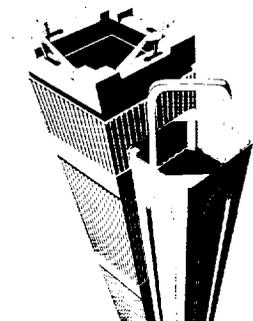


# SIEMENS

EMF-2110(NP)(A)  
Revision 1

## TELEPERM XS: A Digital Reactor Protection System

May 2000



Siemens Power Corporation  

---

Nuclear Division

**TELEPERM XS: A Digital Reactor Protection System**

Prepared: *J.F. Mallay / for* 9/2/99  
L. E. Erin, Manager  
Quality and Regulatory Affairs Date

Reviewed: *W.J. Catullo* 9/2/99  
*for* W. J. Catullo, Manager  
Engineering and Projects Date

Concurred: *D.A. Nauman* 9/2/99  
*for* D. A. Nauman, Director  
Strategic Operations and Business Integration Date

Approved: *J.F. Mallay* 9/2/99  
J. F. Mallay, Director  
Regulatory Affairs Date

arn/cel

**U.S. Nuclear Regulatory Commission  
Report Disclaimer**

**Important Notice Regarding the Contents and Use of This Document**

***Please Read Carefully***

This technical report was derived through research and development programs sponsored by Siemens Power Corporation. It is being submitted by Siemens Power Corporation to the U.S. Nuclear Regulatory Commission as part of a technical contribution to facilitate safety analyses by licensees of the U.S. Nuclear Regulatory Commission which utilize Siemens Power Corporation fabricated reload fuel or technical services provided by Siemens Power Corporation for light water power reactors and it is true and correct to the best of Siemens Power Corporation's knowledge, information, and belief. The information contained herein may be used by the U.S. Nuclear Regulatory Commission in its review of this report and, under the terms of the respective agreements, by licensees or applicants before the U.S. Nuclear Regulatory Commission which are customers of Siemens Power Corporation in their demonstration of compliance with the U.S. Nuclear Regulatory Commission's regulations.

Siemens Power Corporation's warranties and representations concerning the subject matter of this document are those set forth in the agreement between Siemens Power Corporation and the Customer pursuant to which this document is issued. Accordingly, except as otherwise expressly provided in such agreement, neither Siemens Power Corporation nor any person acting on its behalf:

- a. makes any warranty, or representation, express or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this document, or that the use of any information, apparatus, method, or process disclosed in this document will not infringe privately owned rights;

or

- b. assumes any liabilities with respect to the use of, or for damages resulting from the use of, any information, apparatus, method, or process disclosed in this document.



**UNITED STATES  
NUCLEAR REGULATORY COMMISSION**

WASHINGTON, D.C. 20555-0001

May 5, 2000

Mr. James F. Mallay  
Director, Nuclear Regulatory Affairs  
Siemens Power Corporation  
2101 Horn Rapids Road  
Richland, WA 99352

**SUBJECT: ACCEPTANCE FOR REFERENCING OF LICENSING TOPICAL REPORT  
EMF-2110(NP), REVISION 1, "TELEPERM XS: A DIGITAL REACTOR  
PROTECTION SYSTEM" (TAC NO. MA1983)**

Dear Mr. Mallay:

The NRC staff has completed its review of Topical Report EMF-2110(NP), Revision 1, "TELEPERM XS: A Digital Reactor Protection System," submitted by Siemens Power Corporation (SPC) on September 1, 1999. Revision 0 of the topical report was submitted on September 23, 1998, and Revision 1 incorporated the resolution of all comments received from the NRC on the review of Revision 0.

On the basis of our review, the staff finds the subject report to be acceptable for referencing in license applications to the extent specified, and under the limitations delineated in the report, and in the enclosed safety evaluation (SE). The SE defines the basis for NRC acceptance of the report.

Pursuant to 10 CFR 2.790, we have determined that the enclosed SE does not contain proprietary information. However, we will delay placing the SE in the public document room for a period of ten (10) working days from the date of this letter to provide you with the opportunity to comment on the proprietary aspects only. If you believe that any information in the enclosure is proprietary, please identify such information line by line and define the basis pursuant to the criteria of 10 CFR 2.790.

The staff will not repeat its review and acceptance of the matters described in the report, when the report appears as a reference in license applications, except to assure that the material presented is applicable to specific plant involved. Our acceptance applies only to the matters described in the report.

In accordance with the procedures established in NUREG-0390, the NRC requests that SPC publish accepted versions of the report, including the safety evaluation, in the proprietary and non-proprietary forms within 3 months of receipt of this letter. The accepted versions shall incorporate this letter and the enclosed evaluation between the title page and the abstract. The accepted versions shall include an "-A" (designating accepted) following the report identification symbol. The accepted versions shall also incorporate all communications between SPC and the staff during this review.

James F. Mallay

- 2 -

May 5, 2000

Should our criteria or regulations change so that our conclusions as to the acceptability of the report are no longer valid, SPC and the licensees referencing the topical report will be expected to revise and resubmit their respective documentation, or to submit justification for the continued effective applicability of the topical report without revision of their respective documentation.

Sincerely,

A handwritten signature in black ink, appearing to read "S.A. Richards", with a stylized flourish at the end.

Stuart A. Richards, Director  
Project Directorate IV and Decommissioning  
Division of Licensing Project Management  
Office of Nuclear Reactor Regulation

Project No. 702

Enclosure: Safety Evaluation



**UNITED STATES  
NUCLEAR REGULATORY COMMISSION**  
WASHINGTON, D.C. 20555-0001

**SAFETY EVALUATION BY THE OFFICE OF NUCLEAR REACTOR REGULATION**

**SIEMENS POWER CORPORATION**

**TOPICAL REPORT EMF-2110(NP), "TELEPERM XS: A DIGITAL REACTOR**

**PROTECTION SYSTEM"**

**PROJECT NO. 702**

**SUMMARY**

This safety evaluation provides the results of the NRC staff's review of Topical Report EMF-2110 (NP), Revision 1, "TELEPERM XS: A Digital Reactor Protection System" and accompanying proprietary documents. The staff also audited the TELEPERM XS (TXS) design implementation and associated documentation at the Siemens' office in Erlangen, Germany, in December 1999, and the results of the audit are also included in this safety evaluation. Based on the information provided and the review conducted, the staff concludes that the design of the TXS system is acceptable for safety-related instrumentation and control (I&C) applications and meets the relevant regulatory requirements.

The TXS system is a distributed, redundant computer system. It consists of three or four independent redundant data-processing automatic paths (channels), each with two or three layers of operation and running asynchronous with respect to each other. Layers of operation include signal acquisition, data processing, and actuation signal voting. In addition to the computers associated with the automatic paths, there are two redundant message and service interface computers to interface with each channel. The design provides for manual trip capability at the system, and component level, independent of the TXS system computers. Isolation and interaction between Class 1E and Non-Class 1E equipment is accomplished through end-to-end fiber optic cables found acceptable in previous license applications in the United States.

The TXS system architecture basic building blocks can be grouped into four categories:

1. **System hardware** - The TXS selected hardware platform uses a processing computer module, which includes random access memory for the execution of programs; flash erasable programmable read only memory for storing program code; and electrical erasable programmable read only memory for storing application program data.
2. **System software** - The TXS consists of a set of quality-controlled software components. The execution of the software centers around the operating software system which was developed by Siemens, specifically for the TXS systems. The operating system communicates with the platform software, and application software. The platform software includes the runtime environment program that provides a unified environment for execution of the function diagram modules.

3. **Application software** - The application software performs the plant-specific TXS safety-related functions using function block modules which are grouped into function diagram modules. The application software is generated by SPACE tools which use the qualified software modules from the function block library to construct a specific application.
4. **SPACE tool** - The SPACE (specification and coding environment) tool is an engineering system that is used to implement the requirements of plant-specific I&C features.

The German nuclear licensing authority contracted with the German Reactor Safety Association (GRS) to perform product certification. The GRS contracted with two German organizations to review, inspect, and certify products associated with the TXS system equipment and software.

The TXS system equipment qualification, and software verification and validation included temperature and humidity tests, seismic tests, electro-magnetic interference/ radio frequency interference qualification, and software type testing.

Equipment qualification was performed through type testing according to German safety standards which were compared by the staff against the U.S. nuclear industry equipment qualification standards. Except for minor deviations between German and U.S. standards, the equipment qualification design was considered to be acceptable. These deviations will be resolved by Siemens and evaluated by NRC during the review of plant-specific applications.

The process for independent verification and validation (IV&V) of the software is consistent with the IV&V process followed in U.S. standards.

The Siemens development process for the software was audited by the NRC staff at the vendor site. The NRC staff conducted a life cycle process audit of the TXS by tracing selected requirements through the software life cycle.

The design principle for software of Class 1E systems is to ensure that the sequence of processing executed for each expected situation can be deterministically established. It discourages the use of non-deterministic data communications, non-deterministic computations, multitasking, dynamic scheduling, use of non-deterministic interrupts and event driven designs. Based on its review, the NRC staff determined the design of the TXS system satisfies this design principle for Class 1E system software.

Concern with common-mode failures in digital systems and defense-in-depth were the subject of SECY-91-292 and positions addressing those concerns were established and documented in the Standard Review Plan (SRP) Chapter 7, Branch Technical Position (BTP) HICB-19. BTP HICB-19 set forth two principle factors for defense against common-mode/common-cause failures: quality and diversity. Maintaining high quality increases the reliability of both individual components and complete systems. The NRC staff has reviewed the TXS qualification and software development life cycle process and determined that TXS has the required quality.

In regard to defense-in-depth and diversity (D-in-D&D), the NRC staff has established acceptance guidelines for D-in-D&D assessment and has identified four echelons of defense against common-mode failures: control systems, reactor trip system, engineered safety feature actuation system, and monitoring and indication system. These guidelines are presented in

BTP HICB-19. The generic methodology proposed by Siemens follows the guidance in BTP HICB-19. Applications following this methodology for a plant-specific D-in-D&D assessment should be found acceptable in this area.

The TXS design is intended to provide a qualified generic digital I&C platform that meets the regulatory requirements and that can be used for a wide range of plant-specific applications. When using this platform for any plant-specific application, the licensee or applicant will need to verify that the qualification details in this topical report meet the plant license requirements. Because this topical report is for a generic platform, licensees referencing this topical report will need to document the details regarding the use of TXS design in plant-specific applications and address all plant-specific interface items including, but not limited to, those listed in Section 6.0 of this safety evaluation.

## 1.0 INTRODUCTION

By letter dated September 23, 1998, Siemens Power Corporation (Siemens) submitted non-proprietary Topical Report EMF-2110(NP), "Topical Report for the Generic Approval of TELEPERM XS Equipment at the United States Nuclear Regulatory Commission," for staff review. By letter dated September 1, 1999, Siemens submitted Revision 1 of non-proprietary topical report EMF-2110(NP), for staff review, changing the title of the topical report to "TELEPERM XS: A Digital Reactor Protection System." Revision 1 included major editorial and format changes focusing on regulatory aspects described in the U.S. NRC Standard Review Plan (SRP), NUREG-0800, Chapter 7, "Instrumentation and Controls," Revision 4, June 1997.

The TXS is a digital I&C system designed to be used in safety-related I&C applications in nuclear power plants as replacements for or upgrades to analog I&C systems. Typical applications include the reactor protection functions and the engineered safety features (ESF) functions. The non-proprietary topical report EMF-2110 (NP) describes the TXS hardware and software design, qualification testing, and application capabilities. In addition, Siemens submitted the following proprietary documents:

1. Letter NRC:99:037, dated September 1, 1999, "Supporting Documentation for Review of EMF-2110 (NP) Revision 1, TELEPERM XS: A Digital Reactor Protection System"
2. Letter NRC:99:052, dated December 16, 1999, "ERPI and QA Documentation Supporting Review of EMF-2110(NP) Revision 1, TELEPERM XS: A Digital Reactor Protection System"
3. Letter NRC:99:056, dated December 28, 1999, "Additional Information in Support of the TELEPERM XS Review"
4. Letter NRC:00:004, dated January 13, 2000, "Additional Information in Support of the TXS Review"
5. Letter NRC:00:007, dated January 19, 2000, "Additional Information in Support of the TXS Review"

6. Letter NRC:00:008, dated January 25, 2000, "Clarification of Selected Design Aspects of the TXS System"
7. Meeting presentation view graphs dated October 14 and 15, 1999
8. Meeting presentation view graphs dated November 16, 17, and 18, 1999
9. Letter NRC:00:017, dated March 3, 2000, "Clarification of EMF-2341(P), Generic Strategy for Periodic Surveillance Testing of TXS System in U.S. Nuclear Generating Stations"

These documents provide additional information to support the review of the design details of the TXS system. The staff's review of this topical report is generic. Some plant-specific review items that are not addressed in this topical report will need to be addressed and resolved during a plant-specific application review. Those plant-specific items are identified in Section 6.0 of this report.

This safety evaluation follows the guidance of the U.S. NRC Standard Review Plan (SRP), NUREG-0800, Chapter 7, "Instrumentation and Controls," Revision 4, June 1997. Chapter 7 provides guidance to the staff on reviewing complete nuclear power plant designs of the I&C systems. Revision 4 to SRP Chapter 7 also includes review criteria for digital systems.

## 2.0 SYSTEM DESCRIPTION

TXS is a distributed, redundant computer system. It consists of three or four independent redundant data-processing paths (channels), each with two or three layers of operation and running asynchronous with respect to each other. Layers of operation include signal acquisition, data-processing, and actuation signal voting. The communication between redundant channels uses end-to-end fiber optic cable connections.

The signal acquisition layer in each channel acquires analog and binary input signals from sensors in the plant (such as for temperature, pressure, and level measurements). Each signal acquisition computer distributes its acquired and preprocessed input signals to the data-processing computers in the next layer. Thus, each data-processing computer is provided with the same set of input information.

The data-processing computers perform signal processing for plant protective functions such as signal online validation, limit value monitoring and closed-loop control calculations. The data-processing computers then send their outputs to two independent voter computer units.

The signal on-line validation uses a 2<sup>nd</sup> minimum (or 2<sup>nd</sup> maximum) principle. For a redundant measurement system, each protection channel uses the 2<sup>nd</sup> lowest measurement to compare the low setpoint value and then determines the partial trip status of that channel for a "low trip" parameter. Similarly, it uses 2<sup>nd</sup> highest measurement to compare the high setpoint value and then determines the partial trip status of that channel for a "high trip" parameter. This method will reject the outlying signal in the process measurement and thereby minimize inadvertent trips.

In the voter computers, the outputs of the data-processing computers of redundant (three or four) channels are processed together. A voter computer controls a set of actuators. Each voter receives the actuation signal from each of the redundant data-processing computers. The voter's task is to compare this redundant information and compute a validated (voted) actuating signal, which is used for actuating the end devices.

The actuation logic employs either a TXS relay voter or a TXS digital voter. The reactor trip signals are voted by relay voters and the ESF actuation signals are voted by TXS digital voters, or TXS relay voters depend on the plant ESF system interface condition. The TXS relay voter votes redundant trip signals, one from each TXS channel set, with a simple 2-out-of-4 logic. The relays are duplicated for Train A and Train B. For each relay an additional contact is wired to the TXS monitoring and service interface (MSI) as a relay check-back signal. This is used for test and monitoring purposes. The digital voter performs 2-out-of-4 logic voting of the actuation signals from the processing computers. Each train has two voter computers.

Each TXS digital voter uses a pair of master-checkers in the voting logic to ensure there are no spurious actuations of safety-related equipment. Each master-checker set consists of redundant processors that process the same input signals. The results of the processing are compared, and differences in the results are flagged as possible errors in the processing that developed the voter input signal or in one of the processors that performed the voting operation. Since the master-checker redundant processors must use the same input data, the processors run synchronously, unlike the asynchronous processor operations between any two of the characteristic four channels of safety functions. If the processor outputs do not agree, the master-checker pair selects the default fault state, in which the output signals are set to 0 and the load power supply is disconnected. This use of the master-checker pair ensures that failures of a processor will not result in a spurious initiation of a safety function and that a master or checker processor will not disable the protection function in that channel.

For fail-safe and fault-tolerant outputs to the reactor switchgear, the TXS uses a voter configuration. The voter configuration consists of two sets of master-checker pairs in each channel, with a separate power supply. This configuration ensures that random signal failures only affect one half of a voter. Because the output signal of the two halves of the voter are applied to a hardware "OR" element, the second half of the voter assumes control of the switchgear in the event of such a failure. Both master-checker pairs in the two halves of the voter and interaction of the halves of the voter operate synchronously. The voter configuration ensures that single failures can result neither in spurious signals nor in loss of function.

In addition to the computers of the above-described automatic path, there are two redundant MSI computers to interface with each channel. The MSI computers are connected to each automatic path and to its assigned voters. The MSI serves as a gateway between the computers of the automatic path and other non-safety-related systems such as service units, process control computers, and monitoring computers. Either MSI unit can perform interfacing and diagnostic functions. The safety protection system signal passes through the MSI to display information at the main control board. The non-safety-related service unit requests access through the MSI to perform the diagnostic function at the safety-related processor. The service unit can be implemented as a single computer system or as a distributed system with several workstation computers. It can be installed temporarily or permanently. Parallel operation of several workplaces is possible.

The service unit contains the central data of the I&C system. It is the central means for interventions into the safety-relevant software of the function processors. The test machine for conducting periodic surveillance testing of the I&C system has a bus interface with the service unit. The software for the test machine operates as a client of the service unit. The service unit is protected against unauthorized interventions. The control mechanisms are installed by software so that only authorized persons may access the service unit, only authorized interventions may be performed, and interventions are restricted to a single redundant channel at a time. All signaling messages cyclically transferred by the service unit are recorded, checked for changes, and archived by the service monitor assigned to the service unit computers.

Manual reactor trip capability is provided on a per train basis. The manual trip signal at the system level totally bypasses the TXS processing and voting computers. Manual controls of safety actuators also bypass the TXS processing units and go directly to individual components via the priority logic which is located in the output of the voting units for the ESF actuation systems, and via the relay logic, which is independent of the voting computers, for the reactor trip system.

The TXS system architecture basic building blocks can be grouped into four categories:

1. SPACE tools - The SPACE tool is an engineering system that is used to implement the requirements of plant-specific I&C features.
2. System software
3. System hardware:
  - a. Hardware structure including backplane bus
  - b. Processing modules
  - c. Communication systems
  - d. I/O modules
4. Application software - The application software is generated by SPACE tools which use the qualified software modules from the function block library to construct a specific application. The programs will be stored in FEPRM (flash erasable programmable read only memory).

To create a specific application project, the first step is to define the hardware specification, which contains the complete hardware structure of the target system with all of its components. The hardware specification is created using the SPACE editor. The SPACE editor is a graphical user interface tool to create I&C function diagrams and hardware diagrams. Each function diagram is assigned to one processing module, on which it is processed. This assignment is done while creating the hardware diagrams. The information is stored in the specification database.

SPACE code generators are used to interpret the contents of the specification database and to automatically generate high-level language code (in C language) for each function diagram. Communication between function diagrams is done using data messages. These are also automatically generated by interpreting the hardware specification and the software-hardware

assignment. Thus the complete code for all function diagrams is automatically generated. Automatic code generation reduces the probability of coding errors and reduces coding time. Independent tools are developed to perform automatic code verification. The SPACE tools parse the generated code, transform it into an internal representation, and compare this representation to the information stored in the specification database.

Specific communication methods are applied to ensure interference-free communication inside the TXS system and within the plant process information system. The TXS design requires that in case of a single failure of one of the independent processing channels or within one communication path in the same processing channel, the channels still available will continue to operate as designed on the basis of the remaining information to ensure the required safety functions do not fail. The communication from the safety I&C system to the plant process information system is done via the MSI computer as previously explained. This communication channel is used by signaling messages to the plant process system. The MSI serves as a means of isolation within the TXS architecture. The link through the MSI gateway computer is configured so that the faults in the non-safety-related I&C systems cannot affect the operation of the safety-related I&C system.

An important aspect of the system software is the functioning of the runtime environment (RTE) which is essential for the TXS communication. The internal system clock (every millisecond) triggers and controls all actions during the processing cycle. The central control unit sequentially starts the main processing phases in each processing cycle (typical cycle time is 50 ms). A typical real-time processing cycle starts in the following sequence:

1. Reading input data,
2. Input checks of received messages,
3. Processing application software,
4. Handling of transmitted messages,
5. Transferring the output messages,
6. Processing of diagnostic programs for the remaining processing cycle time, and
7. Processing self monitoring programs.

The TXS CPU features a hardware watchdog timer which has an independent clock. This watchdog is triggered by the cyclic task of the RTE. On the beginning of each computing cycle the watchdog timer is set. If the watchdog is not triggered by the RTE in time, it times out and activates the exception handler. Thus, the hardware watchdog monitors the RTE cyclic task.

The RTE includes a feature that monitors the internal counter of the cyclic self-monitoring task. The cyclic self-monitoring function keeps an internal counter, which is incremented once in every complete cycle of the cyclic self-monitoring. This timer is triggered by the self-monitoring task when the cycle is started. If this counter has not been changed (incremented) within a specified period of time, the RTE issues an error message. The time needed for a complete cycle of self-monitoring depends on the CPU load. When more of the cycle time is used by application software, less time is available for cyclic self-monitoring. However, the complete cycle is typically carried out in 300 to 600 seconds.

The detailed hardware and software descriptions are discussed in Sections 2.1 and 2.2 of this safety evaluation.

## 2.1 Hardware Description

The TXS-selected hardware platform uses a processing module, which includes Random Access Memory (RAM), Flash Erasable Programmable Read Only Memory (FEPROM) for storing program code, and Electrical Erasable Programmable Read Only Memory (EEPROM) for storing application program data. Input and output modules are standard components designed for an automation system; these modules have been in service in other technological fields for many years and in many applications. Communication processing modules are available for the Local Area Network (LAN) standard Ethernet and Profibus (process field bus). These boards are mounted in racks and communicate via a 32-bit multi-master-capable parallel backplane bus. Other LAN components such as electrical-to-optical transducers and star coupler components are also industry standard components.

The hardware-associated interrupts are the following:

**Time 0 interrupt:** This interrupt is generated by a hardware timer once every 1 ms. It is used as the operating system's time interrupt. This interrupt is handled by the MICROS operating system. The interrupt service routine increments the internal operating system time by 1. Then, it enters the scheduler and resumes the execution of the task previously interrupted which is normally associated with an application program phase under the control of the processing cycle, unless the application program cyclic task has completed. All the safety-related application programs are assigned the same high priority to ensure that once the execution of a program starts, it goes to completion without being interrupted by another application program. The only task assigned higher priority than the application program is the MicroNet Interrupt Service task which can only be activated by the LBUS hardware interrupt. The LBUS hardware interrupt does not occur during normal operation.

**K32 backplane bus interrupt (IRE):** This interrupt is generated by another CPU via a memory mapped write access. If a CPU writes to a certain memory address, an IRE interrupt can be generated on another CPU. The IRE interrupt is handled by the MICROS operating system. It is configured so that MICROS will start a predetermined task when the IRE occurs. This interrupt is used in TXS in two cases:

1. If a CPU wants to send data via a communication processor, this CPU initiates an IRE interrupt on the communication processor. The IRE activates a task on the communication processor, which then looks for new data to be sent and sends them via its network interface. Using this technique, the time delay on the communication processor is minimized. The sending of data is a periodic process controlled by the RTE's cycle task.
2. The IRE is also used on voter master/checker pairs. Here the IRE is used to start the checker's cyclic task, thus ensuring synchronous operation of master and checker. This works in the following way: On the master, the RTE's cyclic task is started by the MICROS operating system based on the operating system's time (for example every 50 ms). Once the master's cyclic task is running, one of its first actions is to send dummy data to his checker via the backplane bus initiating an IRE interrupt on the checker. The data are not relevant here, they are only sent to initiate the IRE on the checker. On the checker, the IRE is handled by the MICROS operating system, which is configured in a way that the checker's RTE cycle task will be activated. The master's RTE cycle task is

started periodically (typically every 50 ms). Likewise, the IRE on the checker will occur with the same frequency.

**LBUS interrupt:** This interrupt can be generated by subsystems connected to the processing module (SVE1) local extension bus (LBUS). It is generated by a subsystem when it is reset or powered up. This interrupt is handled by the MICROS operating system. When it occurs, MICROS starts a task which notifies the communication software MicroNET, that a subsystem has restarted. By this method, MicroNET is able to re-establish the communication channels to communication partners in this subsystem (if any). This interrupt can only occur if subsystems are connected to the SVE1 processor. Even in this case, the interrupt only occurs when the subsystem is reset or powered up, which is not the case during normal operation.

**LAX-module interrupt:** This interrupt is generated by the LAX-module (Ethernet interface board) of the communication processor (SCP1). The interrupt is generated by the Ethernet controller on the LAX board when a frame is received from the Ethernet network. The interrupt is handled by the SCP1 protocol handler firmware. This interrupt is only used on the SCP1 communication processors.

All other interrupts are either disabled or do not occur during normal operation. All of these hardware interrupts have been designed for strictly deterministic behavior.

### 2.1.1 Physical Description

TXS hardware basically consists of four types of components: the subracks, function processors, communication modules, and input/output (I/O) modules. These basic components can be configured to constitute a digital safety I&C system to replace an existing analog safety I&C system. The new configured digital safety I&C system may be located in same place as the existing cabinet and may utilize the existing field cables for input and output signals. The existing channel separation will be maintained.

#### Subrack

The subrack contains the electronic printed-circuit boards (PCBs). Cooling fans are energized with Class 1E power. Fan operation is monitored and fan failures resulting in an excessive temperature inside the cabinet will be monitored and alarmed. The subrack is equipped to cool the modules and protects them from electromagnetic interference.

#### Function Processor

The function processor module is the application programmable module for executing the safety functions. The I&C functions are stored as executable programs in the write-protected FEPRM along with the necessary system functions, and are validated by cyclic redundancy checks (CRCs). An executable program is a sequence of executable commands executed by the CPU. All components of a function processor are always used in the same way and in a recurring sequence. Data and programs that are required to implement I&C functions are permanently assigned to defined memory locations in advance.

The input signals for I&C functions that are implemented on a function processor are provided either in the form of messages via the MSI or as single-wire signals via I/O modules. Similarly,

the output signals of the I&C functions are passed on either as messages or as single-wire signals. Signals from the I/O module are read by the function processor from the data buffers of the I/O modules via the backplane bus or written to the data buffers. Data exchange with the MSI passes through the dual-port RAMs. Physically different areas of the dual-port RAM are dedicated to receiving and sending data.

### Communication Modules

Communication is based on serial buses. Data transmission is performed in the following steps:

- The function processor writes the data to be transmitted into the dual-port RAM of the interface module.
- The data are serially transmitted via the network to the interface module of the destination system in accordance with protocol used.
- The interface module transmits the data to the dual-port RAM of the destination function processor.
- The destination function processor reads the data and checks the data integrity.

### Input/Output Modules

All modules are plug-in PCBs. Analog modules contain an analog-to-digital or digital-to-analog converter and a multiplexer. Each module is capable of handling eight input variables. Digital modules can handle 32 inputs. The analog input data is stored in dedicated memory locations for each input. This data is read by the function processor. The digital input data is directly read by the function processor from the digital module.

#### 2.1.2 Product Qualification

Equipment qualification was performed through type-testing according to German safety standard KTA-3503, "Type Testing of Electrical Modules for the Reactor Protection System." The hardware type tests began in 1993 and ended for the first set of hardware modules in 1996. The results of the type tests were documented by certificates and associated evaluation reports. Each qualified component has its own certification and its own evaluation report. If a certified product requires a modification, the modified product is required to have a new certification.

The German nuclear licensing authority contracted with the GRS to perform product certification. The GRS contracted with the German Technical Inspection Agency (TÜV, Technischer Überwachungs Verein) to review, inspect and certify products. The TÜV verifies the consistency of the design documents and the requirements specified in the safety-related applications. The TÜV also inspects the design process and test results. In some cases TÜV may perform independent tests to verify the performance of a product. When TÜV completes its review, it issues a product certification. The certified product can be used for subsequent applications.

The equipment qualification tests were performed in the following test sequence:

1. Visual inspections,
2. Functional tests,
3. Test of electrical characteristics,
4. Climatic tests (including temperature and moisture tests),
5. Test to demonstrate the electromagnetic compatibility conditions,
6. Mechanical stress tests (including seismic qualification test), and
7. Another functional test to verify no degradation of function.

The TXS system was developed in several stages. Qualification was implemented in four steps:

**Step 1:** The conceptual design was reviewed by GRS and TÜV, and the design was accepted in 1992.

**Step 2:** The following hardware components were type tested:

- Components for signal processing,
- Components for communication,
- Components for input and output of digital and analog signals, and
- Subracks and electrical accessories.

Third-party assessment of hardware type testing was completed in 1998. The test results were summarized in two test reports:

- (1) "Summary Test Report for Type Test of Modules in TXS," TÜV Nord, March 18, 1998.
- (2) "Documentation of the Practical Test - Overview of Test Documentation - Summary of Test Results of the TXS System," TÜV Rheinland, March 18, 1998.

**Step 3:** Type testing of software includes the following:

- The runtime environment,
- The function diagram group modules, and
- The function block modules.

The test procedure was closely linked to the different stages of software development. GRS was contracted to perform the third-party assessment for software type testing. The assessment began in 1992 and was finished in 1997.

**Step 4:** Quality verification of the program development tools for integrating the platform with plant specific application programs (such as SPACE system) was included in the software type testing. The quality verification confirmed that the tools are functionally and qualitatively suitable for performing their tasks. Quality verification was completed in 1997.

The staff requested that Siemens provide a comparison between the German type testing standard and the U.S. nuclear industry equipment qualification standards. By letter NRC:00:007 dated January 19, 2000, Siemens submitted report EMF-2352(NP), "TXS Qualification Testing." Elements of the TXS qualification testing program were evaluated against the requirements and acceptance criteria identified in the Electric Power Research Institute (EPRI) report TR-107330, "Generic Requirements Specification for Qualifying a Commercially Available PLC (programmable logic controller) for Safety-Related Application in Nuclear Power Plants," for seismic and environmental qualifications. The results of this assessment are summarized below.

### Seismic Qualification Testing

#### Applicable Standards:

- IEEE Standard 344-1975
- EPRI TR-107330, Figure 4.5

#### Acceptance Criteria:

EPRI TR-107330, Section 4.3.9

The I&C system shall operate as intended for the specified level of vibration. All connections in the system shall remain intact, all modules shall remain fully inserted, and no functional or non-functional parts shall fall off their specified levels. If relay output modules are included for qualification, then the relay contacts shall be capable of changing state from energized to de-energized and deenergized to energized during application of the operating basis earthquake (OBE) and the safe shutdown earthquake (SSE). Any spurious change of state shall not exceed 2 milliseconds for both energized and deenergized relays.

### Environmental Testing

#### Applicable Standards:

- EPRI TR-107330, Sections 4.3.6.1, 4.3.6.2, and 4.3.6.3
- EPRI TR-107330, Figure 4-4

#### Acceptance Criteria:

EPRI TR-107330, Sections 4.3.6.1, 4.3.6.2, 4.3.6.3 and 5.3

The I&C system shall operate for the temperature and humidity environmental profile provided in EPRI TR-107330, Figure 4-4, and the operability requirements stated in Section 5.3.

The seismic and environmental qualification tests performed on the TXS I&C system satisfied the majority of the test requirements specified in EPRI TR-107330. However, some deviations were noted as explained in Sections 2.1.2.1 and 2.1.2.2.

The staff has reviewed the documentation of the practical test results of the TXS system. The purpose of the practical test is to verify that the test objects (TXS modules) meet the

safety-related requirements and regulations specified in KTA-3503 and the requirements specified by the supplier. The tests were performed by TÜV Rheinland and TÜV Nord (both of the German Technical Inspection Agency) from March 1994 through January 1998. The test results were documented in TÜV reports 945/K 72999/98 and TXS-980318-PB, both dated March 18, 1998.

Based on the audit of these reports, the staff found that the TXS system hardware design has complied with German safety standards. However, in order to meet the generic qualification requirements for U.S. nuclear plants, the product should also meet the U.S. testing requirements specified in the EPRI report TR-107330 for the two issues discussed in the following sections. EPRI TR-107330 was approved by the staff on July 30, 1998.

The software qualification findings are presented in Sections 2.2.2.14, 2.2.2.15, and 4.4.

#### 2.1.2.1 Environmental Qualifications (Temperature and Humidity Tests)

A detailed compliance matrix showing the relationship between EPRI requirements and the TXS system design was documented in proprietary report TR-104017, "Siemens TXS Compliance with EPRI TR-107330." As stated in TR-104017, environmental testing was performed on a fully loaded representative TXS system, and the operation was under the control of generic application software. The objective of these tests was to prove the adequacy of the materials and system design under all conditions to which the system might be subjected from factory to final in-service operation. The following tests were performed:

- Steady-state cold with modules not in operation
- Steady-state dry heat with modules not in operation
- Steady-state damp heat with modules not in operation
- Temperature cycle test with modules not in operation
- Cyclical humidity with modules not in operation
- Steady-state damp heat with modules in operation
- Cyclical dry heat with modules in operation

As stated in the TR-104017, comparing the TXS testing condition and the EPRI testing requirements, the TXS tests did not achieve the maximum temperature in the EPRI testing requirements. An environmental qualification retest of the TXS was planned to meet the EPRI requirements. The above tests were performed without the cabinet enclosure. The absence of a cabinet enclosure did not allow for internal cabinet temperature effects and confirmation of the effectiveness of the cooling configuration. The TXS design has a provision to monitor temperature inside the cabinet. The plant-specific application should identify in the plant operating procedures monitoring internal cabinet temperature to ensure that the internal cabinet temperature will be always under the environmental qualification envelope, and develop plant-specific procedures to respond to TXS cabinet/subrack high temperature alarms. This is a plant-specific action item.

#### 2.1.2.2 Seismic Qualification

To demonstrate that the operability of the component is not degraded by mechanical stresses, tests of mechanical stress were performed by Siemens. The seismic qualification test was one of the test procedures in hardware qualification tests. The seismic qualification test was based

on IEEE Standard 344-1987 criteria. The amplitude of acceleration for the test input accounts for the natural frequency (18 Hz), and the magnification factor for the supporting structure (1.56) of the I&C cabinets to be used, as well as the acceleration spectrum for the design of a typical PWR plant. The input excitation used was multiple frequency ranging from 5 to 35 Hz, and 3 axes, each staggered by 90°. Test duration per axis was a minimum of 1 minute. The functioning of the component was monitored during the test. The technical data were not violated. Following the test, a visual inspection and an intermediate functional test were performed. The test results were documented according to IEEE-344, Section 10.

However, as stated in TR-114017, the TXS seismic testing level is below the EPRI requirements. The TXS seismic test level does not completely envelop all U.S. nuclear plants required test levels. In order to be useful for most of the U.S. nuclear plants, a seismic qualification retest for the TXS system was planned to comply with the EPRI seismic testing requirements. A U.S. licensee that uses the TXS system for a safety system application should compare its required seismic qualification level to the Siemens' qualified level, and identify areas requiring further action. This is a plant-specific action item.

#### 2.1.2.3 Electro-Magnetic Interference (EMI) / Radio Frequency Interference (RFI) Qualification

EPRI submitted Topical Report TR-102323, "Guideline for Electromagnetic Interference Testing in Power Plants," for staff review in 1994. The topical report was developed by EPRI to recommend alternatives for performing site-specific electromagnetic interference (EMI) surveys for qualifying digital plant safety instrumentation and control equipment in a plant's electromagnetic (EM) environment. The recommendations in TR-102323 include (1) a set of electromagnetic interference and radio frequency interference (EMI/RFI) susceptibility testing levels, (2) EMI eliminating practices, and (3) equipment EMI/RFI emission testing levels. The above recommendations are based on EMI/RFI emission data collected during 1993 and 1994 at seven nuclear power plants and data collected before 1993 at other nuclear power plant sites.

In 1996, the staff issued a safety evaluation concluding that the TR-102323 recommendations and guidelines provided an adequate method for qualifying digital I&C equipment for a plant's EM environment without the need for plant-specific EMI surveys if the plant-specific EM environment is confirmed to be similar to that identified in TR-102323.

Siemens performed electromagnetic compatibility (EMC) qualification tests on the TXS components as described in the TÜV report TXS-980318-PB, "Documentation of the Practical Test Overview of Test Documentation-Summary of Test Results of TELEPERM XS System," dated March 18, 1998. The results of the EMC qualification tests show that the tested TXS did not satisfy all the test requirements specified in EPRI TR-102323. The main reason that the TXS did not meet all the test requirements specified in EPRI TR-102323 was the difference in the requirements of the European EMC test standard (used by Siemens) and the EPRI TR-102323 specification. Additionally, the configuration of equipment used for the EMC qualification tests might not be the most adverse EMC configuration of the TXS I&C system equipment. To resolve these two issues, Siemens stated that it will perform EMC qualification tests in accordance with EPRI TR-102323 on the most adverse EMC TXS I&C equipment configuration for the EMC qualification tests.

Based on this commitment, the staff finds that the TXS I&C system equipment EMC qualification would be considered acceptable provided that Siemens successfully completes the proposed EMC tests. Additionally, before installing the TXS system as a safety system in a nuclear power plant, a licensee should verify that its plant electromagnetic environment and its TXS system configuration are enveloped by EPRI TR-102323 EMC qualification tests. This is a plant-specific action item.

#### 2.1.2.4 Power Supply Quality Requirements

TXS applications in Europe were designed to operate with a DC power source providing input power to the TXS racks. However, most nuclear power plants in the United States provide an AC power source for Class 1E instrumentation and control systems. For these applications, Siemens will qualify an AC to DC power supply converter that complies with EPRI TR-107330 electrical power supply and qualification requirements. This is a plant-specific action item.

#### 2.1.3 Isolation and Interaction Between Class-1E and Non-Class-1E Equipment

In the TXS system design, signals interact between redundant Class-1E channels and transmit from Class-1E channels to non-Class-1E devices. The communication between Class-1E channels uses end-to-end fiber optic cables found acceptable in previous license applications in the United States. The communication from the safety I&C system to the non-safety plant information system is done via the MSI. The MSI serves as a means of isolation within the TXS architecture. For the upgrade of existing analog instrumentation and control systems in United States nuclear power plants, there is a need to provide an interface between Class-1E and non-Class-1E systems by means of both analog signal and relay contacts. For these applications, Siemens will qualify an analog isolation device and a mechanical relay to provide adequate coil-to-contact isolation. This qualification will be performed in accordance with the class 1E to non-Class-1E isolation requirements of EPRI TR-107330. This is a plant-specific action item.

### 2.2 Software Description

The TXS is a digital instrumentation and control system for safety-related applications in nuclear power plants. The system provides a framework in which engineers may design and implement plant-specific safety-related applications. Typical applications are closed-loop controls of reactor processes, reactor trip applications, and applications that initiate engineered safety features actuation signals. The TXS consists of a set of quality-controlled software components that are qualified according to the specific requirements of nuclear reactor safety systems. To control and facilitate development efforts, the TXS system includes a specification and coding environment (SPACE) tool for designing and assembling safety-related applications.

The conceptual architecture for a TXS application serves as the functional specification and design of a TXS plant-specific system, and serves as a bridge between the plant-specific system requirements and the resulting system implementation in software code.

The software and data that do not change are stored in the write-protected flash erasable programmable read only memory (EPROM) area of the function computer. Examples of this type of data are the runtime environment, function diagram group modules, and system parameters that do not change over the life of the system application. This data cannot be

changed without first erasing the data stored in the applicable flash EPROM 64 Kb sector, and then rewriting the entire flash EPROM sector. The flash EPROM data, however, may be changed without removing the flash EPROM from the SVE1 processor board. To ensure the application software and invariable data remain unchanged, flash EPROM data integrity is checked by a self-diagnostic routine that calculates the CRC value of each 64Kb sector of the EPROM and compares the result to the CRC value that is stored in each 64 Kb sector of the EPROM with the application software and invariable data.

Data that is subject to change over the nuclear plant fuel cycle are stored redundantly in electrically erasable programmable read only memory (EEPROM). Examples of this type of data include plant system parameters and setpoints that may require changing by the plant operator, programs required for SVE1 (processing module) startup, and the loader for programming the flash EPROM. Unlike the unchanging data and programs that are stored on flash EPROM, data stored in EEPROM may be changed without first erasing all the data stored in a block of memory in the EEPROM. In addition to checking the EEPROM data with a CRC diagnostic routine, this data is also checked using the redundant copies of the stored data.

Data that changes from one processing cycle to the next are stored in the random access memory (RAM). Examples of this data include plant process input data, intermediate results of the system applications, and output data. Data in the RAM may be changed without the special processes used to write data to flash EPROM and EEPROM. The integrity of the RAM used to store data is cyclically checked by a self-diagnostic routine that writes data to each RAM address and then reads the data to ensure the RAM address correctly stores information.

### 2.2.1 Teleperm XS (TXS) Software Description

The TXS software can be divided into three categories: operating system software, platform software, and application software. The following sections describe the software in these three categories. The components of each layer of the processing system are shown in Figure 1.

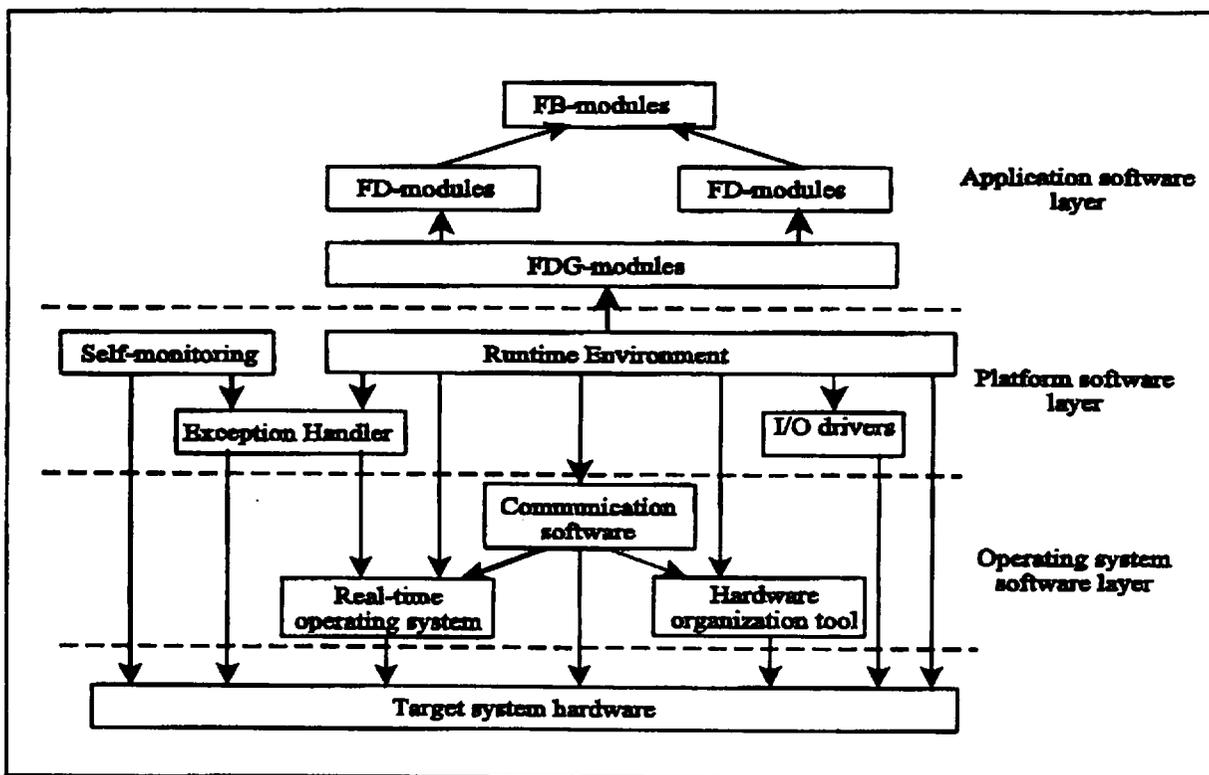


Figure 1. Software layers of one application processor.

### 2.2.1.1 Operating System Software

The operating system was developed by Siemens Plants and Technical Service Division ATD (Anlagen und Technische Dienstleistungen) specifically for the TXS systems. Development was according to TXS requirements and specifically to International Electrotechnical Commission Standard (IEC-880), "Software for Computers in the Safety Systems of Nuclear Power Stations." Standard IEC-880 is referenced in IEEE Standard 7-4.3.2, "IEEE Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations." IEC-880 is comparable to IEEE Standard 7-4.3.2 and NRC has found Standard IEC-880 acceptable.

The operating system software layer consists of the real-time operating system, communication software, and the hardware organization tool. The components that make up these three categories are summarized in the following discussion.

- **MICROS (Real-time Operating System)**

MICROS is the real-time operating system kernel of TXS systems. It is small (approximately 1 Kbyte) and static. In this context, the term static means that all the operating system objects are defined on startup and cannot be changed during run time. For example, the operating system does not use dynamic allocation of system resources such as memory-heap or dynamic task definitions.

- **NMI-Handler (Real-time Operating System)**

The NMI handler, in cooperation with the exception handler, is responsible for handling abnormal conditions. In many cases, hardware failures are signaled by means of interrupts. For example, communication bus accesses to I/O modules are monitored for tolerable response times (time-outs) in this way. If these times are exceeded, it is concluded that a failure has occurred and this is signaled by means of an interrupt. Both the NMI handler and the exception handler are responsible for initiating the required measures in such cases. One important measure involves signaling failure and returning the computer to a predefined state (e.g., disabling of all outputs).

- **MicroNET (Communication Software)**

MicroNET is responsible for the communication between different function computers, as well as between the communication processors and function computers within the same subrack. MicroNET-L2 is an extension of the communication services that supports L2 (protocol) communication.

- **CP486 (Communication Software)**

CP486 is the protocol handler implemented in the TXS communication processor. This software module is responsible for transmitting and receiving H1 (communication processor) messages.

- **Driver 3964 R (Communication Software)**

The 3964R driver constitutes the interface to the local V.24 interface which is used for loading programs and for the output of debug information.

- **Hardware Organization Tool**

The Hardware Organization Tool (HOT) is responsible for the parameterization of the operating system on computer startup. HOT configures the communication memories and detects whether modules are assigned to the slots in a subrack. The HOT also detects the module types.

In addition to the software modules described above, the TXS operating system provides the following services:

- **Debug Services**

Debug services support system diagnostic tasks used mainly during the system development and integration.

- **Diagnostics Monitor (monitor to support diagnostics)**

The diagnostics monitor is a low-level debugging tool. If the computer fails, for example, it is possible to branch to the diagnostics monitor to analyze causes of failure that are difficult to diagnose after the computer has been returned to the defined state (outputs disabled).

Two communication protocols are used in the TXS. Communications between subracks in the same safety channel are via a backplane bus using an H1 protocol (IEEE-802.3). Communications between different channels are over fiber optic connections using the Profibus L2 protocol (DIN 19245). All communications are on serial busses. Data transmission is always performed in the following manner:

- The function processor writes data to be transmitted into the interface module dual-port RAM.
- The data is transmitted via the applicable network and protocol to the interface module of the receiving function processor or system.
- The receiving interface module writes the data into the receiving function processor dual-port RAM.
- The receiving function processor reads the data from its dual-port RAM and checks the data integrity.

Communication between different processors is done by messages. These messages can contain signals from function diagram group (FDG) modules (data messages); control commands from the service unit (control messages); or error messages, trace data, or command responses to the service unit (signaling messages).

Communications cycle between function processors in the same channel and communications between redundant safety channels. That is, each function processor sends one signaling message and receives one command message per processing cycle. Communications via shared dual-port RAM use a handshake protocol: the sender can only send into the channel when it is empty, and the receiver can only read from the channel when it is filled. All messages have an individual, but fixed message length. This ensures that the load on the communication busses is constant and the communications occur in a deterministic manner.

Messages are also transmitted from and to the service unit via the MSI, which organizes the exchange of this information with the function processors. Each MSI can service up to 10 function processors. The cyclic communications are invariant and, therefore, allow the TXS safety system to be fully tested during the system integration phase of the system life cycle.

As discussed above, in addition to communications between subracks in the same safety channel and communications between redundant safety channels, the TXS has communication gateways between the TXS safety channels (through the MSI) and the non-safety-related service unit, plant process control computers, and monitoring computers. Typically the service unit is installed in an electronic or I&C service room near the main control room, and is connected to the TXS MSI via an H1 bus. The gateways are not Class 1E equipment, and are therefore isolated from the Class 1E TXS safety system MSI with optical isolation devices.

Communications between the initiation trains of the safety system and the service unit are cyclic and cannot interrupt the normal functions of the safety system function processors without plant operator intervention. This operator intervention cannot be performed on more than one safety system channel at a time. Interventions of safety system channel operations by

the operator are permitted to change system parameters or tracing signals, to perform periodic tests, and to allow diagnosis of safety system failures.

The TXS RTE controls all processing cycle activities, including communications. The TXS processing cycle is started by the central control unit of the RTE by triggering the internal MicroNet controller to transfer the messages in the receive dual-port RAMs of all linked communication modules into the corresponding message input buffers. After the integrity of the message is checked by means of a 16-bit cyclic redundancy check (CRC) (for the occurrence of random bit errors) and by means of a sequence increment (to ensure that a new message has been received), the message is flagged as a valid or an invalid message for subsequent processing.

The RTE software automatically marks the invalid message and all signals stored in this message with the ERROR status flag. Signals marked with ERROR status flag are excluded from further processing by the function blocks. For example, a "2-out-of-4" voting function block will calculate a "2-out-of-3" function of the remaining 3 input signals, if one input signal is marked with the ERROR status flag. For example, a "2.MAX" analog signal selection block function block will select:

- The "2<sup>nd</sup> highest" signal of the remaining 3 input signals, if one input signal is marked with ERROR status flag.
- The "2<sup>nd</sup> highest" signal of the remaining 2 input signals (that means the lower one), if two input signals are marked with ERROR status flag.
- The remaining input signal, if three input signals are marked with ERROR status flag.

Additionally, a communication error flag is set by the RTE. This information is transferred to the service unit and to the main control board alarm system.

The safety function can be postulated to be lost only if all of the incoming data is old or corrupt. For this case a fail-safe state of the function can be designed on the application software level. In all other cases (loss of 1, 2 or 3 input signals) the function will be executed correctly based on a reduced set of available input information. As soon as the communication failure is repaired (that means, the receiving CPU finds new and consistent data in the dual port RAM), the ERROR status for the incoming data will be automatically reset and this data will be used for function processing. No manual initialization is necessary.

During function diagram processing, the valid signals are further processed to determine out-of-range conditions and defective-instrument conditions. All provisional signals are stored in dedicated signal buffers during function diagram processing.

After function diagram processing has been completed, the central control unit triggers the "function diagram group output function" to create output message data from the results of the function group processing. The RTE then adds a message header to the message data, which includes a CRC checksum and the cycle counter, and stores the output message data in the message output buffers.

As the last step in the processing cycle, the RTE triggers the communications control program (MicroNet) to transfer the message data from the output buffers into the sending dual-port RAMs of the respective communication processor. This message data is then sent to other subracks and the MSI, as required, using either the H1 protocol (within the same channel) or the L2 protocol (between channels).

To achieve a deterministic system behavior, all communication is performed strictly cyclically, with the system-wide unique communication cycle. No event-driven communication is used. Thus, communication loads are constant under all circumstances.

The communication protocols used for sending messages are not acknowledged by the receiver. Thus, the subrack receiving the message cannot influence the operation of the sending subrack.

#### 2.2.1.2 Platform Software

The platform software consists of the RTE and its modules, the input/output (I/O) drivers for the input/output module interface, the exception handler, and the self-test software. The purpose of the RTE is to give a unified environment for execution of the function diagram group (FDG) modules. The RTE controls the cyclic processing of the FDG modules and controls signal transfers via messages or directly by I/O modules. The RTE also provides the interface of the runtime system software to an external service unit, through which it can be monitored and controlled (e.g., by signal trace, reading error messages, switching operation modes, and function block (FB) module parameterization).

The RTE provides four operation modes:

- **OPERATION:**

This is the normal operation mode for cyclic processing of the FDG-modules.

- **PARAM:**

This mode is the same as the OPERATION mode, but parameterization of FB-modules and definition of trace data are now allowed.

- **TEST:**

This mode is used for functional testing. The FDG-modules can be processed in single-step mode, and all external input signals can be inserted from the service unit. The results can be monitored by tracing internal and external signals.

- **DIAGNOSIS:**

In this mode, direct memory access is granted to the service unit. Special diagnosis programs can be downloaded from the service unit into RAM. Additionally, the target system debug functions are activated, so that debugging with an external debugger is possible.

To prevent unauthorized interference from the service unit, operation mode release signals control transitions between operation modes. These release signals are acquired either directly by input modules or via data messages. They are specified in function diagrams and passed to the RTE by a special interface function block.

The RTE also includes three sub-modules:

1. Input/output (I/O) drivers:

An I/O driver module is provided for every type of input/output module. The drivers transfer input/output signals between the RTE and the specific I/O module. The I/O drivers also initialize the I/O modules and detect I/O errors.

I/O modules are process interface modules between the different processors and the plant instrumentation. There are only nonprogrammable I/O modules in the TXS. All modules are on printed circuit boards with a connector to the backplane bus and a male multipoint connector for connecting test and monitoring equipment to the front panel. Analog modules contain an analog-to-digital converter and a multiplexer. The method of operation is always such that signal transmission between the signal buffers and the male multipoint connector toward the module operates independently of the data transmission between the module and the backplane bus. Accesses to the I/O modules by the function processors are by direct addressing via the backplane bus by module-specific software drivers that perform both data conversion and fault processing.

Monitoring equipment on I/O modules mainly aim at external circuitry. Failures in the interface with the backplane bus are detected and signaled by the function processors during their cyclic access to the signal buffers. The majority of failures concern signal input or signal output channels. If a multiplexer or a signal converter fails, the entire module is affected. Failures in the interface can, in rare cases, affect the backplane bus. Failures of the signaling equipment (e.g., light emitting diodes) do not have any effect on the safety functions.

2. Exception handler:

The exception handler module responds to unexpected situations, such as time-out, watchdog, or unexpected operating code exceptions. The exception handler saves information about the exception and its context for subsequent analysis of system behavior. Depending on the type of exception, the exception handler then either restarts the processing module (through a software-activated reset) or shuts down the processing module in a defined state. Information saved by the exception handler can be read from the service unit via services of the RTE or read directly via serial line connection from the front plate of the processing module.

3. Self-monitoring software:

The self-monitoring software performs a sequence of self-monitoring checks on the various hardware components of the processing module, such as RAM-test, FEPRM-test, and watchdog test. The self-monitoring is performed during time intervals when no cyclic processing of the FDG-modules is active. It is repeated

continuously, and its cyclic processing is monitored by the RTE. Any errors found are reported to the exception handler, which stores the information and takes care of the error handling.

The RTE has two major interfaces: the interface to the FDG-modules, and the interface to operating system software layer/target system hardware (target system interface). The FDG module interface consists of a set of unified functions. The RTE calls these interface functions via function pointers. The function pointers and the associated data structures and data types are defined in the RTE configuration module, which is generated by an automatic code generator (SPACE). These functions are described below:

- **Input function**

This function is used to pass input signals (from messages and/or input modules) to the function diagram (FD) modules contained in the FDG module.

- **Compute function**

This function is used to execute the computation of the FDG-modules. The compute function calls the associated FD compute functions in the correct order. These internally call the basic function blocks of each FD in the required order.

- **Output function**

This function is used to pass the calculated output signals of the FDG modules to the RTE. The signals are then sent via messages to output modules.

- **Interface function**

This function is an universal interface for read and write access to all FDG-module internal data structures, such as parameters, state variables, signals, etc. The RTE uses this interface function for accessing signals when tracing or changing the parameters of FDG-modules is required.

The target-system interface is the other major interface of the RTE. As a result of the nature of the system requirements, this interface is not as unified as the FDG-module interface. Basically, the target system interface consists of the following:

- **Operating system interface**

The operating system interface is restricted to an absolute minimum set of services: real-time related service (pause, task end, and resume after specified time interval), semaphore services, and event-flag services.

- **Communication software interface**

The communication software interface provides a unified interface for sending and receiving messages via communication channels. This is independent of the media used (media supported by the communication software are 32-Bit parallel backplane bus,

16-Bit parallel local extension bus, Ethernet 802.2/3 LAN, and Profibus LAN). Communication services are: create communication channels, send data via communication channels, receive data via communication channels, and get communication channel status.

- **Target system hardware interface**

The RTE also has a direct interface to certain components of the target-system hardware, either by direct access or by functions provided by HOT. This includes: access to LEDs on the processing module front-plate, EEPROM programming services, and watchdog services. The RTE target system interface, although not as unified as the FDG-module interface, has been designed to facilitate portability. This was accomplished by concentrating all target system-dependent interface functions in one sub-module, SYSTEM. Porting the RTE to another platform is done by adapting the module SYSTEM to the new platform.

