ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-TUSH202

THE LASER MEGAJOULE FACILITY: PERSONNEL SAFETY SYSTEM

M. G. Manson, CEA, Le Barp, France

Abstract

The Laser MegaJoule (LMJ) is a 176-beam laser facility, located at the CEA CESTA Laboratory near Bordeaux (France). It is designed to deliver about 1.4 MJ of energy to targets, for high energy density physics experiments, including fusion experiments. The first 8-beams bundle was operated in October 2014 and a new bundle was commissioned in October 2016. The next two bundles are on the way. The presentation gives an overview of the Personnel Safety System architecture, focusing on the wired safety subsystem named BT2. We describe the specific software tool used to develop wired safety functions. This tool simulates hardware and bus interfaces, helps writing technical specifications, conducts functional analysis, performs functional tests and generates documentation. All generated documentation and results from the tool are marked with a unique digital signature. We explain how the tool demonstrates SIL3 compliance of safety functions by integrating into a standard V-shaped development cycle.

LMJ PROCESS HAZARDS

LMJ process hazards types are laser, high voltage, and radiations. These hazards are transmitted between bays as shown in Fig. 1. High voltage hazard are generated in capacitor bays and transmitted to laser bays. In laser bays, electrical energy is transformed into laser energy. Laser beams then travel to target bay, where physics experiment occurs. Experiments may generate radiations (X, neutrons…). These radiations may be transmitted to some diagnostics rooms.

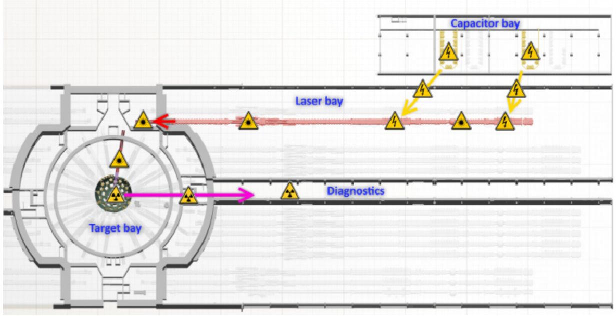


Figure 1: LMJ hazards.

PERSONNEL SAFETY SYSTEM

The PSS protects personnel by managing risks presence and transmission between bays, using safety interlocks and transmission barriers. The PSS manages personnel presence using access control and doors switches.

Conception follows International Electrotechnical Commission 61508 standard. The PSS is built around two systems named "BT1" and "BT2". These systems are designed using different technologies. Both BT1 and BT2 systems manage hazards and the presence of staff. Figure 2 shows PSS architecture.

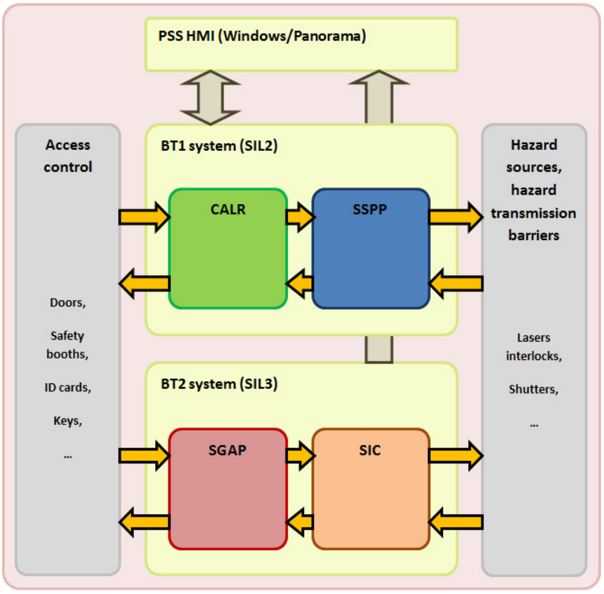


Figure 2: PSS subsystems.

The PSS architecture is detailed in a previous paper [1].

