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

www.europeanspallationsource.se

Outline

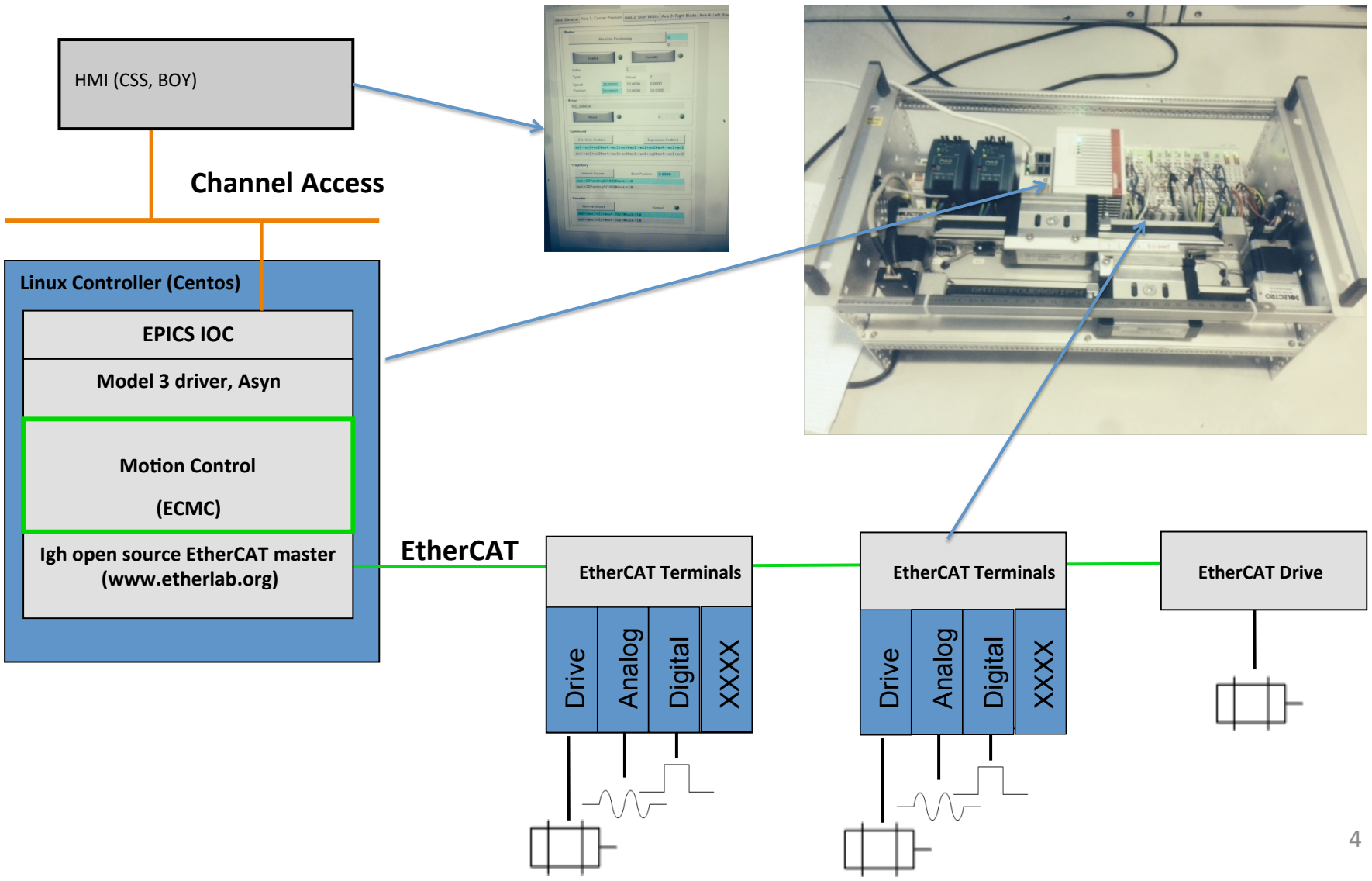
- EtherCAT fieldbus
- Architecture
- Axis Object
- Trajectory Object
- Monitoring Object
- Basic Kinematics
- Opportunities / Challenges
- Summary
- Acknowledgments

