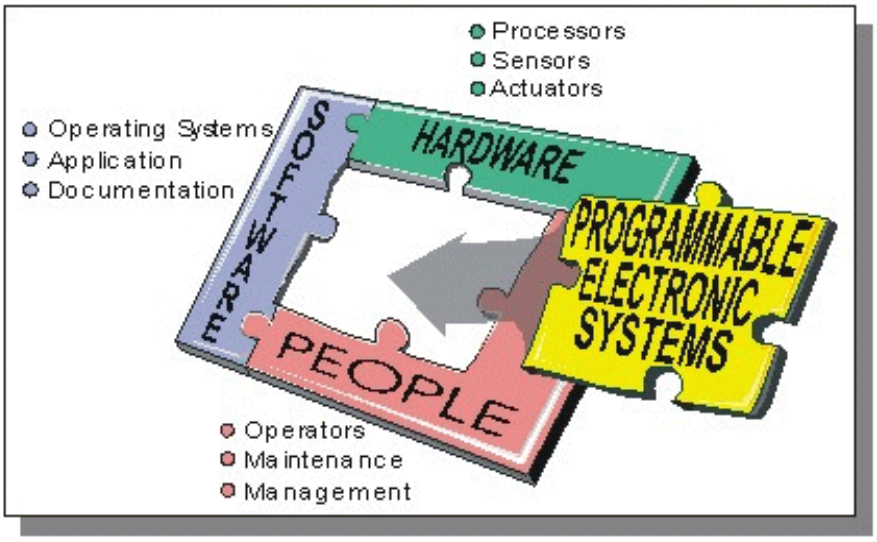
SYSTEM SAFETY EVALUATION PROGRAM

Programmable Electronic Mining Systems: Best Practice Recommendations (In Nine Parts)



Part 2: 2.1 System Safety

Mine Safety and Health Administration
Approval and Certification Center
Electrical Safety Division
Triadelphia, West Virginia

Department of Health and Human Services
Centers for Disease Control and Prevention
National Institute for Occupational Safety and Health
Pittsburgh Research Laboratory
Pittsburgh, Pennsylvania

May, 2001

CONTENTS

ILLUSTRATIONS

TABLES

PROGRAMMABLE ELECTRONIC MINING SYSTEMS: BEST PRACTICE RECOMMENDATIONS (In Nine Parts)

Part 2: 2.1. System Safety

By John J. Sammarco[1] and Thomas J. Fisher[2]

ABSTRACT

This report (System Safety 2.1) is the second in a nine-part series of recommendations addressing the functional safety of processor-controlled mining equipment. It is part of a risk-based system safety process encompassing hardware, software, humans, and the operating environment for the equipment's life cycle. Figure 1 shows a safety framework containing these recommendations. The reports in this series address the various life cycle stages of inception, design, approval and certification, commissioning, operation, maintenance, and decommissioning. These recommendations were developed as a joint project between the National Institute for Occupational Safety and Health and the Mine Safety and Health Administration. They are intended for use by mining companies, original equipment manufacturers, and aftermarket suppliers to these mining companies. Users of these reports are expected to consider the set in total during the design cycle.

• 1.0 *Safety Introduction*.—This is an introductory report for the general mining industry. It provides basic system/software safety concepts, discusses the need for mining to address the functional safety of programmable electronics, and includes the benefits of implementing a system/ software safety program.

• 2.1 *System Safety* and 2.2 *Software Safety*.—These reports draw heavily from International Electrotechnical Commission (IEC) standard 61508 [IEC 1998a,b,c,d,e,f,g] and other recognized standards. The scope is "surface and underground safety mining systems employing embedded, networked, and nonnetworked programmable electronics." System safety seeks to design safety into all phases of the entire system. Software is a subsystem; thus, software safety is a part of the system's safety.

• 3.0 *Safety File*.—This report contains the documentation that demonstrates the level of safety built into the system and identifies limitations for the system's use and operation. In essence, it is a "proof of safety" that the system and its operation meet the appropriate level of safety for the intended application. It starts from the beginning of the design, is maintained during the full life cycle of the system, and provides administrative support for the safety program of the full system.

---

[1]Electrical engineer.
[2]Senior research physical scientist.
Pittsburgh Research Laboratory, National Institute for Occupational Safety and Health, Pittsburgh, PA.

• 4.0 *Safety Assessment*.—The independent assessment of the Safety File is addressed. It establishes consistent methods to determine the completeness and suitability of safety evidence and justifications. This assessment could be done by an independent third party.

• 5.0 *Safety Framework Guidance.*—It is intended to supplement the safety framework reports with guidance that provides users with additional information. The purpose is to help users in applying the concepts presented. In other words, the safety framework is *what needs to be done* and the guidance is *how it can be done*. The guidance information reinforces the concepts, describes various methodologies that can be used, and gives examples and references. It also gives information on the benefits and drawbacks of various methodologies. The guidance reports are not intended to promote a single methodology or to be an exhaustive treaty of the subject material. They provide information and references so that the user can more intelligently choose and implement the appropriate methodologies given the user's application and capabilities.

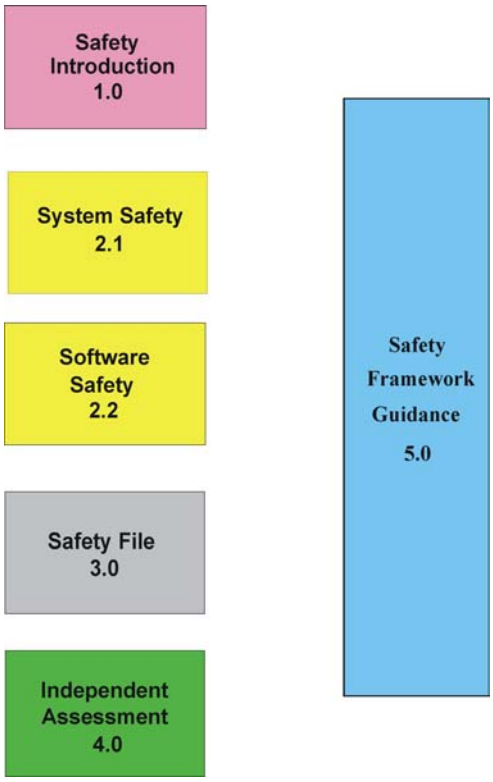


Figure 1.—The safety framework and associated guidance.

ACKNOWLEDGMENTS

The authors thank David C. Chirdon, Gerald D. Dransite, and Chad Huntley with the Mine Safety and Health Administration's (MSHA) Approval and Certification Center, Triadelphia, WV, for their assistance in developing this series of reports.

BACKGROUND

The mining industry is using programmable electronic (PE) technology to improve safety, increase productivity, and improve mining's competitive position. It is an emerging technology for mining that is growing in diverse areas, including longwall mining systems, automated haulage, mine monitoring systems, and mine processing equipment. Although PE provides many benefits, it adds a level of complexity that, if not properly considered, may adversely affect worker safety [Sammarco et al. 1997]. This emerging technology can create new hazards or worsen existing ones. PE technology has unique failure modes that are different from mechanical systems or hard-wired electronic systems traditionally used in mining. PE includes microprocessors, embedded controllers, programmable logic controllers (PLCs), and associated software.

The use of a safety life cycle helps to ensure that safety is applied in a systematic manner for all phases of the system, thus reducing the potential for systematic errors. It enables safety to be "designed in" *early* rather than being addressed after the system's design is completed. Early identification of hazards makes it easier and less costly to address them. The life cycle concept is applied during the entire life of the system because hazards can become evident at later stages or new hazards can be introduced by system modifications. The safety life cycle for mining is an adaptation of the safety life cycle in part 1 of IEC 61508 [IEC 1998a].

System safety activities include identifying hazards, analyzing the risks, designing to eliminate or reduce hazards, and using this approach over the entire system life cycle. These system safety activities start at the system level and flow down to the subsystems and components. More detailed information on the fundamentals of system safety is presented by Sammarco et al. [1999].

This report incorporates some of the "best practices" for safety in the world and some of the latest international thinking on safety for PE. It uses a key group of standards selected from approximately 200 safety standards pertaining to PE. These key standards are listed in table 1.

Existing safety standards are built on collections of expertise and experiences (lessons learned) involving fatalities, injuries, and near misses of systems using PE. In general, standards also provide uniform, systematic approaches. History has shown standards to be an effective tool for safety [Leveson 1992]. Thus, by adapting existing standards, mining can build upon the valuable information captured in these standards documents.

1.0 Introduction

1.1 Document Conventions

This report follows a general format where major sections consist of an objective and associated recommendations. The formats are shown below:

Objective(s):
Recommendation(s):
NOTE(S):

The **NOTES** give brief clarification, reasoning, or guidance. More in-depth information is found in supplemental guidance documents.

1.2 Scope

1.2.1 Surface and underground mining systems using PE for control or monitoring of safety-critical mining systems and functions are within the scope. It is not intended to apply to handheld instruments; however, many of these principles would be useful in assessing this equipment.

1.2.2 Systems, protection layers, and devices using PE that are associated with the system are within the scope. These include—

- Mining control and monitoring systems (MCMS) using PE
- Safety instrumented systems (SIS)
- Critical alarms

1.3 General

1.3.1 These recommendations do not supersede Federal or State laws and regulations.

1.3.2 These recommendations are not equipment- or application-specific.

1.3.3 These recommendations do not serve as a compliance document.

1.3.4 These recommendations apply to the entire life cycle of the mining system.

1.3.5 These recommendations apply mainly to the safety parts of the system. However, many of the recommendations can also be applied to the basic system.

1.4 Purpose

1.4.1 To present the minimum set of processes, techniques, and methods for a safety life cycle.

1.4.2 To provide uniform recommendations for the entire safety life cycle.

1.4.3 To provide a disciplined approach to identifying hazards.

1.4.4 To aid in management and technical aspects of identified hazards and eliminating them when practical or reducing them to an acceptable level of risk.

1.4.5 To help mining equipment manufacturers, evaluators, and mine operators.

1.4.6 To enable the generation of safety plans based on the safety life cycle. These include plans for system safety, software safety, operation and maintenance, safety validation, and management of change.

1.5 Applicability

1.5.1 These recommendations *do not* apply to low-complexity systems satisfying these criteria:

(1) The failure modes of the system, all subsystems, and components are well defined and understood.
(2) The system's behavior under all fault conditions can be completely understood.

NOTE 1: *Example:* A system comprising one or more limit switches that operates, possibly via interposing electromechanical relays, one or more contactors to deenergize an electric motor is a low-complexity safety system.

1.5.2 These recommendations apply to systems, protection layers, and devices using PE that are associated with the system as stated within the scope. This includes both hardware and software.