BT1 Subsystem

The BT1 system is designed using programmable technology, following IEC61508 requirements to achieve Safety Integrity Level 2. It is composed of two subsystems named SSPP ("Système de Sécurité du Personnel Programmé" – Programmed Personnel Safety System) and CALR ("Contrôle d'accès des Locaux à Risques" – Hazardous Premises Access Control).

SSPP subsystem manages all process hazards (lethal and non-lethal) of LMJ facility, such as pointing laser beams hazard.

CALR subsystem performs access control on all process bays using safety booths and contactless ID cards.

The BT1 system is operated through a computer HMI.

It is currently operational.

BT2 Subsystem

The BT2 system is designed using non-programmable technology, following IEC61508 requirements to achieve SIL3. BT2 logic is built using PLANAR4 products from HIMA. It is composed of two subsystems named SIC ("Système d'Inter verrouillage Centralisé" – Centralized Interlock System) and SGAP ("Système de Garantie d'Absence de Personnel" – Absence of Personnel Proof System).

The BT2 system focuses on nuclear safety and on lethal hazards.

SGAP subsystem performs access control only in bays where radiation hazard or lethal hazard could occur, using keys and door electrical locks.

The BT2 system is operated through physical interfaces such as keys and buttons.

The BT2 system will be commissioned on Q3 2018.

BT2 logic is dispatched into 24 PLANAR4 racks. 10 racks control accesses (6 personnel access booths, 4 equipment transfer rooms), 5 racks implement the core SGAP logic, and 9 SIC racks manage beams bundles.

BT2 SIMULATOR

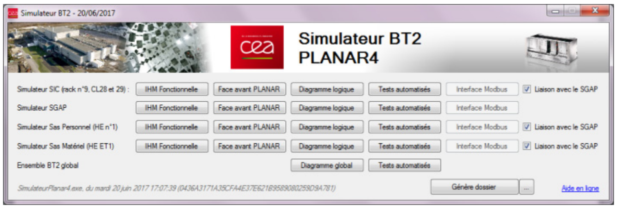
While being simpler than BT1 system functions, BT2 safety functions are much more complex than classical wired safety systems. We developed a software simulator to help us in the BT2 system engineering. BT2 simulator software was developed using Microsoft Visual Studio and C# programming language.

BT2 simulator can be used to:

- Check subsystems internal functions ;
- Check subsystems interactions ;
- View logic modules status ;
- View logic diagram ;
- Validate functional definition and generate functional analysis report ;
- Run functional tests and generate simulated test report ;
- Generate off-site and on-site test plans ;
- Generate wiring specifications ;
- Simulate bus communications used by computer HMI software ;
- Manage versioning by adding digital signature annotation on every generated document.

Simulator HMI

Main menu window. Main menu window is shown in Fig. 3. This is the default window. It gives access to all other windows, allows selection of partial or global simulation, and presents a menu to start global report generation.



SimulateurPlanar4.exe, du mardi 20 juin 2017 17:07:39 (0436A3171A35CFA4E37E621B9589080259D9A781)

Figure 3: Simulator main window.

Functional view windows. Figure 4 shows SGAP core functional view. Functional views show process states, and allow manually triggering of any external signal. A functional view is available for each BT2 subsystem (personnel booth rack, equipment transfer rack, SGAP core racks, SIC rack). These views have some current state helpers. They can be used to train future operators of the system.

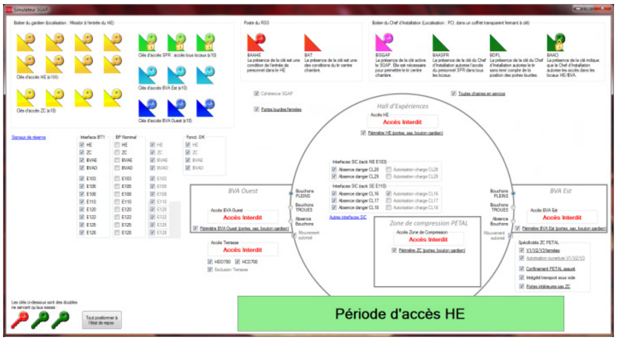


Figure 4: SGAP core functional view.

