
UKRI
Science and Technology Facilities Council
ISIS Neutron and Muon Source
Welcome

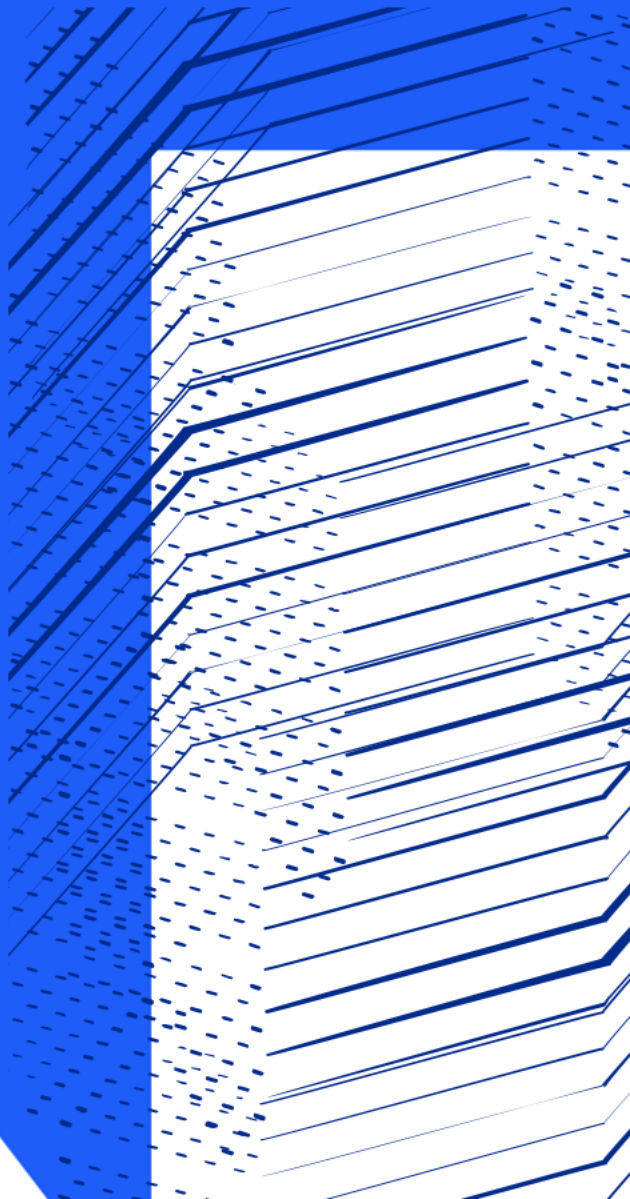
Science and Technology Facilities Council

ISIS Neutron and Muon Source

Python based EPICS interface with CIP enabled devices

EPICS Collaboration Meeting September 2022

Basil Aljamal



ISIS control system modernization

- Intel Itanium ➔ Intel Xeon
- OpenVMS ➔ Linux
- Bare metal + Xen VMs ➔ Docker containers
- Vsystem ➔ EPICS
- ABB, Schneider, Omron CX/CJ➔ Omron NX/NJ
- FINS ➔ CIP

CIP vs FINS

CIP (Common Industrial Protocol)	FINS (Factory Interface Network Service)
access memory via tags	access memory via physical memory addresses
complex data structures support	Basic data types
Hard to implement	easy to implement

PLC side implementation

- Array of structures
- Each structure holds all information related to certain process variables including pvname
- PLC is the source of truth

▼ PVData	STRUCT	NJ
PVName	STRING[60]	
PVDesc	STRING[40]	
PVValue	REAL	
PVHigh	REAL	
PVHiHi	REAL	
PVLow	REAL	
PVLoLo	REAL	
PVAlarmState	UINT	
Hyst	BOOL	
PVAlarmDesc	STRING[40]	
PVEGU	STRING[10]	

EPICS side implementation

- Python based PVAserver (using PvaPy library) and cpppo library for CIP communication
- Read full array of structures at startup
- Read just the value and alarm state every second
- Reread full structure every 30 seconds

Lessons learned

- CIP has a maximum packet length of 500 bytes for forward open connection and 2000 bytes for large forward open connection
- CIP protocol has a feature called “read tag fragmented service” but is not supported by all PLC vendors
- Increasing the communication percentage of the PLC processing cycle won’t help
- Having a structure of arrays is better than an array of structures

Science and Technology Facilities Council

ISIS Neutron and Muon Source

Thank you

isis.stfc.ac.uk @ISISNeutronMuon STFC