DOCUMENTING CONTROL SYSTEM FUNCTIONALITY FOR DIGITAL CONTROL IMPLEMENTATIONS

J. Harber, M. Borairi, S. Tikku, A. Josefowicz
Atomic Energy of Canada Limited
Mississauga, Ont.

Abstract

In past CANDU designs, plant control was accomplished by a combination of digital control computers, analogue controllers, and hardwired relay logic. Functionality for these various control systems, each using different hardware, was documented in varied formats such as text based program specifications, relay logic diagrams, and other various specification documents. The choice of formats was influenced by the hardware used and often required different specialized skills for different applications.

The programmable electronic systems in new CANDU designs are realized in a manner consistent with latest international standards (e.g., the IEC 61513 standard [1]). New CANDU designs make extensive use of modern digital control technology, with the benefit that functionality can be implemented on a limited number of control platforms, reducing development and maintenance cost. This approach can take advantage of tools that allow the plant control system functional and performance requirements to be documented using graphical representations. Modern graphical methods supplemented by information databases can be used to provide a clear and comprehensive set of requirements for software and system development. Overview diagrams of system functionality provide a common understanding of the system boundaries and interfaces. Important requirements are readily traced through the development process. This improved reviewability helps to ensure consistency with the safety and and production design requirements of the system.

Encapsulation of commonly used functions into custom-defined function blocks, such as typical motor control centre interfaces, process interlocks, median selects etc, eases the burden on designers to understand and analyze the detailed functionality of each instance of use of this logic. A library of encapsulated functions will be established for complex functions that are reused in the control logic development. By encapsulation and standardisation of such complex functions, the time required for development and verification activities can be reduced significantly. The library functions can be pre-validated and re-used as trusted components. This reduces development time and minimizes errors. Maintainability is also improved.

Background

For many years, AECL led the implementation of direct digital control in process control and safety applications. In existing CANDUs, plant control is performed by digital control computers (DCCs), analog control devices and relay logic. At the highest level, system control is performed by dual redundant computers which execute a set of programs for monitoring, operator display, annunciation, and control of important plant systems. At a lower level, control devices such as analog controllers and hardwired relay logic handle individual device control functions. The application programs for the control computers are written in low-level programming languages such as assembler, while the lower level device control logic is written in a symbolic language or is performed in hardwired logic.

AECL has made advances in the application of computerized systems by developing and implementing multiple diverse computerized safety systems for reactor shutdown in the recent past. In implementing computerized safety systems, AECL has produced, in concert with Canadian utilities, software development standards and internal software development practices for design of safety related computer systems.

Monitoring and control programs for the existing DCCs were developed based on text format program specification documents. Overall understanding of the program functionality is developed by reviewing the program specification in conjunction with the system flowsheets and other relevant documentation.

Lower level device control was typically implemented using low voltage relays and switches configured to perform control and interlock functions. Each control circuit was typically derived from an application standard and tailored to the specific application in terms of the process inputs/outputs, specific operator control/display components, and interlocks. Implementation of a control circuit generally meant assigning specific relays/contacts located in control panels to the circuit, assignment of specific electrical conductors and terminals in the bulk cabling network to the circuit.

Modern Implementation Considerations

The availability of modern and efficient programming tools for producing application logic (i.e., control software) which can be implemented on computer hardware (i.e. typically a controller) with comprehensive self checking and diagnostics brings new opportunities in control system design. Modern tools for documentation, traceability, and simulation of control functions also provide new benefits for designers.

Modern programming tools provide a format, (e.g. the programming languages documented in the IEC 61131-3 standard)[2], for documenting control functionality in graphical and hierarchical representations. Based on the process system design, system overview diagrams, supplemented by safety analysis and previous experience, form a starting point for the development of the control strategy . Overview representations of system functionality help clarify the control and monitoring functions required for the system and the interfaces between plant systems. In accordance with modern design practices, such as the IEC 61513 standard [1], system monitoring and control actions are analyzed from first principles to establish the safety significance of the individual functions.

The safety significance of the function is noted as a key parameter in the design process as it determines the rigour to be applied in the subsequent design activities. While safety requirements are emphasized in the relevant safety standards, compliance with production related requirements is also vitally important. System and software requirements are tracked throughout the development process for validation and verification of system functions in the final system. Tracking of control system requirements ensures consistency in the plant design in terms of implementation and plant safety analysis documentation.