NOTE 2: Examples of MCMSs using PE include mine hoists, elevators, longwall mining systems, mine monitoring systems, and remote-controlled equipment. A basic diagram for an MCMS is shown in figure 2.

NOTE 3: Examples of PE include programmable logic controllers (PLCs), embedded controllers, firmware, programmable gate arrays (PGAs), application-specific integrated circuits (ASICs), custom or commercial single-board computers (SBCs), or any other configurable electronics.

1.6 Responsibilities

1.6.1 It is the manufacturer's responsibility to create the safety plan with input from the end users and other parties, where appropriate.

1.6.2 These recommendations are to be implemented by manufacturers and subcontractors involved in developing systems, subsystems, or software.

1.6.3 Any party involved in any phase of the life cycle for a safety system is responsible for communicating to the appropriate responsible party safety plan discrepancies. This includes the manufacturer and personnel internal to the organization, subcontractors, and end users. The manufacturer is responsible for documenting reported discrepancies and results.

1.7 General Approach

1.7.1 The general approach is to establish a safety life cycle (see figure 3).

1.7.2 The general approach *within* the safety life cycle is to use system safety, risk-based processes and methods to—

(1) Identify hazards and associated risks
(2) Mitigate these risks to acceptable levels of safety

NOTE 4: Safety plans will depend on the given application, organization, and other factors, including the—
- Organization's management structure
- Organization's technical processes, skills, and resources
- Size of system
- Previous experience for the system and application
- Nature of the hazards
- Consequences in the event of failure
- Degree of complexity
- Degree of design novelty
- Safety integrity level

2.0 Key Documents

2.1 This recommendation document is based on information and concepts from the documents listed in table 1.

Table 1.—Key documents used for these recommendations

Standard identification	Title
IEC 61508 parts 1-7	Functional safety of electrical/electronic/programmable electronic safety systems [IEC 1998a,b,c,d,e,f,g].
ANSI/ISA S84.01	Application of safety instrumented systems for the process industries [ANSI/ISA 1996].
ISA Draft Technical Report and TR84.0.02, Parts 1-5	Safety instrumented systems (SIS) - safety integrity level (SIL) evaluation techniques [ISA 1998].
MIL-STD-882C	Standard practice for systems safety program requirements [U.S. Department of Defense 1993].
UK Def Stan 00-58	HAZOP studies on systems containing programmable electronics [Ministry of Defence 1998].
UL 1998	Software in programmable components [Underwriter Laboratories, Inc. 1998].

3.0 Definitions

The definitions are directly from IEC 61508, part 4 [IEC 1998d] and ISA S84.01 [ANSI/ISA 1996]. A few definitions are adaptations or newly formed definitions specific to mining.

Error - A discrepancy between a computed, observed, or measured value or condition and the true, specified, or theoretically correct value or condition.

Failure - The termination of the ability of a functional unit to perform a required function.

Fault - An abnormal condition or state that may cause a reduction in, or loss of, the capability of a functional unit to perform a required function.

NOTE 5: A "failure" is an event; a "fault" is a state.

NOTE 6: Faults are random or systematic.

Field Devices - Peripheral devices hard-wired to the input/output terminals of a logic system. Field devices include sensors, transmitters, operator interface devices (i.e., displays, control panels, pendant controllers), actuators, wiring, and connectors.

Hazard - Environmental or physical condition that can cause injury to people, property, or the environment.

Human-Machine Interface - The physical controls, input devices, information displays, or other media through which a human operator interacts with a machine for the purpose of operating the machine.

Mean Time to Failure (MTTF) - The expected time that a system will operate before the first failure occurs.

Mining Control and/or Monitoring System (MCMS) - A system, using programmable electronics (PE), that responds to input signals from the equipment under control and/or from an operator and generates output signals, causing the equipment under control to operate in the desired manner (see figure 2).

Mishap - An unplanned event or series of events resulting in death, injury, occupational illness, or damage to or loss of equipment or property, or damage to the environment. In the real world, complete freedom from adverse events is not possible. Therefore, the goal is to attain an acceptable level of safety.

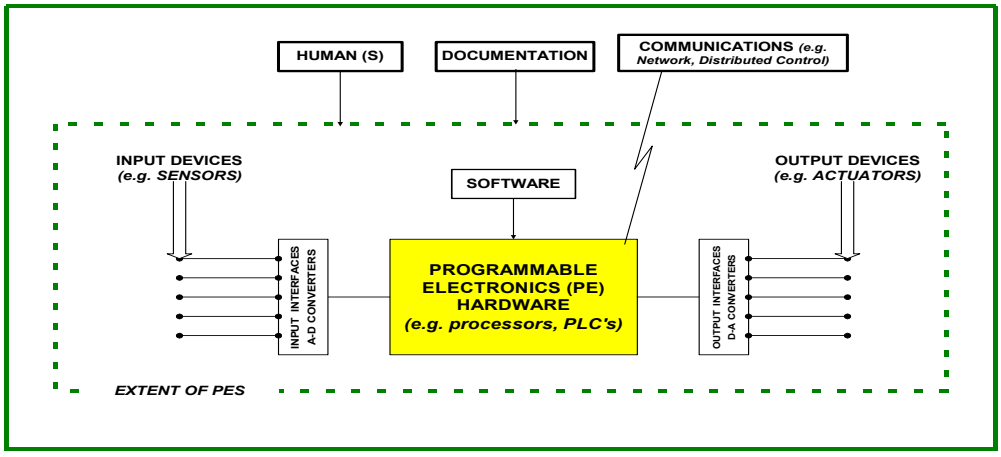


Figure 2.—Boundaries of an MCMS using programmable electronics.

Probability of Failure on Demand (PFD) - A value that indicates the probability of a system failing to respond on demand. The average probability of a system failing to respond to a demand in a specified time interval is referred to "PFD avg."

Programmable Electronics (PE) - Refers to electronically programmable or configurable devices (e.g., embedded controller, programmable logic controller, single-loop digital controller, distributed control system controller) that are effectively the "brain" of a PE system.

Programmable Electronic System (PES) - Any system used to control, monitor, or protect machinery, equipment, or a facility that has one or more programmable electronics (PE), including all elements of the system such as power supplies, sensors and other input devices, data highways and other communications paths, and actuators and other output devices.

Random Hardware Failure - A failure, occurring at a random time, that results from one or more of the possible degradation mechanisms in the hardware.

NOTE 7: There are many degradation mechanisms occurring at different rates in different components. Since manufacturing tolerances cause components to fail due to these mechanisms after different times in operation, failures of equipment comprising many components occur at predictable rates, but at unpredictable (i.e., random) times.

NOTE 8: A major distinguishing feature between random hardware failures and systematic failures is that system failure rates (or other appropriate measures) arising from random hardware failures can be predicted with reasonable accuracy, but systematic failures, by their very nature, cannot be accurately predicted. That is, system failure rates arising from random hardware failures can be quantified with reasonable accuracy, but those arising from systematic failures cannot be accurately statistically quantified because the events leading to them cannot be easily predicted.

Risk - The combination of the probability of occurrence of harm and severity of that harm.

Risk Reduction Factor (RRF) - A measure of lowering the probability of an event from happening. RRF ' inherent risk/acceptable risk, or RRF ' 1/PFD.

Safety - Freedom from unacceptable risk.

Safety Availability - Fraction of time that a safety system is able to perform its designated safety service when the process is operating (PFD ' 1 & safety availability).

Safety Function - A function implemented by singular or multiple MCMSs, protection layers, and devices using PE intended to achieve or maintain a safe state for a specific hazardous event.

Safety Instrumented System (SIS) - System composed of sensors, logic solvers, and final control elements for the purpose of taking the mining system to a safe state when predetermined conditions are violated. Other terms commonly used include "emergency shutdown system," "safety shutdown system," and "safety interlock system."

Table 2.—Assignment of SIL values for low-demand modes of operation

Safety integrity level (SIL)	Probability of failure on demand average range (PFD avg.)	Risk reduction factor (RRF)	Qualitative methods
1	$10^{\&1}$ to $10^{\&2}$	10- 100	Method-dependent.
2	$10^{\&2}$ to $10^{\&3}$	100- 1,000	Method-dependent.
3	$10^{\&3}$ to $10^{\&4}$	1,000-10,000	Method-dependent.

Table 3.—Assignment of SIL values for high (continuous) demand modes of operation

Safety integrity level (SIL)	Probability of failure on demand average range (PFD avg.)	Risk reduction factor (RRF)	Qualitative methods
1	$10^{\&5}$ to $10^{\&6}$	100,000- 1,000,000	Method-dependent.
2	$10^{\&6}$ to $10^{\&7}$	1,000,000- 10,000,000	Method-dependent.
3	$10^{\&7}$ to $10^{\&8}$	10,000,000-100,000,000	Method-dependent.

Safety Integrity Level (SIL) - One of three possible discrete integrity levels (SIL 1, SIL 2, SIL 3) of safety instrumented functions. SILs are defined in terms of quantitative or qualitative methods. SIL 3 has the highest level of safety integrity (see tables 2 and 3).

NOTE 9: SILs apply to safety functions of systems, protection layers, and devices using PE.

Safety Life Cycle - The necessary activities involved in the implementation of safety-critical systems. The activities begin at the concept stage and cease after the systems' decommissioning.

Systematic Failure - A failure related 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.

NOTE 10: Corrective maintenance without modification will usually not eliminate the failure cause.

NOTE 11: A systematic failure can be induced by simulating the failure cause.

NOTE 12: Example causes of systematic failures include human error in the—
- Safety requirements specification.
- Design, manufacture, installation, operation of the hardware
- Design, implementation, etc., of the software

Validation - The activity of demonstrating that the safety system under consideration, before or after installation, meets in all respects the safety requirements specification for that safety system.

Verification - The activity of demonstrating for each phase of the relevant safety life cycle by analysis and/or tests that, for the specific inputs, the deliverables meet in all respects the objectives and requirements set for the specific phase.

4.0 Management of Functional Safety

Objective:

4.1 To define management activities and processes throughout the system life cycle necessary to ensure and maintain functional and operational safety.

Recommendations:

4.2 Management of functional safety is conducted *during all stages* of the safety life cycle.

4.3 Management of functional safety is integrated with the safety life cycle and the system development life cycle.

4.4 Management responsibilities and activities during the safety life cycle should include the following:

- Setting the policy and strategy for functional safety
- Demonstrating commitment and support for safety
- Planning, organizing, controlling, leading, and communicating functional safety processes
- Ensuring competency of people for functional safety activities
- Establishing, documenting, and implementing a safety life cycle
- Establishing, documenting, and implementing safety plans within the framework of the safety life cycle (plans for system safety, software safety, operation and maintenance, safety validation, and management of change)
- Establishing review, approval, and authorization responsibilities and processes
- Defining responsibilities for people and organizational units
- Ensuring that sufficient and accurate safety documentation is created and maintained
- Ensuring sufficient validation and verification activities and processes are in place
- Ensuring that accident data from MSHA and other sources are reviewed to avoid repeat occurrences
- Monitoring and reviewing safety outcomes

NOTE 13: In many cases, a relationship exists between a list of management responsibilities and activities and established quality procedures. For example, if the manufacturer is ISO 9000-qualified or has implemented basic corporate level quality procedures, then many of the procedures/processes listed above for the safety plan can reference the appropriate quality procedure(s) or use the framework of a quality procedure as the basis for a project-specific procedure/process.

