ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THPHA096

ALBA EQUIPMENT PROTECTION SYSTEM, CURRENT STATUS

A. Rubio, G. Cuní, D. Fernandez-Carreiras, S. Rubio-Manrique, N. Serra, J. Villanueva
ALBA-CELLS Synchrotron, Cerdanyola del Vallès, Spain

Abstract

ALBA [1] is the name of Barcelona's 3 GeV Synchrotron Light source. In operation since 2012, it currently hosts experiments 24/7 in its 8 beamlines with 3 more in development. The aim of ALBA Equipment Protection System is to avoid damage of hardware by managing sets of permits and interlock signals. The EPS scope covers not only ALBA accelerators and its beamlines but also the other existing laboratories like Radiofrequency, Optics, Vacuum, etc. It is built on B&R PLCs with CPUs installed in cabinets in ALBA service and experimental areas and a network of remote I/O modules installed in shielded boxes inside the tunnel and other irradiated zones. CPU's and Remote models are interconnected by the X2X field-bus. Signals managed by PLC's include interlocks, temperature readouts, flow-meters, flow-switches, thermal switches, shutters, pneumatic actuators, fluorescence screens, etc. This paper describes the design and the architecture of the Equipment Protection System, the current status, the tools used by the EPS team and the recent improvements in terms of reaction time and interaction with other systems via Powerlink and fast interlock system.

INTRODUCTION

The ALBA Equipment Protection Systems (EPS) [2] is a distributed PLC-based system autonomous from the TANGO Control System [3]. EPS is a homogeneous system based on B&R PLCs distributed along the facility. It became a versatile system that has been adapted for interlock, diagnostic acquisition and motion control in both, accelerators and beamlines.

EPS is being continuously improved and some changes has been made in the last times, among others: differential temperature Interlocks added, EPS Vacuum linked to the fast Machine Protection System (MPS) [4], Powerlink V2 [5] upgrade and adding all storage ring temperatures as interlocks. This last upgrade implied that no temperature diagnostics is available in the storage ring. To cover this need it’s been created a wireless diagnostics system [6].

Although other PLC based systems are used in ALBA to control the Radio Frequency circulators, Linac control, bake out controllers and water or air cooling systems; the EPS is the most complex system managed by PLC's, using 59 B&R CPUs and 128 periphery cabinets to collect more than 7300 signals. In addition to the main purpose of protection, several hundreds of signals distributed across the whole system are acquired for diagnostics and control of pressures, temperatures and movable elements.

EPS installation in numbers:

- 35% cables installed
- Powerlink installation (only Service area)
  - 1 CPU B&R master Powerlink
  - 15 Network switches
  - 43 CPUs connected
- Vacuum
  - 512 thermocouples from storage ring
  - 172 Vacuum pneumatic valves (VAT)
  - 166 Ion Pump Controller (DUAL)
  - 112 Vacuum gauge controller (MKS)
- Magnets:
  - 698 Thermal Switches
  - 376 Flow Switches
- 8 Radiofrequency plants preconditions and interlock management
- 11 Front End
- 8 Insertion Devices
- Building facilities, Cooling Diagnostics:
  - 14 flow reading
  - 81 pressure transmitter
- Diagnostics:
  - 26 Fluorescence Screen (2 position)
  - 11 Fluorescence Screen with OTR (3 position)

SOFTWARE TOOLS FOR EPS

The integration of the management of an independent system like EPS in the TANGO Control System required several phases, starting from the collection of cables from the Cabling Database to the final Auto-generation of UI's for both EPS Expert GUI and operator users (Taurus).

ALBA Cabling and Controls Database

The CCDB python API [7] provides full access to the Cabling Database from our control system tools. The API methods allow searching for equipment and getting lists of cables connections details.

Every cable and equipment installed in the ALBA Synchrotron is registered in our Cabling and Controls Database (CCDB). It was developed in 2007 by our Management and Information Software section (MIS) using MySQL and web technologies. It was the main support tool for the design and construction phase and now it is still kept updated as the main repository of equipment and configurations in our Accelerators and Beamlines (Fig. 1).

As of 2017 it lists 397 racks with 7594 equipment of different 1057 equipment types. These equipments are connected using 21010 cables of 638 different cable types with a total length of 215.8 Km.