Functionality previously provided by a mix or hard-wired, analog, and digital control hardware may be implemented on a single platform reducing interfaces within the control system. Requirements at the initial (i.e., conceptual) design stages are determined without reference to the eventual implementation platform, as selection of the platform itself is determined by these requirements. In the final realization of the system design, some functions may be implemented on a commercial grade controller platform, other portions in a safety qualified platform, and a limited amount of functionality may be implemented in hardwired relay logic depending on the safety significance and production reliability required for the function, cost factors, or the need for independent back-up functionality. Figure 1 provides a typical overview of the various control platforms in a plant control system.

Mathematical models of the system are also valuable in the development of process control simulations. Models of plant functionality and proposed control system are necessary in developing advanced safety-related control strategies. Modern modelling and simulation packages such as Matlab and Simulink enable rapid development of complex control algorithms. The Matlab software application provides analysis tools, which enable the use of idealized linear models or if necessary, to explore more complex non-linear models. Models are hierarchical and can be built using top-down or bottom-up approaches. The use of such modeling tools also facilitates integration of different simulation environments to design and develop deterministic and supervisory control systems in a graphical environment.

Operator interface stations in the control room and the field make use of modern display technologies for improved operational assistance over those provided by hardwired controls and displays. Use of modern Human Factors Engineering (HFE) concepts in the development of operator interfaces can significantly enhance plant operation. Reliable and secure communication protocols are available for connections between control and monitoring stations and operator interface stations.

Definition of Control Requirements

The Control Functional Specification documents capture the requirements for the application functions in functional terms rather than in terms of computer technology so that they are platform independent and are readily understood and reviewable by the I&C functional engineers, process engineers, and plant operators. Figure 2 provides an overview of the development of a process control system and illustrates how the ‘Control Functional Specifications’ form an integral component in the development process.

Starting with a system overview diagram, the functions for the system are defined in a hierarchical manner. A graphical format is primarily used. This can take the shape of information flow diagrams (as is the case with the System Overview Diagram of Figure 3) or functional block logic (as in Figure 5). The graphical representation is supplemented by text (when needed) to cover complex mathematical equations, parameter definitions and performance measures such as functional timing and accuracy requirements. Fail-safe behaviour of the control functions and their outputs is identified. Based on the configuration of the process system, monitoring and control functions of the various components within the system are developed. Interfaces with the controls of the service systems, e.g. cooling water and electrical power, are identified.

Special considerations may be required in the development of low level device control logic. Translation of coil and contact based logic typically used in hardwired relay based logic circuits provides a challenge to programmers who are more familiar with procedural languages. The intent is not necessarily to implement all of the nuances of the relay logic implementation. The interpretation and understanding of complex relay circuit based concepts such as a "fail-safe" de-energized state and redundant path circuit principles represent a challenge to an inexperienced engineer. Figures 4 and 5 show the differences between a typical standard function implemented in relay and contact representation verses a functional block representation.

All functions are categorized based on their importance to safety. Although the classification may not affect the way the functional requirements are defined, they are an important input in determining the allocation of functions to the target hardware in the detailed design. They also determine the safety classification and qualification requirements for the target platform and the software work practices to be followed for implementation. Safety-related functions need to be traceable to the plant safety analysis.

A review of final elements (i.e., interfacing end devices such as motorized valves, motor control centres, circuit breakers) is performed at the beginning of the project to define typical end devices where repetitive logic may be encapsulated in a library of standard functions. Standard functions will encapsulate the required control permissives, interlocks, necessary outputs and any associated intermediate output states in a re-useable package. Process interlocks are also classified and standardised based on factors such as permitted operator overrides, auto / manual, return to normal etc. Standard re-useable functions also help reduce the complexity by minimizing the number of different ways of accomplishing the same functionality. These standard functions are then captured in project design guides so that a consistent implementation across different systems can be facilitated.

Development of encapsulated logic allows standard functions to be implemented consistently across the control system. Implementation of standard functions in software reduces the quantity of comprehensive "wire by wire" checks and testing of hardwired logic implementations. This also reduces the quantity of connection wiring diagrams and wiring lists as logic is equivalent to that shown in elementary wiring diagrams.

