Design specifications and test of the HMPID's control system in the ALICE experiment.

E. Carrone
For the ALICE collaboration

CERN, CH 1211 Geneva 23, Switzerland,

Enzo.Carrone@cern.ch

Abstract

The HMPID (High Momentum Particle Identification Detector) is one of the ALICE subdetectors planned to take data at LHC, starting in 2006. Since ALICE will be located underground, the HMPID will be remotely controlled by a Detector Control System (DCS).

In this paper we will present the DCS design, accomplished via GRAFCET (GRAphe Fonctionnel de Commande Etape/Transition), the algorithm to translate into code readable by the PLC (the control device) and the first results of a prototype of the Low Voltage Control System. The results achieved so far prove that this way of proceeding is effective and time saving, since every step of the work is autonomous, making the debugging and updating phases simpler.

I. INTRODUCTION

The HMPID DCS can be considered as made of five main subsystems: High Voltage, Low Voltage, Liquid Circulation, Physical Parameters and Gas. Each of them requires a specific control and all of the controls have to be integrated into the ALICE DCS mainframe. The HMPID DCS will be represented via a single interface which will include the above-mentioned systems and will be part of the whole ALICE DCS.

We will deal with three main subjects:

1. Providing a common way to represent and design the control system
2. Designing the Low Voltage control system
3. Presenting the first results of tests performed on the Low Voltage System.

A possible software architecture of the HMPID's control is shown in Fig.1. It actually mirrors the hardware architecture, since one can distinguish the three main layers: Physical, Control and Supervisor, each characterised by a specific functionality [1].

In fact, the lowest layer [2] will deal with PLC programming (by mean of Instruction List language) in order to read data from the physical devices (pressure and temperature sensors) and to send commands to actuators (switches, motors, valves).

The Control layer permits the communication between the other two layers: indeed, it translates data from the bottom into a language understandable by the SCADA (Supervisory Control and Data Acquisition system) software and also translates commands coming from the top into a language understandable by the PLC. The communications among the layers are accomplished via an OPC (OLE for Process Control) server. In addition, the PVSS DBASE (a module of the SCADA software) stores data for subsequently retrieval.

The supervisory level represents the highest control, since it runs control programs by means of Man Machine Interfaces remotely located.

The three layers communicate over the Ethernet via the TCP/IP protocol.

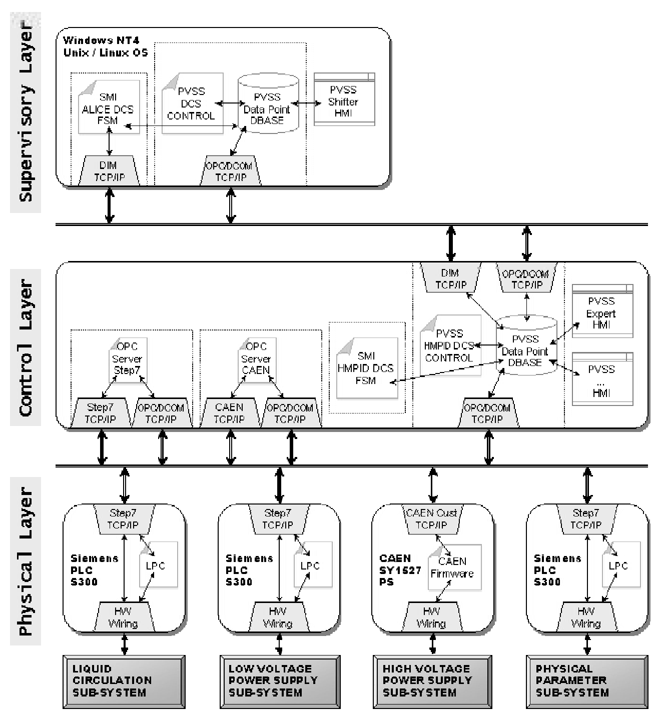


Figure 1: DCS software architecture

II. A SYSTEMATICAL APPROACH TO DCS DESIGN

Since we have to program the whole DCS (meaning that we have to deal with all of the three layers, and program PLC as well as SCADA systems) it is compulsory to establish a very well defined way of designing the system. This becomes necessary as many people are going to make intervention on the system itself; these people, in most cases, will not be control specialists. Clarity and portability are the two main concerns.