EtherCAT Fieldbus

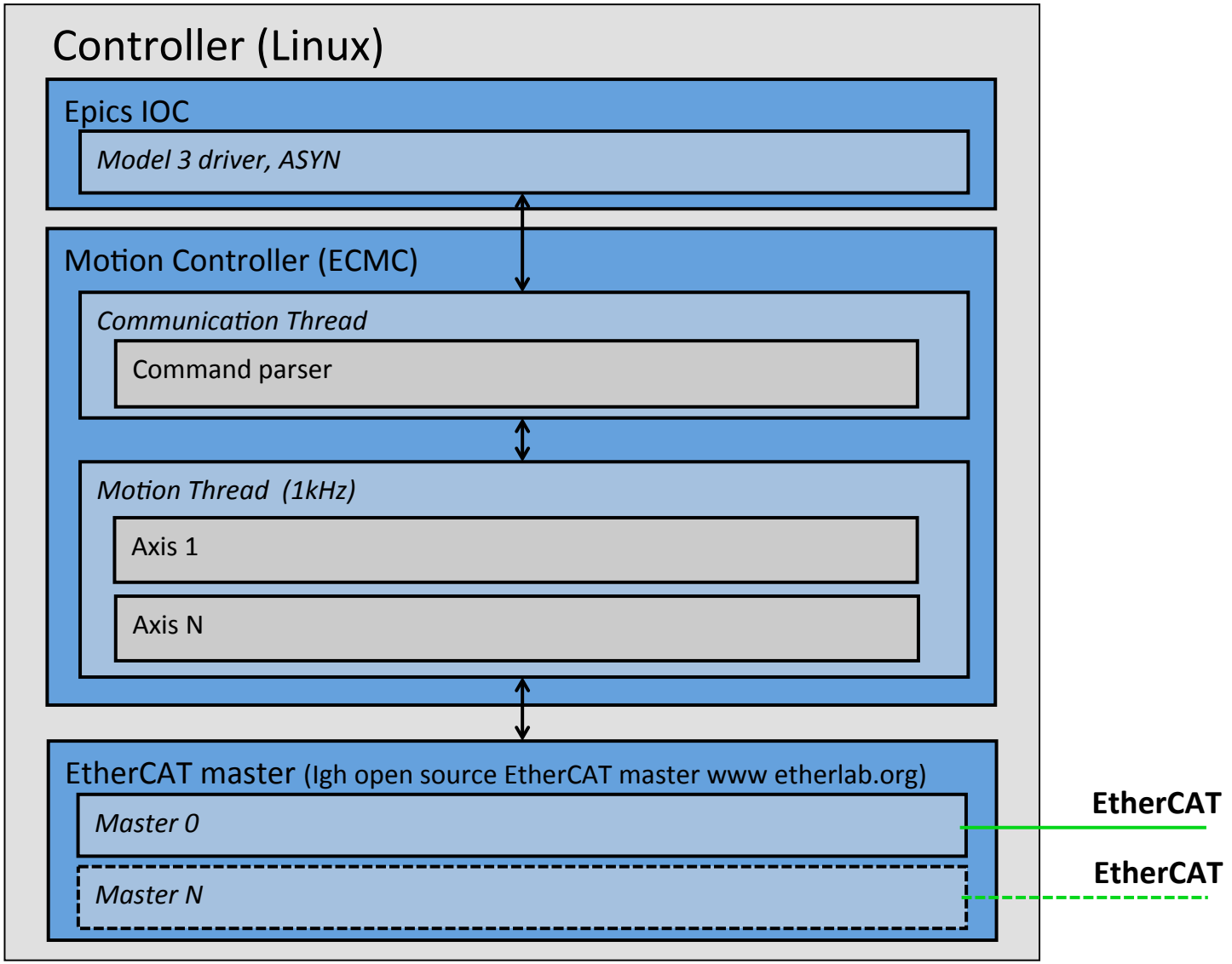
EUROPEAN SPALLATION SOURCE

- EtherCAT = **Ether**net for **C**ontrol **A**utomation **T**echnology
- Open fieldbus standard originally developed by Beckhoff GmbH
- Maintained by EtherCAT Technology Group (www.EtherCAT.org).
- Hardware requirements:
  - **Master: standard computer hardware (NIC)**
  - **Slaves: dedicated hardware, EtherCAT Slave Controller (ESC)**
- Masters: Several commercial and **open source** masters available
- Slaves: Several 100 manufacturers of slaves (drives, I/O, sensors, robots)
- Topologies: Line, Star, Ring
- Media: Cat 5 cable, plastic fiber, glass fiber
- **Supports Distributed Clock (DC) in slaves**
- Bandwidth utilization: 80%-97% (100 Mbit/s , Ethernet, Full-Duplex)
- Applications: **Motion, large or long distance systems, synchronized systems**
- Cycle times > 50µs

