MARY KAY O'CONNOR PROCESS SAFETY CENTER
TEXAS A&M ENGINEERING EXPERIMENT STATION

20th Annual International Symposium
October 24-26, 2017 • College Station, Texas

How and When Do I Validate, Proof Test and Re-Validate My SIS logic Solver?

Prasad Goteti,
P. Eng, FS Expert (TÜV Rheinland), CFSE
Safety Engineering Consultant
Honeywell Process Solutions
Houston, TX, USA

KEYWORDS

Basic Process Control System (BPCS), Distributed Control System (DCS), Safety Integrity Level (SIL), Safety Instrumented Function (SIF), Safety Instrumented Systems (SIS), Proof Test Interval (PTI), Diagnostic Test Interval (DTI), Validation, Proof Test, Re Validation,

INTRODUCTION

A major component of every Safety Instrumented System (SIS) is the Logic Solver. It plays an important role as it is the sub-system which performs the logic in every Safety Instrumented Function (SIF). But sometimes there is confusion as to what constitutes:

1. Validation of the SIS logic solver
2. Proof Test of the SIS logic solver
3. Re-Validation of the Logic Solver

This paper will attempt to clarify the above with an explanation based on IEC 61508, 61511 (ISA84.00.01). It will answer the questions “When and How” for each of the activities listed.

BASIC CONCEPTS

A Safety Instrumented System (SIS) is a system that implements Safety Instrumented Functions (SIF) to maintain or to bring back a process into a safe state. Those SIFs are classified into four

different levels depending on the probability that they will be successful when asked to perform. That probability based level is known as the Safety Integrity Level (SIL).

A SIF is composed of three distinct sub-systems: Sensor elements that detect certain process conditions, a logic solver that through some logic determines if those conditions are unsafe and sends commands to restore a safe state, and actuating devices that act on the appropriate process variables, executing the commands of the logic solver. So for a SIF to meets its SIL value, all three sub-systems need to perform with a certain amount of reliability.

TYPES OF COMPONENT FAILURES

IEC 61511-part 1 defines **Random Hardware failures** as "failure, occurring at a random time, which results from a variety of degradation mechanisms in the hardware". Online Diagnostics help detect SIS component hardware failures which can be dangerous leading the SIS component to "fail to function" on demand. The objective of Proof Testing is to reveal potential dangerous undetected failures which are not revealed by the online Diagnostics of the SIS component.

IEC 61511-part 1 defines **Systematic failures** as "failure related in a deterministic way to a certain cause, which can only be eliminated by a modification of the design or of the manufacturing process, operational procedures, documentation or other relevant factors." To reduce or eliminate Systematic errors, which are generally design based errors (both in the hardware and software of a SIS component), checks need to be carried out whenever there is a modification done to the validated SIS component after commissioning. IEC 61508 has recommended techniques to reduce the Systematic errors after a SIS is commissioned.

RANDOM HARDWARE FAILURE RATE

Random Hardware Failure rate (λ) is defined as the Number of failures per unit time

$\lambda = \lambda_S + \lambda_D$, where :

λ_S = Safe Failure, failure which does not have the potential to put the safety related system in a hazardous or fail to-function state

λ_D = Dangerous Failure, failure which has the potential to put the safety-related system in a hazardous or fail-to-function state

Additionally :

Safe failure (λ_S) can be categorized as :

$\lambda_S = \lambda_{SD} + \lambda_{SU}$, where

Safe detected (λ_{SD}) = DC_S * $\lambda_{S,,}$ Safe failures detected by online diagnostics while the component is in use

Safe Undetected (λ_{SU}) = (1- DC_S) * λ_S, Safe failures not detected by online diagnostics while the component is in use and can only be detected during the Proof test of the component

DC_S is defined as the Diagnostic Coverage of Safe failures

Dangerous failure (λ_D) can be categorized as :

$\lambda_D = \lambda_{DD} + \lambda_{DU}$, where

Dangerous detected (λ_{DD}) = DC_D *$\lambda_{D,}$ Dangerous failures detected by online diagnostics while the component is in use

Dangerous Undetected (λ_{DU}) = (1- DC_D)*$\lambda_{D,}$ Dangerous failures not detected by online diagnostics while the component is in use and can only be detected during the Proof test of the component

where DC_D is defined as the Diagnostic Coverage of Dangerous failures

Diagnostic Coverage (DC):

Fraction of failures detected by automatic on-line diagnostic tests.