4.5 The competency of persons, including subcontractors, involved in critical safety life cycle activities and management activities should be appropriate.

NOTE 14: The following factors should be considered when assessing the competence of persons:

- Engineering knowledge, training, and experience appropriate to the mining application
- Engineering knowledge, training, and experience appropriate to the technology (e.g., electrical, electronic, programmable electronic, software engineering)
- Safety engineering knowledge, training, and experience
- Knowledge of the mining legal and safety regulatory framework
- Management knowledge, training, and experience appropriate to the mining application

- The safety integrity levels of the PE safety systems: the higher the safety integrity levels, the more rigorous the specification and assessment of competence should be.
- The novelty and complexity of the technology and application

NOTE 15: Safety is best achieved by a combination of managerial and technical processes and activities.

5.0 The Safety Life Cycle Overview

Objective:

5.1 A general overview of the overall safety life cycle is presented to establish, at a conceptual level, the general life cycle phases and objectives. Figure 3 shows the overall safety life cycle. Table 4 lists the objectives of each phase. Some life cycle phases are simplified and do not show all of the details within the phase. However, these phases are expanded in subsequent sections of this report.

Recommendations:

5.2 The safety life cycle applies to each safety system, protection layer, and device used by mining systems or equipment.

NOTE 16: The use of a safety life cycle is required to ensure that safety is applied in a systematic manner, thus reducing the potential for systematic errors. It enables safety to be "designed in" early rather than being addressed after the system's design is completed. Early identification of hazards makes it easier and less costly to address them. The safety life cycle concept is applied during the entire life of the system since hazards can become evident at later stages or new hazards can be introduced by system modifications.

NOTE 17: When development and implementation phases of the life cycle are expanded, there should be clear paths back to previous steps in the cycle. For example, in the design phases there must be a path back to the specifications phase to modify or further define the specifications resulting from design activity or to even regress back to the hazard/risk analysis phase if design activities resulted in the identification of a previously unconsidered hazard or risk. The safety life cycle is not a once-through process, but a looping process that may have many iterations through some phases before progressing to the next phase.

Activities in one phase may be done concurrently with other phases, as in the case of verification and validation. Development of verification and validation plans and procedures should be early in the design cycle to be of most benefit in ensuring that the resultant design meets the specifications. This is an iterative process where verification is used to check the outputs from various design phases and validation is used to check the completed system.

NOTE 18: Traceability is an important feature of the safety file. If the life cycle model is followed and the safety file is appropriately populated with deliverables from each phase, one will be able to select a hazard and trace from the hazard/risk analyses, through specifications and safety function allocation, to design, verification, and implementation, and finally, see at validation that the selected hazard was addressed, designed for, and resolved in an acceptable manner. When the concept of traceability is applied throughout a project, the cohesiveness of the safety file is greatly improved.

5.3 The safety life cycle must be integrated with the overall product development life cycle. This integration is the responsibility of the manufacturer.

NOTE 19: The integration is required because safety issues impact overall development issues and vice versa. Secondly, an integrated approach minimizes the likelihood of addressing safety as an afterthought of the system design. An example merging of the safety life cycle and a development life cycle is shown in figure 4.

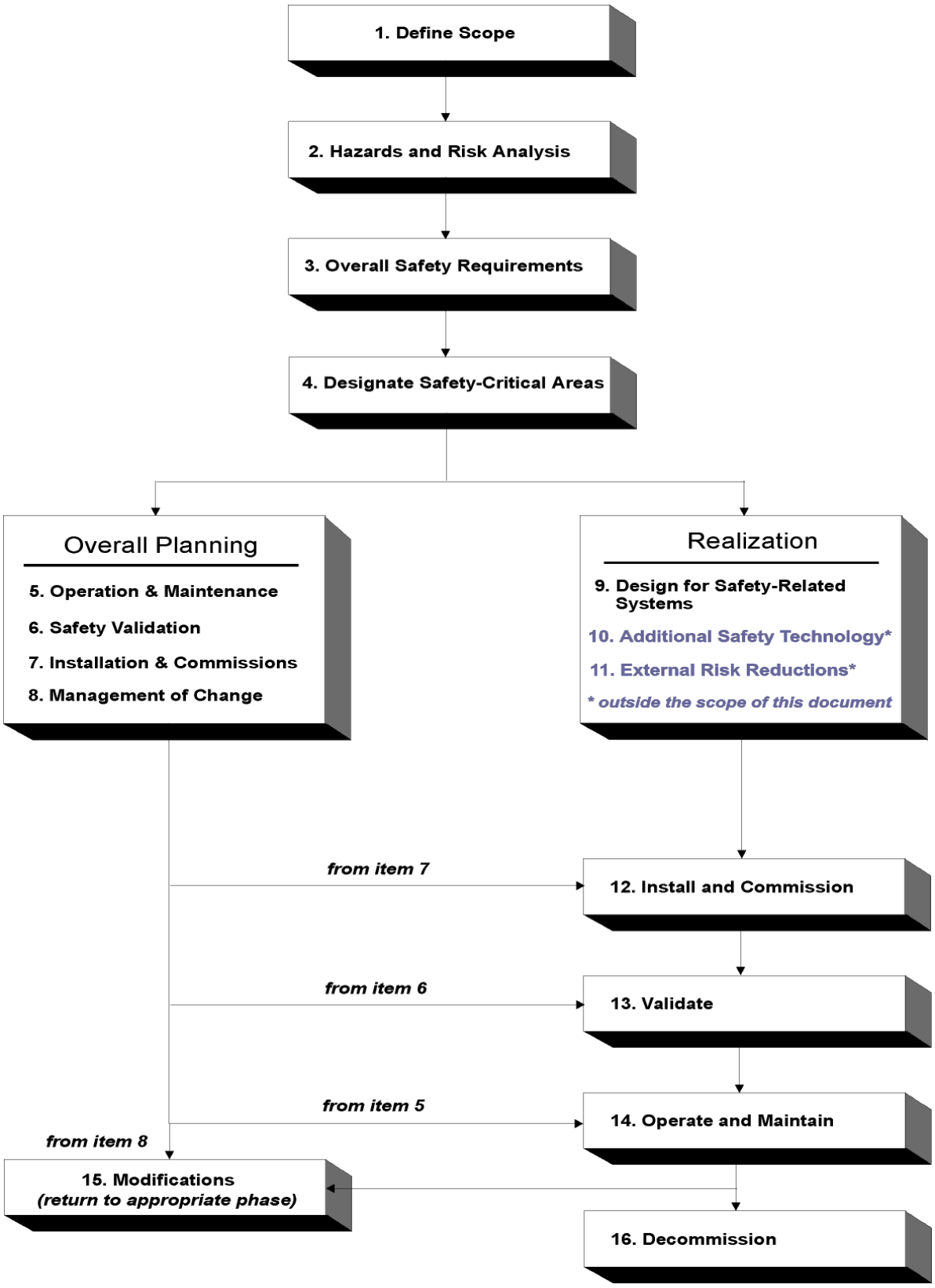


Figure 3.—The safety life cycle.

Table 4.—Overall safety life cycle overview
(adapted from IEC 61508, part 1, Overall Safety Life Cycle)

Life cycle phase	Objectives
1. Define scope	To determine the boundaries for the PE system and to bound the hazard and risk analysis.
2. Hazards and risk analysis	To identify and analyze hazards, event sequences leading to hazards, and the risk of hazardous events.
3. Overall safety requirements	To specify the safety functions and associated safety integrity for the safety system(s).
4. Designate safety-critical areas	To assign safety functions to various PE-based and non-PE-based safety systems and protection layers. To assign safety integrity levels (SILs).
5. Operation and maintenance plan	To plan how to operate, maintain, and repair the PE-based safety system to ensure functional safety.
6. Safety validation plan	To plan how to validate that the PE-based safety system meets the safety requirements.
7. Installation and commissioning plan	To plan how to install and commission the PE-based safety system in a safe manner and to ensure that functional safety is achieved.
8. Management of change plan	To plan how to ensure that changes will not adversely impact functional safety. To plan how to systematically make and track changes.
9. Design for safety systems	To design and create the PE-based safety system. To follow safety practices for the PE-based safety system and the basic system design.
10. Additional safety technology	As needed; not within the scope of this report.
11. External risk reduction	As needed; not within the scope of this report.
12. Install and commission	To install and commission the safety system properly and safely.
13. Validate	To carry out the safety validation plan.
14. Operate and maintain	To operate, maintain, and repair the PE-based safety system so that functional safety is maintained.
15. Modifications	To make all modifications in accordance to the management of change plan.
16. Decommission	To ensure the appropriate functional safety during and after decommissioning.