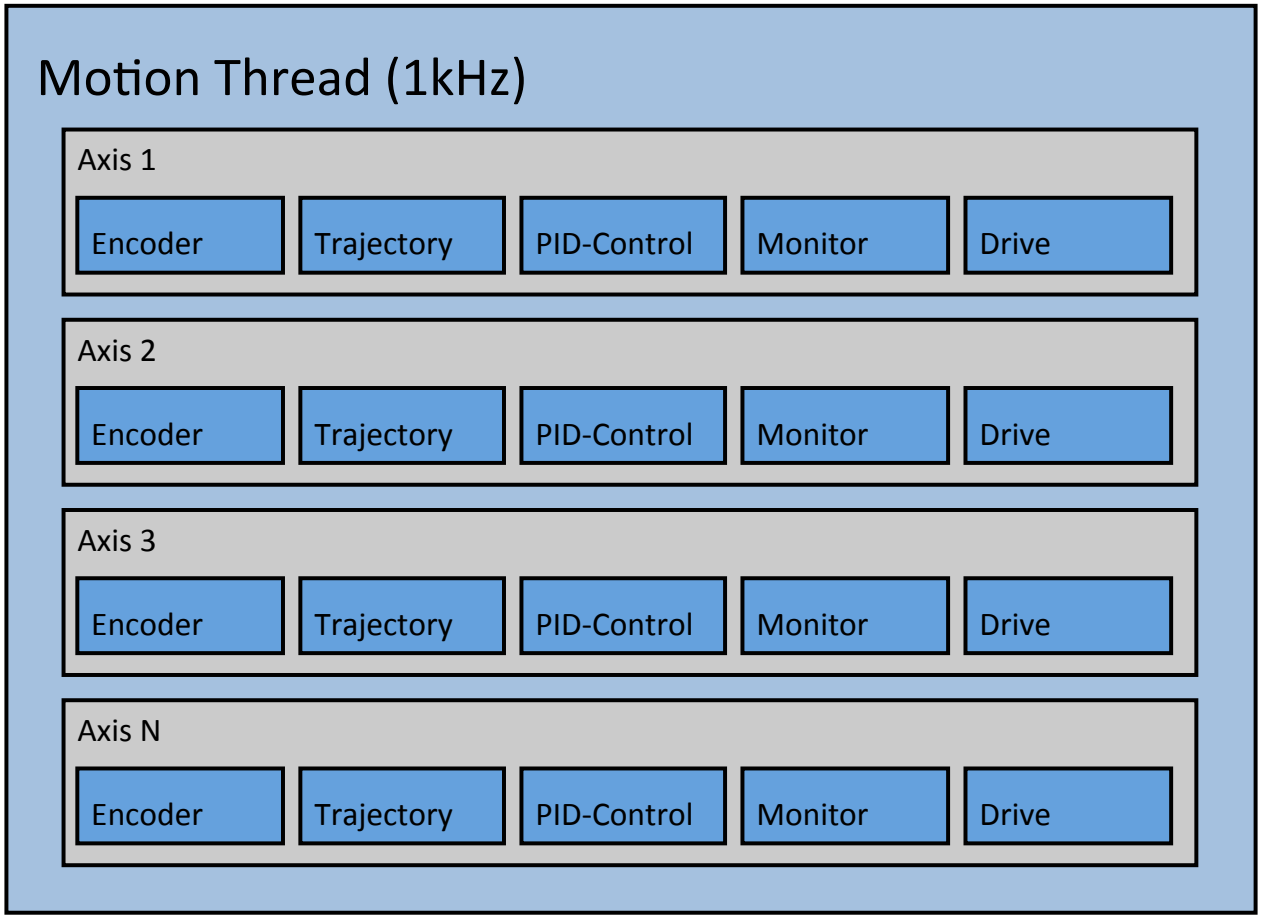
Architecture



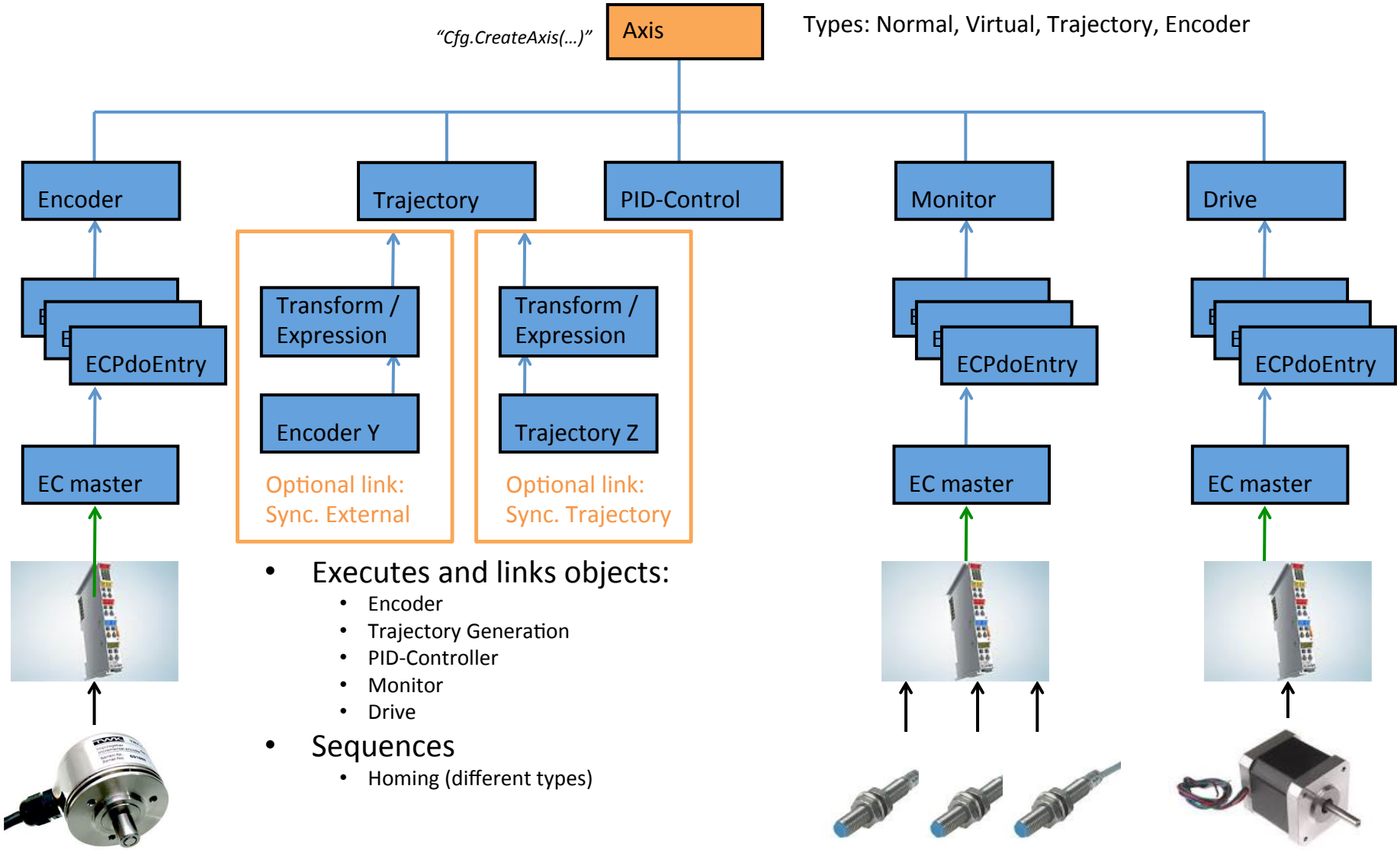
Motion Controller



Motion Thread



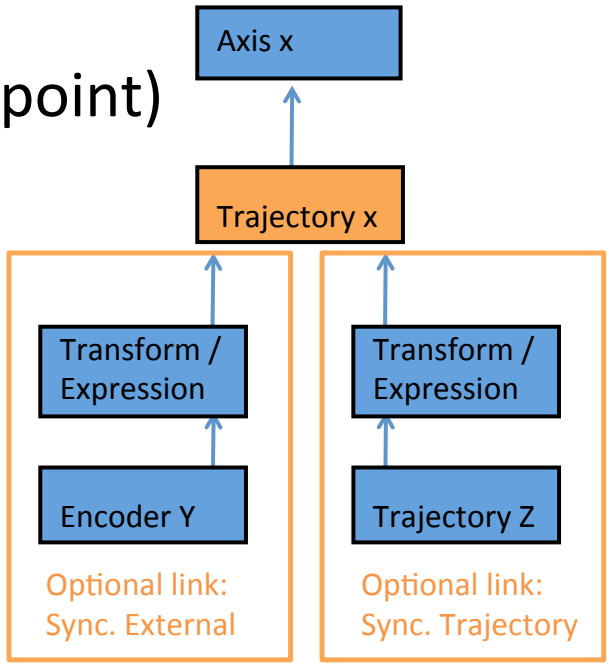
Structure: Axis object



- Executes and links objects:
  - Encoder
  - Trajectory Generation
  - PID-Controller
  - Monitor
  - Drive
- Sequences
  - Homing (different types)