In order to satisfy to these needs, we have defined six fundamental steps required for the DCS design:

1. Definition of the Operations List.

The Operation List is the first tool we use to understand how the detector works. Actually, it contains as much details as possible about the specifications of the system.

The list has to be written in strong collaboration with the designers of the system, which are the most valuable source to understand the actions which are to be performed via the automatic control.

2. Description of the process as a Finite State machine (FSM).

This step represents the first attempt to interpret the system into a fashion closer to the control design: the Transitions Diagram describes the evolution of the system yet without going deep into the controls aspects, but giving a general idea.

3. GRAFCET modelling.

The GRAFCET language [3] is a further step towards the definition of the control system: not only it is a visual tool near to the FSM representation, but it is a powerful language useful for the description of whatever system. It means that it does not matter if one is going to program PLCs or SCADA: GRAFCET describes the system in a fashion which is completely independent from the hardware one will use. Furthermore, it is also simple and clear to non-control specialists. Among the other possibilities (i.e. Petri Nets above all) GRAFCET remains for us the best choice

4. Coding of GRAFCET into Instruction List.

The PLCs adopted hereby belong to the family of Siemens S-300. However, the procedures are applicable to any PLC. Moreover, since GRAFCET allows the design of very complex systems, the PLC language which best suits the needs for complex instructions managing and execution speed is the Instruction List (IL), included into the IEC 1131-3 rules [4]. In order to accomplish this task we developed an original algorithm to translate univocally the GRAFCET into IL. This step corresponds to the programming of the PLC.

5. Check of the parameters read by the PLC

Once the PLC runs its program, one needs to check how the program is running and the values read by, e.g., the ADC (Analog-to-Digital Converter) modules.

Siemens PLCs are supplied with the Step7 programming environment, which comprises the Variable Table (VAT) reading utility. It means that one can display the variables read by the ADC modules directly on the workstation used for programming.

6. Coding of the Man-Machine Interfaces into the SCADA PVSS environment.

At this step the PLC is running autonomously the control program, but the operations have to be performed by the operator manually (e.g. pushing buttons). To operate the system remotely one needs to program an interface at high level, by means of synoptic panels where each functionality of the system is represented and the user can send commands, read values, generate historical trends and so on. These panels are programmed into the PVSS environment, which is the SCADA adopted by CERN for all the LHC experiments' DCS.

All the subdetectors' DCS will merge into the most general control system, the Experiment Control System (ECS).

III. THE LOW VOLTAGE SYSTEM

The HMPID detector consists of seven modules, each sizing about 142 x 146 x 15 cm^3 and including three radiator vessels, a Multi Wire Proportional Chamber (MWPC), the Front End Electronics (FEE) and the Read-Out Electronics.

In [5] we have already reported some results from the Liquid Circulation sub-System, when the design phase was accomplished, along with some preliminary considerations on the High Voltage (HV) and Low Voltage (LV) subsystems.

In the following we will focus on the Low Voltage control system, starting from the control of the Power Supply units up to the Man-Machine Interface.

The system we will deal with represents a "custom" solution to provide the Low Voltage supply to the HMPID front-end and read-out electronics; as a result of the tests and the evaluations subsequently performed (costs, reliability, maintenance) we will be able to decide on the implementation of this solution for the whole detector.

In order to guarantee continuity of operations, even in case of faults, the "custom" layout is intended to split the available power into different channels via a PLC. The power supply of each module has been divided into six Low Voltage and High Voltage segments, and other four segments for electronics circuits. In this layout, a fault of a single chip will not compromise the functioning of the entire module.

A. *The apparatus set up*

We set up a test bench station in order to carry out some tests on a single Low Voltage power supply segment. A schematic representation of the test bench is shown in Fig. 2.