6.0 Safety Life Cycle Phases

NOTE 20: The specific safety life cycle activities and degree of detail to meet the object(s) of a life cycle phase vary for each system and the specific circumstances. The objective(s), however, of each life cycle phase remain intact. Broadly defined recommendations are given for each life cycle phase. These recommendations are to be shaped by the user based on specific factors, including—

- Project size
- Previous experience with a similar design
- Degree of complexity
- User resources

6.1 Define Scope

Objectives:

6.1.1 To determine the boundaries for the MCMS and to bound the hazard and risk analysis.

6.1.2 To conduct a preliminary identification of safety items within the MCMS scope.

Recommendations:

6.1.3 The scope of the MCMS must be defined and documented. An analysis of tasks and activities should be performed.

6.1.4 The MCMS subsystems must be identified and documented.

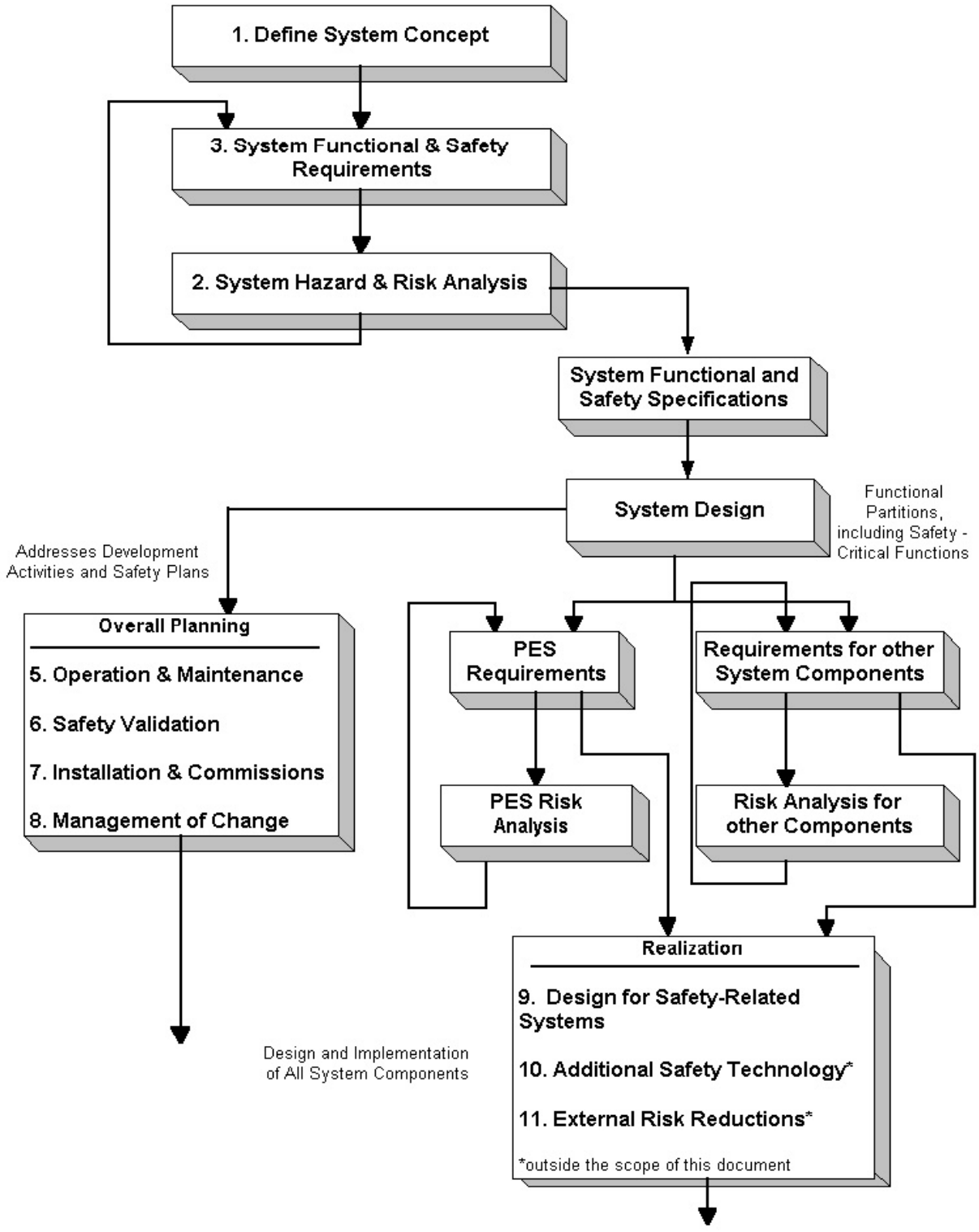


Figure 4.—A merged life cycle diagram.

6.1.5 The following systems and subsystems are considered to be safety systems:

- Those that monitor the state of the mine or another system for safety purposes
- Those that control, regulate, or contain potentially dangerous energy sources
- Those that control or partially control moving equipment, moving parts of equipment, or moving material
- Those that collect, compute, store, display, or manipulate data that are safety-critical

NOTE 21: A clear definition of the scope is required so that hazards are not omitted from safety analyses. The scope defines what is to be assessed for safety and is also necessary to identify important information, including characteristics and limitations of the system.

NOTE 22: Figure 2 can be used to define the scope of an MCMS.

6.2 Hazards and Risk Analysis

Objectives:

6.2.1 To identify and analyze hazards and event sequences leading to hazards during operation, maintenance, fault conditions, foreseeable misuse, and foreseeable human mistakes.

6.2.2 To determine the risk of each hazard.

6.2.3 To determine safety-instrumented functions for risk elimination and reduction.

Recommendations:

6.2.4 More than one type of safety analysis should be used to identify and analyze hazards and hazardous event sequences.

NOTE 23: The use of multiple hazard and analysis techniques is recommended since each has its own purpose, strengths, and weaknesses. Typically, each technique only addresses certain aspects of safety. Thus, one technique alone is not sufficient to effectively identify and analyze all hazards of a system. For example, fault-tree analysis is not well suited for time-based events such as data arrival rates and clock frequencies.

6.2.5 The safety analysis techniques of table 5 should be considered.

NOTE 24: Many safety analysis techniques exist. The *System Safety Society Handbook* [Stephans and Talso 1997] has consolidated these to a compendium of more than 100 safety techniques and methodologies. Table 5 lists a set of practical safety analysis techniques. These are some of the most commonly used by system safety practitioners and are some of the most often cited techniques in the literature pertaining to systems with PE. Secondly, these techniques are not complex. They require less extensive experience and training compared to other techniques. Table 5 gives guidance and does not mandate the techniques listed. The selection of techniques depends on the application and the expertise of the user.

NOTE 25: More information concerning these techniques is presented in the guidance information.

6.2.6 Hazard and risk analysis should begin early and continue to be updated during *all* safety life cycle phases.

6.2.7 The methods for hazard and risk analysis should be documented.

Table 5.—Key safety analysis techniques

Safety analysis	Technique(s)
Hazard identification and analysis	Preliminary hazard analysis Hazard and operability studies (HAZOP) [Ministry of Defence 1998] Checklists
Causal analysis	Fault-tree analysis [Vesely et al. 1981]
Consequence analysis	Failure modes effects analysis [U.S. Department of Defense 1980] Event-tree analysis [Suokas and Rouhianinen 1992]
Human error analysis	Human reliability analysis Job safety analysis (JSA)
Miscellaneous	Checklists

6.2.8 Determination of risk can be done by methods that are qualitative, quantitative, or both.

6.2.9 A method or procedure should be developed to document and track hazards and their status. The following information should be documented (adapted from MIL-STD-882C, Task 106) [U.S. Department of Defense 1993]:

- A description of each hazard and the associated risk.
- Status of each hazard. Hazard status can be defined as:
  Open: A known or suspected hazard without corrective action.
  Monitor: A known or suspected hazard for which corrective action, or study of corrective action, is identified. If the process of implementation is in question, this status may be maintained while the fix is in work and not completed.
  Closed: A known or suspected hazard for which corrective action is identified, initiated, completed, and accepted. Completion is defined as redesign, test, or procedural change and may be closed with the necessary documentation still incomplete.
- The recommended controls to eliminate or reduce the hazard to an acceptable level of risk.
- Approvals accepting the risk and thus effecting a "closed" status.
- Source(s) of hazard identification. Examples include subsystem and system analysis, management of change (MOC) analysis, and accident reports.

NOTE 26: A hazard log could be used to track hazards, their associated tasks and activities, and their resolution. The hazard log could be implemented with computer database or paper document form. The hazard log can be useful for other projects by identifying common hazards and their resolutions.

6.2.10 Mine accident data should be reviewed to identify hazards and risks that could apply to the system.

NOTE 27: The MSHA Web site (www.msha.gov) contains this information.

6.3 Overall Safety Requirements (figure 5)

Objective:

6.3.1 To specify the overall safety requirements in terms of safety functions and associated safety integrity for the safety system(s).

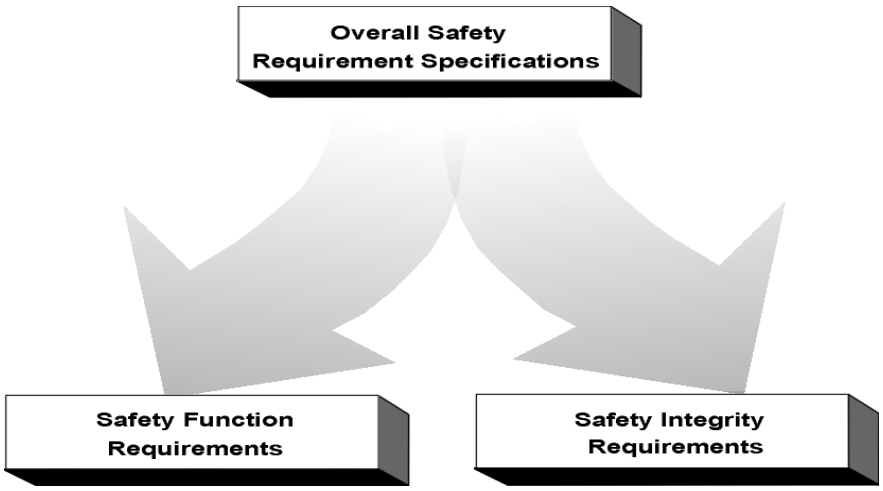


Figure 5.—Overall safety requirement.

Recommendations:

6.3.2 The overall safety requirements should include all operating and maintenance conditions of the system.

6.3.3 The hazard and risk analysis results should be the primary source of input for deriving the overall safety specifications.

6.3.4 Safety functions and integrity requirements should be specified for each identified hazard.

NOTE 28: The requirements do not identify specific technologies for implementation at this point. This is done at the next phase (safety requirements allocation).

6.3.5 The safety function specification should describe *what* the safety system does. The safety integrity requirements should describe the *performance and constraints* (i.e., in terms of capabilities, speed, accuracies, and probabilities).

6.3.6 The "characteristics of safety functions" [BSI 1997] should be considered when formulating the safety function specifications.

6.3.7 The overall safety requirements specification should define the functional safety and safety integrity requirements that address the risks identified by the hazard and risk analysis. Table 6 lists the recommended information for the specification.

NOTE 29: An example overall safety requirements specification is given in appendix D of S84.01 [ANSI/ISA 1996].

Table 6.—Major information components for the safety requirements specification

Safety function requirements	Safety integrity requirements
• List of safety functions • Mapping of each hazard to a safety function • Clear description of each safety function • Default state description of each safety function • Define safe states for all safety functions • Define events to trigger safety functions • Define inputs/outputs of SIS • Define interfaces • Human-machine interface requirements	• SIL of each safety function • Diagnostic requirements • Testing/maintenance requirements • Reliability requirements • Constraints and performance (i.e., range, rate, response time, and trip points)

NOTE 30: The following are examples of safety functions:
- Functions providing emergency stops
- Functions providing manual intervention
- Functions providing manual reset
- Functions providing mode transitions
- Functions monitoring a safe state
- Functions controlling or regulating energy sources
- Functions displaying safety-critical information (e.g., diagnostic displays, warning lights and alarms)
- Functions controlling machine movement
- Functions controlling system startup

6.4 Safety Requirements Allocation

Objectives:

6.4.1 To allocate the safety functions to various PE-based and non-PE-based safety systems and protection layers.

6.4.2 To assign safety integrity levels (SILs).

Recommendations:

6.4.3 An SIL for each safety function should be assigned based on tables 2 and 3 depending on the demand.

6.4.4 If a safety function is allocated to multiple safety systems and protection layers, then the quantitative and qualitative determination of SIL must take this into account.

