# CLS SAFETY SYSTEMS*

Robby Tanner, Elder Matias, Hao Zhang, CLS, Saskatoon, Saskatchewan

*Abstract*

The Canadian Light Source has adopted the use of IEC 61508 [1] SIL 3 certified equipment and associated methods in the development of safety critical software. This paper examines the successful application of industrial safety rated PLC equipment in the development of accelerator and beamline safety systems. Of specific note is the application of this technology to a biomedical beamline at the CLS.

## INTRODUCTION

The Saskatchewan Accelerator Laboratory (SAL) operated from the 1960s to the late 1990s when the pulse-stretcher ring was decommissioned and the accelerator reconfigured as part of the injector for a new 2.9 GeV, 3rd generation synchrotron light source on the University of Saskatchewan (U of S) campus; the Canadian Light Source (CLS).

In general, the CLS preference is for off-the-shelf, commodity hardware over custom or proprietary offerings. Because of the small staff relative to other facilities of this nature, the CLS has taken a more industrially-based approach to providing solutions vs. one of internal R&D.

The CLS has gained valuable operational experience over several years developing Access Control and Interlock System (ACIS) for accelerator hall/tunnel and beamline hutches, CLS is now bringing this knowledge to bear on the Bio-Medical Imaging and Therapy (BMIT) beamline currently under construction. The focus of this discussion will be on the ACIS, which has the primary objective of preventing unintended exposure of personnel, users and the general public to radiation.

### *Regulatory Context*

Licensing for the CLS is regulated by the Canadian Nuclear Safety Commission (CNSC). The CLS is licensed as a Class 1B facility; as a result, the definition of internal processes is left to the CLS with the CNSC providing review, oversight and auditing.

### *Safety System Process*

Generally speaking, safety system development follows a V-model variant. For safety-critical implementations, the objective is to mitigate hazards and manage risk. Therefore, Requirements or Needs Analysis becomes a Hazard Analysis (HAZAN) [2].

Various requirements can be inferred and specified for the system from the results of the HAZAN. The design and installation naturally follow from there. Testing takes place in multiple stages. Integration and Unit testing verify that the design meets the requirements and that the installation is the same as the design, respectively. Validation, in the form of "black-box" testing, consists of ensuring that the anticipated hazards are eliminated; or that the constraints have not been violated.

### *Hazard Analysis*

Prior to the BMIT facility, ACIS requirements and methodology were grandfathered in from the LINAC hall system, which had been in use for over two decades. As a result, those areas did not undergo a formal HAZAN. The majority of the Phase II beamlines were developed in the same manner.

The potential for live human subjects in BMIT has made it unique from the other beamlines. A much more rigorous process is appropriate. The HAZAN and conclusions have been issued as an individual document.

### *Requirements*

Constraints imposed from the HAZAN become system requirements. At this stage additional requirements are incorporated; such as others identified as input documentation (i.e. Canadian Electrical Code, Medical Devices, Elevating Devices Act, R.S.O. 1990, c. E. 8, etc), human factors considerations [3] and any other design guidelines gleaned from operational experience such as providing a beamline lockout to bypass interlocks while the beamline is disabled for the sake of convenience.

A combination of qualitative and quantitative specifications is used. Formal definitions can be convenient at various stages for showing that two adjacent stages are identical; comparing Requirements to Design, for example. In some instances, however, tracing a formal specification through the entire process can be onerous if not impossible. For example, it is difficult to derive an equation that justifies human factors decisions. In such cases, a "handwaving", qualitative justification is sufficient.

All functionality is performed in the PLC. Safety-Critical functions (Safety Instrumented Functions or SIF in the IEC 61508 parlance) are backed up by an independent, redundant relay chain. Such an arrangement introduces simplicity and diversity for an extra layer of safety or defense-in-depth.

## DEVELOPMENT

### *Hardware*

Hardware that has been certified by the TUV up to SIL-3 in accordance with IEC 61508 has been selected for the safety-critical functions. These modules provide a layer

* The research described in this paper was performed at the Canadian Light Source, which is supported by NSERC, NRC, CIHR and the University of Saskatchewan.

of diagnostic functions by default when configured to run in SIL2/SIL3 mode; such as checking for shorts, open-circuits and imposing timing restrictions on code execution, communications, etc. All of these parameters can be set by the developer along with timing requirements identified in the HAZAN.