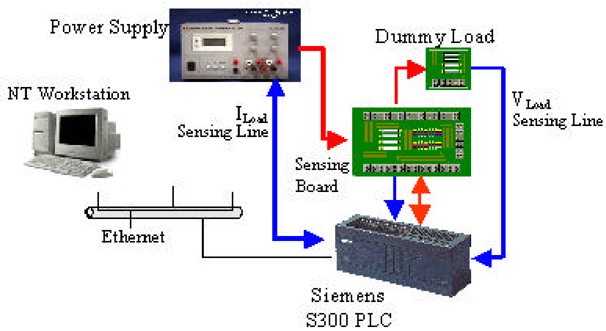


Figure 2: DCS software architecture

The power supply is an Eutron BVD720S, 0-8V, 0-25 A, 0.1±1 dgt. The PLC belongs to the S-300 Siemens family, equipped with two ADC 12 bit modules. The dummy load is made of resistors which represent the LV segment, while the "sensing board" is a resistors network needed for the current detection and the signal conditioning.

In fact, we measure the current drained by the load by means of the voltage drop on a "sensing resistor"; but, in order to overcome the common mode voltage U_{CM}=2.5 V, characteristic of the ADC input preamplifier, a resistor network has been designed and assembled. So, both the sensing resistor and network providing the signal conditioning have been placed on the sensing board.

Afterwards, the sensing board and the dummy load have been connected to the ADC module of the PLC, to get voltage and current values.

Fig.3 shows the electrical diagram of one bipolar channel, including the sensing wires.

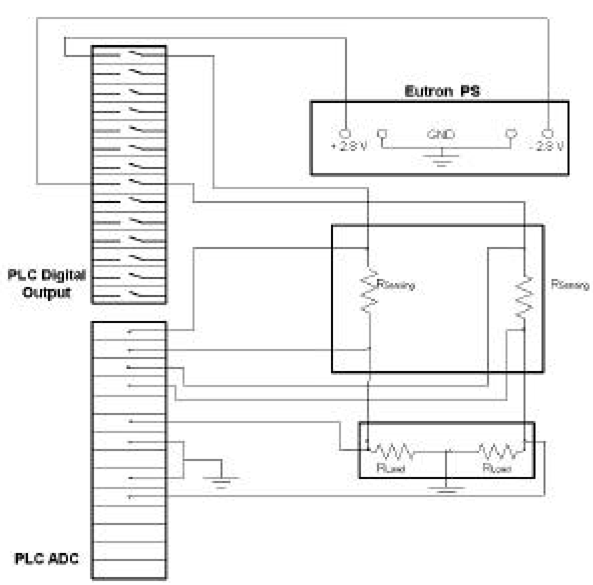


Figure 3: Test bench wirings

The scheme of the sensing board is shown in details in Fig.4.

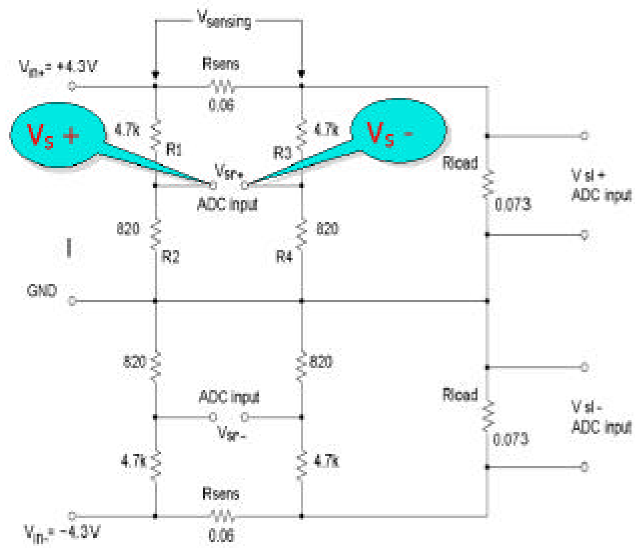


Figure 4: Sensing board scheme

The new voltage values are evaluated according to the following equation:

$$V_{sr+} = V_{+s} - V_{-s} = V_{in+} \cdot \left(\frac{R2}{R1+R2} - \frac{R4}{R3+R4} \right) + \\ + V_{sen\sin g} \cdot \left(\frac{R4}{R3+R4} \right)$$

The calibration of the sensing board let us provide the correct algorithm to the PLC program in order to present in the VAT the correct values of voltage and current.