On top of the RTEs internal module structure there are three modules, which control the three major functions of the RTE:

- **Module INIT**

This module is the central control instance during start-up of the RTE. It controls the complete initialization phase of the RTE. After the operating system has started the automatic startup tasks, the RTE monitoring tasks perform the RTE initialization and then start the RTE cycle task which starts operation in mode INIT. In this mode, the status of the input signal messages are checked. If all input signal messages are received satisfactorily (or when a timeout of typically 2 minutes has expired), the FDG-modules are initialized. After FDG initialization has been successfully completed, control is branched to the modules CYC and MONIT. If the initialization fails, cyclic operation will not be achieved and module INIT ends in an endless loop.

- **Module CYC**

This module is the top-most control module for the cyclic operation of the RTE. CYC uses services of underlying modules like FDGIFC (for interface to the FDG-modules), MODE (for handling operation mode transitions), or ERRORMSG (central error-message handling module).

- **Module MONIT**

This module controls the RTE interface to an external service unit. MONIT accepts a set of basic control commands, e.g., reading an error-message-buffer, requesting a change of operating mode, or setting a new parameter value. Not all control commands are permitted in every operating mode. For example, during normal operating mode OPERATION, only a very small subset of control commands is accepted by the RTE. This prevents unintended interference from the external service unit during normal operation.

The RTE, acting as a virtual machine, hides all target specifics such as hardware, operating system, communication media and protocols, and I/O modules from the FDG-modules. The RTE is activated cyclically by the operating system and then processes the following functions cyclically:

- resets the watchdog timer,
- increments the cycle counter,
- reads in the process data from the input/output (I/O) modules,
- reads messages from the dual-port RAM,
- transfers the data to the function diagram group modules,
- processes the function diagram group modules,
- checks fault messages from processing the function diagram group modules and sets the fault status signals,
- outputs the results of the function diagram group modules via the I/O modules,
- transmits messages to other function processors,
- activates service tasks if permissible manual control requests have been identified, and
- deactivates its own task until the next cycle.

At the beginning of the processing cycle, the RTE resets the watchdog timer to a value that is greater than the activation cycle for the RTE cycle time set in the operating system. If the RTE does not terminate correctly because of a fault in the signal flow, the watchdog timer times out and generates a hardware interrupt request. This interrupt request then activates a special interrupt service (the exception handler) that saves the current state of the processor for subsequent analysis and then puts the computer into a defined fault state. In this fault state, all output signals are set to predetermined states and the processor is kept in a waiting loop. The signal outputs are disabled in several different ways by explicit driver calls and by a hardware signal (BASP), which disconnects the local power supply for the I/O modules.

When the watchdog times out the first time, the hardware interrupt leads to an automatic reset (re-boot) of the CPU (it is assumed that a transient failure was responsible for the watchdog activation). During the automatic re-boot of the CPU the complete startup self-test is executed (in-depth check of the CPU hardware). In case the CPU passes the startup self-test it resumes the cyclic operation. If the watchdog times out again, the interrupt activated by the exception handler leads to the shutdown of the CPU (a severe hardware failure is assumed). A shut-down CPU can be re-activated only by re-setting the complete I&C subrack, which has to be done manually. However, prior to re-start of the CPU a complete check of the failure information would be appropriate.

The RTE also increments the local cycle counter. This 16-bit counter forms the internal relative short-time base sign-of-life clock for communication and for time-sequencing fault signals. The cycle count of the RTE at the time of transmission is appended to every message. This information is used by the receiving processor to monitor the validity of the message and the correct functioning of the transmitter.

Data are input and output via the I/O modules directly through driver programs that access the buffers of the I/O modules that are reading or writing the data. A separate driver program exists for each I/O module, and is also responsible for module-specific conversion of the data. Fault alarms that are detected on the I/O module (wire break, overflow, underflow) are used to mark the signals with the status "ERROR." Each configured signal in the TXS contains the

signal value and a signal status attribute with the status flags "Fault" and "Test." These flags are used for fault masking. Missing front connectors or a missing load power supply can result in the enable signal for addressing I/O modules not being formed. The missing enable signal is detected and signaled by the time-out monitoring. At the same time, the status flags are set to "ERROR" for all signals concerned so that these signals are not used in further signal processing.

The number of signal ERRORS and/or failures in the entire system that would be permitted before leading to the degradation of the safety function depends on the system architecture selected for the task to be fulfilled (such as for the reactor trip system and engineered safety feature actuation system). How long such a degraded system operation would be allowed is governed by the technical specification for a specific system application.

The function processor reads messages by direct accessing the local dual-port RAM. In safety-related applications, all messages contain additional data for monitoring system integrity at the application layer. This includes the cycle count of the RTE that transmitted the message, the message identification number, the message length, and a checksum with which the integrity of the data from the RAM of the transmitting function processor to the RAM of the receiving function processor is monitored. If the RTE detects that the cycle count has not been incremented properly in a received message, the receiving processor considers the data faulted and an up-circuit processor is no longer functioning properly. An incorrect checksum also indicates that the messages are inconsistent and must be excluded from all further processing. If the data received are current and consistent, they are passed on to the application functions (function diagram group modules).

For the master/checker pair of voter computers, the necessary signal exchange and result comparison between the redundant computers is performed by the runtime environment.

After transferring the data to the function diagram group modules, the RTE begins processing the data in the function diagram group modules. During normal system operations, this processing activity is assigned the highest priority, such that, if an operator requests a service function, the service function is not processed until after the function diagram group modules are processed. During function diagram group module processing, the RTE checks fault messages arising from the processing, and sets fault status signals for use by the exception handler. Details regarding the function diagram group modules are provided in the next section.

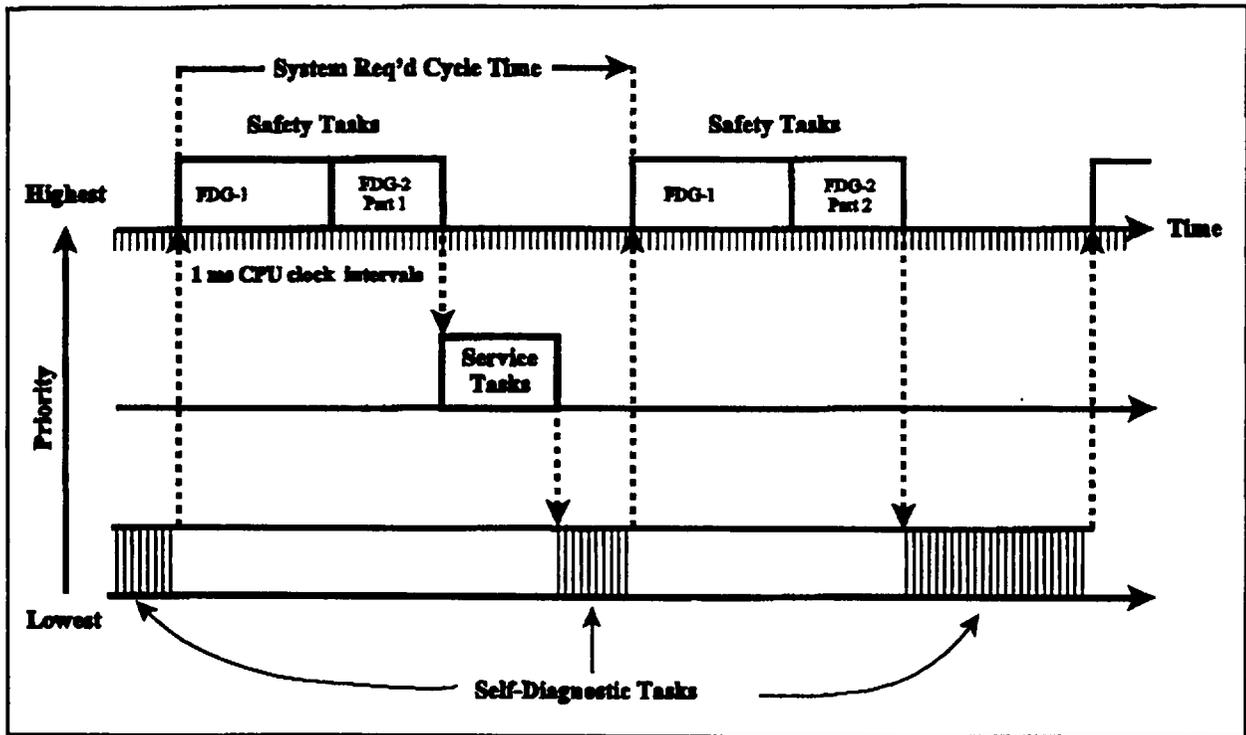


Figure 2. Task scheduling during normal operations.

Upon completing the processing of the function diagram group modules, the RTE places the processing results into the sending port of the dual-port RAM, for processing by the I/O modules. The I/O modules then process the results for subsequent transmission to other function processors.

If service tasks have been requested by the operator, these tasks are then scheduled for processing in the time remaining during the cycle (see Figure 2). The service tasks are kept from being processed by use of a semaphore, which is held by the safety function. After the safety function is processed, the semaphore is passed to the service task for one clock cycle (1 ms). The service task retains the semaphore for that clock cycle then returns the semaphore to the RTE. If the cycle time has not elapsed, the RTE passes the semaphore back to the service task for further processing. This exchange of a semaphore continues until the service task is completed or the cycle time expires, whichever occurs first. Using this technique, the safety function retains the highest priority for system resources.

Upon completion of the safety function(s) and the service task(s), the RTE deactivates its own task until it is again activated by the operating system scheduler at the start of the next operating cycle. During this time period, the operating system activates the self-test functions, which are processed in 1-ms time increments until the next scheduled processing cycle is to begin. The operating system scheduler then activates the RTE for processing the safety functions. The 1-ms time increments are used to ensure that safety function processing is performed at the required frequency.

RTE provides computing time monitoring. If the computing time is greater than the cycle time minus 2ms, an error message is generated. The RTE also monitors execution of the cyclic

self-monitoring. If not completed after a specified time, an error message is issued and sent to the service unit, and to the exception handler which will immediately shutdown the CPU. In addition, RTEs in the various system CPUs receive messages from each other and monitor the message age. By this means the sender's cyclic operation is monitored independently by other RTEs. Furthermore, the watchdog timer monitors execution of the RTE's cyclic task, and is activated after a very short period of time following the start of the last cycle if a problem is detected.

The TXS system automatically detects failures in the subracks, the function processors, the I/O modules, and the communication functions. Failures that affect the subrack internal power supplies or control of the backplane bus will cause a transition to predefined fault conditions (e.g., reset) on the function computers, which results in a nonresponsive state in relationship to other subracks. Additionally, the TXS system monitors cabinet temperatures and cabinet cooling fan speed and provides the plant operators with an alarm if setpoints are exceeded.

The function processors are designed so that two independent monitoring systems for detecting failures could affect safety system operation. One approach for detecting failures is the strictly cyclical use of all hardware components. The watchdog timer and the system support controller provide this monitoring capability. The watchdog timer monitors program operation and the system support controller monitors hardware access times. The hardware times monitored by the TXS system are:

- access times and time-out times on accesses to the I/O bus and the backplane bus, and
- the waiting time for allocation of the backplane bus.

The cyclic self-monitoring function also addresses:

- the integrity of the data permanently stored in the flash EPROM and in the EEPROM by use of CRC checks,
- the function of the read and write memory,
- the function of the processor and coprocessor,
- the function of the timers, interrupts, and watchdogs,
- the function of the I/O ports, and
- the correct setting of the jumpers.

These monitoring capabilities ensure the detection of function processor failures that could affect safety system operations.

Monitoring functions for the I/O modules primarily address wire breaks, connectors, and measuring ranges. Consequently, a failure of the module itself is only partly detected by system-inherent equipment. Failures in the interface with the backplane bus are detected and signaled by the cyclic accesses of the function processors to the signal buffers. The majority of failures concern signal input or output channels. If a multiplexer or a converter component fails, the entire module is affected. In unusual cases, a failure in the interface can also affect the backplane. Failures of LEDs (light emitting diodes) and other signaling equipment do not affect the safety function of the system. The use of voting, which uses redundant signals to arrive at a safety state, provides assurance that a single failure in an I/O module will not affect the safety function.

TXS communication functions have three independent methods of detecting failures. These methods are the LAN protocol detection mechanisms, the communication processor detection processes, and the CRC on the application layer. Using these three methods, the TXS can detect component failures that are restricted to the bus interface, component failures of an entire module, and component failures that affect an entire subrack or an entire bus segment.

Typically, communication failures affect the links that serve failed equipment. For equipment connected to the subrack, failures of the interface with the backplane bus can affect the entire subrack. Modules that function directly on the serial bus can cause an entire bus to fail. These types of failures are easily detected by the system.

Configured monitoring mechanisms in the TXS make use of redundant information processing with subsequent voting of the results. By comparing information from redundant channels, the TXS can detect deviations in the redundant signals and so identify module failures. Four basic types of configured monitoring mechanisms are used in the TXS system: redundant measured value processing, configured monitoring of I/O modules, use of master-checkers for comparing results, and use of voters.

In nuclear power plant safety systems, safety-related values are acquired and processed in redundant channels. One way TXS systems process redundant signals is to use the signals to determine a real time best-estimate representation of the actual process being measured. The process for obtaining this best-estimate value is usually by voting the appropriate signals and then selecting the best-estimate value from the result of this voting. For example, the voting process could use the second highest value of four signals for selecting the signal to be used for actuating a safety system on a high trip setpoint. This process also allows the system to monitor the consistency of the signals. All failures that do not result in a "frozen" value can be detected with redundant measured value processing.

The TXS system applies test signals to each TXS safety circuit to continuously monitor the I/O modules. The test signals are monitored to ensure the I/O modules are correctly processing the signals. The degree of monitoring depends on the safety function to be performed. This method of detecting I/O module failures is only incrementally better than redundant value processing at detecting failures.

For output modules that control important items of safety equipment, another method is to read back and compare output signals for consistency with the expected value. The advantage of this method is that, for frequently used equipment, equipment errors may be detected before the failure has a significant effect on safety.

### 2.2.1.3 Application Software

The application software performs the plant-specific TXS safety-related functions using function block modules, function diagram modules, and function diagram group modules.

Function block modules perform all numeric operations involving input signals. These operations include basic functions such as adding signals, integrating signals, and comparing signals to predefined values. These function blocks are available in the form of type-tested libraries. Each function block module is associated with a data structure that contains the input data, output data, all buffer addresses, and parameters specific to the function of that module.

These data structures allow complete verification of each calculation step in the processing sequence. Within a processing cycle, all temporary data are preserved and not overwritten. At the end of the cycle, these data can be transmitted to the service unit for subsequent system maintenance, diagnostic activities, and status tracking.

All function block modules process the value and status of their input signals. The signal status is an attribute that indicates the quality of the output signal. Function block modules have either active-status processing or passive-status processing. For passive-status processing, the output signal of a function block is simply formed by OR gating the status information of all signals. Using passive-status processing, any signal marked as faulted causes the resulting signal of the function block to contain the attribute ERROR. For function blocks with active-status processing, the resulting signal is only calculated from fault-free input signals so that the output signal is also marked as fault-free. In this case, a fault-free output signal can only result if the input signals give mutually redundant information. Active-status processing, therefore, is only possible with function blocks that perform selection and majority voting functions. These include blocks such as for second-maximum, second-minimum, and "m-out-of-n" coincidence logic. Function blocks with active-status processing are used to screen, or mask, faulted signals and thereby prevent their propagation to the next function. Function blocks are not plant-specific functions. Rather, these blocks are stored in a development system library for use by system designers.

Function diagram modules consist of groupings of function blocks that are used for plant-specific applications. The function diagram modules are developed from plant-specific function diagrams using an automated and qualified generation process. The function diagrams are composed of graphic function blocks that represent a separately testable function. Once the function diagram is completed, the diagram is converted into a database, from which the graphical application is developed into function diagram modules, which have corresponding predefined software modules. The function diagram modules are activated by the runtime environment in the form of function calls. All safety functions are specified as function diagrams.

Function blocks are connected by signals via their input and output ports. Each port has an associated type, which defines the type of signal that can be connected to it. Three types of signals are defined in the TXS: analog (float valued) signals, binary (boolean valued) signals, and message signals.

The rules that define the connectivity of function blocks are based on the signal types of the ports. Only ports of the same type can be connected by a signal. This is checked on-line during the specification of a function diagram by the graphical TXS editor.

An underlying restriction is imposed for message signals. Each message signal is of the generic type, although the bit-mapped message code is specific to each function block type. As a result, for example, the message signal output port of the function block that evaluates two or more trip signals out of four signals may not be connected to the message signal input port of a different type of message decoder even though both ports are of the same generic type. This type checking is performed at runtime by the message decoders themselves, based on the function block (FB) information that is part of each message signal.

Because several function block modules are typically implemented in one processor, all function diagram modules that are to be processed with the same cycle time are grouped together to form function diagram group modules. A function diagram group module therefore consists of a sequence of calls to function diagram modules and copy functions by which signal transfers between the function diagram modules are implemented. A function diagram module consists of a sequence of calls to function block modules that are interconnected by data structures. Function diagram group modules are stored in the flash EPROM on the system processor printed circuit board.

Two function diagram group modules can be implemented on one function processor. The processing cycle time of the runtime environment corresponds to the cycle time of the faster function diagram group module. Processing of the slower function diagram group module is distributed over several basic cycles to achieve a constant load distribution. The allocation of the function diagram group processing to several basic cycles is generated explicitly by the code generator. Function diagram modules and function diagram group modules do not use system services. Only the runtime environment is responsible for supplying data and passing on the results by calling the required copy functions and initiating output communication requests.

Function blocks in different function diagrams (FD) are connected in the same way as they are within the same FD. The only difference is that the connecting signals must be exported by the source FD and imported by the receiving FD. Each signal gets a unique name code, based on the name code of the source FD. This is in contrast to local signals within a FD, which are not identified by a unique FD name code.

The TXS conceptual architecture supports the software specification, the hardware specification, and the software-hardware interface. The software specification consists of the function diagrams discussed above. The software specification is independent of the target system and is a formal, domain-specific specification that defines the signal processing used to implement the elementary safety functions defined by the system requirements specification. The software specification is created using the TXS editor.

The hardware specification contains the complete hardware structure of the target system, with all its hardware components. For this specification, hardware diagrams and hardware blocks representing the target system hardware components are used. The hardware specification is also created using the TXS editor.

After creating the hardware and software specifications, each function diagram is assigned to one processing module. This assignment is done while creating the hardware diagrams, by adding the function diagram identification to the parameter list of the processing module using the TXS Editor. The information for the integrated system design is stored in the specification database.

The complete specification captures functional aspects and the system's detailed hardware structure. Nonfunctional aspects such as independence constraints, fault tolerance, and timing requirements are also implicitly contained in the specification. The specification can be prepared by I&C engineers using notations and methodologies that have been common practice in the I&C community. The software specification remains independent of the specific details of the target system. The verification of the specification by the process engineers who

prepared the system requirement specification is facilitated by the use of a commonly understood notation.

The specification is formal in the sense that all information needed to implement the final code running in the distributed target system is available from the specification database. Also, certain verification procedures such as check of completeness, unambiguity, consistency with naming scheme, and parameter checks can be done automatically at specification time. Using the formal specification and a set of predefined rules, the target system code is generated automatically, thus improving code quality and reducing costs. On the other hand, by applying the inverse rules, the generated target system code is analyzed by independent tools and compared to the original database representation.

### 2.2.2 Software Documentation

This section summarizes the software documentation associated with the TXS system development. The type tests of the TXS software components were performed in accordance with German standard KTA-Standard 3503. The principles of type testing and the test activities were defined from this standard. These were applied to the following areas: separation in the theoretical and practical tests, institutions to be involved in type tests, roles of these institutions in type tests, and documentation of type tests.

The content of the theoretical and practical tests is defined by the software standard DIN IEC-880.

KTA standards also require that the present state-of-the-art be taken into account during the qualification. In addition to KTA-1401, which defines criteria for quality assurance systems, the following software standards were applied and verified:

- ISO-9000-3, "Management for Quality and Requirements of Quality Assurance,"
- IEEE-830, "Software Requirement Specifications,"
- IEEE-828, "Software Configuration Management Plan,"
- IEEE-1012, "Software Verification and Validation Plans,"
- IEEE-829, "Software Test Documentation,"
- IEEE-1008, "Software Unit Testing,"
- IEEE-1028, "Software Reviews and Audits," and
- ANSI/ANS-10.4, "Verification and Validation of Scientific/ Engineering Programs for the Nuclear Industry."

Among the standards referenced in the Standard Review Plan and the Branch Technical Positions, IEEE-7-4.3.2 gives specific requirements concerning software development. Most of these requirements are given by reference to the standards ASME NQA-10.4, IEEE-730, IEEE-828, IEEE-1012, and IEC-880. The requirements of ASME NQA-10.4 are covered by KTA -401, and the requirements of IEEE-730 are covered by ISO-9000-3. All other standards were directly applied in the development and evaluated in the type tests.

#### 2.2.2.1 Vendor/Customer System Specification

Documentation supporting the system specification will be reviewed on a plant-specific basis.

#### 2.2.2.2 Software Management Plan

The software management plan for development of a Siemens digital safety system is the same procedure as used for all Siemens safety-critical software development projects. The software management plan is incorporated into Siemens Engineering Procedure FAW-1.1, "Software Life-Cycle Processes." FAW-1.1 specifies the management structure and the processes to be used in the project. This procedure is compatible to IEEE-1074, "Developing Life Cycle Process," and is, therefore, acceptable.

#### 2.2.2.3 Software Development Plan

The software development plan is documented in Siemens Engineering Procedure FAW-1.1, "Software Life-Cycle Processes." This procedure defines the software life cycle processes to be used in the development of a safety-related digital system.

#### 2.2.2.4 Software Quality Assurance Plan

The software quality assurance plans are incorporated into three Siemens Engineering Procedures, FAW-3.4, "Contents and Structure of System Specifications for Software Components," FAW-3.5, "Contents and Structure of Design Documents for Software Components," and FAW-3.6, "Contents and Structure of Implementation Documents for Software Components." FAW-3.5 describes the process by which the software specification is translated into the software design description. FAW-3.6 describes the process by which the software design description is implemented. The staff has reviewed these procedures and found that these procedures are compatible to the U.S. IEEE standards listed in the NRC SRP Chapter 7, therefore, the staff considered these procedures acceptable. Additional Siemens corporate quality assurance procedures include EMF-1, Part II, Rev. 29, "Siemens Power Corporation Quality Assurance Manual for Nuclear Fuels and Services" (approved by the NRC by letter dated June 11, 1998), and QMH 12E KWU NL, "Quality Program, Quality Manual Handbook for Nuclear Services."

#### 2.2.2.5 Software Configuration Management Plan

Configuration management activities are controlled by Siemens Engineering Procedure FAW-1.5, "Configuration Management," which outlines the procedures and tools for creating and implementing the configuration management structure and procedures. This procedure is compatible to IEEE-828, "Software Configuration Management Plan," and is, therefore, acceptable.

#### 2.2.2.6 Hardware and Software Specification

The procedure for controlling the hardware and software specifications is Siemens Engineering Procedure FAW-3.3, "Organization of the General Specification for SW and HW Components." This procedure governs the organization of the specifications for the digital safety systems created under this set of tools and processes. This procedure is compatible to IEEE-830, "Software Requirement Specifications," and is, therefore, acceptable.

#### 2.2.2.7 Software Requirements Specification (SRS)

The software requirements specifications are controlled by Siemens Engineering Procedure FAW-3.4, "Contents and Structure of System Specifications for Software Components," and FAW-3.5, "Contents and Structure of Design Documents for Software Components." FAW-3.4 describes the process to be used for converting the system requirements into software specifications. FAW-3.5 describes the technical processes for converting the software specification into a module structure that may be used for implementing the software requirements. These procedures are compatible to IEEE-7-4.3.2, "IEEE Standard for Digital Computers in Safety Systems of Nuclear Power Generating Stations," and are, therefore, acceptable.

#### 2.2.2.8 Software Requirements Review (SRR)

The processes by which software requirements are reviewed are described in Siemens Engineering Procedure FAW-4.2, "Reviews." This procedure describes the software review process, including responsibilities, review methods, the review processes, and activities to be performed after the review is completed. This procedure is compatible to IEEE-1028, "Software Review and Audit," and is, therefore, acceptable.

#### 2.2.2.9 Software Design Description (SDD)

The processes controlling the software design description are specified in Siemens Engineering Procedure FAW-3.5, "Contents and Structure of Design Documents for Software Components," and FAW-3.6, "Contents and Structure of Implementation Documents for Software Components." FAW-3.5 describes the process by which the software specification is translated into the software design description. FAW-3.6 describes the process by which the software design description is implemented. These procedures are compatible to IEEE-7-4.3.2, "IEEE Standard for Digital Computer in Safety Systems of Nuclear Power Generating Stations," and are, therefore, acceptable.

#### 2.2.2.10 Software Design Review (SDR)

The processes by which software design is reviewed are described in Siemens Engineering Procedure FAW-4.2, "Reviews." This procedure describes the software review process, including responsibilities, review methods, the review processes, and activities to be performed after the review is completed. This procedure is compatible to IEEE-1028, "Software Review and Audit," and is, therefore, acceptable.

#### 2.2.2.11 Source Code Listing

The structure of the source code listings is specified in Siemens Engineering Procedure FAW-2.1, "Coding Rules." These rules are incorporated into the SPACE tool, which generates the function diagram modules and function diagram group modules. FAW-2.1 provides specific programming guidelines for the C, C++, and FORTRAN 77 software languages. Source code listings of specific applications will be reviewed on a plant-specific basis. This procedure is compatible to IEEE-1028, "Software Review and Audit," and is, therefore, acceptable.

#### 2.2.2.12 Source Code Review

The processes by which software requirements are reviewed are described in Siemens Engineering Procedure FAW-4.2, "Reviews." This procedure describes the software review process, including responsibilities, review methods, the review processes, and activities to be performed after the review is completed. This procedure is compatible to IEEE-1028, "Software Review and Audit," and is, therefore, acceptable.

#### 2.2.2.13 Safety Analyses

Safety analyses of specific applications are the licensee's responsibility and will be reviewed on a plant-specific basis.

#### 2.2.2.14 Software Verification and Validation Plan (SVVP)

The processes for conducting software verification and validation (V&V) activities are described in Siemens Engineering Procedure FAW-1.6, "Verification and Validation Plan." FAW-1.6 specifies the areas of application, the organizational responsibilities, requirements for IV&V activities, and requirements for documentation. This procedure is compatible to IEEE-1012, "Software Verification and Validation Plans," and is, therefore, acceptable. The requirements for V&V are described in IEC-880-1986, "Software for safety Systems in Nuclear Power Stations," which Siemens has followed throughout the life cycle. IEC-880 is compatible to IEEE-7-4.3.2, "IEEE Standard for Digital Computers in Safety Systems of Nuclear Power Generating Stations," and is, therefore, acceptable. Additionally, an IV&V plan has been prepared by the IV&V groups TÜV Nord and iSTec, which are working under a subcontract for the German Reactor Safety Association, which has a contract with the Bavarian nuclear licensing authority to perform third party software type tests.

#### 2.2.2.15 Verification and Validation (V&V) Report

Verification and validation reports are prepared by Siemens for each specific application and for each component of the application development database. The staff reviewed V&V reports for three components during its audit of Siemens in December 1999 and found the reports acceptable as discussed in Section 4.4 below.

#### 2.2.3 Development and V&V Organization and Process

The V&V processes are defined in Siemens Engineering Procedure FAW-1.6, "Verification and Validation Plan." This plan specifies all activities performed during the safety system development process. The responsibility for V&V activities is with the person responsible for the system or module development. This procedure is compatible to IEEE-1012, "Software Verification and Validation Plans," and is, therefore, acceptable.

Siemens internal V&V processes were performed by members of the same development team, a member of another team within the digital I&C organization, or by employees outside the digital I&C organization. The person performing the internal V&V activity was not the same person who generated the product to be reviewed. External IV&V activities were performed by TÜV organizations and iSTec.

The person responsible for ensuring the V&V activities are performed develops the draft development document. This person then assigns the document draft status and releases the document for a consistency check and a technical review. The consistency check of the draft document is normally performed by a member of the V&V team. The purpose of the consistency check is to ensure that all interfaces to adjacent components have been considered.

The participants of the technical review follow the requirements of FAW-4.2, "Reviews." The participants for this review are equivalently qualified and are not involved in the generation of the document to be reviewed. The results of the review are incorporated into the draft document by the person responsible for the document. The resulting document is made available for external verification and review.

Next comes external verification of the development documents for components that do not involve a specific plant application, which involves verifying that the safety-related features of the system have been correctly incorporated into the safety system. A final check of the verified documents is then performed, and the resulting documents are sent to the testing organization for checking the external modification requirements.

The project manager has the final approval responsibility of the documents.

Validation activities include testing the application to ensure it performs according to the system requirements. These activities are controlled by Siemens Engineering Procedure FAW-4.1, "Testing." Testing includes specifying the test requirements, performing the tests, and producing the test report. Testing includes module testing, component testing, and system testing in a simulated and real environment. This procedure is compatible to IEEE-1008, "Software Unit Testing," and is, therefore, acceptable.

Documentation of the test results includes verification of the phase results, validation of the test specification, and integration of the component into the system.

Siemens performed V&V activities during each phase of the software development. The purpose of the activities was to verify that the requirements were correctly addressed throughout the software life cycle (component specification, component design, implementation, and testing). The system integration and operations phases are plant specific, and the associated V&V activities and documentation will be reviewed on a plant-by-plant basis.

#### **2.2.4 Configuration Management**

Configuration management activities are controlled by Siemens Engineering Procedure FAW-1.5, "Configuration Management." This procedure provides the requirements and 71 procedures necessary for maintaining configuration control of the project. The procedure defines the configuration requirements and specifies the processes for generating configuration identifiers, controlling changes, and maintaining version control during the development process. Configuration identification is applied to all software and associated documentation.

Baselines are established to control design, product, and engineering changes. These baselines are defined by the configuration manager, and cannot be changed without approval

of the configuration control board. The procedure to change a configuration item consists of the following steps:

- A change request is prepared by the person requesting the change.
- The development group evaluates the change.
- A proposal describing the process for making the change is prepared.
- The development group recommends disposition of the change request.
- The configuration item is changed according to the approved change request and proposal for change.

Software configuration management and control is applied to all documents and software code. This control is implemented using the configuration identification number assigned by the configuration manager.

Siemens maintained the documentation of configuration management for the TXS system platform that includes long-term service with compatible hardware and software components. Siemens also maintained the documentation of project-specific configuration management activities that include project-specific application software consistent with functional requirements. Every hardware and software component has a certification that identifies the modification version of that component.

The staff found that the configuration management procedure FAW-1.5 is compatible to IEEE-1042, "IEEE Guide to Software Configuration Management," and is, therefore, acceptable. However, the licensee should demonstrate that the plant-specific configuration management activities have been carried out in the life cycle process implementation. Documentation should exist that shows the configuration baselines have been established for the activity group, and an adequate change control process has been used for changes to the product baseline. This is a plant-specific action item.

### 3.0 REVIEW CRITERIA AND METHOD OF REVIEW

#### 3.1 Review Criteria

The following acceptance criteria and guidelines for reviewing a safety-related reactor protection system such as the TXS system are identified in the Standard Review Plan (NUREG-0800), Sections 7.1 and 7.2:

1. 10 CFR Part 50, §50.55a(h), "Protection and Safety System."
2. General Design Criteria 2, "Design Basis for Protection Against Natural Phenomena."
3. General Design Criterion 4, "Environmental and Missile Design Basis."
4. General Design Criterion 20, "Protection Systems Functions."
5. General Design Criterion 21, "Protection System Reliability and Testability."
6. General Design Criterion 22, "Protective System Independence."

7. **General Design Criterion 23, "Protection System Failure Modes."**
8. **General Design Criterion 24, "Separation of Protection and Control Systems."**

The following regulatory guides and industry standards provide information, recommendations and guidance and, in general, provide an acceptable basis to implement the above requirements for both hardware and software features for safety-related digital systems such as the TXS system:

1. **Regulatory Guide 1.152, "Criteria for Programmable Digital Computer System Software in Safety Related Systems of Nuclear Power Plants."**
2. **IEEE Standard 7-4.3.2-1982, "Application Criteria for Programmable Digital Computer Systems in Safety Systems of Nuclear Power Generating Stations."**
3. **IEEE Standard 323-1974/1983, "IEEE Standard for Qualifying Class 1E Equipment for Nuclear Power Generating Stations."**
4. **IEEE Standard 338-1987, "IEEE Standard Criteria for Periodic Testing of Nuclear Power Generating Station Safety Systems."**
5. **IEEE Standard 344-1987, "IEEE Recommended Practices for Seismic Qualification of Class 1E Equipment for Nuclear Power Generating Stations."**
6. **IEEE Standard 379-1988, "Application of the Single Failure Criterion to Nuclear Power Generating Station Class 1E Systems."**
7. **IEEE Standard 384-1992, "Criteria for Independence of Class 1E Equipment and Circuits."**
8. **IEEE Standard 472-1974, "Guide for Surge Withstand Capability Tests."**
9. **IEEE Standard 730-1989, "Software Quality Assurance Plans."**
10. **IEEE Standard 828-1990, "Software Configuration Management Plans."**
11. **IEEE Standard 829-1983, "Software Test Documentation."**
12. **IEEE Standard 830-1984, "Guide for Software Requirements Specifications."**
13. **IEEE Standard 1012-1986, "IEEE Standard for Software Verification and Validation Plans."**
14. **IEEE Standard 1016-1987, "IEEE Standard for Recommended Practices for Software Design Descriptions."**
15. **IEEE Standard 1028-1988, "IEEE Standard for Software Reviews and Audits."**

16. MIL-Std-461, "Electro-magnetic Emission and Susceptibility Requirements for the Control of Electro-magnetic Interference."
17. MIL-Std-1399, "Interface Standard for Shipboard Systems, DC Magnetic Field Environment."
18. IEC-801-2, "Electromagnetic Compatibility for Industrial-Process Measurement and Control Equipment Part 2: Electrostatic Discharge Requirements."
19. SAMA PMC 33.1-1978, "Electro-magnetic Susceptibility of Process Control Instrumentations."
20. ASME NQA-2a-1990, Part 2.7, "Quality Assurance Requirements of Computer Systems for Nuclear Facility Applications, American Society of Mechanical Engineers."
21. EPRI Topical Report TR-107330, "Generic Requirements Specification for Qualifying a Commercially Available PLC for Safety-Related Applications in Nuclear Power Plants." TR-107330 was approved by NRC on July 30, 1998.
22. EPRI Topical Report TR-102323-R1, "Guidelines for Electromagnetic Interference Testing in Power Plants." TR-102323-R1 was approved by NRC on April 30, 1996.

### 3.2 Method of Review

The purpose of the NRC review is to determine whether the proposed use of equipment and other technical requirements provide reasonable assurance that the applicant or licensee will comply with the Code of Federal Regulations, Title 10, Chapter I, and that the public health and safety will be protected. The review, audit, or inspection activities are not intended to completely evaluate all aspects of the design and implementation of the Siemens TXS I&C system. The review scope was sufficient to allow the reviewer to reach the conclusion of reasonable assurance described above.

This topical report was submitted for generic review, and will be referenced in the future for a plant protection system upgrade or replacement. To ensure that the digital plant protection system will perform its safety function as designed, the staff concentrated on the basic operation of the TXS software system, the life cycle activities of TXS hardware and software systems, and the qualification testing. Meetings were held at the NRC office on October 14 and 15, 1999, and November 16, 17, and 18, 1999, and an audit review meeting was held at Siemens' office on December 6 - 10, 1999.

Based on previous advanced reactor digital I&C systems reviews, the staff developed a generic digital safety evaluation outline. The outline describes the kinds of information the staff needs to address in the safety evaluation for a complex digital I&C upgrade system. This outline was provided to Siemens as a framework for the agenda of the meetings listed above. A documentation audit was conducted at Siemens office in Germany to verify the information related to the TXS system life cycle activities. The staff made a site visit to a German nuclear power plant GKN (Gemeinschaftskernkraftwer) to observe the TXS system in operation. The plant operators demonstrated the operation, diagnosis, and maintenance capabilities of the TXS system and discussed the experience of changing from an analog system to a digital

system. The plant operators stated that they are very pleased with the capabilities and maintainability of the new digital system.

The staff has also relied on nuclear industry efforts to establish an appropriate qualification program for upgrading I&C systems in nuclear power plants. The EPRI Topical Report TR-107330, "Generic Requirements Specification for Qualifying a Commercially Available PLC for Safety-Related Application in Nuclear Power Plants," describes the generic functional and the qualification requirements for a PLC (programmable logic controller) and provides guidance on implementing PLC-based applications. TR-107330 was endorsed by NRC (SE dated July 30, 1998). The staff will require a licensee referencing the TXS system for upgrading I&C systems to meet the seismic and environmental qualification requirements specified in TR-107330. Likewise, the licensee should meet the EMI/RFI qualification requirements specified in EPRI TR-102323-R1, "Guidelines for Electromagnetic Interference Testing in Power Plants," (NRC SE dated April 17, 1996).

#### 4.0 SYSTEM EVALUATION

This section discusses the defense-in-depth and diversity assessment of the TXS system, surveillance testing, system response time testing, and software lifecycle evaluation of the TXS.

##### 4.1 Defense-in-Depth and Diversity (D-in-D&D) Assessment of the TXS System

The staff described concerns with common-mode failures and other digital system design issues in SECY-91-292. SECY-91-292 describes how common-mode failures could defeat the redundancy achieved by the hardware architectural structure, and also result in the loss of several echelons of defense-in-depth (provided by the monitoring, control, reactor protection, and engineered safety functions performed by the digital I&C systems).

The staff has established acceptance guidelines for D-in-D&D assessments and has identified four echelons of defense against common-mode failures:

- Control system - The control system echelon consists of nonsafety equipment which routinely prevents reactor excursions toward unsafe regimes of operation, and is used for normal operation of the reactor.
- Reactor trip system (RTS) - The reactor trip echelon consists of safety equipment designed to reduce reactivity rapidly in response to an uncontrolled excursion.
- Engineered safety feature actuation system (ESFAS) - The ESFAS echelon consists of safety equipment which removes heat or otherwise assists in maintaining the integrity of the three physical barriers (cladding, vessel, and containment) to radioactive release.
- Monitoring and Indication - The monitoring and indication echelon consists of sensors, displays, data communication systems, and manual controls required for operators to respond to reactor events.

As a result of the reviews of advanced light-water reactor (ALWR) design certification applications that used digital protection systems, the staff documented its position in SECY 93-087, "Policy, Technical, and Licensing Issues Pertaining to Evolutionary and

Advanced Light-Water Reactor Design," with respect to common-mode failure in digital systems and defense-in-depth. This position is also documented in the SRP BTP HICB-19, "Guidance for Evaluation of Defense-in-Depth and Diversity in Digital Computer Based Instrumentation and Control Systems." Points 1, 2, and 3 of this position, described below, apply to digital system modifications for U.S. operating plants:

1. *The applicant/licensee should assess the diversity and defense-in-depth of the proposed instrumentation and control system to demonstrate that vulnerabilities to common-mode failures have been adequately addressed.*
2. *In performing the assessment, the vendor or applicant/licensee shall analyze each postulated common-mode failure for each event that is evaluated in the accident analysis section of the safety analysis report (SAR) using best-estimate methods. The vendor or applicant/licensee shall demonstrate adequate diversity within the design for each of those events.*
3. *If a postulated common-mode failure could disable a safety function, then a diverse means, with a documented basis that the diverse means is unlikely to be subject to the same common-mode failure, should be required to perform either the same function or a different function. The diverse or different function may be performed by a non-safety system if the system is of sufficient quality to perform the necessary function under the associated event conditions.*

The two principle factors for defense against common-mode/common-cause failures are quality and diversity. Maintaining high quality increases the reliability of both individual components and complete systems. The TXS system quality has been described in Sections 2.1.3 and 2.2.1 of this safety evaluation. The staff requested Siemens to provide additional information to address the defense-in-depth and diversity in the TXS design. By letter NRC:99:037, dated September 1, 1999, Siemens submitted Technical Report EMF-2267(P), "Siemens Power Corporation Methodology Report for Diversity and Defense-in-Depth." The report addresses diversity and defense-in-depth for applications involving replacement of obsolete analog instrumentation in operating nuclear power plants.

The Siemens methodology follows acceptance criteria stated in SRP BTP HICB-19 and recommends that the applicant/licensee's diversity and defense-in-depth analysis follow the detailed guidance of NUREG/CR-6303, "Method for Performing Diversity and Defense-in-Depth Analyses of Reactor Protection Systems."

By letter NRC:00:004, dated January 13, 2000, Siemens submitted proprietary report EMF-2340(P), Revision 0, "Siemens Power Corporation Typical Diversity and Defense-in-Depth Assessment in Accordance with the Methodology of EMF-2267(P)." The purpose of this assessment is to demonstrate a typical PWR I&C upgrade with a TXS platform. The report defines a primary protection signal as the one which the protection function first occurs in the licensing basis analysis, and the backup protection signal (or signals) as those to occur if the primary protection function fail to actuate.

After evaluating each event, the report divides events into four categories:

- Category 1 Events that do not rely on functions processed by TXS for either primary or backup mitigation.
- Category 2 Events that do not rely on TXS for primary mitigation, but have a backup actuation processed by TXS.
- Category 3 Events that credit a TXS function for primary mitigation, but receive backup t6d backup mitigation. Operator action would be necessary to initiate manual actuations for these events. The indications and information available to allow the operator to initiate the appropriate actions are discussed in each event analysis.

The Siemens diversity and defense-in-depth methodology credits the inherent diversity between the TXS and TXP (control system) products. During a meeting at NRC on October 15, 1999, Siemens made a detailed presentation on the diversity between the TXS and TXP systems and the following are the major differences between the two systems:

- The design architecture are completely different.
- The design organization, management, designers, programmers, and testing engineers are different.
- The microprocessor CPU, input/output circuit boards and bus structure are from different manufacturers.
- The AC/DC power supplies and DC/DC power supplies are from different manufacturers.
- The computer languages are different.
- The software operating systems are different.
- The software development tools are different.
- The software validation tools are different.
- The software algorithms, logic, program architecture, timing, and order of execution are different.
- The application programs are functionally diverse.

On the basis of its review, the staff found that Siemens' diversity and defense-in-depth assessment methodology as described in EMF-2267(P) is consistent with the staff position stated in SRP BTP HICB-19 and is, therefore, acceptable. Applications following this methodology for a plant-specific diversity and defense-in-depth assessment should be found acceptable for this area. The actual review and verification of the major differences between the TXS and TXP was outside the scope of staff review. This is a plant-specific action item.

## 4.2 Surveillance Testing of the TXS System

By letter NRC:99:056, dated December 28, 1999, Siemens submitted report EMF-2341(P), "Generic Strategy for Periodic Surveillance Testing of TELEPERM XS Systems in U.S. Nuclear Generating Stations," for staff review. By letter NRC:00:017 dated March 3, 2000, Siemens provided additional clarification on recommended periodic surveillance test requirements for TXS applications. The report describes measures to be implemented in safety I&C systems configured with a TXS architecture to comply with requirements for channel checks, functional tests, channel calibration verification tests, response time verification tests, and logic system functional tests.

The measures include:

- Periodic verification (during refueling outages) of accuracy and time constants of the analog input modules.
- Continuous self-monitoring and on-line diagnostics to verify proper functioning of digital systems and to ensure integrity of the installed application and system software.
- Periodic actuation of output channel interposing relays. The reactor trip function is tested at the same surveillance test interval as current technical specifications (typically quarterly) and the engineered safety features actuation system (ESFAS) function is tested consistent with the licensee's refueling outage (typically 15 to 24 months).

As defined in the ALWR Standard Technical Specifications, a logic system functional test is a test of all required logic components (i.e., all required relays and contacts, trip functions, solid-state logic elements, etc.) of a logic path, from as close to the sensor as practicable up to, but not including, the actuated device, to verify operability. The logic system functional test may be performed by means of any series of sequential, overlapping, or total system steps so that the entire logic system is tested.

For some applications, interposing relays may be used in the logic component. The licensee should test those relays in accordance with the existing TS requirements. It is prudent to verify the logic system functions at least every refueling outage. This is a plant-specific action item along with the plant-specific technical specification requirements.

## 4.3 System Response Time Test

The TXS system response time testing is performed in overlapping steps:

- Verification of time constants of the input channels during input module tests, and
- Verification of the signal propagation time within the digital system

The accuracy and response time of the analog input channels are tested periodically by injection of test signals in the input circuits. An external test computer will be temporarily connected to the I&C system via permanently installed test plugs. While the input from the process is deactivated, calibrated electrical signals are generated and acquired. The digitized values are transmitted within the TXS system to the service unit and fed back to the test computer via a local network connection within the cabinet.

To verify the signal propagation time, two system properties of the TXS permit testing of the response time during operation without affecting the safety function:

- All signals are transferred from one computing node to the next strictly cyclically (it will never stop or wait for incoming data). Thus, even if the response time varies, it is the same for all signals using the same communication path.
- Actuation of one application function does not affect other application functions running on the same hardware resources.

As a means for verifying the reaction time of the logic, a binary input is provided to the data acquisition computers. Signal distribution to other computers is designed in the application software (functional diagrams) in the same way as for the normal measuring signals. Separate outputs are provided in the voting computers for each path. During periodic tests, the test machine connected to the I&C system generates a start signal and measures the reaction time of each signal path separately to verify that it does not exceed the worst case conditions specified for the specific system configuration. The measurements are performed a number of times to determine the statistical characteristics of each signal path.

BTP HICB-21, "Guidance on Digital Computer Real-Time Performance," states that digital system architecture affects the performance because communication between components of the system takes time, and allocation of functions to various system components affects timing. The architecture may also affect timing because an arrangement of otherwise simple components may have unexpected interactions. Specific timing requirements may affect the system architecture because it may not be possible to get sufficient computational performance for a specific function or group of functions from a single processor.

The staff has reviewed the TXS system architecture and the system response time test methodology as discussed in report EMF-2341(P) and found the TXS system design consistent with BTP HICB-21. It is, therefore, acceptable.

The protection system response time tests will be performed plant-specifically by licensees in accordance with plant technical specification requirements. The licensee must evaluate plant-specific accident analyses to confirm that a TXS reactor trip system (RTS) includes the provision to detect accident conditions and anticipated operational occurrences in order to initiate reactor shutdown (safety analysis confirmation for accuracy and time response) consistent with the accident analysis presented in Chapter 15 of the plant safety analysis report. This is a plant-specific action item.

#### 4.4 Software Development Life Cycle

The software development life cycle used at Siemens for the TXS system is based on the waterfall life cycle described in various software development standards. The life cycle consists of the system specification phase, the functional specification phase, the detailed design description phase, the implementation phase, the test specification phase, and the test phase. The remaining phases (i.e., the integration phase and the operation and maintenance phase) are plant-specific. The licensee's plant-specific software development procedures must be equivalent to industry standards and practices endorsed by the NRC.

The adequacy of the Siemens development process was reviewed by the staff during an audit at the vendor facility. The staff conducted a life cycle process audit of the TXS by tracing three requirements through the software life cycle. The first audit of requirements conducted by the staff involved change request 200 (CR200), which implemented the S706 input/output driver, version 1.20. The staff reviewed the documentation supporting the life cycle phases of system specifications, functional specifications, detailed design specifications, implementation (by code review), test specifications, test results, and the certification update. In the review of the test specifications and the test results for consistency with test specifications, the staff concluded that the development of the S706 input/output driver was consistent with the Siemens software life cycle processes and, therefore, was acceptable.

The second audit of requirements conducted by the staff involved the 2.MIN function module life cycle processes. The purpose of this software module is to select a signal from a group of signals for subsequent use in a function group. The staff found one discrepancy in the ANSI C source codes in which the comment for one requirement was not changed when the CASE/SWITCH was changed to incorporate a DEFAULT-label in the source code. The discrepancy had been noted in the IV&V comments, but had not been incorporated into the module source code documentation because the change did not affect the function of the module. The correction to the comment has been listed as an action item for the next revision to the source code. The staff found the decision to delay action until the next revision to the source code to be appropriate. The staff suggested to Siemens that the Siemens coding guidelines should be followed for documentation as well as for actual coding. The rest of the source code documentation was consistent with the corresponding source code.

The third audit of requirements conducted by the staff involved CR203, which addressed the process by which data may be entered in a SPACE function block module diagram. The change was required to remove the capability to enter hardware-specific parameter information from the network diagram parameter entry dialog screen. The change request was processed according to Siemens procedures, following all of the Siemens requirements, including verification and validation.

The staff noted that Siemens does not have in place a requirements traceability matrix (RTM) for enumerating and tracking each system requirement throughout its life cycle. There are no standards that require the use of an RTM, but the practice of enumerating each requirement does assist in tracking requirements during future modifications. There was a discussion regarding the use of an RTM, and the possible creation of an RTM for future development efforts. There is no outstanding commitment or action item regarding this matter, however, the need for an RTM will be pursued as part of assessing future modifications on plant-specific applications of the TXS.

During the staff audit, the staff discussed the development history of the application-specific integrated circuit (ASIC) chip that is used for the system support controller (SSC) on the SVE1 central processor unit (CPU) board and on the SCP1 communication processor board. The SSC was developed approximately 10 years ago for general applications (not just the TXS) by another subsidiary of Siemens. Consequently, documentation of the SSC development was not available for review. On the basis of known broad usage, documented failure data, and ability to identify critical characteristics, Siemens commercially dedicated the SSC for applications in the TXS system.

There have been three SSC chip failures on VE286 and VE386 processor modules. These older modules were used in industrial automation computers of industries other than the nuclear power industry. Five years ago, the Siemens subsidiary that manufactures the SSC adopted a more detailed recording system to document failures. However, these three failures occurred more than five years ago and are not documented under the more detailed system that has been in place since 1994. The faults were identified as complete failures of the SSC chip as opposed to isolated internal logic anomalies. There has been no redesign of the SSC logic as a consequence of these faults. Siemens concluded that these initial three failures were random physical occurrences. The staff finds this conclusion acceptable.

The only failure recorded in 2688 nuclear industry applications was on an SVE1 module. Siemens determined that this failure was caused by a soldering defect at one of the pins of the SSC and was not a failure of the SSC chip itself. The staff, therefore, finds the commercial dedication of the SSC chip to be acceptable.

The staff reviewed the Siemens configuration management process to confirm that various versions of documentation are properly controlled, and did not identify any discrepancies.

On the basis of its review of the software processes used throughout the software life cycle, and on the basis of its audit of software development documentation, the staff concludes that the software development process used at Siemens is acceptable.

The independent verification and validation (IV&V) process for the Siemens TXS is consistent with the IV&V process described in IEEE 1012-1998. Two organizations (iSTec and TÜV Nord) perform IV&V activities through a contract with the GRS. These two organizations report their findings to GRS and to Siemens, and also provide formal certification for each product. If a certified product requires modification, the modified product is submitted to iSTec or TÜV Nord for IV&V for a new certification. The staff concludes that the Siemens IV&V effort is sufficiently independent in personnel, management, and financial resources.

The IV&V processes address all phases of the Siemens software life cycle up to the testing of plant-specific applications. The staff did not address plant-specific applications of IV&V activities, as these activities were not in the scope of the staff review.

On the basis of its review of the Siemens engineering procedures and the results of its audit of Siemens software development processes, the staff concludes that Siemens has an acceptable software development methodology and follows this methodology consistently in developing safety-related software. The staff also determines that SPACE (specification and coding environment) tool for designing and assembling safety-related applications has the capability and safeguards to ensure that the implementation of the application programs can be successfully accomplished on a plant-specific basis.

## 5.0 SUMMARY OF REGULATORY COMPLIANCE EVALUATIONS

This safety evaluation discussed the acceptability of the TXS system. The general design criteria (GDC) listed in 10 CFR Part 50 Appendix A establish minimum requirements for the design of nuclear power plants. IEEE-603 is also incorporated in 10 CFR 50.55a(h). The regulatory guides and the endorsed industry codes and standards listed in NUREG-0800, "Standard Review Plan," Table 7-1, are the guidelines used as the basis for this evaluation.

Three Mile Island (TMI) Action Plan requirements for I&C systems are also identified in Table 7-1 of the SRP. Siemens has used a number of German standards in addition to the standards listed in the SRP.

Section 50.55a(a)(1) of Title 10 of the Code of Federal Regulations (10 CFR), "Quality Standards for Systems Important to Safety", is addressed by conformance with the codes and standards listed in the SRP. Siemens uses codes and standards in the development of the TXS system that are the same as or equivalent to the standards in the SRP and, therefore, the TXS system is in conformance with this requirement.

Section 50.55a(h) of 10 CFR endorses IEEE-603, which addresses both system level design issues and quality criteria for qualifying devices. Siemens has addressed these issues in the topical report. The TXS system meets the criteria of IEEE-603 and the supplemental standard IEEE-7-4.3.2-1996. The staff concludes, therefore, that the TXS system is in compliance with this requirement.

Section 50.34(f)(2)(v) of 10 CFR, "Bypass and Inoperable Status Indication (TMI Action Plan Item 1.D.3)," has been addressed in the TXS system design and is, therefore, in conformance with this requirement.

Section 50.34(f)(2)(xii) of 10 CFR, "Auxiliary Feedwater System Automatic Initiation and Flow Indication (TMI Action Plan Item II.E.1.2)," is a plant system requirement and is not specifically addressed in the topical report; however, the TXS system has the capability of meeting this requirement. Plant-specific application should demonstrate the compliance with this requirement.

Section 50.34(f)(2)(xvii) of 10 CFR, "Accident Monitoring Instrumentation (TMI Action Plan Item II.F.1)," has sampling and analyzing requirements. The TXS system is capable of providing plant process data for the display portion of this requirement. Plant-specific application should demonstrate the compliance with this requirement.

Section 50.34(f)(2)(xviii) of 10 CFR, "Instrumentation for the Detection of Inadequate Core Cooling (TMI Action Plan Item II.F.2)," is not specifically addressed in the topical report; however, the TXS system is capable of meeting the processing and display portions of this requirement. If an inadequate core cooling detection system is supported by the TXS system, then a plant-specific application should demonstrate the compliance with this requirement.

Section 50.34(f)(2)(xiv) of 10 CFR, "Containment Isolation Systems (TMI Action Plan Item II.E.4.2)," is not specifically addressed in the topical report; however, the TXS is capable of meeting the processing and display portions of this requirement. A plant-specific application should demonstrate the compliance with this requirement.

Section 50.34(f)(2)(xix) of 10 CFR, "Instrumentation for Monitoring Plant Conditions Following Core Damage (TMI Action Plan Item II.F.3)," is not specifically addressed in the topical report; however, the TXS is capable of meeting the processing and display portions of this requirement. A plant-specific application should demonstrate the compliance with this requirement.

Section 50.34(f)(2)(xx) of 10 CFR, "Power for Pressurizer Level Indication and Controls for Pressurizer Relief and Block Valves (TMI Action Plan Item II.G.1)," requires that the indicators continue to operate during a loss of offsite power. The TXS system is capable of providing the indication and controls. A plant-specific application should demonstrate the compliance with this requirement.

Section 50.34(f)(2)(xxiv) of 10 CFR, "Central Reactor Vessel Water Level Recording (TMI Action Plan Item II.K.3.23)," requires that reactor vessel water level be monitored during post-accident conditions. The TXS system can satisfy this requirement. A plant-specific application should demonstrate the compliance with this requirement.

Section 50.62 of 10 CFR, "Specifies Requirements for Reduction of Risk from Anticipated Transients Without Scram (ATWS)." The TXS system is acceptable for the ATWS mitigation system; however, the reactor protection system would have to be diverse from the ATWS mitigation system. Consequently, if a licensee develops an ATWS mitigation system based on the TXS system, the licensee must show that the system is diverse from the reactor protection system to receive staff approval.

The following 10 CFR Part 50, Appendix A, general design criteria are the applicable design criteria for this review:

- Criterion 1 - quality standards and records
- Criterion 4 - environmental and missile design bases
- Criterion 13 - instrumentation and control
- Criterion 20 - protection system functions
- Criterion 21 - protection system reliability and testability
- Criterion 22 - protection system independence
- Criterion 23 - protection system failure modes
- Criterion 24 - separation of protection and control systems

The following regulatory guides (RGs) are applicable to this review:

- RG 1.22, "Periodic Testing of Protection System Actuation Functions"
- RG 1.47, "Bypassed and Inoperable Status Indication for Nuclear Power Plant Safety Systems"
- RG 1.62, "Manual Initiation of Protection Action"
- RG 1.75, "Physical Independence of Electrical Systems"
- RG 1.97, "Instrumentation for Light-Water-Cooled Nuclear Power Plants to Assess Plant and Environs Conditions During and Following an Accident"
- RG 1.118, "Periodic Testing of Electric Power and Protection Systems"
- RG 1.152, "Criteria for Digital Computers in Safety Systems of Nuclear Power Plants"
- RG 1.153, "Criteria for Power Instrumentation and Control Portions of Safety Systems"

- RG 1.168, "Verification, Validation, Reviews and Audits for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"
- RG 1.169, "Configuration Management Plans for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"
- RG 1.170, "Software Test Documentation for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"
- RG 1.171, "Software Unit Testing for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"
- RG 1.172, "Software Requirements Specification for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"
- RG 1.173, "Developing Software Life Cycle Processes for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"

SRP Section 7.1-C provides guidance for evaluation of conformance to IEEE-603. IEEE-603 provides criteria for I&C systems in general. Reference is made to IEEE-7-4.3.2 for hardware and software issues of digital computers.

