

PLC integration in EPICS environment: comparison between OPC server and direct drivers solutions

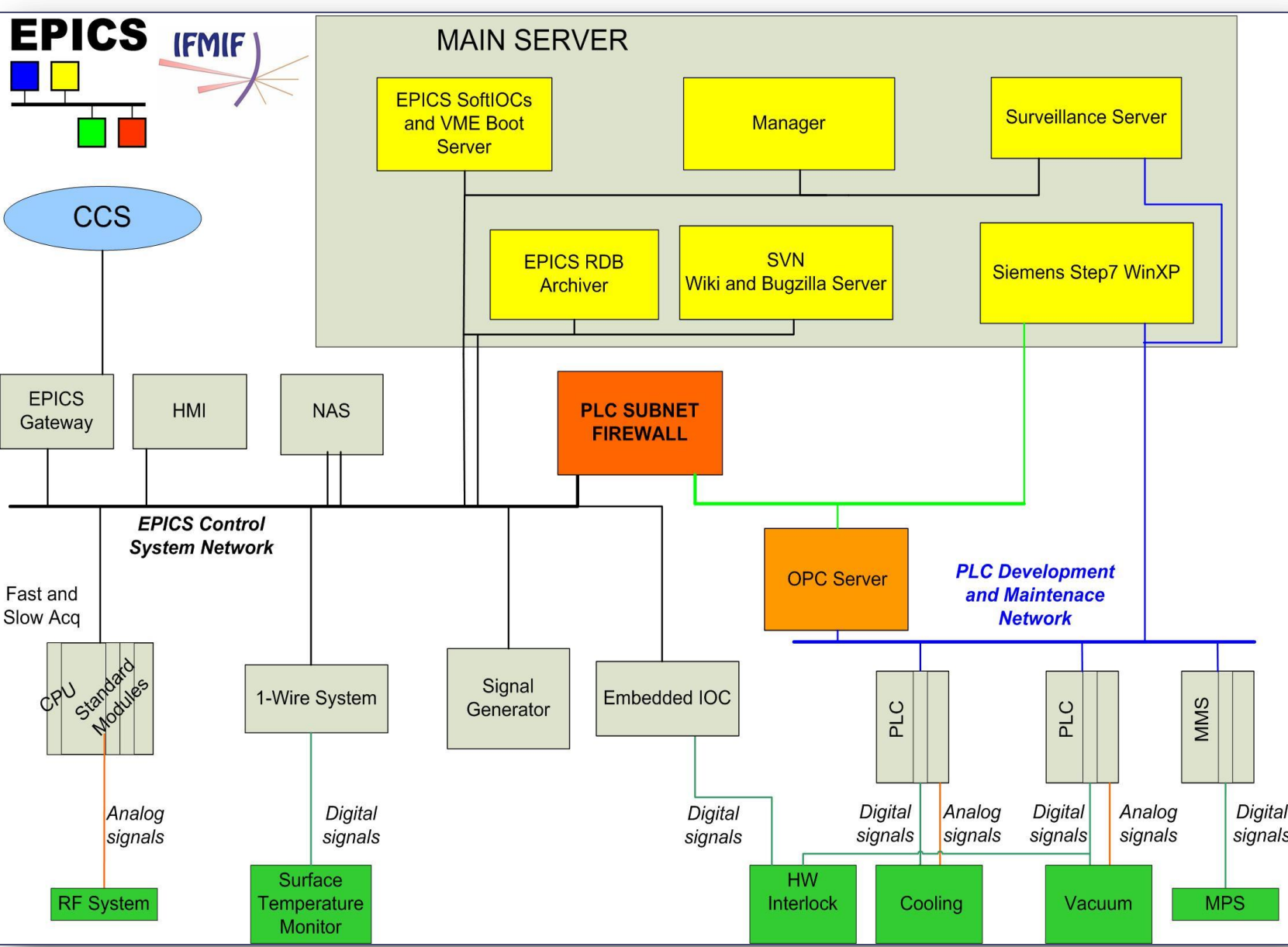
THPHA155



L. Antoniazzi, A. Baldo, M. Giacchini, M. Montis INFN-LNL, Legnaro, Italy

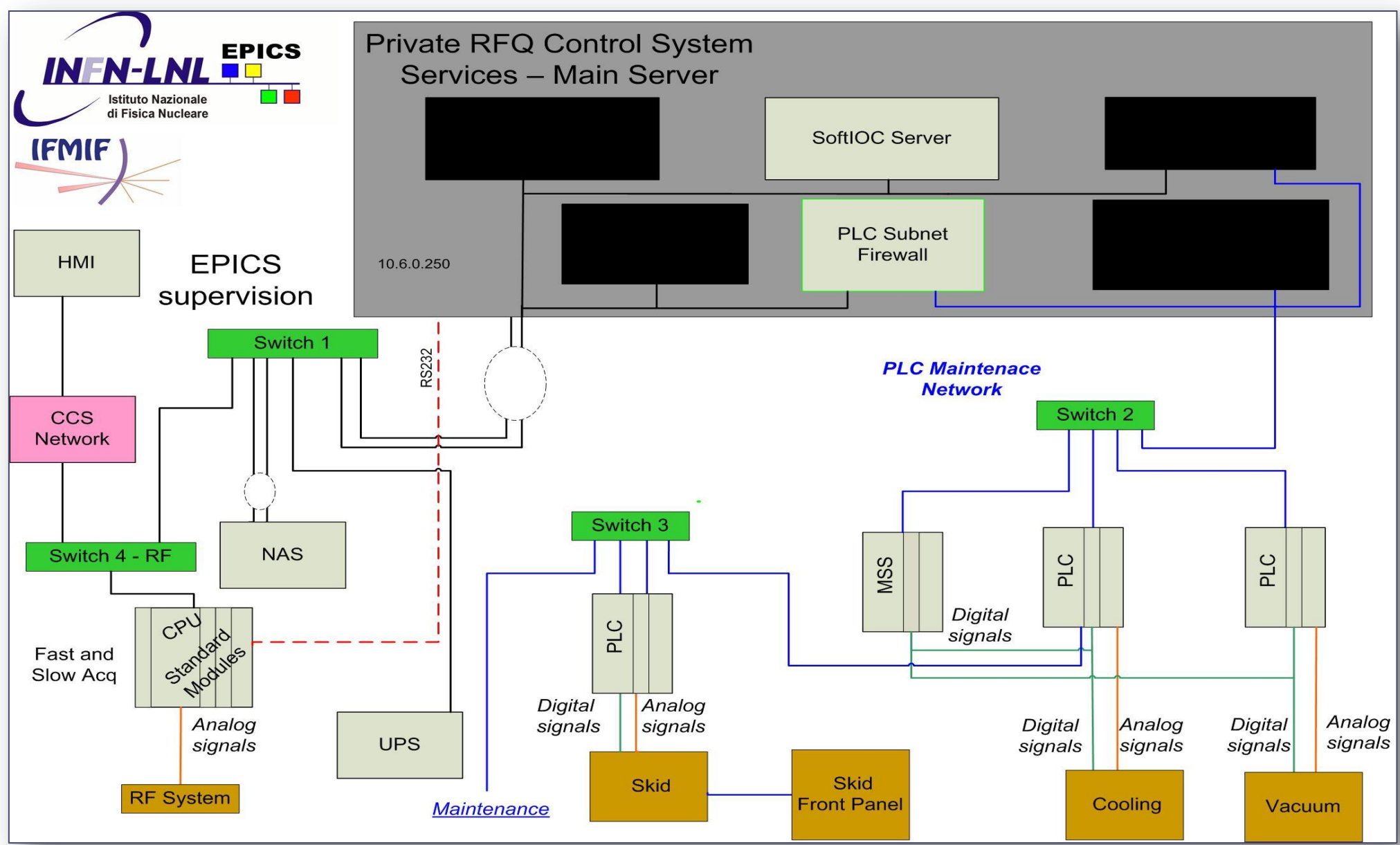
ABSTRACT:

In the IFMIF EVEDA project [1], INFN-LNL Laboratory has been involved in the design and construction of a normal conducting Radio Frequency Quadrupole (RFQ) used to bunch and accelerate a 125 mA steady beam to 5 MeV. The EPICS based control system has been entirely developed in house using different hardware solutions: PLC for tasks where security is the most critical feature, VME system where the acquisition speed rate is crucial, common hardware when only integration is required without any particular feature in terms of security. Integration of PLCs into EPICS environment was originally accomplished through OPC DA server hosted in a Windows embedded industrial PC. Due to the issues analyzed in injector LCS, LNL proposed to migrate to the usage of EPICS Direct Driver solution based on s7plc. The driver itself is suitable for direct communication between EPICS and PLCs, but it doesn't take care of data update and synchronization in case of communication failure. As consequence LNL team designed a dedicated method based on state machine language to manage and verify data integrity between the two environments, also in case of connection lost or fail.