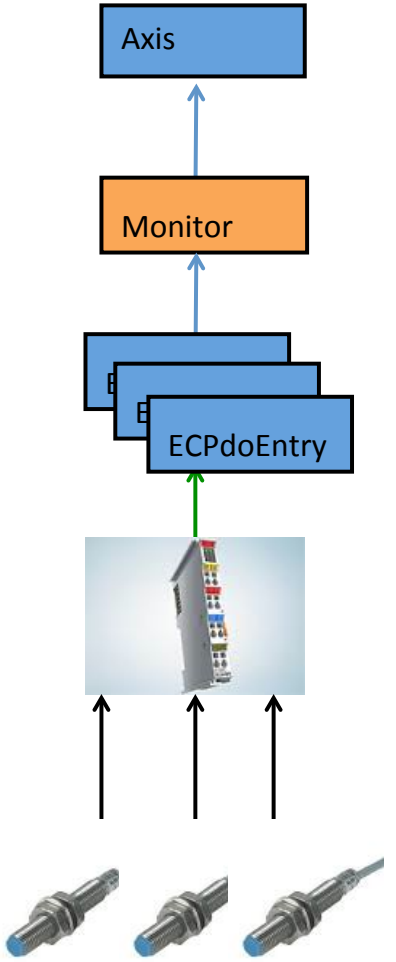
Trajectory object

- 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



Kinematics

- 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)

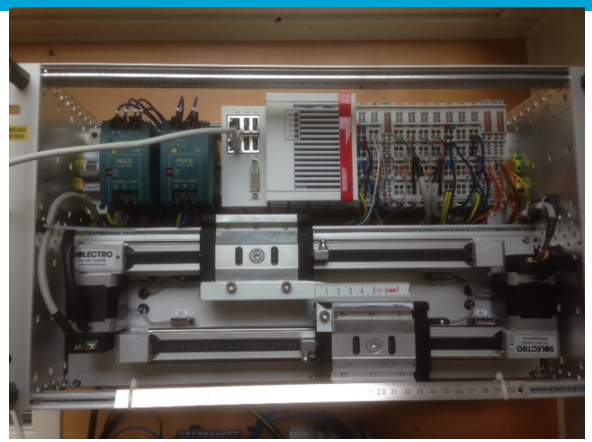
traj*x* = Trajectory generated setpoint for axis *x*
enc*x* = Actual position of axis *x*
en*x* = *Enable of axis x*
il*x* = *Interlock of axis x*

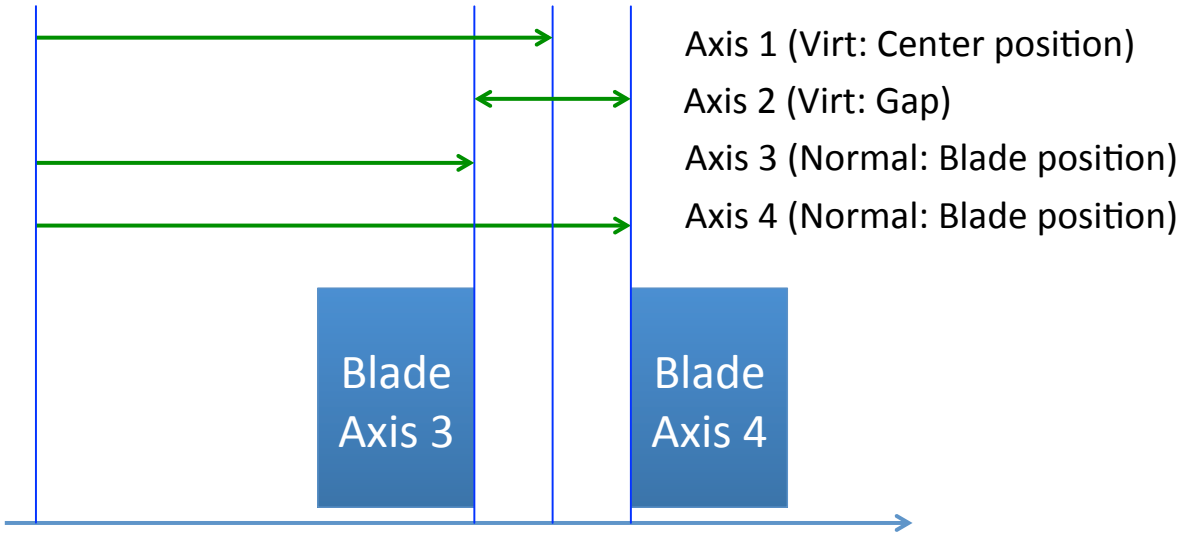
Example:
traj1:=10*sin(traj2+enc3);
il1:=il2 and il5 and enc4>enc3;

*http://www.partow.net/programming/exprtk/)

Kinematics Example: 2 axes slit set

- 2 virtual axes
  - Slit center position
  - Slit gap
- 2 normal axes (blade positions)