HFE concepts applied to operator interface functions are consistent with the complexity and safety significance of the individual monitoring and control function. Standard operator interface templates are developed for the various end devices. The interaction (signal exchange etc) of these templates with the control functions is standardised and documented in project design guides. Backup displays and/or manual controls are provided depending on safety and economic significance of the function.

The control functional specification documents thus developed are used as an input into the development of software specifications. The software specifications take into consideration the target platform and maps the application functions to the chosen hardware architecture. They provide all the information needed for the subsequent activities of the system safety implementation, including the system design and verification phases. The software specifications also specify the software architecture to be used and typically add application functionality influenced by the target platform, such as self-checking features, initialisation of variables, and irrational handling (i.e., self-check functions can be added without adding more components which could in effect, reduce reliability).

The control functional specification documents are also used as an input into the functional validation of the control system design. The validation process verifies the correctness of the control system functionality against the plant performance and functional requirements. Part of the system validation of the control system design includes validation of the realized system design against the system (hardware and software) requirements.

Conclusions

AECL has a long history of successful computer applications for CANDU nuclear power plants. Modern computer methodologies, latest standards in the nuclear industry, and the experience that has been gained in AECL in implementing computer control systems are being applied to defining requirements for a modern nuclear control system for the Advance CANDU Reactor (ACR).

Recent experience with the design of several ACR systems indicates the application of modern digital technology combined with the use of graphical design tools can significantly reduce the effort required to deliver control systems that meet the latest standards and regulations

REFERENCES

[1] IEC 61513, Nuclear power plants, Instrumentation and control for systems important to safety – General requirements for safety.

[2] IEC 61131-3, Programmable Controllers, Part 3: Programming languages.