For dangerous failures, the fraction of dangerous failures is computed by using the dangerous failure rates associated with the detected dangerous failures divided by the total rate of dangerous failures

SIL VALUE OF A COMPONENT BASED ON RANDOM HARDWARE FAILURES

The Probability of failure on demand value for a component, in its simplest form, is expressed as PFDavg in all SIL calculations. This is because, a SIF component can fail on demand anytime between two periods of it being tested, the Proof Test Interval (PTI) and so the PFD value is averaged out over the PTI to calculate $PFD_{AVG.}$

PFDavg = (λ_D. PTI) / 2

The PFDavg equation modifies as a sum of two parts, the Dangerous Undetected Failures (λ_{DU}) part which will only be detected at the next proof test and the Dangerous Detected Failures (λ_{DD}) part that would show up during the Diagnostic Test Interval (DTI):

PFDavg = (λ_{DU}. PTI) / 2 + (λ_{DD}. DTI) / 2

The PFDavg of a SIF is the sum of the PFDavg of each of the sub-systems (ie the sensor, logic solver and final element).

Indicating the above as an equation:

PFDavg of SIF = PFDavg of sensor + PFDavg of logic solver + PFDavg of final element

Sometimes the SIL value is also expressed in terms of the Risk Reduction Factor (RRF):

RRF = 1 / PFDavg of SIF

Table 1 indicates the relation between SIL, PFDavg, RRF and Reliability

Table 1 - SIL, PFDavg, RRF and Reliability

SIL	Average probability of failure to perform its design function on demand (PFDavg)	Risk reduction Factor (RRF)	Safety Function Reliability
4	10^{-4} to 10^{-5}	10000 to 100000	≥ 99.99% to < 99.999%
3	10^{-3} to 10^{-4}	1000 to 10000	≥ 99.9% to < 99.99%
2	10^{-2} to 10^{-3}	100 to 1000	≥ 99% to < 99.9%
1	10^{-1} to 10^{-2}	10 to 100	≥ 90% to < 99%

SIL VALUE OF A COMPONENT BASED ON SYSTEMATIC FAILURES

Presently there is no mathematical way to quantify Systematic failure rate. However both IEC 61508 and 61511 have recommended various ways to reduce Systematic errors. A systematic capability scale (SC1 to SC4) measures the confidence that the Systematic safety integrity of a component meets the requirement of the specified SIL. As an example:

1. A “Proven in use” process while designing and manufacturing components with good Quality checks will help reduce Hardware and Software Systematic failures.
2. Poor design and detailed engineering can introduce Hardware and Software Systematic errors

As an example to illustrate point 2 above, a Programmable Logic Solver may come with a SIL3 certification from a third party (such as TUV, Exida) based on Random and Systematic failures, but if the application program (code) is not written , implemented and tested properly for a Process application, new Software Systematic failures may be introduced.

