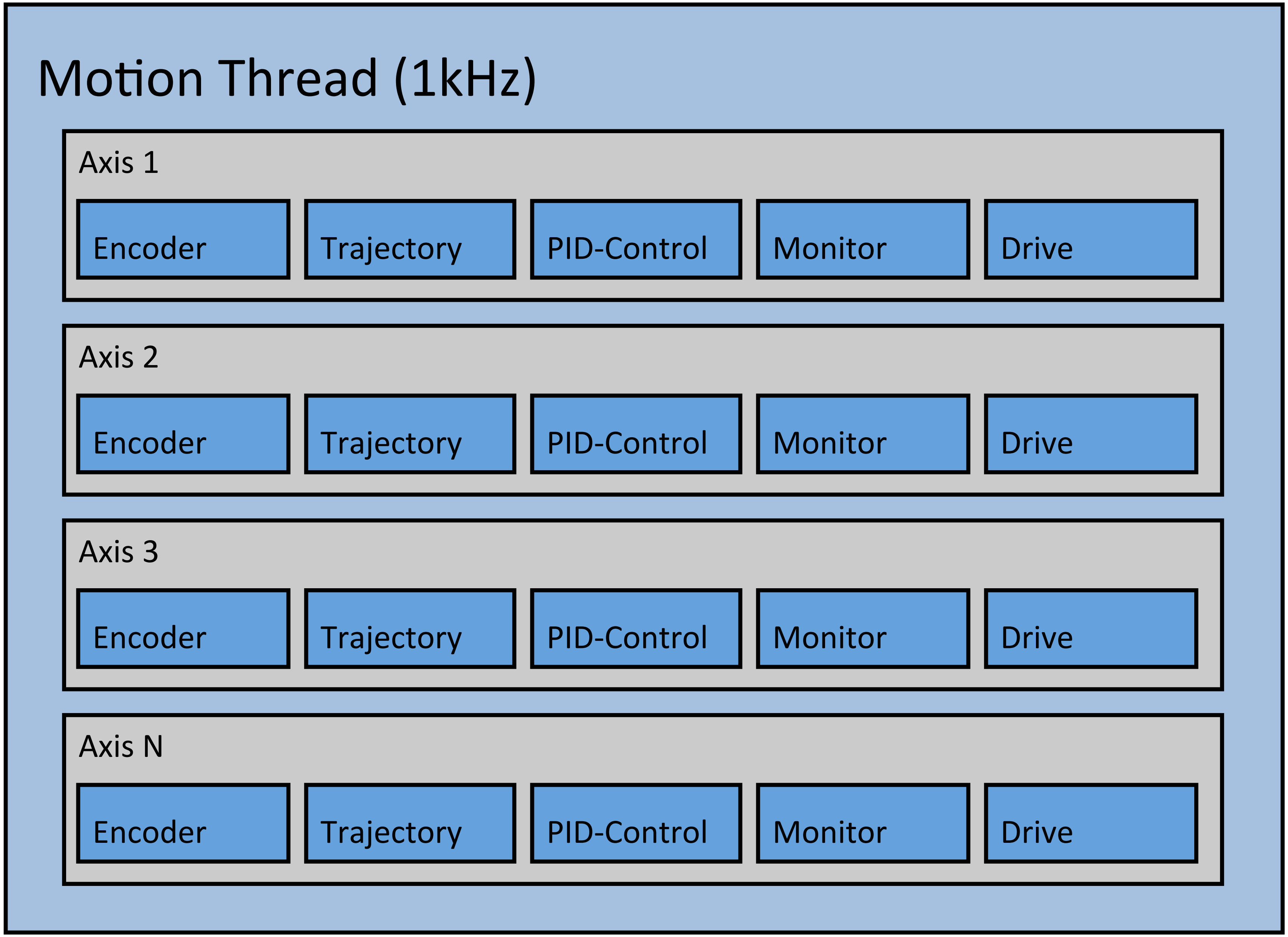
Axis N

EtherCAT master (Igh open source EtherCAT master www etherlab.org)

Master 0

EtherCAT

Motion Thread



Motion Thread (1kHz)