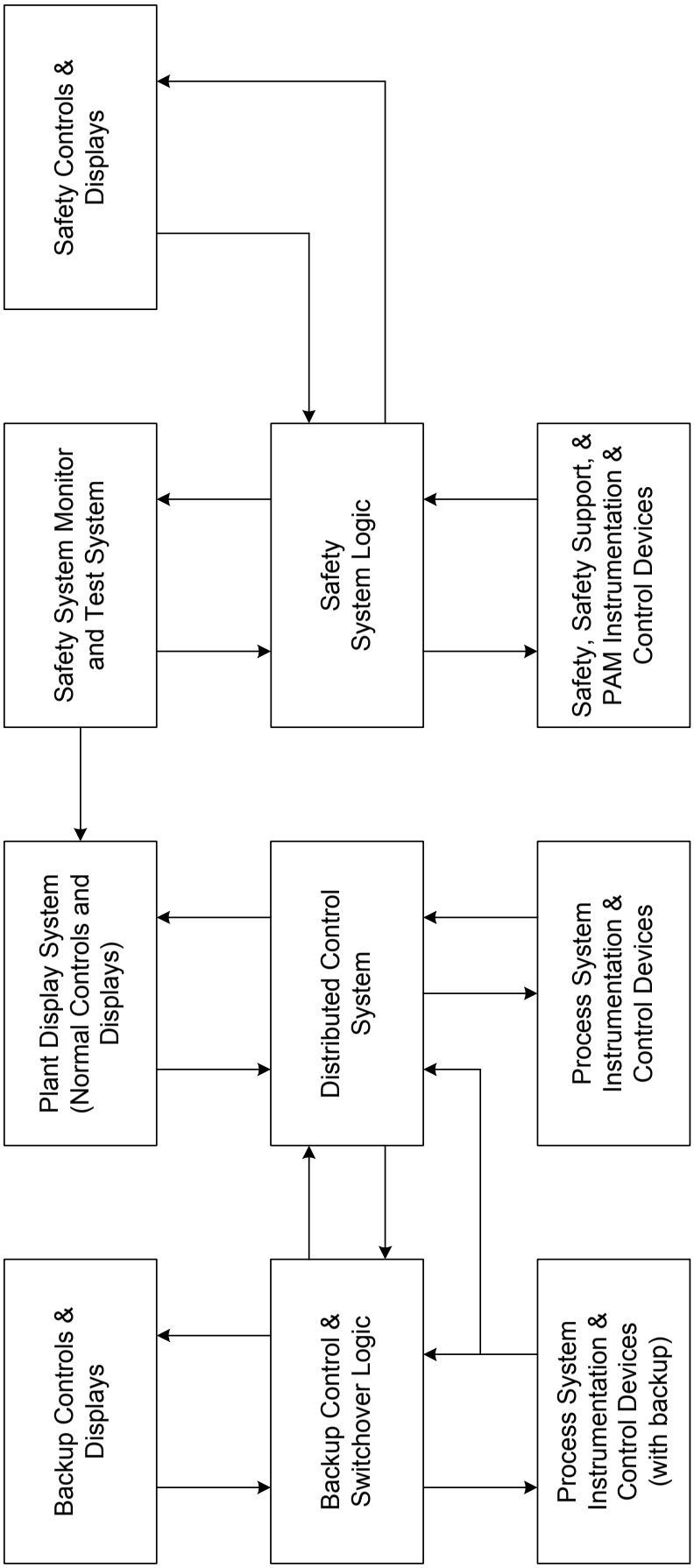

Figure 1 - Overview of the Various Control Platforms in a Plant Control System.