Subsequently, the sensitivity obtained in this way amounts to 2.8 mA and is enough to detect even a single FEE chip failure.

B. *The LV control system*

According to the 6-steps list introduced above, first we study the system and write the Operations List; the most important constraint is given by the relationship with the High Voltage system: actually, the ON/OFF switching is the most critical, along with the current and voltage values.

When the LV chain has to be switched ON, since the FEE requires ±2.8 V, both these polarities must be supplied simultaneously.

When the LV is switched OFF, the facing HV segment must be checked: it must be turned OFF before the LV. This sequence is mandatory to prevent FEE breakdowns due to charge accumulation on the MWPC cathode pads. (In fact the ground reference to the MWPC sense wires is ensured through the FE electronics, then the low voltage at the corresponding FE electronics segment must be applied before the HV segment is switched ON).

Current and voltage must be within ranges:

$V_{\min} < V_{load} < V_{\max}$, $I_{\min} < I_{load} < I_{\max}$.

If $I_{load} > I_{\max}$, then the corresponding HV-LV segments must be automatically switched OFF, according to the LV switching OFF sequence.

The subsequent step is the design of the transitions diagram, as shown in Fig. 5.

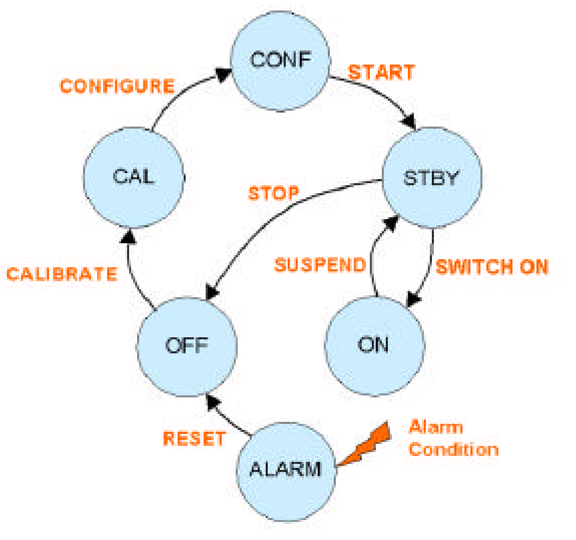


Figure 5: LV transitions diagram

After the OFF state, the first state encountered is CALIBRATE, which is intended to set voltages and currents out of the power supply; it means that no power is yet given to the FEE. Then, the CONFIGURE allows the user choosing how many (and which) segments he wants to power. In STBY the HV power is checked: this state is indispensable for a correct shut down procedure of the LV.

When the ON status is active, voltages and currents are monitored over all the FEE segments active at that moment. Whenever one of these values is out of range, the system goes into the ALARM state, the related segment goes OFF and a notification is sent to the HV system in order to set OFF the facing HV segment also.

The GRAFCET design follows the states just described. Actually, we have three Master grafcet which are needed to manage alarm and stop conditions, and a Normal grafcet to describe the normal evolution of the system, as in Fig. 6.

What has to be pointed out is that states 2 and 3 are actually Macro-States, meaning that they contain some other grafcet to manage the calibration and configuration of each segment. This way, the grafcet shown is the most general one, while the deeper control is demanded to the other sub-grafcet. This is a very useful facility to simplify the view of the system and concentrate on the general functioning.