Axis 1

Encoder | Trajectory | PID-Control | Monitor | Drive

Axis 2

Encoder | Trajectory | PID-Control | Monitor | Drive

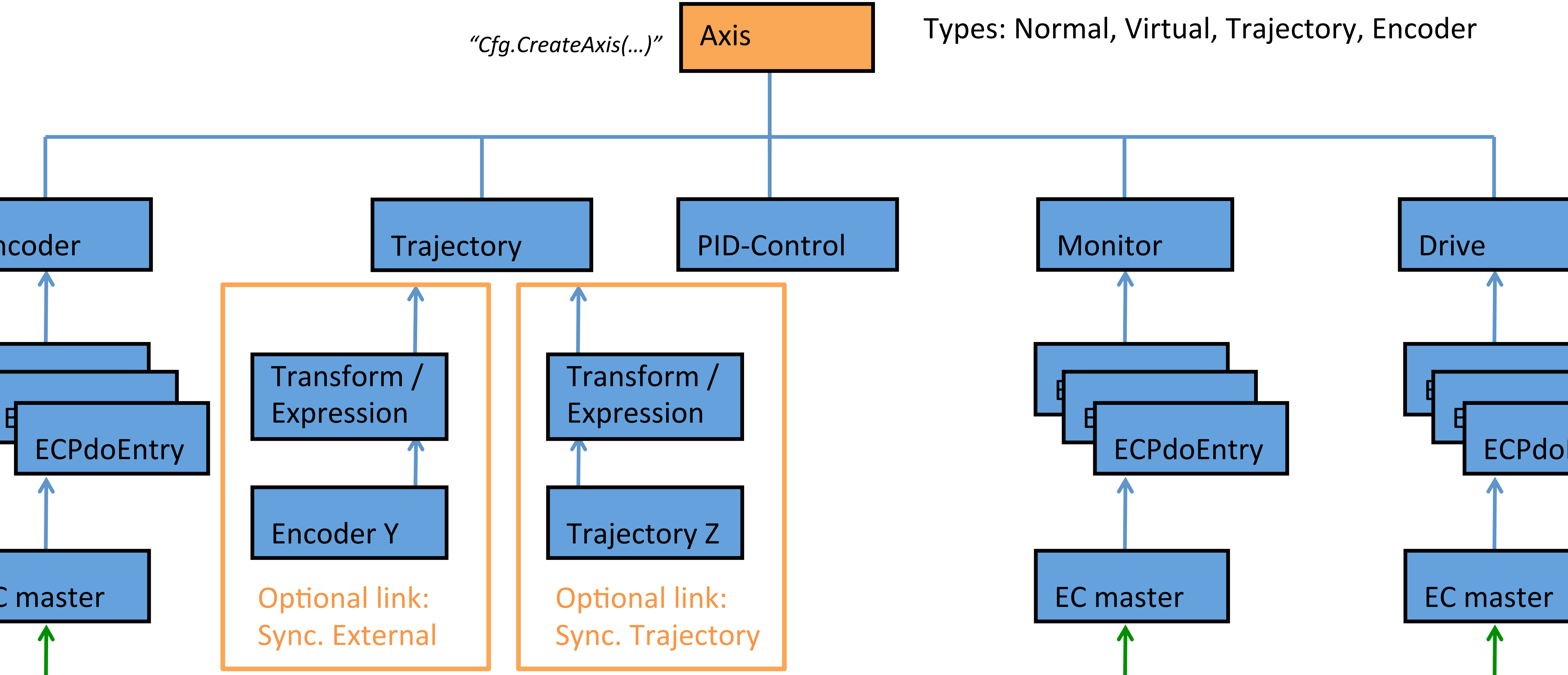
Axis 3

Encoder | Trajectory | PID-Control | Monitor | Drive

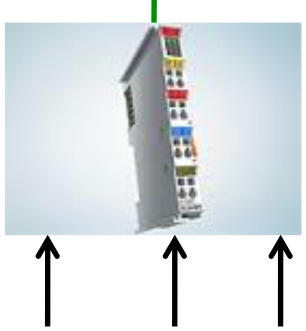
Axis N

Encoder | Trajectory | PID-Control | Monitor | Drive

tructure: Axis object



- Executes and links objects:
  - Encoder
  - Trajectory Generation
  - PID-Controller
  - Monitor
  - Drive
- Sequences



rajectory object

Trapezoidal trajectories

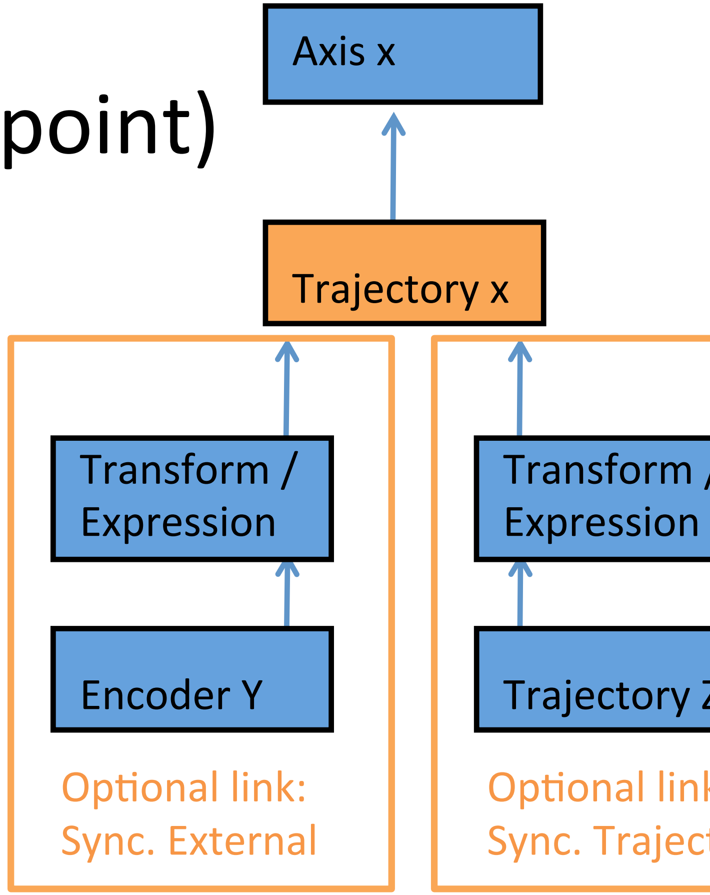
Feed Forward (position and velocity set-point)