6.4.5 If the basic MCMS is also used with a safety system (i.e., SIS), then the basic MCMS must be designed to the level of rigor required for the highest SIL of its safety function(s).

6.4.6 If the safety system (i.e., SIS) is separate from the MCMS and assigned multiple safety functions, then it must be designed to the level of rigor required for the highest SIL of its safety functions.

NOTE 31: *Example:* Five safety functions are assigned to one system or protection layer. One safety function is at an SIL 3; the other four are at an SIL 1. The system must be designed to the level of rigor for an SIL 3.

6.4.7 The allocation of safety functions to multiple safety systems and protection layers should take into account common cause failure modes.

NOTE 32: Greater levels of independence can reduce common cause failures. Independence is evidenced by functional diversity, diverse technologies, physical separation, and no sharing of common parts.

6.5 Overall Planning

Objective:

6.5.1 To define activities and responsibilities of departments and people who ensure functional safety.

Recommendations:

6.5.2 Overall planning should be updated as the life cycle activities and design knowledge progresses.

NOTE 33: Overall planning information for operation and maintenance, safety validation, installation and commissioning, and management of change may be incorporated in other documents (i.e., company procedures, quality plans), exist as separate plans, or exist all in one plan.

NOTE 34: Traceability is an important feature of the safety file. If the life cycle model is followed and the safety file is appropriately populated with deliverables from each phase, one will be able to select a hazard and trace from the hazard/risk analyses, through specifications and safety function allocation, to design, verification, and implementation, and finally, see at validation that the selected hazard was addressed, designed for, and resolved in an acceptable manner. When the concept of traceability is applied throughout a project, the cohesiveness of the safety file is greatly improved.

6.5.3 Operation and Maintenance Planning

Objective:

6.5.3.1 To plan how to operate, maintain, and repair the PE-based safety system to ensure functional safety.

Recommendations:

6.5.3.2 Plans should identify—

- Normal and abnormal operation activities
- Preventative maintenance activities and schedules
- Repair activities
- Diagnostic activities
- Procedures to prevent an unsafe state during operation and maintenance
- Circumstances and procedures for bypassing or overriding safety functions or interlocks
- Circumstances and procedures for restoring and verifying safety functions or interlocks after they have been bypassed or overridden

6.5.4 Safety Verification and Validation Planning

Objective:

6.5.4.1 To plan how to confirm by examination and provision of objective evidence that the PE-based safety system meets the safety requirements.

NOTE 35: The verification plan defines the activities that are to be carried out to confirm that the requirements of each safety life cycle and system life cycle phase have been met. Verification activities are to confirm that each phase of activity correctly uses the information of the previous phase so that errors are not introduced from one phase in the life cycle to the next. Verification activities include review, traceability, testing, and audit.

NOTE 36: The validation plan defines the activities that are to be carried out to confirm that the safety systems and the external risk reduction measures achieve the overall safety requirements and, in particular, that the required level of risk reduction has been met. Validation activities are targeted at demonstrating that the safety requirements for the safety systems are correct and will achieve the benefits claimed for them in the environment. Thus, validation activities could include safety requirements analysis, system simulation, system testing, and monitoring during operation.

Recommendations:

6.5.4.2 A safety verification plan and the safety validation plan should be developed by person(s) independent of the system designer(s). This does not preclude the system designer(s) from participating in the plan development. The plan should detail the following:

- When analysis, testing, or assessment activities take place
- Who conducts the analysis testing or assessment activities
- Activities and tests that confirm the safety requirements
- Activities and tests that confirm the system modes
- Confirmation of system modes and mode transitions for the following:
  a. Startup
  b. Shutdown
  c. Reset
  d. Manual
  e. Remote
  f. Semiautomatic
  g. Automatic
  h. Monitor
  i. Standby
  j. Emergency
  k. Stuck/jammed (abnormal)

NOTE 37: This is not a comprehensive listing. A given system might have a subset of the listed modes and/or additional modes.

- Pass and fail criteria
- Procedures for addressing activities that fail the criteria established for the safety requirements

6.5.5 Installation and Commissioning Planning

Objective:
6.5.5.1 To plan how to install and commission the PE-based safety system in a safe manner and to ensure that functional safety is achieved.

Recommendations:
6.5.5.2 The installation and commissioning plan should specify—

- Possible hazards during installation and commissioning
- Safety precautions during installation and commissioning
- Installation, training, and commissioning procedures
- Integration sequences
- Criteria for declaring installation and commissioning complete

6.5.6 Management of Change (MOC) Planning

Objective:
6.5.6.1 To ensure that safety is not adversely affected by changes involving modification or retrofit of the system, subsystem, software, firmware, or hardware.

Recommendations:
6.5.6.2 All safety-critical procedures, systems, subsystems, software, firmware, and hardware should be subject to an MOC plan.

6.5.6.3 MOC plans *do not* pertain to—

- Replacements in kind
- Repairs that are not of a corrective nature
- Recalibrations within specification ranges

6.5.6.4 The MOC plan should establish the change process and document the results.

NOTE 38: An example MOC process model is given in the supplemental guidance report.

6.5.6.5 The MOC plan should contain the following items to identify, analyze, control, and track safety modifications [IEC 1998g]:

- Documentation describing the proposed change, the reasons for the change, and the impact on safety and health
- A hazard and risk analysis
- A method to identify and track the change
- A review and authorization process conducted before implementing the change
- A method to verify modifications

6.5.6.6 Software changes must be made by people as authorized by the manufacturer. They must be competent and knowledgeable about the entire system.

6.5.6.7 The software and firmware should contain a version identifier [Sammarco et al. 1997].

6.5.6.8 All relevant documentation affected by the change should be updated as necessary. This is especially important if the change impacts operation and maintenance procedures.

6.5.6.9 Management of change (including configuration management and document control) plan implementation should begin before the hazard risk analysis phase of the life cycle. Without adequate document and change controls in place, life cycle phases subsequent to the scope phase can easily become corrupted through the use of a wrong version of a document or design output.

NOTE 39: The timing of MOC plan implementation should be decided by the developer. It is highly recommended to implement it during the early stages of design realization.

6.6 Realization

NOTE 40: The realization phase of the safety life cycle (figure 3) is shown in greater detail in figure 6.

Objectives:

6.6.1 To design and create the PE-based safety system.

6.6.2 To follow safety practices for the PE-based safety system and the basic system design.

Recommendations:

6.6.3 Overall System Design Recommendations (see item 9.1 in figure 6)

6.6.3.1 System Safety Precedence (adapted from MIL-STD-882C) [U.S. Department of Defense 1993]

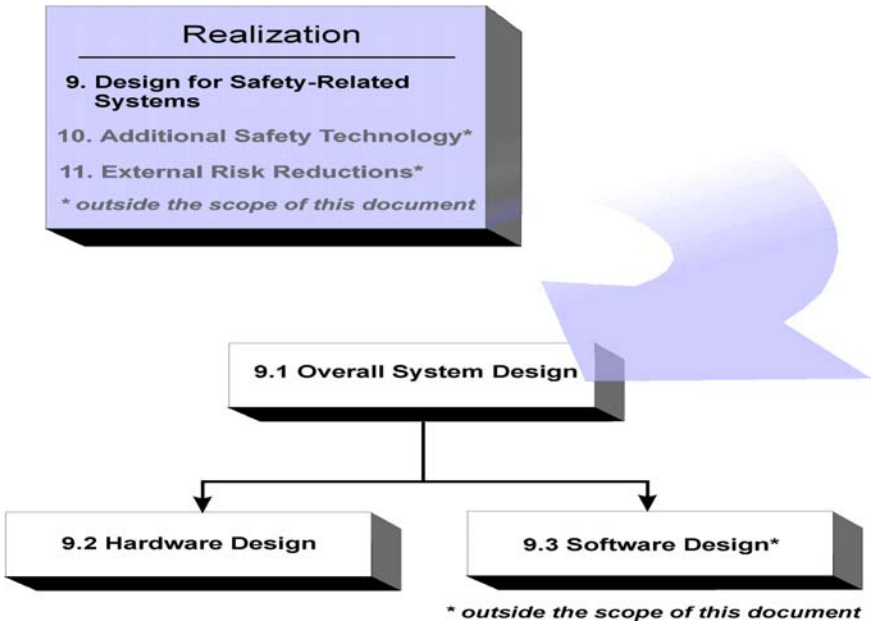


Figure 6.—Realization phase structure.

The main goal is to eliminate hazards by design. The order of precedence for satisfying system safety requirements and resolving identified hazards shall be as follows:

- *Eliminate hazards through design:* The most desired and effective solution.
- *Design for minimum risk:* If hazard elimination is not practical, reduce the associated risk to a tolerable level.
- *Incorporate safety devices:* Reduce the risk to an acceptable level through the use of fixed, automatic, or other protective safety design features or devices. Provisions shall be made for periodic functional checks of safety devices, when applicable.
- *Provide separate warning devices:* Use means to detect the condition and produce an adequate warning signal to alert personnel of the hazard. Warning signals and their application shall minimize the probability of incorrect personnel reaction to the signals and shall be standardized within like types of systems.
- *Develop procedures and training:* Procedures and training shall be used where it is impractical to eliminate hazards through design selection or adequately reduce the associated risk with safety and warning devices. Do not use warning, caution, or other written advisory as the only risk reduction method. Procedures may include the use of personal protective equipment. Standard cautionary notations are recommended.

6.6.3.2 Hazard Elimination: Eliminating a hazard or the sequence of events leading to an accident is most effective and should be realized by these techniques and precedents:

- Substitution
- Simplification
- Decoupling (isolation)
- Elimination of conditions contributing to human error

6.6.3.3 Hazard Risk Reduction: Hazard risk reduction should be realized by these techniques and precedents:

- Simplify operation and maintenance
- Incorporate safety features such as lockouts and interlocks
- Enhance reliability via redundancy and error recovery techniques
- Isolate safety-critical functions

6.6.3.4 Hazard Control: Using methods to minimize the effects of a hazard after it occurs. Control can be realized by these techniques:

- Protection devices, such as watchdog timers
- Fail-safe designs

6.6.3.5 Separation: Physical and functional separation can apply to both hardware and software. This separation is desirable for an SIS. Limited communications between the MCMS and the SIS is acceptable if no common mode failures can occur.

6.6.3.6 Simplification:

- Simplicity of design should be stressed.
- The system should not contain unnecessary or unused features, functions, or components.