To meet the single-failure criterion for U.S. applications, the TXS is applied to four redundant process channels and two trip logic trains for each RPS or ESF actuation function. These redundant channels and trains are electrically isolated and physically separated. Qualified isolation devices have been tested to ensure functional operability when subjected to physical damage, short circuits, open circuits, or credible fault voltages on the device output terminals.

The completion of protective action requirement has been satisfied. Once initiated with the TXS system, the RPS and ESF actuations proceed to completion. Return to normal operation requires deliberate operator action to reset the reactor trip breakers. The reactor trip breakers cannot be reset while a reactor trip signal is present in the safety system. ESF actuations proceed to completion unless deliberate operator action is taken to terminate the function. The design is implemented consistent with plant specific functional logic to enable system-level protective actions to proceed to completion.

The quality criterion is satisfied with the Siemens Power Corporation Quality Assurance Program that meets the requirements of 10 CFR Part 50, Appendix B.

TXS is environmentally and seismically qualified to ensure the system is capable of performing its designated functions while exposed to normal, abnormal, test, accident and post-accident environmental conditions. The type testing was performed in accordance with KTA-3503. Mild environment qualification conforms to the guidance of IEEE-323, "IEEE Standard for Qualifying Class IE Equipment for Nuclear Power Generating Stations." EMC has been verified in accordance with the IEC-61000 standards. However, plant-specific environmental, seismic, and EMI qualification will be retested to be consistent with the guidance of EPRI TR-107330, and TR-102323, as stated in Sections 2.1.2.1, 2.1.2.2 and 2.1.2.3 of this report.

The independence criterion in the TXS system is met through the redundancy and separation of the channels. The communication between channels is via fiber optic cable.

The capability for testing and calibration has been demonstrated in compliance with RG 1.22, RG 1.118, and IEEE-338. The capability exists to permit testing during power operation. The design does not require disconnecting wires, installing jumpers, or other similar modifications of the installed equipment.

Access to the hardware is controlled via the front and rear cabinet doors, which are normally locked. Door positions are monitored, with an alarm to the operator if any door is opened.

The human factors considerations will be evaluated on a plant-specific basis and, therefore, are not included in this review.

Reliability has been assessed with both probabilistic and deterministic reliability analyses. The probabilistic analysis has been used to quantify the nonavailability on demand. The staff has reviewed these calculations; however, the staff does not use probabilistic and deterministic reliability analyses as the sole means of determining acceptability of a safety system. The calculations are related only to the hardware aspects of the TXS system; however, confirmatory testing performed by Siemens and GRS included the software. The deterministic analysis based on codes and standards delineates postulated failures that the system will be able to withstand.

The TXS meets the automatic and manual control requirements. Failure of the automatic controls does not interfere with the manual controls.

Setpoints will be evaluated on a plant-specific basis. The licensee must ensure that, when the TXS system is installed, overly conservative setpoints due to the elimination of analog system drift are not retained, as this would increase the possibility that the TXS equipment may be performing outside the vendor specifications.

The NRC staff concludes that the design of the TXS safety systems meets the relevant requirements of General Design Criteria (GDC) 1, 2, 4, 13, 19-25, and 29, and 10 CFR 50.34(f), 10 CFR 50.55a(a)(1), and 10 CFR 50.55a(h), and is, therefore, acceptable.

The staff conducted a review of the safety system descriptions in the topical report for conformance to the guidelines in the regulatory guides and industry codes and standards applicable to these systems. The staff concludes that the applicant adequately identified the guidelines applicable to these systems. Based upon the review of the safety system designs for conformance to the guidelines, the staff finds that there is reasonable assurance that the TXS system conforms to the guidelines applicable to these systems. Therefore the staff finds that the requirements of GDC 1 and 10 CFR 50.55a(a)(1) have been met.

The review included the identification of those systems and components for the safety systems designed to survive the effects of earthquakes, other natural phenomena, abnormal environments, and missiles. On the basis of this review, the staff concludes that the applicant has identified those systems and components consistent with the design bases for those systems. Therefore, the staff finds that the identification of these systems and components satisfies the requirements of GDC 2 and 4.

Based on the review of safety system status information, manual initiation capabilities, and provisions to support safe shutdown, the staff concludes that information is provided to monitor the safety systems over the anticipated ranges for normal operation, for anticipated operational occurrences, and for accident conditions so as to ensure adequate safety. Appropriate controls are provided for manual initiation of a reactor trip. The TXS safety systems appropriately support actions to operate the nuclear power unit safety under normal conditions and to maintain it in a safe condition under accident conditions. Therefore, the staff finds that the TXS system designs satisfy the requirements of GDC 13 and 19.

Based on the review of system functions, the staff concludes that a TXS system conforms to the design bases requirements of IEEE Std 603 and 10 CFR 50.34(f) and to the guidance of RG 1.105. On the basis of its review, the staff concludes that the TXS includes the provision to detect accident conditions and anticipated operational occurrences in order to initiate reactor shutdown consistent with the accident analysis presented in Chapter 15 of the plant's Safety Analysis Report (SAR). Licensee evaluation of plant-specific accident analyses is required. Therefore, the staff finds that the TXS satisfies the requirements of GDC 20.

The TXS system conforms to the guidelines for periodic testing in RG 1.22 and RG 1.118. The bypassed and inoperable status indication conforms to the guidelines of RG 1.47. The safety systems conform to the guidelines on the application of the single-failure criterion in ANSI/IEEE Std 379, as supplemented by RG 1.53. On the basis of this review, the staff concludes that the TXS system satisfies the requirement of IEEE-603 with regard to system reliability and testability. Therefore, the staff finds that the TXS system satisfies the requirements of GDC 21.

The TXS system conforms to the guidelines in RG 1.75 for protection system independence. On the basis of its review, the staff concludes that the TXS system satisfies the requirement of IEEE-603 with regard to system independence. Therefore, the staff finds that the TXS system satisfies the requirements of GDC 22.

On the basis of its review of the failure modes and effects analysis for the TXS system, the staff concludes that the systems are designed to fail into a safe mode if conditions such as disconnection of the system, loss of energy, or adverse environment are experienced. Therefore, the staff finds that the TXS system satisfies the requirements of GDC 23.

Based on its review of the interfaces between the TXS safety systems and plant operating control systems, the staff concludes that the TXS safety systems satisfy the requirements of IEEE-603 with regard to control and protection system interactions. Therefore, the staff finds the TXS safety systems satisfy the requirements of GDC 24.

On the basis of its review of all the above GDCs, the staff concludes that the TXS system satisfies the requirements of GDC 29, "Protection Against Anticipated Operational Occurrences."

The staff's conclusions are based upon the requirements of ANSI/IEEE-603 with respect to the design of the TXS system. Therefore, the staff finds that the TXS system satisfies the requirement of 10 CFR 50.55a(h) with regard to ANSI/IEEE-603.

On the basis of its review of the Siemens defense-in-depth and diversity analysis methodology, the staff concludes that licensees implementing this methodology will comply with the criteria for defense against common-mode failure in digital instrumentation and control systems. Therefore, the staff finds that adequate diversity and defense against common-mode failure will be provided to satisfy these requirements of GDC 21 and 22 and Item II.Q of the staff requirements memorandum of SECY-93-087. The staff requires, however, that each licensee ensure that the plant-specific application complies with the criteria for defense against common-mode failures in digital instrumentation and control systems.

On the basis of its review of software development plans and inspections of the computer development process and design outputs, the staff concludes that the TXS safety systems meet the guidance of RG 1.152. Therefore, the special characteristics of computer systems have been adequately addressed, and the staff finds that the TXS system satisfies the requirements of GDC 1 and 21.

The staff concludes that the TXS system meets the requirements of 10 CFR Part 50, Appendix A General Design Criteria 1, 2, 4, 13, 19-24, and 29, and IEEE-603 for the design of safety-related reactor protection systems, engineered safety features systems, and other plant systems, and the guidelines of Regulatory Guide 1.152 and supporting industry standards for the design of digital systems and is, therefore, acceptable.

The design principle for software of Class 1E systems is to ensure that the sequence of processing executed for each expected situation can be deterministically established. It discourages the use of non-deterministic data, communications, non-deterministic computations, multitasking, dynamic scheduling, use of non-deterministic interrupts and event-driven designs. Based on its review, the staff determines that the design of the TXS system satisfies this design principle for Class 1E system software.

## 6.0 PLANT-SPECIFIC ACTION ITEMS

On the basis of the above review, the staff concludes that the TXS system is acceptable for use in the development, installation and operation of safety-related systems in nuclear power plants, subject to the following conditions:

The following actions must be performed by an applicant when requesting NRC approval for installation of a Siemens TXS system:

1. The licensee must demonstrate that the generic qualification bounds the plant specific condition (i.e., temperature, humidity, seismic, and electromagnetic compatibility) for the location(s) in which the TXS equipment is to be installed. The generic qualification data must comply with EPRI qualification requirements specified in EPRI TR-107330 and TR-102323-R1 (see Sections 2.1.2.1, 2.1.2.2, and 2.1.2.3).
2. The licensee's plant-specific software development V&V activities and configuration management procedures must be equivalent to industry standards and practices endorsed by the NRC (as referenced in SRP BTP HICB-14, "Guidance on Software Reviews for Digital Computer-Based Instrumentation and Control Systems") (see Sections 4.4, 2.2.3, 2.2.4).

3. If the licensee develops a TXS auxiliary feedwater control system, the licensee must include automatic initiation and flow indication (TMI Action Plan Item II.E.1.2). The licensee needs to confirm that the plant-specific application conforms to the requirements of 10 CFR 50.34 (f)g(2)(xii) (see Section 5.0).
4. If the licensee replaces existing accident monitoring instrumentation (TMI Action Plan Item II.F.1) display capabilities with a TXS system, including the bypass and inoperable status information, the licensee needs to confirm that the new system provides equivalent sampling and analyzing features, and meets the requirement of 10 CFR 50.34 (f)(2)(xvii) (see Section 5.0).
5. If the licensee installs a TXS inadequate core cooling detection system, the licensee needs to confirm that the new system conforms to the requirements of 10 CFR 50.34(f)(2)(xviii) (see Section 5.0).
6. If the licensee installs a TXS containment isolation system (TMI Action Plan Item II.E.4.2), the licensee must verify that the plant-specific application conforms to the requirement of 10 CFR 50.34 (f)(2)(xiv) (see Section 5.0).
7. For monitoring plant conditions following core damage, the licensee must verify that the TXS system meets the processing and display portions of the requirements of 10 CFR 50.34(f)(2)(xix) (see Section 5.0).
8. If the licensee installs a TXS system for monitoring reactor vessel water level during post -accident conditions, the licensee must provide plant-specific verification of the ranges, and confirm that human factors issues have been addressed, as required by 10 CFR 50.34(f)(2)(xxiv) (see Section 5.0).
9. If the licensee installs a TXS reactor protection system, the licensee must provide confirmation that the TXS system is diverse from the system for reducing the risk from anticipated transients without scram (ATWS), as required by 10 CFR 50.62. If the licensee installs a TXS ESFAS, the licensee must provide confirmation that the diversity requirements for plant systems (feedwater, auxiliary feedwater, turbine controls, etc.) are maintained (see Section 5.0).
10. Setpoints will be evaluated on a plant-specific basis. The licensee must ensure that, when the TXS system is installed, overly conservative setpoints that may occur due to the elimination of analog system drift are not retained, as this would increase the possibility that the TXS equipment may be performing outside the vendor specifications. The licensee must provide the staff with a revised setpoint analysis that is applicable to the installed TXS system(s) (see Section 4.4).
11. The licensee must evaluate plant-specific accident analyses to confirm that a TXS reactor trip system (RTS) includes the provision to detect accident conditions and anticipated operational occurrences in order to initiate reactor shutdown (safety analysis confirmation for accuracy and time response) consistent with the accident analysis presented in Chapter 15 of the plant safety analysis report (see Section 4.3).

12. The staff requires that each licensee ensure that the plant-specific TXS application complies with the criteria for defense against common-mode failures in digital instrumentation and control systems (see Section 4.1).
13. The licensee should propose plant-specific Technical Specifications including periodic test intervals (see Section 4.2).
14. The licensee should demonstrate that the power supply to the TXS system complies with EPRI TR-107330 requirements (see Section 2.1.2.4).
15. The licensee should demonstrate that the qualification of the isolation devices were performed in accordance with EPRI TR-107330 requirements (see Section 2.1.3).
16. The licensee should demonstrate that Siemens TXP (control systems) or other manufacturer's control systems satisfy the acceptance guidance set forth in Section 4.1 of this safety evaluation (see Section 4.1).
17. The licensee should address the need for a requirement traceability matrix (RTM) for enumerating and tracking each system requirement throughout its life cycle, particularly as part of making future modifications (see Section 4.4).

Principal Contributors: E. Lee  
M. Waterman  
H. Li

Date: May 5, 2000



UNITED STATES  
NUCLEAR REGULATORY COMMISSION

WASHINGTON, D.C. 20555-0001

September 30, 1999

Mr. James F. Mallay  
Director, Nuclear Regulatory Affairs  
Siemens Power Corporation  
2101 Horn Rapids Road  
Richland, WA 99352

**SUBJECT: REQUEST FOR ADDITIONAL INFORMATION ON SIEMENS TOPICAL REPORT, EMF-2110, REVISION 1, "TELEPERM XS: A DIGITAL REACTOR PROTECTION SYSTEM" (TAC NO. MA1983)**

Dear Mr. Mallay:

By letter dated September 1, 1999, the Siemens Power Corporation (SPC) submitted Revision 1 to Topical Report EMF-2110 (NP), "TELEPERM XS: A Digital Reactor Protection System" for staff review. On September 9, 1999, the staff proposed that Siemens use a staff generic digital Safety Evaluation (SE) outline to identify pertinent sections of the topical report that address information expected in sections of the outline and to the extent possible provide specific narrative that describes the information required to support conclusions of the SE. Areas not in the scope of the topical report should be clearly identified as plant specific. The staff's generic digital SE outline is enclosed.

*In addition, please provide a detailed diversity design comparison between TELEPERM XS (safety protection system) and TELEPERM XP (non-safety control system) to demonstrate the total diversity between the two systems.*

The enclosed request was discussed with your staff on September 23, 1999. A mutually agreeable target date of within 30 days of the date of this letter for your response was established. If circumstances result in the need to revise the target date, please call me at the earliest opportunity at 301-415-1480.

Sincerely,

A handwritten signature in black ink, appearing to read "N. Kalyanam", with a horizontal line underneath.

N. Kalyanam, Project Manager, Section 2  
Project Directorate IV & Decommissioning  
Division of Licensing Project Management  
Office of Nuclear Reactor Regulation

Project No. 702

Enclosure: Digital SE Outline

# SIEMENS

September 13, 1999  
NRC:99:039

Document Control Desk  
ATTN: Chief, Planning, Program and Management Support Branch  
U.S. Nuclear Regulatory Commission  
Washington, D.C. 20555-0001

## **Supporting Documentation for Review of EMF-2110(NP) Revision 1, "TELEPERM XS: A Digital Reactor Protection System"**

Ref.: 1. Letter, James F. Mallay (SPC) to Document Control Desk, "Same Subject," NRC:99:037, September 1, 1999.

Enclosed are five copies of each of (1) 12 proprietary engineering procedures and (2) EMF-2267(P), "Siemens Power Corporation Methodology Report for Diversity and Defense-in-Depth." A single copy of these documents had been provided to the NRC on September 1, 1999 (see Reference 1). The review of this documentation has been expanded and therefore additional copies have been requested. Please note that three copies of this documentation have been sent directly to the lead reviewer, Hulbert Li, and one copy to SPC's project manager, Nageswaran Kalyanam.

The information contained in the enclosures is held proprietary SPC. The affidavit provided with Reference 1 provides the necessary information to support the withholding of the enclosed documents from public disclosure.

Very truly yours,



James F. Mallay, Director  
Regulatory Affairs

/am

Enclosures

cc: Mr. N. Kalyanam (w/enclosures)  
Mr. H. C. Li (w/enclosures)

**Siemens Power Corporation**

2101 Horn Rapids Road  
Richland, WA 99352

Tel: (509) 375-8100  
Fax: (509) 375-8402

# SIEMENS

September 1, 1999  
NRC:99:036

Document Control Desk  
ATTN: Chief, Planning, Program and Management Support Branch  
U.S. Nuclear Regulatory Commission  
Washington, D.C. 20555-0001

## **Submittal of EMF-2110(NP) Revision 1, "TELEPERM XS: A Digital Reactor Protection System"**

Ref.: 1. Letter, James F. Mallay (SPC) to Document Control Desk, "Topical Report EMF-2110(NP),  
"Topical Report for Generic Approval of TELEPERM XS Equipment at US NRC,"  
NRC:98:067, September 23, 1998.

Enclosed are 15 copies of EMF-2110(NP) Revision 1, "TELEPERM XS: A Digital Reactor Protection System." (Note: One copy each has been provided to Mr. John Cushing and Mr. Hulbert Li.) The report was revised in response to NRC questions following a system audit in Germany, reviews of supporting documentation, and reviews of Revision 0 of the topical report.

Siemens Power Corporation submitted Revision 0 of topical report EMF-2110(NP) to the NRC for review last September (see Reference 1). Subsequently, an audit of the TELEPERM XS design and the verification and validation of the design was conducted by NRC Staff members at Siemens' facilities in Erlangen, Germany from September 28, 1998 through October 2, 1998. Siemens has also responded to numerous questions from members of the Instrumentation and Controls Branch of NRR. The enclosed topical report, which presents a detailed description of the TELEPERM XS Reactor Protection System and demonstrates its compliance with applicable NRC regulations and guidance, has been revised to incorporate the resolution of all comments received from the NRC.

Very truly yours,



James F. Mallay, Director  
Regulatory Affairs

/arn

Enclosures

cc: Mr. J. S. Cushing (w/enclosure)  
Mr. H. C. Li (w/enclosure)

**Siemens Power Corporation**

2101 Horn Rapids Road  
Richland, WA 99352

Tel: (509) 375-8100  
Fax: (509) 375-8402

EMF-2110(NP)(A)  
Revision 1

# TELEPERM XS: A Digital Reactor Protection System

September 1999

### Nature of Changes

<u>Item</u>	<u>Page</u>	<u>Description and Justification</u>
1.	All	This document has been rewritten in its entirety.

## Contents

1.0	Introduction.....	1-1
1.1	Background.....	1-1
1.2	Qualification .....	1-1
2.0	Adequacy of Design Criteria and Guidance .....	2-1
2.1	Quality Assurance Program .....	2-1
2.1.1	Introduction .....	2-1
2.1.2	Application Independent Qualification .....	2-3
2.1.2.1	Philosophy and Strategy .....	2-3
2.1.2.2	Concept Review .....	2-6
2.1.3	Type Tests .....	2-10
2.1.4	Summary of the Application Independent Qualification.....	2-10
2.1.5	Record of Documents Submitted for Software Type-Test.....	2-11
2.2	Equipment Qualification .....	2-11
2.2.1	Hardware Type-Test.....	2-11
2.2.1.1	Rules and Standards.....	2-11
2.2.1.2	Qualification Process .....	2-13
2.2.1.3	Scope of Components.....	2-19
2.2.1.4	Submitted Documents and Results .....	2-20
2.2.2	Hardware Test Requirement Specification .....	2-22
2.2.2.1	Introduction .....	2-22
2.2.2.2	Documentation Required for Type Testing .....	2-22
2.2.2.3	Practical Type Qualification.....	2-23
2.3	Channel Integrity, Isolation and Real-Time Performance .....	2-43
2.3.1	Isolation.....	2-43
2.3.2	Limiting Response Time.....	2-44
2.3.3	Digital Computer Timing Requirements .....	2-44
2.3.4	Architecture .....	2-44
2.4	Reliability .....	2-44
2.4.1	Fundamentals and Definitions .....	2-45
2.4.1.1	Design Errors, Faults and Failures .....	2-45
2.4.1.2	Masking of Faulty Signals .....	2-46
2.4.2	Mechanisms Leading to System Failures .....	2-46
2.4.2.1	System Failures Due to Component Failures .....	2-47
2.4.2.2	System Failures Due to Latent Faults .....	2-47
2.4.2.3	Summary of Assessment of Failure Mechanisms.....	2-50
2.4.3	Relevant System Characteristics.....	2-50
2.4.3.1	Basic Components of TELEPERM XS .....	2-50
2.4.3.2	Software Design of TELEPERM XS .....	2-54
2.4.3.3	Validation of the Application Software .....	2-59
2.4.3.4	Allocation of Tasks to the Software Components .....	2-61
2.4.3.5	Interference-Free Communication.....	2-65

2.5	Testability.....	2-65
2.5.1	Introduction .....	2-66
2.5.2	Tasks of the Service Unit .....	2-67
2.5.2.1	Supported Tasks of the Staff.....	2-67
2.5.2.2	Application Functions of the Service Unit .....	2-67
2.5.3	User Interfaces for Monitoring, Test and Diagnostic Services .....	2-71
2.5.3.1	Alphanumeric Interface .....	2-72
2.5.3.2	Simple Graphic Interface.....	2-72
2.5.3.3	Online Display of Function Diagrams .....	2-73
2.5.3.4	Online Display of Hardware Structure .....	2-74
2.5.4	Connection to the I&C System .....	2-74
2.5.4.1	Links to the Function Processors .....	2-74
2.5.4.2	Communication with the I&C System .....	2-75
2.5.5	Architecture of the Service Unit.....	2-76
2.5.5.1	Hardware Architecture .....	2-76
2.5.5.2	Software Architecture of the Service Unit .....	2-76
2.5.5.3	Technical Data .....	2-78
2.5.6	Handling and Use of the Service Unit.....	2-78
2.5.6.1	Logging in and Tool Selection .....	2-78
2.5.6.2	Monitoring of the I&C System.....	2-79
2.5.6.3	Conduct of Periodic Testing .....	2-79
2.5.6.4	Elaboration of Test Programs.....	2-79
2.5.6.5	Log-Off.....	2-79
2.5.7	Testing and Maintenance .....	2-80
2.6	Control of System Access .....	2-80
2.6.1	Objectives .....	2-80
2.6.2	Technical Measures .....	2-81
2.6.3	Additional Protection Against Errors .....	2-81
2.6.4	User Rights and Operating Modes .....	2-82
2.7	Fault Tolerance Features .....	2-84
2.7.1	Types of Component Failures and Their Effects.....	2-84
2.7.1.1	Inherent Mechanisms for Detecting and Signaling Failures .....	2-86
2.7.1.2	Configured Monitoring Mechanisms .....	2-89
2.7.1.3	Masking of Component Failures.....	2-90
2.7.1.4	Fail-Safe Failure Behavior.....	2-91
2.7.1.5	Summary .....	2-92
2.7.2	System Architecture and Component Failure Effects.....	2-92
2.7.2.1	Single-Level System Architecture with Fault Masking in the Switchgear .....	2-93
2.7.2.2	Two-Level Architecture with Fault Masking for Measured Value Acquisition.....	2-94
2.7.2.3	Three-Level Architecture with Fault Masking for Signal Processing .....	2-95
2.7.2.4	Summary on Architectures .....	2-96
2.8	Identification.....	2-96

2.9	Interference-Free Communication .....	2-97
2.9.1	Specification of the Requirements .....	2-97
2.9.2	Profibus Communication .....	2-98
2.9.3	Ethernet Communication .....	2-102
3.0	Software Life Cycle Process Planning .....	3-1
3.1	Functional Software Characteristics .....	3-1
3.1.1	Introduction .....	3-1
3.1.1.1	System Overview .....	3-1
3.1.1.2	Global Analysis .....	3-3
3.1.1.3	Technological Factors .....	3-4
3.1.1.4	Requirements and Architectural Challenges .....	3-5
3.1.1.5	System Design Guidelines .....	3-5
3.1.2	Conceptual Architecture .....	3-7
3.1.2.1	Central Design Tasks: Component Modularization, Connector Characterization, Instance Characterization.....	3-11
3.1.2.2	Resource Allocation .....	3-13
3.1.2.3	Summary of Conceptual Architecture .....	3-14
3.1.3	Module Architecture .....	3-16
3.1.3.1	Function Block Modules (FB) .....	3-17
3.1.3.2	Function Diagram Modules (FD) .....	3-17
3.1.3.3	Function Diagram Group Modules (FDG) .....	3-17
3.1.3.4	Runtime Environment (RTE) .....	3-18
3.1.3.5	Defining Layers .....	3-19
3.1.3.6	Interfaces of the Runtime Environment .....	3-20
3.1.4	Execution Architecture .....	3-23
3.1.4.1	Defining Executables .....	3-24
3.1.4.2	Communication .....	3-26
3.1.4.3	Configuration.....	3-27
3.1.5	Code Architecture .....	3-27
3.1.6	Design and Implementation.....	3-30
3.1.7	Descriptions and Uses .....	3-30
3.1.7.1	Architecture Description Techniques .....	3-30
3.1.7.2	Analysis .....	3-31
3.1.7.3	Software Architecture and Software Development.....	3-34
3.1.8	Summary.....	3-36
3.2	Software Development Process Characteristics.....	3-36
3.2.1	Software Type Test .....	3-36
3.2.1.1	Rules and Standards.....	3-37
3.2.1.2	Qualification Process .....	3-38
3.2.1.3	Scope of Components.....	3-43
3.2.1.4	Submitted Documents and Results .....	3-47
3.2.2	Integration and System Test.....	3-50
3.2.2.1	Rules and Standards.....	3-51
3.2.2.2	Goals of the Integration and System Tests.....	3-51
3.2.2.3	Qualification Process .....	3-54
3.2.2.4	Submitted Documents and Results .....	3-56

## TELEPERM XS: A Digital Reactor Protection System

4.0	Independence of Class 1E Equipment and Circuits .....	4-1
4.1	Typical Architecture of Safety I&C Systems .....	4-1
4.2	Design Principles .....	4-3
4.3	Signal Transmission from Class 1E to non 1E Equipment and Circuits.....	4-3
4.4	Single Wire Signal Transmission.....	4-4
4.5	Digital Data Transmission via Serial Busses .....	4-4
4.5.1	Communication With the Service Unit .....	4-5
4.6	Transmitters.....	4-7
4.7	Switchgear .....	4-7
4.8	Power Supply.....	4-7
4.9	Signal Transmission Between Redundant Class 1E Channels.....	4-7
5.0	Summary of Engineering Procedures and Project Instructions .....	5-1
5.1	Engineering Procedure, Software Life - Cycle Process .....	5-1
5.1.1	Pre-Development Process .....	5-1
5.1.2	The Development Process .....	5-1
5.1.3	Post-Development Process.....	5-2
5.1.4	Identification of Tools and Components .....	5-2
5.2	Engineering Procedure, Configuration Management Plan.....	5-3
5.2.1	Management.....	5-3
5.2.2	Configuration Control .....	5-3
5.2.3	Tools.....	5-4
5.3	Engineering Procedure, Guideline for Documentation.....	5-4
5.4	Engineering Procedure, Structure of Contents of Requirements Specifications of Hardware and Software Components.....	5-4
5.5	Engineering Procedure, Structure of Contents of Technical Design Specifications of Software Components.....	5-6
5.6	Engineering Procedure, Structure of the Contents of Detailed Design Specifications for Software Components.....	5-8
5.7	Engineering Procedure, Structure of the Contents of Implementation Specifications for Software Components.....	5-9
5.8	Engineering Procedure, Tests.....	5-10
5.9	Engineering Procedure, Reviews .....	5-11
5.10	Project Instruction No. 2, "Document Control for Software Development in the TELEPERM XS Project" .....	5-12
5.11	Supplemental Engineering Procedures .....	5-13
6.0	Industry Standards .....	6-1
6.1	IEC Standards.....	6-1
6.2	VDE Standards .....	6-2
6.3	IEEE Standards .....	6-3
6.4	KTA Standards.....	6-4
6.5	Code of Federal Regulations.....	6-4
6.6	U.S. NRC Regulatory Guides.....	6-4
6.7	Miscellaneous Standards .....	6-5
6.8	EN Standards.....	6-6

7.0	Conformance With IEEE Standards.....	7-1
7.1	Single-Failure Criterion .....	7-1
7.2	Completion of Protective Action .....	7-2
7.3	Quality.....	7-2
7.4	Equipment Qualification .....	7-2
7.5	System Integrity .....	7-3
7.6	Independence .....	7-3
7.7	Capability for Test and Calibration .....	7-4
7.8	Information Displays.....	7-4
7.9	Control of Access .....	7-5
7.10	Repair .....	7-5
7.11	Identification.....	7-6
7.12	Auxiliary Features .....	7-6
7.13	Human Factors Considerations .....	7-6
7.14	Reliability .....	7-6
7.15	Automatic Control .....	7-7
7.16	Manual Control.....	7-8
7.17	Interaction Between the Sense and Command Features and Other Systems .....	7-8
7.18	Derivation of System Inputs .....	7-9
7.19	Operating Bypasses.....	7-9
7.20	Maintenance Bypass.....	7-10
7.21	Setpoints.....	7-10
8.0	Qualification Documents.....	8-1
8.1	Software Type Test.....	8-1
8.1.1	Overall System.....	8-1
8.1.2	Procedures and Instructions.....	8-2
8.1.3	Function Blocks.....	8-2
8.1.4	Function Diagrams .....	8-4
8.1.5	Program Structure.....	8-5
8.1.6	Runtime Environment.....	8-5
8.1.7	Exception Handler.....	8-6
8.1.8	I/O Driver.....	8-6
8.1.9	Self Monitoring .....	8-7
8.1.10	Operating System .....	8-7
8.1.11	NMI Handler .....	8-7
8.1.12	Diagnostic Monitor .....	8-8
8.1.13	MicroNET .....	8-8
8.1.14	SPC1 .....	8-8
8.1.15	HOT .....	8-9
8.1.16	Debug Services.....	8-9
8.1.17	Driver 3964R.....	8-10

8.2	Hardware Type Test.....	8-10
8.2.1	Type Test of TELEPREM XS .....	8-10
8.2.1.1	General Documents for the Type Test .....	8-10
8.2.2	Module-Specific Documents for the Type Test .....	8-11
8.2.2.1	Processing Module.....	8-11
8.2.2.2	Communication Module LEBUS .....	8-12
8.2.2.3	Bus Interface Module LEBUS.....	8-13
8.2.2.4	Digital Input Module .....	8-14
8.2.2.5	Digital Input Module .....	8-14
8.2.2.6	Digital Output Module.....	8-15
8.2.2.7	Digital Output Module, Relays.....	8-15
8.2.2.8	Analog Input Module, Integrating.....	8-16
8.2.2.9	Analog Input Module .....	8-17
8.2.2.10	Analog Output Module .....	8-17
8.2.2.11	Counter Module .....	8-18
8.2.2.12	Relay Module .....	8-19
8.2.2.13	Communication Processor .....	8-19
8.2.2.14	Twin-Transceiver .....	8-20
8.2.2.15	Optical Transceiver .....	8-20
8.2.2.16	Fiber-Optic Transceiver.....	8-21
8.2.2.17	Active Star Coupler .....	8-21
8.2.2.18	Kommunikationsmodule L2 .....	8-22
8.2.2.19	SLLM L2 Link Module .....	8-23
8.2.2.20	Subrack.....	8-23
8.2.2.21	Subrack.....	8-24

## Figures

1.1	Standards Applied in the Qualification of TELEPERM XS.....	1-3
2.1	Automatic Software Generation by the SPACE Engineering System .....	2-3
2.2	Qualification Concept of TELEPERM XS .....	2-5
2.3	Scope of the Hardware Type Test .....	2-14
2.4	States of Systems with Latent Faults .....	2-45
2.5	TELEPERM XS Software Architecture.....	2-55
2.6	TELEPERM XS Development Process .....	2-57
2.7	Engineering Process for Developing the Application Software.....	2-59
2.8	Comparison of the Procedure for Designing Hardwired and Digital I&C Systems.....	2-60
2.9	Typical Structure of a TELEPERM XS System .....	2-66
2.10	Examples for the Display of Dynamic Function Diagrams and Hardware Diagrams .....	2-74
2.11	Software Architecture of the Service Unit UI = User Interface .....	2-77
2.12	Table of Privileges .....	2-83
2.13	Simple Safety Function.....	2-93
2.14	Single-Level Architecture with Majority Voting in the Switchgear .....	2-93
2.15	Two-Level Architecture .....	2-95
2.16	Three-Level Architecture with Voter Configuration.....	2-96
2.17	Communication Channels.....	2-98
2.18	Interference-Free L2 Communication .....	2-99
2.19	Processing Cycle .....	2-101
2.21	Interference-Free H1 Communication.....	2-103
3.1	Hardware Topology of a Typical TELEPERM XS Application .....	3-2
3.2	Example of a Function Diagram (FD).....	3-9
3.3	Example of a Hardware Specification Diagram .....	3-10
3.4	Specification Process of a TELEPERM XS Application.....	3-14
3.5	Software Layers of the Runtime System of One Processing Module .....	3-20
3.6	Dependencies of the Runtime-Environment.....	3-23
3.7	Execution Architecture.....	3-24
3.8	Task Scheduling .....	3-26
3.9	Code Architecture.....	3-29
3.10	TELEPERM XS Software Production.....	3-32
3.11	Verification of TELEPERM XS Software Specification .....	3-32
3.12	Verification of TELEPERM XS Code Generation .....	3-33
3.13	Validation of a TELEPERM XS Application System .....	3-34
3.14	Phase Model and Associated Engineering Procedures.....	3-39
3.15	Type Tested Software Components of TELEPERM XS .....	3-45
3.16	List of Agreed-Upon Non-Conformities to Standard DIN IEC 880 .....	3-50
3.17	Architecture of the Limitation and Control System at Unterweser .....	3-55
4.1	Typical Architecture of an I&C Safety System.....	4-1
4.2	Architecture With Communication Links Crossing Channels.....	4-2
4.3	Architecture With Service Unit and Gateway.....	4-2
4.4	Isolation of Single Wired Signal Transmission .....	4-4
4.5	Release Signals to Enable Change of Mode of Operation .....	4-6
8.1	Type of Documents for Overall System .....	8-1
8.2	Type of Documents for Each Component (Software Type Test) .....	8-1

## Nomenclature

ANS	American National Standard
AU	“Ablaufumgebung” = runtime environment
BNF	Bacchus Naur Form
BMU	“Bundesministerium für Umwelt”) = German Environmental Department
C0,C1	test coverage metrics
CCF	Common Cause Failure
CPU	Central Processing Unit
CRC	Cyclic Redundancy Checks
DBS	Data Base Server
DPRAM	Dual Port RAM
EBNF	Extended Bacchus Naur Form
EEPROM	Electrical Erasable Programmable Read Only Memory
EMC	Electromagnetic compatibility
FB	Function Block
FD	Function Diagram
FDG	Function Diagram Group
FEPRAM	Flash Erasable Programmable Read Only Memory
FO	Fiber optic
GRS	Gesellschaft für ReaktorSicherheit = Reactor Safety Association
HOT	Hardware Organization Tool for set-up- and interface
I&C	Instrumentation and Control
I/O	Input / Output
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
INGRES	a data base system
ISO	International Standardization Organization
ISTec	Institute for Safety Technology
KKS	Kraftwerks Kennzeichnungen System = Identification System for Power Plants
KTA	KernTechnischer Ausschuß = German Committee for Nuclear Technologies
LAN	Local Area Network
LAx	communication module for LAN
LEFU	Elementary safety function (LEittechnik FUNKtion)
MicroNET	communication software
MICROS	operating system
MSI	Message and Service Interface

NPP	Nuclear Power Plant
NRC	U.S. Nuclear Regulatory Commission
OSI	Open System Interconnection
PBC	Printed-circuit Board
PPIS	Plant Process Information System
Profibus	Process Field Bus
QA	Quality Assurance
RAM	Random Access Memory
RSK	"Reaktor Sicherheitskommission" = German Commission for Reactor Safety
RTE	Runtime Environment
RTS	Realtime Transport System
SA/SD	Structured Analysis/Structured Design
SAE	Cable connection element
SBU1	Bus interface module for local bus
SCP1	Communication processor for SINEC H1
SHO1	Optical mini-transceiver for SINEC H1
SHO2	Plug-in optical transceiver for optical star
SHS1	Active optic star coupler for SINEC H1
SHT1	Ethernet transceiver with two interfaces for SINEC H1
SHT2	Ethernet transceiver module for optical star
SKO1	Communication module for local bus
SL21	Communication processor for SINEC L2SPACE Specification and Coding Environment
SMS	Service Monitor Server
SPACE	Specification And Coding Environment (function diagram editor)
SQL	Standard Query Language
SRP	Standard Review Plan
SVE1	Processing module
TCP/IP	
TÜV	Technischer ÜberwachungsVerein Technical Inspection Agency
TXS	TELEPERM® XS
UI	User Interface
VGB	Verband der Großkraftwerksbetreiber (Association of Electric Power Utilities)
VGB	Verband der GroßkraftwerksBetreiber Association of Electric Power Utilities

## 1.0 Introduction

### 1.1 Background

The safety and reliability of nuclear installations depend in large measure on instrumentation and control (I&C) systems. Together, the I&C systems TELEPERM XP and TELEPERM XS cover the entire range of automation applications in nuclear power plants.

TELEPERM XP is primarily oriented to automation of the non-nuclear part of the power plant process. Such applications involve extensive open and closed-loop control systems and encompass all tasks required for process control.

TELEPERM XS is designed to meet all safety-related I&C requirements in nuclear power plants. Typical applications include reactor protection system (RPS) and engineered safety features actuation system (ESFAS) functions. TELEPERM XS can be configured for 1/2, 2/3, or 2/4 coincidence applications. TELEPERM XS has been qualified to meet the requirements of the highest safety category to perform all of these tasks.

Specific requirements for safety I&C systems are defined in national and international standards. Specific system properties, implemented in specially developed and qualified system software, have been incorporated to meet these requirements. However, TELEPERM XS is based primarily on standard hardware. The qualification of TELEPERM XS has been demonstrated by type-testing all hardware and software components.

The use of one way isolated communication protocols allows the system to be linked to both TELEPERM XP, the control system for operational I&C in nuclear installations, and to third-party systems. Thus the states of the safety I&C equipment can be observed via the process control and information system of the operational I&C. The link through a gateway computer is configured so that faults in the operational (non-1E) I&C cannot affect the operation of the safety-related (1E) I&C.

Implementation of the required nuclear power plant-specific I&C features is facilitated by a specially-designed specification process called SPACE. SPACE is an engineered system that has been designed for the generation of software for all safety-related requirements.

### 1.2 Qualification

The TELEPERM XS I&C system was developed in several phases. This phased approach was complemented by two important activities. First, beginning in 1988, the GRS (German Reactor Safety Association) performed a design review in parallel with the development phase. This review was commissioned by the Bavarian licensing authorities in order to verify conformity with national and international standards. Second, specific quality assurance requirements taken from applicable standards were applied. In its conceptual design review report, completed in 1992, the GRS confirmed the suitability of the system for safety-related applications.

Four qualification steps were implemented:

- Conceptual design review report prepared by the GRS:  
The design review was conducted in parallel with the system development. The 1992 GRS report accepted the conceptual design of TELEPERM XS for safety-related applications.
- Type testing of hardware:  
Hardware type testing was begun in February 1992 by the TÜV Nord technical inspection agency. ISEB was subcontracted by TÜV Rheinland to perform the actual tests. This type testing was satisfactory.
- Type testing of software:  
Software type testing was begun in October 1992 by the GRS, and TÜV Nord was subcontracted to provide these services. The testing was completed satisfactorily and certificates of approval were issued for all safety system software modules.
- Quality verification:  
An independent expert was retained to perform a critical review of the development documentation. The review resulted in satisfactory findings.

The involvement of external experts was continued throughout the development of the TELEPERM XS system.

Although TÜV Nord was responsible for the hardware type testing, TÜV Rheinland performed the actual tests. The objective of hardware type testing is to verify that the hardware can perform its design function under all the specified environmental conditions (such as climatic, EMC, and seismic conditions). The specification of testing requirements relied primarily on U.S. and international standards instead of the usual KTA standards. This reliance was necessary to focus on current digital technology, including electromagnetic compatibility and the verification of the reliability of digital systems.

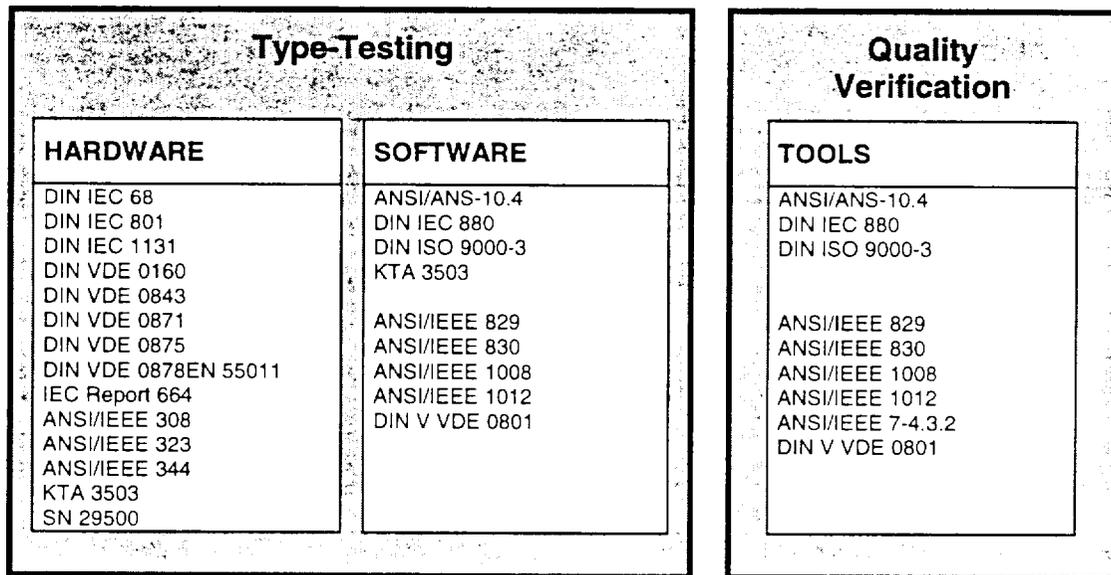
### Type Testing - Quality Verification

A new approach to type testing entails the testing of all reusable software in a manner consistent with KTA Nuclear Safety Standard 3503. This procedure is particularly suited to the TELEPERM XS concept, since the complete application software is compiled by generators from reusable software modules. This means that software modules are used in the same way as hardware modules in hardwired systems.

While the hardware type test focuses mainly on practical testing, the software type test attaches equal importance to theoretical and functional testing.

For example, the theoretical test verifies that the development documentation is consistent with the requirements of KTA 3503, and that the software meets the required design principles (i.e., specifications relating to module sizes, naming conventions, avoidance of insecure constructs, etc.).

The objective of software type testing is to create a framework in which the functionality implemented by software modules can be handled in the same way within the qualification test as functionality implemented by hardware modules.



**Figure 1.1 Standards Applied in the Qualification of TELEPERM XS**

To complete the software type-tests, a plant-independent system test was subsequently performed. This test verified the adequacy of the main system features on representative hardware architectures.

Quality verification of the tools was also included in the software type testing. This quality verification confirmed that the tools are functionally and qualitatively suitable for performing their tasks (e.g., automatic generation of software for safety-critical applications). The main difference between the software type test and the quality verification is that in the quality verification, the tests are performed not by a third party but by the manufacturer. The duties of the independent expert are confined to verification of the development documents and review reports.

Qualification tests were successfully completed in accordance with all acceptance criteria. The criteria specified in the standards referenced in Figure 1.1 have been satisfied unless an exception is noted.

## 2.0 Adequacy of Design Criteria and Guidance

### 2.1 Quality Assurance Program

The qualification of TELEPERM XS was conducted in two parts: an application-independent part and an application-dependent part. This approach reduced the extent of application dependent activities while meeting all applicable requirements. The qualification of TELEPERM XS adheres to the appropriate U.S. requirements as well as applicable German national and international requirements.

The qualification of the application-independent portion of the system consisted of three main aspects. A concept review was performed to evaluate the qualification concept and the important system features. Extensive third party component type-tests were performed for all reusable hardware and software components, including the software engineer's tools. These type-tests were performed to evaluate the components' properties, including their development as well as the verification and validation process. A third party integration and system test was performed to evaluate all safety related system features that could not be assessed at a component level. The successful completion of these tests demonstrating the qualification of all system features are documented in certificates and evaluation reports by third parties.

Through the component type-tests and the integration and system tests, all safety related system properties were certified so that safety systems that meet the design criteria of KTA 3501 or IEEE 279 and IEEE 603 can be designed without examining individual hardware and software features. This means that the evaluation of such a safety system design can be restricted to nearly the same aspects as that performed for conventional solid state protection systems.

#### 2.1.1 Introduction

TELEPERM XS has been qualified as a safety-related system in Germany. Although some countries rely primarily on the evaluation of the manufacturer's quality assurance system, in Germany third party product verification plays an important role. Typically, the German licensing authority orders GRS or a TÜV organization to be directly involved in the licensing process. Because the detailed verification of all safety related product properties is time consuming, the qualification procedure of TELEPERM XS was split in two phases:

- type-tests
- suitability analysis

The type-tests can be considered a kind of generic product qualification. For each type of component belonging to a product family, a third party approves all safety relevant properties on both a theoretical and functional basis. In the scope of type-tests, all component properties are evaluated independently from a specific application. This includes aspects such as functional behavior, robustness against anticipated environmental conditions, failure rates, and testability and maintainability. Based on the results of the type-tests, component properties were certified in evaluation reports. The label "type-tests" is used because these tests were performed on three representative components of each type.

The type-tests also include the evaluation of the component manufacturing process and the quality assurance (QA) system of the manufacturer. The quality assurance requirements are set forth in KTA 1401. The proper implementation of the QA system is verified by audits performed in a three year cycle by utilities. Siemens Quality Management Manual QMH12E implements the requirements of KTA 1401. Additionally, for each development process, tailored engineering procedures were applied. Those used during the TELEPERM XS software development are summarized in "Summary of Engineering Procedures." The requirements for the type-tests are specified in the German Safety Standard KTA 3503.

Whereas the type-tests are product related, the second step of the qualification, the suitability analysis, is application related. The suitability analysis must demonstrate that the safety needs of a specific safety application are met by the design of the safety system, and that the claimed product properties are certified and correspond to the conditions in the plant (e.g., the environmental conditions in the plant are within the design margins of the product, and the system is properly installed). To ensure adequate quality of a system that is to be installed in a plant the components must be identical to those used in the type-tests. The proper implementation of the manufacturer's QA system must be demonstrated and specific manufacturing tests have to be performed separately for each component. The requirements on these manufacturing tests are contained in the German safety Standard KTA 3507. The scope of the tests is fixed at the end of the type-tests. A positive result on the suitability analysis, which includes the design process evaluation and the manufacturing test, is a precondition for obtaining a nuclear license.

The main advantages of this two-phase procedure are:

- The type-tests need not be repeated for each subsequent application.
- The licensing conditions are clarified for application-specific activities, as well as for the manufacture of spare parts.

Although established for hardware components, the basic idea of type-testing, a product-related qualification of reusable components, can also be applied to computer software. If there is a clearly structured software architecture consisting of well-defined modules with proper interfaces, each of these modules can be verified and tested by a third party as a reusable component, independent of a specific application.

It is clear that properties of hardware and software components are quite different, so the term "type-tests" in the context of software qualification requires interpretation. While robustness against anticipated environmental conditions is a major consideration for hardware, software qualification focuses on aspects like defensive programming and a structured and reviewable development process.

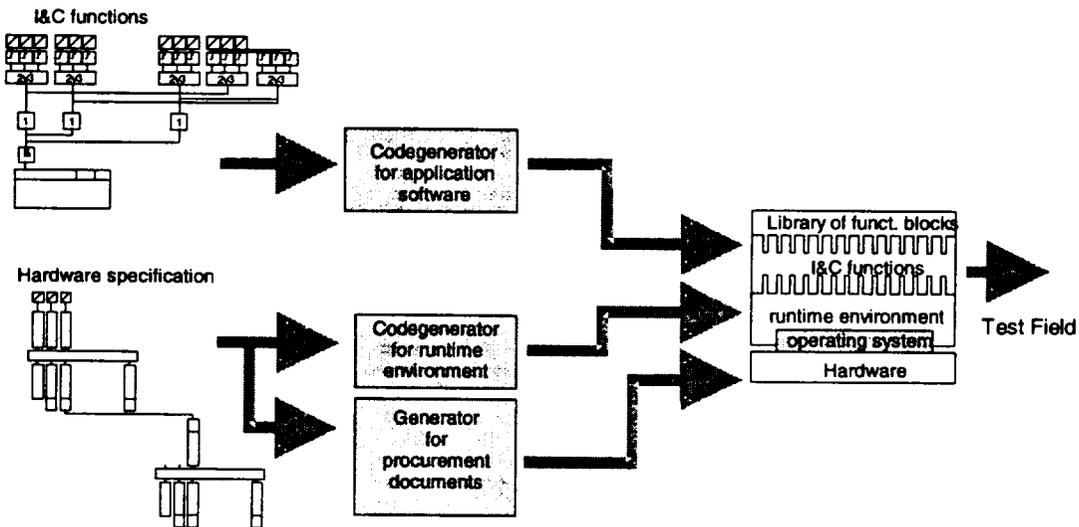
The idea of software type-testing was applied for the first time in the qualification of the TELEPERM XS system.

## 2.1.2 Application Independent Qualification

Because TELEPERM XS was developed for safety-related applications its entire development process was continually assessed by GRS, by order of the Bavarian licensing authority. Defining the safety and qualification concepts was one of the first steps in the development process.

### 2.1.2.1 Philosophy and Strategy

One of the primary goals in the development of TELEPERM XS was to keep as much of the qualification process as possible independent from particular safety applications. Therefore TELEPERM XS, and particularly the newly developed software, was subjected to a generic qualification procedure. In German practice, type-tests are well known as a technique for qualifying hardware modules. The idea of a type-test is to evaluate all safety-related features of a component as well as the manufacturing process independently of specific applications.



**Figure 2.1 Automatic Software Generation by the SPACE Engineering System**

To apply this concept of generic qualification to the software, a robust software architecture needed. The software architecture of TELEPERM XS supports the creation of safety systems for nearly all applications by composing them from pre-existing software modules without modifications. The software modules used to compose the application software have defined interfaces to each other, to application software, and to system software, which allows the safety features to be evaluated mainly at the component level. To reduce the risk of errors during this software assembly phase, the TELEPERM XS, SPACE Engineering System Operation Manual was developed.

Using this engineering manual, the application software for the safety I&C system is completely specified in graphical form. This specification consists of diagrams of interconnected hardware blocks representing the hardware architecture of the safety system, and diagrams of interconnected function blocks representing the safety functions implemented by software. From these specifications the configuration data of the system software and application software is automatically generated by tools of the SPACE (Specification and Coding Environment) engineering system as a composition of interconnected pre-existing and type-tested software components (Figure 2.1). In addition, documents required for hardware manufacturing are also produced by the SPACE system.

Based on this philosophy, the TELEPERM XS qualification concept had been defined (Figure 2.2). It comprises an application independent phase consisting of the three steps:

- concept review,
- component type-tests, and
- integration and system test,

and an application dependent phase based on a standard design process and the associated tool set. The application dependent phase takes credit from all application independent qualification activities.

Within the concept review, the most fundamental principals of TELEPERM XS were evaluated with the aim of reducing the development risk. The concept review was performed at the very beginning of the TELEPERM XS main development and covered the topics

- qualification,
- technical feasibility, and
- specific safety questions

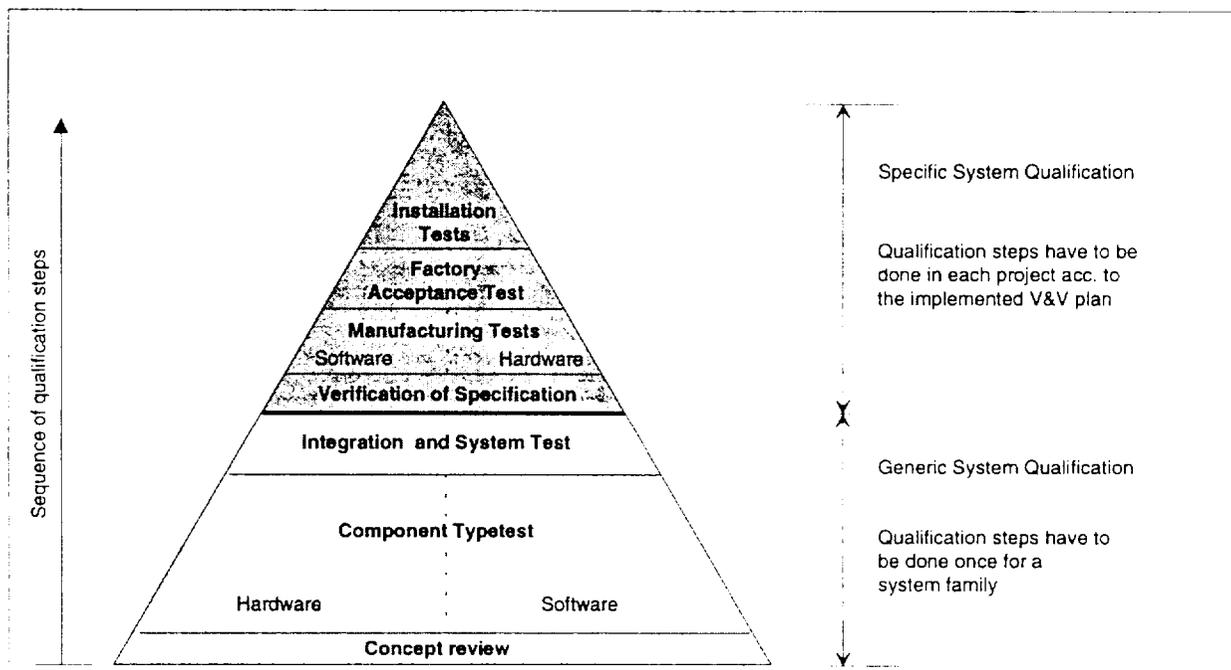
in a generic way. Within this step, the development and the quality assurance plan for the main development were defined, including the engineering procedures that were used later during the software development.

The type-tests comprised all activities necessary to demonstrate and to evaluate the quality of all reusable components of TELEPERM XS. It was split into two branches. One branch focused on the hardware components and the second branch on software components. The type-tests procedure is well defined for hardware components in the German national standard KTA 3503, although to meet international and U.S. requirements some additional tests were defined. For software components the term type-tests had to be newly defined. The test procedure was closely linked to the different phases of the software development, and it included specific safety aspects such as correctness and compliance with the relevant rules and guidelines. According to the development plan, the development of each individual software component was structured into different phases. The results of each phase were documented in an auditable development report, which formed the basis of the input to the subsequent phases.

The software type-tests were mainly based on the evaluation of these development reports and on their compliance with the underlying QA system and the relevant rules and standards.

The integration and system test was newly introduced in the application independent qualification phase as an additional step. Its basis is the fact that it is impossible to demonstrate certain important safety-related system features solely at the module level. Such features are for example:

- deterministic system behavior,
- failure barriers,
- fail safe behavior,
- certain independence properties, and
- response time behavior, etc.



**Figure 2.2 Qualification Concept of TELEPERM XS**

The scope of the integration and system test was derived from a set of safety features that have to be established for all typical safety applications as well as from deficiencies in the type-tests (component features that can only be demonstrated in an integrated system).

The application dependent phase of the TELEPERM XS qualification implements the suitability analysis, which is required according to the German licensing procedure for each individual safety application.

The idea of the suitability analysis is to demonstrate and to evaluate that the safety system really meets the safety needs of the plant. This is mainly done by evaluating the following topics:

- selection of the correct safety functions,
- proper installation in the plant:
  - redundancy and diversity
  - physical separation
  - electrical isolation
  - mounting and erection
- maintainability and testability, and
- coverage of claimed system features by the generic qualification.
- Independent and isolated communication protocols

The standard design process of TELEPERM XS, which is based on the SPACE tool set, greatly reduces the need to address software-related safety questions in the application-dependent qualification phase. Tremendous emphasis was placed on quality design and testing of the SPACE tool set.

#### 2.1.2.2 Concept Review

The concept review was performed as a third party assessment by GRS, as directed by the Bavarian licensing authority (BStMLU). The review started in 1991 and was finished in June 1992. The GRS concept review was completed with acceptable results as documented in the GRS conceptual safety evaluation report.

##### 2.1.2.2.1 Rules and Standards

Given the goal of evaluating whether a software-based safety I&C system could be designed and qualified at acceptable costs in Germany, the concept review used the German national nuclear standards as its basis. The laws, prescriptions and ordinances concerning atomic energy in Germany are prescribed in the "Atomic Energy Act" and subsequent ordinances, with the "Radiological Protection Ordinance" as one of the most important ones. The regulations expressing safety criteria are:

1. BMU-Directives
2. RSK-Guidelines
3. KTA Safety Standards

The BMU ("Bundesministerium für Umwelt") is the German Environmental Department, the RSK ("Reaktor Sicherheitskommission") is the German Commission for Reactor Safety, and the KTA ("Kerntechnischer Ausschuß") is the German Committee for Nuclear Technologies.

