EFDA–JET–CP(11)03/01

C. Hennig, K. Kneupner, D. Kinna, the Wendelstein 7-X CoDaC Team,
and JET EFDA contributors

Connecting Programmable Logic Controllers (PLC) to CODAS CoDaC A Comparison of the JET and Wendelstein 7-X Approach

Connecting Programmable Logic Controllers (PLC) to CODAS CoDaC A Comparison of the JET and Wendelstein 7-X Approach

C. Hennig[1], K. Kneupner[2], D. Kinna[3], the Wendelstein 7-X CoDaC Team[1],
and JET EFDA contributors*

JET-EFDA, Culham Science Centre, OX14 3DB, Abingdon, UK

[1]*Max-Planck-Institut für Plasmaphysik, Wendelsteinstrasse 1, 17491 Greifswald, Germany*
[2]*Klaus Kneupner, klaus@kneupner.de*
[3]*JET-EFDA, Culham Science Centre, OX14 3DB, Abingdon, OXON, UK*
** See annex of F. Romanelli et al, "Overview of JET Results",*
(23rd IAEA Fusion Energy Conference, Daejon, Republic of Korea (2010)).

Preprint of Paper to be submitted for publication in Proceedings of the
8th IAEA Technical Meeting on Control, Data Acquisition and Remote Participation for Fusion Research,
San Francisco, USA
(20th June 2011 - 24th June 2011)

ABSTRACT
The use of Programmable Logic Controllers (PLC) for automation of electromechanical processes is an industrial control system technology more and more in use within the fusion community. Traditionally PLC based systems usually are operated and maintained using proprietary SCADA systems (Supervisory Control and Data Acquisition). They are often not integrated with the in house developed control and data acquisition systems.

To interface PLCs and thus making the CODAS/CoDaC tools usable for PLC controlled components is an option to bridge this gap. Benefits are:

- remote control the PLC from the control room with the same tools as any other control and data acquisition system, ease user interaction
- archive PLC data together with the operational and/or physics data
- communicate read/write and read only variables, i.e. set points and control variables/measured values

At JET an inhouse “black box protocol” approach has been developed to communicate with any external system via a dedicated http based protocol. However, a PLC usually can’t be modified to implement this special protocol. Hence, a software layer has been developed that interfaces a PLC by implementing the PLC specific communication part on one side and the black box protocol part on the other side. The software is completely data driven i.e. editing the data structure changes the logic accordingly. It can be tested using the web capability of the black box protocol. Multiple PLC types from different vendors are supported, thus multiple protocols to interface the PLC are in use. Depending on the PLC type and available tools it may or may not necessary to program the PLC accordingly.

Wendelstein 7-X uses another approach. For every single PLC a dedicated communication from and to CoDaC is implemented. This communication is projected (programmed) in the PLC and configurable (data driven) on the CoDaC side. The protocol is UDP based and observed via timeout mechanisms. The use of PLCs for Wendelstein 7-X is standardized. Therefore a single implementation on the CoDaC side allows communication with any PLC. Measured data from the PLC is archived in the mass data store. Set points from CoDaC can be visualized from within the PLC visualization environment.

A comparison of both approaches and their advantages and disadvantages is given.

1. INTRODUCTION

1.1 THE FUSION EXPERIMENTS JET AND WENDELSTEIN 7-X

Fusion research aims to generate power by a controlled nuclear fusion process. Fusion devices in the world are unique experiments each having individual construction design and operation regime. JET – the Joint European Torus – is Europe’s largest fusion device and is collectively used by more than 40 European laboratories [1]. It is operating since 1984 and has been improved ever since. JET is a tokamak type machine operating pulsed. Wendelstein 7-X (W7-X) is a fusion device currently under construction with optimized magnetic field. It is intended to demonstrate that fusion devices of the stellarator type are suitable for power plants [2]. W7-X is planned to start operation in 2014. It will be operated steady state.

