Logic solver application software and operator interface

By RJ Perry, Control Systems Consultant

Correctly implemented and structured functional logic, together with operator interface displays, can improve overall functional safety management and reduce systematic failures leading to fewer process plant accidents.

A great deal of technical literature exists on sensors with associated self diagnostics and final element ESD valves with trip action diagnostics and partial stroke testing. However, there seem to be far fewer application examples available on the logic solver functional logic structure, including process operator graphics indicating the SIS status and operator/maintenance interface. IEC 61508 Part 3 [1] clearly defines the objectives of safety-related application software, but as with IEC 61511 [2], provides few practical examples.

Prior to 1998, there were few standards dedicated to functional safety management as applied to programmable instrumentation systems, and initially there was a reluctance to use ‘software’ in safety applications due to issues with reliability and systematic ‘bugs’. A typical electromechanical trip relay system as shown in Figure 1 was the standard design of the day, but although reliable with well defined failure modes, it suffered from many drawbacks such as system modification and lack of intelligent communications.

One of the earlier functional safety publications in 1987 was the UK Health & Safety Executive 2 part guideline on ‘Programmable Electronic Systems in Safety Related Applications [3]’ and in 1989 the German DIN 19250 [4]. In 1996 ANSI/ISA published S84.01 – ‘Application of Safety Related Systems for the Process Industries [5]’, probably driven by their impatience of the time being taken by the IEC body in developing an international approved standard. However, at the end of 1998 the seven part IEC 61508 [1] – ‘Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems’ started to be released. This was followed in 2003 by the three-part IEC 61511 [2]. These IEC standards addressed all aspects of functional safety from developing an overall safety instrumented system (SIS) safety requirements specification (SRS), performing a hazard risk analysis to determine the required safety integrity level (SIL), safety instrumented function (SIF) hardware and software design realisation, through to testing and maintenance. In all, these cover 16 Phases collectively known as the Safety Lifecycle Model. This article explores some key aspects of an application software safety requirements specification, which is part of the design realisation Phase 10, together with a typical control room operator SIS graphic display configuration indicating the status and health of the SIS.

Software requirements

IEC 61508 Part 3 [1] (mainly for safety equipment manufactures) and also IEC 61511 Part 1 [2], provide the minimum application software framework requirements and functional guidelines, they also allude to the need for a good operator interface, which is lacking with many current system designs. The embedded software which forms an integral operating part of the programmable electronics, and also ensures safety certification, will not be reviewed. Probably one of the first decisions to be made is which programming language to use for the safety approved programmable electronic system application software. This will normally be a limited variability language (LVL), being textual or graphical conforming to IEC 61131-3 [6] such as Ladder, Boolean including Function Block Diagrams or Sequential Flow Chart also referred to as State Transition Diagrams. The author’s personal preference is Boolean with Function Blocks, which has a high degree of configuration flexibility and is well understood by most control systems practitioners. It is good practice to follow the external failsafe principle ie logic ‘1’ or 24 Vdc being the normal healthy state and Logic ‘0’ or 0 Vdc being the safe tripped state, right through to the functional logic. One of the prime requirements is that the software safety functions together with applicable software systematic capabilities are well specified to enable initial design; these are defined in the Software SRS. The software execution plan should define the strategy for procurement, development (normally by the SIS Logic Solver vendor), integration with other systems, verification, validation and any required modifications following a management of change (MOC) procedure. As with the overall SIS safety lifecycle, a specific software safety lifecycle is also followed, refer to Figure 2.

Verification and overall software validation testing will use the typical V-model, where each step is cross checked to ensure compli-
ance and correctness that the output requirements of the previous step satisfies the input requirements of the next. Software is often developed which consists of both safety and non-safety related functions and these should be segregated within the software structure wherever practical. An example would be the trip outputs to ESD isolation valves being safety related and the associated valve status limit switch feedbacks, which form no safety function per se, but provides the operator with indication of correct trip action. Should an ESD trip valve fail to operate on command, this would raise an Emergency alarm; if it fails with no command say on a trip solenoid coil failure, then a High priority alarm is raised. It should be noted that unless it can be demonstrated that failures of non-safety functions cannot adversely affect the safety related functions, then all software should be treated as safety related.

Figure 2: Realisation Phase – Software Safety Lifecycle.