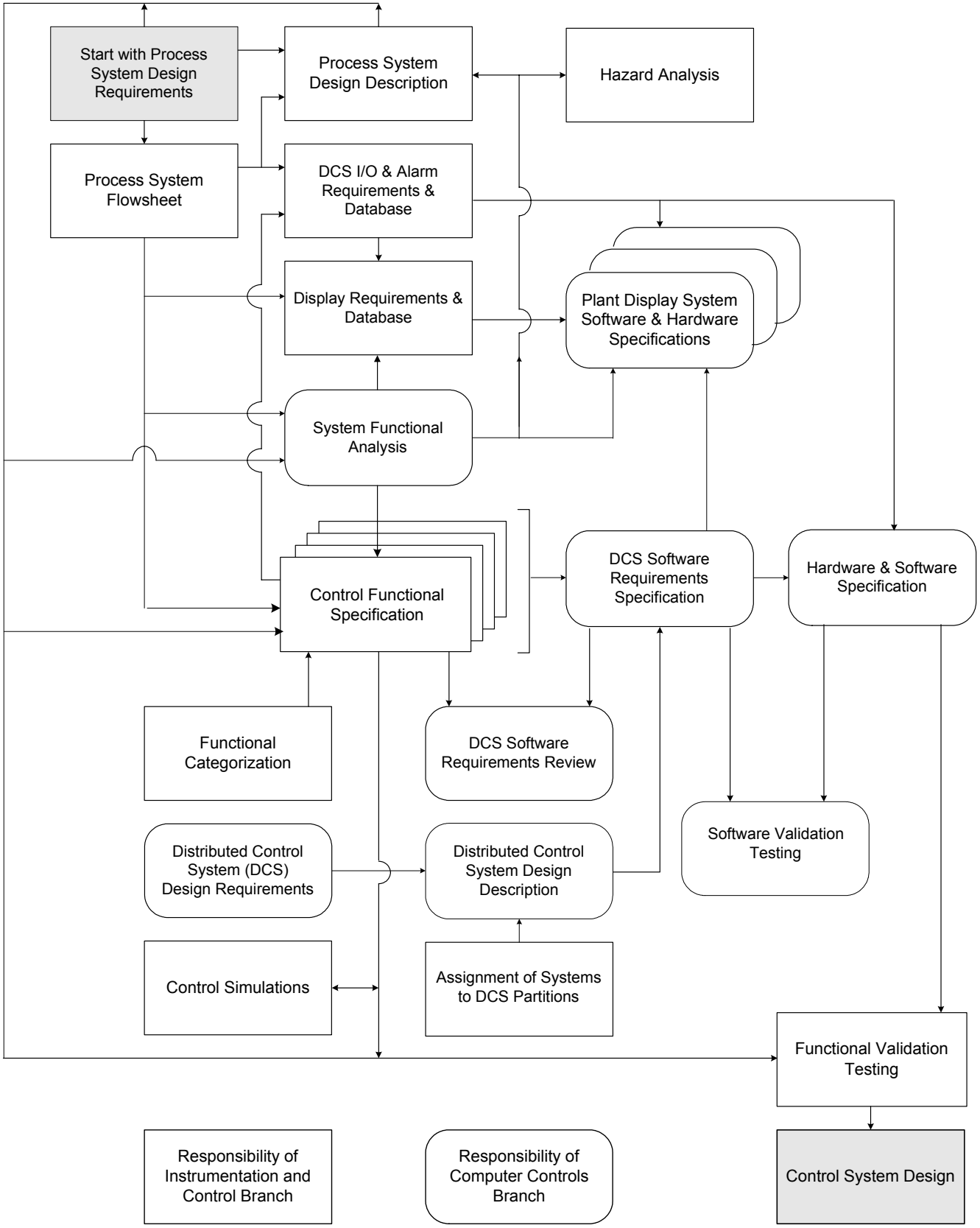

Figure 2 –Process Control Software Development Process

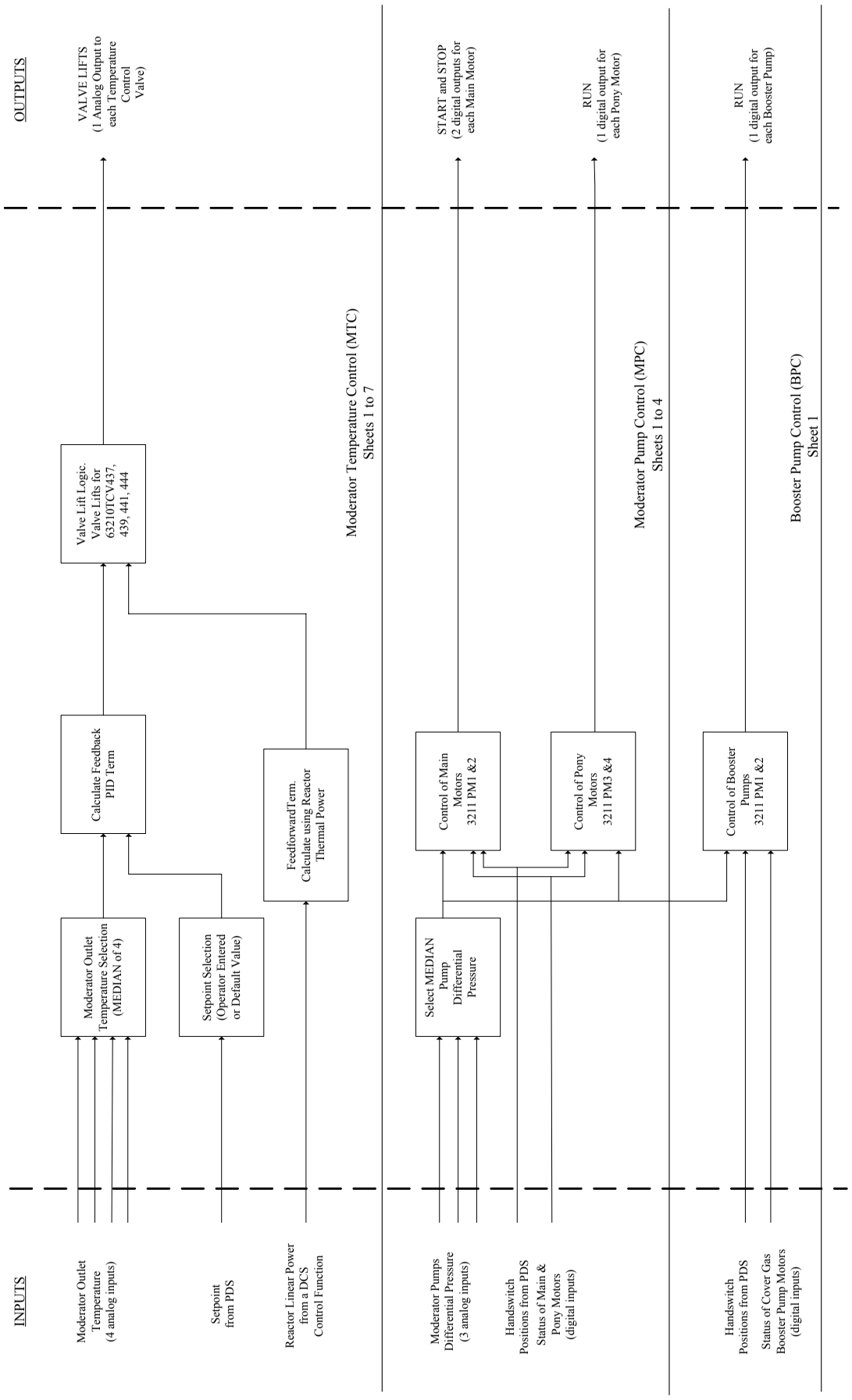

Figure 3 – Typical Control Functional