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THPHA096

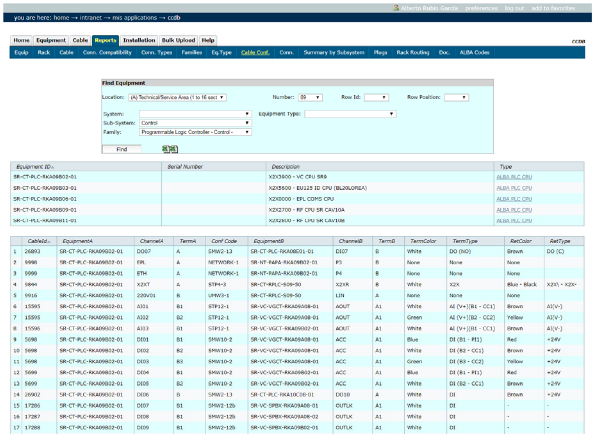

Figure 1: Cable Data base.

Auto-generation Tool

The CCDB python API provides full access to the Cabling Database from our control system tools. The API methods [8] allow to search for equipments and get lists of connections, names and network information. These links are used to enable our Auto-Generation coding tool correlating the information of the equipments from the CCDB with the logics defined for them in the EPS.

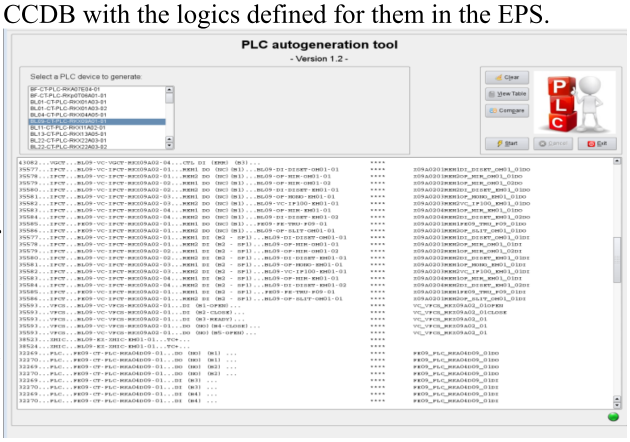

Figure 2: PLC Auto-generation tool.

For each CPU of the EPS, the PLC Auto-Generation tool (Fig. 2) extracts exhaustive reports of all equipment connected to the CPU and its remotes, parses the naming of these equipment and its logics depending on the cable and terminal connectors used in both sides (along with notations introduced by the Controls EPS engineers).

The final output of the Auto-generation process consists (Fig.3):

- spreadsheet VarDec: global variable declaration.
- spreadsheet Offset: Offset initialization to 0 (default)
- spreadsheet Refresh: Save/Load permanent values
- spreadsheet chekvars: Call variable suitable type function
- spreadsheet Init type: Initialize EPS variables
- spreadsheet valvesct: basic position control and command definition
- spreadsheet map: script to read command and post values under request from modbus comms.
- spreadsheet TANGO vars: variable modbus mapping full list of the PLC variables with its matching PLC code structures. (EPS CSV)

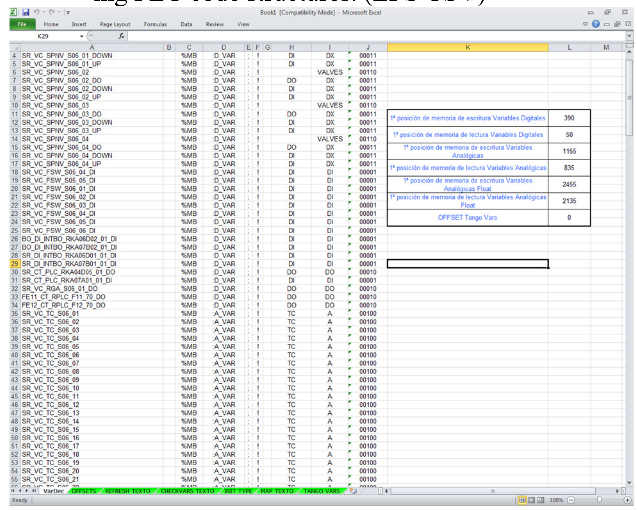

Figure 3: Auto-generation tool output files.

These steps do not produce the complete final code of the PLC (Fig. 4), but a detailed guide of the code and data structures to be used in the software project tasks. The control engineer then must coordinate logics of individual variables and program the most specific logics of every beamline or accelerators section, being capable of focusing in the most critical parts instead of the tedious, systematic and iterative variable naming task.