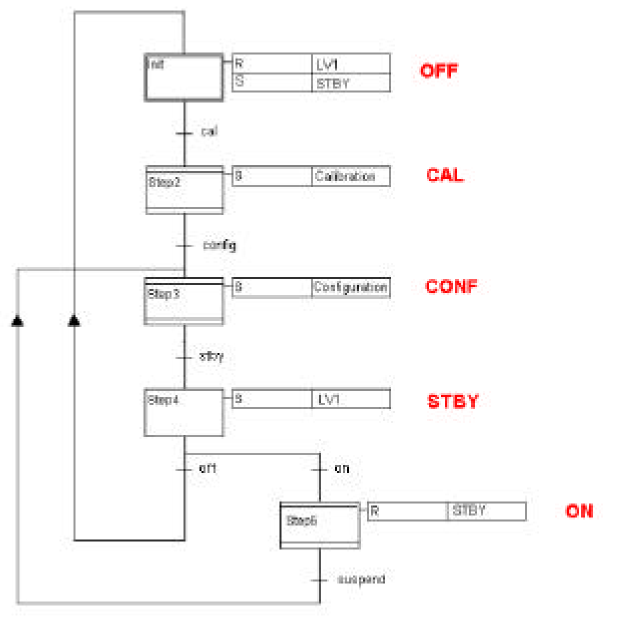


Figure 6: Normal grafcet

The algorithm we designed operates the conversion from grafcet (sequential and parallel processes) to Instruction List (a strictly sequential language), as in Fig. 7.

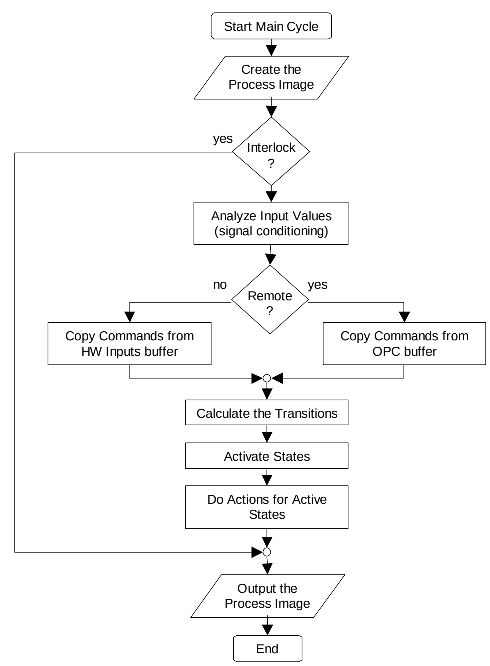
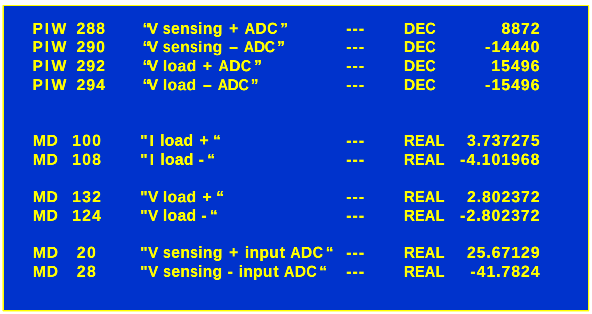


Figure 7: Grafcet → Instruction list conversion algorithm

The initialisation reads the input variables and decides whether to put them into a local or remote buffer, in dependence of the local/remote operation. Then, the transitions are evaluated: each of them will be considered crossed if the related condition is true and the preceding state is active. If the transition is crossed, the next state is activated, while the preceding state is deactivated.

The VAT shows the exactness of our calculations, as in Fig. 8.

The first elements (PIW) represent the raw data read by the ADC module: it is a decimal number in the range [-27648, +27648]. In order to read currents and voltages, we applied the algorithms for the offset correction. The final results are the "I_{load}" and "V_{load}" values. The last two elements are useful to check the real voltage going into the ADC module from the sensing board.