The SIS objective is to provide the operational task requirements necessary to implement the safety instrumented functions consistent with the SIS architecture and specified SIL. The Logic Solver application software SRS provides the executable software functionality with operating properties, and specifies how the inputs condition the outputs together with associated communications, both internal and external. Where a physically separate SIS logic solver is used with a dedicated basic process control system such as a DCS, one should not underestimate the amount of inter-communications data required and the integration testing time, this can be up to double that of a combined DCS/SIS equipment package. The software SRS as a minimum, needs to cover the following functions as applicable:

- How a safe process state is achieved and maintained
- Safety related communications
- Capacity and response time performance
- Online software modifications
- SIF structure partitioning
- Provide guidance as to application software configuration eg required function block (FB) library needs and FB linking
- Internal Tag conventions and notations
- Trip Reset requirements including interlock Permissives
- Sensor and final element fault handling including field wiring
- Online testing of SIF loop components (mainly sensors and final elements)
- Interfaces with other systems specifically the DCS
- Operator SIS display graphics with interaction requirements
- Invalid or potentially dangerous operator commands
- Sequence of events recording

Software configuration

The application software structure is partitioned into specific Group SIFs using certified or well proven function blocks selected from a library. These are interconnected with maybe a few discrete Boolean

Figure 3: Typical SIF schematic.
logic elements to form the required functional safety application. The connections to other SIS SIF Groups as well as the external communication requirements with the DCS SIS operator’s graphic page are also defined, refer to Figure 3 for a typical schematic.

Implementing the SIF functional logic using standard function blocks as shown is very quick and also aids factory acceptance testing (FAT). The functionality within the function blocks contains various options, should any option not used be used, it will not prevent the correct operation of that block, all that may be required is to apply a dummy logic ‘1’ on that option input. Figure 4 shows a more detailed function block interconnection for a simplified SIF, ie no voting requirements. The following is a brief description of those I/O connections.

The smart sensor 4 – 20 mA input (analogue is always preferred), is connected to the logic solver analogue input module, this then transmits the digital value in engineering units to the trip value comparator. The input module also monitors for a faulty input signal, the sensor if a conventional transmitter, will use internal diagnostics to detect any malfunction and drive the output signal in a pre-determined direction using the NAMUR NE43 standard, which differentiates between a transmitter under/over range and an actual fault, refer to Figure 5.

FB 01 - any input fault raises an alarm and if Fault Tolerance (override) is selected, will initiate an automatic maintenance override switch (MOS) for a limited time eg four hours to enable corrective action. The Group Trip input is used to cancel the Input Fault MOS if in a tripped state. The Process Trip is the safety related function input. A Time Delay can be included which delays the trip function, this is always set to at least one or two seconds if Fault Tolerance is used to allow the sensor to drive to one end, but can be set longer if required such as on low flow trips; if the process safety time allows. The FO (first out) Reset is used to reset the ‘hold’ on a Process Trip. First Out holds any subsequent Process Trip inputs so that the operator can immediately see on the DCS Graphics what initially tripped the SIS Group (within one CPU cycle time).

Figure 4: Function Block connections.
The link. The trip setting value should be displayed on the associated DCS values should be transmitted to the DCS over the communications Software SRS. All SIS analogue input values together with trip setting and functional safety management. The operator SIS display, with as - is being acknowledged as a very important aspect of process control and associated DCS Graphic display design, although recently this In the past, little attention has been given to SIS operator interface or has a fault, as this will also trip the SIF.

When online. Some outputs may require an additional sequential Reset which has been catered for, and any valve fault will flash the respective Tag. A final element Test is available which allows for operating the valve during commissioning or following a Group trip under permit conditions, also any Manual valve partial stroke test can be performed when online.

Operator Interface

In the past, little attention has been given to SIS operator interface and associated DCS Graphic design display, although recently this is being acknowledged as a very important aspect of process control and functional safety management. The operator SIS display, with associated response, needs to be well defined as a subsection of the SIS Software SRS. All SIS analogue input values together with trip setting values should be transmitted to the DCS over the communications link. The trip setting value should be displayed on the associated DCS controller. so if a MOS is applied on a 1oo3 Mid-Value will change to the Average of the remaining two good inputs. They also prevent a MOS being applied to ‘healthy’ inputs if a fault is present on any input as this would often trip the SIF, in addition, they only allow one simultaneous MOS to be applied on voted SIF ‘healthy’ inputs. An additional feature, if selected, prevents a spurious trip (for a short period) and much embarrassment from occurring, when a MOS is initiated and the technician starts to work on the wrong transmitter of a voted input. Another feature which prevents a spurious trip due to technician error is that the MOS will not be allowed to be removed if the trip input is still in a tripped state or has a fault, as this will also trip the SIF.