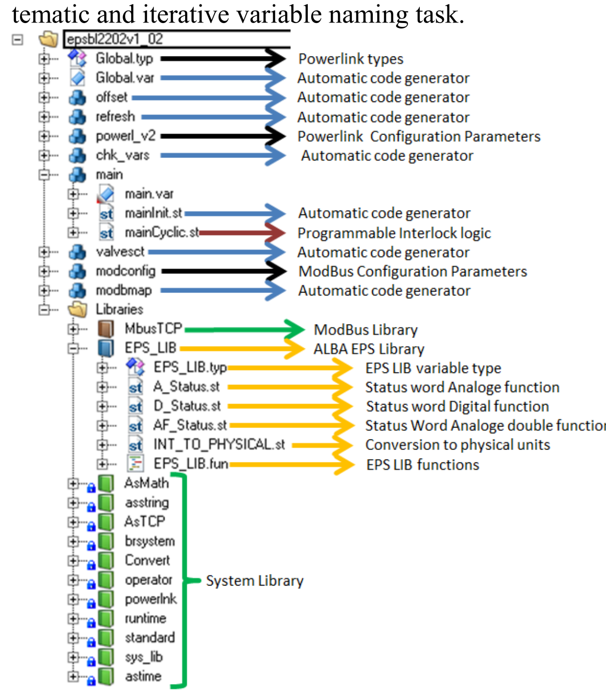

Figure 4: PLC project structure.

The final output of the Auto-generation process is the modbus' variable mapping spreadsheet, commonly known as the EPS CSV. This file will be used later to generate the EPS Expert GUI and the TANGO Attributes used by User-Level GUI's.

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THPHA096

EPS_LIB Tool

The EPS_LIB runs in the different CPU's and provides logic behaviour for managing each equipment type and signal connected to the Equipment Protection System, all the common elements have been standardized in our EPS_LIBRARY. This Automation Studio Project library also defines EPS variables types (Fig.5).

```
PLC_CONF                                D_VAR
    TotalDI      UINT                       Name         STRING[2]
    TotalDO      UINT                       Status       UINT
    TotalAI      UINT                       ForceTo      BOOL
    Status       UINT                       Filter       USINT
                                            Value        BOOL
A_VAR
    Name         STRING[2]              AF_VAR
    ForceTo      INT                        Name         STRING[2]
    Status       UINT                       ForceTo      REAL
    WarningUp    INT                        Status       UINT
    WarningDown  INT                        WarningUp    REAL
    AlarmUp      INT                        WarningDown  REAL
    AlarmDown    INT                        AlarmUp      REAL
    Filter       USINT                      AlarmDown    REAL
    scan_tick    TIME                       Filter       TIME
    Offset       INT                        scan_tick    TIME
    Value        INT                        Offset       REAL
    ValueINT     INT                        Value        INT
    Delta        INT                        ValueFLOAT   REAL
    DeltaT       TIME
    ValuesAT     INT[0..5]
```

Figure 5: EPS variable types.

These standard procedures include managing in/out digital signals for valves, alarm/warning ranges for analogue values and updating each variable status word (Fig.6)

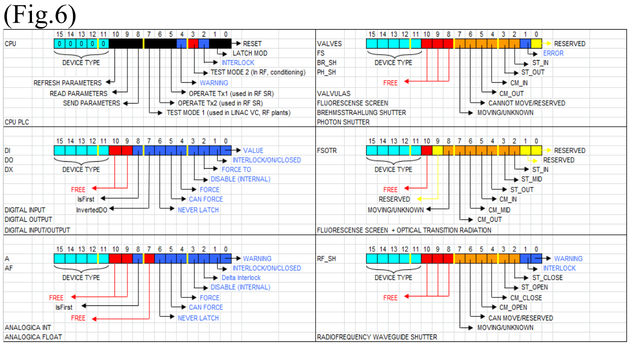

Figure 6: EPS variable status word

Status Word codification where information and commands for each variable is stored; commands as disable, forceTo,… and information as Value, Interlock, IsFirst,...

Value conversions for thermocouples, linear correlation for 4-20 mA or 0-10 V transducers and standard conversion scales for flowmeters amongst others are done using suitable conversion defined in the INT_TO_PHYSICAL script.

TANGO INTERFACES FOR EPS

The integration of the management of an independent system like EPS in the TANGO Control System [9] required several phases, starting from the collection of cables from the Cabling Database to the final Auto-generation of UI's for both EPS Expert GUI and operator users (Taurus).