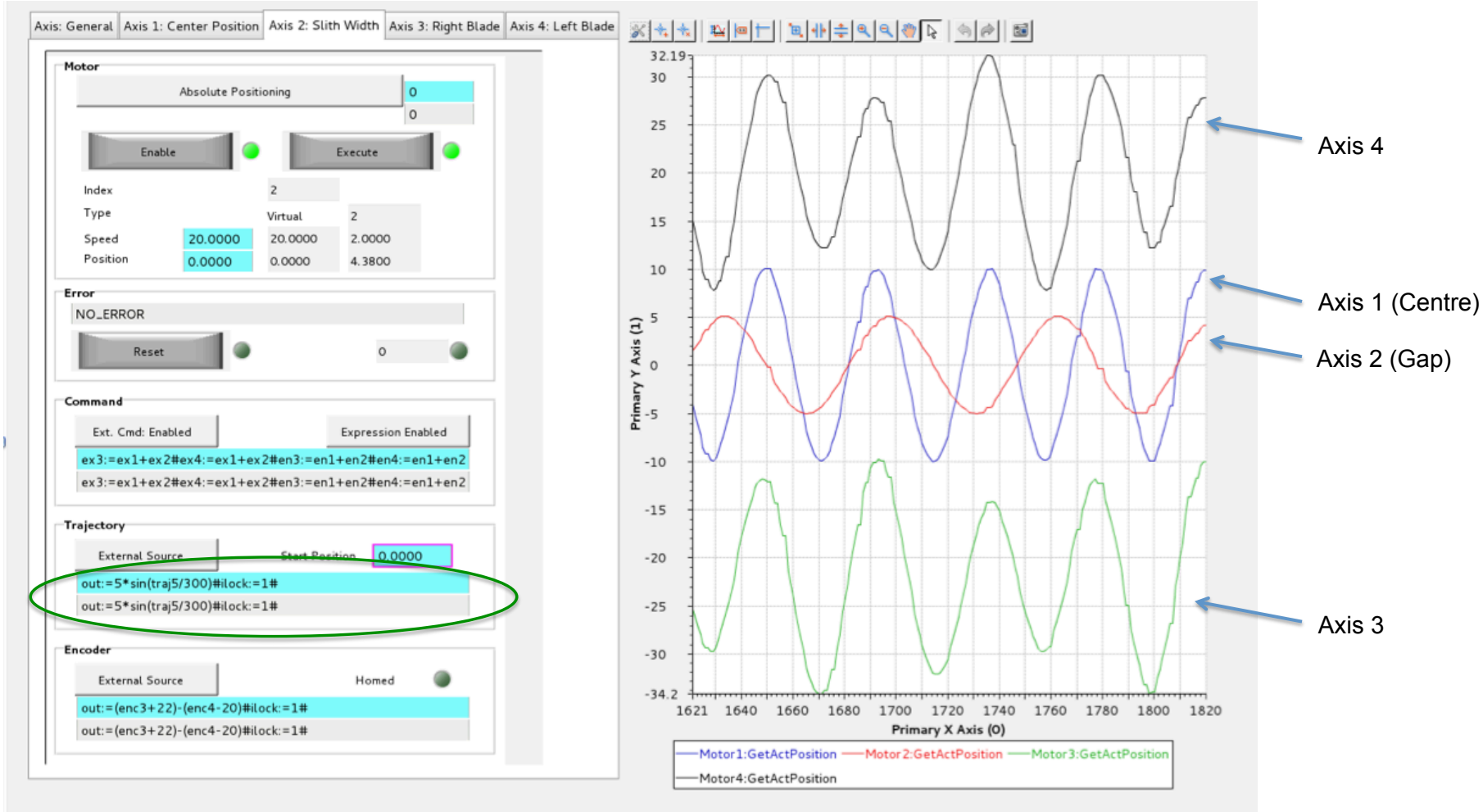


Forward Kinematics:
traj3:=traj1-traj2/2;
traj4:=traj1+traj2/2;

Inverse Kinematics:
enc1:=(enc3+enc4)/2;
enc2:=(enc4-enc3);

Amplifier enable:
En3:=En1 or En2;
En4:=En1 or En2;

Kinematics Example: 2 Axes slit set



Configuration / how to use

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

Configuration directly in EPCIS startupfile (“st.cmd”):
ecmcConfigController ”asynport” “command”

Example 1: Create Axis object:
ecmcConfigController ”asynport” ”Cfg.CreateDefaultAxis(1)”

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

Challenges

1. Epics and drivers:

- None (maybe some strange behavior of motor record)
- In-house expertise available

2. 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

3. ExprTK:

- No issues.