NOTE 41: Simplification is one of the most important design aspects for safety. Complexity of design makes it more difficult to understand, design, test, document, maintain, modify, and review. Complexity also makes it more likely for errors, failure, and unplanned interactions that may cause unsafe conditions. It can also increase demands on humans to operate and maintain the system. This can increase the training demands for operators and maintenance personnel. As a result, humans may unknowingly put the system in an unsafe state during operation or maintenance.

Software can be especially prone to complexity since it can contain numerous branches and interrupts and can contain temporal criticality. Software complexity is mainly determined by the degree of difficulty to understand it. One of the largest contributors to software complexity is control flow, the decision points, and branches for software. A complexity matrix is helpful to analyze complexity. Thus, to improve safety, the software should minimize control flow complexity and contain only the required features and capabilities. Undocumented, unused, and unnecessary code should be avoided.

6.6.3.7 Power-up/power loss [NATO 1997]:

- The system must power up in a known safe state (i.e., no machine actions resulting in immediate movement or startup or any other hazardous condition).
- Upon power-up, initialization testing (self-checking) should be used to verify the system is in a safe state and that safety systems such as an SIS are working properly.
- The system must be in a safe state during and after power-up, power loss, and intermittent power faults/interruptions.
- If power is restored after a power loss, the system must power up in a known safe state (i.e., no machine actions resulting in immediate movement or startup or any other hazardous condition).

NOTE 42: *Example:* Power is lost on a remotely controlled continuous mining machine. Unknowingly, the remote-control pendant is activated to pivot the machine while power is lost. Power is unexpectedly restored and the pendant is still activating a pivot command, yet the machine does not create a hazardous situation by pivoting immediately after power-up without giving prior warning.

6.6.3.8 Mode transitions: Mining systems can have manual, semiautomatic, automatic, and remote operational modes, as well as abnormal modes, such as stuck or jammed. A mode transition can cause a hazardous situation.

- A mode transition should not result in a hazardous state, such as machine actions of immediate movement or startup or unexpected movement or startup.

NOTE 43: *Example:* An accident occurred when changing from automatic to manual, then back to automatic operation. The system resumed operation from the last state during automatic mode rather from the last state from the manual mode. This resulted in an unexpected machine movement, injuring the operator.

- Mode transitions should be considered for system operation, repair, and maintenance.
- The following list of modes should be considered for analyzing the safety of mode transitions:
    a. Startup
    b. Shutdown
    c. Reset
    d. Manual
    e. Remote
    f. Semiautomatic
    g. Automatic

h. Monitor
i. Standby
j. Emergency
k. Stuck/jammed (abnormal)

NOTE 44: This is not a comprehensive listing. A given system might have a subset of the listed modes and/or additional modes.

- A mode transition from an abnormal condition, such as stuck or jammed, to a normal condition should not result in a hazardous state, such as machine actions of immediate movement or startup or unexpected movement or startup.

NOTE 45: *Example:* A worker is inspecting a jammed milling machine to determine a course of corrective action. Suddenly, the machine resumes operation, pulling the worker into the rotating blades. Although human error was a significant factor (failure to lock out the machine before inspection), this unexpected machine movement could have been avoided by a machine design that defaulted to a safe state when the mode transition occurred. This design is necessary because it is foreseeable that human error is inevitable, and in this case the consequences are severe. In similar applications, the following design criteria should have been considered:

a. Mode transitions require human initiation and acknowledgment.
b. Designs should provide a warning indication before transition to another mode.
c. A method must be provided for overriding automatic control under emergency situations.
d. The method must have the highest priority to enable manual intervention.

NOTE 46: For example, a remote-controlled system must have alternate provisions for emergency intervention.

- Manual, rather than automatic, reset is recommended.
- A manual reset should be possible only if all safety functions and protective devices are operating properly.
- A manual reset should not initiate machine movement or motion.
- The manual reset should be positioned so that it can be safely accessed.
- Additional measures for consideration concerning manual reset are found in the standard EN 954-1 [NATO 1997].

6.6.3.9 Mode Confusion:

- The system's mode (i.e., manual, automatic, remote, semiautomatic) should be readily identifiable and unambiguous so as not to create confusion or uncertainty.
- Manual intervention to prevent a hazard, failure, or mishap should require a single action and be readily identifiable and unambiguous.
- If a mode offers a number of safety functions, then that mode must only be available if all of those safety functions are available.

6.6.3.10 Extreme Conditions (boundary conditions): Operation or maintenance at extreme conditions or boundary points can be an area of potential high risks and should receive rigorous hazard and risk analysis. Areas of consideration for extreme conditions include—

- Environmental (i.e., temperature, moisture, dust, and vibration)
- Electrical (i.e., EMI, power sources, supply voltages, and data signals)

- Physical (i.e., rate and range of movement)
- Logical (i.e., conditional responses based on Boolean expressions)
- Temporal (i.e., clock times, response times, and delay times)

6.6.3.11 System Changeovers: System changeovers from hardware to software should be carefully analyzed with respect to safety. Hazard and risk analysis should be at a high degree of rigor.

NOTE 47: *Example:* A hardware-based protection circuit with a long, proven life of operation used simple switches and hard-wiring. When the safety function was changed from hardware to software, faults occurred since the new function was more complex, and the design did not take into account new error sources associated with the technology.

6.6.3.12 Unexpected Events: The following measures are recommended to help prevent unexpected events. Additional measures for consideration are found in the standard EN 1037 [BSI 1996].

NOTE 48: An example of an unexpected event is "ghosting" of longwall shields. Sources of ghosting include unintended or accidental human actuation, an component failure, an external event such as an electrical disturbance, a lack of training (understanding) of operation, or a systematic (design) failure.

- Operation and maintenance personnel should have training that identifies *normal* conditions that will cause startup or movement.
- Operation and maintenance personnel should have training that identifies *abnormal* conditions that will cause startup or movement.
- Accidental actuation of controls that could cause startup or movement should be prevented using appropriate protection guards, clear and unambiguous labeling, and proper location of controls.
- Selection and application of field devices that have high reliability and/or failure modes that are readily detectable or failure modes resulting in a safe state.

NOTE 49: Some examples of devices that are highly desirable for safety systems are relays that always fail to a known state, high-reliability dual-coil solenoid valves, and smart analog transmitters having diagnostics rather than discrete on/off limit switches.

- Diagnostic coverage to detect failure modes and bring the system to a safe state.
- A single electrical component failure must not result in startup or movement.
- Cross activation of controls must not be possible.

NOTE 50: An example of cross activation involves a remote-control pendant. While the machine is under control from a given remote-control pendant, other remote control pendants cannot override that same machine.

- If multiple control stations exist, the station with priority control must clearly be defined.

6.6.3.13 Safety Interlocks:

- Designs incorporating safety interlocks must not enable them to be in a bypass or an override state during normal operation.
- Safety interlocks must not be designed such that they could be inadvertently bypassed or overridden.

- Safety interlocks must be designed using human factor principles such that they are not unduly difficult or uncomfortable.

6.6.3.14 *Reuse:* Component or subsystems reused from other systems or applications must be subjected to all safety life cycle phases in the context of the new system and operating environment.

NOTE 51: Reuse applies to software or hardware. A software module or hardware component from another application cannot be assumed to be "safe" and omitted from any safety life cycle activity.

6.6.4 Hardware Recommendations *(item 9.2 of figure 6)*

6.6.4.1 Safety requirements specific to the hardware should be developed and should use as its inputs the results of the overall safety requirements and the safety requirements allocation phases.

6.6.4.2 The hardware safety requirements should be structured with safety functions and safety integrity requirements.

6.6.4.3 A safety life cycle should be used for the hardware design. The hardware design life cycle phases should include the safety requirements, design and development, installation and commissioning, operation and maintenance, integration, and safety validation phases.

NOTE 52: Essentially, this life cycle is a reiteration of the system level life cycle process presented in this report. It contains details specific to the hardware design.

6.6.4.4 If the hardware is used to implement a set of safety functions, then the SIL requirements for that hardware is the highest SIL within the set of safety functions.

6.6.4.5 If the hardware implements both safety functions and nonsafety functions, then the hardware is treated as a safety function.

6.6.4.6 Safety functions should be separate and independent from nonsafety functions.

6.6.4.7 The hardware architecture should be appropriate to meet the safety requirements.

NOTE 53: For example, a dual, redundant architecture is typical for a complex subsystem using PE and requiring an SIL 2.

6.6.4.8 To avoid failures, appropriate techniques and measures should be taken to control systematic failures and random hardware failures. For additional information, consult the following references:

- *Systematic failures:*
  Tables A.16 to A.19 and tables B.1 to B.6 of IEC 61508, part 2 [IEC 1998b]. Definitions of these techniques can be found in annex A of IEC 61508, part 7 [IEC 1998g].

- *Random hardware failures:*

Tables A.1 to A.15 in IEC 61508, part 2 [IEC 1998b] and table A.2.1 in appendix A of the second edition of UL 1998 [Underwriter Laboratories, Inc. 1998]. Definitions of these techniques can be found in annex A of IEC 61508, part 7 [IEC 1998g] and in section A7 of appendix A of the second edition of UL 1998 [Underwriter Laboratories, Inc. 1998].

6.6.4.9 For additional safety design considerations, consult the information presented in appendix B of the standard ISA-S84.01 [ANSI/ISA 1996]. This information is of an informative nature and is not a requirement of the standard.

6.6.4.10 For additional basic design and test information concerning control devices and programable controllers for electric motor control, consult the information presented in the standard UL 508 [Underwriter Laboratories, Inc. 1997].

6.6.4.11 Fault Detection and Tolerance

- Fault detection should be active for all modes of operation, especially for startup, shutdown, and unintended mode transitions.
- If a dangerous fault is detected, then the results listed in table 7 should be achieved.

NOTE 54: The reaction to achieve or maintain a safe state should be specified in the safety requirements.

Table 7.—Fault tolerance outcomes for detection of a dangerous fault

Without fault tolerance	With fault tolerance
• Achieve or maintain a safe state • Correct fault as soon as possible	• Achieve or maintain a safe state • Continue normal operation with fault annunciation or degraded operation with fault *and* degraded operation annunciation • Correct fault as soon as possible

A hardware fault tolerance of N means N % 1 faults could cause an unsafe state or loss of safety function.

NOTE 55: A hardware fault tolerance of 0 means a single fault could cause a dangerous failure. A fault tolerance of 1 means a single fault from a random hardware failure can occur without leading to a dangerous failure.

- Fault tolerance and fault detection is applicable to subsystems composed of logic devices, sensors, and actuators.
- Sufficient fault tolerance should be used as shown by table 8 to provide system robustness to failures relative to the safety integrity level required for the safety function. The recommended fault tolerance number is given in the various boxes of the table.