Because the BMU-Directives and the RSK-Guidelines contain mainly global criteria, the concept review was primarily based on the design criteria and requirements defined in the following KTA series:

- KTA 1401, "Quality assurance system,"
- KTA 3501, "Reactor protection systems,"
- KTA 3503, "Type-test of electrical equipment,"
- KTA 3506, "System test,"
- KTA 3507, "Factory tests,"

and in the nuclear software standard

- DIN IEC 880, "Software for Computers in the Safety Systems of Nuclear Power Stations."

The KTA Safety Standards specify the required precautions that must be taken when constructing and operating a modern nuclear facility (Sec. 7 par. 2 No. 3 Atomic Energy Act). Adherence to these safety-related requirements ensures that the protective goals of the Atomic Energy Act and the Radiological Protection Ordinance are fulfilled.

Because of its central role in the licensing of I&C systems and because the type-testing standard 3503 is derived from it, a short overview of KTA-Standard 3501 follows.

KTA-Standard 3501 applies to the reactor protection system and to the monitoring equipment of the safety system of nuclear power plants. This safety standard specifies requirements for the

- design,
- equipment quality,
- installation, and
- testing

of the reactor safety system and its components.

With respect to a new design of a reactor protection system, it comprises

- design principles for the reactor protection system, and
- the requirements regarding qualification and testing.

The basic requirements concerning the design principles for the reactor protection system can be expressed as follows:

- It shall be demonstrated that the reactor protection system together with the active and passive safety system equipment is designed, manufactured and operated such that

intolerable effects are prevented from occurring. Here, failure-inducing events within the reactor protection system (e.g., shorts, fires, mechanical failures) and in the nuclear power plant (e.g., fires, flooding, debris of failed components, mechanical jet effects of media such as steam, water, gas and oil) shall be assumed to occur simultaneously with, but independently of the incident.

The basic requirements concerning the qualification and testing for the reactor protection system can be summarized as follows:

Proven equipment with good service records should be applied in the reactor protection system. The properties important to safety should be demonstrated in the framework of type-tests according to KTA 3503. The scope of type-tests and the test procedures should be approved by an assessor in charge of the supervision of a nuclear power plant. The suitability of the system should be demonstrated for each safety application in the framework of the suitability analysis. Within the suitability analysis it must be demonstrated that the type-tested equipment with its type-tested properties fulfills the safety needs of the specific application. The correct implementation of the safety functions must be demonstrated by:

- factory tests (in detail KTA 3507), and by
- system test (in detail KTA 3506).

#### Correspondence to U.S. Requirements

According to the Standard Review Plan (NUREG 0800, Rev. 4, June 1997) and the Branch Technical Positions, the following standards must be applied in the development of safety system for nuclear power plants:

- IEEE 279,
- IEEE 603, and
- IEEE 7-4.3.2.

The safety philosophy of the IEEE 279 and the IEEE 603 standards was the basis of the first nuclear power plants in Germany and the design criteria are very similar to those of KTA. KTA 3501 covers the requirements of IEEE 279, IEEE 603, IEEE 338, IEEE 379, and IEEE 384, while KTA 3503 covers the requirements of IEEE 323 and IEEE 344. As in KTA 3501, the requirements in IEEE 279, IEEE 379, IEEE 384 and IEEE 603 address the application-specific design of a safety system. They influence the development and the type-tests of hardware and software components in that they set forth the required features of these components. Therefore these standards play an important role in the concept review and in the application-specific qualification, but they have a low impact on the generic qualification.

IEEE 7-4.3.2 addresses also specific requirements concerning the software development. Most of them are given by referencing the standards ASME NQA, IEEE 730, IEEE 828, IEEE 1012, and IEC 880. The requirements of ASME NQA are covered by KTA 1401 and the requirements of IEEE 730 are covered by those of ISO 9000-3.

#### 2.1.2.2.2 Qualification Process

The concept review was done as a group of reports. Topical reports were prepared and submitted to GRS for evaluation. The topical reports were placed in the context of a safety analysis report of a KONVOI plant (1300 MW pressurized water reactor). Thus assumptions about aspects not covered by the topical reports themselves but necessary for the safety evaluation (e.g., safety functions, response time requirements, number of trains, amount of safety systems etc.) were taken from the most modern German pressurized water reactor. The reports covered the following items:

- the design of a computer-based reactor protection system,
- the design of a computer-based limitation system, and
- the design of a computer-based closed-loop control system including:
  - spatial separation principles
  - electrical isolation principles
  - principles for the distribution of safety tasks
  - response time behavior
  - verification and validation activities during design and installation
  - principles for coping with design or manufacturing defects
  - principles and assumptions of the probabilistic analyses
  - safety features of the computer based system itself, including:
    - the software architecture
    - principles and modes of operation
    - response time behavior
    - principles of the system software development
    - principles of the application software development
    - test and self-monitoring features
    - fail-safe features
- qualification of the computer based system.

The most important evaluation criteria were:

- compliance of the design with the underlying safety criteria,
- technical feasibility of the design, and
- licensability of the design.

#### 2.1.2.2.3 Submitted Documents and Results

Reports submitted for concept review:

1. Konzeptbeschreibung Teil 1 Sicherheitsbericht
2. Zuverlässigkeitsanalyse der digitalen SILT (abl. Ort. A.03.02.20)

3. Platzbedarf und räumliche Aufteilung der digitalen SILT für KONVOI (abl. Ort B.02.03.019)
4. Platzbedarf und räumliche Anordnung der Meßtechnik-Peripherie (abl. Ort B.03.01.028)
5. Bewertung der Maßnahmen zur Vermeidung und Beherrschung systematischer Ausfälle (abl. Ort A.03.02.022)
6. Geräte Qualifizierungskonzept für die Leittechnik des Sicherheitssystems (abl. Ort B.08.02.017)
7. Prüfung und Wartung der Sicherheits-Leittechnik (abl. Ort B.05.05.001)

### Evaluation Report

- Gutachten zum Konzept der digitalen Sicherheits-Leittechnik von Siemens/KWU (GRS-A-1921)

### Results of the Evaluation (Summary)

The result of the concept review had been summarized by GRS in (GRS-A-1921) as follows:

"In the opinion of the assessor the concept submitted constitutes a new I&C system for critical and highly critical safety applications in nuclear power plants. It is based on future-orientated technology which corresponds to the state of the art. The concept submitted goes into such depth, that its consistent implementation in line with the pertinent standards will be possible and that no problems are to be expected in the course of this."

#### 2.1.3 Type Tests

Both software and hardware type tests have been performed. Please see Sections 2.2.1, "Hardware Type Test," and 3.2.1, "Software Type Test."

#### 2.1.4 Summary of the Application Independent Qualification

The qualification of TELEPERM XS was divided into an application-independent part and an application-dependent part, in order to reduce the proportion of application dependent activities. The underlying standards meet the German national and the international requirements, with specific attention to meeting the U.S. requirements.

The qualification of the application-independent portion of the system had three main aspects. A concept review was performed to evaluate the qualification concept and the important system features. Extensive third party component type-tests were performed for all reusable hardware and software components, including the software engineer's tools. These type-tests have been performed in order to evaluate the components' properties, their development as well as the verification and validation process. A third party integration and system test was performed to evaluate all safety related system features that could not be assessed on a component level. All tests were passed in accordance with required acceptance criteria and the qualified system features were confirmed in certificates and associated evaluation reports by third parties.

Through the component type-tests and the integration and system tests, all safety related system properties were certified so that safety systems that meet the design criteria of KTA 3501 or IEEE 279 and IEEE 603 can be designed without going deep into hardware and software features. This means that the evaluation of such a safety system design can be restricted to nearly the same aspects as for conventional solid state protection systems.

The application-dependent qualification benefits from a standardized system design process that is supported by a qualified tool set. This design process identifies the verification and validation activities and proposes documents to submit for evaluation.

### 2.1.5 Record of Documents Submitted for Software Type-Test

Please see Section 9.0, "Software Type Test," for a summary of the documents.

## 2.2 **Equipment Qualification**

### 2.2.1 Hardware Type-Test

The hardware type-tests were performed as a third party assessment by TÜV-Nord. To broaden the base of expert knowledge for this qualification activity and to keep the qualification time in acceptable limits, TÜV-Nord subcontracted TÜV-Rheinland to perform the practical tests. The test specifications as well as all theoretical tests were performed or evaluated (analyses that were performed by the manufacturer) by TÜV Nord. On the basis of these test specifications, TÜV Rheinland performed the practical tests. The hardware type-tests began in 1993 and ended for the first set of hardware modules in 1996. As a result of the first safety applications, some additional hardware components were included in the TELEPERM XS system. The type-tests of these new components were performed by the same parties. The results of the type-tests are documented by certificates and associated evaluation reports. Each qualified component has its own certificate and its own evaluation report, including the associated configuration and authentication.

#### 2.2.1.1 Rules and Standards

The hardware type-testing was executed in accordance with KTA 3503 "Type-test of Electrical Modules for the Reactor Protection System." This standard gives requirements for the overall structure of the test activities including:

- separation between the theoretical and practical tests,
- parties to be involved in type-tests,
- roles of these parties, and
- test documentation of type-tests.

The standard also gives requirements for the scope of the theoretical and practical tests, such as:

- worst case analyses,

- determination of reliability data,
- functional test, and
- environmental tests.

Because the In addition to KTA -Standard 3503, international standards was reviewed and included in the type-tests of the hardware. The main groups of these standards are:

- KTA 3503 "Type Testing of Electrical Modules for the Reactor Protection System,"
- IEEE 308 and 323 (concerning Standard Class 1 E Equipment for Nuclear Power Stations),
- DIN VDE-Standards (various national German standards for electrical equipment), and
- DIN IEC-Standards (primarily concerning environmental influences on electrical equipment).

Within these groups, the requirements of the following standards were satisfied:

- KTA, IEEE:
  - KTA 3503.  
Typprüfung von elektrischen Baugruppen des Reaktorschutzsystems
  - IEEE 308.  
Standard Criteria for Class 1 E Power Systems for Nuclear Power Generating Stations.
  - IEEE 323.  
Standard for Qualifying Class 1 E Equipment for Nuclear Power Generating Stations.
- DIN VDE:
  - DIN VDE 0160.  
Ausrüstung von Starkstromanlagen mit elektrischen Betriebsmitteln.
  - DIN VDE 0160.  
Ausrüstung von Starkstromanlagen mit elektrischen Betriebsmitteln, Änderung 1.
  - DIN VDE 0110 P1.  
Isolationskoordination für elektrische Betriebsmittel in Nieder-spannungsanlagen.
  - DIN VDE 0110 P2.  
Isolationskoordination für elektrische Betriebsmittel in Nieder-spannungsanlagen.
- DIN IEC
  - DIN IEC 1131-2.  
Draft Programmable Controllers Part 2: Equipment Characteristics.
  - DIN IEC 68 Teil 1.  
Grundlegende Umweltprüfverfahren.
  - DIN IEC 68 Teil 2.  
Grundlegende Umweltprüfverfahren Feuchte Wärme, zyklisch.
  - DIN IEC 68 Teil 2-1.  
Grundlegende Umweltprüfverfahren Kälte.
  - DIN IEC 68 Teil 2-2.  
Grundlegende Umweltprüfverfahren Trockene Wärme.

- DIN IEC 68 Teil 2-3.  
Grundlegende Umweltprüfverfahren Feuchte Wärme, konstant.
- DIN IEC 68 Teil 2-14.  
Grundlegende Umweltprüfverfahren Temperaturwechsel.
- DIN IEC 68 Teil 2-6.  
Grundlegende Umweltprüfverfahren Schwingen, sinusförmig.
- DIN IEC 68 Teil 2-27.  
Grundlegende Umweltprüfverfahren Schocken.
- DIN IEC 1000-4-1.  
Elektromagnetische Verträglichkeit von Meß-, Steuer-, und Regeleinrichtungen in der industriellen Prozeßtechnik Allgemeine Einführung.
- DIN IEC 1000-4-2.  
Elektromagnetische Verträglichkeit von Meß-, Steuer-, und Regeleinrichtungen in der industriellen Prozeßtechnik Störfestigkeit gegen die Entladung statischer Elektrizität.
- DIN IEC 1000-4-3  
Elektromagnetische Verträglichkeit von Meß-, Steuer-, und Regeleinrichtungen in der industriellen Prozeßtechnik Störfestigkeit gegen elektromagnetische Felder, Anforderungen und Meßverfahren.
- DIN IEC 1000-4-4.  
Elektromagnetische Verträglichkeit von Meß-, Steuer-, und Regeleinrichtungen in der industriellen Prozeßtechnik Störfestigkeit gegen schnelle transiente Störgrößen (Burst).

#### Correspondence to U.S. Requirements

Among the standards referenced in the Standard review plan and the Branch Technical Position, the IEEE 603 has some importance for the hardware type-tests because it contains references to IEEE 308 and IEEE 323, both of which are part of the qualification basis.

#### 2.2.1.2 Qualification Process

The essential elements of the type-tests according to KTA 3503 are:

- the basic task,
- the parties involved,
- the test procedure, and
- the distribution of activities to be performed by the involved parties.

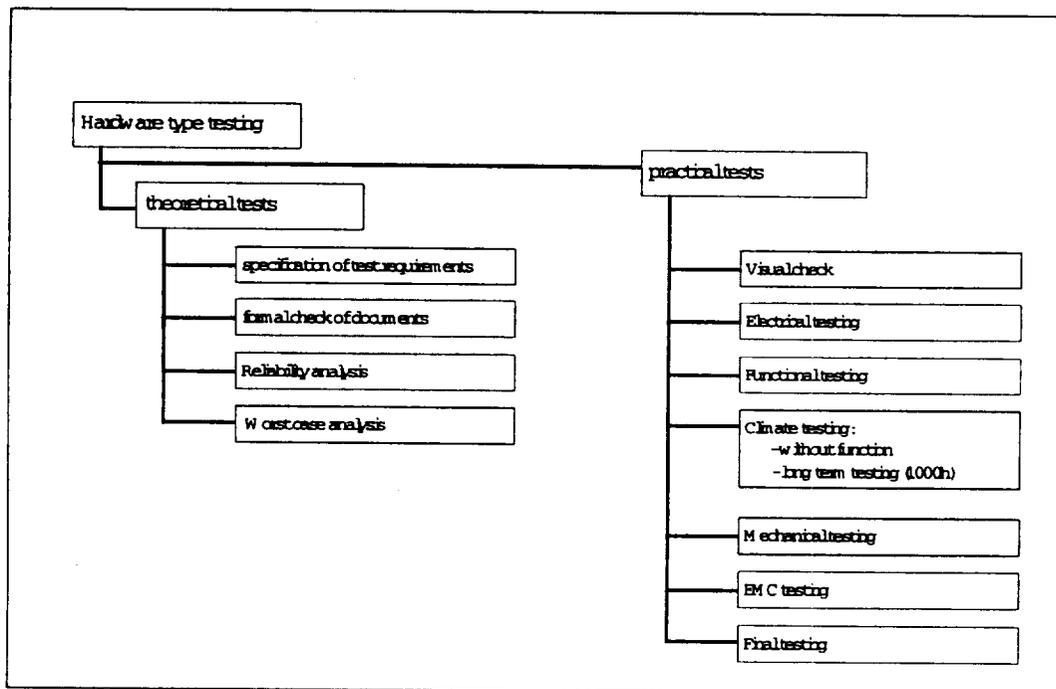
The basic task is the demonstration of the proper design of a piece of equipment, and of its correct functionality at the interfaces under worst case conditions.

The parties involved are the manufacturer and third party experts. The third party experts are authorized by the licensing authority.

The test procedure itself is divided into two major parts, the theoretical part and the practical part. The theoretical part deals with all documents describing and analyzing the components subjected to the type-tests, and with the test specification for the practical part. The underlying

standards demand a particular set of documents, including data sheet descriptions of the input/output relations and all other data that characterize the components. When the test specification is finished, practical tests may begin. At the end of the theoretical and practical part, certificates and evaluation reports are prepared by the authorized third party experts.

For TELEPERM XS, the distribution of activities within the type-tests was as follows. Type-tests were mainly performed by the third party experts (TÜV-Nord and TÜV-Rheinland), with support from the manufacturer. This support included the preparation of all the required documents, the performance of specific analyses in the theoretical part, and the execution of certain specific tests.



**Figure 2.3 Scope of the Hardware Type Test**

According to KTA 3503, the responsibilities of the manufacturer are to define:

- the scope of components to be type-tested,
- the features of the components important to safety that will be used in the licensing procedure, and
- the environmental conditions under which the system must operate.

Next the manufacturer must prepare the documents for the theoretical part of the type-tests and the test plan for the practical part of type-tests. The authorized third party experts must check and evaluate the documents for the theoretical part of the type-tests, and must check and approve the test plan proposed by the manufacturer.

The accepted test plan is the basis for the detailed test specification, which dictates how the practical tests will then be performed. The results of the practical tests are evaluated by the authorized third party experts. On basis of this evaluation, the third party experts prepare the certificates and the evaluation reports.

For electronic components of the reactor protection system, the type-tests must be carried out in accordance with KTA 3503.

#### 2.2.1.2.1 Theoretical Part

In the theoretical part of the type-tests for TELEPERM XS,

- the test plan and the test specification were elaborated,
- the consistency of the submitted documentation was checked,
- critical load analyses were performed, and
- reliability data was calculated.

First TÜV-Nord identified the documents required by the underlying standards. Three types of documents were submitted to TÜV-Nord:

- Requirements related documents, including:
  - List of documents,
  - Requirement specification,
  - Environmental conditions, and
- Development related documents, including:
  - Description of the function,
  - Description of interfaces,
  - Description of the functional scheme, and
- Manufacturing related documents, including:
  - Description of the construction,
  - Description of the layout,
  - Manufacturing films,
  - Wiring diagrams,
  - List of components,
  - User documents, and
  - Data sheet.
  - Software specification.

For each component, TÜV-Nord verified the consistency of these documents and the safety-related design features.

Using the test specification as a basis, the manufacturer created a test plan, which described the test method, the test concepts, and the standards to be applied.

The test plan specified that the following practical tests be performed in the following test sequence:

1. Visual inspections:

- To see whether the equipment matches the valid manufacturing documentation in the documentation list.
- To prove
  - o Cleanness (stains, residual solder, discoloration, crystallized solder, etc.)
  - o Correct soldering (wetting, soldering tags, etc.)
  - o Positioning of the components (spacing between conductive parts etc.)
  - o Electrical safety in accordance with VDE 0160.
- To prove clearance and creepage distances.
- Software version.

2. Function tests:

To demonstrate that the hardware component functions in accordance with its description in the data sheet. The function test is performed under nominal and limit load conditions.

3. Test of electrical characteristics:

- Maximum current consumption,
- Maximum power consumption,
- Module insertions and removals,
- Connection and disconnection short power interruption,
- Connection and disconnection Shutdown and startup,
- Non-periodic drop in the supply voltage,
- Non-periodic rise in the supply voltage,
- Injection of AC voltage into the supply voltage,
- Isolation,
- Effectiveness of the protective conductor,
- Self-heating, and
- Short-circuit and overload withstand capability.

4. Climatic tests:

To demonstrate that the operability of the modules is not degraded by the permissible climatic stress defined by the system description. Tests were performed with the rack doors closed and temperature rise was monitored at numerous locations within the rack. The following tests were performed:

- Constant cold after a rapid change in temperature (without operation),
- Dry heat after rapid temperature change (without operation),
- Dry heat, constant (with operation),
- Humid heat, constant (with operation),
- Humid heat, cyclic (with operation), and
- Cyclic dry heat (long-term 1000 hour test with operation).

#### 5. Tests to demonstrate the EMC:

To demonstrate that the operability of the TELEPERM XS system and its components is not degraded by the electromagnetic compatibility conditions. The components are stressed as part of a representative system configuration or a typical application-oriented system configuration. Tests were performed with the rack doors both open and closed. Shielded standard I&C cables were used. The following issues have been checked:

- Interference emission,
- Noise immunity against conducted interference for systems,
- Noise immunity against bursts,
- Interference immunity against surges,
- Interference immunity against radio frequency interference, and
- Interference immunity against electrostatic discharge.

#### 6. Mechanical stress:

To demonstrate that the operability of the component is not degraded by the mechanical stresses permitted according to the system data:

- Simulation of operational vibrations,
- Resistance to vibration in the frequency range 2 to 35 Hz,
- Resistance to vibration in the frequency range 5 to 100 Hz, and
- Shock resistance.

#### 7. Function test:

Repetition of the function test specified in point 2 to verify that the modules have passed the above tests without any degradation of function.

The test plan outlined above was verified and accepted by TÜV-Nord.

Because the hardware components of TELEPERM XS were not specifically developed for safety applications but are commercial grade components, they were qualified for safety-related applications. Hardware features can only be used if the system software and the engineering tools support them. All supported features are qualified and described in the user manuals. It is not possible to perform unintended functions. The new user manuals were prepared by the manufacturer and contain descriptions of all features required for safety applications. These user manuals were submitted to the TÜV-Nord, together with all other documents listed above, both for evaluation and to elaborate the detailed test specification. The detailed test specification was then created by TÜV-Nord. In order to support the function tests, the development and manufacturing documents were analyzed to identify the critical signal paths and to define methods to test and monitor these paths under all conditions defined by the test plan. In addition, the consistency of the documents was checked and the component design was evaluated.

The critical load analyses and the calculation of reliability data (failure rates for each individual components and failure effects on system level) were performed by the manufacturer and evaluated by TÜV-Nord.

#### 2.2.1.2.2 Practical Part

As required by KTA 3503, all practical tests were performed using three components of each type. The idea behind this was to make it easier to distinguish between random failures that may occur during the tests and systematic failures resulting from design or manufacturing defects. All components were manufactured under the same qualified manufacturing process that is used later to produce the components for safety systems. In addition, all components were worst-case factory tested before they were used for the type-tests.

The following principles were applied during the practical tests:

- All tests were based on a detailed test specification describing the test configuration, all individual test steps, tools used for monitoring the response of components, etc.
- Before a test was started, the test configuration was evaluated by TÜV-Nord in the laboratories. This was done to avoid misunderstandings resulting from the test specification and to keep the test procedure transparent for all parties involved. The evaluation checked whether the test configuration was in line with the specification and that it was designed such that the test could be repeated under exactly the same conditions.
- Before starting a test, all test programs (software programs) were stored on magnetic tapes or disks for use if tests had to be repeated under the same conditions.
- If a failure occurred during a test, the failure cause had to be evaluated and described by the manufacturer. If the failure could be identified without doubt as a random failure, then the failed component could be repaired and the last test repeated. If the failure was identified as a consequence of a design or a manufacturing defect, then the defect had to be corrected and additional control mechanisms had to be added to the QA program of the manufacturer. Under the new QA program, new components had to be manufactured and the whole type-test sequence had to be performed from the very beginning for the failed components.
- The function tests and the electric tests were performed on a individual component basis. That means the features of each component were tested separately. This is in contrast to the climatic and mechanical stress tests, which were done with components in operation. The EMC tests were also performed using an integrated system, which contained three components of each type being tested and a set of additional components as test-bay. The integrated systems used for the type-tests contained as far as possible the original system software. The application functions were specified with SPACE and generated with the code generators like in real TXS projects. However, the application functions were not designed to control accidents but to monitor hardware features. For example, one computer generates analog output signals, with cyclic variation from 4 to 20 mA to monitor the D/A converters of the output modules. The same signals are read back by a second computer, to monitor the A/D converters of the input modules. Then, the values are transmitted to the first computer via a serial link and compared with the original signal. A lot of functions of this type are specified to monitor all hardware components continuously.

- After those climatic and mechanical stress tests that were done with components not in operation, a selected and agreed set of function tests was performed to verify whether the component works correctly after that test.
- During each test detailed test logs were prepared by the party in charge to perform the test. These test logs were later integrated in the test reports. After the tests these test reports were checked by TÜV-Nord to evaluate whether the test was successful or not.

The practical tests were performed by TÜV Rheinland using its own laboratories or, for certain specific tests, in the laboratories of the manufacturer. If tests were performed by the manufacturer for practical reasons, the tests were supervised by TÜV Rheinland.

### 2.2.1.3 Scope of Components

In order to be able to use all components of TELEPERM XS in safety applications, all components supported by TELEPERM XS were type-tested according to the same criteria. The TELEPERM XS consists of the following kinds of hardware components:

- function computer for signal processing,
- components for communication,
- components for input and output of digital or analog signals, and
- electrical accessories such as electrical connections, cables, subracks, power supply and cabinets for housing the above modules.

The function computers used for signal processing are taken from the powerful multi-microprocessor system AS 990 and are equipped with i80486 INTEL processors. The programs are stored in read-only memories (ROMs) that are either flash EPROMs or EEPROMs.

The communication modules are from the multi-microprocessor system AS990 as well as from the SINEC product family. The SINEC-Communication modules consist of SINEC L2 and SINEC H1 components. The SINEC L2 bus is used with TELEPERM XS for system internal data transmission and as a field bus for the connection to SINEC L2-compatible data acquisition and control units in the power plant. The SINEC H1 bus conforms to the Ethernet standard. The SINEC H1 bus with its very high data transmission rate is suitable for connections to external equipment such as the process information system, Local Area Networks (LAN) or other automation equipment.

Input and Output Modules for analog and digital signals are taken from the SIMATIC S5 equipment family, which are popular programmable controllers of SIEMENS.

The scope of the hardware components was defined at the beginning of the type-tests but was changed during the test procedure for two different reasons. First, certain components were found to have weak points that would make it difficult for them to pass the type-tests. Therefore these components were replaced by other components or were taken completely out of the TELEPERM XS system. Second, specific safety applications required additional components,

so these were added to the set of type-tested components. To date the following components have been type-tested:

- Function computer for signal processing:
  - SVE1 Function computer
  
- Components for communication:
  - SCP1 H1 communication processor
  - SHS1 Active star Coupler
  - SL21 L2 communication module
  - SBU1 Local bus interface module
  - SKO1 Local communication module
  - SHO1 Fiber optic transceiver
  - SHO2 Fiber optic interface card
  - SHT2 Twin transceiver card
  - SLLM L2 link module
  
- Components for input and output of digital or analog signals:
  - S430 Binary input module
  - S431 Binary input module
  - S451 Binary output module
  - S458 Binary output module (relays)
  - S460 Slow analog input module
  - S466 Fast analog input module
  - S470 Analog output module
  - S706 Pulse counter module
  
- Subracks and electrical accessories:
  - SBG1 Subrack with one backplane
  - SBG2 Subrack with two independent backplanes

Other electrical accessories such as cables and power supplies have already been type-tested as part of other qualified hardware families.

#### 2.2.1.4 Submitted Documents and Results

The hardware type-test was performed according to KTA 3503, which defines the test procedure in detail. Thus the test included:

- document reviews,
- analyses of the design,
- inspections of the hardware, and
- extensive tests under extreme environmental conditions.

### Reports Submitted for Document Reviews

For each of the type-tested hardware components, the documents listed in Section 2.2.1.2.1, Theoretical Part, were submitted. In addition, the test plan and the mounting and shielding principles were submitted. The mounting and shielding principles are described in the document TELEPERM XS Hardware Installation Guide, KWU NLL-1025-76-V1.0/02.97.

### Reports Submitted for Analyses of the Design

Failure rate analyses and critical load analyses were performed and were submitted for all components.

### Documentation of Test

Each test was documented by a test report that was elaborated by the party in charge of performing the test. All test reports were submitted to TÜV-Nord for evaluation.

### Evaluation Reports

For each of the type-tested hardware components listed above, a certificate and a summarizing evaluation report were elaborated by TÜV-Nord. The certificate identifies the component and contains a short evaluation of whether the component has an acceptable quality for safety applications in nuclear power plants. It also contains a summary of the most important evaluation criteria. The evaluation report contains a more detailed description of the evaluation process and of the findings.

All components listed above were accepted as having adequate quality for safety applications in nuclear power plants.

The certificates and the evaluation reports confirm that:

- the quality assurance system of the manufacturer meets KTA 1401,
- the associated components fulfill their (restricted) functions under all conditions specified in the data sheets,
- the components have a sufficiently robust and conservative design (critical load analyses),
- the component design in combination with the mounting and shielding principles provides sufficient protection from electromagnetic interference and seismic disturbances,
- the development documentation is complete and consistent such that the component can be maintained in the long term, and
- the calculated failure rates are acceptable with respect to probabilistic reliability assessments, that will be performed later on during the application dependent phase.

## 2.2.2 Hardware Test Requirement Specification

### 2.2.2.1 Introduction

During the course of type qualification, the practical tests together with the theoretical tests aim to prove that the components of the digital I&C system TELEPERM XS are consistent with the TELEPERM XS System data and with the instruction manuals for the respective component.

This section describes the main procedure and the test steps that are to be conducted for the practical test of all TELEPERM XS components used for digital safety I&C. For each of the test steps,

- the test basis,
- the test parameters,
- guidance for performing the respective test steps, and
- the test criteria (where required)

are described.

Three specimens are tested for each component type.

Unless otherwise stated, the modules are mounted in subracks and connected to each other according to the respective test requirements and standards.

The practical tests are documented in accordance with the requirements of KTA 3503, Chapter 6.2.

### 2.2.2.2 Documentation Required for Type Testing

#### 2.2.2.2.1 General Information

The documentation is compiled by the manufacturer, as described in Sections 2.2.2.2.2, 2.2.2.2.3, and 2.2.2.2.4. This documentation must be appended to the test documentation.

#### 2.2.2.2.2 Component Documentation

All component documentation must state the manufacturer, type, and version of the modules. This includes documentation such as function descriptions, data sheets, document lists, circuit diagrams, part lists, component mounting diagrams, and PCB drawings. This component documentation forms part of the test documentation.

#### 2.2.2.2.3 Reliability Data

- The failure rate of the modules under their intended usage must be specified in the form of cumulative failure rates. The basis for calculating the cumulative failure rates is the component failure rates given in SN29500 Parts 1 to 14.

- If failure rates for function units or components can be calculated with a sufficient level of confidence based on operating experience, the evaluation based on operating experience takes precedence over theoretical failure rates.
- The method by which the reliability data is calculated must be specified.

#### 2.2.2.2.4 Critical Load Analysis

- The critical load analysis must verify that the components and their electrical connections are not loaded beyond their permissible limit data, either statically or dynamically.
- For selected component combinations, the effects of component tolerances on the functioning of the module must be examined.
- The verification can be done either by calculation or experimentally, e.g., using thermography.

#### 2.2.2.2.5 Test Instructions for the Practical Test

The test instructions must describe the type of test and the procedure (sequence and scope of the test steps) for the test (see Section 2.2.2.3).

#### 2.2.2.3 Practical Type Qualification

- The practical tests of the type qualification are performed in a manner consistent with KTA 3503, IEC780 (is equivalent to IEEE 323-1974), and IEC 1131-2.
- For the practical tests, the technical data and requirements specified in the module-specific instructions provide the basis for demonstrating compatibility with electrical, climatic, and mechanical ambient conditions.
- The practical tests for demonstrating the compatibility of TELEPERM XS components with their ambient conditions must be performed in the following sequence:
  - visual inspection in accordance with functional test in accordance with Section 2.2.2.3.1
  - and test of electrical characteristics in accordance with Section 2.2.2.3.3
  - climatic tests in accordance with Section 2.2.2.3.4
  - mechanical stress in accordance with Section 2.2.2.3.5
  - test to demonstrate EMC in accordance with Section 2.2.2.3.6
  - functional test according to Section 2.2.2.3.7
- The test steps are performed with three specimens of each component.
- The components are to be mounted in the position consistent with their intended usage. The connections (plug-in connectors, cables) are to be installed according to plant requirements. (IEEE 323 Section 6.3.1)
- Components whose practical test results can be derived from similar components, that are already type-tested, do not need to be tested again. Instead, a comparative analysis must be drawn up. This must contain the common design characteristics (such as input circuitry,

output circuitry, power supply and components used) and art work, as well as the differences between the components considered.

- During the practical tests, intermediate functional tests must be performed at certain hold points, in accordance with this test requirement specification. Selected individual test steps must be performed with a single value for power supply, output load, and ambient temperature, and in a single mode of the component or system.
- The function monitoring must be performed as specified in Sections 2.2.2.3.3, 2.2.2.3.4, 2.2.2.3.5, 2.2.2.3.6 during certain practical tests. Function monitoring must be performed (e.g., using test software) in such a way that even short-term function failures of the components are detected.

#### 2.2.2.3.1 General Tests

##### Test Step 1: Visual Inspection

##### Test Step 1.1: Identification of the Components

- Test basis:
  - KTA 3503, Sec. 5.1
  - IEC 1131-2, Sec. 4.23, 6.2, 6.3.13
  - DIN VDE 0160, Sec. 5.1, 7.8
  - Instruction manual of the respective component
  - Manufacturing documentation according to documentation list
- Test parameters:
  - Technical documentation of the components
- Performing the test:
  - Visual inspection of the components to see whether they match the valid technical documentation according to documentation list.

##### Test Step 1.2: General Visual Inspection

- Test basis:
  - DIN VDE 0160, Sec. 5
- Test parameters:
  - None
- Performing the test:

This test comprises examination of the external condition of the component. As a minimum, the following points must be examined:

- Cleanness  
(stains, residual solder, discoloration, crystallized solder, etc.)
- Correct soldering  
(wetting, soldering tags, etc.)

- Positioning of the components  
(spacing between conductive parts, e. g. metal housing of transistors, connection wired, etc.)
- Electrical safety in accordance with VDE 0160

### Test Step 1.3: Testing Clearance and Creepage Distances

- Test basis:
  - IEC 1131-2, Sec. 4.3, 6.3.2.2, 6.3.5.5.3
  - DIN VDE 0160, Sec. 5.7, 7.7
  - DIN VDE 110, Part 2
  - TELEPERM XS System data
  - Manufacturing documentation according to documentation list
- Test parameters:
  - Supply voltages
  - Operating conditions
  - The clearance and creepage distances as rated for the following design:
    - Rated voltage: Is defined by the voltage of the component-internal circuits (VDE 0110, VDE 0160)
    - Overvoltage category: 2
    - Pollution severity: 1
    - Insulation group (for creepage distance): printed circuit
- Performing the test:

The test is performed in one of three ways:

  - by visual inspection of the printed-circuit boards
  - by testing the films
  - by testing the layout system

### 2.2.2.3.2 Functional Tests Under Nominal and Limit Load Conditions

#### Test Step 2: Functional Test

- Test basis:
  - KTA 3503, Sec. 5.2.2, 5.2.3, 5.7.3
  - IEC 1131-2, Sec. 6.3.7.1.1
  - TELEPERM XS System data
  - Instruction manual of the respective component
- Test parameters and performing the test:
  - To demonstrate that the hardware components function in accordance with its description in the data sheet, a functional test must be performed under nominal and limit load conditions.

### 2.2.2.3.3 Electrical Tests

#### Test Step 3: Testing the Electrical Characteristics

- Test basis:
  - KTA 3503, Sec. 5.7
  - IEC 1131-2, Sec. 6.3
  - Instruction manual of the respective component
- Test parameters:
  - Nominal voltages
  - Room temperature
- Performing the test:
  - Unless otherwise stated, the electrical characteristics are tested under nominal conditions following the individual test steps described below.

#### Test Step 3.1: Testing the Maximum Current Consumption

- Test basis:
  - KTA 3503, Sec. 5.7.1
  - IEC 1131-2, Sec. 6.3.2.2
  - Instruction manual of the respective component
- Test parameters:
  - Input signals
- Performing the test:
  - With maximum input and output signals applied to all inputs and outputs simultaneously, the maximum current consumption of the component is to be measured at nominal voltage and room temperature.

#### Test Step 3.2: Testing the Maximum Power Consumption

- Test basis:
  - KTA 3503, Sec. 5.7.2
  - IEC 1131-2, Sec. 6.3.13
  - Instruction manual of the respective component
- Test parameters:
  - None
- Performing the test:
  - The power loss of the component must be calculated from the nominal voltages and the maximum current consumption previously determined.

### Test Step 3.3: Component Insertion and Removal

- Test basis:
  - KTA 3503, Sec. 5.7.4
  - IEC 1131-2, Sec. 4.10, 6.3.2.2, 6.3.5.7
  - Instruction manual of the respective component
- Test parameters:
  - Number of withdrawing/plugging-in cycles: 50
- Performing the test:
  - The component is inserted in the respective subrack and removed again. The front connectors (if any) are also plugged in and then unplugged. When the front connector is unplugged, the designed response of the component must be checked as well. An intermediate functional test is performed after every fifth cycle (removal and insertion).

### Test Step 3.4: Connection and Disconnection - Short Power Interruption

(only for load power supply, SBG1, SBG2 and SHS1)

- Test basis:
  - IEC 1131-2, Sec. 3.2.1.2, 6.3.2.2, 6.3.7.2.1
  - TELEPERM XS System data
- Test parameters:
  - Interruption of the power supply
    - Duration 1: 1 ms
    - Duration 2: 10 ms
    - Recovery time:  $1\text{ s} < t < 10\text{ s}$
    - Number of interruptions: 20
- Performing the test:
  - If the power supply is interrupted for 1 ms (duration 1), both analog and digital inputs and outputs may be affected for a short time. The bus must return to normal operation after the fault recovery.
  - If the power supply is interrupted for 10 ms (duration 2), the component must enter a defined operating mode.

### Test Step 3.5: Connection and Disconnection Shutdown and Startup

- Test basis:
  - IEC 1131-2, Sec. 6.3.2.2
  - following the example of IEC1131-2, Sec. 6.3.7.2.2, 6.3.7.2.3
  - TELEPERM XS System data
  - Instruction manual of the respective component

- Test parameters:
  - Interruption of the power supply
    - Number of connections to supply: 2
    - Number of disconnections from supply: 2
- Performing the test:
  - The test is performed in accordance with the instruction manual for the respective component and the TELEPERM XS system data. This states that the component can only be used in subracks and/or in conjunction with the power supply component. During a short power interruption with a duration of greater than 5 ms, automatic transfer of the measured signal processing to a defined shutdown status is achieved; an incorrect signal state after recovery of the supply voltage is also prevented. This behavior must be verified for each component during connection and disconnection.
  - Beginning from connection of the power supply on, then from the start of the interruption to disconnection, no change can occur that is not caused by the operating program.
  - After the power supply has been re-connected, the component must start up again as stated in the technical data of the manufacturer. No incorrect component state should arise during start-up.

Test Step 3.6: Connection and Disconnection - Gradual Voltage Change

(only for load power supply, SBG1, SBG2, SHS1)

- Test basis:
  - IEC 1131-2, Sec. 6.3.2.2, 6.3.7.3.3 - Test B
  - TELEPERM XS System data
- Test parameters:
  - Supply voltage: 24 V
  - Voltage change: 24 V → 0 V 0 V → 24 V
  - Duration: approx. 5 s
  - Dwell duration: none
  - Number of voltage changes: 3
- Performing the test:
  - This test step begins with a check of the functioning of the voltage monitoring of the digital part. The check is done with a controlled gradual switching-off and switching-on again of the 5 V power supply.
  - The behavior of the component during voltage changes is to be monitored.
  - As long as the voltage is below the minimum operating voltage, the component may be in operating condition, but no faulty or unintended states or changes to the component may occur.

### Test Step 3.7: Gradual Switching Off the Load

(only for load power supply, SBG1, SBG2 and SHS1)

- Test basis:
  - IEC 1131-2, Sec. 6.3.2.2, 6.3.7.3.2 - Test A
- Test parameters:
  - Power supply: 24 V
  - Voltage change: 24 V → 0 V
  - Duration of changes: 60 s
  - Number of changes: 3
- Performing the test:
  - The behavior of the component is to be monitored during the voltage change.
  - As long as the voltage is below the minimum operating voltage, the component may be in operating condition, but no faulty or unintended states or changes to the component may occur.

### Test Step 3.8: Gradually Bringing onto Load

(only for load power supply, SBG1, SBG2, SHS1)

- Test basis:
  - IEC 1131-2, Sec. 6.3.2.2, 6.3.7.3.2 - Test A
  - TELEPERM XS System data
- Test parameters:
  - Supply voltage: 0 V
  - Voltage change: 0 V → 24 V
  - Duration of change: 60 s
  - Number of changes: 3
- Performing the test:
  - The behavior of the component during the voltage change is to be monitored. As long as the voltage is above the minimum operating voltage, the component must start-up according to the technical data of the manufacturer. During start-up, no faulty or unintended states or changes to the component may occur.

### Test Step 3.9: Connection and Disconnection - Partial Fluctuation

(only for load power supply, SBG1, SBG2 and SHS1)

- Test basis:
  - IEC 1131-2, Sec. 6.3.2.2, 6.3.7.3.3 - Test C
  - TELEPERM XS System data

- Test parameters:
- Supply voltage: 24 V
  - Voltage change: 24 V  $\rightarrow$   $U_{ab} - 10\%$   
 $U_{ab} - 10\% \rightarrow 24$  V
  - Duration of changes: 60 s
  - Number of changes: 3
- Performing the test:
  - $U_{ab}$  is the voltage for switching off the component. This value has to be determined beforehand. During the voltage changes, no faulty or unintended states or changes to the component may occur.

#### Test Step 3.10: Electrical Isolation

- Test basis:
  - IEC 1131-2, Sec. 6.3.2.2, 6.4.1.2
  - DIN VDE 0160, Sec. 7.6
  - Instruction manual of the respective component
- Test parameters:
  - Test voltage at inputs against grounding point: Is determined according to the voltage of the component- internal electric circuits (VDE 0160) (\*)
  - Test duration 1 minute  
(\*)(1250 V AC for digital input)  
(2000 V AC group against group, 1500 V AC for group against ground for digital output, relay)
- Performing the test:
  - The test voltage is applied between each input (one after the other) and grounding point for a duration of 1 minute in each case. No flashovers may occur. Following this test, the functioning of the component is to be tested by means of an intermediate functional test.

#### Test Step 3.11: Flammability of Insulating Material

- Test basis:
  - IEC 1131-2, Sec. 6.3.5.5.5
  - IEC 707
  - IEC 950
  - TELEPERM XS System data
- Test parameters:
  - Flammability at least class V-1, V-0 aimed at.

- Performing the test:
  - To meet the requirements of this test step, it is sufficient to obtain from the manufacturer a certificate of conformity.

#### Test Step 3.12: Self-Heating

- Test basis:
  - IEC 1131-2, Sec. 6.3.2.2, 6.3.5.5.6
  - DIN VDE 0160, Sec. 5.2.3
  - Instruction manual of the respective component
- Test parameters:
  - Nominal operating conditions
- Performing the test:
  - In this test step (which can be replaced by the thermography performed as part of the limit load analysis) it is necessary to verify that the limit temperature stated in Section 4.4.2 of IEC 1131.2 and Section 5.2.3 of DIN VDE 0160 is not exceeded for the materials used.

#### Test Step 3.13: Durability of the Protective Coating

- Test basis:
  - IEC 1131-2, Sec. 6.3.5.5.7
  - Instruction manuals of the respective component
- Test parameters:
  - None
- Performing the test:
  - The long-time durability of the protective coating has to be proven for three printed-circuit boards without inserted components, as specified in IEC 1131-2 Sec. 6.3.5.5.7.
  - The test may be substituted by the submission of a test certificate issued by an independent test laboratory, certifying that the same or an equivalent test was passed.

#### Test Step 3.14: Effectiveness of the Protective Conductor

- Test basis:
  - IEC 1131-3, Sec. 6.3.6.1.2
  - Instruction manual of the respective component
- Test parameters:
  - Test current: 30 A
  - Test duration: 2 min.

- Performing the test:
  - The test must demonstrate the effectiveness of the protective conductor within the system. The test must conform to IEC 1131-2 Sec. 6.3.6.1.2, which states that the leakage resistance of 0.1  $\Omega$  must not be exceeded.

Test Step 3.15: Interchange of DC Supply Voltage Connections (\*)

(\*) only for S430, S470, SHS1; not tested: S451, SBG1, SBG2

- Test basis:
  - IEC 1131-2, Sec. 6.3.7.4.1
  - TELEPERM XS System data
  - Instruction manual of the respective component
- Test parameters:
  - Nominal operating conditions
- Performing the test:
  - The test must check whether the component behaves as designed when the DC supply voltage connections are interchanged.

2.2.2.3.4 Climatic Test

Test Step 4: Climatic Test

Test Step 4.1: Constant Cold without Function

- Test basis:
  - KTA 3503, Sec. 5.5.2
  - IEC 68-2-1, Test Aa
  - IEC 1131-2, Sec. 6.3.4.2
  - TELEPERM XS System data
  - Instruction manual of the respective component
- Test parameters:
  - Operating state: Component not in operation
  - Temperature: - 25 °C + 3 °C
  - Duration: 96 + 1 hours
- Performing the test:
  - The component is packed ready for dispatch.
  - The component at room temperature is put into the test chamber in which the ambient temperature is - 25 °C. The component remains in its packing.
  - Immediately after climatic stressing, the module is allowed to recover without electric function. Drops of water and condensation may be removed. The module must be stored at room temperature and humidity until the temperature has equalized.
  - Following the test, a visual inspection and an intermediate functional test are performed.

#### Test Step 4.2: Constant Dry Heat Without Function

- Test basis:
  - KTA 3503, Sec. 5.5.3
  - IEC 68-2-2, Test Ba
  - IEC 1131-2, Sec. 6.3.4.2
  - TELEPERM XS System data
  - Instruction manual of the respective component
- Test parameters:
  - Operating state: Component not in operation
  - Temperature: + 70 °C + 2 °C
  - Duration: 96 + 1 hours
- Performing the test:
  - The component is packed ready for dispatch.
  - The component at room temperature is put into the test chamber in which the ambient temperature is + 70 °C.
  - Immediately after the climatic stressing, the module is allowed to recover without electrical function. The module must be stored at room temperature and humidity until the temperature has equalized.
  - Following the test, a visual inspection and an intermediate functional test are performed.

#### Test Step 4.3: Constant Humid Heat Without Function

- Test basis:
  - KTA 3503, Sec. 5.5.4
  - IEC 68-2-3, Test Ca
  - TELEPERM XS System data
  - Instruction manual of the respective component
- Test parameters (according to IEC 68 Part 2-3):
  - Operating state: Component not in operation
  - Temperature: + 40 °C + 2 °C
  - Relative humidity: 93 % + 2 % - 3 %
  - Duration: 96 hours
- Performing the test:
  - The component is packed ready for dispatch. The component at room temperature is put in the test chamber in which the ambient conditions are as stated above.
  - Immediately after the climatic stressing, the module is allowed to recover without electrical function. The module must be stored at room temperature and humidity until the temperature has equalized.
  - Following the test, a visual inspection and an intermediate functional test are performed.
- Tests performed at a temperature of 40 °C ± 2 °C are “humidity tests” or temperature change tests. In this step the crucial point is not the absolute temperature, but the humidity

and change in temperature. Absolute temperature and aging are tested at temperature of 55 °C in Test Step 4.8.

Test Step 4.4: Constant Humid Heat with Function

- Test basis:
  - KTA 3503, Sec. 5.5.4
  - IEC 1131-2, Sec. 6.3.2.2
  - TELEPERM XS System data
  - Instruction manual of the respective component
- Test parameters:
  - Operating state: Component in operation with cyclic variation of the supply voltage between 20 V and 30 V every 6 hours; the functioning of the component is to be monitored during stressing.
  - Temperature: + 40 °C ± 2 °C
  - Relative humidity: 93% + 2% - 3%
  - Duration: 24 hours ± 30 minutes
- Performing the test:
  - The component at room temperature is put in the test chamber in which the ambient conditions are as stated above.
  - Following the test, a visual inspection is performed.

Test Step 4.5: Stressing by Temperature change without Function

- Test basis:
  - IEC 68-2-14, Test Na
  - IEC 1131-2, Sec. 6.3.4.3
  - TELEPERM XS System data
  - Instruction manual of the respective component
- Test parameters:
  - Operating state: Component not in operation
  - Low temperature: - 25 °C ± 3 °C
  - High temperature: 70 °C ± 2 °C
  - Exposure time: 3 hours ± 30 minutes for each temperature
  - Transport time: < 3 min
  - Cycles: 2
- Performing the test:
  - according to IEC 68-2-14.
  - Following the test, a visual inspection and an intermediate functional test are performed.

Test Step 4.6: Stressing by Temperature Change with Function

- Test basis:
  - IEC 68-2-14, Test Nb
  - IEC 1131-2, Sec. 6.3.2.2, 6.3.4.3
  - TELEPERM XS System data
  - Instruction manual of the respective component
  
- Test parameters:
  - Operating state: Component in operation
  - Low temperature: 5 °C ± 2 °C
  - High temperature: 40 °C ± 2 °C
  - Exposure time: 3 hours ± 30 minutes for each temperature
  - Rate of temperature change: 3 °C / min. ± 0,6 °C
  - Cycles: 5
  
- Performing the test:
  - according to IEC 68-2-14. The functioning of the component is to be monitored during the test.
  - Following the test, a visual inspection and an intermediate functional test are performed.

Test Step 4.7: Cyclic Humid Heat without Function

- Test basis:
  - KTA 3503, Sec. 5.5.5
  - IEC 68-2-30, Test Db, variant 1
  - IEC 1131-2, Sec. 6.3.4.4
  - Instruction manuals of the respective module
  
- Test parameters:
  - Operating state: Component not in operation
  - Temperature and humidity variation: according to Fig. 2a of IEC 68- 2-30
  
- Performing the test:
  - The component at room temperature is put in the test chamber in which the ambient conditions are as stated above.
  - Following the test, a visual inspection and an intermediate functional test are performed.

Test Step 4.8: Cyclic Dry Heat with Function

- Test basis:
  - KTA 3503, Sec. 5.5.6
  - IEC 1131-2, Sec. 6.3.2.2
  - TELEPERM XS System data
  - Instruction manual of the respective component

- Test parameters:
  - Operating state: Component in operation with cyclic variation of the supply voltage between 30 V and 20 V every 24 hours. The functioning of the component must be monitored during the stressing.
  - Temperature:  $+ 25\text{ °C} \pm 3\text{ °C} \rightarrow + 55\text{ °C} \pm 2\text{ °C}$
  - Duration at + 50 °C: 20 hours
  - Duration at + 25 °C: 2 hours
  - Total duration: 1000 hours
- Performing the test:
  - The temperature in test chamber containing the component (supply voltage 30 V) is raised from + 25 °C to 55 °C. After remaining at this temperature for 20 hours, it is cooled back down to + 25 °C. After another two hours at this low temperature, the cycle is to be repeated with the supply voltage changed to the minimum (20 V).
  - After 100 hours of test duration, a visual inspection and an intermediate functional test are performed.

#### 2.2.2.3.5 Tests of Mechanical Stress

##### Test Step 5: Tests of Mechanical Stress

##### Test Step 5.1: Simulation of Operational Vibrations

- Test basis:
  - IEC 68-2-6, Test FC
  - IEC 721-3-3, Sec. A 2.5 Class 3M3
  - IEC 1131-2, Sec. 6.3.2.2, 6.3.5.1
  - IEEE 344, Sec. 7.1.5
  - Siemens Specification Environment
- Test parameters:
  - Components (mounted in a subrack and wired) according to IEC 1131-2, 6.3.5.1;
    - Operating state: in operation
    - Stressing: sinusoidal excitation, sliding frequency
    - Amplitude of deflection: 0,075 mm at 10 to 58 Hz
    - Amplitude of acceleration: 1 g of 58 - 150 Hz
    - Sweep rate: 1 octave / min.
    - Duration: 20 cycles for each main axis
- Performing the test:
  - The equipment is stressed in all three main axes.
  - The functioning of the component is to be monitored during the test.
  - The technical data must not be violated.
  - Following the test, a visual inspection and an intermediate functional test are performed.

**Test Step 5.2: Resistance to Vibration in the Frequency Range 5 Hz to 35 Hz**

- Test basis:
  - IEC 69-2-6, Test Fc
  - KTA 3503, Sec. 5.6.1, 5.6.2
  - TELEPERM XS System data
  
- Test parameters:
  - Operating state: component in operation
  - Stressing: sinusoidal excitation, sliding frequency
  - Amplitude of deflection: 10 + 2,5 mm
  - Amplitude of acceleration: 10 m / s<sup>2</sup>
  - Sweep rate: 1 octave / min.
  - Duration: 1 cycle for each main axis
  
- Performing the test:
  - The component is mounted in a subrack and wired.
  - The equipment is stressed in all three main axes.
  - The functioning of the component is to be monitored during the test.
  - The technical data must not be violated.
  - Following the test, a visual inspection and an intermediate functional test are performed.

**Test Step 5.3: Vibration in the Frequency Range 5 to 100 Hz**

- Test basis:
  - KTA 3503, Sec. 5.6.1, 5.6.3
  - IEC 68-2-6, Test Fc
  - TELEPERM XS System data
  - Instruction manual of the respective component

**Test parameters:**

- Operating state: component in operation
  - Frequency range: 5 ... 100 Hz
  - Stressing: sinusoidal excitation, sliding frequency
  - Amplitude of deflection: 10 ± 2,5 mm
  - Amplitude of acceleration: 20 m / s<sup>2</sup>
  - Sweep rate: < 10 octave / min.
  - Duration: 1 cycle for each main axis
- 
- Performing the test:
    - The component is mounted in a subrack and wired.
    - The equipment is stressed in all three main axes.
    - The functioning of the component is to be monitored during the test.
    - The technical data must not be violated.
    - Following the test, a visual inspection and an intermediate functional test are performed.

#### Test Step 5.4: Stressing Caused by Transport

- Test basis:
  - IEC 68-2-6, Test Fc
  - TELEPERM XS System data
  
- Test parameters:
  - Operating state: component not in operation
  - Frequency range: 5 ... 9 Hz
  - Deflection: 3,5 mm
  - Frequency range: 9 ... 500 Hz
  - Amplitude of acceleration: max. 10 m / s<sup>2</sup>
  - Stressing: sinusoidal
  
- Performing the test:
  - Each component is to be packed individually and tested separately. Following the test, a visual inspection and an intermediate functional test are performed.

#### Test Step 5.5: Shock Resistance

- Test basis:
  - KTA 3503, Sec. 5.6.1, 5.6.4
  - IEC 68-2-7, Test Ea
  - TELEPERM XS System data
  
- Test parameters:
  - Operating state: component not in operation
  - Amplitude of acceleration: 294 m / s<sup>2</sup>
  - Pulse shape: half-sine
  - Pulse duration: 11 ms
  - Number of shock pulses: 3 for each direction of the main axes
  
- Performing the test:
  - The components are to be stressed individually and without packaging.
  - Following the test, a visual inspection and an intermediate functional test are performed.

#### Test Step 5.6: Seismic Stress During Operation

- Test basis:
  - IEEE 344, Sec. 7.1, 7.4, 7.5, 7.6, 10, Appendix D
  - TELEPERM XS System data

- Test parameters:
  - Operating state: in operation
  - Frequency range: 5 - 35 (100) Hz, multiple frequency
  - Stressing: 3 axes, each staggered by 90°
  - Duration per axis: min. 1 minute
  - Amplitude of acceleration: For amplitude of acceleration, the natural frequency (18 Hz) and the magnification factor for the supporting structure (1,56) of the I&C cabinets to be used, as well as the acceleration spectrum for the design of the PWR 1300 are taken into account, according to the manufacturer's request.
- Performing the test:
  - The components are mounted in a subrack and wired.
  - The equipment is stressed in both 2 main axes, one after the other. During the test, the functioning of the component must be monitored. The technical data must not be violated.
  - Following the test, a visual inspection and intermediate functional test are performed.
  - The test results are documented according to IEEE 344 Sec. 10.

#### 2.2.2.3.6 Tests to Demonstrate the EMC

##### Test Step 6: Tests to Demonstrate the EMC

- Test basis:
  - see individual test steps
- Test parameters:
  - see individual test steps
- Performing the test:
  - For this tests, the components are mounted in the cabinet. The doors are closed during the test unless the test procedure requires otherwise.
  - The functioning is to be monitored during the tests.
  - The inputs and outputs of the components are connected with cables or equivalent circuits, in a manner that depends on the component type.

##### Test Step 6.1: Interference Emission

- Test basis:
  - DIN VDE 0875 Part 11 / EN 55011 / CISPR 11 (mod.)
  - DIN EN 50081 Part 1 / VDE 0839 Part 81-1 / EN 50081-1
  - DIN EN 50081 Part 2 / VDE 0839 Part 81-2 / EN 50081-2
  - IEC 1131-2, Sec. 393
  - TELEPERM XS System data
  - Instruction manual of the respective component

TELEPERM XS: A Digital Reactor Protection System

- Test parameters:
  - Classification: Group 1, Class A/B
  - Measurement of the radio interference voltage in the range: 10 kHz ... 30 MHz
  - Measurement of the interference field strength in the range: 30 MHz ... 1000 MHz
- Performing the test:
  - This test serves to demonstrate that permissible interference emission limit values are not violated. The classification in class A or B depends on the test results.
  - The test is to be conducted in a measuring station.
  - During the test, the components are to be operated such that the interference level generated is as high as possible.
  - The output power must be varied.