PyPLC Device Attributes

All Modbus-based PLC's are accessed using the PyPLC TANGO Device Server. This python device server [10] (in use since 2008) have been developed over the Modbus C++ (Fig.3) device class to allow four different types of access:

- Reading the whole Modbus address space into array attributes, method optimized to get maximum update frequency, used by the Expert GUI.
- Exporting EPS variables as individual TANGO attributes, thus enabling TANGO features like ranges, alarms, archiving, labels, etc.
- Exporting EPS variables as individual TANGO devices, typically valves with its own Open/Close commands.
- The PyPLC exports the most standard Modbus commands (Read/Write Input/Holding Registers) and PLC variable types (Coil / Flag / Int / Long / Float / Ieee Float).

These dynamic generation of attributes boosted the initial prototyping of the devices, providing enough functionality for the most elementary systems (temperature controllers, stand-alone PLC's) and allowing customizing the attribute generation done by the EPS Auto-generation tools.

TANGO Dynamic Attributes

PLC's at ALBA are accessed using two protocols: Modbus (over TCP or RS485) for most of the systems (EPS, PSS, RF Circulators, and Cooling Diagnostics) and Fetch & Write for those Siemens PLC's (Linac Thales, Cooling Systems) [11].

User-defined formulas allow to create meta variables (e.g. BL_READY) or special commands to be used in PANIC Alarms [12] or our Sardana Experimental Framework to control pneumatic devices in the Beamline.

Taurus User Interfaces

The current application developed to interact with the EPS system is the EPS Expert GUI (Fig. 7). This application loads all the modbus mapped variables from the PLC and organizes the whole memory structure in several tabbed panels, grouped by variable type.

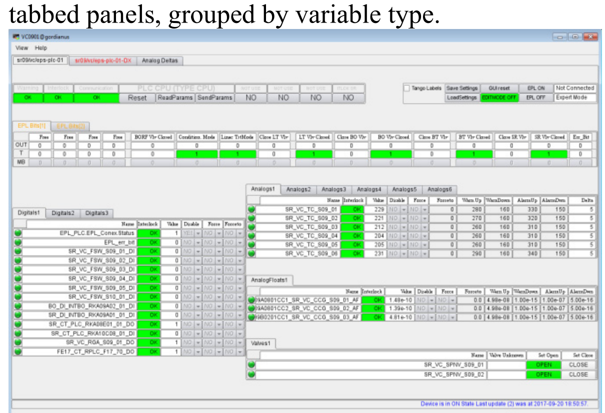

Figure 7: EPS expert GUI

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THPHA096

From the EPS GUI application we can manage, after LDAP user validation, the Alarm/Warning levels of every analogue signal, interlock status (active/latched/was_first) for analogue and digital inputs, forced and disable states both for interlocks and digital outputs. The status of the PowerLink word, which is synchronized in all the EPS CPUs to share interlock signals between them. The information regarding the several subsystems managed by the EPS (vacuum, magnets, Radiofrequency, bpm's, front-ends) is later summarized in 2 user-level applications: the ALBA-EPS, that allows fast diagnose and acknowledge of interlocks in the accelerators side, and the Machine Protection System. Many analogue and digital signals in ALBA beamlines have been exported to our Tango-based [13] experiment control framework, Sardana [14].

CURRENT STATUS

EPS has integrated some new functionalities or improvements in cycle speed.

ALBA Thermocouples

All Storage Ring temperatures (512) are treated as interlock, it means that no temperature diagnostics in the storage ring are available anymore. Since then, adding diagnostics temperature readouts will be not possible without upgrading the EPS system infrastructure which would have huge cost in shutdown time. Other solutions have to be evaluated

Incremental Interlock

Analog function has been upgraded with the new functionality incremental Temperature Interlock. Integration time is fixed to 1 second and the operator sets the increment of temperature. Above this value the PLC will generate an Interlock.

Interlock Board

Each EPS Vacuum sector has been connected to Timing using Interlock Board (INTBO) in order to stop RF plants using both systems Timing and EPS (MPS).

Powerlink V2

Powerlink is an open strict, deterministic real-time protocol based on Fast Ethernet (100 MBit). Time-isochronous transfer of cyclic I/O and motion data is supported along with asynchronous communication between network nodes; a part of network bandwidth is reserved for this.