Motor Record Support

- Constant velocity
- Relative positioning
- Absolute positioning
- Homing
- Hard and soft limits

Synchronization / Kinematics

- Absolute/Relative
- Gear
- Expressions
  - Other axis encoder
  - Other axis trajectory set-point



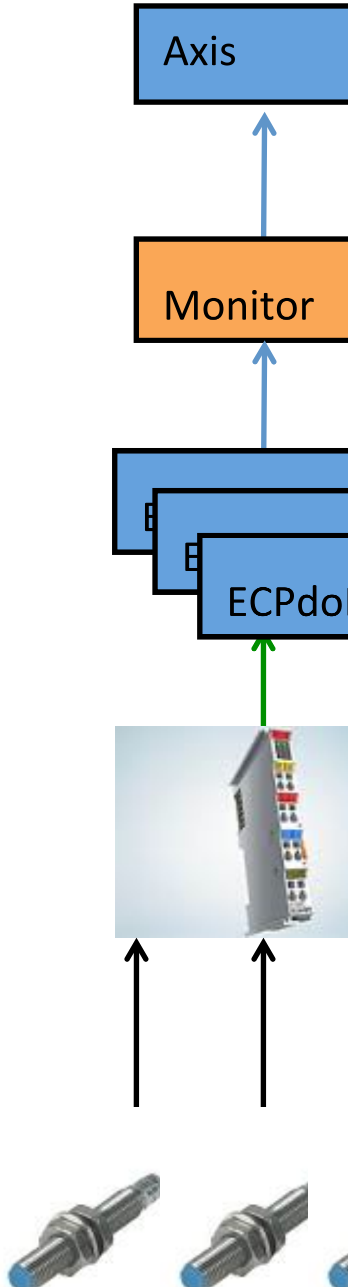
Monitor object

Position lag monitoring

“At Target” Monitoring

Limit switches

Over Speed Monitoring



inematics

Synchronization of axes by expressions (exprTK*)

- Set-points
- Actual values
- Amplifier enable
- Interlocks (stop at problem)

Update of expression at runtime possible

Expressions evaluated in 1kHz (synchronization)