Test Step 6.2: Interference Caused by Transient Interference Voltages (Burst)

- Test basis:
  - DIN EN 50082 Part 2 / VDE 0839 Part 82-1
  - IEC 1000-4-4, Class 3 / 4 (for signal lines > 3 m)
  - EN 61000-4-4
  - IEC 1131-2, Sec. 3.9.1, 6.3.6.2.3
  - TELEPERM XS System data
  - Instruction manual of the respective component
- Test parameters:
 

– Interference on power lines:	Test voltage	2 kV
	2,5 kHz	Test frequency
– Interference on signal lines > 3 m:	Test voltage	2 kV
– leading out of the cabinet:	Test frequency	2,5 kHz
– Interference on signal lines < 3m:	Test voltage	1 kV
– not leading out of the cabinet:	Test frequency	5 kHz
– Evaluation criterion:	B	
- Performing the test:
  - The test is to be conducted in a measuring station.
  - During the test, the component is to be operated such that its level of electromagnetic susceptibility is as high as possible.
  - The arrangement of the components must be varied.
  - The test is conducted on all inputs and outputs as well as direct voltage and alternating voltage supply lines.

Test Step 6.3: Conducted Interference - Surge Voltage

- Test basis:
  - DIN EN 50082 Part 2 ( VDE 0839 part 82-1)
  - ENV 50142
  - IEC 1000-4-5
  - EN 61000-4-5
  - TELEPERM XS System data

- Instruction manual of the respective component
- Test parameters:
  - Surge voltage between the inputs / outputs and L-: DC: 1kV, 1,2/50  $\mu$ s AC: 2kV, 1,2/50  $\mu$ s
  - Interference voltage between the inputs / outputs and L-: DC: 1KV, 1 MHz AC: 2 kV, 1 MHz
  - Evaluation criterion: B
- Performing the test:
  - The test is to be conducted in a measuring station.
  - During the test, the component is to be operated such that its electromagnetic susceptibility is as high as possible.
  - The arrangement of the components must be varied.
  - The test voltages (symmetric / asymmetric) are applied between the inputs or respective outputs (one after the other) and L-.

#### Test Step 6.4: Conducted Interference - Coupling

- Test basis:
  - DIN EN 50082 Part 1 / VDE 0839 Part 8.2-1
  - IEC 1000-4-6
  - EN 61000-4-6
  - VG 95373 Part 14, Part 24
  - Instruction manual of the respective component
- Test parameters:
  - Frequency range 1: 10 Hz ... 400 MHz
  - Method of measurement: lines / shields LF 02G
  - Frequency range 2: 150 Hz ... 400 MHz
  - Method of measurement: enclosure LF 06G
  - Limit value class: 3 (for both procedures)
  - Evaluation criterion: A
- Performing the test:
  - The test is to be conducted in a measuring station.
  - During the test, the component is to be operated such that its electromagnetic susceptibility is as high as possible.
  - The arrangement of the components must be varied.
  - The test voltages (symmetric/asymmetric) are applied between the inputs or respective outputs (one after the other) and L-.

#### Test Step 6.5: Interference Caused by Electrostatic Discharge

- Test basis:
  - DIN EN 50082 Part 1 / VDE 0839 Part 82-1
  - DIN VDE 0843 Part 2 / EN 60801 Part 2 / IEC 801-2, Severity 3
  - IEC 1131-2, Sec. 3.91, 6.3.6.2.1

- IEC 1000-4-2
- EN 61000-4-2
- TELEPERM XS System data
- Instruction manual of the respective component
- Test parameters:
  - Test voltage: 6 kV contact discharge 8 kV air discharge
  - Number of discharges: 1B per test voltage
  - Evaluation criterion: B
- Performing the test:
  - The test must be conducted on a measuring station.
  - During the test, the component is to be operated such that its electromagnetic susceptibility is as high as possible.
  - The arrangement of the module must be varied.
  - The test is to be performed with valid application software.
  - The test must be performed on all surfaces that are accessible during normal operation with open doors.

#### Test Step 6.6: Interference Caused by Electromagnetic Fields

- Test basis:
  - DIN EN 50082 Part 1 / VDE 0839 Part 82-1 / EN 50082-1
  - DIN VDE 0843 Part 3 / IEC 801-3, Severity 3
  - IEC 65A/77B (Sec.) 145/110 / IEC 801-6 (Draft) / ENV 50141
  - IEC 1131-2, Sec. 6.3.6.2.2
  - IEC 1000-4-3
  - EN 61000-4-3
  - ENV 50140
  - ENV 50240
  - TELEPERM XS System data
  - Instruction manual of the respective component
- Test parameters:
  - Frequency range: 27 MHz ... 1000 MHz
  - Modulation: 1 kHz, 80%
  - Test field intensity: 10 V/m
  - Evaluation criterion: A
- Performing the test:
  - The test must be conducted in a shielded test cell.
  - During the test, the component is to be operated such that its electromagnetic susceptibility is as high as possible.
  - The arrangement of the module must be varied.

### Test Step 6.7: Interference Caused by Damped Oscillation(\*)

(\*) only for digital input / output modules

- Test basis:
  - IEC 255-4
  - IEC 1131-2, Sec. 3.9.1, 6.3.2.2, 6.3.6.2.4
  - TELEPERM XS System data
  - Instruction manual of the respective component
- Test parameters:
  - Peak value of the interference voltage: 1 kV
  - Frequency of the interference voltage: 1 MHz
  - Duration: > 2 s
- Performing the test:
  - The damped oscillation is to be coupled into the supply voltage inputs and all digital inputs / outputs with  $U_0 \geq 24 \text{ V}$ .
  - The test equipment is to be arranged according to IEC 1131-2. The analog inputs and outputs and the digital inputs may be influenced temporarily during the interference but must return to normal operation when the interference is over.

#### 2.2.2.3.7 Final Functional Test Under Nominal and Limit Loadings

### Test Step 7

- Test basis:
  - KTA 3503, Sec. 5.2.2, 5.2.3, 5.7.3, 5.8 e)
  - IEC 1131-2, Sec. 6.3.7.1.1
  - TELEPERM XS System data
  - Instruction manual of the respective component
- Test parameters and performing the test:
  - See 0.

## 2.3 **Channel Integrity, Isolation and Real-Time Performance**

### 2.3.1 Isolation

Electrical isolation is one of the means utilized to achieve isolation between class 1E circuits and non class 1E circuits. Electrical isolation is performed such that the maximum credible voltage or current transient applied to non class 1E circuits cannot degrade the operation of class 1E circuits.

### 2.3.2 Limiting Response Time

The maximum permissible response time is prescribed by the process engineering of the specific application. Thus for each safety function the limiting response times will be shown to be consistent with the safety requirements application specifically.

### 2.3.3 Digital Computer Timing Requirements

The safety functions are distributed on the application architecture after being engineered. Thus afterwards several load analyses can be performed for bus load, processor load, and response times for safety functions.

During the plant-independent system test, it was proven that in general this analytic determination is sufficiently exact. For the specific application, the response time analyses will be verified by random tests.

### 2.3.4 Architecture

For software architecture see Chapter 3.1, "Functional Software Characteristics." For hardware architecture, application-specific descriptions will be compiled.

## 2.4 **Reliability**

The fundamental quality requirement for a safety I&C system is the reliability with which it performs its assigned safety functions. To assess this reliability, two mutually complementary methods are in standard use in Germany. These two methods are the probabilistic and the deterministic reliability analysis. Probabilistic analysis is used to quantify the reliability, with the "non-availability on demand" used as the standard measure of this. This term is defined as the probability of a given system not being able to perform its safety function when it is called upon to operate. This quantification of the quality characteristics is used as a yardstick for assessing different equipment designs.

Practical determination of reliability requires suitable modeling of the circumstances that could cause a system to lose its ability to perform the safety function demanded of it. Probabilistic analysis essentially assesses the physical and chemical aging phenomena which can lead to degradation of system characteristics with the passage of time. If the system characteristic affected is needed for performance of the safety function, the system is then no longer able to perform its intended function properly. Safety assessment of these "aging-related failures" requires qualitative analysis of the failure modes and effects, and a quantitative analysis of failure frequency. With respect to qualitative analysis, this report describes the system characteristics relevant to failure behavior. The fault and failure effects are then analyzed on the basis of the system characteristics. On the basis of this analysis it is then shown how the choice of suitable hardware architectures can have a positive impact on the response to faults and failures.

Deterministic analysis serves to assess design errors. German standards stipulate that equipment with high importance to safety must perform its task even if a plausible design error is assumed. This report shows the system characteristics that delimit the confinement area for a design error and the architecture features required for a safety I&C system to permit postulated design errors to be accommodated.

The substance of this chapter was subject to a conceptual review by the GRS ("Gesellschaft für Reaktorsicherheit", a public company whose business is reactor safety). Their concluding report /7/ on the "Digital Safety I&C" confirms the validity of the design.

## 2.4.1 Fundamentals and Definitions

### 2.4.1.1 Design Errors, Faults and Failures

In the following analysis, systems are assumed to contain latent faults from the very start, i.e., design errors or manufacturing defects. This approach is not intended to imply that errors of this type actually exist, it is merely a way of considering the common cause faults that must be postulated according to German standards.

Figure 2.4 shows the different states for a system with latent faults. The system is designated as "defective" or "faulty" if it does not meet all design features. Conversely, a system that is "fault-free" or "correct" meets all design features at all times. Thus on demand a fault-free system will always perform its function correctly. A defective system, on the other hand, could "fail" if a requested feature does not exist or does not exist in the correct form. In this context, a failure is defined as the system's loss of ability to perform its intended functions properly, i.e., an admissible loading that results in a faulty response. A common mode failure is defined as the simultaneous failure of two or more trains of a redundant system such that its intended safety function is not performed correctly.

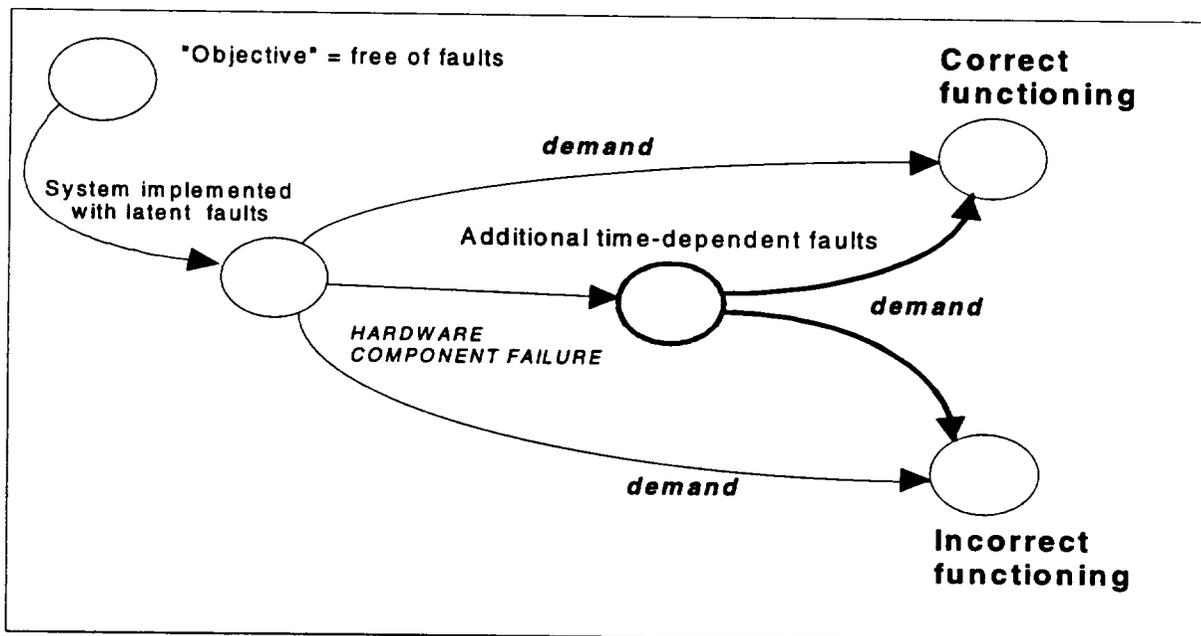


Figure 2.4 States of Systems with Latent Faults

In addition to latent faults, time-dependent faults can arise due to aging or wearing that results in component failures. The term component failure is understood here as an inadmissible change caused by physical or chemical processes by the effect of aging, wearing or external influences. Initially, no distinction is made as to whether the component failure rate is correspondent to experience or is increased due to production deficiencies. The terms fault and component failure are used partly as synonyms within this document. The term "component failure" is used in preference where the transition from fault-free to faulted conditions is the point of interest. The term "fault" is used to denote a changed system feature. "Error" means an incorrectly designed system feature ("design error") relating to the engineering process. The term "error" is also used in the field of data transmission ("bit error") if the cause is neither a software nor a hardware fault but for example drift effects. "System failure" means the non-performance of the required action by the entire system - not just by one of the redundant trains - under admissible loadings. Admissible loading means that a component is operated within the allowable range of operating conditions, which is typically specified by the component data sheet or user manual. "Failure" alone is equivalent to "component failure" never meaning a failure of the entire safety system.

In this analysis the emphasis is on the identification and assessment of failure mechanisms, i.e., potential faults and design errors are assessed in terms of their effect on the safety function. This results in design requirements which delimit the confinement areas for potential failure mechanisms to subfunctions or a redundant subsystem such that nevertheless the overall safety objectives are met.

#### 2.4.1.2 Masking of Faulty Signals

Faulty signals can be propagated, thus enlarging their area of impact. To prevent this, barriers to the propagation of faulty signals are used in safety-related systems. These confine the spread of faults to within predefined areas. Isolation and decoupling measures are examples of such barriers, as they serve to prevent propagation of energy-related faults.

Majority voting mechanisms form special fault propagation barriers that use the redundancy in information to identify incorrect information and exclude this from further processing. This ensures that the function processor down stream of the majority voting processes correct information. Examples of such equipment are selection features that select the average signal from a number of redundant signals. In computer-based automation equipment, other more effective checking mechanisms are also implemented to prevent incorrect information from being propagated. Exclusion of incorrect information from further processing is termed "fault masking". This permits the propagation of faulty information to be confined to defined areas within a networked system. The fault detection mechanisms in communication networks are typical examples of this.

#### 2.4.2 Mechanisms Leading to System Failures

Safety I&C systems in nuclear power plants are designed with multiple redundancy and physical separation of their hardware. This ensures that random single failures of components cannot result in system failure. Safety I&C systems are designed according to stringent quality criteria that meets the requirements of nuclear standards.

On the basis of these features, the mechanisms leading to system failure of a safety I&C system as defined above are first explained in more detail. On this basis, the system features of TELEPERM XS that ensure reliability are described and the failure modes and effects are analyzed.

#### 2.4.2.1 System Failures Due to Component Failures

System failures due to component failures include all mechanisms that are caused by inadmissible physical and chemical changes in the hardware characteristics. If these changes are the result of aging or wearing, the component failures are not statistically correlated. Component failures of this type are denoted "random component failures". Component failures due to external events, on the other hand, may be statistically correlated.

##### 2.4.2.1.1 Random Single Component Failures

Because of the redundant design of safety I&C systems, random single failures can cause a system failure only if allowed to cumulate to an inadmissible extent. The low failure rates for TELEPERM XS due to low operating temperature, and high quality of manufacturing ensure that the rate of random component failures is low. The specific measures taken to detect and mask faults are described in Section 2.7. The impact of random failures on reliability of the safety function is verified quantitatively in the course of the reliability analysis. For typical safety functions, the reliability is higher than for equivalent hardwired systems.

##### 2.4.2.1.2 System Failures Due to External Events

System failures due to external influences on the hardware are ruled out in the same way as in proven hardwired safety I&C systems, namely by the physical separation of mutually redundant equipment, by preventive measures during system planning and by conservative design with regard to the assumed environmental conditions. The consistent use of fiber optics for communication between the redundant trains of equipment provides a reliable means for preventing the energy-related propagation of failures and is a considerable improvement over hardwired systems. A detailed account of the definition of suitable architectures is given in Section 2.7.2 and of system planning and system integration in the plant in Section 1.3.

##### 2.4.2.2 System Failures Due to Latent Faults

System failures due to latent faults include all mechanisms resulting in design errors or manufacturing-related defects that can occur without accompanying additional changes in the hardware characteristics. Due to the fact that a system failure can only occur in the event of simultaneous failure of several redundant items of equipment to function properly, a distinction can be made with regard to the cause of the associated simultaneity. Here one can distinguish between mechanisms in which a special loading is responsible for the simultaneity, and mechanisms in which the latent fault is itself the cause of the simultaneity. This distinction is useful because different measures are used to delimit the confinement areas on which the system failure have an impact for each of these two mechanisms.

#### 2.4.2.2.1 System Failures Due to Special Loadings

A system failure due to a special loading is the typical failure mechanism that operates in complex systems in which hidden design errors and production deficiencies manifest themselves in the event of a rare and untested loading. For these effects, a distinction can be made between faults that can affect only a single safety function and faults that can influence more than one or all safety functions processed within a single processor system as a result of inadmissible interference of the application software with the system services.

##### Erroneous Design of the Safety Function

Design errors or production deficiencies that can only cause a failure of the faulty safety function in the event of a special loading are grouped together under the term "erroneous design of the safety function." These are typical faults that arise from errors in the fluid system requirements or incorrect implementation of the fluid system requirements. The probability of such faults is expected to be reduced compared with proven hardwired technology, because of the improved tool-based design process for digital I&C systems with the engineering system SPACE and its integrated tools for the validation of the application software. Nevertheless, the use of diversity (i.e., incident identification and control on the basis of physically diverse parameters) will remain the accepted method for accommodation of a postulated design error within the approval and licensing procedure.

##### Impermissible Interference of Special Loadings on System Behavior

In digital safety I&C systems, varying input signals are the only way that the I&C system is influenced by the plant process. These input signals, which change over time, are denoted "data trajectories". From the point of view of an outsider, there are two ways in which the safety I&C systems might fail as a result of data trajectories:

Incorrect realization of the required function because of a design error (for example, an incorrect designed setpoint value). In this case, the I&C system further on processes the incorrect application software. A necessary initiation may be delayed or omitted, but all other safety functions are performed correctly.

- The TELEPERM XS safety system is designed to revert to "processor halt" state for the following conditions: data-dependent increase in CPU load,
- impermissible increase in bus loading,
- use of RAM space which is required to remain free,
- inadmissible data operations that require intervention in the operating system via exceptions.

The cause of the system failure discussed here is interference from the application software on the processor system caused by data trajectories. One of the most important objectives in designing the TELEPERM XS system right from the very start was to rule out such interference, as this is of paramount importance in reliability assessment. The relevant system characteristics were therefore set by the objective of ruling out inadmissible interference from

special loadings on system behavior. A detailed description of the procedures implemented to guarantee freedom from interference and the test principle for verifying absence of interference are described under Section 1.0.

The mechanisms considered here that could cause a failure in the event of a special loading are of special importance in the reliability assessment because they constitute a correlation between the thermodynamic process in the plant and the failure of the safety I&C system. As such correlations can be eliminated for the TELEPERM XS system, the reliability of the TELEPERM XS system is set by its non-availability, a parameter which is very much easier to determine. This is one of the reasons why this mechanism was considered especially important for TELEPERM XS development and in this report.

The special nature of this type of mechanism is illustrated by the following comparisons:

- Proven analog safety I&C system technology is characterized by the fact that for actuation of protection actions in the hardware modules concerned in response to a demand, voltages and currents must be varied and relays - mainly in older systems - must pick up or drop out mechanically. Unlike for continuous closed-loop controls, it is not possible to observe that safety I&C systems are functioning as designed until a demand actually occurs or tests are run, so that the probability of system failure in the event of a demand is also relevant for quantification. A lot of energy-related interference due to the loading of the safety I&C system exist.
- The experience with the behavior of standard digital systems both in office computer systems and standard industrial automation systems has been that the superposition of several design requirements can cause system failures. To optimize both the cost/performance relation and the time response behavior, systems of this type manage their resources dynamically, responding to a varying load by activating functions and releasing resources. In this case the system level is not free from interference with the data. Standard experience with system failures of digital systems is therefore of no relevance to safety I&C systems.
- Freedom from interference of a loading on the behavior of the system is a verified system characteristic that forms the basis for deriving the system reliability from the observable system behavior in normal strictly cyclic operation. The reaction to demands from the plant process is merely to change data in the application software, with the processor and bus load being constantly in strictly cyclic operation. The only remaining interference due to a demand caused by the plant process occurs in the final signal output amplifiers of the binary output module that must supply power to the interposing relays (so as to operate the electric actuators to initiate a protection action in the switchgear). This actuation of the interposing relays in the switchgear is still comparable with analog protection technology, both in terms of design and of its assessment.

Because of the freedom from interference from the level of application software on the operating system and the hardware resources, the system availability in normal operation is the relevant factor for the probability of the system behaving as designed in the event of a demand. The system availability during normal operation is determined by the system failures of the hardware components and the probability of a failure due to system-internal causes without any connection with the demands.

#### 2.4.2.2.2 System Failures Due to System-Internal Mechanisms

Design errors or production deficiencies that can cause a failure due to aging but independent of any loading are grouped together under the term "system failure due to system-internal mechanisms." These include faults that cause resources to be used up slowly or that can have an effect on the system time. Examples of such system failures are:

- faults in releasing system resources or
- faults in time switchover, e.g., between daylight saving and standard time.

In order to be able to rule out such mechanisms to the extent that the system reliability can not be effected in quantitative terms, TELEPERM XS has been designed without an internal system clock and calendar. The time dependencies that are required to implement the fluid system requirements are derived as multiples of the processing cycles. Process data that have to be transmitted (for example, to the process information system or to the service unit) are provided with a timestamp for chronological sequencing with other plant data or for logging, but with this performed in the gateways outside the automatic action path. The TELEPERM XS has also been successfully tested for year 2000 compliance.

#### 2.4.2.3 Summary of Assessment of Failure Mechanisms

From the above considerations it is possible to draw the conclusion that the reliability of a three-train or four-train redundant safety I&C system in TELEPERM XS technology is primarily determined by random failures. The improved design process has significantly reduced the probability of system failure due to erroneous design of the safety functions compared with hardwired equipment systems.

For a cyclic TELEPERM XS safety I&C system, the non-availability calculated from component failure rates and observable in operation is the relevant quantity for quantification of system reliability. For assessment of reliability for a cyclic digital I&C system it makes no difference whether this system is a closed-loop control system that acts on the process continuously or a safety I&C system. This is one important difference to a hardwired I&C system in that such a system only performs its function once a year outside of system testing, or possibly not at all.

#### 2.4.3 Relevant System Characteristics

The basic components of TELEPERM XS are described with their features relevant to reliability on the basis of the specific failure mechanisms. Potential fault and failure effects are derived from this. Subsequently the way in which fault and failure behavior is influenced by the choice of a suitable system architecture is explained for typical system architectures.

##### 2.4.3.1 Basic Components of TELEPERM XS

TELEPERM XS basically consists for four types of components, the subracks, function processors, communication means and input / output modules. These basic components can be configured application-specifically to constitute safety I&C systems. The specific architecture of the I&C system is determined both by the safety function to be implemented with its associated reliability characteristics and by the failure combinations required to be tolerated. Architectures that include several redundant and physically separated computation nodes are

typical. A computation node consists of a subrack with one or more function processors and of the requisite I/O modules and communication means. TELEPERM XS makes a distinction between two types of computation node, the computers of the automatic action pass (automation computers) and those of the monitoring and service interface. Automation computers perform the actual safety function. They do not have any direct serial interface to unclassified I&C equipment. The monitoring and service interface computers are used as the interface with the operational I&C equipment and for monitoring the automation computers. Every automation computer of TELEPERM XS is connected with one and only one monitoring and service interface computer, and with the service unit via this monitoring and service interface. The monitoring and service interface passes on alarms regarding the fluid system or faults in the I&C system from the automation computers and also operating and test requests from the service unit.

To illustrate the method of operation of TELEPERM XS system, the way in which each component of TELEPERM XS works and the interfaces that it has to other components will now be described. The software architecture and the interaction of the hardware and the software are also detailed.

#### 2.4.3.1.1 Subracks

The subrack contains the electronic printed-circuit boards (PCBs). In addition to the mechanical rack, it also has the following components:

- parallel 32-bit wide backplane bus for communication between the PCBs,
- arbiter for coordinating accesses to the backplane bus,
- DC/DC converter for the power supply (5 V, 15 V) for the PCBs,
- monitoring equipment for monitoring the external and internal voltages, fan operation and the ambient temperature,
- fan tier for cooling the PCBs.

Cooling fans are energized with Class 1E power. Fan operation is monitored and failures resulting in an excessive temperature are detectable. The subrack is the basis of a computation node. It is used for mechanical fixture of the PCBs, supplies the requisite auxiliary power to them, enables communication via the parallel backplane bus, cools the modules and protects them from electromagnetic interference.

#### 2.4.3.1.2 Function Processors

The processing module (SVE1) is the only application programmable module within the TELEPERM XS system, used as function processor for executing the safety functions in the form of function diagram modules. It is a plug-in PCB with the following main components:

- 32 bit-CPU (Intel I486 DX),
- static RAM, EEPROM and flash EPROM memory area,

- system control and monitoring unit,
- timer and interrupt controller,
- interfaces with the backplane bus,
- dual-port communication RAM.

The I&C functions are stored as executable programs in the write-protected flash EPROM areas along with the necessary system functions and protected by cyclic redundancy checks (CRC). An executable program is a sequence of executable commands that are executed by the CPU. Program execution on the function processor is cyclic. That means that all components of a function processor are always used in the same way in an immutable and recurring sequence. The physical memory allocation to data and programs is static. That means that data and programs that are required to implement I&C functions are permanently assigned to defined memory areas in advance.

The input signals for I&C functions that are implemented on a function processor are provided either in the form of messages via the communication means or as single-wire signals via I/O modules. Similarly, the results of the I&C functions are passed on either as messages or as single-wire signals. Signals from the I/O module are read by the function processor from the data buffers of the I/O modules via the backplane bus or written to the data buffers. Data exchange with the communication means passes through the dual-port RAMs. Input data for a function processor are directly written from the communication means into the dual-port RAM of the receiving function processor. In the opposite direction, results from the function processor are written into the dual-port RAM of the interface module concerned (part of the communication means) via the backplane bus. Physically different areas of the dual-port RAM are dedicated to receiving and sending. Reading data from and writing data to a dual-port RAM is initiated by the function processor in the course of its cyclic processing. Accesses to the backplane bus are coordinated by the bus arbiter of the subrack.

#### 2.4.3.1.3 Input/Output Modules

I/O modules are process interface modules between the different process signals and the corresponding representations of these signals in the computer. A distinction is made between input and output modules for the direction of conversion and between analog and digital modules for the type of signal. Within these four groups there are different types with differing numbers of channels, conversion speeds or measuring ranges. In TELEPERM XS only non-programmable I/O modules with internal isolation from the process are used. All modules are plug-in PCBs with a connector for the backplane bus and a male multipoint connector for plugging on a front panel connector. The modules contain the following components:

- male multipoint connector for the front panel connector,
- filter for interference suppression and measuring range adaptation,
- optocoupler isolation between input/output signal conditioning part and the internal interface to the backplane bus (isolation voltage • 500 V DC),

- signal buffer stage as the interface with the function processor,
- interface with addressing logic and connector with the backplane bus.

Analog modules also contain an analog-to-digital or digital-to-analog converter and a multiplexer. The method of operation of the I/O modules is always such that signal transmission between the signal buffers and the male multipoint connector toward the I/Os of the module functions autonomously while data transmission between the function processor and the signal buffer of the function processor is performed by direct addressing via the backplane bus. Accesses to I/O modules by the function processor are made by module-specific software drivers that are responsible both for data conversion and for handling fault signals. I/O modules do not have autonomous access to the backplane bus.

#### 2.4.3.1.4 Communication Means

Special communication means are used for efficient data transmission between function processors in different subracks. While in this chapter the communication means are described, the logic procedure to ensure the communication free from interference is documented in section 2.9. Communication is based on serial buses. Data transmission is always performed in the following steps:

- writing the data to be transmitted into the dual-port RAM of the interface module by the function processor,
- serial transmission of the data via the network in accordance with the protocol used to the interface module of the destination system,
- transmission of the data to the dual-port RAM of the destination function processor by the interface module,
- reading the data and checking the data integrity by the destination function processor.

Two protocols are available for serial data transmission, the H1 protocol (IEEE 802.3) and the Profibus L2 protocol (DIN 19245). A special communication processor (SCP1) is used as the interface module for processing the H1 protocol. This module is a plug-in PCB for operation on the backplane bus. The H1 communication processor consists of a processing module (SVE1) (a module identical in hardware to the function processor but with a different software and a different function) and a piggyback module (LAX) containing the actual controller for the Ethernet bus. The firmware implemented on the processing module is mainly used for transmitting data from the dual-port RAM to the bus controllers. Only protocols on layer 2 of the OSI reference model are used.

As with H1 communication, there is also a communication processor for the L2 bus interface. This consists of a processing module (SVE1) with one or two piggyback modules (SL21), each of these piggyback modules containing two independent bus controllers. In this way it is possible to connect four independent L2 buses to one L2 communication processor. The task allocation to the processing module and the piggyback module is the same as on the H1 bus. If several function processors are operated in one subrack, the communication processors can be used by all function processors. In addition to communication via the communication processor,

direct L2 bus communication is also possible if the piggyback modules with the bus controllers are plugged directly into a programmable function processor. In this case, the L2 bus is permanently assigned to this function processor.

In addition other components such as transceivers, star couplers or repeaters are used to implement communication networks. These have no influence on the communication principles, but are necessary to realize the required architectural flexibility.

#### 2.4.3.2 Software Design of TELEPERM XS

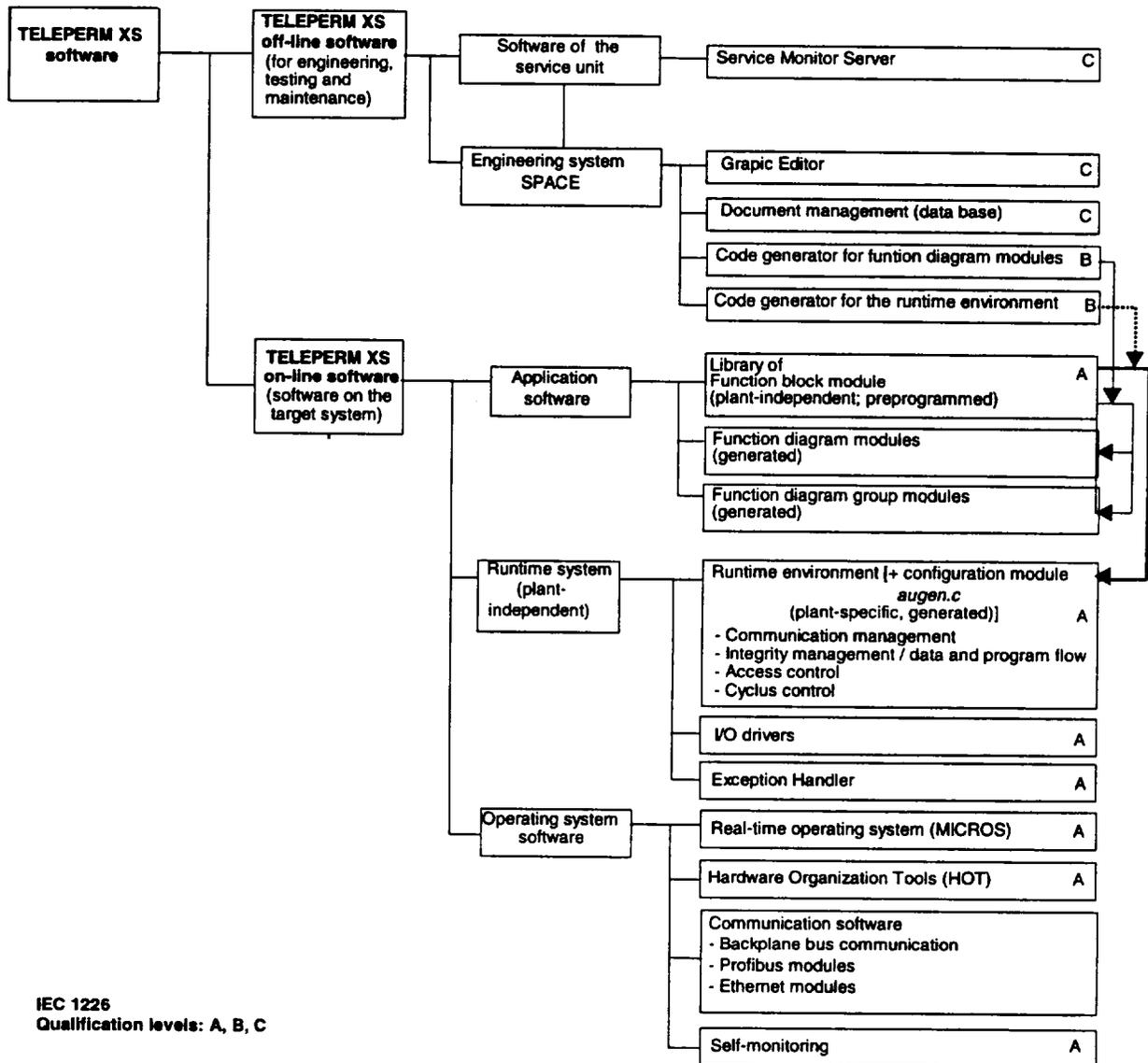
The system behavior of TELEPERM XS is not only determined by the hardware but also to a great extent by the software on the programmable function computers.

The software of the TELEPERM XS system is divided in

- off-line software:  
for engineering, verification, configuration, testing and maintenance. This software is run on the service unit and the computer networks for system monitoring independent of plant operation and does not contribute to the execution of I&C functions, and
- on-line software:  
which is executed on the function processors of the system and directly realizes I&C functions, communication and on-line self-monitoring during plant operation.

The on-line software is subdivided in application software, runtime software and operating system software.

TELEPERM XS: A Digital Reactor Protection System



IEC 1226  
Qualification levels: A, B, C

Figure 2.5 TELEPERM XS Software Architecture

The following approaches were selected for software development:

- The on-line software is structured in type-tested modules.
- The components of the operating system software and the runtime system are run strictly cyclically during normal operation, but in the following cases:
  - the software required for startup,
  - the software handling exceptions, particularly for handling hardware faults,
  - software supporting service functions (service unit, fault location).

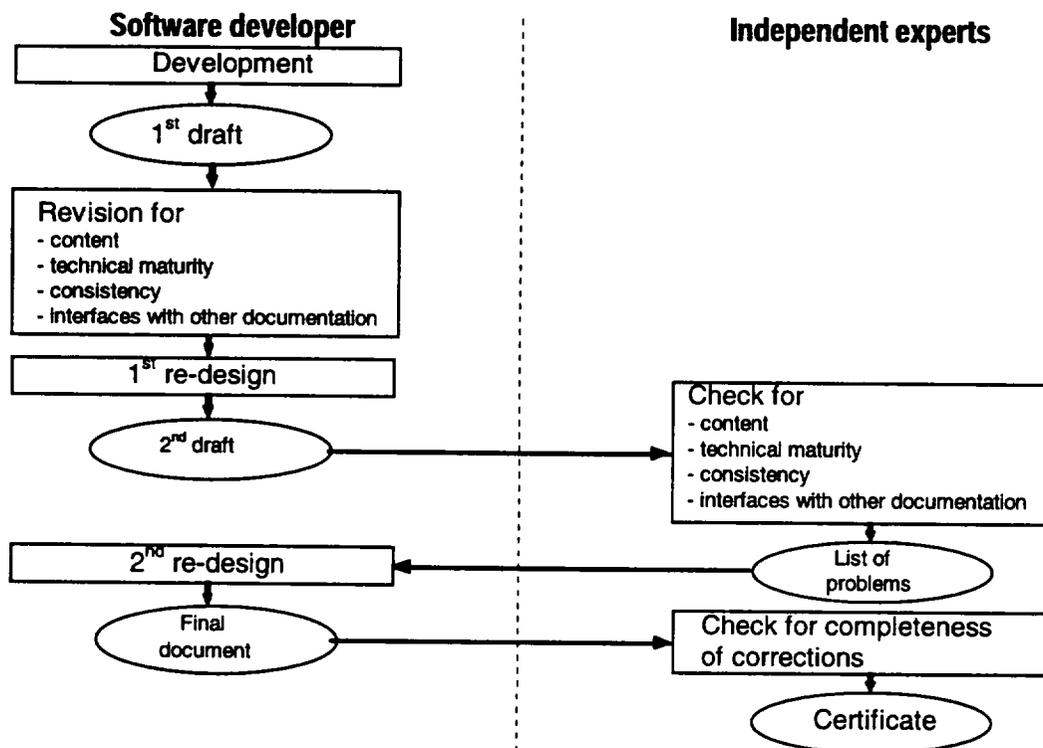
The integrity of the entire software, i.e., the application software, the runtime system and the operating system software is checked cyclically with cyclic redundancy checks (CRC sums) as part of the processor self-monitoring programs. The above software functions that are not processed cyclically can only be activated in a single train (service or on cold restart) or as a result of a single failure (hardware exception).

- There are different reasons to apply different qualification requirements to on-line software and tools:
  - Requirements for tools can be restricted to those Quality aspects that influence the output of the code generator. Other characteristics of safety software like static memory allocation, performance, avoidance of interrupts etc. are less important. As another example, dynamic memory allocation is acceptable for tools but not for on-line software.
- All function blocks which are taken as basic elements for building up the application software have been created according to a qualified phase plan complying with IEC 880 with the participation of GRS-ISTec and TÜV-Nord as independent assessors.

The phase steps for each software module have been:

- requirement specification,
- software specification,
- design documentation,
- implementation and software code,
- test specification,
- test report.

The documentation for each of the above phase documents was drawn up and qualified as shown in Figure 2.6.



**Figure 2.6 TELEPERM XS Development Process**

- During normal system operation, the required system functions are called strictly cyclically. Input data are only handled in the application software, meaning that there is no possibility of interference of input data on the operating system software and the runtime system.
- of interference of the input data on the system software is verified in a generic system test Freedom with the participation of GRS-ISTec and TÜV-Nord.

The application software is developed from task-related specified function diagrams in an automated and qualified generation process. The function diagrams are composed of graphic function blocks that represent an easily understood and separately testable function. The associated software modules are written in compliance with IEC 880, and with the involvement of GRS ISTec and TÜV-Nord. The main criteria for assessing the software quality are:

- The software modules are strictly capsulated, i.e., each module has a defined software interface for input and output signals.
- The software modules can be thoroughly tested because of their small scope.
- The occurrence of inadmissible numerical operations must be reliably prevented. Because of the small scope of the function block modules, implementation of software quality assurance measures to rule out operations that might cause a software exception can be

ensured if double (internal and external) reviews are performed. Examples of inadmissible operations are:

- overflow of the number range in a value-increasing operation (addition, multiplication),
  - overflow of the defined array area of indices,
  - operations with inadmissible operands (e.g., division by zero, square root or logarithm of negative number).
- Irregular jumps from or into the strictly encapsulated software modules can be reliably ruled out, because the data base structure is designed to be deterministic.
  - Normal operation for a processor module is only terminated by the activation of the exception handler as a result of a hardware fault or operator action (pressing of reset button). These causes only apply to a specific processor in a single train and are therefore not a potential cause for failure in more than one train.
  - The automatically generated software is created according to few, very simple and tested rules and has a very simple and fixed structure.
  - The automatically generated software is compiled, linked and loaded with the same compiler, linker and locator for all applications. This facilitates consistent generation of compiled code.
  - All preprogrammed software components such as operating system software or the library of function blocks are present as qualified binary coded components (prelink and libraries) and are generated with the same compiler and linker that are used for compilation and linking of the automatically generated plant-specific software.

The procedure for the generation of the application software from the graphical representation of the specified functions - included the compiler, linker and locator used, is verified on each generation of software. For software generation only an easily handled number of rules is required and implemented. In this way, all connection rules and all function modules are applied several times in the case of an application with a large functional scope. A postulated design error in the software modules or connection rules therefore causes such a large number of consequential failures in the application software that it would most likely be detected in the already performed system test. A diverse feature in this context that supports the very high-quality software generation (in compliance with IEC 880, and with independent assessment performed by GRS-ISTec and TÜV-Nord for the entire TXS software), applying in particular for the function block modules of the application software with the associated connection rules (these modules forming the basic building blocks for the application software and therefore the subject of special attention in the software type test) is the feedback of experience. This means that if an error is nonetheless postulated here, this will be detected during the system tests for the initial application due to the high degree of reuse of the function block modules. In other words, after commissioning tests for the initial application have been completed, it can most likely be ruled out that any software faults remain in the function modules and their connection rules. To give an example of the scale of module reuse, the total number of function block modules applied in reactor control and limitation system for the Unterweser NPP is approximately 31,000 whereas the number of function block module types is <100.

2.4.3.3 Validation of the Application Software

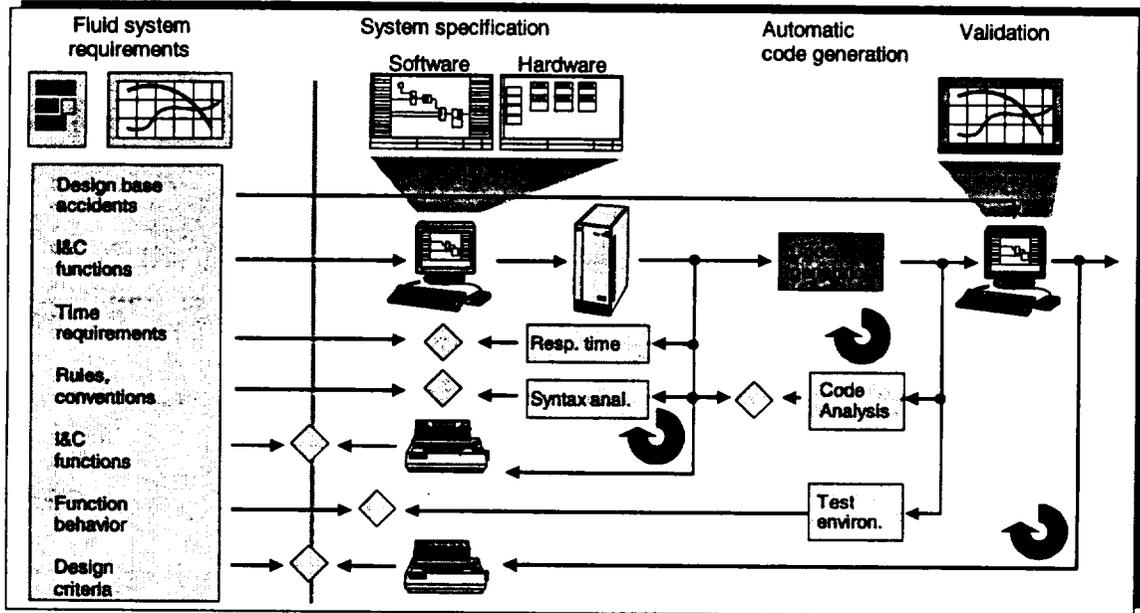


Figure 2.7 Engineering Process for Developing the Application Software

2.4.3.3.1 Verification of the Functional Design

The configured I&C functions are printed out as function diagrams from the data stored in the database after having been input on the SPACE editor. The function diagrams meet the formal requirements of the VGB ("Verband der Großkraftwerksbetreiber," an association of power plant owners). They contain the input measured signals, the signal validation, the processing of the analog values, the limit signal formation, the logic gating and the majority voting for the formation of output signals in an unambiguous form. The representation is understandable both to engineers with an I&C background and to engineers with a process engineering background, and is also used as a review document for checking correct implementation of the fluid system requirements.

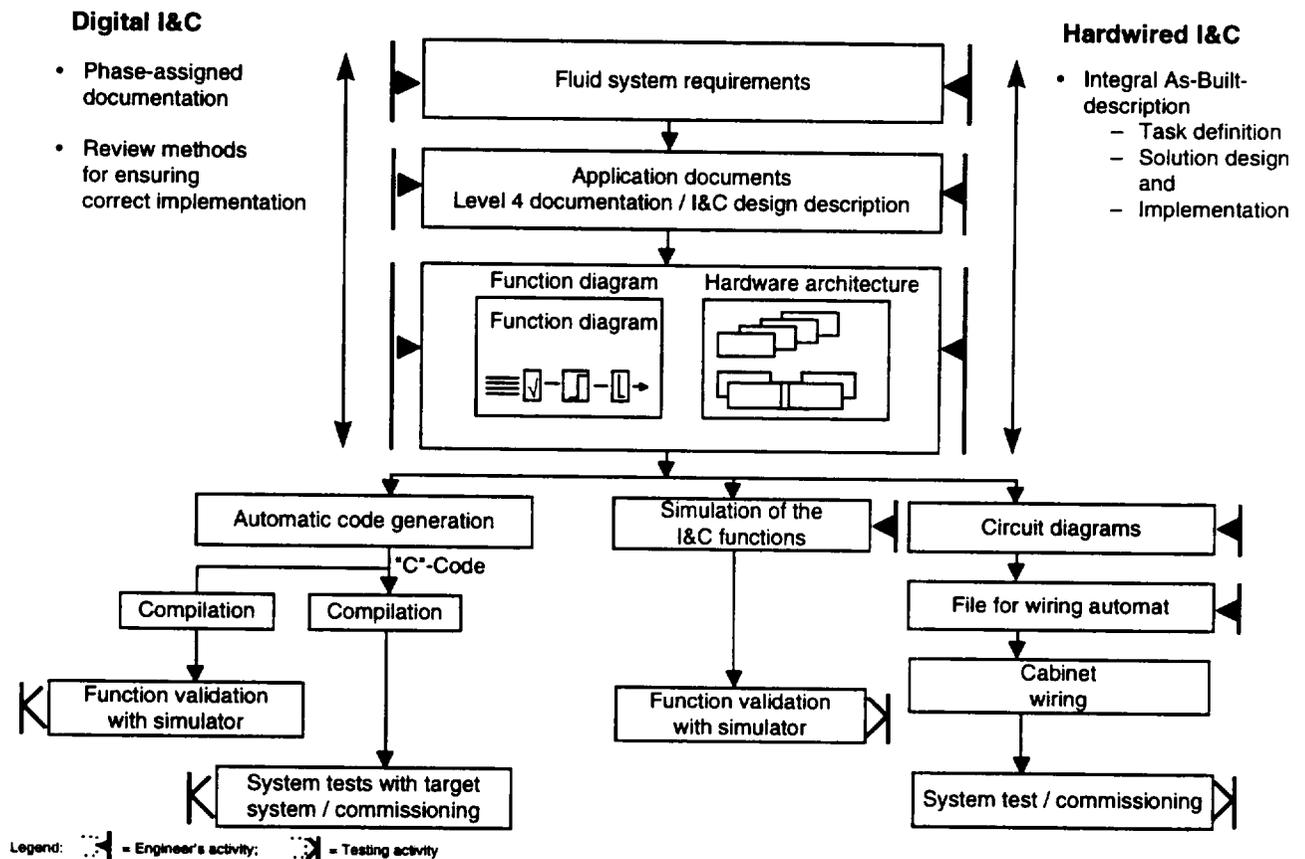
To ensure correct input to the following configuration steps, the SPACE tool is used to calculate the load on CPUs and buses and the response times of the configured system from the available specified data of the I&C functions.

The SPACE tool will only select function modules necessary to implement the prescribed application. The scope of these function modules has been selected on the basis of the experience with hardware modules used to implement hardwired I&C systems with many years of successful service. One important advantage is that these functions can be executed with mathematical precision, and low probability of error or long-term drift.

Deliberate restriction to the above mentioned scope of the function modules means:

- No application of highly sophisticated or tricky algebraic operations.
- The chosen method of representation of the function diagrams ensures that they are just as easy to understand for I&C and process engineers.

If the entire design process from specification to implementation on site is considered, it can be seen that the integrated qualified design process, (thanks to the possibility of direct comparison with the stored code of the application functions during the qualified process steps of code generation and compilation of the original specification data) has considerably reduced the probability of design errors.



**Figure 2.8 Comparison of the Procedure for Designing Hardwired and Digital I&C Systems**

#### 2.4.3.3.2 Simulator-Based Validation

Because a generally applicable code in a high-level language (mostly C and some assembly code for fast functions) is generated from the graphic specification of the I&C functions, this code can also be used to validate the functional correctness of the specified I&C functions using a plant simulator. The technique of simulator validation has been used previously in the development of new functions for analog safety I&C. As for those systems, however, a simulation module had to be created as a precondition for the I&C itself, in addition to validation of the simulation code for the plant process. Direct use of the high-level code from the design process for the safety I&C rules out the possibility of implementation errors. As a result, the quality of the software for application functions is higher than that of analog technology.

#### 2.4.3.3.3 Validation of the Software Generation

As a diverse measure to detect potential software faults not found by the means described in Section 3.2.1, the verification tool "RETRANS" developed by GRS-ISTec is used as an independent testing tool. The generated code can be analyzed by RETRANS to identify the function block modules and reveal the connections between them. The result of this process should yield the information elements contained in the design database as input on the SPACE editor for engineering the I&C functions. A comparison of the result of the validator analysis with the content of the design database for the I&C functions confirms correct application of the tool for code generation and relieves the code generator of exaggerated quality verification demands, particularly in the introductory phase.

The following criteria are very important in determining the quality of the application software generated:

- the validator has been developed by experts who were not involved in the development of the SPACE engineering tool and the function modules,
- the validator analyzes the generated "code" and identifies the function block modules within it as well as the connections between them and reconstructs the content of the database from this,
- comparison of the database contents with the analyzed code.

#### 2.4.3.4 Allocation of Tasks to the Software Components

The software components of TELEPERM XS that determine the system behavior are:

- the operating system,
- the runtime environment,
- the function diagram modules and the function diagram group modules, and
- the function block modules.

The software with invariable data (program code and invariable parameters) is stored in the write-protected flash EPROM area of the function computer and protected by CRC checks. Data that is subject to planned changes (variable parameters) are stored redundantly in the EEPROM area whereas cyclically varying data (signals and status memory) are stored in the RAM area. The integrity of the data which is invariable or infrequently changed is cyclically checked by self-monitoring routines in which the CRC sums and the redundant information are checked.

#### 2.4.3.4.1 Operating System

The operating system is a static multitasking real-time operating system. Static in this context means that all the operating systems resources used are specified on system startup and are not variable during runtime. The kernel of the operating system includes a small scheduler that can manage up to 16 tasks. The scheduler is activated cyclically by a millisecond hardware timer and is responsible for coordinating the task. During normal operation of the TELEPERM XS system, up to three tasks are active: the runtime environment, the service task and the automatic self-monitoring. The runtime environment is the actual platform for the application function. It is activated cyclically by the operating system and calculates the specified application functions after this. The service task is activated by the runtime environment if it identifies a permissible request from the service unit. After the request has been processed the service task deactivates itself. The automatic self-monitoring operates as a continuous background task which cyclically checks the hardware equipment of the function processor for correct functioning.

In addition to the three tasks that are used during normal operation, two other tasks are included in the operating system which are only activated during startup or under fault conditions.

In addition to task coordination, the operating system also provides services for accessing the hardware, setting and resetting hardware timers and for communication between the various function processors. The communication is implemented in such a way that it is always transparent to the application software whether the function processors are plugged into the same subrack or whether they are addressed via a serial bus.

#### 2.4.3.4.2 Runtime Environment

The runtime environment is the most important task of the function processor because it calls the actual application functions. It is activated cyclically by the operating system and then processes the following functions:

- resetting of a hardware timer as a watchdog,
- incrementing the cycle counter,
- reading in the process data from the I/O modules,
- reading the messages from the dual-port RAM,
- transfer of the data to the function diagram group modules,
- processing the function diagram group modules,

- checking the fault messages from processing the function diagram group modules and setting the fault status signals,
- output of the results via I/O modules,
- transmission of messages to other function processors,
- activation of the service tasks, if permissible manual control requests have been identified and
- deactivation of own task until the next cycle.

For master/checker pair or voter computers, the necessary signal exchange and result comparison between the redundant computers is performed by the runtime environment.

The functions processed cyclically by the runtime environment are described below.

At the beginning of the processing cycle, on the one hand, the local cycle counter is incremented and the watchdog timer is set to a value that is larger than the activation cycle for the runtime environments set in the operating system. If the runtime environment does not terminate correctly due to a fault in the signal flow, the watchdog timer times out and generates a hardware interrupt. This interrupt then activates a special interrupt service (exception handler) that saves the state of the computer for subsequent analysis and puts the computer into a defined fault state. In this fault state, all output signals are set to predetermined states and the processor is kept in a waiting loop. The signal outputs are disabled in several different ways by explicit driver calls and by a hardware signal (BASP) via which the load power supply for the I/O modules is disconnected.

The 16-bit cycle counter forms the internal relative short-time base. It is used as the sign of life clock for communication and for time-sequencing of fault signals. The cycle count of the runtime environment at the time of transmission is appended to every message. This information is used by the receiving function processor to monitor the validity of the message and correct functioning of the transmitter.

Data are input and output via the I/O modules directly through driver programs which access the buffers of the I/O modules either writing or reading data. A separate driver program exists for each I/O module and is also responsible for module-specific conversion of the data. Fault alarms that are detected on the I/O module (wire break, overflow, underflow) are used to mark the signals concerned with the signal status "ERROR." Each configured signal in TELEPERM XS contains not only the signal value but also a signal attribute with the status flags "Fault" and "Test." These flags are, as will be explained later, used for fault masking. Missing front connectors or a missing load power supply can result in the enable signal for addressing I/O modules not being formed. The missing enable is detected and signaled by the time-out monitoring. At the same time, the status flags are set to "ERROR" for all signals concerned so that these signals do not have any effect on further function processing.

Messages are read by direct access to the local dual-port RAM. In applications with a high safety relevance, all messages contain additional data for integrity monitoring on the application layer. This includes the cycle count of the runtime environment which has transmitted the message, the message identification number, the message length, and a checksum with which

the integrity of the data from the RAM of the transmitting function processor to the RAM of the receiving function processor is monitored. If the runtime environment detects that the cycle count has not been incremented properly in a message received, this means, on the one hand, that all data contained in the message are too old and must be considered faulted and, on the other hand, that the information must be stored in the cyclically transmitted signaling messages indicating that an upcircuit function processor is no longer functioning properly. An incorrect checksum also indicates that the signals of the message are inconsistent and must be excluded from all further processing. If the data received are up-to-date and consistent they are passed on to the actual application functions (function diagram group modules).

#### 2.4.3.4.3 Function Diagram and Function Diagram Group Modules

The function diagram group modules are activated by the runtime environment in the form of function calls. Function diagram modules and function diagram group modules are generated by code generators from the formal specification (hardware and software specification). All safety functions are specified as function diagrams. Each function diagram is implemented in software by one and no more than one software module, i.e., the function diagram module. Because several function diagram modules are typically implemented on one function processor, all function diagram modules that are to be processed with the same cycle time are grouped together to form function diagram group modules. A function diagram group module therefore consists of a sequence of calls to function diagram modules and copy functions by which signal transfers between the function diagram modules are implemented. A function diagram module consists of a sequence of calls to function block modules that are interconnected by data structures.

Two function diagram group modules can be implemented on one function processor. The processing cycle of the runtime environment corresponds to the cycle of the fastest function diagram group module. Processing of the slower function diagram group module is distributed over several basic cycles, to achieve as constant a load distribution as possible. The allocation of the function diagram group processing to several basic cycles is generated explicitly by the code generator. Function diagram modules and function diagram group modules do not use system services. Only the runtime environment is responsible for supplying data and passing on results by calling the requisite copy functions and outputting communication calls.

#### 2.4.3.4.4 Function Block Modules

Numeric operations on signals are only performed within the function block modules. The function block modules implement basic I&C functions such as "Limit signal generator," "Adder" or "Integrator." They are available in the form of type-tested libraries. Each incarnation of a function block module is statically connected with a data structure that contains not only the input data and the output data but also all internal buffers and parameters. Using these data structures the correct processing of the each calculation can be completely verified. Within the processing cycle all temporary data are preserved and are not overwritten. At the end of the cycle these data can be transmitted to the service unit. This can be done by means of the TRACE-Command of the service unit. On the service unit, the data can be logged and analyzed. As for each functional block, the input data, all the internal memories and the output data are available. The service unit can verify the correct processing of each functional block. This feature is used for debug and diagnostic purposes. All function block modules process the value and status of the input signals. The signal status is an attribute of each signal that indicates its quality (signal under test or signal faulted). For the function block modules a

distinction is made between blocks with active and passive status processing. For passive status processing the output signal of a block is simply formed by OR gating the status information of all input signals. This means that even a single input signal marked as faulted causes the result signal of the function block processing to contain the attribute "ERROR" as well. For function blocks with active status processing, the resulting signal is only calculated from fault-free input signals so that the result signal is also marked as fault-free. This is only possible if the input signals contain mutually redundant information. Active status processing is therefore only possible with function blocks that perform selection and majority voting functions. These include blocks such as for second-highest value selection or for "m-out-of-n" coincidence logic. Function blocks with active status processing are used for masking, i.e., as propagation barriers for faulted signals.

After processing of the function diagram group modules, the results are passed on by the runtime environment either to the I/O modules via drivers or to other function processors in messages. The cyclic signaling messages are also transmitted to the monitoring and service interface and on to the service unit. These messages provide information on the current state of the runtime environment. If faults occur during the processing of the function diagram group modules, all the signals affected are marked with status "FAULT" before being passed on.