SAFETY LIFE CYCLE

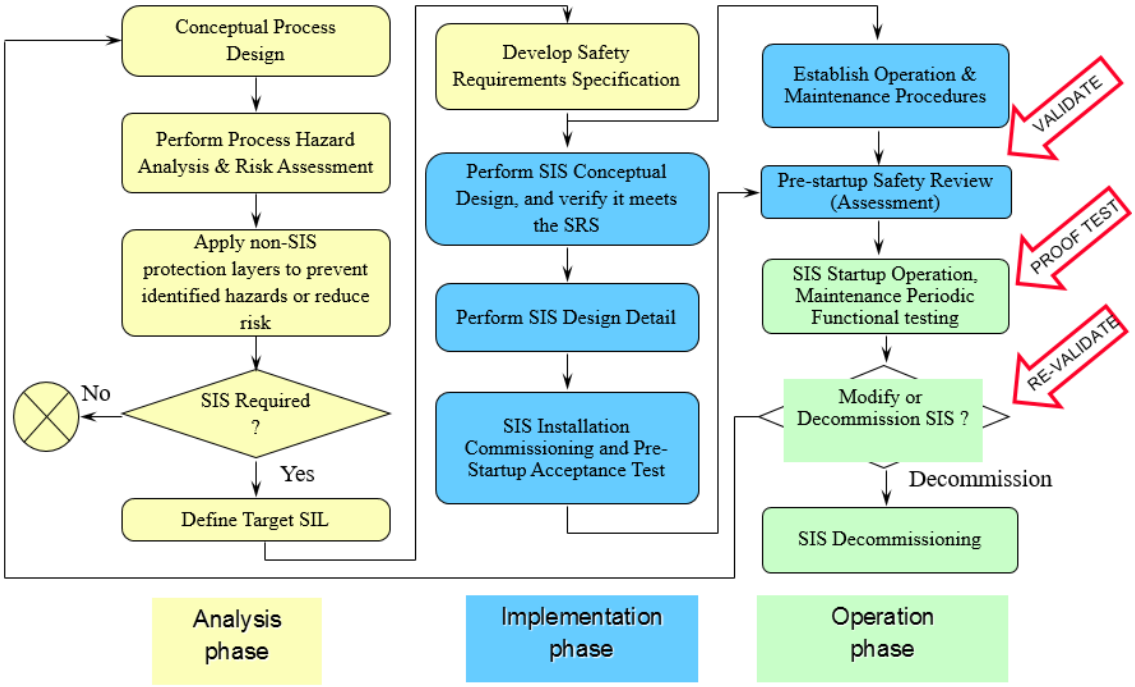
As per IEC 61511, “necessary activities involved in the implementation of safety instrumented function(s) occurring during a period of time that starts at the concept phase of a project and finishes when all of the safety instrumented functions are no longer available for use” is the definition of the Safety Life Cycle.

The Analysis phase of the Safety Life Cycle is the conceptual phase in which the various SIFs of a SIS are identified with their SIL values and ends with the generation of the Safety Requirement Specification (SRS)

In the Implementation phase the SIS is engineered and it ends with the **Validation** of the SIS

During the Operation and Maintenance phase, which usually goes on till the life of the process unit, the SIS is regularly **Proof tested** and if any modifications are made to the SIS, it is **Re-validated** either in part or fully

Figure 1 - Safety Life Cycle



SIS VALIDATION

As per IEC 61511, Validation is defined as the "activity of demonstrating that the safety instrumented function(s) and safety instrumented system(s) under consideration after installation meets in all respects the safety requirements specification".

When? - Validation of the SIF including the Logic Solver during commissioning and process startup will help reduce Systematic failures. This ends the Implementation phase of the Safety Life Cycle

SIS Validation should include the following activities:

1. **Testing the Hardware and Software (where applicable) of every SIF component**. These test maybe done for each SIF component separately (i.e. when not yet hooked up together in the field). If a SIF component comes with a "SIL certificate" and "calibration certificate" for first time use, one should make sure that the details certified meet the requirement of the SIF application. This step also includes validation of the SIS logic solver as per details given below.

   Validation of the SIS logic solver could be done independently as a Factory Acceptance Test (FAT) and / or when installed, wired up with field instruments and powered up at site. Usually a FAT is conducted so that before the complete SIS validation at site, there is enough time to fix errors if found. SIS logic solver validation should include:

   A. Testing the Logic Solver Hardware to make sure :
      a. All Hardware components of the Logic Solver are installed and functioning as per the requirements in the SRS
      b. No Random Hardware failures (example – failure of a communication module after installation)
      c. No Systematic Hardware Failures (example – a 24 VDC output module installed instead of a 120 VAC output module)

   B. Testing the Logic Solver Software (if it is a Programmable Logic Solver) to make sure :
      a. All SIF components interfaced to the Logic Solver are configured correctly
      b. All SIFs are programmed to function as specified in the SRS
      c. All SIFs are programmed to have Human interface as required
      d. No Systematic Software Failures (examples – a 1oo2 input voting programmed for a SIF which needed a 2oo3 input voting, incorrect range of an input module etc.)