Technical view windows. SGAP technical view is shown in Fig. 5. Technical views show rack faces, including status indicators and configuration buttons. These allow virtually disconnecting any module and simulating fuse states.

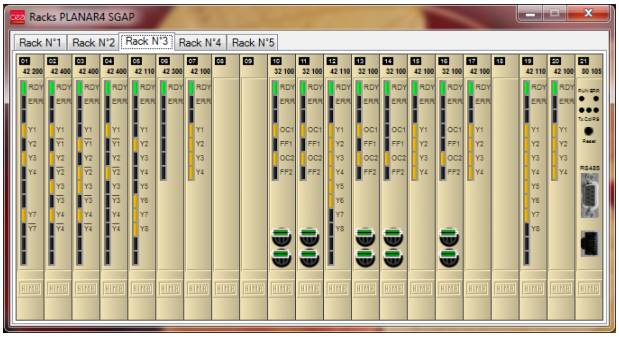


Figure 5: SGAP technical view.

Logic diagram windows. Figure 6 shows part of SGAP core logic diagram. Logic diagram windows show all the racks logic. All signals may be forced or monitored. Always visible spy windows are available on every signal.

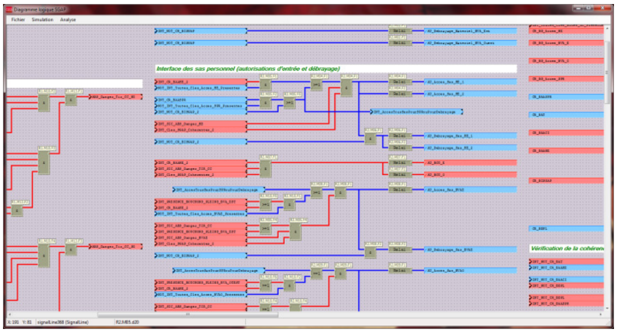
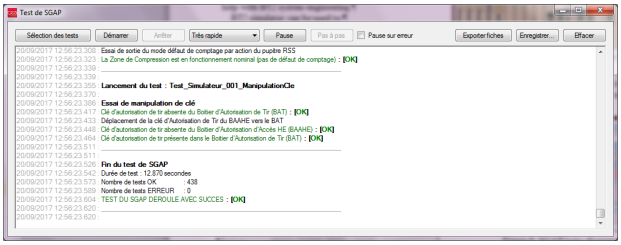

Fichier
Simulation
Analyse
Interface des sas personnel (autorisations d'entrée et déblayage)
Vérification de la cohérence

Figure 6: SGAP core logic diagram.

Test windows. A test window is shown in Fig. 7. These allow selecting and running tests. Tests can be interrupted and run step by step. Test report can be saved. Manual test instructions can be generated.



```
20/09/2017 12:56:23.308 : Essai de sortie du mode défaut de comptage par action du pupitre RSS
20/09/2017 12:56:23.323 : La Zone de Compression est en fonctionnement nominal (pas de défaut de comptage) : [OK]
20/09/2017 12:56:23.339 :
20/09/2017 12:56:23.339 :
20/09/2017 12:56:23.355 : Lancement du test : Test_Simulateur_001_ManipulationCle
20/09/2017 12:56:23.370 :
20/09/2017 12:56:23.386 : Essai de manipulation de clé
20/09/2017 12:56:23.417 : Clé d'autorisation de tir absente du Boitier d'Autorisation de Tir (BAT) : [OK]
20/09/2017 12:56:23.433 : Déplacement de la clé d'Autorisation de Tir du BAAHE vers le BAT
20/09/2017 12:56:23.448 : Clé d'autorisation de tir absente du Boitier d'Autorisation d'Accès HE (BAAHE) : [OK]
20/09/2017 12:56:23.464 : Clé d'autorisation de tir présente dans le Boitier d'Autorisation de Tir (BAT) : [OK]
20/09/2017 12:56:23.511 :
20/09/2017 12:56:23.511 :
20/09/2017 12:56:23.526 : Fin du test de SGAP
20/09/2017 12:56:23.542 : Durée de test : 12.870 secondes
20/09/2017 12:56:23.573 : Nombre de tests OK         : 438
20/09/2017 12:56:23.589 : Nombre de tests ERREUR     : 0
20/09/2017 12:56:23.604 : TEST DU SGAP DEROULE AVEC SUCCES : [OK]
20/09/2017 12:56:23.620 :
20/09/2017 12:56:23.620 :
```