aj*x* = Trajectory generated setpoint for axis *x*

nc*x* = Actual position of axis *x*

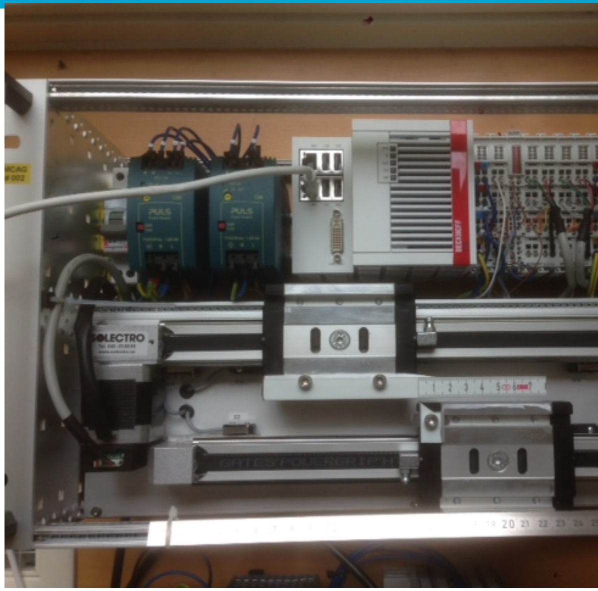
n*x* = *Enable of axis x*

x = *Interlock of axis x*

xample:

```
aj1:=10*sin(traj2+enc3);
:=il2 and il5 and enc4>enc3;
```

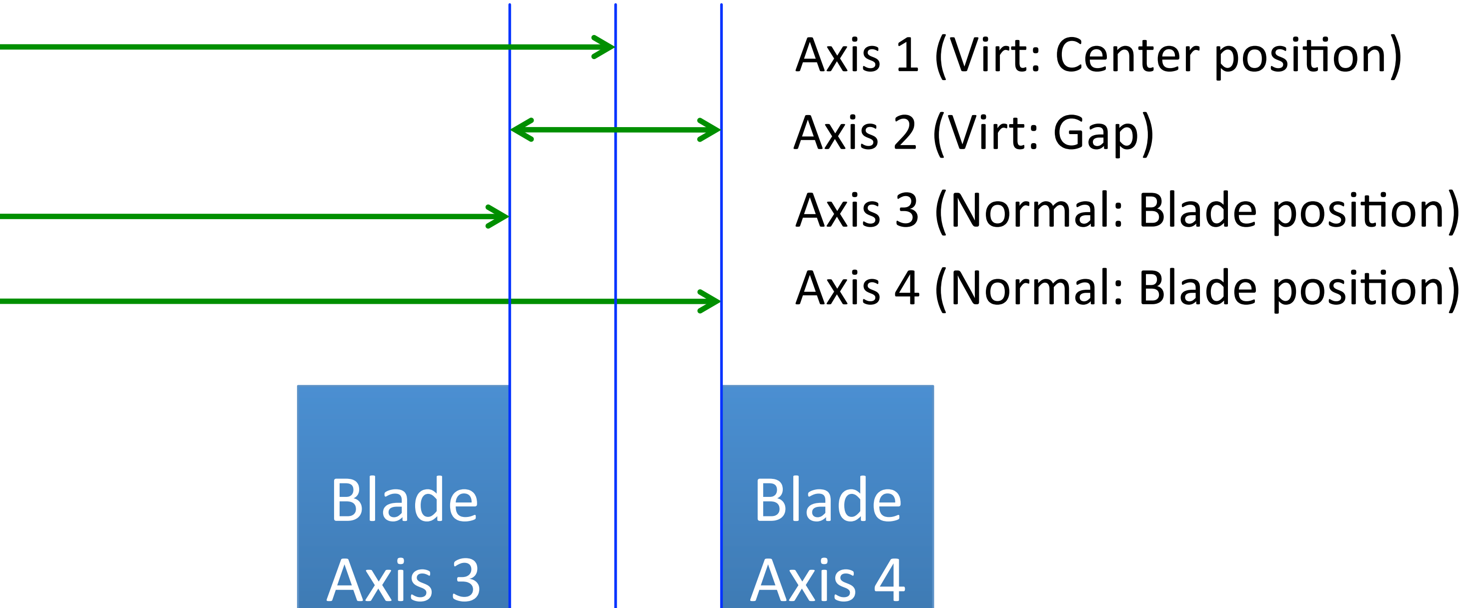
inematics Example: 2 axes slit set



2 virtual axes

- Slit center position
- Slit gap

2 normal axes (blade positions)



Axis 1 (Virt: Center position)

Axis 2 (Virt: Gap)

Axis 3 (Normal: Blade position)

Axis 4 (Normal: Blade position)

Blade Axis 3

Blade Axis 4

Forward Kinematics:

traj3:=traj1-traj2/2;

traj4:=traj1+traj2/2;

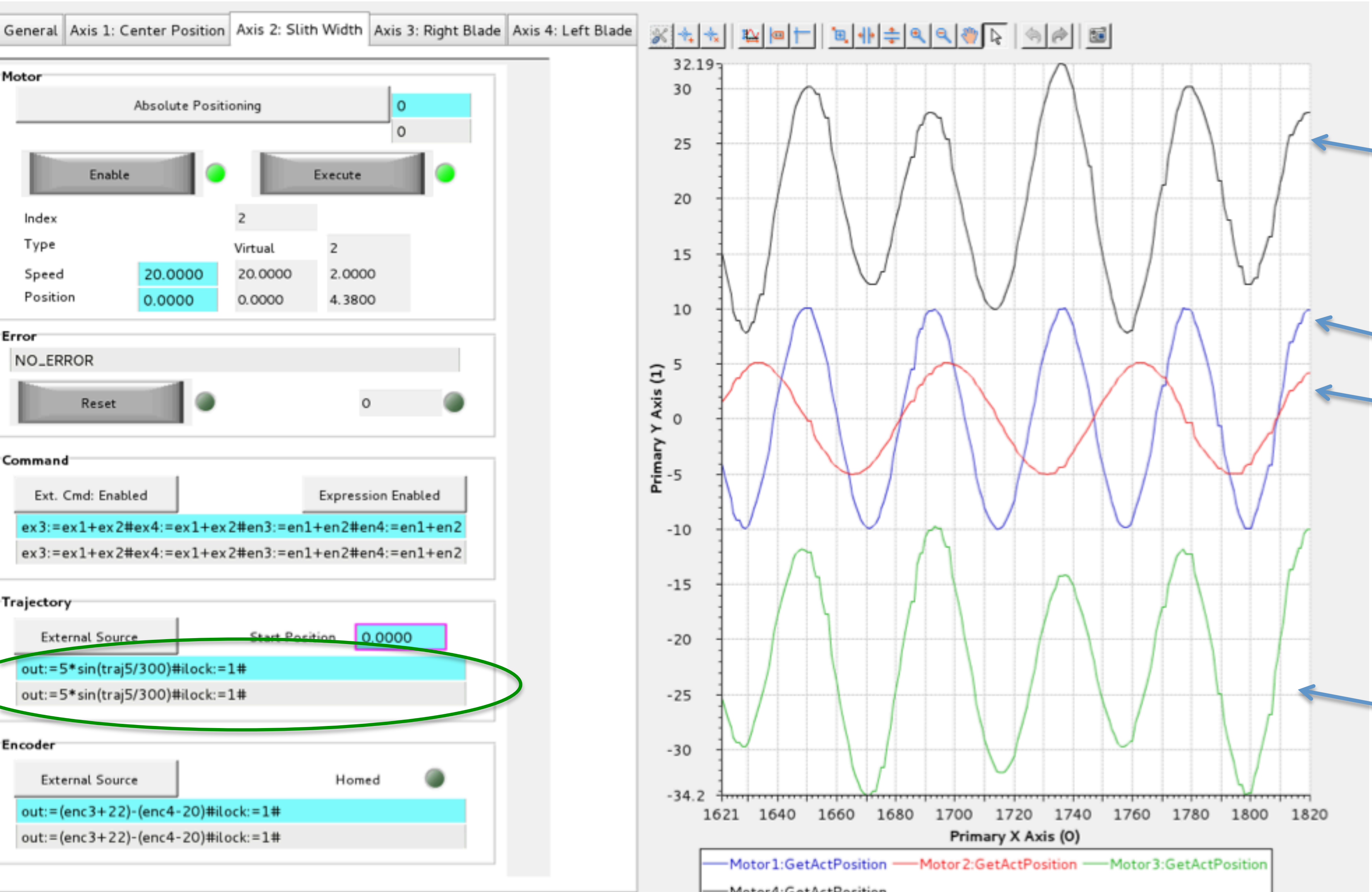
Inverse Kinematics:

enc1:=(enc3+enc4)/2;

enc2:=(enc4-enc3);

Amplifier enable:

inematics Example: 2 Axes slit set



General | Axis 1: Center Position | Axis 2: Slith Width | Axis 3: Right Blade | Axis 4: Left Blade

Motor

Absolute Positioning 0 0

Enable ● Execute ●

Index 2

Type Virtual 2

Speed 20.0000 20.0000 2.0000

Position 0.0000 0.0000 4.3800

Error

NO_ERROR

Reset ● 0 ●

Command

Ext. Cmd: Enabled Expression Enabled

```
ex3:=ex1+ex2#ex4:=ex1+ex2#en3:=en1+en2#en4:=en1+en2
ex3:=ex1+ex2#ex4:=ex1+ex2#en3:=en1+en2#en4:=en1+en2
```

Trajectory

External Source Start Position 0.0000

```
out:=5*sin(traj5/300)#ilock:=1#
out:=5*sin(traj5/300)#ilock:=1#
```

Encoder

External Source Homed ●

```
out:=(enc3+22)-(enc4-20)#ilock:=1#
out:=(enc3+22)-(enc4-20)#ilock:=1#
```

Axis 4

Axis 1 (Ce

Axis 2 (Ga

Axis 3

onfiguration / how to use

ne motion system is precompiled as an EEE-module (ESS Epics nvironment module) and only needs configuration.

onfiguration directly in EPCIS startupfile (“st.cmd”):
cmcConfigController ”asynport” “command”

xample 1: Create Axis object:
cmcConfigController ”asynport” ”Cfg.CreateDefaultAxis(1)”

xample 2: Set position controller gain for axis 1 to 0.1
cmcConfigController ”asynport” ”Cfg.SetAxisCntrlKp(1,0.1)”

Challenges

Epics and drivers:

None (maybe some strange behavior of motor record)

In-house expertise available

Etherlab EtherCAT master:

Functionality:

- DC-clock synchronization issues
- Solution: Patches exist

Hardware compatibility:

- Fail to communicate with a slave (EL7211)
- Solution: Compile time option

Good support forum/mailing list

Wide user community

ExprTK:

None

pen/Propriety source: Pros/Cons

Open Source

Pros	Cons
exibility	Need for development
o Licenses	Ownership
voids Vendor ock-In	Certifications/ Safety
ast Bug Fixes	No bug fixes (risk)
ree Support	Support availabilit

Propriety Source

Pros	Cons
Support	Licenses
Stability	Stability
Certifications/ Safety	Software Opa
Compliance to standards	Dependency/ Vendor lock in
Educatio	Slow bug fi

ummary

An open source motion control frame work for use with EPICS based on open source EtherCAT master (www.etherlab.org) and ExprTK expression parser have beer successfully developed.

Many opportunities with open source motion control however also some challenges.

Acknowledgments

- Igh open source EtherCAT master (www.etherlab.org)
- EPICS community (base, motor, asyn, stream device)
- ExprTK C++ Mathematical Expression Library (www.partow.net/programming/exprtk)

uestions?

ncoder object

Actual position value links to any input process value
EtherCAT process image.

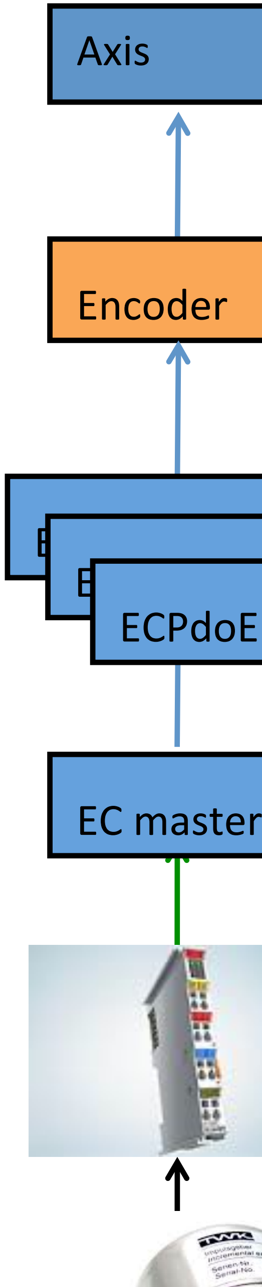
Scaling

1-64bit

Handles over/under flow (1-63bit)