PIW	288	"V sensing + ADC"	---	DEC	8872
PIW	290	"V sensing – ADC"	---	DEC	-14440
PIW	292	"V load + ADC"	---	DEC	15496
PIW	294	"V load – ADC"	---	DEC	-15496
MD	100	"I load + "	---	REAL	3.737275
MD	108	"I load - "	---	REAL	-4.101968
MD	132	"V load + "	---	REAL	2.802372
MD	124	"V load - "	---	REAL	-2.802372
MD	20	"V sensing + input ADC"	---	REAL	25.67129
MD	28	"V sensing - input ADC"	---	REAL	-41.7824

Figure 8: LV VAT

Although not shown above, the VAT can also read the states of the system; we can check whether it is in OFF or ON or CALIBRATE, or whatsoever. Moreover, we can simulate alarm conditions via some switches that let us produce short circuits, or wiring interruptions.

The last point of our six-steps method consists in programming the Man-Machine Interfaces into the PVSS environment; these interfaces let the user operate the system, monitor parameters, perform actions, acknowledge alarms.

For instance, we monitored the values of current and voltage; the trend is shown in Fig. 9. It confirms subsequently the reading of the VAT, but presents the same data into a more readable fashion.

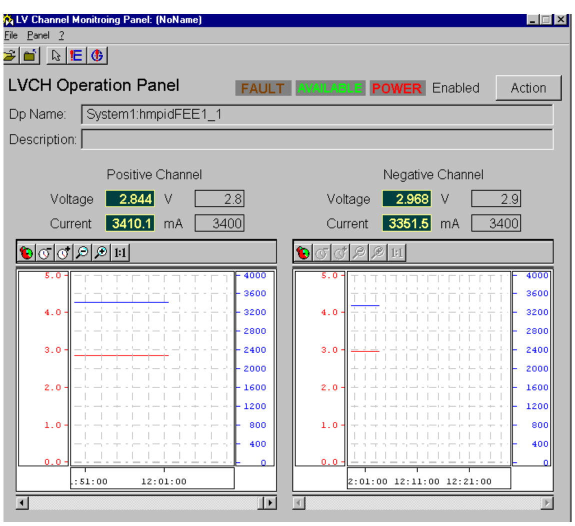


Figure 9: LV variables trend

In order to avoid a proliferation of interfaces different from each other, the JCOP (Joint COntrol Project) at CERN is releasing layouts written into PVSS and named "framework", in which dimensions, colours, positions of all the elements of the panels are defined, giving a coherent look to every control interface of whatever detector or experiment.

Our efforts are now directed towards the programming of all the panels according to the JCOP's framework guidelines. The first step will consist into the integration of both the Liquid Circulation and the low Voltage system into a single panel. The other control systems will follow and find place into the same framework, which will represent the whole HMPID DCS.

IV. Conclusions

The methodology hereby introduced and adopted has shown to be effective and time saving; in fact, it allows an easy interaction between control engineers and physicists in charge of the design and operation of the systems. The GRAFCET language has proved to be powerful and useful for the programming of the system at every level of hierarchy. Moreover, the measurements displayed on the VAT are readable directly also on a man-machine interface in form of diagram, making easy a monitoring over long times in order to check stability and performance of the power supply system.

V. Acknowledgements

The author would like to thank A. Franco for the fundamental contributions given during the design stage and the hardware set up as well.

Not only he gave precious aids in programming the SCADA systems, but also he helped in the whole PLCs environment.

VI. References

[1] Swoboda D., *The Detector Control System for ALICE : Architecture and Implementation*, 5th Conference on Electronics for *LHC* Experiments, Snowmass, 1999

[2] Lecoeur G., Määtta E., Milcent H., Swoboda D., *A control system for the ALICE-HMPID liquid distribution prototype using off the shelf components*, CERN Internal note, Geneva, 1998

[3] David, R. *Grafcet: A Powerful Tool for Specification of Logic Controllers*. IEEE transactions on control systems technology, Vol. 3, N.3, September 1995

[4] Seok Kim H., Young Lee J., Hyun Kown W., *A compiler design for IEC 1131-3 standard languages of programmable logic controllers*, Proceeding of SICE 99, Morioka, 28-30 luglio 1999

[5] De Cataldo G., *The detector control system for the HMPID in the ALICE experiment al LHC*, 6th Conference on Electronics for LHC Experiments, Krakov, 2000