After processing of the actual application software, the command messages cyclically transmitted from the monitoring and service interface to the function processor is evaluated. If this command message contains a permissible request from the service unit, the service task that interprets this request is activated and executed. Commands from the service unit include requests to output specific data such as the contents of the fault buffers or parameters, etc., but also requests for changing the operating mode of the runtime environment. All requests for changing the operating mode of a function processor are only executed if a second, independent enable signal is set for the runtime environment of the dedicated function processor. The path via which this second, independent enable signal is issued is configured application-specifically on function diagrams. After a command from the service unit has processed, the command task deactivates itself.

When the runtime environment and the service task are inactive, cyclic self-monitoring is executed as a low-priority background task by the operating system. This task checks the hardware functions of the function processor in a recurring cycle.

The safety I&C system is designed such that the actual safety functions (the application software) does not require more than approximately 50 of the CPU time (the runtime of the runtime environment only amounts to a few milliseconds). This permits the self-monitoring to also make use of approximately 50% of the CPU time. In this case, a complete check of the hardware of a function processor takes approximately 10 minutes.

#### 2.4.3.5 Interference-Free Communication

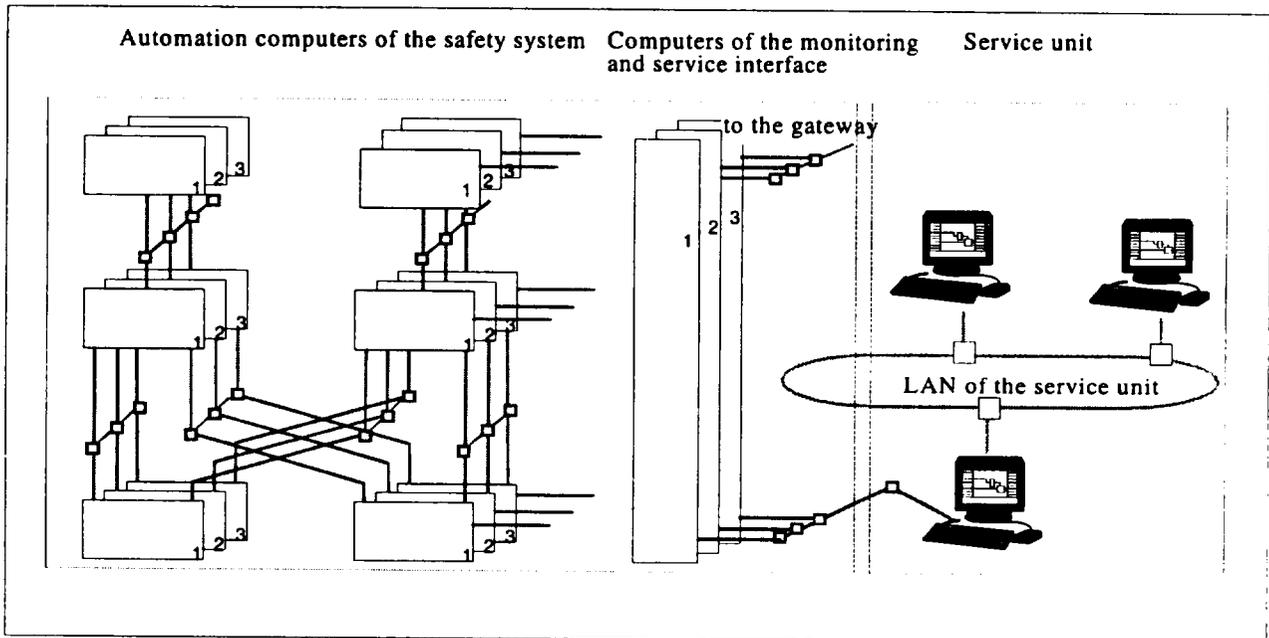
The interference-free communication properties are described in Section 2.4.3.1.4.

### 2.5 *Testability*

For surveillance and self-test features see Sections 2.5.7 and 2.7.1.1. The following chapters describe the scope of the service unit, which is a system to be connected to the "automation computers" for testing the application.

### 2.5.1 Introduction

In addition to the computers of the automatic action path (the so-called “automation computers”), there exists also a monitoring and service interface (MSI) computer in every redundancy. It is connected via LAN to every automation computer in its redundancy. It serves as gateway between the automation computers and other non-safety relevant systems, like the service unit or gateways to process computers.



**Figure 2.9 Typical Structure of a TELEPERM XS System**

On every automation computer and the MSI computers, the application software is embedded in a runtime environment, interfacing the application software with the operating system and the communication services. The application software itself is generated by the TELEPERM XS engineering system SPACE, based on function diagrams and hardware diagrams engineered with this tool.

The TELEPERM XS service unit provides the hardware and software environment which is needed for the commissioning and functional tests of a TELEPERM XS system in the test bay and for periodic test and maintenance on site. This comprises the staff activities concerning

- software loading,
- functional tests and periodic tests,
- monitoring of correct processing,
- fault detection and failure diagnosis, and

- modification of parameters and/or application software.

The corresponding features supporting these activities are described in this section. There are several functions more which are implemented in the service unit but described in other sections, in particular those of the engineering system SPACE, which is employed e.g. for engineering and modifying function diagrams (application software) and hardware diagrams (hardware configuration) as well as for code generation.

## 2.5.2 Tasks of the Service Unit

### 2.5.2.1 Supported Tasks of the Staff

The TELEPERM XS service unit provides the I&C staff with all the functions needed for

- monitoring,
- testing,
- failure diagnosis, and
- modification

of automation computers, monitoring and service interfaces and (partially) gateways of a TELEPERM XS system.

Monitoring means that the I&C staff evaluates the TELEPERM XS system (or one of its components) simply on the basis of the consistency of the observed behavior with the expected behavior of the system (or component), without imposing a specific test situation. This comprises also the display of fault and failure messages produced by the different software components and the self-monitoring functions of the system.

Testing is required in the course of development and qualification of a system. It is also required during operation of the system for safety functions for which it is not possible to demonstrate solely on the basis of the self-monitoring of the system that all faults are discovered spontaneously. This comprises imposing test signals simulating the situation which is to be controlled by the function and comparing the response with the expected one.

When faults or failures are detected, diagnosis results in locating the faulty element (hardware or software) and determining the cause of the failure. This typically comprises:

- a systematic search of the failed element, or
- specific tests capable to reveal the assumed causes of an observed failure

when messages do not directly indicate the cause of a fault respectively the failed component.

### 2.5.2.2 Application Functions of the Service Unit

Typically, a large set of functions may equally be used for monitoring, testing and diagnosis. Therefore, the description is organized according to the functions offered by the service unit and

not according to the tasks of the I&C staff. The link between the set of functions offered and the staff activities is given in a short description "How to use the service unit."

#### 2.5.2.2.1 Displaying State Information and Messages of Function Processors

Monitoring of the state of a TELEPERM XS system requires the possibility to select signals processed by a function processor and to get information about the state of function processors, and to display them on the screen of the service unit or to store this information in computer files. The following is provided:

- Output of the current value of a single or of a group of signals on the screen or to a computer file.
- Selection of a single signal or a group of signals or messages of the I&C system for tracing of their values. This service causes that signals or parameters of a function diagram module in one of the function processors are cyclically sent to the service unit, displayed on the screen and/or written to a data file.
- Reading of the fault buffer of a function processor and displaying the messages on screen. The fault buffer contains fault messages generated by the runtime environment, the operating system or the self-monitoring. The fault messages can as well be acknowledged.
- Provision of a rapid overview of the most important status information, like number of present fault messages, the operating modes of the function processors, the cycle times of the function processors, the current authorizations for changes of operating modes.

#### 2.5.2.2.2 Reading and Setting of Parameters and Hardware-Dependent Data

In addition to reading data from function processors, setting signals, messages and memories is possible as well. This renders possible

- to set the value and the status of signals and messages,
- to set parameters of a function diagram module,
- to read from and write to EEPROMs of a function processor,
- to transfer data between EEPROMs and RAM,
- to read from and write to I/O ports of a function processor.

These functions are completed by functions for the verification of their correct execution (e.g., when a parameter has been modified, the new value is read back to the service unit and compared with the value entered by the maintenance staff).

For best convenience, messages and signals are generally not accessed via hardware addresses but by their names corresponding given by SPACE during the specification of the function diagrams and hardware diagrams. They are typically composed of

- the name of the function processor, function diagram, function block or I/O-module, and

- the name of the message, signal, signal status.

For signals which pass with the same name through several computers, the signal source has to be specified.

However, for detailed diagnosis and more hardware related activities, also direct access to memories and I/O ports of the computing nodes is provided. This requires the knowledge of the memory locations respectively of the hardware of the input/output modules.

#### 2.5.2.2.3 Control of the Operating Mode of Function Processors

In addition to reading (and writing) data and signals from/to a function processor, its operating mode may be controlled. In order to avoid unauthorized interventions, all these functions are subjected to special release functions.

- Change of the operating mode of the runtime environment of a processing node: switch-over between the operating modes: Cyclic Processing, Parameterization, Functional Test and Diagnosis.
- Triggering the restart of a function processor and the initialization of a function diagram module.
- Triggering a given number of cycles of the programs on a function processor.
- Changing of parameters of the runtime environment, e.g., the behavior in case of a restart or the propagation of the signal statuses "ERROR" and "TEST" etc.
- Enabling and disabling software modules of a function processor (e.g., drivers for the input/output modules, communication services, or execution of the function diagram modules). This makes possible to execute programs with externally provided inputs for functional tests or detailed diagnosis.
- Executing programs at specific memory addresses of a function processor. This may typically be used for the execution of hardware test and diagnosis programs being loaded with the service unit or MICROSCOPE (see below).

#### 2.5.2.2.4 Documentation of Maintenance and Tests

For the documentation of the tests and maintenance work performed, the service unit offers additional functions for logging:

- the commands sent to the function processors (change of operating mode, of parameters etc.),
- the changes of the operating mode of function processors during a service session, and
- the results of tests and monitoring.

The data written to the log file has to be specified explicitly in advance by commands to the service unit.

The level of detail of these logs (and of the representation on screen) can also be controlled, e.g., by switching between display of only the labels of messages or full display of the message text.

Stored data may be read, either as input data for future tests, or for further evaluation and analysis, e.g., for

- comparison of test outputs and associated reference data, and
- comparison of test results before and after modifications of parameters or software modules.

#### 2.5.2.2.5 Command Interpreter

For the execution of system tests or of periodic tests of a TELEPERM XS system, a command interpreter for the execution of test programs is provided. The range of the commands also renders possible to execute test programs or to carry out monitoring tasks like counting events or outputting messages or signals when special, previously defined conditions occur in the I&C system.

The command interpreter corresponds to the functionality of current programming languages. It enables

- to access the signals of the I&C system (as listed above) by appropriate commands,
- to define internal variables and alias-names, and
- to execute mathematical operations and to evaluate logical expressions and comprises commands for time-dependent program control: e.g., one may enable the execution of a program during several subsequent cycles, during a defined time interval, or when predefined conditions (described by a logical equation) occur.

The available commands are divided in two sets:

- low-level commands corresponding exactly to the commands directly understood by the runtime environment of the function processors, and
- high-level commands being used to control the program flow on the service unit or even to automate tests. They also provide a more comfortable interface to the low-level commands, e.g., by providing access to signals and variables via a naming scheme, not only via hardware addresses.

This set of commands may not only be used to perform monitoring, test and diagnosis manually, but also to write command scripts. Command scripts are simple text files in which the sequence of commands to be executed for a given task is entered. These files are executed automatically by the command interpreter, after having been initiated by the maintenance operator. They enable automating

- sequences of elementary tests,

- the evaluation of tests, as e.g., the comparison between the response of a tested component and the reference value, and
- modification of a larger set of parameters, simply by initiating the script.

The scripts can be created in a simple way by using a standard text editor.

#### 2.5.2.2.6 Environment for Further Services

The service unit is also designed as hardware and software system for further activities involved in design, maintenance and modification of the safety functions of a TELEPERM XS system. For this purpose:

- The engineering tool SPACE is installed as an integrated part of the service unit and may be used here for all tasks involved in
  - specification of hardware and application software of the function processors,
  - code generation and verification, and
  - load analysis.

Of course, it may also be installed on other hardware systems independent of the service unit. A detailed description of SPACE is given in separate sections under Section 3.1.

- The tool set MICROSCOPE belongs to the operating system software of the function processors (described in an other report). It provides functions for
  - loading of the entire software of a function processor,
  - loading of the software for a single task, and
  - debugging of the software of a function processor (reading, writing of variables and registers; starting/stopping program execution; source-code display for monitoring of program execution).
- A standard communication interface is provided so that future services with online-communication to the function processors may be easy to install and will share the functions already provided by the service unit. This interface is based on a TCP/IP socket interface with a simple opening protocol and a command language. It uses the same commands and syntax as the command language interface provided for the I&C maintenance staff.

#### 2.5.3 User Interfaces for Monitoring, Test and Diagnostic Services

The user interface of the service unit is determined

- by the operating system interface, through which the programs of the service unit can be activated on the one hand, and
- by the individual user interfaces of the different application programs installed on the service unit on the other hand.

The user interfaces of the service unit comprise simple alpha-numerical user interfaces as well as very comfortable, graphical user interfaces.

### 2.5.3.1 Alphanumeric Interface

The service monitor with alphanumeric interface has the simplest but most flexible user interface in form of a simple alphanumeric input/output terminal. This interface renders possible to process low-level and high-level commands and to display the replies on screen. The main field of application of this interface is diagnosis and the execution of system test programs. These two tasks require the entire scope of the command language of the user interface as especially for fault diagnosis and system test the most flexible and comprehensive means are needed. This requires, of course, an expert user using this interface on a regular basis.

### 2.5.3.2 Simple Graphic Interface

This interface is provided for

- periodic tests,
- monitoring of a TELEPERM XS system, e.g., by regularly requesting status information, and
- for use by less experienced users.

It makes it possible to trigger readily prepared command sequences by simply activating a push-button and to display the replies in more elaborated form than with the alphanumeric interface.

This monitor is preferably used for regularly applied operator actions like querying status information, initiating changes of the operating mode of function processors or starting test programs as efficiently and error-free as possible.

The simple graphic user interface is mainly based upon three basic elements. These are:

- push buttons for activation of command sequences,
- input masks for creating commands, and
- selection menus with scroll bars for selection of an action out of a limited number of possible actions.

Answers of the service unit are presented through specific masks or windows for text output. Output masks are used for short outputs with fixed format, whereas windows are used for very comprehensive outputs or outputs with very differing output formats.

As an extension of this type of interface, additional modes of presentation are provided. Especially for monitoring the behavior of the safety functions (function diagram modules in the processing nodes), and for obtaining a rapid overview of the status of a TELEPERM XS system, services are provided which allow to display:

- dynamic function diagrams, and
- dynamic hardware arrangement diagrams.

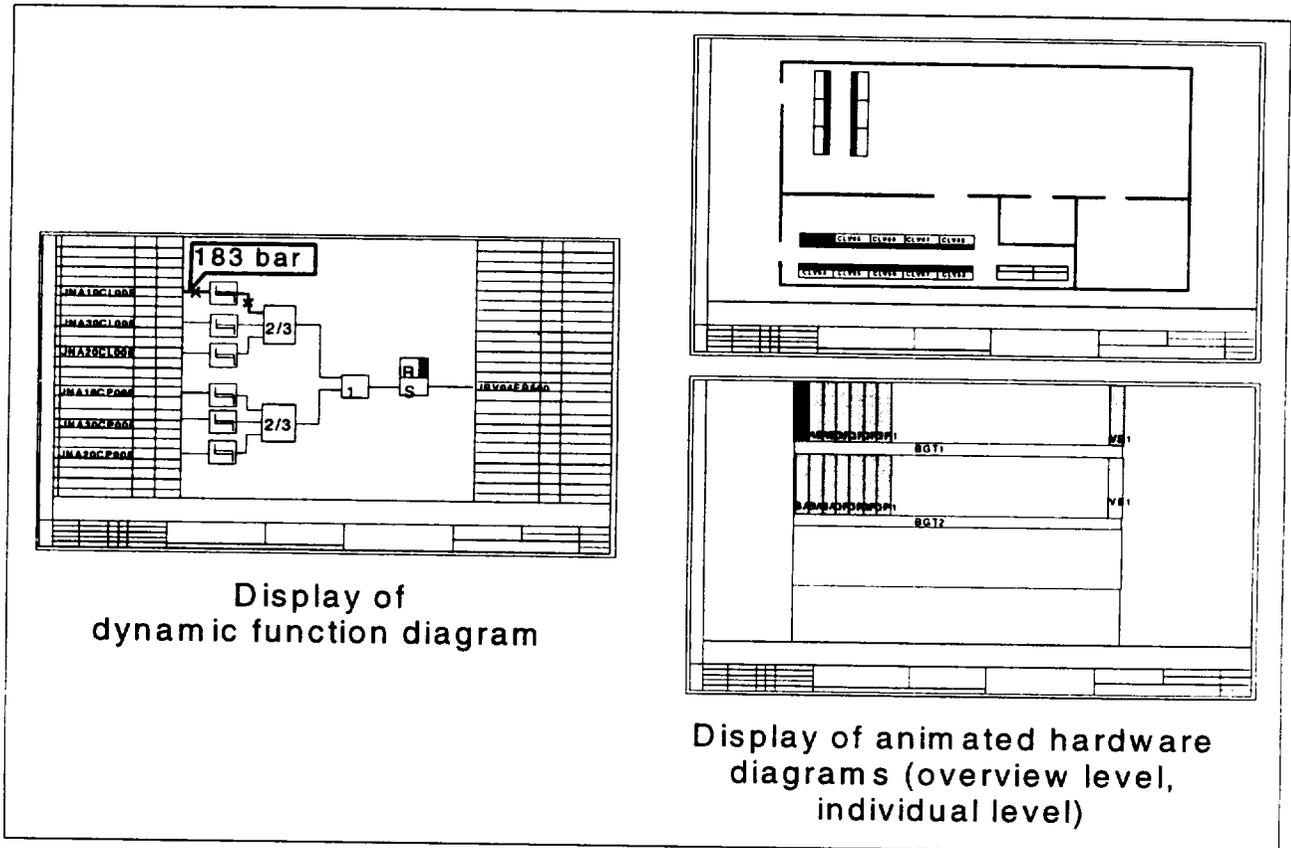
They are outlined in the following two sections.

### 2.5.3.3 Online Display of Function Diagrams

For the display of dynamic function diagrams, the function diagram editor of the SPACE engineering system is used, whereby the functions used for the specification of function diagrams and of the hardware structure are disabled. It displays:

- values of analog and binary signals, and
- status information of analog and binary signals.

The values of the analog signals are shown as digital numbers appended to the signal line. Binary signals (TRUE, FALSE) are displayed through the color of the signal line. The information about the signal status (TEST; ERROR; OK) is given by graphic symbols in the signal line (cross = ERROR; circle = TEST). Several function diagrams may be displayed in parallel, and the user can switch over between the function diagrams by all the navigation functions available in the SPACE editor.



**Figure 2.10 Examples for the Display of Dynamic Function Diagrams and Hardware Diagrams**

2.5.3.4 Online Display of Hardware Structure

Fault messages and status information can also be displayed on hardware diagrams. This may comprise the display of the operating mode of function processors and the presence of fault messages and potentially of their gravity. This is displayed by shape and color coding on the hardware diagram. Detailed information on the fault message can be queried as well as statistical information like operating time since last restart, number of fault messages etc..

This user interface is primarily used to check the status of safety functions, to localize faults and to check the effects of faults on the safety function.

2.5.4 Connection to the I&C System

2.5.4.1 Links to the Function Processors

The service unit consists of equipment which communicates with the function processors through the monitoring and service interface (MSI). Typically the service unit is installed in an

electronic or I&C service room near to the main control room, and connected to the MSI via a SINEC H1 bus.

Gateways are the interface to the process computer or other non-TELEPERM XS systems. Since both gateways and service unit are NON-1E equipment, both devices may be directly connected without interconnection of a monitoring and service interface. Gateways may be therefore operated on the same bus as the service unit.

#### 2.5.4.2 Communication with the I&C System

There are three types of information to be exchanged between the service unit and the other components of a TELEPERM XS system:

- Signaling messages serve for signaling the status of the hardware, software components and faults.
- Data messages serve for signaling the values and status of engineered signals in the function diagram modules.
- Command messages are used to transfer commands from the service unit to a monitoring and service interface and further to a function processor.
- The service unit communicates with the function processors via the monitoring and service interface.

Messages are exclusively sent in a cyclic way. Each function processor SVE1 sends one signaling message and receives one command message per cycle. These messages are transmitted from and to the service unit via the MSI. The MSI organizes the exchange of this information with the function processors.

The number and length of the messages is constant. This ensures constant bus loads and a predictable behavior on the communication busses within the safety I&C system.

Each MSI can service up to up to 10 function processors. They are all accessible via the service unit.

The runtime environment which is installed on the function processors of automation computers, MSI computers and gateways is engineered with the engineering tool SPACE. They use the same runtime environment command interface for exchange of commands and sending replies. Thus signaling and command messages for communication between service unit and MSIs are generated on the same basis as those for the communication within the automation computers. This is simply done by integrating the hardware specification of the MSIs and of the service unit in the overall hardware specification in the SPACE database. Signaling and command messages are then automatically generated later on in the course of the automatic code generation.

## 2.5.5 Architecture of the Service Unit

### 2.5.5.1 Hardware Architecture

The service unit is flexible in concept. Depending on the extent of the I&C system and the amount of the work intended, it can be implemented as a single powerful computer, or as a distributed system with several workstation computers. It is possible to provide only one or several working places. These could be installed temporarily or permanently. Parallel operation of several workplaces is possible. The central services

- access to the function processors, and
- access to the SPACE data base

are realized by servers which are assigned to the service unit computers so as to make optimal use of resources and assure optimal load distribution.

With a distributed structure, the servers can be accessed by all client computers, via local networks. The local network is independent of other networks. It links all work places of the service unit.

The minimal configuration of the service unit is a single powerful computer, on which all programs belonging to the service unit are installed, including the SPACE data base and the communication link to the function processors. In this case, only a network connection to the MSIs is needed.

An important advantage of the distributed architecture of the service unit is that working places may be easily made available in the electronic rooms where the concerned TELEPERM XS cabinets are installed, for tests or diagnosis. This simply requires that the LAN of service unit is accessible in the electronic rooms. By this, input signals could be set on the input module of a TELEPERM XS computer and easily compared with the values read out by the service unit, or output signals could be set by the service unit and directly compared to the output signals measured on the output modules.

The tools for software loading and debugging may be also installed independently of the service unit workstations, simply on portable computers, so as to ease local interventions.

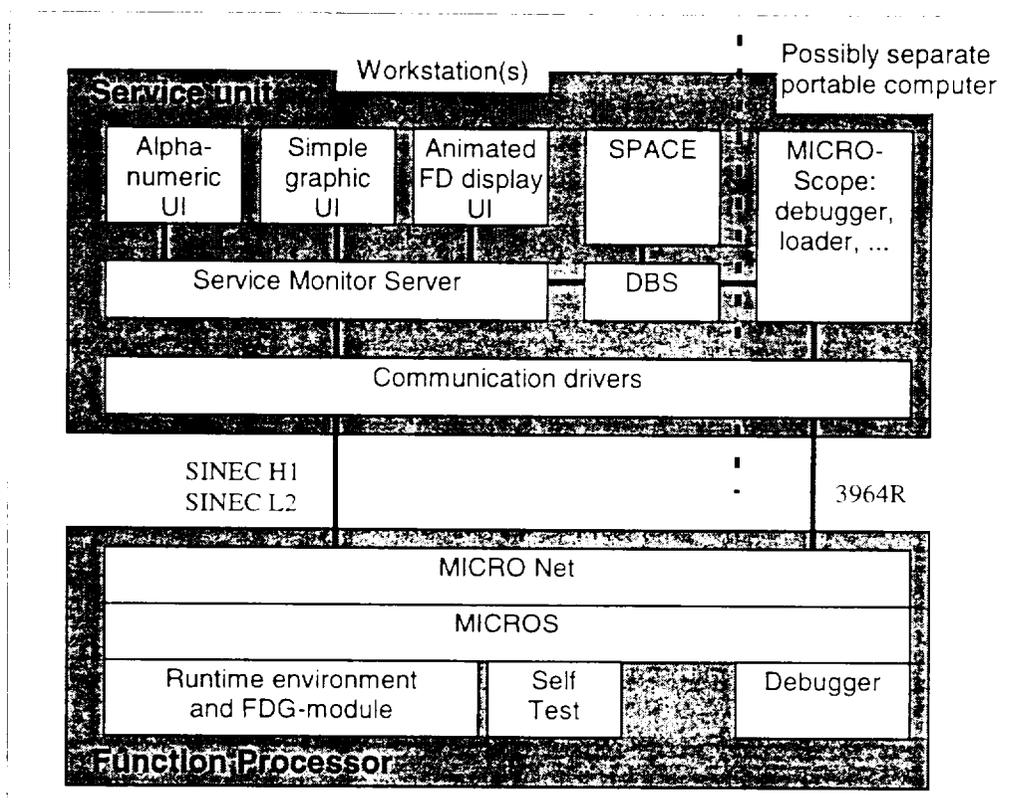
### 2.5.5.2 Software Architecture of the Service Unit

In order to support this distributed architecture, to enhance modular structured software development and to ease the amendment of future functions of the service unit, its software is divided in the following elements:

- The service monitor server (SMS) manages the data communication between the function processors and the different monitoring and diagnosis tools of the service unit. It is always installed on the computer which manages the physical link to the MSI computers.
- The SPACE data base server (DBS) is the central information source for the SMS, containing the function diagrams, the hardware structure of the system, the relation between

names, software module positions in the hardware, the names and addresses of signals and parameters, the user rights etc.

- Several client processes (service monitors) are used to implement the different types of user interfaces of the service unit. These are for the time being the service monitors with alphanumeric interface and with simple graphic interface, and the display of dynamic function diagrams. They may be installed on the same computer as the service monitor server, or on another one.
- A harmonized communication interface is used for the information exchange between the service monitor server and the different clients (service monitors). It is based on a TCP/IP socket interface, comprising a simple protocol for connection set up and clear down; all communications use the same command language, as mentioned above.



**Figure 2.11 Software Architecture of the Service Unit UI = User Interface**

The service monitoring server is the central software component of the service unit. It ensures:

- cyclic acquisition and evaluation of all signaling messages,
- a current mapping of the messages of all automation computers and MSIs,
- recording of all changes to the message mapping,

- execution of high-level commands issued by one of the clients (e.g., commands concerning the program flow of test programs, or execution of logical equations),
- translation of high- or low level commands from client processes into commands to the runtime environment of the concerned function processor,
- retransmission of the replies to the concerned client process, and
- check of the authorization of commands issued by client processes.

It is automatically started after start-up of the service unit whereas the different clients are only started after log-in of a user and explicit request.

#### 2.5.5.3 Technical Data

The computers employed for the service unit should have the following features:

- color monitor with resolution of at least 1280 \* 1024 pixels,
- RAM: 64 Mbytes,
- hard-disk: 1 GByte,
- computing performance: recommended > 50 int SPECmarks 95,
- data back-up with DAT magnetic tape cartridge or optical disk,
- operating system HP-UX 10.20 or NT 4.0, and
- data base INGRES.

The monitor resolution allows to display function diagrams as a whole, and to display at the same time all relevant text entries (at least all editable texts).

A printer with 11x17 sized paper is recommended for printing function diagrams. In more spacious plants the printer should be available near to the service unit working place(s), in smaller plants the printer may be within reach in greater distance via network or transport of magnetic or optical data medium. Otherwise a printer with letter sized paper may be used, if function diagrams are to be quickly printed as working paper. A hard-disk of more than 1 GBYTE should be available for storage of data.

#### 2.5.6 Handling and Use of the Service Unit

##### 2.5.6.1 Logging in and Tool Selection

The user logs in to the service unit via the "log-in" procedure of the operating system, which is at the same time used for user identification and check of the authorization. All tools of the service unit can be activated on the user interface of the operating system.

#### 2.5.6.2 Monitoring of the I&C System

As a matter of principle, all signaling messages cyclically transferred by the service unit are recorded, checked for changes and archived by the service monitor. The current status of the I&C system and the respective statistic information can be made available via the different user interfaces.

#### 2.5.6.3 Conduct of Periodic Testing

Periodic testing of the distributed processing systems is conducted with the aid of application-specific test programs. Test programs consist of a sequence of commands to the service monitor server. Hereby commands of the high-level command language are preferred. Using the service monitor server with graphic user interface for testing comprises the following steps:

- requesting test release for the redundancy of the respective function processor from the control room (release with key switch),
- logging of the respective function processor to the service unit,
- requesting operating mode "test" for the respective processor module from the service unit (also release with key switch),
- selecting and activating test programs,
- checking and printing the test log,
- setting the operating mode of the respective function processor to "CYCLIC PROCESSING," and
- log-off of the respective function processor from the service unit.

A completely automatic repeated testing by use of application-specific client programs is possible as well.

#### 2.5.6.4 Elaboration of Test Programs

Test programs consist of text files comprising a sequence of valid commands to the service monitor server. These commands are operator requests to the runtime environment and initiate selective reactions of the I&C systems, which permit inferences on the real behavior of the system. Test programs are written as command scripts with simple text editors, tested by the service monitor server with alphanumeric interface and afterwards integrated as test program in the service monitor server with graphic interface.

#### 2.5.6.5 Log-Off

The user logs off from the service unit via the "log-off" procedure of the operating system, which at the same time closes all client programs. In contrast, server programs remain active and still monitor and log all status information of the I&C system.

### 2.5.7 Testing and Maintenance

For periodic testing and for diagnostic activities, individual function computers can be put into a special testing and diagnostic mode via the service unit. The function processor that is being tested then behaves like a computer with a "detected fault" for the system. After transition to the mode "FUNCTIONAL TEST," the signal outputs via the I/O modules are disabled and those sent via the communication means are marked with the status "TEST" as test signals. However, output of specific signals can be initiated as part of the test. To test the communication means, for example, messages with any content required can be sent. All messages sent will have the status attribute "TEST."

If the receiving function computer is in cyclic operating mode, all signals received with the status "TEST" are converted to the signals with status "ERROR" and therefore masked by selection blocks with active status processing. In this case the receiving function processor behaves as if the transmitting function processor had failed.

If the receiving function processor is itself being tested, the receiving function processor processes test signals without changing the status information. This permits integral testing covering several computers.

To test the outputs to the peripherals, it is possible to set and reset individual output signals explicitly. The test interlocks required to do this are stored in procedures in the service unit. Tests on I/O modules or live testing of components can also be specified as function diagrams and implemented in the same way as a safety function on the function processor. In this case, the test can be controlled via single-wire signals from the control room without the service unit.

## 2.6 **Control of System Access**

This section only covers the system access via the service unit. The features described here will be quoted later application specifically. The system access, e.g., concerning access to cabinets, rooms, etc. is not covered in this topical report.

### 2.6.1 Objectives

The service unit contains the central data of the I&C system. It is also the central means for interventions into the safety relevant software of the function processors. That is why the service unit is protected against non-authorized interventions.

The installed control mechanisms assure that:

- only authorized persons may access the service unit,
- that only the authorized interventions may be performed, and
- that at a given time interventions are restricted to a single redundancy (exceptions might be possible during plant outage).

### 2.6.2 Technical Measures

The safety I&C, the service unit, and the associated LAN are typically installed in the security area of the nuclear power plant. Thus all measures on site controlling the access to this area apply by default also to the service unit.

Additionally, access to the functions of the service unit are subjected to further safety mechanisms. These primarily serve for the identification of authorized persons and furthermore for the assignment of the agreed user rights ("privileges").

The admissibility of the access to and the use of the service unit is controlled in two ways:

- Within the service unit, the user rights are fixed on two levels: the level of the operating system and the level of the application software. On the level of the operating system, it is checked whether a user is allowed to use any services of the service unit at all and if yes which services these are.
- The user is identified in the course of "log-in" by his/her name and an associated password, whereby together with this identification user rights are assigned. In addition, it is checked within the application programs of the service unit which service operations may be executed by the specific user, whereby the programs refer to the identification of the user by the operating system.
- Authorized operator actions are monitored by the central server of the service unit.
- Independently of the control of the rights to use the service unit, commands sent from the service unit to the function processors are only executed if the function processors are in an appropriate operating mode.
- The change-over to an operating mode different from "CYCLIC PROCESSING" is only possible if an additional release signal which is independent of the service unit is present. This release signal has to be set in accordance with the main control room staff. It is given by key switches and transmitted via single wires to the monitoring and service interface computer of each redundant initiation chain. Only one redundant initiation chain is permitted not to be in "CYCLIC PROCESSING" mode at a time (exceptions might be possible during plant outage). As soon as the release signal is reset, the processing modules in the respective redundant chain are restarted and take over "CYCLIC PROCESSING" mode.

If especially high security requirements are to be met, additionally possession-based access control and monitoring of user rights may be also employed on the service unit. In this case, the user may additionally be identified by a chip card in combination with the password.

### 2.6.3 Additional Protection Against Errors

The service unit also supports multi-user operation. In order to avoid conflicting orders sent by two users simultaneously to a function processor, a function processor may be reserved (locked) by a user, thus inhibiting commands sent by another user to the same function processor.

In order to reduce the consequences of erroneous commands, a user may voluntarily restrict his privileges temporarily to lower ones, which correspond just to his current task.

#### 2.6.4 User Rights and Operating Modes

There are graded rights for the use of

- off-line functions (use of the engineering system SPACE), and
- on-line functions (access to function processors)

which are checked in the course of the log-in procedure of the service unit. For the use of the on-line functions, there are six graded privileges. A privilege includes all lower privileges.

Commands sent from the service unit to a function processor are only executed if it is in an operating mode in which the command is permitted. A rough summary of the graded privileges for on-line services, of the service actions they permit, and of the required operating mode of the concerned function processor is given in Figure 2.12.

<b>PRIVILEGE</b>	<b>SCOPE OF POSSIBLE SERVICES</b>	<b>REQUIRES AT LEAST OPERATING MODE...</b>
(1)READ BUF	reading function processor buffers [lowest privilege]	CYCLIC PROCESSING
(2)ACK BUF	acknowledging buffers	CYCLIC PROCESSING
(3)TRACE	tracing and reading function diagram signals	PARAMETERIZATION (or CYCLIC PROCESSING, if allowed by specific RTE parameter)
(4)PARAM	modifying parameters of function diagram modules or of the runtime environment	PARAMETERIZATION
(5)TEST	writing signals in function diagrams or I/O modules; controlling the execution of the function diagram modules and I/O	FUNCTIONAL TEST
(6)DIAG	loading software; executing diagnostic programs [highest privilege]	DIAGNOSIS

Figure 2.12 Table of Privileges

## 2.7 **Fault Tolerance Features**

### 2.7.1 Types of Component Failures and Their Effects

Not only the component failure rate is decisive for a probabilistic analysis of safety I&C systems but also whether and how component failures or the resulting faults affect the safety functions. For the purposes of a probabilistic analysis, the main criteria by which failures are classified are the following:

- the way in which a safety function is affected,
- the time that elapses before the failure is detected and the fault is repaired (time until fault clearance), and
- the confinement area for the effect of the failure.

Safety I&C systems functioning automatically act on the plant process via commands to the switchgear. Any potential component failure in the safety I&C can be classified in one of the following groups in terms of their impact on this interface:

- 1a) component failures that can prevent issue of the necessary commands to the switchgear in the event of a demand,
- 1b) component failures that can cause spontaneous issue of spurious commands to the switchgear, or
- 1c) component failures that have no impact on commands to the switchgear.

Because computer-based I&C systems work in timesharing mode, it is on principle possible for one and the same component failure to prevent command issue or cause spurious command issue, with this depending on the time of occurrence. Component failures that have no effect on the interface to the switchgear mainly comprise failures in the equipment for monitoring and signaling and in the equipment installed to extend the service life of components.

Classification on the basis of the time until fault detection (detection time) is matched to the measures taken to detect failures. Systems which function dynamically - this including the TELEPERM XS system - have the characteristic that nearly all failures with a potential for influencing the safety function have effects which can therefore be detected immediately. This characteristic is a consequence of the simple fact that nearly all failures of components cause static conditions in the components, with the result that dynamic signals can then no longer be correctly processed. All systems that have defined failure behavior are based on this fact. Independently of this, TELEPERM XS has both hardware and software monitoring features that detect nearly all failures immediately. Failures that are neither detected by dynamic use nor by self-monitoring remain hidden until the periodical test. As a result of the system design, these mostly comprise failures in the monitoring and signaling equipment and some elements of the I/O modules. A distinction can be made between the time until component failures are detected as follows:

- 2a) failures that are detected immediately, or

2b) failures that are detected by periodic testing.

Because the safety functions are actually run through during periodic testing, any failures that exist but are not detected during periodic testing will not have any impact on the safety functions.

Classification on the basis of the confinement areas for component failures requires that the relevant system characteristics be considered. The use of multi-channel I/O modules means that the minimum confinement area for a failure comprises the failure of a single channel of an I/O module. The next largest confinement area is the entire module. Failures on modules that are operated in timesharing mode generally affect the entire module. The next largest confinement area comprises the failure of the entire subrack. One example of this is the failure of the DC/DC converter for generating the module-specific supply voltages. The failure of a bus segment is equivalent to the failure of a subrack. This can be caused by failures of transceivers or a break in the bus cable.

Failures that go beyond the confinement area of a subrack are only possible as a result of common cause influences. These will be dealt with separately at a later point in this report. The four confinement areas for failures are therefore:

- 3a) single channel of a multi-channel module,
- 3b) complete module,
- 3c) complete subrack, and
- 3d) complete bus segment.

In conjunction with the three types of faults (1a-1c) and the two different detection times (2a, 2b), this yields  $3 \times 2 \times 4 = 24$  different categories by which the effects of failures can be described.

By analyzing the mechanisms for detecting, signaling and masking failures it is now possible to determine for each basic component for TELEPERM XS the resulting effects possible on the component level and above this on the system level. In conjunction with a quality analysis it is then possible to determine how probable or how frequent such effects are.

In TELEPERM XS there are two ways of detecting failures:

- failure detection by mechanisms inherent in the system, and
- detection by a configured monitoring function.

The inherent mechanisms make use of special monitoring equipment to identify deviations from the expected system behavior. This is implemented independently of the specific application. In contrast to this, in configured failure monitoring, application-specific redundant information is generated to detect deviations by comparing this information and therefore to detect failures both in the equipment of the TELEPERM XS system and in the peripheral equipment for acquiring the plant conditions. The mechanisms inherent in the system for failure monitoring will now be described for the basic components of TELEPERM XS. The type of failures that are

possible, the way that failures are detected and the confinement areas for their effects are derived from this.

### 2.7.1.1 Inherent Mechanisms for Detecting and Signaling Failures

#### 2.7.1.1.1 Subracks

The supply voltages, the fan speed and the ambient temperature are monitored in the subracks. This function is performed in a special monitoring module in the subrack. Depending on the type of failure either only a signal to the cabinet alarm unit is output, e.g., temperature monitoring responded or a signal is sent to the backplane bus at the same time by which all function processors in the subrack are be put into a defined condition. This condition is characterized by the fact that the cause of failure and other status information are stored in the function processors which then become inactive with respect to external equipment.

Independently of the signal to the function processors the voltage supply for all PCBs is switched off after a delay.

The time for which the backplane bus is assigned to the function processors is monitored by one of the function processors itself (timer monitoring) and independently by the bus arbiters as well. If the bus arbiter detects inadmissible accesses, the module concerned is excluded by a hardware signal. Faults in the bus arbitration are detected by the function processors during normal operation.

Failures of the main components of the subrack either cause the function processors to change to the defined fault state, if they have the potential for effecting the safety function, or they are only signaled, if this is not the case. On the basis of task allocation and the monitoring mechanisms, the following types of failure result for the subrack:

- Failures that concern the subrack-internal power supply or the control of the backplane bus cause transition to the pre-defined fault conditions on the function computers (the subrack then behaves passively toward external equipment).
- Failures in the temperature monitoring or fan failures only cause alarm output and have no effect on the safety functions.
- Failures in the monitoring equipment can remain concealed.

All failures affect the entire subrack.

#### 2.7.1.1.2 Function Processors

The function processor is designed and used such that there are two independent detection mechanisms with broadly overlapping monitoring scopes for the detection of failures with the potential for affecting the safety functions. The most effective and simplest mechanism is the strictly cyclic use of all hardware components. Because of timesharing operation, failures of central components such as the processor itself but also the bus interface are detected spontaneously, to its large confinement area as it affects several I&C functions. Components that are not operated in timesharing mode such as the memory or the address decoder have smaller confinement areas for failures, so that concealed failures cannot be ruled out completely

without additional monitoring mechanisms. Coverage here is provided by the self-monitoring programs that check the functions used by TELEPERM XS cyclically.

Failures are detected not only thanks to cyclic operation but also by hardware monitoring equipment and self-monitoring programs. The watchdog and the system support controller are used for hardware failure monitoring. The watchdog monitors the program control and the system support controller monitors access times. These times comprise:

- access times and time-out on accesses to the I/O bus and the K32 backplane bus, Figure 2.20, "Interference-free H1 Communication"
- the waiting time for allocation of the K32 backplane bus.

The following are also monitored by the cyclic self-monitoring function:

- the integrity of the invariable data in the flash EPROM and in the EEPROM by the use of cyclic redundancy checks,
- the function of the read and write memory,
- the function of the processor and the coprocessor,
- the function of timers, interrupts and the watchdogs,
- the function of the ports, and
- the correct setting of the jumpers.

The monitoring mechanisms described above ensure that all failures with a potential for influencing the safety functions are detected. Depending on the monitoring mechanism concerned, detection is either spontaneous or delayed, with the maximum delay time being the cycle time of the self-monitoring function, i.e. approximately 10 minutes. Depending on their significance, detected failures are either only signaled (if no direct effect on the safety functions is possible, e.g., failure of an item of monitoring equipment), signaled and masked (if the component failure can be clearly restricted to a certain confinement area, e.g., failure of a dual-port RAM) or the computer is put into a pre-defined condition. This is accomplished through a combination of vendor and application programming.

Detected failures always cause the signals concerned to be excluded from further processing. It is not possible to rule out generation of spurious signals in the period between the occurrence of a component failure and detection of the component failure. The probability of a failure of this type, however, is extremely low and can be estimated quantitatively. Appropriate design (e.g., master-checker configuration) permits the detection times for failures to be reduced to the extent that spurious signals are ruled out. This means that the following types of failures are possible on the function processors:

- The safety function can no longer be performed.
- Spurious signals are generated for a short time.

- There is no effect on the safety functions.

TELEPERM XS experience has demonstrated that nearly all failures are detected immediately by either hardware features or checks of the runtime environment. The typical confinement area of effect for a failure is the module. Only a few component failures can affect the entire subrack, these comprising component failures located within the area of the backplane bus.

#### 2.7.1.1.3 I/O Modules

Monitoring equipment on I/O modules mainly aim at external circuitry such as wire-break, connector or measuring range monitoring so that failure on the module itself is only partly detected by system-inherent equipment. Failures in the interface with the backplane bus are detected and signaled spontaneously by the cyclic accesses of function processors to the signal buffers. The majority of the failures concern signal input or output channels. If multiplexer or converter components have failed, the entire module is affected. In some rare cases, failures in the interface with the backplane bus can also affect the entire backplane bus. Failures of the signaling equipment (e.g., LEDs) do not have any effect on the safety function.

By use of configured monitoring functions hidden failures which affect the safety function can also be ruled out to the greatest possible extent in I/O modules.

#### 2.7.1.1.4 Communication Means

The communication means has three main independent mechanisms for detecting failures. These are the error detection mechanisms that are already defined in the LAN protocol, the monitoring mechanisms in the communication processors and the cyclic redundancy checks on the application layer.

Both the H1 bus and the Profibus have two error detection mechanisms on the protocol level with a hamming distance of 4 and various monitoring mechanisms. The specification of the protocols is defined in IEEE 802.3 for the H1 bus and in DIN 19245 for the Profibus. The protocol error detection mechanisms ensure the integrity of data transmission from the transmitting bus controller to the receiving bus controller. Inconsistent data are detected and rejected by the receiving controller. The use of level 2 communication means that rejected messages are not repeated so that the loss of messages is detected on the application layer. All missing messages are masked by the runtime environment in TELEPERM XS so that spurious signals due to missing messages can be ruled out. Because of the cyclic operation of TELEPERM XS, loss of single messages is tolerated on the application layer. Since messages can be lost in loosely connected computer systems because of drift phenomena or bit error rates in individual transmission elements, the runtime system of TELEPERM XS has been designed so that single missing messages are tolerated by the system without a fault signal. Only several consecutive missing messages are signaled as a fault.

The monitoring equipment of the communication processors detects and signals failures in their own hardware. Because the hardware of communication processors consists of processing module SVE1 with a piggyback bus interface module, the self-monitoring mechanisms of the function processor are valid as well, as both modules are identical in hardware. These have already been dealt with in detail in a separate section.

The cyclic redundancy checks implemented in the runtime environment coupled with sign of life monitoring by the cyclic counter is an effective mechanism for detecting failures in communication means, this providing coverage of all components involved in communication. This protection feature coupled with cyclic communication ensures that all failures in communication means that can affect the transmission of data and therefore have a potential influence on the safety function are detected immediately. Hidden failures are therefore restricted to monitoring and signaling equipment. Any failure which is detected is masked in the system so that spurious signals will be ruled out.

A distinction is made between the confinement area for:

- component failures that are restricted to the bus interface,
- component failures that concern the entire module,
- component failures that concern the entire subrack and failures that concern an entire bus segment.

Confinement of the effects to one channel only concerns the L2 bus interface because this module has two independent bus controllers. All other items of communication means are each assigned to a single bus.

It is typical for failures in communication means to affect those links that serve failed equipment, e.g., the entire module. For items of equipment that are plugged into the subrack as modules, failures in the interface with a backplane bus can affect the entire subrack. Modules that function directly on the serial bus can cause the entire bus segment to fail.

#### 2.7.1.2 Configured Monitoring Mechanisms

In addition to the monitoring mechanisms inherent in the system there are also configured mechanisms whose aim it is to reduce the number of concealed failures. The configured monitoring mechanisms always make use of redundant information processing with down-circuit majority voting. By comparing redundant information, deviations can be detected which indicate the presence of a failure.

##### 2.7.1.2.1 Redundant Measured Value Processing

In nuclear power plants, safety-related measured values are always acquired and processed redundantly. A "true" signal is formed from the redundant signals by appropriate voting and the required measures are then derived from this. The "true" signal is formed by voting features (e.g., second-highest value for analog signals or 2-out-of-4 coincidence logic for binary signals). The monitoring of consistency between the redundant signals is also performed within the voting logic. With redundant measured value processing, the coverage factor for failures on analog input modules can be increased to the extent that all failures that do not cause "freezing" of signals are detected.

##### 2.7.1.2.2 Configured Monitoring of I/O Modules

For continuous monitoring of I/O modules additional monitoring functions specified in the function diagrams can be configured. This is usually implemented by applying test signals to a

channel of the module continuously and monitoring that these signals are correctly processed. This method permits all components of the I/O modules that are common to signal channels to be monitored. Whether and at what point such configured monitoring functions are used is governed by the importance to safety of the module and by its potential for hidden component failures. However, compared with redundant measured value processing this method only increases the coverage factor for component failures by a small amount.

For output modules that control important items of equipment, the coverage factor for failures can be increased by reading back and comparing the output signals. This is a particular advantage in the event of items of equipment that are operated frequently and via redundant paths, as this permits early detection of component failures before they can have an effect on the safety functions.

#### 2.7.1.2.3 Master-Checker Configuration

If outputs to the switchgear must be "fail-safe", i.e., if spurious signals even within a short period must be ruled out, the detection time of failures on the function processors must be reduced by the use of master-checker configurations. Master-checker configurations process the safety functions in parallel in cyclic synchronism and compare the results before they are output. Differing results cause both the master and the checker to prevent output of the results by diverse methods and both change to the defined fault state. Whereas the checker sets all outputs to "0 signal" explicitly by driver calls, both use hardware means to inhibit the output modules by disconnecting the load power supply. The use of master-checker configurations serves to ensure that failures on function processors modules will not result in the issue of spurious signals even within a short period.

#### 2.7.1.2.4 Voter Configuration

If fail-safe and fault-tolerant outputs to the switchgear are required, a voter configuration is used. The voter configuration consists of two independent pairs of masters/checkers each in a separate module subrack with a separate power supply. This configuration ensures that random signal failures only affect one half of a voter. Because the output signals of the two halves of the voter are OR-gated in a hardware "OR element," the second half of the voter assumes control of the switchgear in the event of such a failure. Both the master-checker pairs in the two halves of the voter and the interaction of the halves of the voter function in synchronous cycles. The voter configuration ensures that single failures can result neither in issue of spurious signals nor in loss of function.

#### 2.7.1.3 Masking of Component Failures

In TELEPERM XS all failures detected are masked. Depending on where the failure is detected this can be performed on three different levels, on the level of the individual signal, on the system level or in the switchgear.

On the level of the individual signal, failures are masked if it can be unambiguously determined which signals are affected by the failure. This applies to all failures in communication means with a potential for affecting the safety function and nearly all failures in input modules. Failures in the communication means are detected by the runtime environment of the receiving computer. The signals affected are marked as faulted and masked by the next selection block with active status processing. Failures of input modules either result in appropriate marking as

faulted by the module drivers if the failure is detected by the system-inherent mechanisms or the failures are detected and masked via redundant measurement signals and the configured monitoring in the selection blocks.

On the system level, failures are masked whose effect cannot be confined to single signals. This mainly concerns failures in the subrack and on the function processors with a potential for influencing the safety function. In the event of such failures the function processor changes to the defined fault state in which the function processor behaves passively toward external equipment by disabling all signal outputs via hardware interrupts. The computer which is down-circuit in the signal processing detects and masks the failure again on the signal level in exactly the same way as if it were a failure in the communication means. The precondition for this is that redundant signals be available for masking in the down-circuit computer. If this is not the case, each signal adopts the configured preferred state.

If a component failure occurs that cannot be confined to single signals in the last function processor in a processing train, i.e. in the computer that outputs the control commands to the switchgear, masking on the system level is only possible by a voter configuration. In this case the intact half of the voter masks the failed half of the voter. Concealed failures of output modules are also masked by the voter configuration. Masking of component failures in the system is essential in all cases where there is no selection logic in the switchgear and it is required that single component failures be accommodated. If no voter configuration is used and it is required that single component failures be accommodated, the component failures must be masked in the switchgear. The preferred method of doing this is to configure the interface with the switchgear in such a way that the defined fault behavior of the function processor is the safe state for the plant.

#### 2.7.1.4 Fail-Safe Failure Behavior

For TELEPERM XS, an extremely high detection rate for system-internal faults is accomplished by means of

- self-monitoring of the function processors,
- CRC checksum monitoring of programs,
- Watch Dog,
- message monitoring by cyclically update checks and CRC checksum checks, as well as
- monitoring of the voters for synchronism and train monitoring

makes it possible to engineer the status of the command output of the affected subsystem (redundant train on the processing level or in the voter system) can be engineered to be in accordance with the fluid system requirements in case of detected faults:

- Fail-safe behavior of the affected system of a redundant train in the case of faults independent of the actual input signal (self-monitoring of the function processors, watch dog initiation, discrepancy of master and checker in the voter) for:
  - a definite safety-related command, as e. g. for the initiation of a reactor trip,

- commands for the initiation of measures, which have no negative consequences regarding safety in the case of normal power operation as the initiation of the safety power supply.
- Passive behavior in the case of faults.

The initiation of a control rod insertion caused by a single fault in the limitation system during a power operation undisturbed until then is regarded as disadvantageous concerning safety in contrast those mentioned above. In the case of a failure of a voter the concerned control rods are blocked for the commands "drive rods" and "insert rods" (not reactor trip) until repair is completed.

#### 2.7.1.5 Summary

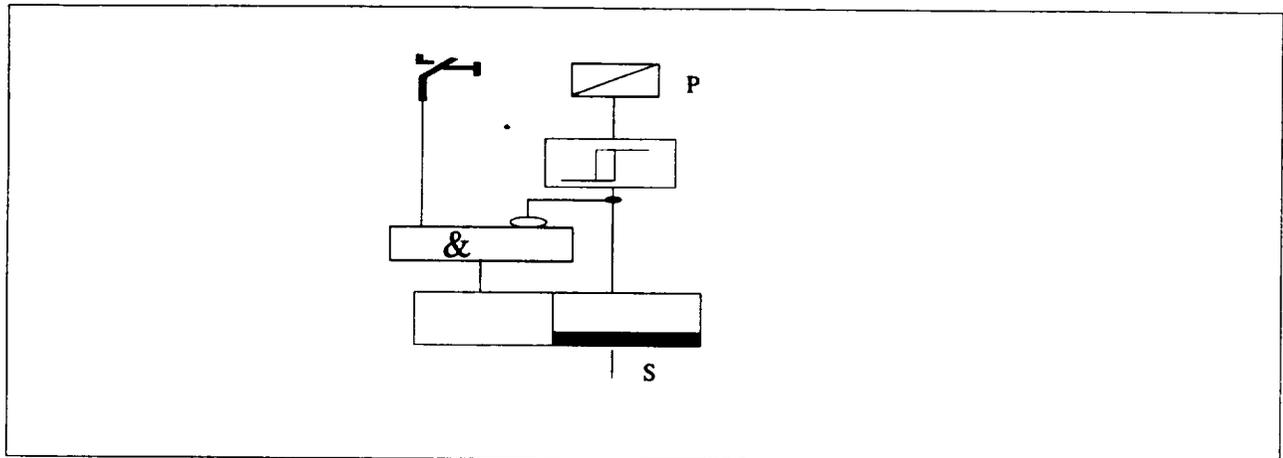
Failures are classified according to the type of effect that they have on the safety function, by their time to detection and their confinement area. Detected failures are then masked depending on the degree to which they can be confined to the individual-loop level, the system level or to the switchgear. Cyclic operation of TELEPERM XS in conjunction with self-monitoring and protection measures ensure that all failures with a potential for influencing the safety function are detected, signaled and masked immediately in communication means, function computers, subracks and analog input modules. Hidden failures with a potential for affecting the safety function are possible in digital input modules and output modules with infrequent signal transitions. Failures that affect the entire subrack have the greatest impact. This is possible for failures in the subrack itself and in the backplane bus interfaces of the modules inserted. Failures in interfaces with serial buses can cause failure of whole bus segments in rare cases. The quantitative contribution of each type of failure needed for overall reliability calculation is derived as part of a module-specific effect analysis for the component level.

#### 2.7.2 System Architecture and Component Failure Effects

The system architectures of the safety I&C are largely set on the basis of probabilistic and deterministic design requirements. A suitable architecture is usually defined empirically, the task being to find an architecture that is suitable for all safety functions under defined constraints. The constraints result from the postulated component failure combinations, from probabilistic requirements regarding loss of system function and spurious actuation, from test and repair requirements and from operational requirements such as space reserves. When an architecture is found, it is then possible to show by deterministic and probabilistic analysis that all constraints have been observed with due consideration of failure behavior.

For simple systems, relevant system characteristics can usually be determined without detailed analysis. Even for complicated system architectures, good estimations can be obtained without detailed analysis.

A simple safety function will now be used to illustrate how different architectures can affect the main reliability features and which typical failure effects are decisive in different architectures. The safety function to be employed is defined in Figure 2.13.

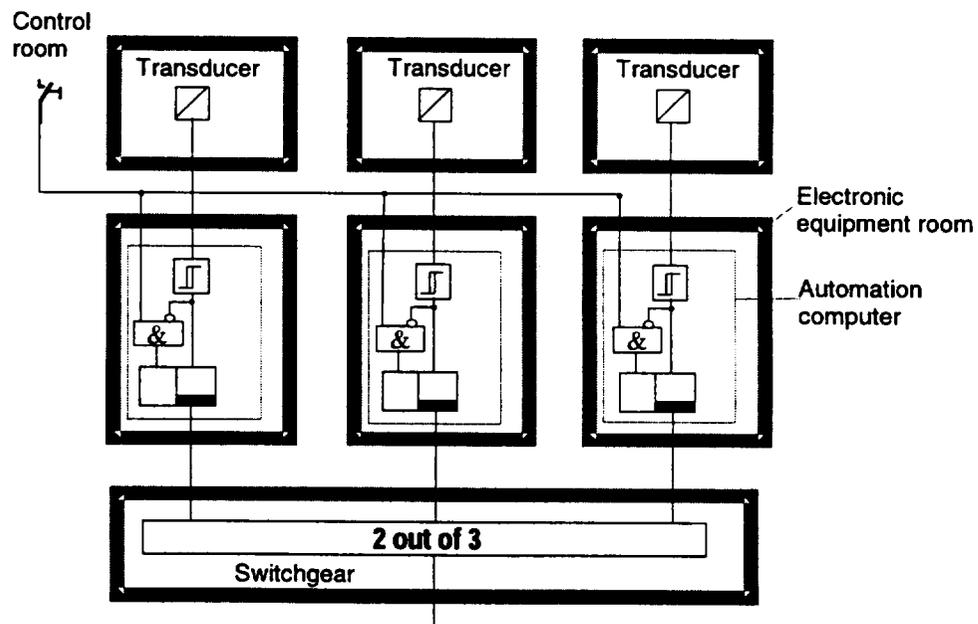


**Figure 2.13 Simple Safety Function**

If the process variable “P” exceeds a defined limit, the safety action “S” must be actuated. The safety action must remain active until it is canceled by operator intervention. A constraint which applies here is that a single fault in the I&C must not have any effect on the mechanical systems.