The Ethernet POWERLINK Standardization Group (EPSG) is responsible for development and maintenance of the protocol [3]. POWERLINK V2 (open IEC standard) is the expansion of POWERLINK V1 (B&R proprietary) to include a standardized application layer. In other words:

POWERLINK V2 = POWERLINK V1 + CANopen.

For this reason, it is sometimes also referred to as "CANopen over Ethernet".

Analog Value Filter

Interlock will be generated only if it remains out of limits during a certain time fixed by EPS programmer for each variable.

Version Control

All EPS projects are stored using SVN to manage version control

FUTURE ROADMAP

The next targets for the project are:

- Time Stamp.
- Motion control.
- Version control (SVN->GIT?).
- PID control
- Safety functions

CONCLUSION

The design of the control system had as main premises being functional and economic. From the beginning, special attention was paid to cost and effectiveness. Choices done during the commissioning, like Ethernet as fieldbus or using compact and economic PLC's instead of VME (extremely popular in this type of installations), are working very well. The tendency is to reduce the number of computers and rely even more on Ethernet.

Moreover, in order to make the installation and maintenance easier, a central repository (so called cabling database or “ccdb”) was created. It soon turned out into a central repository for the Computing installation, kept up-to date and from where important pieces of software were automatically created. Some examples are the declaration of variables in PLCs, files for configuration of network services, dynamic attributes in device servers and components of graphical interfaces. This is a great tool for the maintenance since the information related to the installation is only in one place.

PyPLC TANGO Device provides a common interface to all PLC's at ALBA using the Modbus protocol. This developer-friendly interface allowed dynamic and effortless integration of new PLC signals into our Archiving, Alarm System and Beamlines SCADA (Sardana). This work-flow enabled by PyPLC reduced the time needed to upgrade or modify PLC systems in Beamlines.

Equipment Protection System is currently under continuous improvement adding new complete beamlines, front ends, insertion devices and equipments to the existing facilities and adding new functionalities to the EPS library.

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THPHA096

ACKNOWLEDGEMENT

The whole controls, electronics, Management Information System, Network System administration groups actively participated in the EPS tools, installation and tests of the control system.

Many ALBA engineers have collaborated in the PLC-related projects in the last years: D.Fernández, S.Rubio-Manrique, R.Ranz, R.Montaño, R.Suñé, M.Niegowski, M.Broseta, J.Villanueva and N.Serra. The collaboration of TANGO core developer, Emmanuel Taurel, was fundamental in the development of Dynamic Attributes templates and debugging of PyPLC performance.

REFERENCES

[1] ALBA website, `http://www.cells.es`
[2] R. Ranz *et al*., "ALBA, The PLC based Protection
[3] TANGO website, `http://www.tango-controls.org`
[4] D. Fernández-Carreiras, *et al.*, "Personnel protection, equipment protection and fast interlock systems", ICALEPCS'11, Grenoble, France (2011).
[5] Powerlink website: `http://www.ethernet-powerlink.org`
[6] R.Homs-Puron *et al.*, "Internet of Things (IoT): Wireless Diagnostics Solutions" ICALEPCS2017, Barcelona, Spain (2017).
[7] D. Beltran, *et al.*, "ALBA Control and Cabling Database", ICALEPCS'09, Kobe, Japan (2009).
[8] S. Rubio-Manrique, *et al.*, "A Bottom-up Approach to Automatically Configured Tango Control Systems", ICALEPCS'11, Grenoble, France. (2011).
[9] D. Fernández-Carreiras, *et al.*, "Alba, a Tango based Control System in Python", ICALEPCS'09, Kobe, Japan (2009).
[10] S. Rubio-Manrique, *et al.*, "PyPLC, A Versatile PLCto-PC python interface", PCaPAC'14, Karlsruhe, Germany (2014).
[11] S.Rubio, *et al.*, "Dynamic Attributes and other functional flexibilities of PyTango", ICALEPCS'09, Kobe, Japan (2009)
[12] S. Rubio-Manrique, *et al.*, "Extending Alarm Handling in Tango", ICALEPCS'11, Grenoble, France (2011).
[13] S. Rubio-Manrique, *et al.*, "Unifying All TANGO Control Services in a Customizable Graphical User Interface", ICALEPCS2015, Melbourne, Australia, (2015), WEPGF148.
[14] T. Coutinho, *et al.*, "SARDANA: The software for building SCADAS in Scientific Environments", ICALEPCS'11, Grenoble, France, (2011), WEPMS023.