Open/Propriety source: Pros/Cons

EUROPEAN SPALLATION SOURCE

Open Source

Pros	Cons
Flexibility	Need for development
No Licenses	Ownership
Avoids Vendor Lock-In	Certifications/ Safety
Fast Bug Fixes	No bug fixes (risk)
Free Support	Support availability

Propriety Source

Pros	Cons
Support	Licenses
Stability	Stability
Certifications/ Safety	Software Opacity
Compliance to standards	Dependency/ Vendor lock in
Education	Slow bug fixes

Summary

- 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 been successfully developed.

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

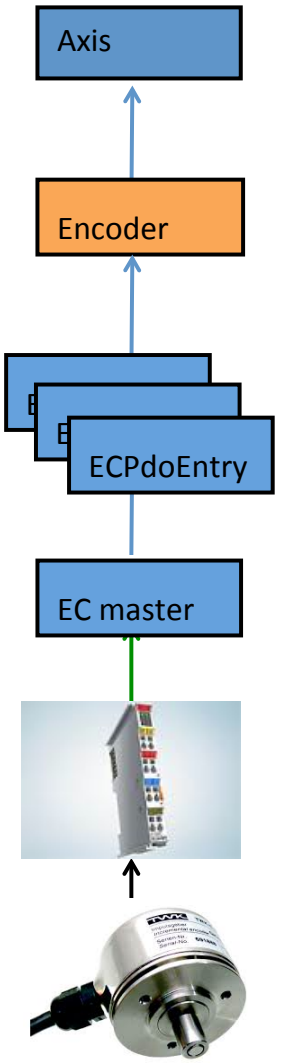
Acknowledgments

- 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)

Questions?

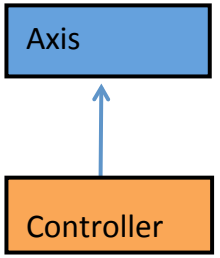
Encoder object

- Actual position value links to any input process value in EtherCAT process image.
- Scaling
- 1-64bit
- Handles over/under flow (1-63bit)
- Calculates velocity
- Filter