1.2 CODAS AND CODAC

To allow scientific exploration of a unique fusion device highly specialized tools are required. These tools are usually developed and maintained inhouse as is done with both JET and W7-X. Main common functionality is control and data acquisition.

CODAS & IT (Control and Data Acquisition System and Information Technology) is responsible for the design, implementation, testing, operation and maintenance of the software and hardware required for the Operation of JET.

CoDaC (Control, Data Acquisition and Communication) provides the tools to operate W7-X in a controlled and safe way, to allow observation and data archiving and to plan and evaluate the scientific and technological results.

1.3 USE OF PLCS

Operation of large scientific devices like JET and W7-X needs industrial standards for the automation of the electromechanical processes that "drive" the device. The use of industrial standards offers the required reliability, availability and maintainability.

PLCs (programmable logic controllers) are widespread in industries and machines. PLCs are well-adapted to a range of automation tasks. These are typically industrial processes in manufacturing where the cost of developing and maintaining the automation system is high relative to the total cost of the automation, and where changes to the system would be expected during its operational life. [3]

This makes the use of PLCs for automation of the fusion device equipment very attractive. When refurbishing existing components using a PLC is often the appropriate option. The PLC software provided by the vendor is usually specialized for factory applications and offers no easy integration to the fusion inhouse coda-software.

2. PLC INTERFACING

2.1 USED PLC TYPES

W7-X solely uses PLCs of type Siemens Simatic S7 and does not accept any other PLC types.

JET as a joint experiment potentially allows multiple PLC types depending on the selection of the association that developed the component. It ranges from high end to low cost PLCs and also covers Soft PLCs (a PLC emulation running on a local host computer). See Table 1.

2.2 MIDDLEWARE/PROTOCOL

Depending on the type the PLC vendors offer different protocols to interface the PLC. The large vendors have their own protocols e.g. Siemens S7 communication, the niche vendors tend to use standardized protocols e.g. the modbus protocol in various dialects [4].

The Application Programming Interfaces (APIs) evaluated vary between: closed source, open source, third party, C/C++, java, http, connectionless. Some PLCs are also offering multiple access methods. See Table 1.

2.3 PROTOCOLS USED WITHIN CODAS/CODAC

The JET CODAS connection to a PLC is done using the JET http based Black box protocol [9]. The concept of the black box protocol is to treat a component as a black box. It allows:

- Setting and reading back of plant equipment parameters,
- Reading of plant equipment state,
- Monitoring of plant equipment status,
- Setting up of plant equipment data channels for pulse data collection before a JET pulse and reading of the collected data after the pulse.
- Central logging or plant equipment error and warning messages.

In addition to the black box protocol some enhancements have been added to the PLC communication for test and convenience. They are planned to be added to future black box protocol versions.

The PLC connection is done using java allowing to running the coda part platform independent. It is implemented as eclipse application and is completely data driven via a XML file and an initialization file. Usually there is a single connection from the PLC to CODAS.

W7-XCoDaC communicates with a PLC using the Wendelstein7-X RtDataBus in the UDP option [10], [11]. It allows:

- Reading measured values and setpoints
- Setting setpoints and parameters
- Preparing W7-X for segmented machine operation
- Getting status Information (e.g. operational state, health information)
- Switching between autonomous and subordinated mode

The PLC connection is completely data driven via configuration database entries. It is implemented twice for the two types of control stations:

i) In Java as part of the Data Acquisition stations
ii) In C++ as part of the for Real time Fast Control Stations (running the real time operating system VxWorks)

Both implementations are in use and usually multiple connections to the PLC are alive for control and data acquisition purposes.

The connection of the PLC to the CODAS resp. CoDaC framework should be nearly as reliable and fast as the PLC itself. It proofed that all implementations are fast and reliable over months. Problems found were due to slow network connection and failures in the SoftPLC or in the host operating system.

2.4 INTEGRATION OF THE PLC

The control and data acquisition software of JET and Wendelstein 7-X has more similarities than differences. However, they use different vocabulary and different structuring.