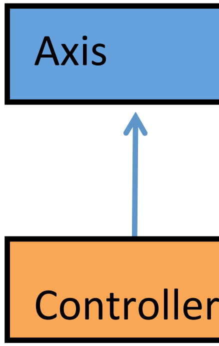
Calculates velocity

Filter



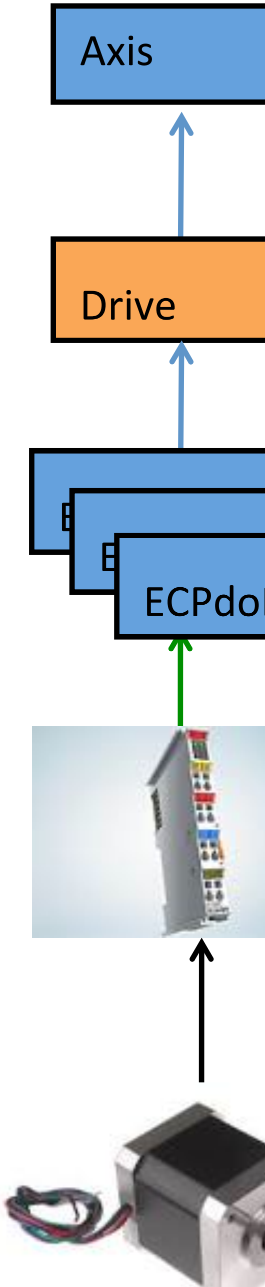
ontroller object

- PID controller
- Feed forward functionality
- Setpoint from trajectory object
- Actual value from encoder object

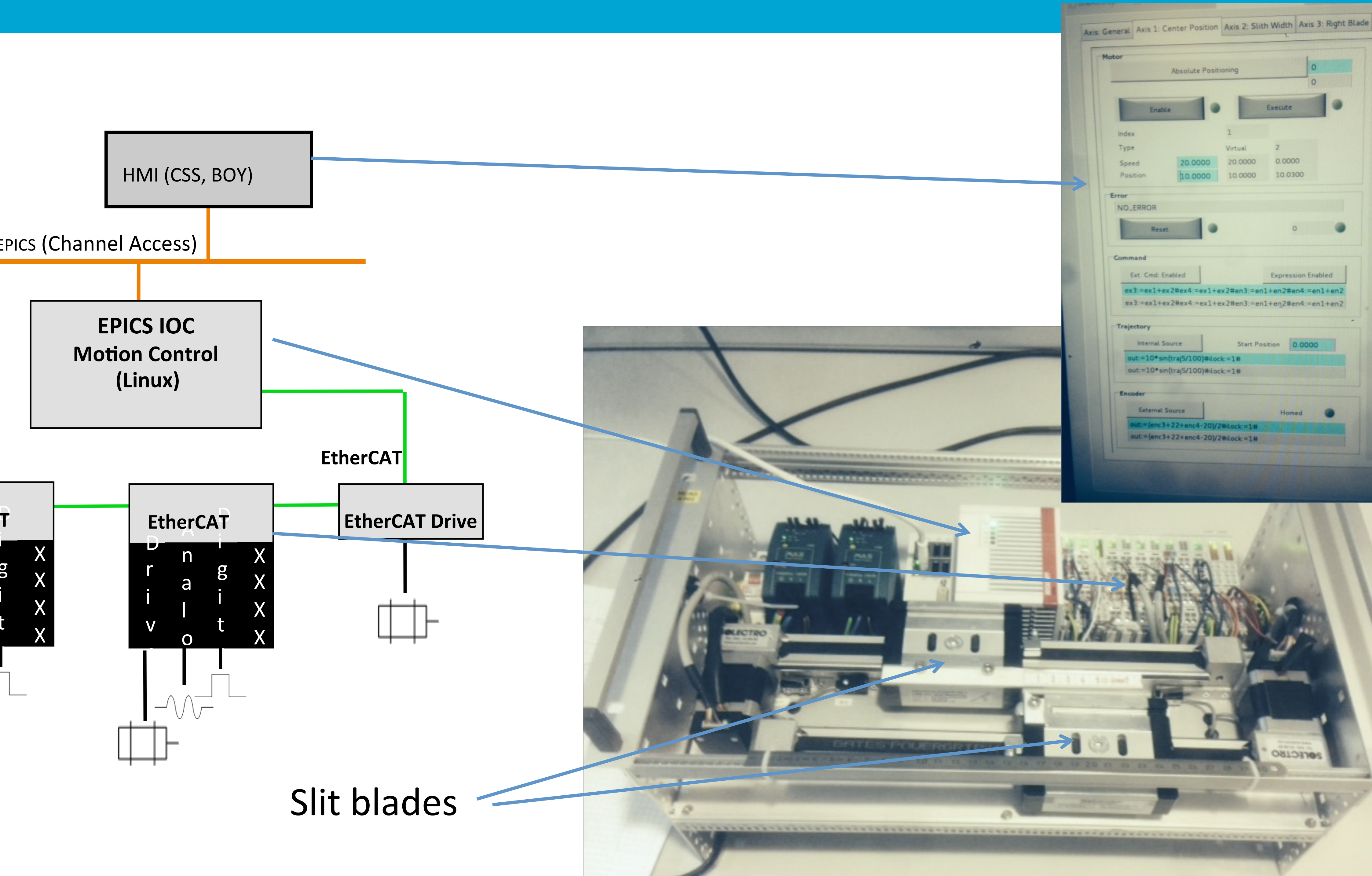


Drive object

- Link Velocity Setpoint to EtherCAT process image
- Link Enable to EtherCAT process image
- Scaling
- 1-64bit



xample Application



HMI (CSS, BOY)