OPC Configuration

Windows embedded industrial PC (Siemens IPC427) hosted the Simatic Net OPC server, IOC application and device support for the OPC server itself to monitor and control about 1100 PVs organized as one unique group with a refresh rate of 1s. The device support module used is the one developed at Helmutz Zentrum in Berlin (HZB).



s7plc Configuration

- Software IOC (softIOC) running on a virtual machine
- PLC subnetwork separated by EPICS network by virtualized firewall
- PLC and EPICS IOC communication via Send/Receive over TCP/IP

LCS OPC Architecture

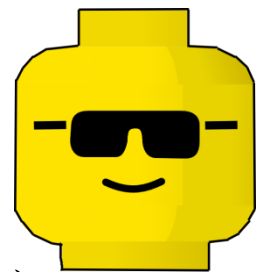
PLC DEVELOPER

PRO:

- Complete PLC memory addressing
- Configurable variable access rights (read, write and read/write)
- Bidirectional read/writes access and multiclient architecture
- Separation between PLC sub-network and the rest of control network
- Data access if required (upon change)

CONS:

- Identification of the exchanged variables in PLC memory
- Overwriting of the PLC data in case of IOC reboot
- Detected frozen PVs at IOC level



migration due to reliability issue

How can I have bidirectional write access with s7plc?



LCS s7plc Architecture s7plc OUT OF THE BOX

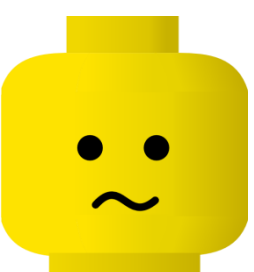
PLC DEVELOPER

PRO:

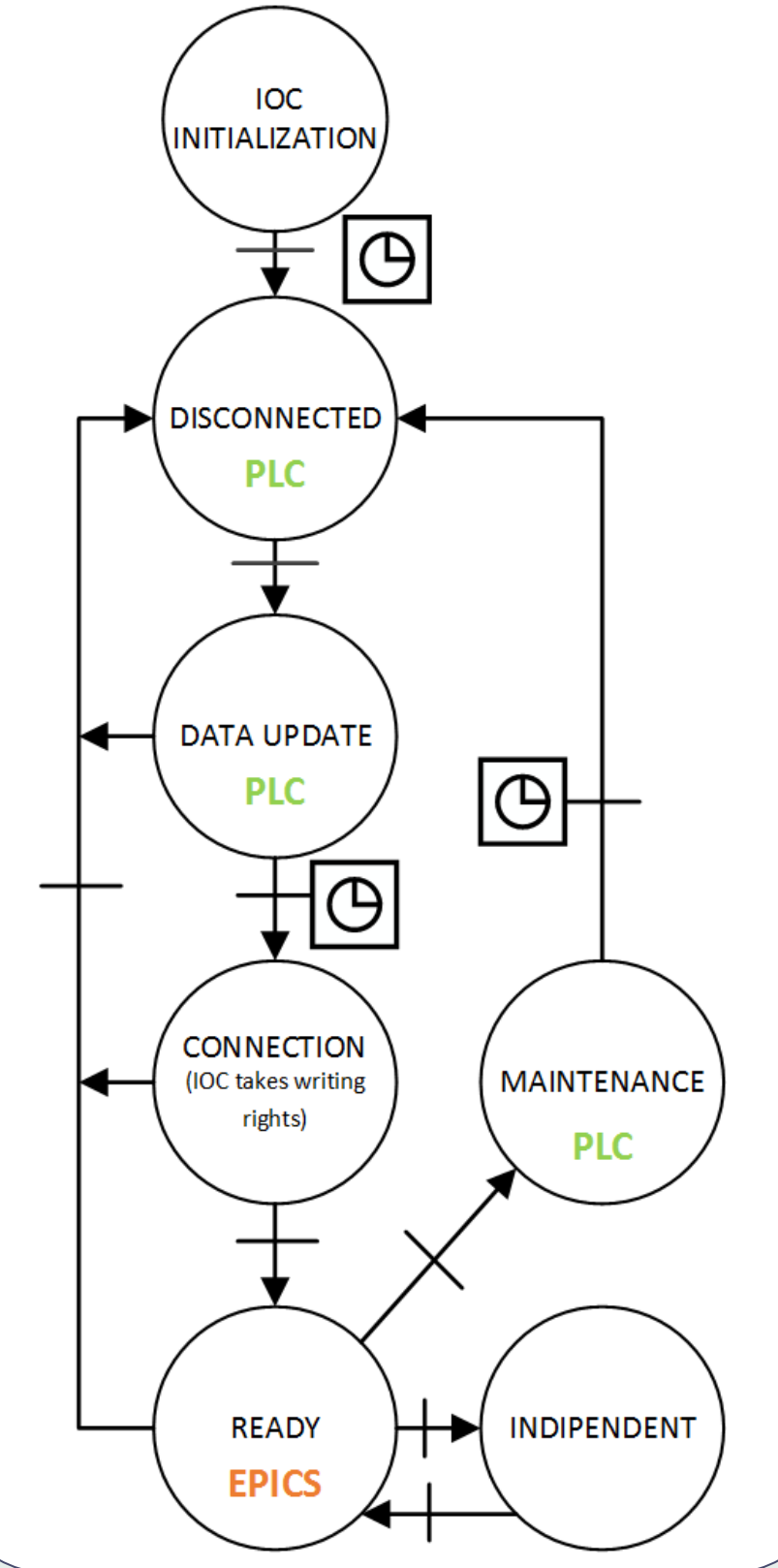
- Identification of the exchanged variables in PLC memory

CONS:

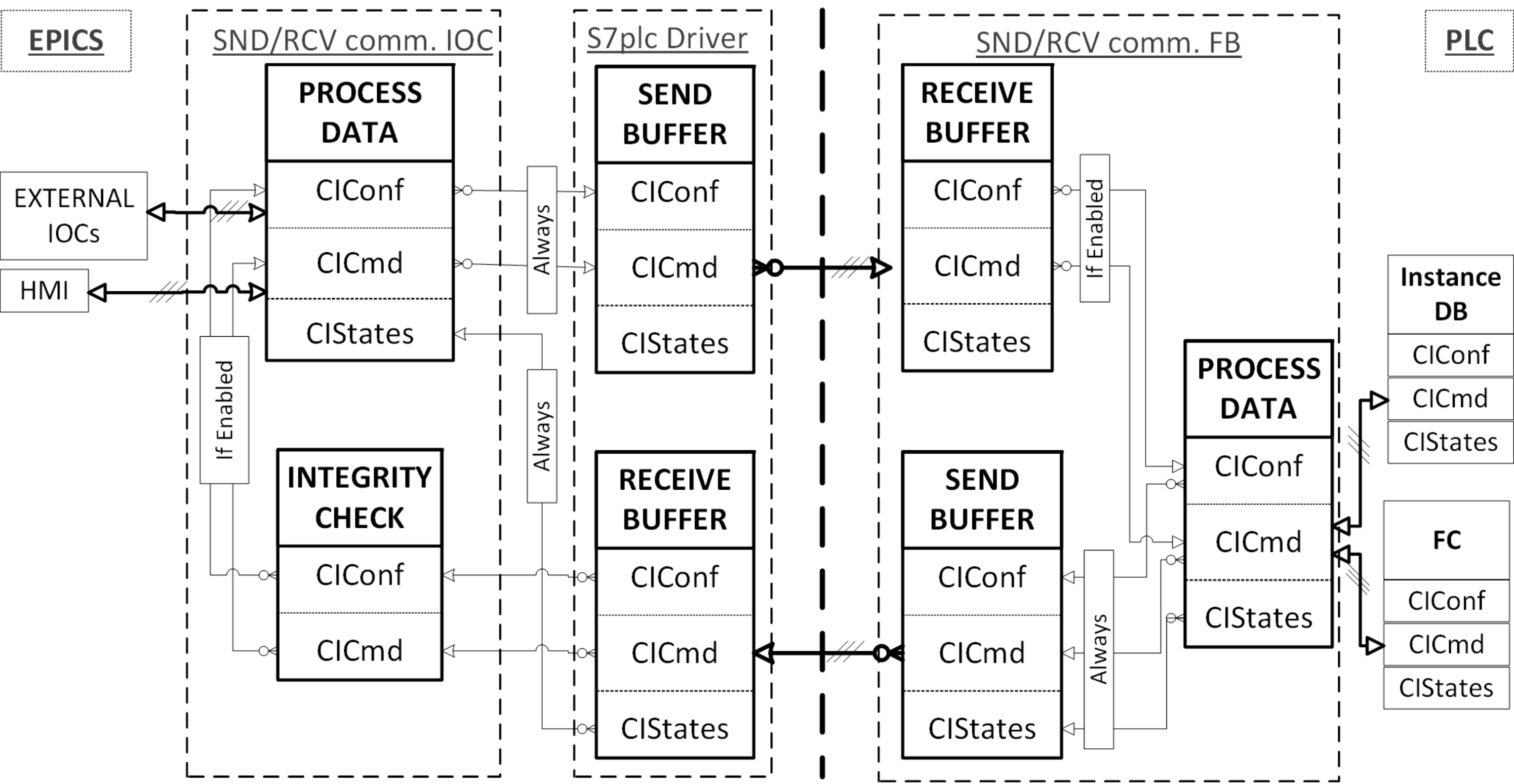
- Bidirectional read/writes access not supported
- Memory consumption
- Increased execution time
- Exchange of big data buffer can takes several PLC cycle