The DCS interface must under no circumstances influence the safety integrity of the SIS. The operational state of the SIS needs to be monitored for potential problems, also the interface can perform vital system trouble shooting and maintenance activities such as MOS control. A number of display formats exist and a typical SIS interface Graphic example is shown in Figure 7. The basic display page takes the form of a ‘flat’ cause and effect diagram which helps the operator to clearly understand the SIS Group I/O functional relationship. At the top of the page is some general information and common SIS alarms. The display is divided into the different unit ESD or SIS Groups and the Groups contain the applicable SIF information split into the five sections.

The first on the left gives the SIS Group identification, followed by the applicable Permissives, and then we have the functional Trip Inputs, this is then followed by the actual Group Trip and Resets information, and lastly we have the Trip Outputs.

There are a number of ‘Help’ buttons which when selected pop-up a window to assist the operator with additional information. Under the ESD Group description a Help button describes the overall safety function of that SIS Group.

The Permissives are displayed and need to be satisfied to allow a safe SIS Group Reset following a trip, once the Group is reset then the Permissives play no further part. Should any Permissive input be faulty, there is a short Manual Override of 30 seconds (under permit conditions), to allow the SIS Group to be reset, provided all Trip Inputs are healthy.

All the Group SIF Trip Inputs are shown each with a Help button which describes that specific safety function. Should any input be faulty, its respective Tag will start flashing. Clicking on an input Tag will pop up the respective process value faceplate, a good feature when performing loop testing and maintenance. Any trip time delay of five seconds or more will be indicated.

Each SIF will have its own MOS for online periodic input testing and maintenance, it will flash on an Automatic MOS initiation from a faulty input (if Fault Tolerance is selected), and be permanently lit on a Manual MOS initiation. If the SIF has a process start-up OOS, this will also be displayed when in operation.

The Group Trip and Reset section displays the overall SIS Group state, first there is a ‘soft’ Manual ESD (double click confirmation action), then there is a warning if the Group is about to trip following a time-out function, an actual Group Tripped indication is displayed, followed by a First-Out Reset command, indication when a Local Reset has been initiated (if applicable) and finally the Group Reset which will flash when ready to reset. The final section covers the trip outputs including the tripping of associated SIS Groups and any DCS controller FMC actions via the Help button, and the final element status.

Some outputs may require an additional sequential Reset which has been catered for, and any valve fault will flash the respective Tag. A final element Test is available which allows for operating the valve during commissioning or following a Group trip under permit conditions, also any Manual valve partial stroke test can be performed when online.

The limitation to this article prevents a description of the more interesting voting blocks and others such as 2oo3 Mid-Value to DCS for control purposes, but it is suffice to say that all voting function blocks automatically change their voting logic to the next safest state, eg 2oo3 to 1oo2, following a MOS or faulty sensor input on one leg. A MOS or fault on a 2oo3 Mid-Value will change to the Average of the remaining two good inputs. They also prevent a MOS being applied to ‘healthy’ inputs if a fault is present on any input as this would often trip the SIF, in addition, they only allow one simultaneous MOS to be applied on voted SIF ‘healthy’ inputs. An additional feature, if selected, prevents a spurious trip (for a short period) and much embarrassment from occurring, when a MOS is initiated and the technician starts to work on the wrong transmitter of a voted input. Another feature which prevents a spurious trip due to technician error is that the MOS will not be allowed to be removed if the trip input is still in a tripped state or has a fault, as this will also trip the SIF.

Figure 5: NAMUR NE43.
Conclusion

It is hoped that this article, although only being a basic introduction to SIS logic solver functionality, has provided the reader with a useful insight in developing a practical Software SRS.

References


Dick Perry has been in the Measurement and Control profession for 50 years. He started his career as an apprentice instrument technician in the chemical industry in the UK. He came to South Africa in 1970. He was with Fluor SA for over 20 years and held the position of control systems design director. He retired in 2012. He currently consults to a number of companies specialising in safety instrumented systems, hazardous area instrumentation and burner management systems. He is a Fellow of the SAIMC and a past president. He is registered with ECSA as a Professional Technologist (Pr Tech Eng). Enquiries: Email rjperry@nashuisp.co.za.

Figure 7: Typical DCS SIS display.