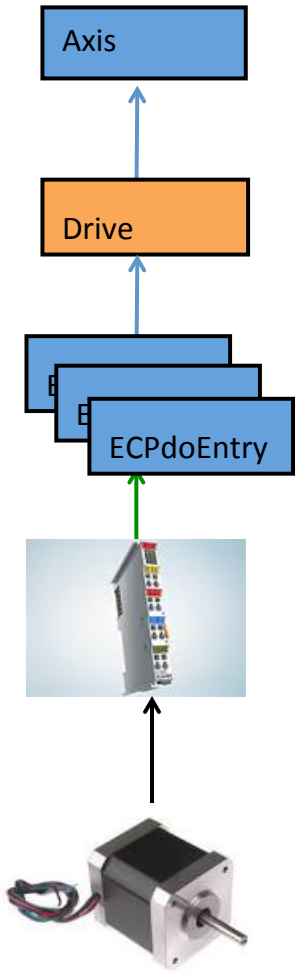
Controller object

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

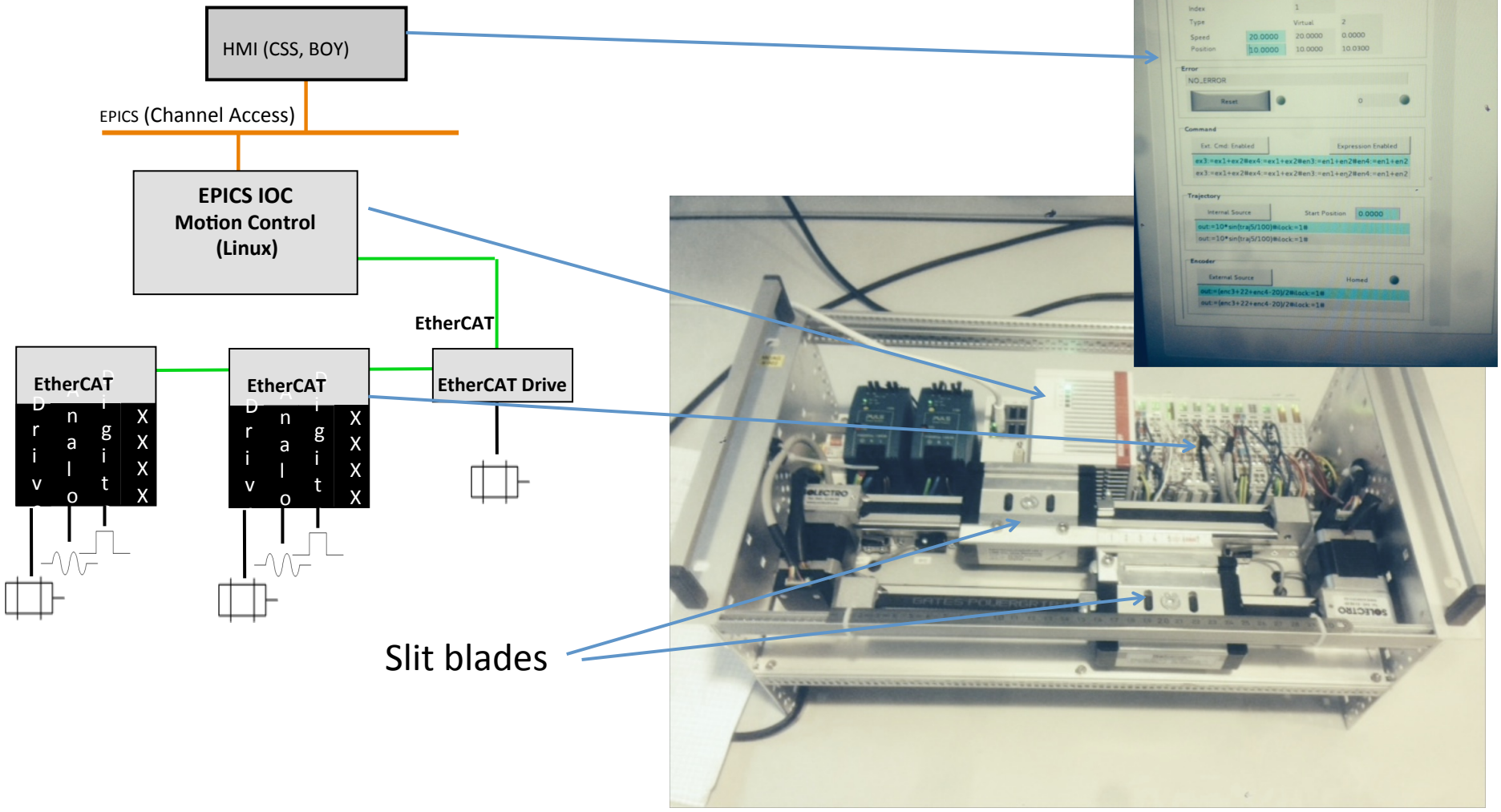


Drive object

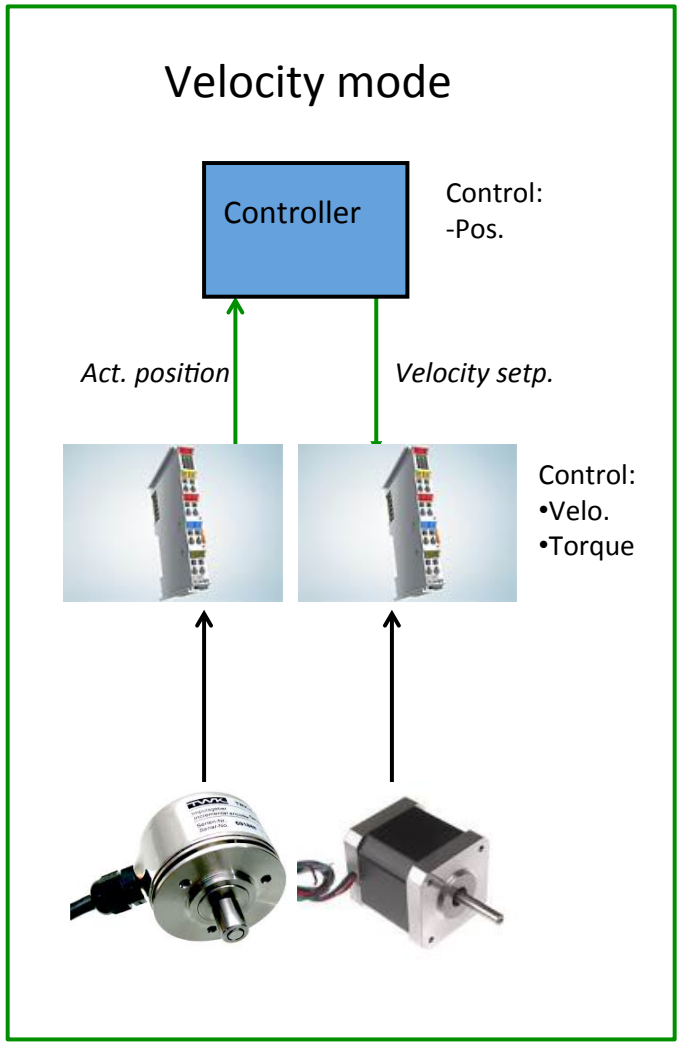
- Link Velocity Setpoint to EtherCAT process image
- Link Enable to EtherCAT process image
- Scaling
- 1-64bit



Example Application



EtherCAT Motion: Control Modes



EtherCAT Hardware Configuration Tree