**2.7.2.1 Single-Level System Architecture with Fault Masking in the Switchgear**

A simple structure that fulfills this requirement consists of three independent function processors, each of which monitors its own measured value and forms actuation signals that are then applied to 2-out-of-3 voting in the switchgear (Figure 2.14).



**Figure 2.14 Single-Level Architecture with Majority Voting in the Switchgear**

Each automation computer contains I/O modules for analog input, digital output modules to operate the switchgear and a function processor to implement the safety functions. For periodic testing and for alarm outputs, a communication processor is also required that is connected with the service unit via a monitoring and service interface (not shown in the figure).

Considering the individual automation computers at first, it has to be stated that each automation computer acquires the process variables and the reset signals via analog and digital input modules, processes the safety function and outputs the result of this processing via digital output modules to the switchgear where the signals are gated by voting.

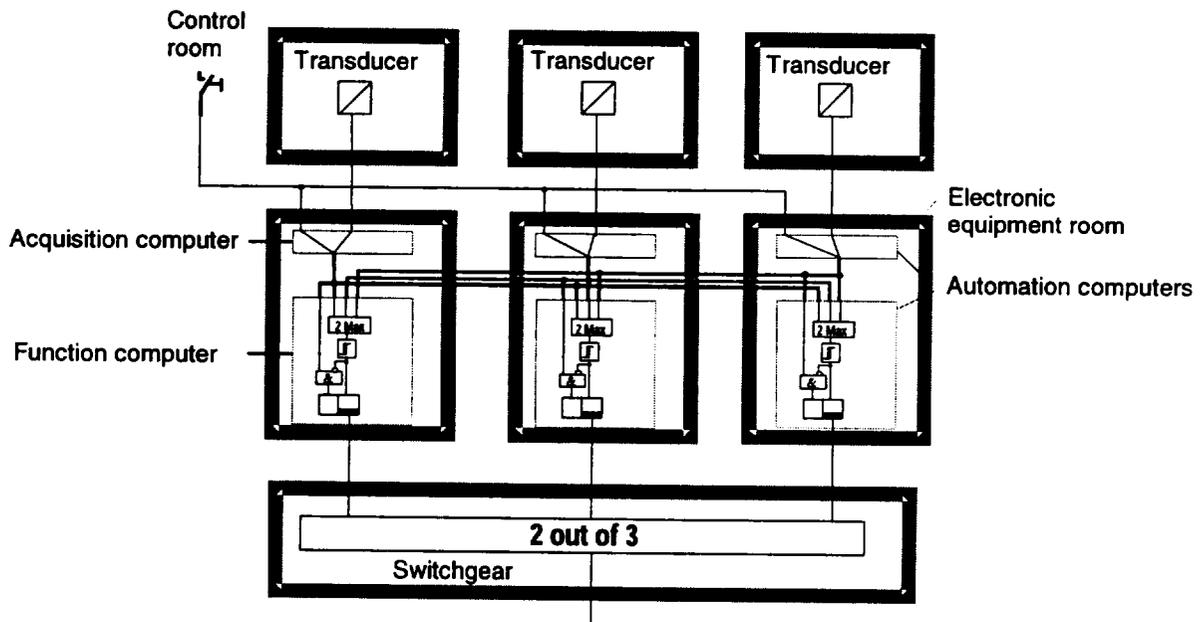
The failure behavior of this architecture is as follows: failures in the measuring peripherals and failures in the analog input modules that are not detected by system-inherent monitoring mechanisms remain concealed down-circuit for signals that fail low, with signals that fail high resulting in spurious actuation on one channel. The failures in digital input modules not detected by system-inherent monitoring mechanisms remain concealed. Failures in the subracks and on the function processor cause change to the defined fault behavior which is detected and signaled via the monitoring and service interface. The failures in the digital output modules not detected by the system-inherent monitoring mechanisms remain hidden in the failure direction "false," whereas the failure direction "true" results in spurious actuation on one channel. The hidden failures in the measuring peripherals and in the analog input modules are decisive for reliability.

A configured monitoring function in the monitoring and service interface that compares the redundant signals of the three automation computers and annunciates a fault if deviations are found detects such failures spontaneously. This means that concealed failures in the measuring peripherals and in the analog input modules are ruled out to the greatest possible extent. Otherwise the failure behavior remains unchanged.

On principle, this simple structure has the disadvantage that all failures, both in the measuring peripherals, in the automation computers and in the switchgear are not masked until the final processing level, i.e. the switchgear. The non-availability of a processing train of this type increases quadratically with the number of elements in the train for a triple redundant configuration followed by 2-out-of-3 voting. From the probabilistic point of view it is an advantage to mask failures at every processing level. In this case the non-availability of a processing train only increases linearly with the number of elements in the train. It should not be forgotten, however, that the simplicity of this structure has considerable advantages.

#### 2.7.2.2 Two-Level Architecture with Fault Masking for Measured Value Acquisition

Because the measuring peripherals have a major effect on the non-availability of safety functions, architectures that render possible failure detection in the measuring peripherals and masking early on in processing are advantageous from the probabilistic point of view. This is achieved by separating the task "Acquisition and distribution of measurement signals" from the task "Processing of the safety function." This by definition results in a two-level architecture (Figure 2.15) in which the first level acquires, filters and then distributes the measurement signals. The second level, i.e. the processing level then receives the three redundant measured values of the process variables. The true value is determined from these three measured values by on-line validation in a selection function block. The safety function is then executed with the true value. The formation of the true value includes both signal monitoring and masking of any faulted value detected.

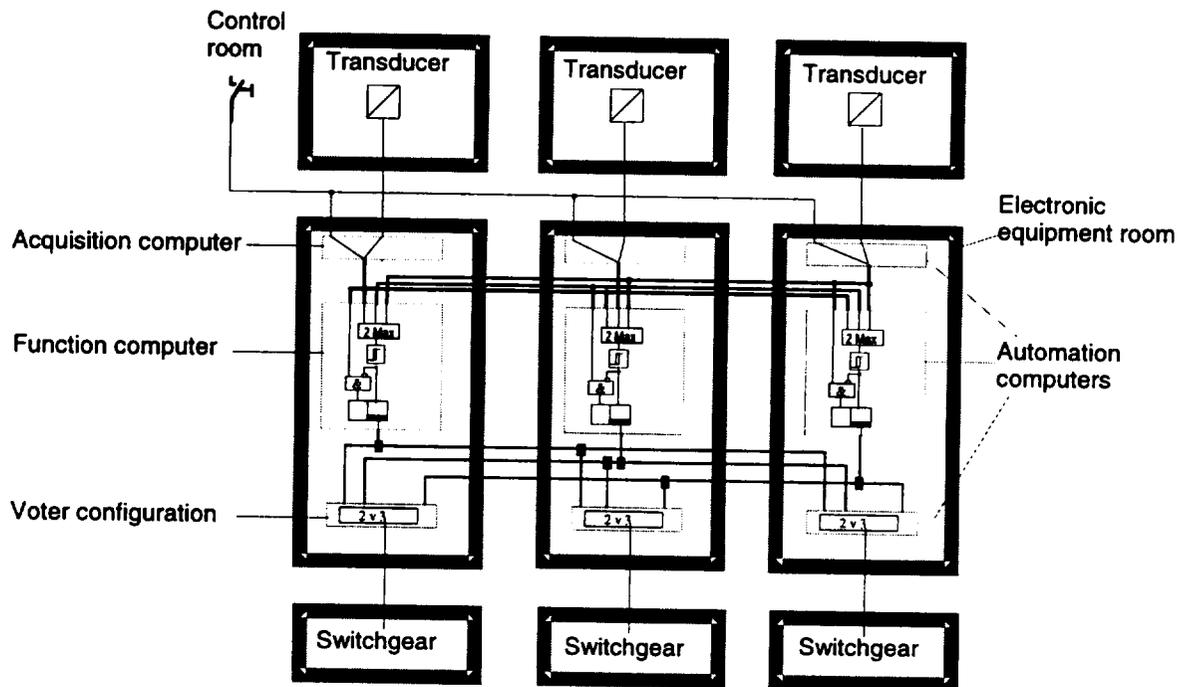


**Figure 2.15 Two-Level Architecture**

Nearly all faults in the measuring peripherals and in the acquisition level can be masked with this fault propagation barrier configured. Only faults in the measuring peripherals or in the analog input module that cause “frozen” measured signals for plant values that are almost constant remain hidden. Spurious signals to the switchgear are only possible in the event of failures in the corresponding digital output module or a processing computer. Failures in digital output modules can also cause passive failures. Failures in subracks and in the function processor are detected spontaneously result in the defined failure behavior of the function processor.

### 2.7.2.3 Three-Level Architecture with Fault Masking for Signal Processing

In many cases, it is required that a single failure is controlled within the safety I&C (that is within the system). In this case, a voter configuration is installed at the interface with the switchgear. This evaluates the results of the three function computers with 2-out-of-3 voting deriving a valid signal from this.



**Figure 2.16 Three-Level Architecture with Voter Configuration**

As already described, the voter consists of two independent master/checker pairs whose results are OR gated so that a component failure cannot cause loss of function. Spurious actuations due to failures in the digital output module are possible but very infrequent. Reading back of the digital output signals makes it possible to detect and mask failures of this type.

The 2-out-of-3 voting circuit of the voter masks all failures from the up-circuit processing level. This affects the level of the function computer itself and all communications equipment between the two processing levels. Each half of the voter also masks failures of the other half.

#### 2.7.2.4 Summary on Architectures

The architecture not only affects the reliability but also the response time behavior for a safety function. It can be estimated that every level implemented with computers in signal processing causes a signal delay of one or two computer cycles. A typical cycle time of 50 ms is assumed so that the time delay between analog input and digital output to the switchgear is 50 or 100 ms for a single-level configuration, 100 till 200 ms for a two-level configuration and 150 till 300 ms for a three-level configuration. More shallow system architectures therefore have response time advantages while multi-level architectures are often advantageous from the point of view of failure behavior and reliability.

## 2.8 Identification

The identification system is open for application specific demands. Thus in general the plant specific identification system can be considered.

## 2.9 ***Interference-Free Communication***

### 2.9.1 **Specification of the Requirements**

Specific communication methods are applied to ensure interference-free communication inside the TELEPERM XS system as well as to other systems e.g., the plant process information system. The following communication channels (Figure 2.17) have to be considered:

#### a) **Communication Between the Redundant Initiation Trains of a Safety I&C System**

It is required that in case of a single failure of one of the redundant initiation trains ( $R_i$ ) or within one communication channel ( $C_{n,m}$ ) the trains still available will continue to operate as designed on the basis of the remaining information to ensure the required safety functions without consequent failures.

#### b) **Communication from the Initiation Trains to the Plant Process Information System**

The Communication ( $C_{ex}$ ) from the initiation trains of the safety I&C system to the plant process information system (PPIS) is done via the monitoring and service interface (MSI). This communication channel is only used unidirectionally by signaling messages to the plant process information system according to the application specifically designed messages. The intermediate monitoring and service interface serves as isolation means in conformity with the TELEPERM XS system architecture.

#### c) **Communication Between the Initiation Trains and the Service Unit**

The communication ( $C_s$ ) between the initiation trains of the safety I&C system and the service unit has to be examined in two different ways:

- For normal cyclic operating, it has to be ensured that normal cyclic operating of all function processors (SVE1) can not be impaired as far as no specific release is given.
- In case of intended interventions from the service unit by the service personnel e.g., for:
  - changing parameters or tracing signals
  - performance of periodic tests
  - diagnosis for identification of failures inside the I&C system.

It has to be ensured by the release logic independently processed by the service unit that only one of the redundant initiation trains of the safety I&C system can be influenced from the service unit at a time.

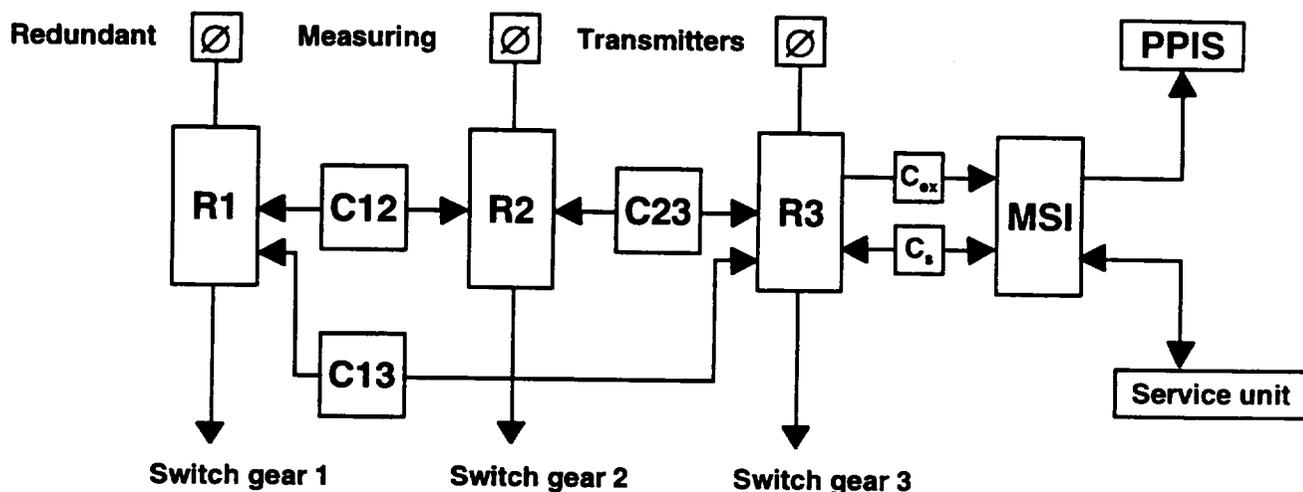


Figure 2.17 Communication Channels

### 2.9.2 Profibus Communication

The L2 LAN of the TELEPERM XS system is applied for communication to:

- a) a redundant initiation train of the safety I&C system,

as defined in Section 2.9.1.1.

The use of L2 communication in TELEPERM XS is demonstrated by for the example of communication between two SVE1 function processors belonging to two different initiation trains of the safety I&C system. It is valid for the SINEC L2 communication as well as for the SINEC H1 communication that all sending activities are initiated by a SVE1 processor and that the messages are addressed to the receiving processor. The intermediate communication modules serve for data transfer only without influencing the message data.

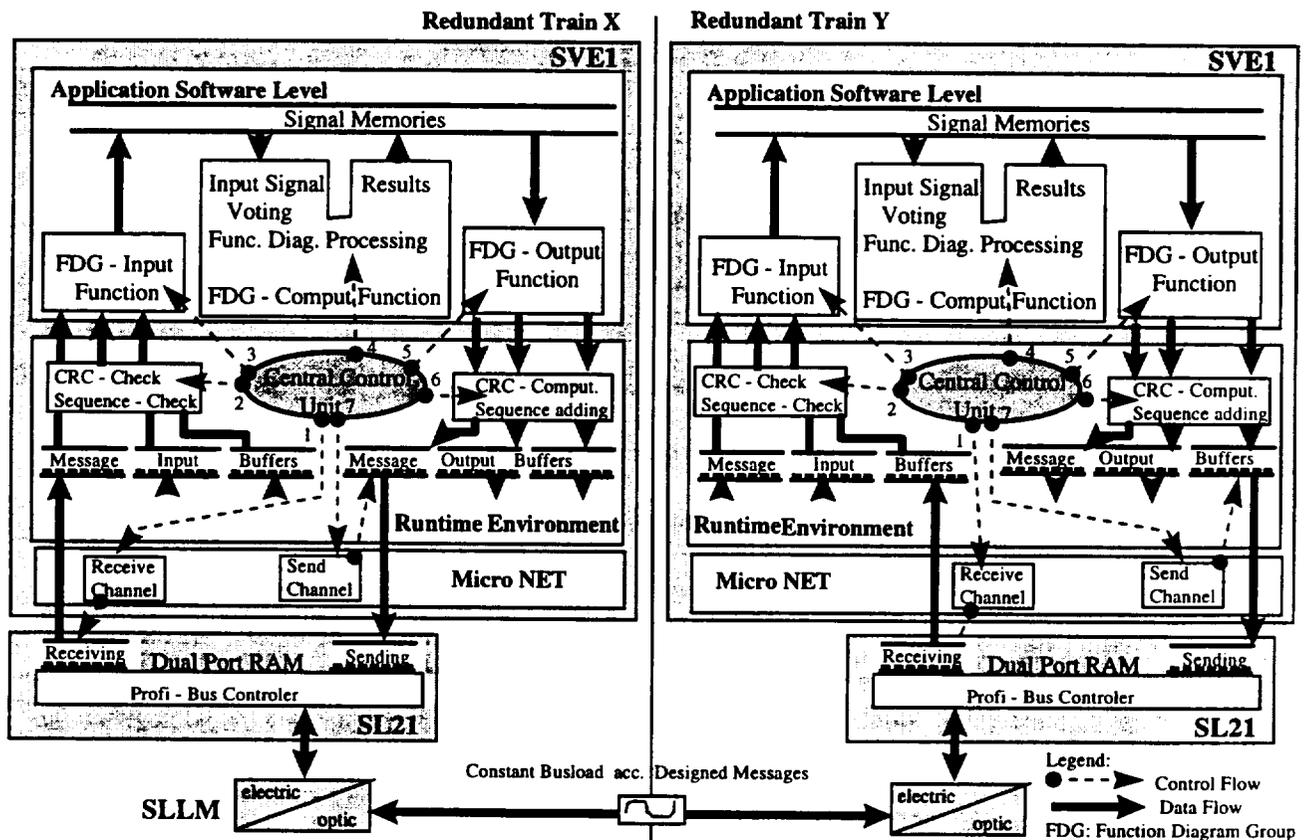


Figure 2.18 Interference-Free L2 Communication

The functioning of the runtime environment is essential for the TELEPERM XS communication principle. It triggers and controls all actions during the processing cycle (Figure 2.19). The central control unit sequentially starts the main processing phases in each processing cycle (indicated with 1 to 7 in Figure 2.18). The TELEPERM XS processing cycle is started by the central control unit of the runtime environment by triggering the MicroNET to transfer the messages from the "Receive Dual-port RAMs" of all linked SL21 modules into the corresponding message input buffers, which are implemented inside the runtime environment (phase "1"). After that the integrity of the message transfer from the sending SVE1 processor to the receiving SVE1 processor is checked by means of CRC checksums (for the occurrence of individual bit errors) and by means of sequence increment (to ensure that a new message has been received) (increment > phase "2"). It is essential that the message transfer is performed by the MicroNET as a buffer read operation without any dependencies on how the SL21 module works, as the content of the "Receive Dual-port RAMs" is taken as it is and checked. Incorrectly transferred messages (detected by means of CRC checksums) or old messages are marked invalid.

The "Function Diagram Group Input Function" starts the processing on the application software level. The individual signals inside the message are identified and allocated to the signal memories (phase "3"). During the next phase (phase "4"), the function diagram modules belonging to the individual functions are processed.

It is essential for guaranteeing absence of interference for communication that signals received via communication channels are first validated on the function diagram level before the algorithm is processed to overcome the problem of missing or faulty information, so that in spite of any kind of faulty information caused outside the discussed SVE1 processor the results of function diagram module processing is correct.

During function diagram processing (phase "4A" to phase "4H" in Figure 2.19), all provisional results are stored in dedicated signal buffers.

As last step of application software processing, the "Function Diagram Group Output Function" is triggered by the central control unit to collect the results of the processing to data structures of new messages (phase 5). The runtime environment then adds the message header to the message data including the correct CRC checksum and the cycle counter and stores the messages in the message output buffers (phase 6).

The last application-specific activity performed within one processing cycle by the central control unit is to trigger the MicroNET to transfer the output messages from the output buffers to the sending dual-port RAMs of the respective SL21 module (phase 7). Then in the time remaining until the end of the discussed processing cycle the self-monitoring program is processed triggered by the runtime environment (Figure 2.19).

According to the token passing principle when receiving the token, the SL21 module starts an automatic polling to recognize that new messages are stored in its dual-port RAM. The SINEC L2 communication control works autonomously without any possibility to influence the strictly cyclic processing of the linked processor systems.

As to guaranteeing freedom from interference for the communication between the redundant trains of a TELEPERM XS safety I&C system, the following items are essential:

- application of a fiber optic transmission medium to ensure that effects caused by electromagnetic interference EMI can not propagate,
- individual memories for each message ensuring the separation of the data flow for sending and receiving,
- cyclic processing of all tasks (message transmission included) without any possibilities of influencing the linked communication systems (independent control flow of SL21 and SVE1),
- check on the received messages for whether the transmission has been performed with valid message data as a matter of principle,
- input data voting on principle to generally ensure correct input data for function diagram module processing.

TELEPERM XS: A Digital Reactor Protection System

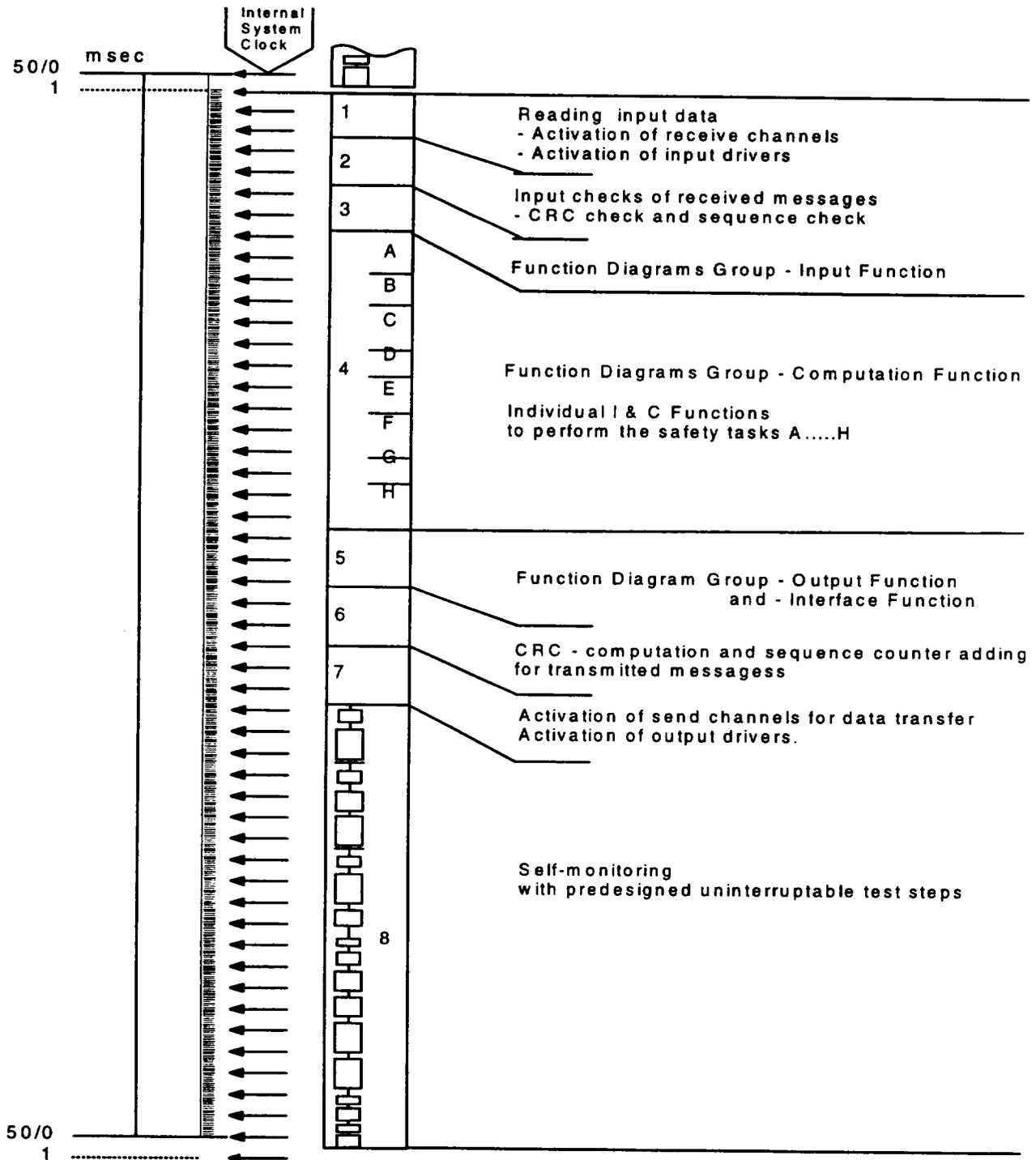


Figure 2.19 Processing Cycle

### 2.9.3 Ethernet Communication

The H1 LAN of the TELEPERM XS system is applied for communication to:

- a. the plant process information system and other comparable information systems, and
- b. the service unit,

via the monitoring and service interfaces (MSI) in conformity with the TELEPERM XS concept.

The monitoring and service interfaces are equipped with SVE1 function processors with a runtime environment and application software generated by means of SPACE as are the computer systems in the TELEPERM XS safety initiation trains. The data from and to the computer systems in the safety initiation trains is processed according to the alarm and information concept engineered and documented on function diagrams and the information purposes.

Figure 2.20 shows the communication between the SVE1 function processor - via the SCP1 communication processor and the SINEC H1 LAN - and the monitoring and service interface MSI.

For the adaptation of LAN protocols to other processor systems with protocols differing from that of the TELEPERM XS system as e.g., the plant process information system, additional receiver-specific gateways are applied as a link.

For the realization of both the above mentioned communication tasks (a and b), SINEC H1 communication processors SCP1 are installed in the TELEPERM XS subrack. The number and length of messages on the SINEC H1 LAN is constant.

Each function processor SVE1 sends one signaling message and receives one command message per cycle. These messages are transmitted from and to the service unit via the monitoring and service interface. In addition a constant number of engineered (SPACE) data messages are cyclically send to the MSI for further use by any other receivers.

The number and length of data messages is fixed according to the designed signal transmission quantity from the function diagram modules whereas the signaling and command messages have a designed maximum information quantity transferring "no information" most of the time.

For discussion of interference-free communication, the data flow and the control flow between the relevant hardware and software modules is represented in.

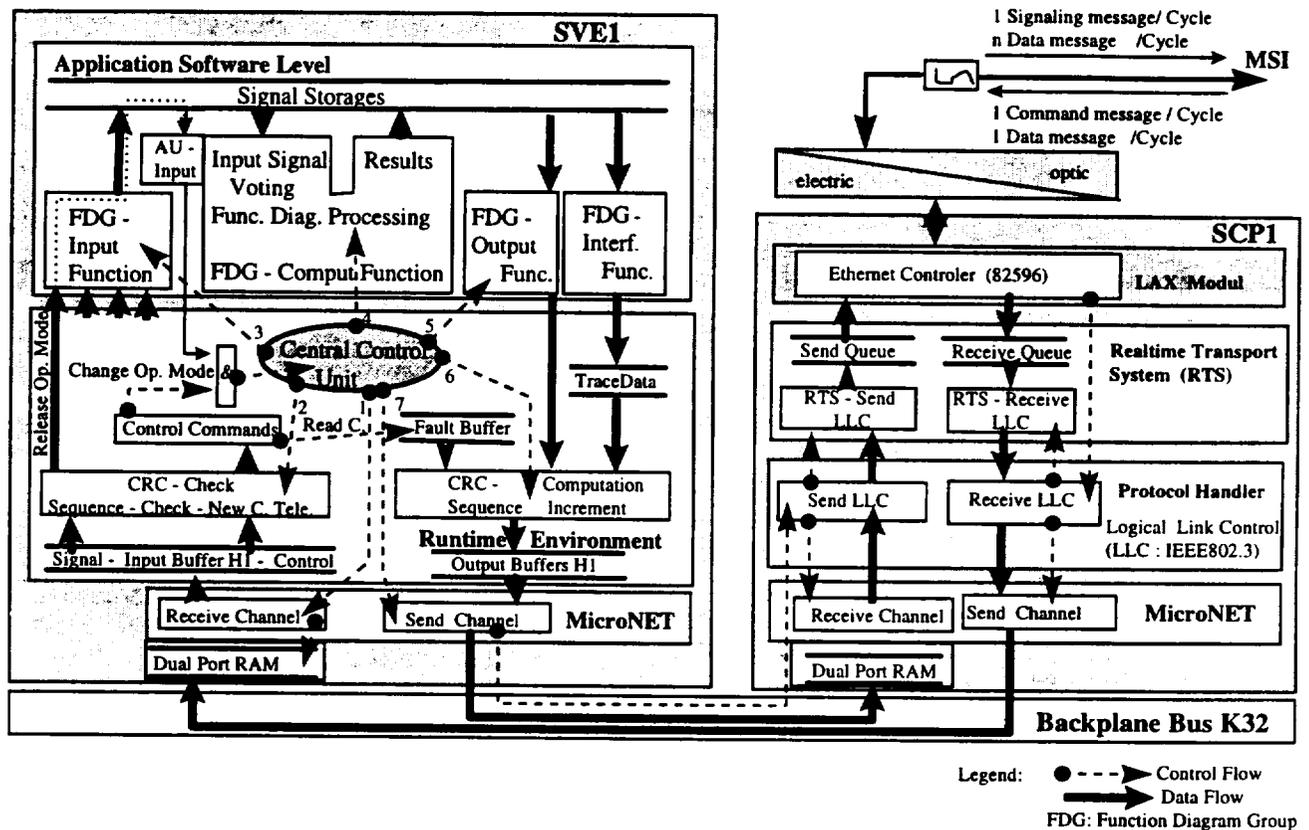


Figure 2.20 Interference-Free H1 Communication

The operation of communication processor SCP1 is described in the following subchapters.

The SVE1 Sends Data via the SCP1

The sending SVE1 calls a Send\_Channel function of its own MicroNET. This function then copies the data via the K32 backplane bus into the dual-port RAM on the SCP1. After this, the function triggers the protocol handler on the SCP1. The protocol handler transfers the data from the own dual-port RAM to the realtime transport system (RTS) system using MicroNET services. The RTS system adds the data to the send queue. The send queue is processed by the Ethernet controller of the LAX module, which is responsible for the Level 2 (OSI reference model) protocol handling.

The SVE1 Receives Data via the SCP1

Each packet on the H1 network is received by the Ethernet controller on the LAX module. The Ethernet controller processes the Level 2 protocol and transfers the data to the protocol handler via the RTS system. The protocol handler transmits the data to the dual-port RAM of the receiving SVE1 using MicroNET services. The use of the dual-port RAM ensures the isolation of data flow and control flow.

### The Operation of the Runtime Environment

As discussed above for the SINEC L2 communication, the operation of the runtime environment is decisive for ensuring freedom from interference on the operation of the SVE1 processor for SINEC H1 communication.

The central control unit initiates the main processing phases of the SVE1 processor in the same way as it is described above for SINEC L2 communication.

The processing of the data messages up to the final step in which the SVE1 processor stores them in the dual-port RAM of the SCP1 is fully comparable to SINEC L2 communication. What has to be discussed specifically for SINEC H1 is the communication from and to the service unit via the monitoring and service interface (MSI).

Although the service unit only sends command messages if somebody of the service staff has initiated a command, the MSI cyclically repeats the last initiated command with the engineered (maximum) message data length.

Each SVE1 processor cyclically reads the data content of the SVE1's dual-port RAM via the MicroNET services. This dual-port RAM is installed on the SVE1 board and linked to the K32 backplane bus but is handled by the runtime environment in the same way as described for the dual-port RAM of the SL21 module in Section 2.9.1.2. The MicroNET service copies the data into the input buffer of the runtime environment. The CRC check is applied to ensure that only messages which have been transmitted correctly are used. The sequence check is used to detect whether a command message is new. Only new command messages are noticed by the runtime environment. Repeated command messages serve for a cyclic check on the correct functioning of the Ethernet communication from the MSI to each SVE1 processor but the content is ignored.

There are three different types of commands which are transmitted as command messages:

- a) Commands which do not influence the cyclic processing of the application software and therefore do not violate the requirement for interference-free communication (e.g., commands for reading the fault buffer and transfer the stored information to the service unit). These commands are called information commands.
- b) Commands for changing the operating mode of a SVE1 processor, which in the first step influence the central control unit. Of course this type of command communication strongly influences the operation of a processor in one safety initiation train. But this is an intended action by the service staff. It has to be demonstrated during factory acceptance test that there are sufficient barriers to ensure the integrity of the safety I&C system.
- c) After the operating mode has been changed, there is of course a greater number of commands available, e.g., for analyzing faults, tracing data or changing parameters. But these commands introduce no generally new safety state compared to item b) above.

### Information Commands

In case of a faults detected by self-monitoring or by the runtime environment (e.g., loss of messages, data integrity), the information on this fault is sent to the service unit by means of a cyclic signaling message at the end of the cycle.

As a back-up to enable a later fault analysis if the service unit is out of operation, this information is stored in the fault buffer. The complete fault buffer (data capacity for 60 faults) can be read from the service unit on demand by means of a command message.

The length of a signaling message is constant and is engineered according to the required data tracing quantity (TELEPERM XS permits a maximum number of 200 analog values and 600 binary values or equivalent combinations).

### Change of the Operating Mode

There are four operating modes of the runtime environment:

- cyclic processing,
- parameterization,
- functional test, and
- diagnosis.

For each of these additional commands are released - increasing in number downwards in the list - until in the operating mode "diagnosis" the full range of commands is available.

Regarding the item "changing the operating mode of a SVE1 processor," the following sub-items have to be considered:

- the level of influence on the processor, i.e., the amount of possibilities to influence the processor fixed by the operating mode, and
- the release conditions for changing the operating mode of a SVE1 processor depending on the key release given by the reactor operator from the control room and on the operating modes of all SVE1 processors with respect to the architecture-dependent fault tolerance conditions of the entire system.

Typically the release conditions are processed by the monitoring and service interfaces, each for the SVE1 processors of its redundant initiation train.

### Operating Modes

The following operating modes form different levels of influence on SVE1 processor in a safety initiation train from the service unit.

- Cyclic processing  
No possibility to influence a SVE1 processor from the service unit. The change-over to another operating mode requires a specific release.
- Parameterization  
Prerequisite is the release for changing the parameterization mode. In the parameterization mode, the application software (function diagram group modules) is continued to be processed in the same way as in the mode "cyclic operating." A return to normal "cyclic operating" is possible at any time without additional conditions.

Possible actions from the service unit are:

- changing parameters which are designed as "changeable during power operation" for e.g., optimizing parameters of close loop control or adapting parameters in case of a stretch-out operation,
- initiating the tracing of signals belonging to selected function diagrams presented on the service unit.

After having fixed the signals to be traced, it is possible to return to normal cyclic operating while tracing is continued.

- Functional test  
Prerequisite is the release for changing to the "functional test" mode with respect to plant operating conditions (decision by the reactor shift) and to the operating modes of the TELEPERM XS system in other initiation trains. If one processor in another chain is already in the functional test mode or is identified to be faulted, the release for changing to the operating mode of an additional processor is locked. When changing to the functional test mode, the processing of the application software (function diagram group modules) is stopped at first.

By means of additional control commands given from the service unit, the processing functions are activated according to test conditions:

- activation / deactivation of input / output drivers
- activation / deactivation of message send functions and message receive functions
- activation / deactivation of function diagram module processing
- preset of data in input and output buffers
- tracing of signals

The operating mode "functional test" is finished by a processor reset command. After a delay time of about 10 seconds, processing is continued in "cyclic operating" mode.

- Diagnosis  
Prerequisite is the release for changing to the "diagnosis" mode. The release is aging depending on the decision by the reactor shift and the operating mode of the TELEPERM XS systems in the other initiation trains.

Generally, this operating mode is activated during power generation only in case of a faulted safety initiation train to identify the defective component. This operating mode is also required if as a consequence of changed functional requirements by the plant process, a

new version of the application software has to be loaded on the SVE1 processors. Generally this procedure is applied during refueling periods only, as then the functional requirements on the safety system are reduced.

The operating mode "diagnosis" is finished by a processor reset command.

#### Release conditions

The release conditions for the SVE1 processors of the redundant initiation trains are processed by the respective monitoring and service interface (MSI).

The relevant operating state of each TELEPERM XS subsystem is detected by processing a valid life signal by means of a specific "AU-Output" function diagram module (AU = runtime environment) and is then transmitted via a data message to the TELEPERM XS monitoring and service interface. The correct operating of the SVE1 processor is confirmed by this life signal with respect to the following criteria:

- The function diagram modules are processed cyclically.
- The "cyclic operating" mode is actuated.
- No active fault indications.
- Input signal voting indicates no deviations exceeding the tolerated measured signal deviation limits.
- All modules - the input/output modules included are plugged in the subrack.
- Message transmission from the SVE1 to the MSI operates correctly.
- Transmitted messages have neither status "TEST" nor status "FAULT."

Through this messages, each MSI receives the information on the correct operating of the SVE1 processors of the respective redundant trains. This information is exchanged between the MSIs via the SINEC H1 LAN in such a way that each MSI gets the information on the operating conditions of all processors.

As all messages are to be transmitted cyclically, the loss of messages (two or more missing messages) is interpreted by the receiving MSI as that the related SVE1 processor does not have a confirmed status "cyclic operating."

To ensure that a change of the operating mode for cyclic operating to an other mode can not violate the engineered fault tolerance conditions of a safety systems, the release for changing the operating mode requires that cyclic operating is confirmed for all SVE1 processors of the other initiation trains and in case the architecture additionally includes an independent system to apply functional diversity, cyclic operating has also to be confirmed for all processors of this independent system. Only SVE1 processors of the same safety initiation train are tolerated to be e.g., in the mode "functional test" at the same time.

This release depending on the system state is logically gated with the release conditions given by the reactor shift in the control room by key switches, to ensure that e.g., the start of actuators of the plant safety system's initiation via the TELEPERM XS safety I&C system during integral tests is tolerable for the plant operating conditions.

The valid enable signal is differentiated in the levels "parameterization," "functional test," and "diagnosis" and processed by the MSIs for the coordinated processors is cyclically transmitted via specific data messages to the SVE1 processors.

The logic conditions are engineered on function diagrams (SPACE) with respect to the system architecture under application of the same quality assurance measures as applied for all parts of the application software.

A change of the operating mode can be initiated only after both the command message from the service unit and the data message with the enable signal from the MSI are received by the SVE1 processor.

The AU-Input function diagram additionally permits to engineer processor-dedicated specific change-over conditions.

In case that during the performance of a test a fault is spontaneously detected in an other train, the functional test mode is automatically finished according to the above mentioned criteria, the SVE1 processor-specific life signal included.

In order to achieve this, the command for finishing the functional test mode and returning to normal cyclic operating is copied in the header of each cyclically transmitted command message by the MSI. The return to normal cyclic operating conditions needs about 10 seconds.

Information on the operating state of all SVE1 processors is also displayed on the service unit, so that the service staff is informed why a command message to a dedicated SVE1 processor is ignored. It may be necessary that in advance e. g. the test in an other train is to be finished or that in case of a faulted SVE1 in an other train this problem is to be solved with priority.

The following features are ensured for TELEPERM XS communication via SINEC H1:

- Communication to other digital systems (outside TELEPERM XS) is engineered to be unidirectionally in the application software of the MSIs.
- The change of the operating mode of a dedicated SVE1 processor required the transmission of two different messages coordinated in time: a command message initiated by the service staff from the service unit and a signal message containing the enable signal. An unintended change of the operating mode by faulty or erroneous messages can be excluded.
- The enable signal is processed such
  - that the required system fault tolerance conditions are ensured (which is considered as the basis for the system architecture) and
  - that an intended change of the operating mode of a dedicated SVE1 processor is compatible to the being plant operating conditions.

- If during test performance the initial release conditions which allowed the start of a test are violated (either by plant operating conditions, decided by the reactor operator, or by a fault in the TELEPERM XS system) the test is automatically finished by a reset and the system returns to normal cyclic operating in some seconds.

### 3.0 **Software Life Cycle Process Planning**

#### 3.1 ***Functional Software Characteristics***

##### 3.1.1 Introduction

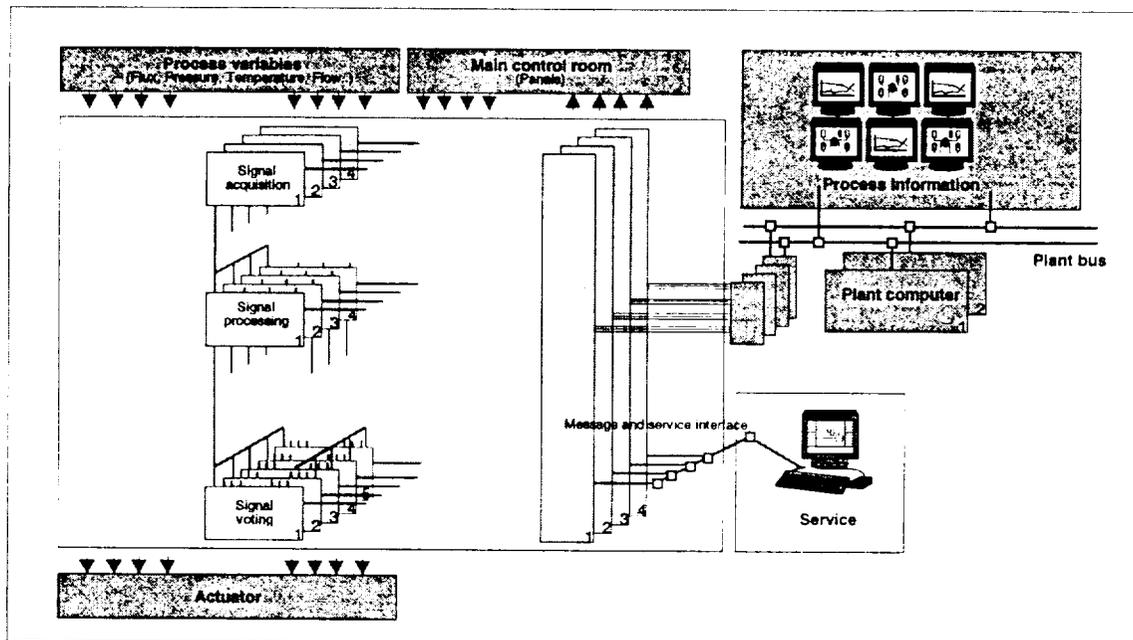
TELEPERM XS is a digital instrumentation and control (I&C) system for safety-related and safety-critical applications in nuclear power plants. It provides a flexible framework for I&C engineers to create safety applications tailored to the specific needs of an individual application. Typical TELEPERM XS applications range from closed-loop control of reactor process variables to monitoring safety critical process variables, limit value monitoring, and initiating reactor trip procedures. TELEPERM XS consists of a set of high quality software components that have been developed and qualified according to the specific needs of digital nuclear reactor safety systems. These components support basic safety mechanisms like error propagation barriers, redundant system structures, cyclic operation, and deterministic system behavior. These features form the basis for any safety application. TELEPERM XS supports I&C engineers with an easy to use specification environment for creating high-quality safety applications in a cost effective way.

The software architecture of TELEPERM XS was designed to give I&C engineers this simple, flexible software platform that also meets the stringent safety and quality demands for software in safety systems of nuclear reactors. In order to achieve this goal, the TELEPERM XS architecture incorporates a great deal of domain-specific knowledge.

Beginning with the safety criteria that influenced the software architecture, this section describes the conceptual architecture, the module architecture, the layer architecture, the execution architecture and the code architecture of TELEPERM XS.

##### 3.1.1.1 **System Overview**

TELEPERM XS applications can be flexibly tailored to the needs of a specific application. The scope of applications ranges from simple single failure tolerant 1-out-of-2 systems to large 2-out-of-4 systems. A typical TELEPERM XS application is a distributed, redundant computer system. Typically it consists of three or four independent redundant data processing paths (redundancies), with two or three layers of operation each. The individual computers communicate via networks, which are typically used as end-to-end connections.



**Figure 3.1 Hardware Topology of a Typical TELEPERM XS Application**

The signal-acquisition layer in each redundancy acquires analog and binary input signals from transducers in the plant (e.g., temperature, pressure, and level measurements). Each signal-acquisition computer distributes its acquired and pre-processed (for example filtered) input signals to the data-processing computers in the next layer. Thus each data-processing computer is provided with the same set of input information.

The data-processing computers perform signal processing such as limit value monitoring and closed-loop control calculations. The results are signals destined for actuating actuators like pumps, valves, or control rods. The data-processing computers send their results to the voter-computers.

In the voter-computers, the results of the four redundancies join together. A voter-computer controls a set of actuators. Each voter receives the redundant actuation signal of the redundant data-processing computers. The voter's task is to compare this redundant information and compute a validated (voted) actuating signal, which is used for actually actuating the actuators. This is done typically by 2-out-of-3 or 2-out-of-4 voting of the redundant input signals. The voted signals are then output by the voters to the switchgear system to actuate the assigned actuators. A voter computer itself can be built as an inherent redundant and fault tolerant computer to increase its availability.

In addition to the computers of the above-described automatic path, typically there is a message and service interface computer (MSI) in each redundancy. The MSI is connected via LANs to each automatic computer in its redundancy and to its assigned voters. It serves as a gateway

between the computers of the automatic path and other, non-safety-relevant systems, such as the service unit, gateways to process-control computers, or monitoring and control computers.

Each computer can consist of one or more racks mounted in cabinets. It contains one or more processing modules for performing the signal processing. Communication is handled by dedicated communication processors. Communication links can be electrical (within the same cabinet) or optical (for links between different cabinets). Signal acquisition and output is done by a set input and output modules for analog and binary signal types.

### 3.1.1.2 Global Analysis

The global analysis identifies the key characteristics that are to be met by the TELEPERM XS system. These are:

- Support of a distributed computer systems architecture.
- High LAN communication performance (up to 200 messages of 500 Byte per second).
- Sufficient computing power.
- Diskless operation.
- Use of standard components and protocols wherever possible.
- Portability and independence from a special hardware platform, in order to save the high costs of the TELEPERM XS software development resulting from future changes in the hardware platform.
- High quality software components developed and qualified according to the specific demands of digital safety systems in nuclear power plants.
- Implementation of fault tolerance mechanisms and error propagation barriers.
- Deterministic system behavior: computing times and network loads must be independent from actual state of the process to be controlled, and guaranteed response times must be met.
- Avoidance of data dependencies in computing algorithms.
- Need for a domain specific engineering tool to allow I&C-engineers to easily create applications and to communicate the results with the process engineers for verification purposes. Strict separation of the specification of the required I&C functionality (which is independent from the specific realization) and the specifics of a computer-based target system.
- Simplicity is required for reactor protection system in general. With regard to TELEPERM XS, this meant keeping things as simple as possible, i.e., no unnecessary functionality, simple and easy to understand protocols, and static system design with no dynamic resource allocations.

While the first 4 points were handled by selecting suitable hardware components, the latter points were the driving forces in the design of the TELEPERM XS software architecture.

### 3.1.1.3 Technological Factors

#### Hardware Platform

Innovation cycles in digital automation systems are rather short compared to the typical lifetime of a nuclear power plant (up to 40 years). On the other hand, the expected number of components to be sold is rather low, compared to other standard industry application fields. As a result, the decision was to avoid using specially developed hardware, but instead to use specially selected and qualified standard industry components. Equipment qualification is performed through type testing.

The selected hardware platform uses an Intel 486 processing module, with 512 KB RAM, 512 KB EEPROM for storing program code, and a 32 KB EEPROM that can be used for storing application-specific data. Input and output modules are standard components designed for a state-of-the-art automation system; these modules are in service in very high numbers in different other technological fields. Communication processing modules are available for the LAN standards Ethernet and Profibus (Process Field Bus). These boards are mounted in racks and communicate via a 32 bit multi-master-capable parallel backplane bus. Other LAN components like electrical to optical transducers and star coupler components are also state-of-the-art industry standard components.

All hardware components were type-tested according to national and international standards to ensure that the specific requirements for components of reactor control systems for nuclear power plants are met.

#### Software Quality Requirements

The high quality demands for software in digital safety systems of nuclear power plants had a significant impact on the TELEPERM XS development. There are several national and international rules, standards, and guidelines that must be observed. Among these, the IEC 880 "Software for Safety-Systems in Nuclear Power Stations" specifically addresses the concerns of digital safety-systems. This standard is internationally accepted and applied by licensing authorities all over the world. It specifically addresses the software development process, and the required documentation and verification and validation loops.

#### Specification Method

The specific nuclear quality requirements apply not only to the TELEPERM XS development itself but also to any TELEPERM XS-based application. One key aspect is the strict separation between the system requirements specification and the hardware and software system design and implementation. The system requirements specification is typically prepared by process engineers, using informal methods like textual descriptions, diagrams and mathematical notions. On that basis, system design and implementation is done by I&C engineers. The results of the system design phase must be reviewed by the authors of the requirements specification. Normally neither party is an expert in software-engineering, so an approach using classical SA/SD techniques like data-flow diagrams or EBNF notations is not suitable for them. Thus another specification method had to be used for developing TELEPERM XS applications.

#### 3.1.1.4 Requirements and Architectural Challenges

Architecture plays a major role in achieving desired levels of non-functional system properties such as performance, schedulability, failure avoidance, fault tolerance, and redundancy. Consequently, requirements for such properties and for transient functionality such as start-up, shutdown, and error-recovery have a major influence on the architecture.

TELEPERM XS had to support the following requirements through built-in system features. These requirements had a global impact on its architecture:

- Deterministic system behavior to guarantee maximum response times, computing times and network loads under all situations.
- Scalability from simple stand-alone systems to large scale, 4-times redundant distributed system structures.
- Support of explicitly specified redundant system structures.
- Fault tolerance mechanisms like exception handling and error-propagation barriers. These are needed to guarantee limited failure-confinement areas, outside of which an assumed failure inside the area will have no effect.
- Support for maintenance and diagnosis from a central service unit. This is needed for system-testing as well as for periodic testing on-site. It is also used for reading and acknowledging system error messages. It must be possible to put individual computers into test mode or turn them off for maintenance purposes without affecting the operation of the remaining system.

#### 3.1.1.5 System Design Guidelines

The factors, requirements, and standards listed so far led to particular strategies and design guidelines, summarized below:

##### Use of Function diagrams and Automatic Code Generation

In order to meet the specific quality standards required for digital safety systems of nuclear power plants (namely the requirements of the IEC 880 standard), and to meet them at a reasonable cost, it was decided that project-specific software must rely extensively on prefabricated and type-tested software components. A formal specification method would be used to automate the specification process as far as possible. To facilitate the required verification of the software specification by the process engineers, the decision was to use function diagrams as a specification language. Code generation and its subsequent verification would be automated in order to improve software quality and to keep the costs low.

Function diagrams (FDs) are a standard documentation notation used in the I&C field, one that is well understood by both process and I&C engineers. FDs can be used to specify I&C functionality in a way that is completely independent from the specific implementation method (hardwired analog systems, digital systems or any other). TELEPERM XS uses function diagrams as a formal specification method for developing TELEPERM XS based applications. Based on this formal specification, automatic code generators generate the code for the target system. This approach offers several advantages:

- Process and I&C engineers can use a notation that is well known to them and with which they are already familiar.
- The comprehensibility of the I&C engineer's specification by the process engineers is increased, thus reviewing the specification results can be done more effectively.
- Because FDs provide a method for specifying a system's functionality without considering any target-specific features, portability and independence from the target systems is inherent.
- The use of FDs as a domain-specific formal notation method allows the use of automatic code-generators to generate the target systems code. No manual coding is needed, thus decreasing costs and increasing the quality of the code.
- A set of independent tools can be used to analyze the automatically generated code and to check its correctness against the original specification. This verification step can be done automatically.
- The FDs also provide the documentation for the implemented functions.

#### Portability and Target System Independence

The concept of using function diagrams supports portability and target-system independence because the function diagrams specify only the system's functional behavior without any dependencies on the target-system hardware. To maintain this independence in the resulting code, the code for the function diagrams must be independent from the target-system platform. The link between the function diagrams and the target-system platform is provided by a target-system adaptation layer, called the runtime-environment (RTE). The RTE provides a target-system-independent execution environment for the function diagram modules. The RTE itself uses services of the operating system software and the target-system hardware. Thus the function diagrams are kept completely target-system independent. The RTE, although target-system dependent, was designed for easy portability to other platforms.

#### Operating System Software

Because of the IEC 880 standard requirements, the operating system software of the TELEPERM XS target system had to be developed to meet the required quality standards. The development of the operating system software was based on pre-existing software components for the standard industry system. This forms the basis of the TELEPERM XS hardware platform. Development was done according to a phase-model and QA plan set up for the TELEPERM XS development. Another benefit of this approach was that the operating system software could be tailored to the needs of TELEPERM XS. Functionality not needed for TELEPERM XS could be removed, and the operating system's kernel could be simplified. This serves one of the key characteristics: simplicity.

#### Deterministic Behavior

To meet the requirements for deterministic system behavior with guaranteed maximum computing and reactions times and constant communication loads, the design calls for each processor unit to process its assigned function in a strictly cyclical way with a predefined cycle time. During each operating cycle, the sequence of processing steps is always the same: read input signals from input modules and/or messages, compute the assigned function diagram

modules, and send output signals to output modules and/or messages. The control flow is independent from the process data. The message sizes and communication rates are constant, resulting in constant communication loads under all circumstances.

### Scalability

Scalability from simple stand-alone systems to large distributed 4-time redundant systems with several processing layers is achieved by separating function diagrams from the hardware structure specification, and by using a highly configurable runtime-environment (RTE), which processes the FDs on each CPU.

### Redundancy

Implementation of explicitly specified redundant systems is supported both by the scalability concepts and by the definition of special function blocks (FBs) that can be used in FDs for signal validation of redundant signals.

### Fault-Tolerance and Failure-Propagation Barriers

Fault-tolerance is supported by the use of explicitly specified redundant system structures. For implementing failure-propagation barriers, the concept of message and signal status was introduced. Each message and each signal carries a status. Faulty signals or signals from a missing or corrupted input message are marked as faulty and are disregarded by the function blocks used for signal validation. When unexpected exceptions arise, the computer's cyclic operation must be stopped and its output signals set to defined values (zero signals).

#### 3.1.2 Conceptual Architecture

The conceptual architecture for TELEPERM XS applications serves as the functional specification and design of I&C functions and as a bridge between the project-specific system requirements and the resulting code.

A project-specific system requirements specification contains the functional requirements of a TELEPERM XS application. It defines one or more functional units called LEFU. Usually it is prepared by process engineers. The system requirements specification also contains requirements like maximum reaction times for a specific LEFU or independence constraints of different LEFUs.

The LEFUs defined in the system requirements specification are then analyzed by I&C engineers and transformed into a formal software-specification, using function diagrams (FDs). This analysis includes tasks such as evaluating the input and output signals of the system, identifying common sub-modules in different LEFUs, and partitioning a LEFU into several function diagrams. These diagrams are understood by both process and I&C engineers, thus facilitating the verification process of the software specification.

Function diagrams give independence from specific technology of the target system. They specify the signal processing, but impose no other implementation restrictions. In order to simplify the subsequent automatic code generation, a FD must be processed by a single processor unit, i.e., it can not distributed over several processing units.

A function diagram consists of a set of basic building blocks, called function blocks (FBs), which are connected by lines, indicating the signal flow. A function diagram itself may import and export signals from and to other function diagrams or input/output modules. It consists of one or more pages. Figure 3.2 shows an example of a function diagram. The center area is used for placing and connecting function blocks, thus specifying the FD's signal processing. The left side contains an area for importing external signals from other FDs or input modules, and the right side is used for exporting signals to other FDs or output modules. Signal import and export is based on a naming scheme that identifies each FD and each signal with a unique name code. The bottom area of a FD is used for additional information such as project name, author's name, date of last modification, and the FD's name code. The contents of each FD and the interconnections of its FBs are stored in the specification database.



unit. The software-hardware assignment is also done using the hardware specification diagrams: each processing module contains a list of all FDs that are to be processed on this board. The hardware specification is stored in the same specification database as the software specification.

Two alternative views can be selected for the hardware-specification: one shows the logical connections of the components, and the other shows the physical arrangement of the components in racks and cabinets. Figure 3.3 gives an example of an hardware specification diagram showing the logical connections of the components.