By the end of this Validation step, all SIF components, including the SIS logic solver, are ready to be functionally tested together.

2. **Functional testing of each SIF per the requirements in the SRS**. This is done once the SIF components are installed in the field , "hooked up" to the process and wired up to the SIS logic solver. By this time the loop checks should be complete, the SIS logic solver is powered up after successful installation and the process is ready for startup.

   Functional testing per the SRS requirements would include the following (and more):

   A. SIF functionality on demand. For example - A MooN input on High alarm should trip a valve. The process should be simulated to reach High alarm, ideally from the sensing device and not electronically from the transmitter, and checking if at least M out of N devices reach the alarm value which will trigger the SIL logic solver to trip the valve.
   B. SIF reset after demand and all alarms are cleared.
   C. Test procedure to bypass one (or more) of the SIF inputs.
   D. SIF input bypass philosophy – Behavior of (M-x)ooN voting when one or more (x) inputs is bypassed.
   E. SIF input failure philosophy – Behavior of (M-x)ooN voting when one or more (x) inputs has failed (either the input sensor, transmitter or the SIS logic solver input channel / module).

When the SIS Validation step is completed, the SIS is ready for the process unit to start up.

SIS LOGIC SOLVER PROOF TEST

As per IEC61511, Proof Test is defined as a "Test performed to reveal undetected faults (both Random and Systematic) in a safety instrumented system so that, if necessary, the system can be restored to its designed functionality".

When? - SIF component Proof Testing needs to be done after SIS commissioning based on the Proof Test Intervals (PTI) considered for each SIF component while calculating the component PFDavg. Proof Tests are done during the Operation and Maintenance phase of the Safety Life Cycle

During the operation of the SIS, some Random dangerous failures will be detected by the online diagnostics of the SIS logic solver and the extent to which it is detected is defined by the Diagnostic coverage associated with it. Dangerous failures which are not detected by the online diagnostics should be detected during the Proof test assuming the Proof test coverage is 100%, i.e. all dangerous failures not detected during operation are detected during proof test and rectified.

Most SIL3 capable SIS logic solvers available in the market today have a very high level of online diagnostics which run automatically in the background. The Diagnostic coverage is a minimum of 99%. What that means is most of the dangerous failures are detected during the Diagnostic Test Interval (DTI) which is usually in seconds, i.e. 99% of any new dangerous failures are detected (λ_{DD}) within the next DTI and indicated as a system alarm. Only 1% of the dangerous failures are NOT detected by the online diagnostics and these remain as dangerous undetected failures (λ_{DU}). These failures could potentially cause the SIS logic solver to fail dangerously on demand.

For Random Hardware failures, as we have seen earlier, PFDavg is:

PFDavg (t) = (λ_{DU}. PTI) / 2 + (λ_{DD}. DTI) / 2

As an example, for a SIL3 logic solver, consider:

1. Dangerous failure rate, λ_D = 1.25E-07 failures per hour
2. Diagnostic Coverage for dangerous failures, DC = 99%
3. Then Dangerous Detected failure rate, $\lambda_{DD} = \lambda_D$ x DC = 1.237E-07 failures per hour
4. And Dangerous Undetected failure rate, $\lambda_{DU} = \lambda_D$ x (1-DC) = 1.25E-09 failures per hour

Considering a PTI of 10 years and DTI of 3 seconds:

PFDavg of SIS logic solver = (1.25E-09 x 87600) / 2 + (1.237E-07 x 8.33E-04) / 2

(10 years = 8760 x 10 = 87600 hours, 3 seconds = 3/3600 = 8.33E-04 hours)

PFDavg of SIS logic solver = 1.09E-04 + 1.03E-10

Eliminating 1.03E-10 which is a very small number:

PFDavg of SIS logic solver = 1.09E-04, which is in the SIL3 range

The question now is, what failures are detected by the online diagnostics and what tests need to be done during the Proof test to detect the failures NOT detected by the online diagnostics.