EPICS (Channel Access)

EPICS IOC
Motion Control
(Linux)

EtherCAT

T

EtherCAT

EtherCAT Drive

D r i v

A n a l o

D i g i t

X X X X

g i t

X X X X

Slit blades

Axis: General | Axis 1: Center Position | Axis 2: Slith Width | Axis 3: Right Blade

Motor

Absolute Positioning 0 0

Enable ● Execute ●

Index		1	
Type		Virtual	2
Speed	20.0000	20.0000	0.0000
Position	10.0000	10.0000	10.0300

Error

NO_ERROR

Reset ● 0 ●

Command

Ext. Cmd: Enabled | Expression Enabled

```
ex3:=ex1+ex2#ex4:=ex1+ex2#en3:=en1+en2#en4:=en1+en2
ex3:=ex1+ex2#ex4:=ex1+ex2#en3:=en1+en2#en4:=en1+en2
```

Trajectory

Internal Source | Start Position 0.0000

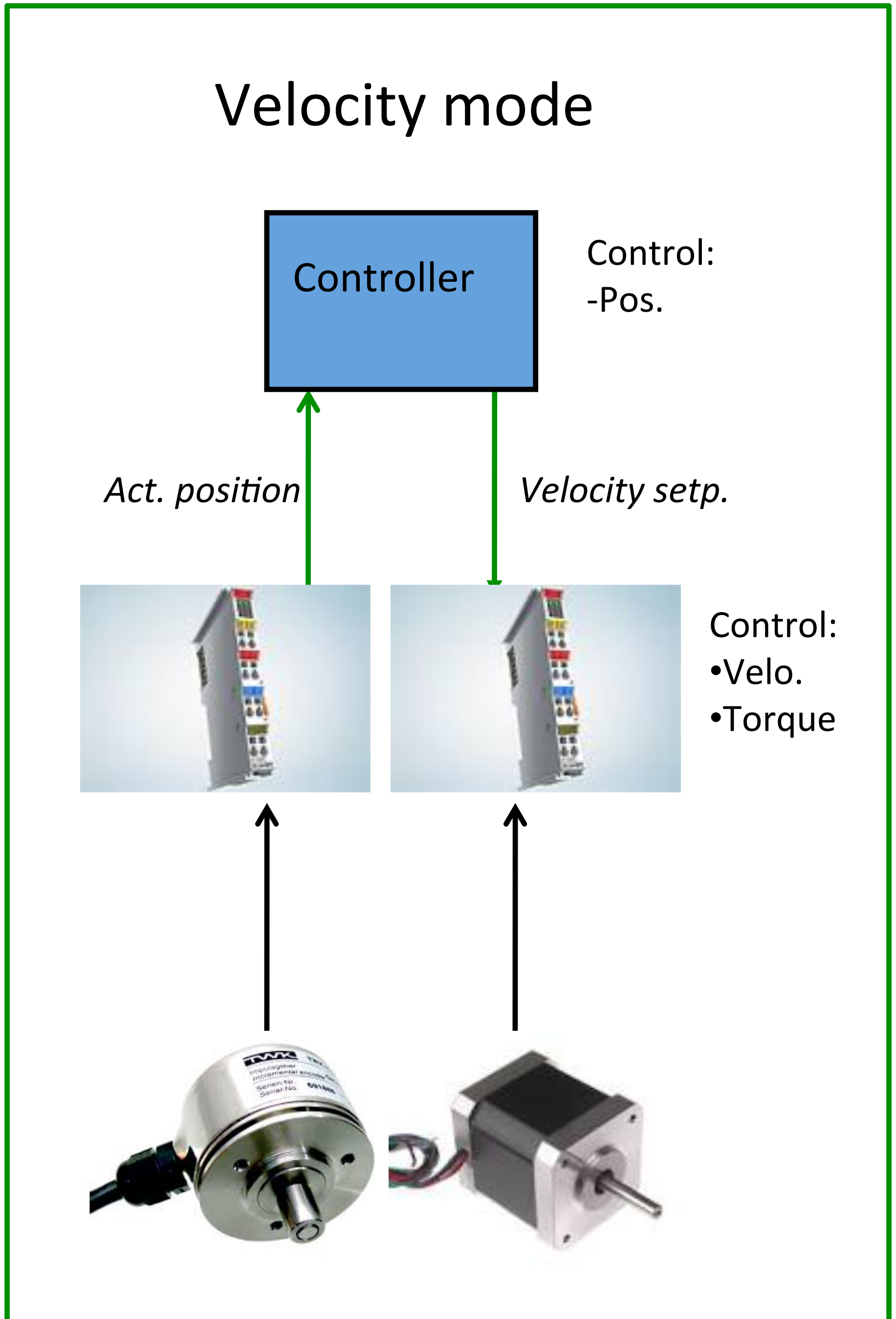
```
out:=10*sin(traj5/100)#ilock:=1#
out:=10*sin(traj5/100)#ilock:=1#
```