Table 8.—Minimum fault tolerance of simple and complex subsystems that deenergize to trip

	SIL 1	SIL 2	SIL 3
Simple subsystem	0	0	1
Complex subsystem	0	1	2

NOTE 56: Simple subsystems have devices without PE (i.e, relays, simple transducers, etc.). Complex subsystems have devices using PE (i.e., PLCs, smart valves and transmitters, etc.).

- Watchdog timers with a separate time base should be used to monitor the behavior and plausibility of computer operation and program sequencing.
- Online detection of faults should be used. Online detection must be in parallel with normal system operation and should not cause an unsafe state.
- Offline detection of faults (i.e., running of diagnostic tests when the system is not in an operating mode) should not cause an unsafe state.
- Diagnostic tests should identify failures to the level of a field replaceable or repairable module.
- Failures of components listed in table 9 should be detected during operation.

Table 9.—Hardware components for fault or failure detection

Component	Consult the following table(s) of IEC 61508, part 2:
Discrete hardware	A.3, A.7, A.9
Digital I/O	
Analog I/O	
Power supply	
Bus	A.3
General	
Memory management	
Direct-memory access	
Bus arbitration	
CPU	A.4
Registers	
RAM	
Program counter	
Stack pointer	
Addressing	
Instruction decoding and execution	
Interrupt handling	A.5
Invariable memory (i.e., ROM)	A.6
Variable memory (i.e., RAM)	A.7
Clock (quartz)	A.12
Communication and mass storage	A.13
Field devices (i.e., sensors, actuators, electromechanical devices)	A.2, A.14, A.15

6.6.4.12 Field Devices

- Where the field device is shared between the basic and safety control system, failure of the basic control system should not degrade the performance of the safety control system.
- Diagnostics for field devices should be used.
- Smart devices (i.e., sensors and valves with built-in diagnostics) are recommended for improving diagnostics and reliability.

NOTE 57: For example, a sensor with an analog output provides more diagnostic capabilities than a discrete on/off limit switch. Diagnostic checks of range, rate, and data plausibility are possible with analog output.

- Safety field devices identified in the safety plan should be visually distinguishable (e.g., special label, markings, or color) from all other devices.
- Deenergization of fail-safe devices should result in a safe state.

- Electrical contacts and connections should be hermetically sealed.
- Electrical cable or connector failures must not cause an unsafe state.

NOTE 58: Electric cables and connectors are field devices. Failures could result from physical damage or environmental stressors such as water, moisture, temperature, dust, and vibration.

- Safety devices should be of a deenergize-to-trip type.
- If a safety device is of an energize-to-trip type, then appropriate techniques are needed to increase the fault tolerance of table 8 by 1.

6.6.4.13 Operator Interfaces

- The operator interface is safety function.
- Designs of operator interfaces should prevent accidental actuations causing a hazardous situation, such as startup or machine movement.

NOTE 59: Accidental actuations or unintentional human contact, designs using such things as guards, interlocks, and control logic are commonly used.

- Operator interfaces must not enable modification of safety software.
- Cancellation of a current operation should require a single action (input) by the operator.
- Cancellation of a current operation should result in a safe state.
- Emergency responses should require a single action (input).
- Incorrect input data or input sequence should result in a safe state.
- When two or more operator inputs are used to initiate an operation resulting in startup, machine movement, or mode change, the system should provide confirmation of the operation.
- The operator interface design should incorporate human factors and ergonomics.
- All interactions between the operator and system should be defined and documented.
- The operator interface should indicate—
    a. The process sequence.
    b. The current mode (i.e., automatic, manual, or abnormal)
    c. The failure or degradation of a safety system.
    d. Degraded system operation due to a failure.
    e. Alerts to any safety function bypass.

6.6.4.14 Maintenance/Diagnostic Interfaces

- Interfaces for maintenance/diagnostic purposes must be accessible, physically and visually, such that personnel are not exposed to a hazardous situation (e.g., an awkward or unsafe location that could place them in danger from unexpected machine movements, roof/rib falls, etc.) in order to use the interfaces.
- Interfaces must not allow unauthorized users to modify any safety application software.

6.6.4.15 Environmental

- The design of the safety system must consider all environmental conditions. This includes consideration of the following: temperature, humidity, water impingement and flooding, electromagnetic interference (EMI), radio frequency interference (RFI), shock, vibration, and contaminants.

6.6.5 Software Design Recommendations *(item 9.3 of figure 6)*

- The recommendations for this clause are contained in the report addressing software safety, shown as part 2.2 of the safety framework of figure 1.

NOTE 60: Software safety is addressed in a separate report in this series to enable detailed coverage. It is important to remember that software is an integral part of the system and must be addressed in the context of the system.

6.7 Install and Commission

Objective:
6.7.1 To install and commission the safety system properly and safely.

Recommendations:
6.7.2 To install and commission in accordance to the plans.

6.7.3 To identify and resolve failures and incompatibilities.

6.8 Validate

Objective:
6.8.1 To carry out the validation plan.

Recommendations:
6.8.2 The validation results should be documented.

6.8.3 Instruments used for validation must be calibrated.

6.8.4 Validation documentation should include the—

- Safety requirements version
- Safety function validated
- Mode validated
- Mode transition validated
- Test, tools, and equipment used
- Validation results

6.8.5 Self-validation is not permitted (i.e., validation is carried out by people independent of the design; see section 6.5.4.2).

6.9 Operate and Maintain

Objective:
6.9.1 To operate, maintain, and repair the PE-based safety system so that functional safety is maintained.

Recommendations:
6.9.2 Operation, maintenance, and repair should be in accordance to plans.

6.9.3 Operation and maintenance manuals must be completed and delivered to the end user prior to operation and maintenance.

6.9.4 Training must be completed before operation and maintenance.

6.9.5 Training—

- Must detail the potential hazards during operation and maintenance and the means to control them.
- Should address the following:
  a. System description
  b. System operating principles (i.e., theory of operation)
  c. Safety functions
  d. Safety systems operation, testing, and maintenance
  e. Hazards that the safety system is protecting against
  f. Description of all modes and mode transitions
  g. Safety warning and alarms
  h. Operator interfaces
  i. System operation
  j. Emergency operation for single-failure modes
  k. Emergency operation for multiple-failure modes occurring at once
  l. Safe system maintenance
  m. Manual operation/intervention

NOTE 61: Training content and materials can be used from some safety life cycle activities. Examples include hazard and risk analysis results, risk controls, safety requirement specifications, and operation and maintenance manuals.

- The degree of rigor for training should increase as the SIL increases, as shown in table 10.
- Training goals should be identified in the early life cycle phases.
- Training using all training materials, including operation and maintenance manuals, must be completed before the system's commissioning.
- Operation or maintenance activities must be done by persons attaining the appropriate skill levels, as shown by table 10.

Table 10.—Training

	SIL 0 or SIL 1	SIL 2	SIL 3
Training level . .	Basic instruction with training material.	Intermediate instruction with training manuals.	Advanced course with testing; certification of person recommended.
Skill level	Basic	Intermediate.	Advanced, i.e., understanding of theory of operation and multiple-failure modes. In-depth understanding of all safety functions.

6.9.6 Maintenance must be conducted as scheduled.

6.9.7 Operation and maintenance documentation must be updated in conjunction with system modifications.

6.9.8 Modifications of operation or maintenance should be documented.

6.9.9 Safety modifications of operation or maintenance are subject to MOC procedures.

6.9.10 The safety file must be updated to reflect safety modifications of operation or maintenance.

6.10 Modifications

Objectives:

6.10.1 To make and document all modifications in accordance to the MOC plan.

6.10.2 To ensure that appropriate safety is established and maintained during and after modifications.

Recommendations:

6.10.3 Review and approval, in accordance with the MOC, is needed prior to any safety modification.

6.10.4 Before normal system operation resumes, testing or other means must be used to verify the modification was properly implemented and the system performs as desired.

6.10.5 If the modification impacts operation or maintenance, then training must take place before normal system operation resumes.

6.11 Decommission

Objective:

6.11.1 To ensure that appropriate safety is established and maintained during and after decommissioning.

Recommendations:

6.11.2 Before decommissioning, procedures should be prepared for—

- Closing down to an inactive, safe state
- Dismantling
- Removal
- Storage (mothballed for possible reuse)

7.0 Safety File

Objective:
7.1 To document safety claims and supporting information that the PE safety system is adequately safe over its lifetime for a given application.

Recommendations:
7.2 The recommendations for this clause are contained in the safety file document, shown as part 3 of the safety framework of figure 1.

8.0 Independent Assessment

Objective:
8.1 To arrive at a judgment on the functional safety of the PE safety system based on investigation and documentation of the safety file.

Recommendations:
8.2 The recommendations for this clause are contained in the independent assessment document, shown as part 4 of the safety framework of figure 1.

REFERENCES

ANSI/ISA [1996]. Application of safety instrumented systems for the process industries. Research Triangle Park, NC: American National Standards Institute, ANSI/ISA S84.01-1996.

BSI [1996]. Safety of machinery - prevention of unexpected start-up. London, U.K.: British Standards Institute, BS EN-1037, pp. 3-8.

BSI [1997]. Safety of machinery: safety related parts of control systems, part 1: general principles for design. London, U.K.: British Standards Institute, BS EN-954-1, pp. 6-11.

IEC [1998a]. Functional safety of electrical/electronic/programmable electronic safety-related systems. Geneva, Switzerland: International Electrotechnical Commission, Draft IEC65108-1, Part 1: General requirements, version 4, May 12, 1998.

IEC [1998b]. Functional safety of electrical/electronic/programmable electronic safety-related systems. Geneva, Switzerland: International Electrotechnical Commission, Draft IEC65108-2, Part 2: Requirements for electrical/electronic/programmable electronic safety-related systems, version 4, May 12, 1998.

IEC [1998c]. Functional safety of electrical/electronic/programmable electronic safety-related systems. Geneva, Switzerland: International Electrotechnical Commission, Draft IEC65108-3, Part 3: Software requirements, version 4, May 12, 1998.

IEC [1998d]. Functional safety of electrical/electronic/programmable electronic safety-related systems. Geneva, Switzerland: International Electrotechnical Commission, Draft IEC65108-4, Part 4: Definitions and abbreviations, version 4, May 12, 1998.

IEC [1998e]. Functional safety of electrical/electronic/programmable electronic safety-related systems. Geneva, Switzerland: International Electrotechnical Commission, Draft IEC65108-5, Part 5: Examples of methods for determination of safety integrity levels, version 4, May 12, 1998.

IEC [1998f]. Functional safety of electrical/electronic/programmable electronic safety-related systems. Geneva, Switzerland: International Electrotechnical Commission, Draft IEC65108-6, Part 6: Guidelines on the application of parts 2 and 3, version 4, May 12, 1998.