Figure 1 and 2 makes the attempt to show schematically the location where the PLC is integrated into the control and data acquisition systems. Both figures show also the local human machine interface allowing for autonomous operation.

In figure1 one can see that the black box approach already separates the host computer from the actors and sensors. The additional hydra daemon wraps the black box protocol to the PLC specific protocol.

In figure 2 one can see that the PLC is able to communicate with the W7X DataBus protocol. In order to do so the PLC has to be programmed accordingly. This is well supported within the Simatic programming software and is no additional burden.

2.5 CASCADING PLCS

Some components either have such complex control programs or are physically distributed that multiple PLCs are in use.

The default strategy is to identify a master PLC and communicate from CODAS/CoDaC to the master PLC only. The master PLC communicates to the client PLCs using proprietary PLC internal communication which is seamlessly integrated into the PLC programming environment. Drawback of the strategy is the single point of failure. This risk can be taken because the PLC hardware is very robust. Moreover, cascading allows using cheaper PLCs of the same vendor behind the master PLC (e.g. Siemens models without advanced communication processor).

[For the JET Neutral beam enhancement system an alternative approach has been taken. There was the requirement to minimize changes to the existing CODAS framework application which was already spread over multiple components resulting in multiple connections to multiple PLCs per component.]

3. SUMMARY AND CONCLUSIONS

Two approaches for connecting PLCs to fusion software frameworks have been introduced.

At W7-X all components use the same PLC type. Some share a PLC if a dedicated PLC is not economical. Standardization of the PLC type eases the maintenance of the software.

At JET different PLC types are in use. However, all PLCs can be interfaces with a hydra daemon that hides the PLC specific part from the CODAS framework via the http black box protocol. New PLC types might need more implementation work if the existing implementations can't be reused and the PLC can't be cascaded and hidden behind a supported PLC. The existing implementations at JET show a wide variety of distributions. Nevertheless, JET could benefit from standardizing the PLC types wherever possible.

Both approaches for JET and W7-X are fully data driven. The use of java whenever possible allows a flexible and platform independent installation and distribution.

The PLC connections are extremely stable and reliable. They are nearly as stable as the host computer operating system. Any problems found are reported in table 1.

There is the potential to unify the approaches towards a common fusion PLC interface. This would require agreement on a single interface or protocol.

REFERENCES

[1]. http://www.jet.efda.org/jet

[2]. http://www.ipp.mpg.de/ippcms/eng/for/projekte/w7x/index.html

[3]. https://secure.wikimedia.org/wikipedia/en/wiki/Programmable_logic_controller

[4]. http://www.industrial-ethernet.org/ethernet-protocols-technologies/standard-ethernet-tcpip-dominates-industrial-installations/

[5]. Siemens Manual SIMATIC NET S7Beans / Applets for IT-CPs C79000-G8976-C180-02; http://support.automation.siemens.com/AU/view/en/22964892

[6]. http://www.schneider-electric.com

[7]. Festo Manual: The CPU I/O Module FEC FC34-FST Manual FEC FC34-FST

[8]. JNA java native access (download via https://jna.dev.java.net/)

[9]. Hogben, C. and Griph, S.: Interfacing to JET Plant Equipment Using the HTTP Protocol ; JDN/H(02)011; Issue 2 : 14th August 2008 (http://www.iop.org/Jet/fulltext/EFDR02004.PDF)

[10]. H. Laqua, H. Niedermeyer, I.Willmann, Ethernet based real time control data bus, IEEE Transactions on Nuclear Science **49** (2002) 478–482.