Failures which are not detected by online diagnostics in general would relate with those failures which are in the incipient stage of development or not detected in earlier tests like validation. Examples are:

1. High relative humidity, High temperature and High dust content in the SIS logic solver system cabinet which could lead to hardware failure of the electronic modules.
2. Any loose connections of system cables which could lead to failure of communication between modules

Some Systematic failures which could be detected and fixed during the Proof test are:

1. A bug in the firmware of the Programmable Logic Solver informed by the Logic Solver supplier after validation of the SIS, which could potentially fail the logic solver on demand.
2. Poor quality of power input (say 130 VAC, 65 HZ) to the SIS Power supply (which can tolerate only till 125 VAC, 63 HZ)

Most SIS Logic Solver suppliers will provide a recommended maintenance checklist and a suggested time period for maintenance. If the suggested time period is less than the PTI used to calculate PFDavg of the SIS logic solver, then the end user should follow the time period suggested by the SIS logic solver supplier for Proof Testing the SIS logic solver.

Some suggested Proof testing activities for Logic Solver hardware are checking and replacing or fixing the following:

1. Cable damage between SIS Logic Solver modules
2. Voltages to the Control Processor if within the tolerable limits
3. Temperature in the Control Processor if within the tolerable limits
4. Airflow obstruction to various modules
5. Presence of any earth faults
6. Availability of spare parts

Suggested Proof testing activities for Logic Solver Software to primarily reduce systematic errors are:

1. Making sure the latest running application software has been backed up.
2. If there has been a change in the firmware of the Logic Solver, it is recommended to upload the new firmware.
3. If the new firmware was already uploaded online earlier or now, or if there were some modifications done to a SIF or SIFs, it is recommended to do a complete functional test of all the SIFs as done during validation. The reason is to make sure that a firmware change or modification to a validated application software has not in any way affected the functioning of all SIFs

SIS LOGIC SOLVER REVALIDATION

IEC61511 / 61508 do not define Revalidation directly. But it could be defined as, “activity of demonstrating that a modified safety instrumented function(s) and safety instrumented system(s) under consideration after modification meets in all respects the modified safety requirements specification”

When? – Revalidation of a SIS is done during the Operation and Maintenance phase of the Safety Life cycle usually when:

A. Additional SIFs may get added, or existing SIFs may get modified or deleted, during the next cycle of a Process Hazard Analysis (usually every 5 years in the USA as per OSHA regulation – 29 CFR 1910.119) or during a system audit or assessment.
B. Modification of a SIF based on Operational feedback, for example – too many spurious trips, too many demands etc.
C. Change of SIS logic Solver or other SIF components due to excessive Random and / or Systematic failures

1. Modifications to a validated SIS are usually based on a Management Of Change (MOC) process. The MOC process generally details:

    A. Personnel in the company who will authorize the modification
    B. Reason for the modification to the SIS
    C. Steps of the Safety Life Cycle that need to be revisited. If the change is due to a new risk assessment scenario, then restart from "Process Hazard Analysis", refer Figure 1
    D. Who and when will detail and implement the subsequent steps in the safety life cycle
    E. SIF components effected by the modification
    F. Impact analysis to make sure that this modification :
        a. Does not lead to any new potential hazardous events, either during implementation or after the modification.
        b. Does not effect other SIFs in the same SIS
    G. Implementation of the modification
    H. Revalidation before "startup" of the modified SIS
    I. Update of all documentation to reflect the changes done during the modification