IEC [1998g]. Functional safety of electrical/electronic/programmable electronic safety-related systems. Geneva, Switzerland: International Electrotechnical Commission, Draft IEC65108-7 Part 7: Overview of techniques and measures, version 4, May 12, 1998.

ISA [1998]. Safety instrumented systems (SIS) - safety integrity level (SIL) evaluation techniques, parts 1-5. Research Triangle Park, NC: Instrument Society of America, ISA dTR84.0.02.

Leveson NG [1992]. High-pressure steam engines and computer software. In: Proceedings of the International Conference on Software Engineering (Melbourne, Australia).

Ministry of Defence [1998]. HAZOP studies on systems containing programmable electronics. Glasgow, U.K.: Ministry of Defence, Directorate of Standardisation, Defence Standard 00-58, parts 1 and 2.

NATO [1997]. Safety requirements and guidelines for munition related safety critical computing systems, Draft NATO Standardization Agreement (STANAG) 4404, edition 1, Document AC/310-D/139.

Sammarco JJ, Kohler JL, Novak T, Morley LA [1997]. Safety issues and the use of software-controlled equipment in the mining industry. In: Proceedings of the IEEE-Industrial Applications Society 32nd Annual Meeting (October 5-9, 1997).

Sammarco JJ, Fisher TJ, Welsh JH, Pazuchanics MJ [1999]. Programmable electronics in mining: an introduction to safety. Pittsburgh, PA: NIOSH Special Workshop Report. Unpublished.

Stephans RA, Talso WW [1997]. System safety analysis handbook. 2nd ed. Albuquerque, NM: The System Safety Society, Section 3.

Suokas J, Rouhianinen V [1992]. Quality management of safety and risk analysis. Elsevier.

Underwriter Laboratories, Inc. [1997]. Standard for industrial control equipment (UL 508). Northbrook, IL: Underwriter Laboratories, Inc.

Underwriter Laboratories, Inc. [1998]. Standard for software in programmable components (UL 1998). Northbrook, IL: Underwriter Laboratories, Inc.

U.S. Department of Defense [1980]. Procedures for performing a failure mode effects and criticality analysis. Military Standard MIL-STD1629A.

U.S. Department of Defense [1993]. System safety program requirements. Military Standard MIL-STD-882C.

Vesely WE, Goldberg FF, Roberts NH, Haasl DF [1981]. Fault tree handbook (NUREG- 0492). Washington, DC: U.S. Nuclear Regulatory Commission, Office of Nuclear Regulatory Research, Systems and Reliability Research.

APPENDIX A.—CHECKLIST EXAMPLES

Checklists can be used as a hazard analysis technique to list hazards, stimulate thinking, and to pass on lessons learned from mistakes and experience. They can also be used to systematically list design procedures and practices, thus reducing the likelihood of errors of omission. Some checklist questions can be answered with a "yes" or "no"; others are for the purpose of stimulating thought.

Checklists are not limited for use in hazard analysis; they can be used in all life cycle phases. It is important for users to tailor the checklists for their situation and to make checklists a "living" entity that is updated as needed.

The following checklists are for example purposes. Users should tailor them for their situation, making additions or deletions as needed. These checklists have been adapted from published material and have been created based on analysis of MSHA accident data.

System Checklist

Can a single-point failure cause a hazardous state?	
Has the role of the operator in maintaining safety been defined?	
Does the specification avoid the need for the safety functions to be inhibited under certain conditions? If not: (1) Have adequate grounds for inhibiting safety functions been established? (2) Have procedures for the safe use of inhibits been developed covering the actions to be taken before, during, and after their application?	
Are all external hardware and software interfaces specified?	
Has a means been specified to limit the ranges of main control inputs (e.g., trip settings) to safe values?	
Has a means been specified to detect out-of-range conditions for system inputs and outputs?	
Has the physical operating environment been defined and a suitable design been specified with regard to— (1) Temperature range? (2) Humidity? (3) Vibration and shock? (4) Ingress of water and dust? (5) Contaminating gases? (6) Hazardous atmospheres? (7) Power supply voltage tolerance? (8) Power supply interruptions? (9) Pressure/vacuum (internal, external, under pressure/vacuum)?	

Are all equipments and components operated within their rated performance for the specified operating and environmental conditions?	
Is the system designed to go to a safe state in the event of— (1) Loss of power supply? (2) Cabling faults (open or short circuit or earth faults)? (3) Loss of hydraulic supply?	
Is there a procedure that strictly controls the conditions under which alarms, trips, and control functions may be inhibited?	
Is there a formal approval procedure that considers the safety implications of all modifications?	
Is there a procedure for the control of changes in requirement?	
Is the final specification checked against the user requirements by persons other than those producing the specification before beginning the design phase?	
Is there a procedure for the control of changes in design or specification?	
If input arrives when it shouldn't, is a response specified?	
Are there sufficient delays incorporated if error-recovery responses, e.g., to avoid returning to the normal state too quickly?	
Are the inputs identified, which, if not received (for example, due to sensor failure), can lead to a hazardous state or can prevent recovery (single-point failures)?	
Are data used by critical software protected by error checking?	
Does the system provide for operator cancellation of current processing?	
Can mode of operation changes between manual, automatic, and remote cause an unsafe state?	
If multiple remote controllers are available, is only one controller at a time activated (i.e., no cross activation)?	
Is the emergency stop system independent of the hardware and software used for system operation?	
Will the loss or restoration of electrical power cause an unsafe state?	
Can failure to turn on or turn off solenoids or activators cause an unsafe condition?	
Does the system contain safeguards for user errors and misuse?	
Can an operational mode change occur during manual operation? For example, can manual control be preempted by automatic or remote control?	

Software Checklist

Within the software specification, is there a clear and concise statement of— (1) Each safety function to be implemented? (2) The information to be given to the operator at any time? (3) The required action on each operator command, including illegal or unexpected commands? (4) The communications requirements between the PES and other equipment? (5) The initial states for all internal variables and external interfaces? (6) The required action on power-down and recovery (e.g., saving of important data in nonvolatile memory)? (7) The different requirements for each phase of plant/machine operation (e.g., startup, normal operation, shutdown)? (8) The anticipated ranges of input variables and the required action on out-of-range variables? (9) The required performance in terms of speed, accuracy, and precision? (10) The constraints put on the software by the hardware (e.g., speed, memory size, word length)? (11) Internal self-checks to be carried out and the action on detection of a failure?	
Does the software contain adequate error detection facilities allied to error containment, recovery, or safe shutdown procedures?	
Are safety-critical areas of the software identified?	
Is access to the safety-critical sections of the software limited to authorized and competent people?	
Are there adequate procedures for the control of software versions?	
Does the software contain only those features and capabilities required by the system? The software should not contain "undocumented" features.	
Does the system design prevent unauthorized or inadvertent access or modification to the software?	
Are feedback loops designed such that software runaway conditions do not occur due to feedback sensor or limit switch failures?	
Does the software detect improper operator inputs or sequences? If so, are alerts used to indicate the error and corrective actions?	
Is the software's response to out-of-range values specified for every input?	
Is the software's response to not receiving an expected input specified (i.e., are timeouts provided)?	
Does the software specify the length of the timeout, when to start counting the timeout, and the latency of the timeout (the point past which the receipt of new inputs cannot change the output result, even if they arrive before the actual output)?	

On a given input, will the software always follow the same path through the code (i.e., is the software's behavior deterministic)?	
Is each input bounded in time, i.e., does the specification include the earliest time at which the input will be accepted and the latest time at which the data will be considered valid (avoiding control decisions based on obsolete data)?	
Can a one-bit change in a register or variable, or a change to all 0's or all 1's, result in an unsafe decision outcome?	
Do critical data communicated between CPUs successfully pass data transmission verification checks in both CPUs?	
Do memory locations have unique references (e.g., one and only one name) to prevent memory usage conflicts between global/local variables?	
Do software-controlled sequences affecting safety require a minimum of two independent procedures for initiation?	
Does the software terminate to a known and predictably safe state?	
Are unused memory locations initialized to a pattern that, if executed as an instruction, will cause the system to revert to a known safe state?	
Are critical operational software instructions resident in nonvolatile read-only memory (ROM)?	
If a safety kernel is used, does it reside in ROM?	
If multiple, identical subsystems are used, is a mechanism in place to verify the proper software version for each subsystem?	

Hardware Checklist

Can any output be produced faster than it can be used (absorbed) by the interfacing module? Is overload behavior specified?	
Can input that is received before startup, while offline, or after shutdown influence the software's startup behavior? For example, are the values of any counters, timers, or signals retained in software or hardware during shutdown? If so, is the earliest or most recent value retained?	
Is the I/O separated such that safety outputs are protected or partitioned?	
Can the failure of any input or output device cause an unsafe state?	
Are all inputs and outputs protected from damage from voltage spikes that may be induced on input cables?	
Are all outputs that switch inductive loads protected from damage from switching spikes?	

Do all nonvolatile devices and relays reset to a known, safe state in anticipation of system restart or random power inputs/outputs?	
Is critical hardware controlled by software initialized to a known safe state?	
Will sticking or malfunctioning solenoid valves place the system in an unsafe state?	
Are there specifications and procedures for hardware testing?	
Do watchdog timers or similar devices ensure that the processor is functioning correctly?	
Will loss or restoration of hydraulic pressure or flow cause an unsafe state?	

Human Factors Checklist

Is the operator/machine interaction defined for every operating state?	
In the event of automatic control failure, is enough information given to the operator to allow him or her to assume manual control safety?	
Can all operational settings be readily inspected to ensure that they are correct at all times?	
Are switches and levers located to prevent accidental contact?	
Are emergency stop switches readily accessible and clearly defined?	
Are positions and functions of switches and controls clearly indicated?	
Has training been given appropriate to the tasks to be carried out and the personnel involved?	
Is there sufficient independence between those carrying out the work and those inspecting it?	
Have appropriate procedures been developed and implemented to prevent unauthorized access to the system?	
Does the operator's manual describe the risk associated with possible failures and the necessary action on failure?	
Are routine alerts distinguished from safety-critical alerts?	
Can unwanted hazardous states be generated in a training mode?	
Do emergency stops require a single keystroke operator response?	
Are inadvertent inputs by the operator prevented?	
Do operator interfaces acknowledge input by methods such as echoing keystrokes on the screen?	

Maintenance and Rebuild Checklist

Are provisions specified for maintaining safety during maintenance and field modification?	
Are there clear indications that a safety function is bypassed for maintenance purposes?	
Are there clear procedures to bypass safety functions?	
Are there clear procedures to restore and verify that a safety function that was bypassed?	
Do maintenance procedures ensure safety during maintenance?	
Are maintenance procedures sufficiently explicit so that they do not leave interpretations or important decisions to be made by maintenance personnel?	
Does the final action of any maintenance or test procedure ensure that the system is returned to its normal operating state?	