[11]. Ch. Hennig et al: Using the network as bus system for long discharge data acquisition and data processing; Fusion Engineering and Design **81** (2006) 1735–1740

PLC type	Protocol	PLC needs special programm	API	Middle-ware runs at	Used at	Problems
Simatic S7	Siemens S7 Communication (UDP)	Yes	Java standard library (java.net)	PC with JVM (windows, linux)	W7-X	None
Simatic S7	Siemens S7 communication (UDP)	Yes	C socket API	PC with VxWorks	W7-X	None
Simatic S7 with CP 343-1 Advanced	S7 beans	No	S7 beans API [5]	Solaris machine with JVM	JET	Needs high speed data link
Schneider Modicon Premium	unkown	No	FactoryCast library (com. schneiderautomation. factorycast) [6]	Solaris machine with JVM	JET	None
Festo FC640	Festo command interpreter [7] over http	Yes	java third party library (apache http client)	Solaris machine with JVM	JET	
IBHSoftec	Windows dll on localhost	No	JNA Java Native Access [8] interface to plc32.dll	PC Windows (host where Soft PLC is running)	JET	SoftPLC freezes sometimes

Table 1: PLC types and the subset of protocols and APIs used to interface the PLC. Comparison and problems encountered.

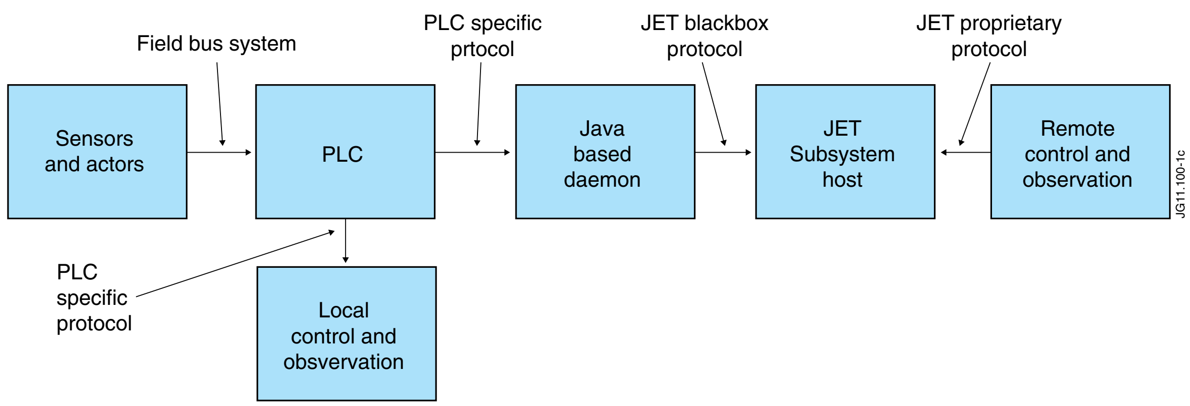

Figure 1: Schema of PLC integration at JET with the hydra daemon.

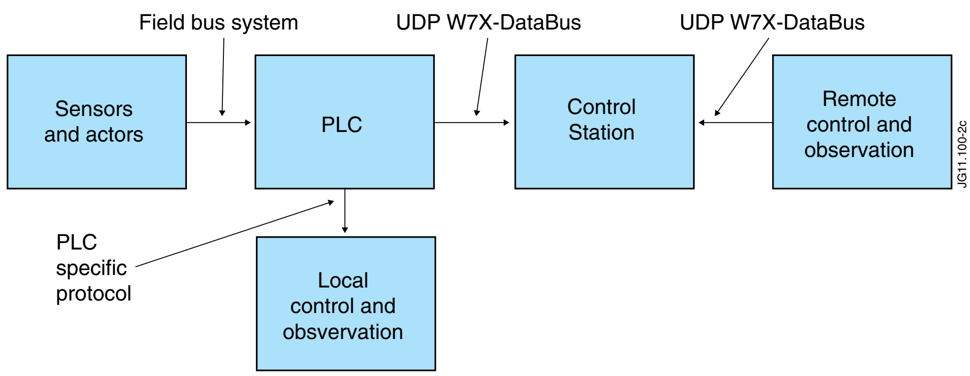

Figure 2: Schema of PLC integration at W7-X.