IOC Integrity Check State Machine



Data exchange between EPICS and PLC



PLC Implementation

Requirements:

- PLC process control data must be correct at each moment including IOC reboot, no connection, etc.
- Naming and organization of exchanged variables (UDT CIConf, CICmd, CIStates)
- Link the variable to the send receive buffer in accordance to the connection status
- Support bidirectional write access at list for CIConf and CICmd

Results:

- SND/RCV buffers common structure with header containing information to manage communication
- SND/RCV communication Function Block to manage data between process and buffers in each connection and communication status



PLC-EPICS shared variables

PLC	Sockets	Variables	Buffer Size [Bytes]
Vacuum	Fast	1361	2260
	Slow	770	2122
	Total	2131	4382
Cooling	Fast	657	2032
	Slow	380	998
	Skid Fast	363	1388
	Skid Slow	623	1592
	Total	2023	6010

Conclusion

Migration from OPC DA to s7plc required several efforts on both side (EPICS and PLC), especially to keep bidirectional variable access, but it pays back us with:

- Consistent data transfer and initialization procedure
- Well organized communication method on PLC side
- Possibility of define several connections respect the exchanged data functionality or scan rate
- Easy export of process data DB to EPICS data base

LCS s7plc Architecture s7plc AS IMPLEMENTED

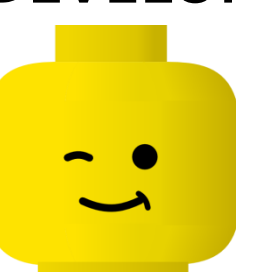
PLC DEVELOPER

PRO:

- Identification of the exchanged variables in PLC memory
- Easy export of send/receive DB to EPICS data base
- Consistent data transfer and initialization procedure
- Bidirectional writes access for a supported by dedicated procedure for a defined subset of variables
- Several connections can be arranged according the exchanged data functionality or scan rate

CONS:

- Bidirectional read/writes access not supported for all the variables
- Memory consumption
- Increased execution time
- Exchange of big data buffer can takes several PLC cycle



INFN-LNL: <http://www.lnl.infn.it>

EPICS: <http://www.aps.anl.gov/epics>

EPICS@LNL: <https://web2.infn.it/epics/>



RFQ LCS Team



IFMIF-EVEDA RFQ at Rokkasho