Specification System Overview Diagram

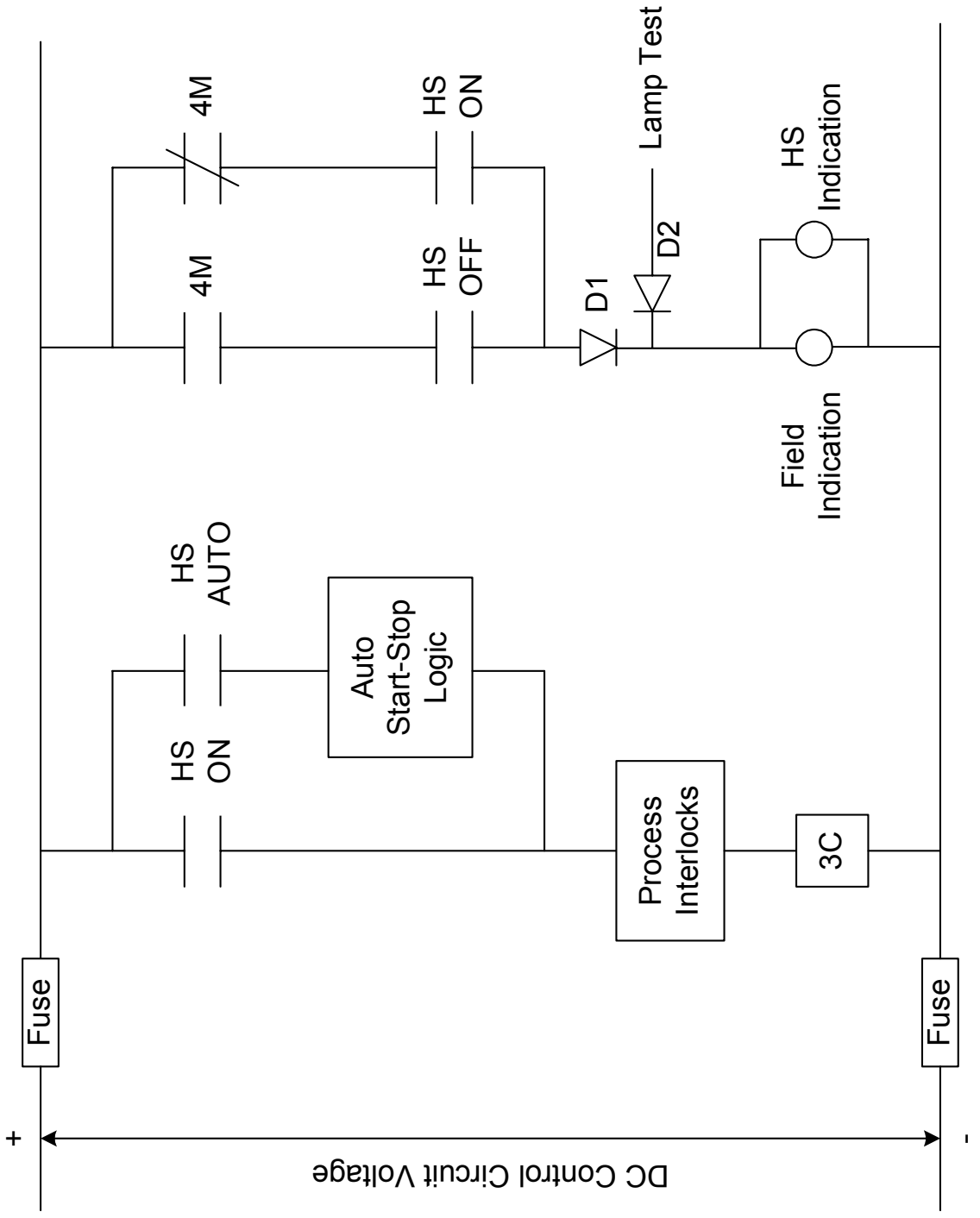

Figure 4 – Typical Relay Logic Standard for On-Off-Auto Control of an Electrical Load