Figure 7: Test window.

All these windows operate on the same simulator instance. You may step a test and watch what is happening on functional view. You may remove a logic module on technical view and watch affected signals on logic diagram view.

Simulator Inputs

Simulator compares two distinct system models. One is the functional model; the second is the technical model. Both models operate on signals.

The simulator being a one shot tool, models are hard written in code using some high-level objects.

Sets of input and output signals are prerequisites of defining both technical and functional models.

The functional model is defined by:

- A set of functions ;
- Some optional functional sequences ;

A function can be complex or simple. Complex functions are aggregation of complex or simple functions. A simple function defines a constraint between one or more input states and one or more output states. All functions are result of a safety analysis.

Functional sequences describe a graph of states and transitions. States define input states and expected output states. Transitions define the state change of one or more inputs. One state is the default state.

Technical model is defined by populating one or more racks with logic modules, and by virtually wiring modules signals with input and output signals.

Tests are written as sequence of input signal state change and output signal state checks.

Simulator Outputs

Simulator generates:

- Wiring specifications ;
- Blank test files ;
- Simulated tests reports ;
- Functional analysis report ;

Wiring specifications are a human readable form of technical model.

Blank test files are a human readable form of test operations.

Simulated tests reports are generated using both technical model and tests operations. All test operations are virtually run on input signals. Technical model computes output signals states. Comparison of output states and test checks validates the tests. Results of comparisons are appended to the test report.

Functional analysis is generated using both technical and functional models. Functional analysis begins with all simple functions validation. For each simple function, all input, output and internal signals are set to an undefined state. Then the simple function inputs are first set to defined state. Finally, the expected output states are compared to computed output states. Next step is functional sequences validation. Simulator will start by computing all paths from default state to default state. Then, for each path, the simulator will run transitions and check excepted input and output states.

DEVELOPMENT PROCESS

BT2 system is complex. Its logic has to be validated before wiring to catch errors before their correction become too costly. By allowing running process steps up to system tests before starting wiring, the BT2 simulator helps defining logic an iterative way. Figure 8 shows simulator managed steps of BT2 system development cycle.

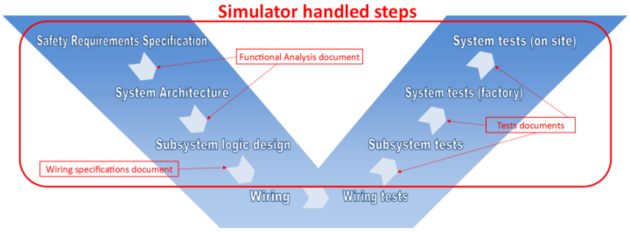


Figure 8: BT2 development cycle.

The simulator runs all tasks to generate a full report in about one hour on any standard office computer.

As we rely on simulator to validate functions and to generate wiring specification, we considered it had to be audited like code verification and generation tool following IEC61508 part 3. This audit work is currently in progress.

CONCLUSION

BT2 simulator helps us to develop a complex safety system using wired technology. We are able to validate a logic definition against specifications and tests before any wiring.

REFERENCES

[1] J. C. Chapuis, J. P. Arnoul, A. Hurst, and M. Manson, “The laser MegaJoule facility Personnel Security and Safety Interlocks”, in *Proc. 13th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'11)*, Grenoble, France, Oct. 2011, paper WEPMU009, pp. 1070-1072.