Encoder

External Source | Homed ●

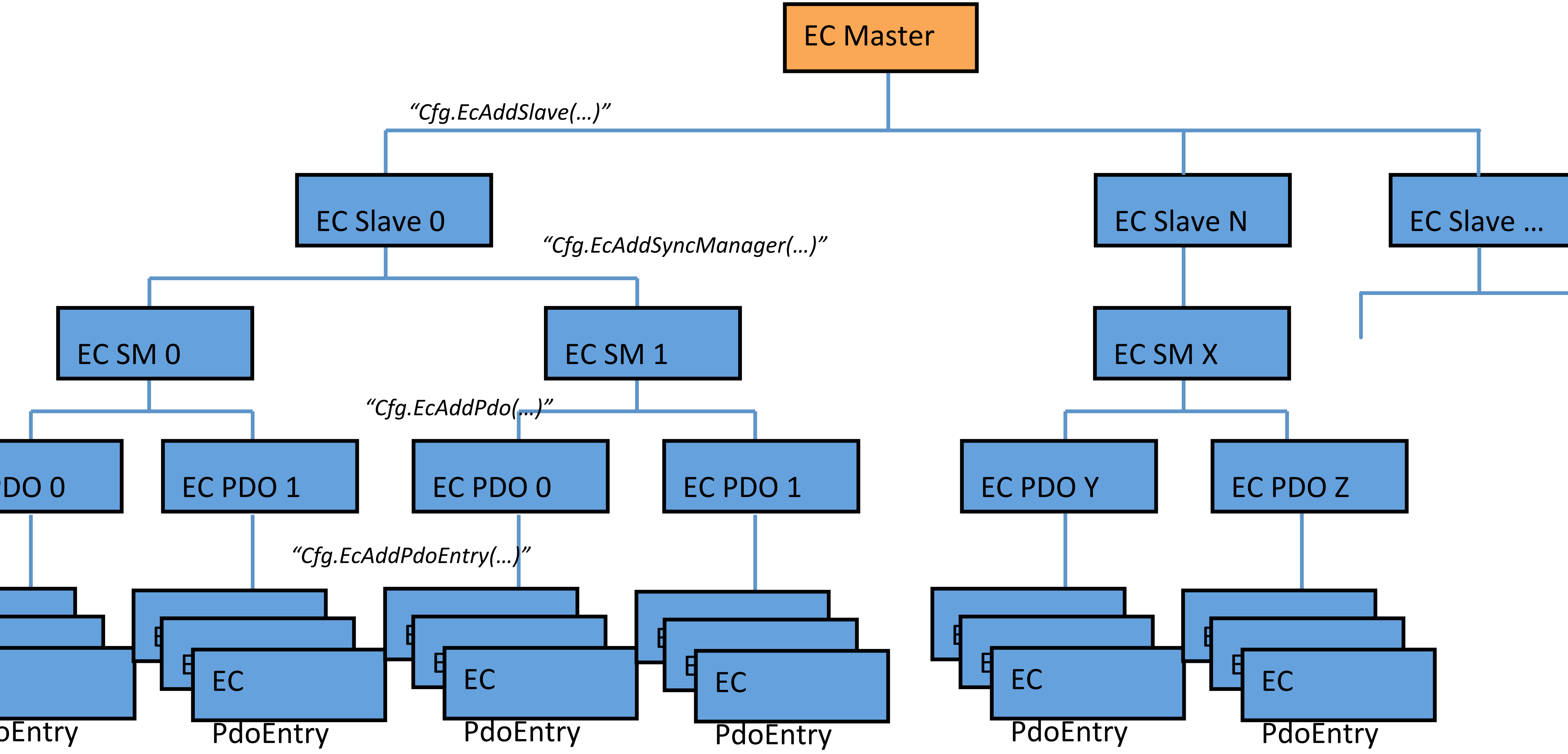
```
out:=(enc3+22+enc4-20)/2#ilock:=1#
out:=(enc3+22+enc4-20)/2#ilock:=1#
```

therCAT Motion: Control Modes



Velocity mode

therCAT Hardware Configuration Tree


EC Master
"Cfg.EcAddSlave(…)"
EC Slave 0
EC Slave N
EC Slave …
"Cfg.EcAddSyncManager(…)"
EC SM 0
EC SM 1
EC SM X
"Cfg.EcAddPdo(…)"
DO 0
EC PDO 1
EC PDO 0
EC PDO 1
EC PDO Y
EC PDO Z
"Cfg.EcAddPdoEntry(…)"
oEntry
EC PdoEntry
EC PdoEntry
EC PdoEntry
EC PdoEntry
EC PdoEntry