2. Modification to a SIF includes some or all of the following activities:

    A. Modification to any of the SIF components :
        a. addition of SIF components (example, modifying a 1oo2 Input to a 2oo3 input to reduce spurious trips)
        b. deletion of SIF components (example, removing a Valve from a SIF output which is no more required for process safety application)
    B. Modification to the hardware and / or application software (if programmable type) of a SIS logic solver based on changes to other SIF components, and / or change in philosophy of the SIF functionality itself (example, if Input is modified from 1oo2 to 2oo3, the logic changes and so does the degradation philosophy for bypass and bad inputs)

3. Extent of SIS Revalidation

IEC61508, part 3, Table A.8 (refer Table 2 below), lists the activities and recommendations during a software modification for SIS Programmable Logic Solver. These recommendations can be extended to every type of SIS logic solver and for hardware modifications too. These techniques and measures will help detect and reduce both Random and Systematic failures.

Table 2 - SIS Modification

(R = Recommended, HR = Highly Recommended)

	Technique/Measure *	Ref.	SIL 1	SIL 2	SIL 3	SIL 4
1	Impact analysis	C.5.23	HR	HR	HR	HR
2	Reverify changed software module	C.5.23	HR	HR	HR	HR
3	Reverify affected software modules	C.5.23	R	HR	HR	HR
4a	Revalidate complete system	Table A.7	---	R	HR	HR
4b	Regression validation	C.5.25	R	HR	HR	HR
5	Software configuration management	C.5.24	HR	HR	HR	HR
6	Data recording and analysis	C.5.2	HR	HR	HR	HR
7	Forward traceability between the Software safety requirements specification and the software modification plan (including reverification and revalidation)	C.2.11	R	R	HR	HR
8	Backward traceability between the software modification plan (including reverification and revalidation)and the software safety requirements specification	C.2.11	R	R	HR	HR

A. **Testing the Hardware and Software (where applicable) of every SIF component effected by the modification**.

Extent of testing (whether all SIFs or only effected SIFs) will depend on Table 2. Procedure is similar to the testing of each SIF component during Validation

B. **Functional testing of each SIF effected by the modification**.

Extent of testing (whether all SIFs or only effected SIFs) will depend on Table 2. Procedure is similar to the Functional testing of each SIF during Validation.

When the SIS Revalidation step is completed, the SIS is ready for the process unit to start up.

CONCLUSION

The Safety Life Cycle as defined in IEC 61508 and 61511 emphasize the need to manage functional safety during all phases of the cycle. As part of managing functional safety, it is vital that:

1. Starting from the commissioning of the SIS when it is **validated** and thereafter, it is operated upon and maintained properly
2. Periodic **proof testing** to make sure that all components of the SIS are still in "good shape"
3. Following a proper management of change (MOC) procedure whenever any modifications need to be done to the SIS. Such modifications would generally involve **revalidation** of either the effected parts of the SIS or the complete SIS depending on the risk reduction levels being catered to by the SIS

Table 3 - Summary Table for SIS Logic Solver

Activity	When?	Why?	How?
SIS Logic Solver Validation	Just before taking SIS logic solver online for the first time	Hardware test to detect Random failures and software to reduce Systematic failures	Test logic solver Hardware and Application Software
SIS Logic Solver Proof test	During the regular maintenance of the SIS Logic Solver dictated by SIL calculations or SIS vendor	Hardware test to detect Random failures by looking for potential errors not detected by online diagnostics and software to reduce Systematic failures	Test logic solver Hardware and Application Software if change in firmware or any modifications have been done
SIS Logic Solver ReValidation	When modifications have been made to a validated SIS logic solver and before taking it online	Hardware test to detect Random failures and software to reduce Systematic failures	Test logic solver Hardware and Application Software. Extent of test will be based on Table 2, which is based on SIL ratings of SIFs in the SIS

REFERENCES

1. ANSI/ISA 84.00.01 (IEC-61511). "Functional safety – Safety instrumented systems for the process industry sector".
2. IEC-61508, "Functional safety of electrical/ electronic/ programmable electronic safety related systems".
3. "SIS Design Basis Revalidation", white paper by Kenexis