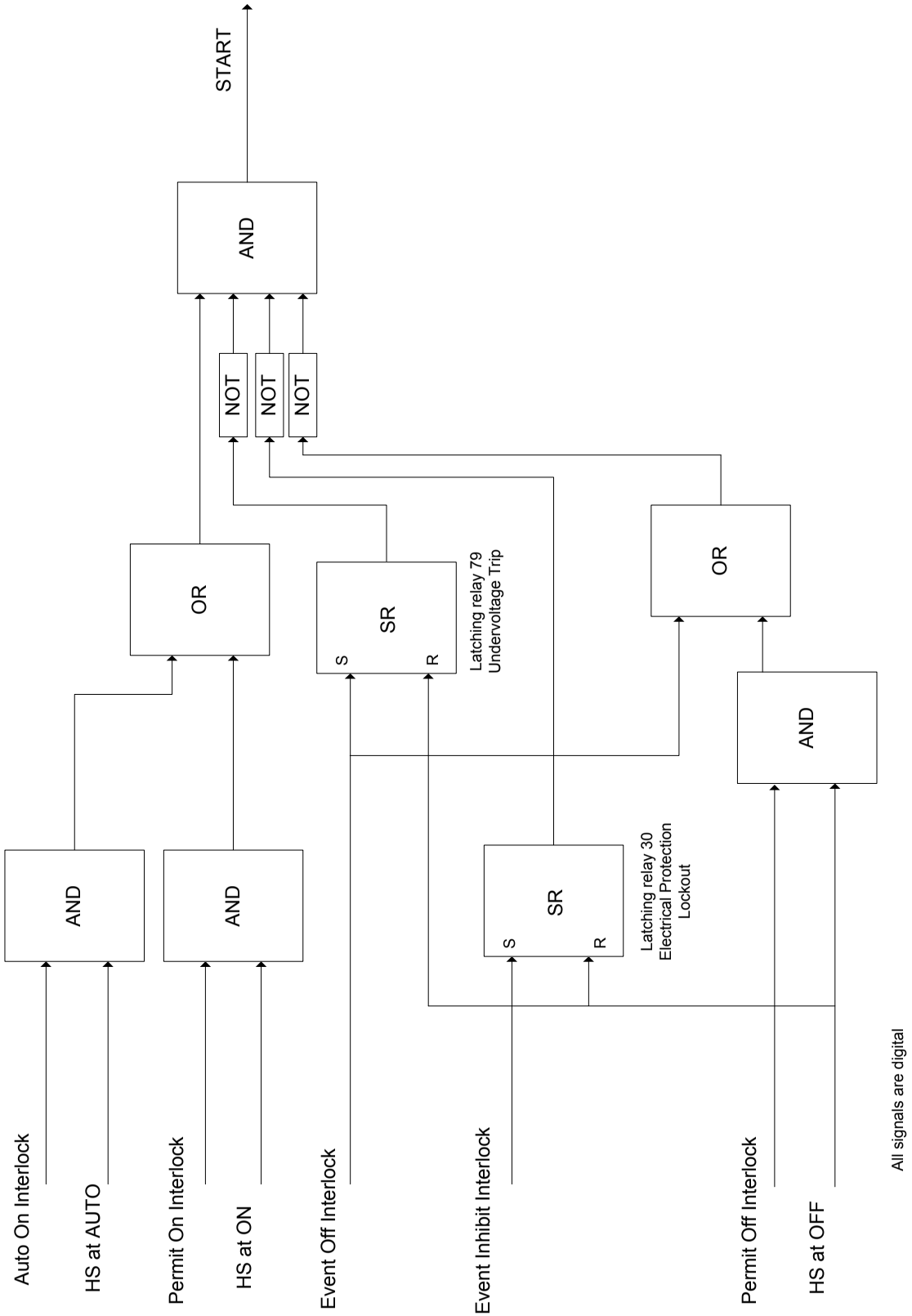

Figure 5 – Typical Standard Function for an On-Off-Auto Control of an Electrical Load