The TUV SIL-3 rating has been achieved in spite of the fact that the CPU module does not contain physical, redundant logic solvers. Instead, each operation executed has its complementary function executed on complementary operands and the two results compared. Thus, the calculations are shifted in time using multiple channels; employing time redundancy instead of structural redundancy

The hardware allows for mixing standard and failsafe I/O modules. Regular plant operations can be conducted by less expensive commodity hardware and mission-critical safety functions allocated to the more expensive, specialized certified equipment.

The BMIT ACIS uses mixed standard and failsafe I/O. Failsafe modules only operate at SIL-2 owing to the parallel, redundant relay chain monitoring safety-critical functions. Less critical features, such as user displays, use standard I/O.

Remote I/O communicates with the CPU using Profisafe. The CPU communicates with Engineering Stations (ES) over industrial ethernet.

### *Software*

The base software package is Siemens Process Control System SIMATIC PCS 7 Version 7.0 SP1. Two other significant add-ons are the failsafe library (Systems) and hardware simulator (PLCSIM).

### *Code Generation*

Development itself takes place, as does most PLC development, starting with the hardware configuration; consisting of CPU, network and I/O parameter assignment.

The user program is built using Siemens Continuous Function Chart (CFC), which is a visual environment following a hardware wiring paradigm. Blocks are connected together in much the same way that pins from various physical devices would be wired together.

The code is organized hierarchically. Blocks are placed in charts, which in turn are placed in runtime groups (RTG), although typically only one chart is placed in a runtime group and only one SIF per chart. RTGs are placed in organizational blocks (alternatively referred to as OBs, tasks or interrupts). Some OBs are called cyclically, such as OB1 which is the main scan cycle of the CPU. By default, failsafe code is placed in OB35 which has a higher priority than most other OBs and is called every 100ms, interrupting lower priority tasks if necessary. Other OBs are asynchronous and called in the event of a fault, such as rack failure (OB86) or a diagnostic error (OB82) including short/open circuit detection, for example.

Part of the fail-safe runtime library is the automatic generation of module drivers which handle communication with the fail-safe hardware. For each physical I/O, a channel driver block is connected to the module drivers which handle channel level diagnostics and communication.

As with the hardware, both standard and failsafe blocks may co-exist on the same platform (albeit in different RTGs) and does at the CLS.

## TESTING/DEBUGGING

Up to this point, most testing had been done in situ. The PLC was taken offline, a new version uploaded and tested until it was validated by HSE or the old version restored if not. This was inefficient for a variety of reasons; testing could only take place during outages in which time is highly competed for and small errors would result in long test-repair-test cycles.

PCS 7 comes with a variety of debugging and diagnostic utilities and features.

### *Online Tracing*

Initial testing is provided by creating variable tables (VATs) that can be used to manipulate inputs and memory areas and observe the response. Online debugging is also available from inside the CFC editor. Signals can be traced graphically to troubleshoot errors. The channel drivers have inputs for simulating values and turning the simulation mode on and off.

### *Diagnostics*

The failsafe runtime has a host of common functions that are transparent to the user. Such monitoring watches for corrupted or bad data, electrical faults (open and short circuits, for example) and enforces timing constraints. In the event of a fault, the associated OB is called to handle the event. Typically, the OB is also called when the fault has cleared.

### *Simulation*

In order to streamline development, a test bed was constructed with a programming workstation and physical Siemens hardware connected to switches and LEDS for simulating I/O. The “hardware” simulator is a full-fledged plug-in replacement for hardware in the field. The approach allowed for offline testing and faster discovery of small errors, reducing total validation time and enhancing pipelining of activities.

Recently, the Siemens software simulator, PLCSIM, has been able to download safety libraries for testing. The availability of an online, local simulator allows for even faster testing of compiled blocks and other software.

Using the simulator has reduced development times considerably, and has the potential to make our validation more efficient.

## LESSONS LEARNED

While the runtime engine performs a variety of diagnostic activity on the developers' behalf, it is important to be aware of what they are.

In an early installation, a number of self-latching circuits were spuriously resetting. The cause was traced to diagnostic functions that test for open and short circuits on failsafe digital outputs (F-DO), called "dark" and "light" periods.

From the Siemens documentation [4]:

"Dark periods occur during switch-off tests and during complete bit pattern tests. This involves test-related 0 signals being switched to the output by the fail-safe output module while the output is active. The output is then switched off briefly (dark period). A sufficiently slow actuator does not respond to this and remains switched on."

In this case, "sufficiently" slow means > 1ms. The devices being used were opto-isolators with a fall time around 0.5 ms. The relays were fast enough to respond, permitting the latching circuit to drop out.

Since the latching circuits were in the relay chain, the event pointed out that designs grandfathered from the LINAC were not independent of each other. It also showed that we were not testing the chains individually, both of which were remedied in subsequent designs.

The failsafe runtime also performs "light" tests:

"Light periods occur during complete bit pattern tests. This involves test-related "1" signals being switched to the output by the fail-safe output module while the output is de-activated (output signal "0"). The output is then switched on briefly (light period). A sufficiently slow actuator does not respond to this and remains switched off."

If a sufficiently fast relay were used to set a latching circuit, the diagnostic function could conceivably enable an interlock outside of the intended function where it could sit unnoticed for an indefinite period of time.

Experience has also pointed out that the requirements are an area where a common-mode failure can arise. If requirements are wrong or a hazard is missed, everything that follows after will be flawed no matter how well it is implemented or how meticulously the development is monitored. The Hazard Analysis should be vetted thoroughly.

Proper evaluation of a hazard and response definition will greatly reduce inconvenience down the road. Separating life-safety functions, from less-important and/or nearly-trivial features (i.e. most user interface elements) and allocating them appropriately avoids unnecessary complexity, reducing development time and likely enhancing safety.

Having the same personnel work on both chains can lead to the use of practices suitable for one system but not the other. For example, in a Ready Chain, the designer will use a number of dry contacts in series to enable a permissive. Having developed a habit of that line of thinking, they may then incorrectly use relay outputs in the PLC chain, when they could be adding the signals in code with a software, boolean AND gate and controlling a single physical relay with the result.

It is a better practice to assign individuals to only one chain of the project for the sake of diversity.

## CONCLUSION

The Siemens SIMATIC S7-400 family of failsafe hardware and software offers a variety of features for designing, developing, unit-testing, verifying and validating safety-critical applications. However, safety is not solely a reliability exercise. For all its many offerings, intensive development program and TUV-certification, the package may still have flaws. If, for whatever reason, the CPU jumps to the wrong location and starts executing arbitrary code or even data, it is difficult to predict the outcome.

While great strides have been made in the automation field to enhance safety, the onus is still on the user to become intimately familiar with their operation.

Conversely, the failure modes of a relay are well-known, the consequences easily predicted and few. The use of a hardware chain has been very useful in helping to identify safety-critical features and those which should be allocated elsewhere. The inconvenience of implementing a given function in both hardware and software has acted as a good test, among others, for determining its candidacy as a SIF.

The comprehensive, all-encompassing approach used at the CLS has prevented unknowns and unk-unks from putting workers at risk.

The CLS will continue to use a relay-based chain to backup simple, life-safety functions. Large gains can be achieved by focusing on improving internal processes to correctly identify, monitor and categorize the functions appropriately. Concentrating efforts on process, moreso than the tools in question, will have the two-fold affect of saving time and enhancing safety.

## REFERENCES

[1] IEC 61508-5. 1998. Functional Safety of electrical/electronic/programmable electronic safety related systems

[2] Matias, E. 2008. *BMIT Hazard and Risk Analysis*. CLS Tech Doc. 26.2.37.1

[3] McKibben, M. 2007. *Human Factors Workscope*. CLS Tech Doc. 0.1.1.1

[4] Siemens 2001. SIMATIC – Automation System S7-300 Fail-Safe Signal Modules, Edition 02/2001, Page 3-1