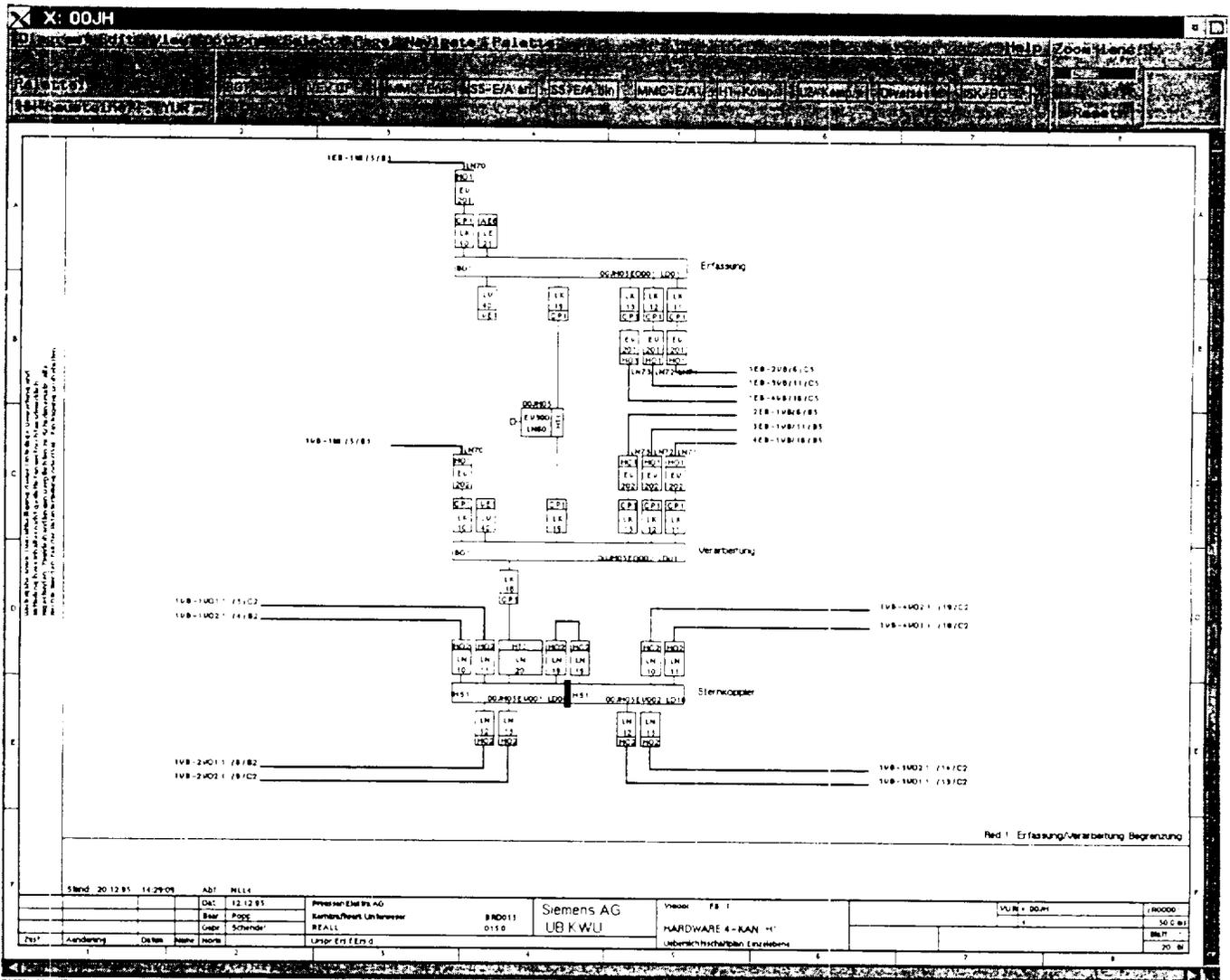


Figure 3.3 Example of a Hardware Specification Diagram

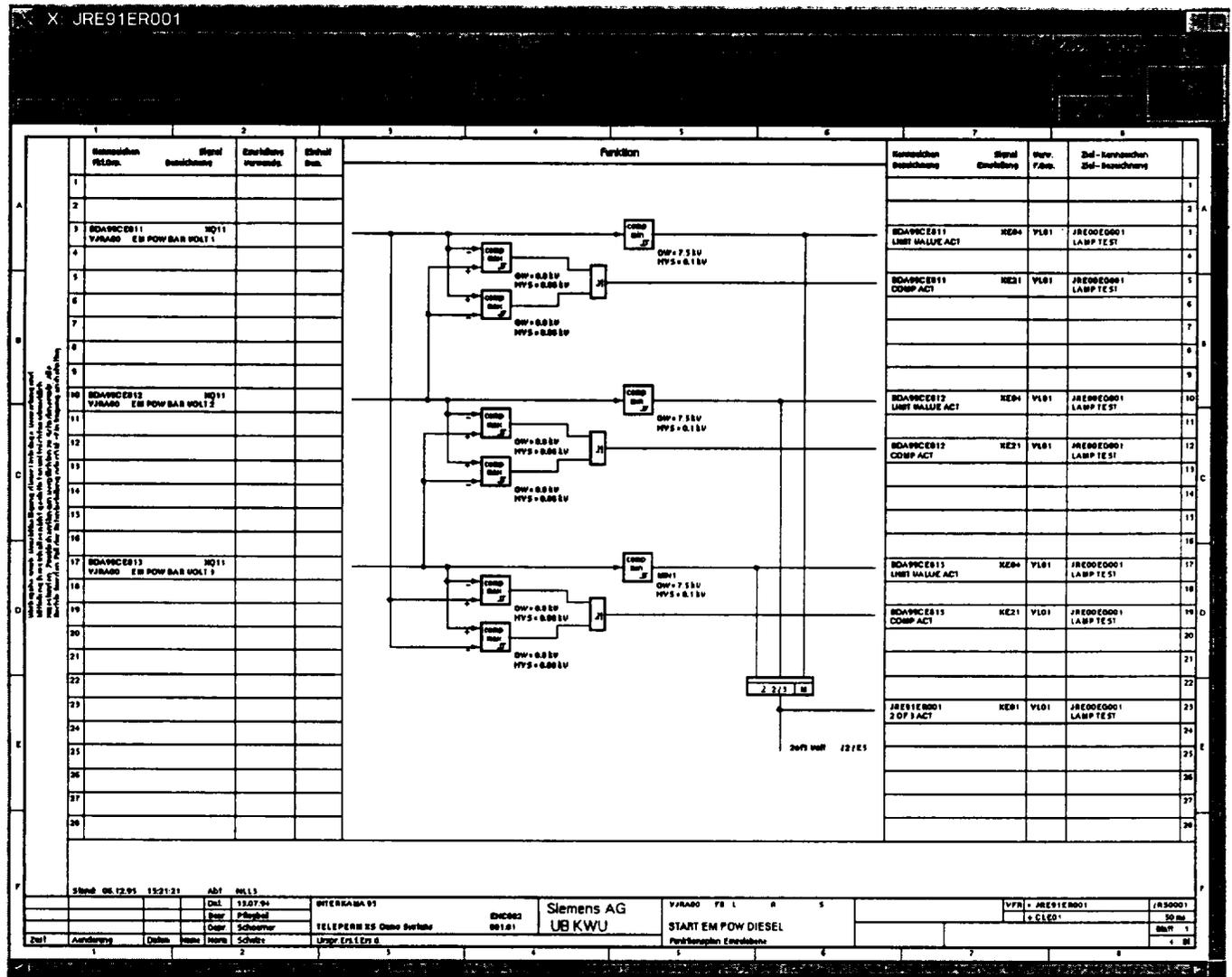


Figure 3.2 Example of a Function Diagram (FD)

In addition to the software specification, the I&C engineers prepare a hardware specification based on their analysis of the system requirements specification. The hardware specification defines all hardware components of the system such as processing modules, communication processors, LAN components, racks, cabinets, etc.; their logical interconnections; and their physical arrangements in racks, cabinets, and rooms.

The hardware specification uses a format similar to that of the software specification, but uses a different set of basic building blocks called hardware blocks. Each hardware block represents a specific hardware component such as a processor-board or input/output module. The hardware boards can be connected to each other: for example, a processor board can be "plugged" into the backplane bus of a rack, or a communication processor can be connected to a star coupler



### 3.1.2.1 Central Design Tasks: Component Modularization, Connector Characterization, Instance Characterization

#### Component Modularization

In order to use function diagrams as a domain-specific formal software specification method, the syntax has to be defined more precisely. A set of well-defined function blocks was defined. Each function block was given a unique symbol, and its functionality, input/output signals, parameters, and start-up behavior were precisely defined. The function blocks are classified into ten categories, as shown in the table below. Although the number of function blocks should be limited for usability reasons, the set of function blocks can easily be expanded with new function blocks if needed.

<b>CATEGORISATION OF FUNCTION BLOCKS</b>			
<b>Category</b>	<b>Description</b>	<b>Examples of FB Modules</b>	<b>Internal States</b>
1	arithmetic operations on analog signals	add, sqrt, abs	no
2	analog signal processing	unit-delay, PT1, differentiation, integrator, controller, digital filter ramp generator	yes
3	selection of analog signals	switches, min/max value selection, sorting	some yes, some no
4	mixed analog / binary signal processing	limit switch	yes
5	logical operations	and, or, xor	no
6	binary signal processing	unit-delay, pulse with specified duration, on- and off-delays, flip-flops	yes
7	selection of binary signals	switches, 3-out-of-4 voting, 2-out-of-4-voting, 2-out-of-3 voting, 1-out-of-2 voting	no
8	domain-specific actuator interfaces	interfaces to control rod drives, pressurizer heating elements	some yes, some no
9	interface to the run-time environment	acquire operation mode permissions, pass error flags	no
10	decode message signals into binary signals	one for each FB module that generates a message signal	no

Function diagrams are used to group and connect function blocks. This is done for several reasons:

- A function diagram must be processed by a single processing module; it may not be distributed over several processors. The processor assignment is specified at the function diagram level, thus supporting the distribution of the software specification to

the different processing modules of the distributed target system. It also facilitates changes or additions in the software specification.

- Each function diagram is identified with a unique name code. This coding scheme gives experts information on the purpose of the function diagram. Thus structuring the software specification using FDs gives a coherent organization for software specification documentation.
- A function diagram will be computed at equidistant time instances. Each FD has a cycle time parameter that determines its rate of processing. The FD's cycle time is specified by the I&C engineers based on required response times or bandwidth considerations. A typical value is 50 ms, although values from 5 ms to several 100 ms are possible. Cycle time considerations also influence the structuring of the software specification into FDs.

### Connector Characterization

Function blocks are connected by signals via their input and output ports. Each port has an associated type, which defines the type of signal that can be connected to it. Three types of signals were defined: for analog (float valued) signals (AS\_t), binary (boolean valued) signals (BS\_t), and message (binary coded) signals (MS\_t).

The rules that define the connectivity of function blocks are based on the signal types of the ports: only ports of the same type can be connected by a signal. This is checked on-line during the specification of a function diagram by the graphical TELEPERM XS Editor.

An underlying restriction is imposed for message signals: each message signal is of the generic type MS\_t, although the bit-mapped message code is specific to each function block type. As a result, for example, the message signal output port of function block 2/4 may not be connected to the message signal input port of the message decoder MD-COUNTER (although both ports are of the same generic type MS\_t). This kind of type-checking is currently not checked at specification time. It is however checked at runtime by the message decoders themselves, based on the FB-ID information that is part of each message signal. Thus a primitive kind of runtime type-checking has been implemented.

Function blocks in different FDs are connected in the same way as they are within the same FD. The only difference is that the connecting signals must be exported by the source FD and imported by the sink FD. Each exported/imported signal gets a unique name code, based on the name code of the source FD. This is in contrast to local signals within a FD, which are not identified by a unique name code.

### Instance Characterization

A unique naming scheme for function diagrams and signals is used. This may be either a standard naming scheme (KKS = Kraftwerks Kennzeichnungs System) or any other plant-specific naming scheme. In general, this naming scheme not only applies to the software specification but is used as a general plant-wide naming scheme for identification of all plant components, such as actuators, transducers, pipes, push-buttons, rooms, cabinets, etc.

Thus each FD gets a unique identification. In addition, all signals that are exported by this FD have a unique identification, containing the identification of the originating FD. For each signal exported by an FD, its destination is identified by naming the destination FD.

During a typical software specification process, the I&C engineer specifies a set of FDs using the graphical TELEPERM XS Editor. To connect function blocks in different FDs, the engineer exports the connecting signal in the source FD and assigns it a unique signal name. Next the engineer opens the target FD and creates a import signal, which is connected to the corresponding function block port. Finally, the engineer assigns the imported signal to the signal name of the previously exported signal, thus defining the connection. The editor automatically checks at specification time whether the imported/exported signal exists and whether the connected function block ports are of the same signal type. If the source or target function block or function diagram is deleted, the signal is automatically marked as "open ended" in the remaining function diagram. By connecting the different FDs with signals, the complete software specification is created.

### 3.1.2.2 Resource Allocation

For the specification of a complete I&C system, its hardware structure and the assignment of the function diagrams to the different processing modules must also be specified.

For the hardware specification, an additional set of symbols has been defined, representing the hardware building blocks of the system (processing modules, communication processors, input and output modules for analog and binary signals, racks, cabinets, LAN components, etc.). The hardware blocks are used for the hardware specification in much the same way as the function blocks are used for the software specification: they are connected by lines, indicating communication connections such as a common backplane bus, or LAN. The complete hardware specification of the I&C system can be defined in this way, with tens of cabinets and hundreds of processing modules.

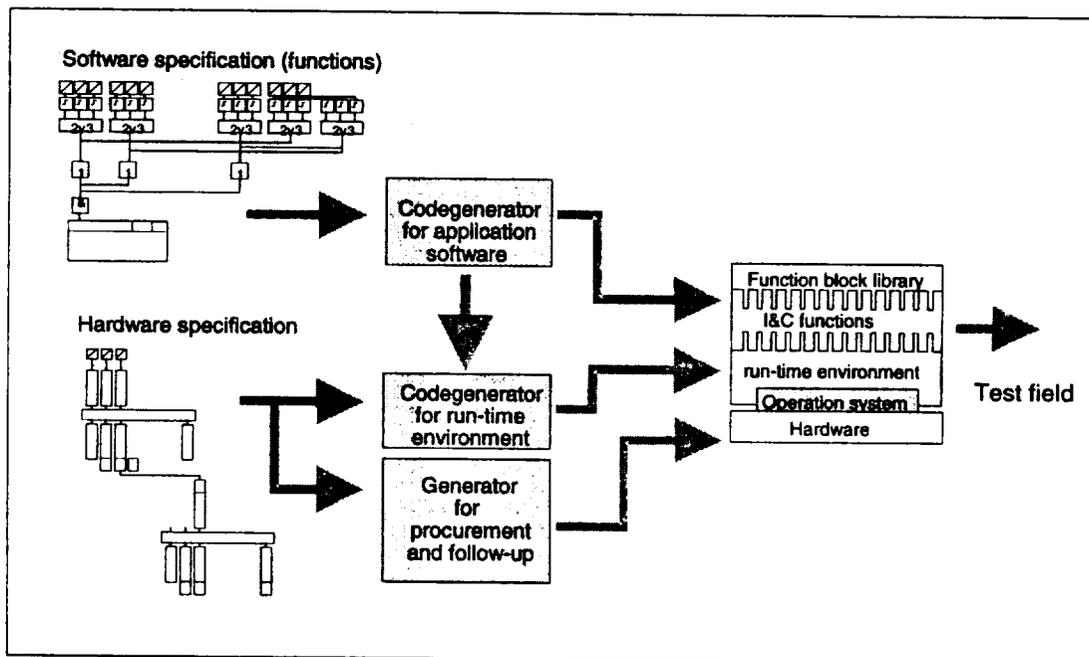
The hardware specification serves several purposes:

- It serves as the definition and documentation of the hardware structure and system architecture of the I&C system.
- It defines the logical interconnections, i.e., communication connections, between the different processing modules. This information is needed later during automatic code generation for evaluating possible communication links, defining messages, and generating the configuration information for each processing module.
- The communication cycle is defined in the hardware specification. The communication cycle is a system-wide parameter. It defines the rate with which messages are exchanged between different processing modules.
- The hardware specification is used to specify hardware-board parameters like addresses, measuring ranges, communication addresses, etc. This information is prepared by a list generator and will be used later during system assembly to set up the hardware components, as well as during operation and for maintenance.
- It gives a complete list of all components used in the system and the numbers of each. This information can be used for preparing the hardware orders.
- It also contains the physical location of each component (room, cabinet, rack, slot), which is used for documentation and during system assembly.

Furthermore, the hardware specification is used for the hardware-software assignment. For this, each processing module contains a parameter list of the FDs that are to be processed by it. Each FD is assigned to a single processing module. Each function diagram has an individual cycle time, at which it must be processed. On one processing module, FDs with up to two different cycle times can be assigned (although this could easily be changed to three or more different cycle times). The ratio of the two different cycle times must be an integer. The faster cycle time determines the rate of operation of the processing module. The ratio between this cycle time and the communication cycle must also be an integer.

### 3.1.2.3 Summary of Conceptual Architecture

The role of TELEPERM XS conceptual architecture is to provide I&C engineers with basic building blocks and tools for designing I&C systems. The system specification is decomposed into a software and a hardware specification phase. Automatic code generators are then used to generate the software specification into code. Figure 3.4 gives an overview of the specification process of a TELEPERM XS application system.



**Figure 3.4 Specification Process of a TELEPERM XS Application**

The TELEPERM XS conceptual architecture supports:

- **Software specification:**  
The software specification consists of function diagrams that are composed from function blocks. The software specification is independent of the target system. It is a domain-specific formal specification that defines the signal processing used to implement the LEFUs defined by the system requirements specification. This software specification is created using a graphical user interface, the TELEPERM XS Editor. The information is stored in the specification database.

- **Hardware specification:**  
The hardware specification contains the complete hardware structure of the target system, with all of its components. For this specification, hardware diagrams and hardware blocks representing the target system's hardware components are used. The hardware specification is also created using the TELEPERM XS Editor. The information is stored in the specification database.
- **Software/hardware assignment:**  
Each function diagram is assigned to one processing module, on which it is processed. This assignment is done while creating the hardware diagrams, by adding the function diagram identification into the parameter list of the processing module, using the TELEPERM XS Editor. The information is stored in the specification database.

The complete specification captures the functional aspects as well as the system's detailed hardware structure. Also the non-functional aspects such as independence constraints, fault-tolerance and timing requirements are implicitly contained in the specification. The specification can be prepared by I&C engineers using notations and methodologies that have been common practice in the I&C community and are well known to these experts. The software specification remains independent from specific details of the target system. The verification of the specification by the process engineers who prepared the system requirement specification is facilitated by the use of a commonly understood notation.

The specification is formal in that sense that all information needed to implement the final code running in the distributed target system is available from the specification database. Also, certain verification procedures such as check of completeness, unambiguity, consistency with naming scheme, and parameter checks can be done automatically at specification time. Using the formal specification and a set of pre-defined rules, the target system code is generated automatically, thus improving code quality and reducing costs. On the other hand, by applying the inverse rules, the generated target system code is analyzed by independent tools and compared to the original database representation. Thus high quality verification of the generated code can be done automatically by independent tools, improving quality and reducing costs.

### TELEPERM XS Editor

A graphical user interface, the TELEPERM XS Editor, is used to create function diagrams and hardware diagrams. Function block symbols or hardware block symbols are selected from a menu bar, placed in the function diagram, parameterized, and connected to other function or hardware blocks (either within the same diagram or across diagrams). The editor supports horizontal and vertical navigation. Horizontal navigation means that signal flow over different function diagrams can be automatically followed. Vertical navigation supports navigation from overview diagrams to detailed level diagrams. All information is stored in a relational specification database. The complete definition of a function block, including its graphical representation, input and output signals, parameters, and state variables, is stored in definition tables in the database. When a function block is added or changed, only its definition data in the database are changed. This is done via SQL-script files. Each function block, as well as each hardware block, has its own SQL-script containing its database definition data.

### Automatic Code Generation

Automatic code generators have been developed to interpret the contents of the specification database and to automatically generate high-level language program code for each function diagram. Communication between function diagrams is done using data messages. These are also automatically generated by interpreting the hardware specification and the software-hardware assignment. Thus the complete code for all function diagrams is automatically generated.

A second code generator generates the configuration data for each processing module. This configuration data contains the set of FDG-modules to be processed, the cycle time, the list of messages and communication channels, the list of input/output modules to be used, etc. The configuration data is also derived from the specification database.

### Automatic Code Verification

Automatic code generation greatly reduces the probability of coding errors and reduces coding time significantly. It also gives the unique opportunity to automate the required verification process of the code itself. For this automatic verification, independent tools were developed. The tools parse the generated code, transform it into an internal representation, and compare this representation to the information stored in the specification database. This approach not only reduces the error probability, but also greatly increases the quality of the verification process itself, as compared to manual verification.

#### 3.1.3 Module Architecture

TELEPERM XS online software can be subdivided into three categories:

- Application Software:  
The application software implements the I&C functions of a specific TELEPERM XS application system by use of FDG-, FD-, and FB-modules.
- Platform Software:  
The platform software consists of the Runtime Environment (RTE) and its sub-modules, the I/O drivers for the input/output module interface, the exception handler, and the self-test software. The runtime environment acts as a virtual machine for executing the application software (I&C functions).
- Operating Software:  
The operating software consists of the operating system (MICROS) itself, the communication software (MicroNET) and a hardware set-up- and interface-tool (HOT).

The application software implements the I&C functions. It consists of the function block modules (FB-modules), function diagram modules (FD-modules), and the function diagram group modules (FDG-modules). A FDG-module groups all FD-modules that are to be processed on the same processing module and have the same cycle time. This is done automatically during code generation and may not be done manually. Up to two FDG-modules can be processed by one runtime environment (RTE).

Due to the fact that a formal specification method with automatic code generation is used for the TELEPERM XS application software, the only software modules that change from application to application are the FD- and FDG-modules. The RTE must also be configured for a specific application. This is done by an automatic code generator, which generates a configuration module for the RTE of each processing module. The RTE configuration module contains the interface definition of the FDG-modules to be processed, a list of all messages and communication channels to be used and also the list of all input/output modules.

All other software components, such as the operating system, communication software, RTE, and FB-modules, are the same across all applications. These software components were developed once using a development and QA process conforming to the requirements of the IEC 880 standard. They have been type-tested by external assessors in order to prove their qualification for safety applications.

#### 3.1.3.1 Function Block Modules (FB)

Each function block used in the software specification is implemented in exactly one function block module (FB-module). Due to the simplicity of most FBs, a FB-module is in general implemented by a single function, although in some cases there might be more. The FB-modules are prefabricated and type-tested. They form a library of reusable software components.

#### 3.1.3.2 Function Diagram Modules (FD)

For the function diagrams, a generic module architecture was defined. Each function diagram is implemented in exactly one function diagram module (FD-module).

An FD-module has a predefined interface, defined by a set of global functions and data structures. Strict naming conventions for all symbols used in an FD-module (functions, variables, data types) have been defined. Additional comments in the code preserve all information needed to reconstruct the initial database representation of the FD-module, even its graphical layout. This supports the automatic verification of the generated code. The code-structure of a FD-module was formally defined using a graphical BNF notation. The automatically generated code corresponds to this definition.

An FD-module consists mainly of a set of data-structures implementing the signal-flow between the FB-modules within the function diagram, and contains storage for parameters and state variables of the FB-modules and internal signals of the FD-module. It also contains a sequence of function calls to the FB-modules that implement the signal-processing of the function diagram. This sequence is automatically determined by the code generator by analyzing the signal flow between the function diagram's function blocks.

#### 3.1.3.3 Function Diagram Group Modules (FDG)

The function diagram group module (FDG-module) serves as a wrapper for the FD-modules it contains. The FDG itself is not visible in the software specification. The FDG-module implements the FD-modules' interface to the RTE. It serves to minimize the interface and to hide the internals of the FDG-modules from the RTE; the RTE does not need to know the individual FD-modules contained in the FDG-module, nor their sequential processing order. The FDG-module calls the compute functions of its assigned FD-modules in a sequential order,

which is automatically determined during code generation by topological sorting, based on the signal flow between the FD-modules.

An FDG-module has a predefined interface, defined by a set of global functions. Strict naming conventions for all symbols used in an FDG-module (functions, variables, data types) have been defined. The code structure of an FDG-module was formally defined using a graphical BNF notation. The automatically generated code corresponds to this definition.

An FDG-module consists mainly of a set of interface functions that are called by the RTE. The interface contains functions for input, compute, output, and accessing internal data of the FDG-module and its contained FD-modules.

#### 3.1.3.4 Runtime Environment (RTE)

The RTE's purpose is to give a unified environment for execution of the FDG-modules. The RTE hides all target specifics such as hardware, operating system, communication media and protocols, I/O modules, etc. from the FDG-modules.

The RTE is the central control instance for the execution of the FDG-modules. It controls the cyclic processing of the FDG-modules and controls signal transfers via messages or directly by I/O modules. The RTE also provides the interface of the runtime-system software to an external service unit, through which it can be monitored and controlled (e.g., by signal trace, reading error messages, switching operation modes, FB-module parameterization).

The RTE provides four operation modes:

- **OPERATION:**  
This is the normal operation mode for cyclic processing of the FDG-modules.
- **PARAM:**  
Same as OPERATION, but parameterization of FB-modules and definition of trace data are now allowed.
- **TEST:**  
Used for functional testing. The FDG-modules can be processed in single-step mode, and all external input signals can be inserted from the service unit. Results can be monitored by tracing internal and external signals.
- **DIAGNOSIS:**  
In this mode, direct memory access is granted to the service unit. Special diagnosis programs can be downloaded from the service unit into RAM. In addition, the target-system's debug functions are activated, so that debugging with an external debugger is possible.

Transitions between operation modes are controlled by a set of operation mode release signals, in order to prevent from unauthorized interference from the service unit. These release signals are acquired either directly by input modules or via data messages. They are specified in function diagrams and passed to the Runtime Environment by a special interface function block.

The runtime-environment also includes three sub-modules:

- **I/O drivers:**  
For every type of input/output module, an I/O driver module is provided, which serves for transferring input/output signals between the RTE and the specific I/O module. The I/O drivers also take care of initialization of the I/O modules and detection of errors.
- **Exception handler:**  
The exception handler module handles unexpected situations, such as time-out-, watchdog-, or unexpected-op-code-exceptions. Information about the exception and its context is saved. Depending on the type of exception, the exception handler then either restarts the processing module (through a software-activated reset) or shuts down the processing module in a defined state. Information saved by the exception handler can be read from the service unit via services of the RTE or read directly via serial line connection from the processing module's front plate.
- **Self-monitoring software:**  
The self-monitoring software performs a sequence of self-monitoring checks on the various hardware components of the processing module, such as RAM-test, FEPRM-test, watchdog-test, etc. The self-monitoring is performed during time intervals when no cyclic processing of the FDG-modules is active. It is repeated continuously, and its cyclic processing is monitored by the RTE. Any errors found are reported to the exception handler, which stores the information and takes care of the error handling.

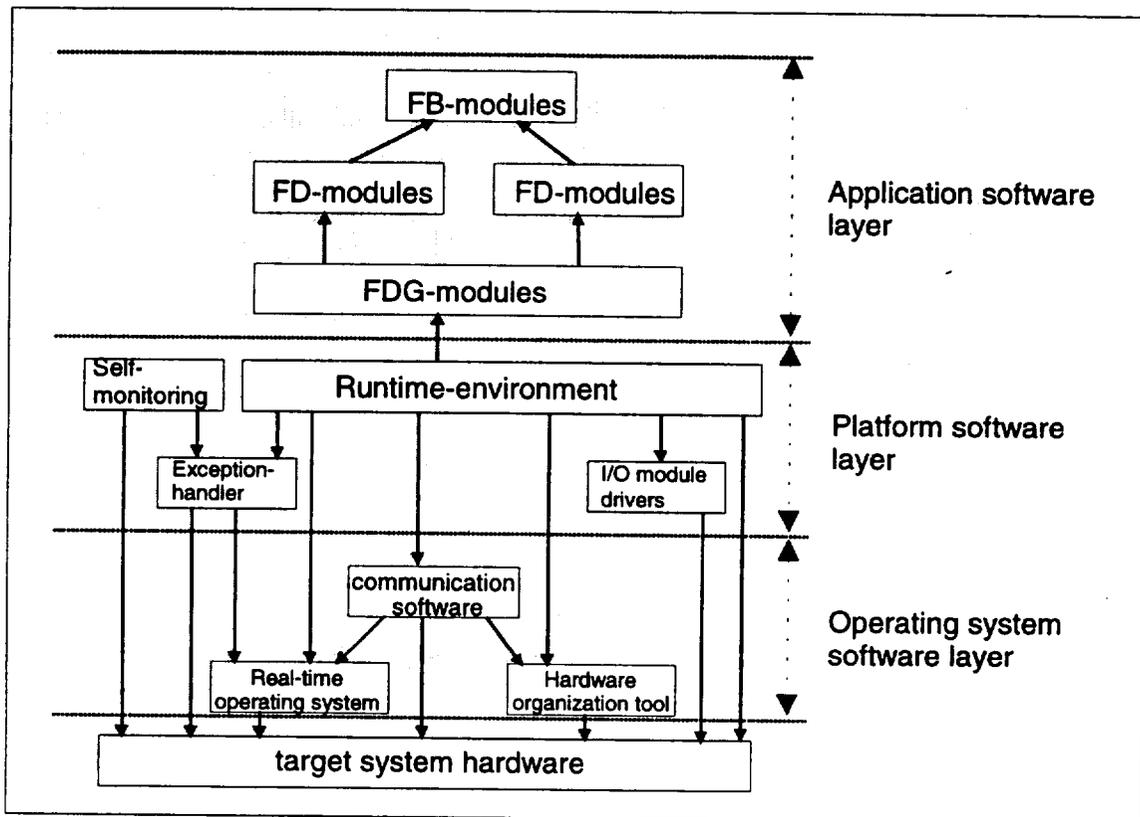
### 3.1.3.5 Defining Layers

Figure 3.5 gives an overview of the three software layers of the Runtime System of one processing module. Each category of modules defined in the previous section is identified as a layer. These layers form a graduated software structure, with increasing target system independence from bottom to top.

The lowest layer is the operating system software layer, consisting of the real-time operating system MICROS, communication software MicroNET and the hardware organization tool HOT. This layer has the strongest dependencies on the target system hardware.

The platform software layer consists of the Runtime Environment and its sub-modules, the I/O drivers, exception handler and self-monitoring software. The platform software interfaces to both the operating system software layer and the application software layer. It also has a direct interface to the target system hardware. The interface between the platform software layer and the application software layer is unified and hides any specifics of lower layers from the application software layer.

The application software layer consists of the FDG- and FD-module and the FB-module library. Since this layer uses only the unified interfaces to the RTE, all modules in this layer are completely independent from target-system specifics.



**Figure 3.5 Software Layers of the Runtime System of One Processing Module**

### 3.1.3.6 Interfaces of the Runtime Environment

The RTE has two major interfaces: the interface to the FDG-modules, and the interface to operating system software layer / target system hardware (target system interface).

#### Interface between RTE and FDG-Modules:

Up to two FDG-modules can be assigned to the RTE. The cycle times of these two FDG-modules must have an integer ratio. The interface consists of a set of unified functions. The RTE calls these interface functions via function pointers. The function pointers and the associated data structures and datatypes are defined in the RTE configuration module, which is generated by an automatic code generator.

- **Input function:**  
This function is used to pass input-signals (from messages and/or input modules) to the FD-modules contained in the FDG-module.
- **Compute function:**  
This function is used to execute the computation of the FDG-module's functionality. The compute function calls the associated FD's compute functions in the correct order. These internally call the basic function blocks of each FD in the required order.

- **Output function:**  
This function is used to pass the calculated output signals of the FDG-module to the RTE. The signals will then be sent via messages or to output modules.
- **Interface function:**  
This function is an universal interface for read and write access to all FDG-module internal data structures, such as parameters, state variables, signals, etc. The RTE uses this interface function for accessing signals when tracing or parameterization of FDG-modules is required by the external service unit.

#### Interface Between RTE and Operating System Software Layer / Target System Hardware:

The target-system interface is the other major interface of the RTE. As a result of the nature of the problem, this interface is not as unified as the FDG-module interface. Basically, the target system interface consists of the following sections:

- **Operating system interface:**  
The operating system interface is restricted to an absolute minimum set of services:
  - real-time related service (pause, task end, and resume after specified time interval),
  - semaphore services, and
  - event-flag services.

No other operating system services are used by the RTE; in particular, there is no dynamic allocation of resources like memory-heap or dynamic task-definitions.

- **Communication software interface:**  
The communication software interface provides a unified interface for sending and receiving messages via communication channels. This is independent of the media used (media supported by the communication software are 32-Bit parallel backplane bus, 16-Bit parallel local extension bus, Ethernet 802.2/3 LAN, and Profibus LAN). Communication services used are:
  - create communication channels,
  - send data via communication channels,
  - receive data via communication channels, and
  - get communication channel status.

#### Target System Hardware Interface:

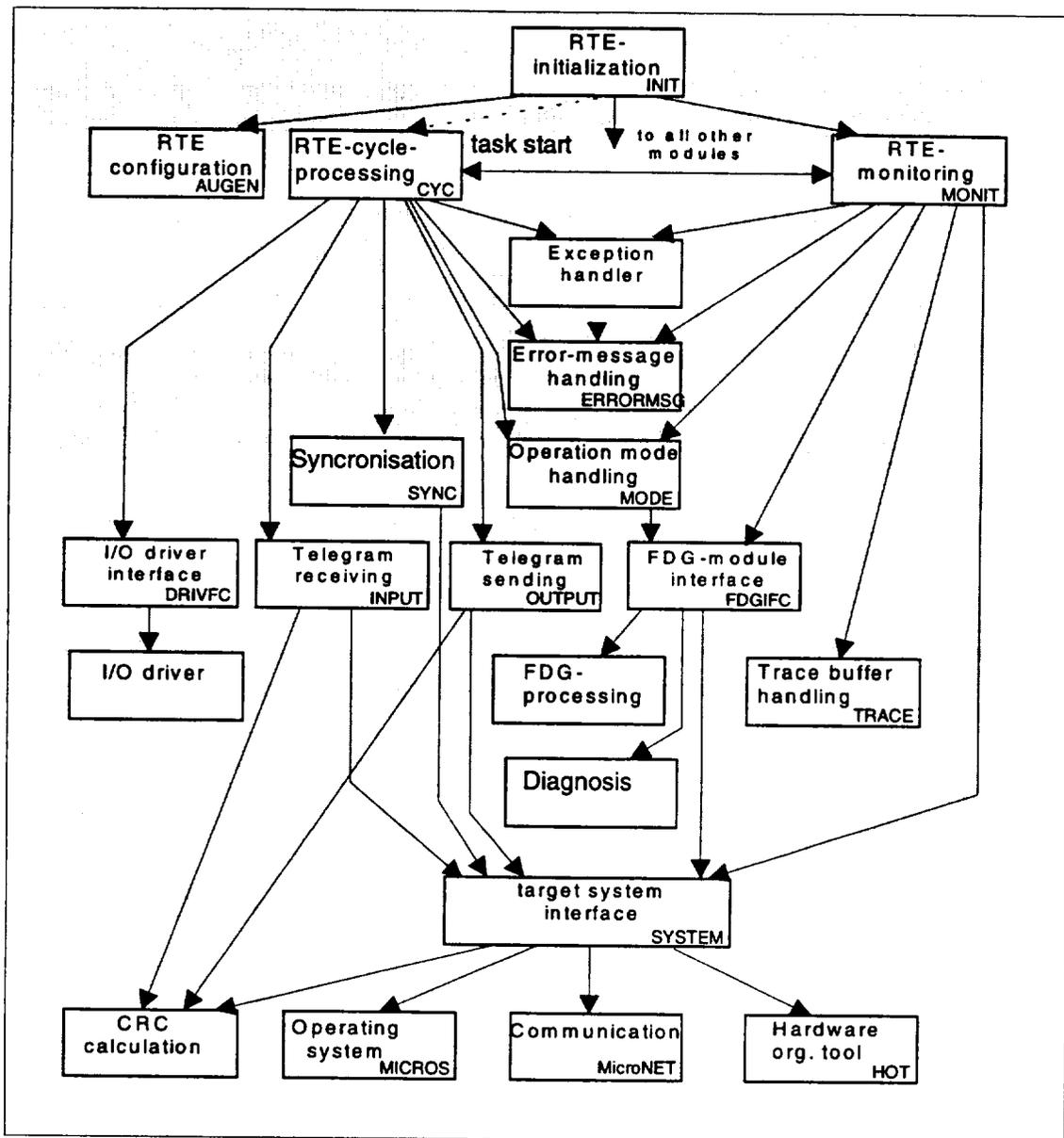
The RTE also has a direct interface to certain components of the target-system hardware, either by direct access or by functions provided by HOT. This includes:

- access to LEDs on the processing module's front-plate,
- EEPROM programming services, and
- watchdog services.

### Defining Dependencies:

The RTE's internal module structure is shown in more detail in Figure 3.6. On top of the structure there are three modules, which control the three major functions of the RTE:

- **Module INIT:**  
This module is the central control instance during start-up of the RTE. It controls the complete initialization phase of the RTE. After initialization has been successfully completed, control is branched to the modules CYC and MONIT. If the initialization fails, the cyclic operation will not be achieved and module INIT ends in an endless loop.
- **Module CYC:**  
This module is the top-most control module for the cyclic operation of the RTE. CYC uses services of underlying modules like FDGIFC (for interface to the FDG-modules), MODE (for handling operation mode transitions), or ERRORMSG (central error-message handling module).
- **Module MONIT:**  
This module controls the RTE's interface to an external service unit. MONIT accepts a set of basic control commands, e.g., reading an error-message-buffer, requesting a change of operating mode, setting a new parameter value. Not all control commands are permitted in every operating mode. For example, during normal operating mode OPERATION, only a very small subset of control commands is accepted by the RTE. This prevents unintended interference from the external service unit during normal operation.



**Figure 3.6 Dependencies of the Runtime-Environment**

The RTE's target system interface, although not as unified as the FDG-module interface, has been designed to facilitate portability. This was accomplished by concentrating all target-system-dependent interface functions in one sub-module, SYSTEM. Porting the RTE to another platform is easily done by adapting the module SYSTEM to the new platform. The RTE has been successfully ported to other platforms like standard PC with OS/2 or Windows NT operating system. In each case, the porting effort took no more than a few days.

### 3.1.4 Execution Architecture

In general, TELEPERM XS application systems are distributed computer systems with several independent processing modules working in the same or different racks. They communicate via

messages, using LAN or backplane bus communication links. Thus the execution of the function diagrams is distributed over several processing modules. This distribution is specified by assigning the function diagrams to processing modules during the hardware specification. The execution structure within a single processing module is fixed and will be described below.

### 3.1.4.1 Defining Executables

Initially the designers considered splitting the software for one processing module into two executables, one containing the operating system software and platform software, and the other containing the application software. This would have meant that after a change in the FD-modules, only the application software executable would need to be loaded into the target system. However, this approach was not implemented because it would have required additional effort in order to resolve address references between the two independently located executables. Also, the benefits of this approach would have been small, due to the relatively short loading times needed.

As a result, one executable is created for each processing module. The executable contains all software modules (operating system software, platform software, and application software). These components are linked and located to give one loadable file for each processing module.

With due regard to the requirements for simplicity and deterministic behavior, a simple and straight forward task organization is used in the runtime system's execution structure, as shown in Figure 3.7.

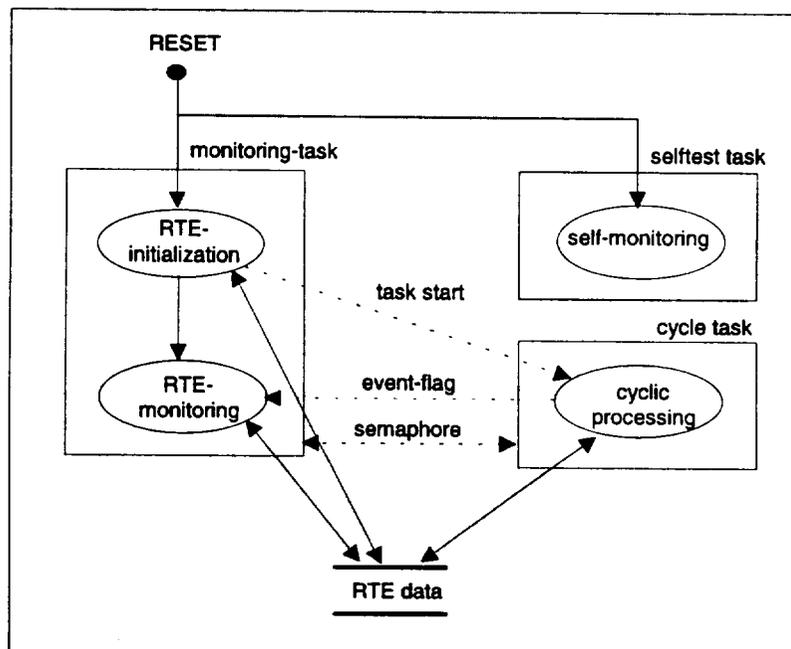


Figure 3.7 Execution Architecture

During runtime, three task are defined:

- **Monitoring task:**  
The monitoring task is automatically started by the operating system after each reset. Control starts in the RTE initialization (module INIT), which controls the complete initialization phase of the RTE. After successful initialization, control is permanently passed to the RTE monitoring (module MONIT). The RTE monitoring controls the processing of control commands received via control messages from the external service unit. When no more control commands need to be processed, the monitoring task is suspended. It is activated by the cycle task each time a new control message is received.
- **Cycle task:**  
The cycle task is activated by the RTE initialization after successful completion of the initialization phase. The cycle task is controlled by module CYC, and operates with a predefined, constant cycle time, which equals the cycle time of the fastest of its FDG-modules. The cycle task handles all communication via messages, the I/O modules, and the cyclic processing of the FDG-modules. It has the highest priority of all three tasks, thus ensuring that the cyclic operation of the FDG-modules always happens with the specified cycle time. If a new control message from the service unit has been received, the cycle task activates the monitoring task to process the control commands. This takes place asynchronously with the cycle task and may last several cycles. After the control commands have been processed, the cycle task takes over the results and sends the command responses to the service unit with the next signaling message.
- **Self-monitoring task:**  
The self-monitoring task is automatically started by the operating system after each reset. It has the lowest priority of all tasks, and is only scheduled when the monitoring and cycle tasks are not active. The self-monitoring task is controlled by the self-monitoring software, which continuously performs tests of all relevant hardware components of the processing module (RAM-test, ROM-checksums, watchdog-test, etc.).

Another task, which is not shown here, is the debug-spooler task. This task can only be activated in operation mode DIAGNOSIS. In this case, it has the highest priority of all tasks. It serves as a target-system debug interface for an external debugger. In all other operation modes, the debug-spooler task is disabled.

The design decision to separate the RTE cycle and the RTE monitoring into two separate tasks was made for the following reasons.

The RTE monitoring processes the control commands received via control messages from the service unit. Processing of these commands requires additional computing time, depending on the command and its parameters. For example, programming the EEPROM might take up to 2 ms per Byte. By separating the RTE monitoring and RTE cycle in two tasks, the computing time for the cyclic operation is decoupled from the computing time needed for command processing. Thus deterministic behavior, i.e., nearly constant computing time, can be achieved for the FDG-module processing, and maintaining the required cycle time is guaranteed.

The scheduling sequence of the three tasks is shown in Figure 3.8. The cycle task and the monitoring task share a set of common functions to access commonly used data, for example in the modules ERRORMSG, MODE, FDGIFC, and TRACE. The coordination between the two

tasks is done by protecting critical regions of control-flow with a mutual-exclusion semaphore. The cycle task holds the semaphore from the start of its operating cycle until its end. Thus the monitoring task cannot access (nor change) common data as long as the cycle task is active. The monitoring task holds the semaphore only for short time intervals (< 1 ms), for example for writing a new parameter value, and then immediately releases it. This guarantees that the start of the cycle task can not be delayed inadmissibly by the Monitoring task.

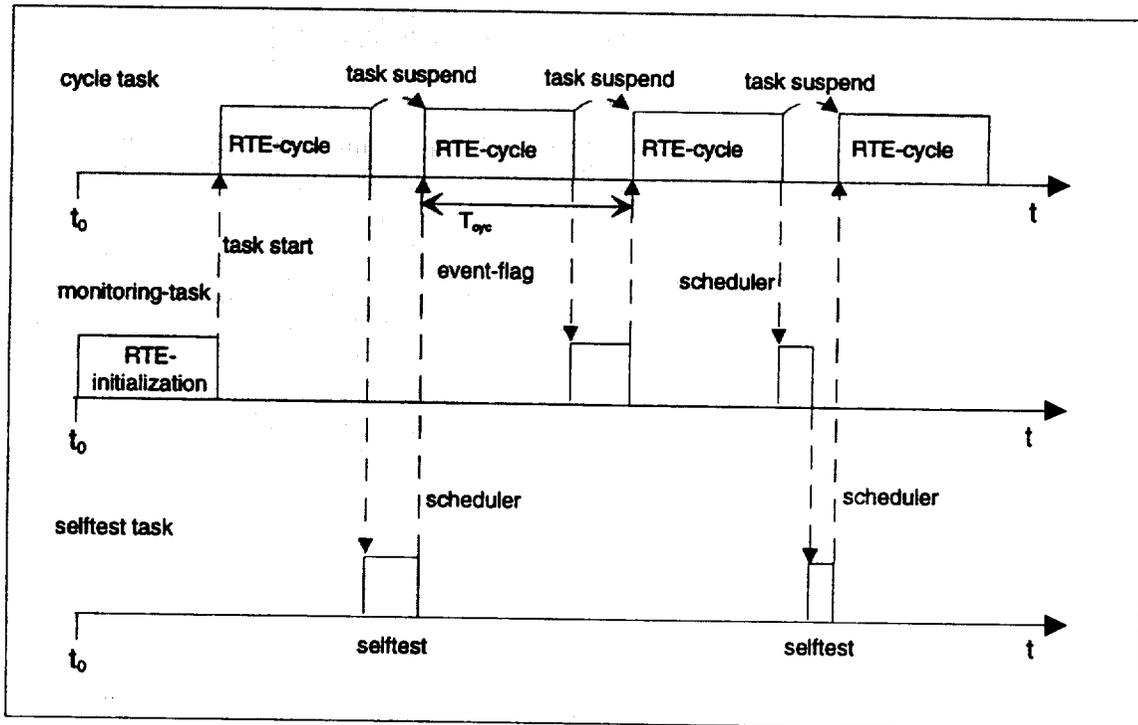


Figure 3.8 Task Scheduling

### 3.1.4.2 Communication

Communication between different processors is done by messages. These message can contain

- signals from FDG-modules (data messages),
- control commands from the service unit (control messages), or
- error messages, trace data, or command responses to the service unit (signaling messages).

To achieve a deterministic system behavior, all communication is performed strictly cyclically, with the system-wide unique communication cycle. No event-driven communication is used. All messages have an individual, but fixed message length. Thus communication loads are constant under all circumstances.

The communication protocols used for sending messages do not use acknowledges by the receiver. Thus it is deterministically excluded that the receiver of a message can have any influence on the sender's operation.

Different communication media are supported by the communication software:

- Processing modules in the same rack communicate directly via backplane bus using shared dual-ported communication RAM (DPRAM).
- Processors in different racks communicate via Ethernet or Profibus LAN. For LAN communication, dedicated communication processors are used, which handle all tasks related to the specific LAN protocol. These communication processors work autonomously, so that the processing modules processing the FDG-modules are not burdened with LAN communication. Messages between the processing modules and communication processors are exchanged using shared DPRAM.

Communication channels via shared DPRAM use a handshake protocol: the sender can only send into the channel when it is empty, and the receiver can only read from the channel when it is filled.

#### 3.1.4.3 Configuration

A fixed operating system software configuration is used for all processing modules. An application specific configuration is neither needed nor even possible.

The platform software (Runtime Environment and its sub-modules) must be configured individually for each processing module. This is done by a RTE configuration module, which is generated by an automatic code generator, based on hardware and software specification in the specification database. The configuration module defines (among other things):

- a unique processor ID,
- the operating and communication cycle times,
- the interface to the FDG-modules to be processed,
- configuration data for the I/O drivers, and
- the complete list of all messages and communication channels.

The application software (FDG- and FD-modules) is completely generated by an automatic code generator, so no additional configuration is needed. During linking of the target system software, the FB-modules used by the specific FD-modules are taken from the FB-module library, so that only FB-modules actually used are included in the target system code.

#### 3.1.5 Code Architecture

Figure 3.9 shows the code architecture of a TELEPERM XS application. This structure is automatically generated by the code generators, the first time code is generated from a specification database. The code architecture of the prefabricated software modules of the operating system and platform software is not shown here.

The code architecture reflects the FDG- and FD-module structure and the individual processing modules of the system. Starting from the directory spool, a subdirectory *<db>* is created, where *<db>* identifies the name of the project specific specification database. Within *<db>*, the subdirectories *fdg* and *rte* are created.

The subdirectory *fdg* contains a subdirectory *fdg\_<fdg\_id>* for each FDG-module, where *<fdg\_id>* is an unique FDG-module identification number. In every FDG-directory, three parallel subdirectories *c*, *h*, and *obj* are created. Subdirectory *c* contains the generated source-code files of the FDG-module (*fdg\_<fdg\_id>.c*) and all of its FD-modules (*fd\_<fd\_id>.c*). Each FDG-module and each FD-module is implemented in one source code file. Subdirectory *h* contains the header files with the external interface definitions of the FDG-module itself and its FD-modules (*fdg\_<fdg\_id>.h* and *fd\_<fd\_id>.h*). The subdirectory *obj* contains an automatically generated makefile for compiling and linking the FDG-module. After compilation and linking, this directory also contains the object files of the FDG-module and its FD-modules, and the FDG-module prelink file *fdg\_<fdg\_id>.plk*, which contains the FDG-module and all of its FD-modules, as well as compiler and linker listing files. The subdirectory *fdg* also contains a subdirectory *tele*, which contains header files with data message datatype definitions for all data messages used in the system. For each data message, its own header file *t\_<tele\_id>* is generated, where *<tele\_id>* is a unique message identification number. This header file is used by the sending and receiving FDG-modules.

The subdirectory *rte* contains a subdirectory *<ve\_id>* for each processing module of the system, where *<ve\_id>* is an unique processing module identification number. In every subdirectory, two parallel subdirectories *c* and *obj* are created. Subdirectory *c* contains the generated RTE configuration module file *augen.c* for this processing module. The subdirectory *obj* contains an automatically generated make-file for compiling the RTE configuration module and linking it with the operating system software and platform software prelinks and the associated FDG-module prelinks of this processing module. The resulting link file is then located to absolute addresses to get a target system loadable file. After this process, the *obj* directory contains the RTE configuration module's object file *augen.obj*, the resulting link file *ve\_<ve\_id>.lnk* and the target system loadable file *ve\_<ve\_id>.mic*, as well as the listing files generated by compiler, linker and locator.

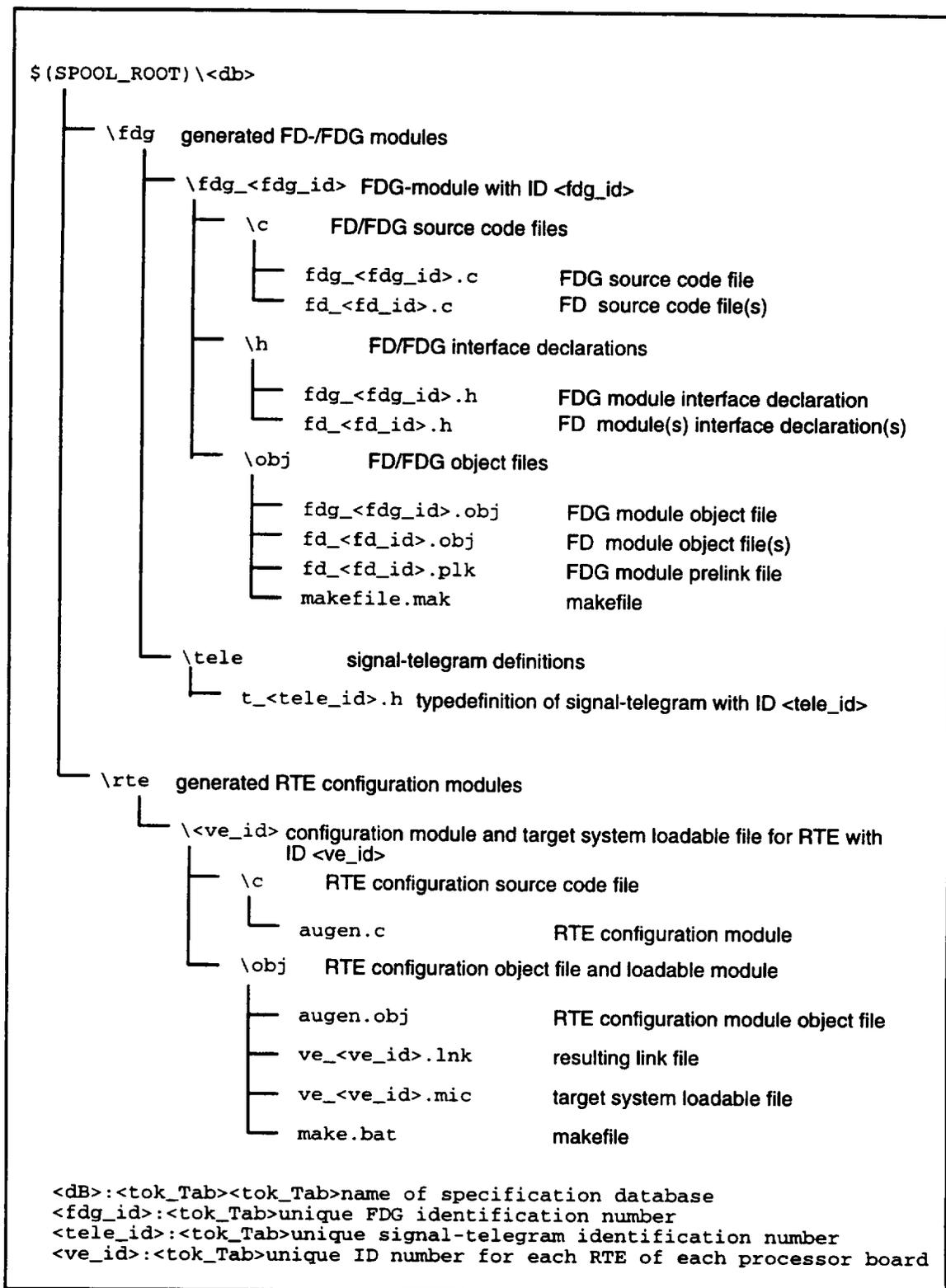


Figure 3.9 Code Architecture

### 3.1.6 Design and Implementation

A number of in general useful techniques have been used to design and implement the software architecture. These techniques are summarized below:

- To support traceability, the FDG- and FD-modules data structures were carefully designed, in order to keep the data of one FD-module in a contiguous memory area. Data of the same type (for example external signals, internal signals, parameters, state variables) are stored in individual, contiguous memory areas.
- Interaction between subsystems that have different architectural styles or use different interaction mechanisms is done through a virtual machine layer or mediators. For example, TELEPERM XS uses the Runtime Environment as a virtual machine that executes function diagrams.
- Code generators along with standardized code structures are used to translate application-specific notation to source code in the selected programming language. Function diagrams in TELEPERM XS are translated into code using standard code structures and a library of function block modules.
- Strict naming conventions were used when implementing the software. Symbol names for functions, variables and constants reflect the scope (global, global within a software subsystem, global within module, or local within a function), the software subsystem defining the symbol, its data type and a part to characterize the symbol's meaning.
- Implementation was done in strict ANSI C programming language. No generic data types like *int* or *char* were used in the implementation. All software modules use a set of basic data types, defined in a central header file. This supports code uniformity and improves portability.
- A lint-like tool was used to carefully check the source code of each module for syntactical errors or suspicious constructs. This tool was used before compiling the source for the first time and before any developer tests were done. This tool proved to be very useful, because errors or weak points in the code were detected very early and thus dramatically reduced the debug time as compared to standard debugging techniques.
- All software modules were unit tested before integration testing, with a branch (c1) coverage of 100 % in most cases. Unit testing was done using a test tool that supports automatic generation of the test-frames, execution of the test-cases, result evaluation, and coverage analysis. The intensive unit testing greatly improved the quality of the resulting code, so that during the system integration testing no major errors were found.

### 3.1.7 Descriptions and Uses

#### 3.1.7.1 Architecture Description Techniques

The conceptual architecture of the I&C functions is documented using a formal, application-specific language that facilitates both documentation and code generation. The graphical language enables process engineers, I&C engineers and plant operators to understand, verify, and diagnose the I&C software in the context of the plant.

The code structure of the automatically generated FD- and FDG-modules was defined using graphical Bacchus-Naur-Form (BNF) notation, resulting in rigorous documentation and facilitating analysis and code generation tools. Furthermore, data-flow diagrams and structure charts were used for analysis and module design documentation. For data dictionaries the use of the Extended-Bacchus-Naur-Form (EBNF) and Jackson diagrams proved to be useful. State-charts and extended finite state machines were used to model protocols and operating mode transitions of the Runtime Environment and also for FB design, where appropriate. Implementation of the source code modules was done using a Nassi-Shneiderman diagram editor. This provided clearness of the code, during editing as well as for review and documentation.

### 3.1.7.2 Analysis

The primary goal of analysis is for the purpose of verification and validation. The quality of the operating system software and platform software (operating system, communication software, Runtime Environment, function block modules, etc.) is assured according to IEC 880 guidelines. In addition to the supplier's internal QA, development of these software modules has been reviewed by independent external assessors. The results of the external qualification is documented in type-test certificates for the software modules.

The quality of the application software (i.e., function diagram modules) is assured by use of prefabricated, type-tested software modules, use of a formal specification method, automatic code generation, and automatic verification tools. The phase model of application software development meets the requirement of the IEC 880.

As described in Figure 3.11, the responsibility for verifying and validating a TELEPERM XS application system is divided among the process engineers and the I&C engineers (in this context the term "process engineers" is used for the people who prepared the system requirement specification, "I&C engineer" is used for the people responsible for system specification and implementation). The process is divided into three main categories, which are described in Figure 3.10.

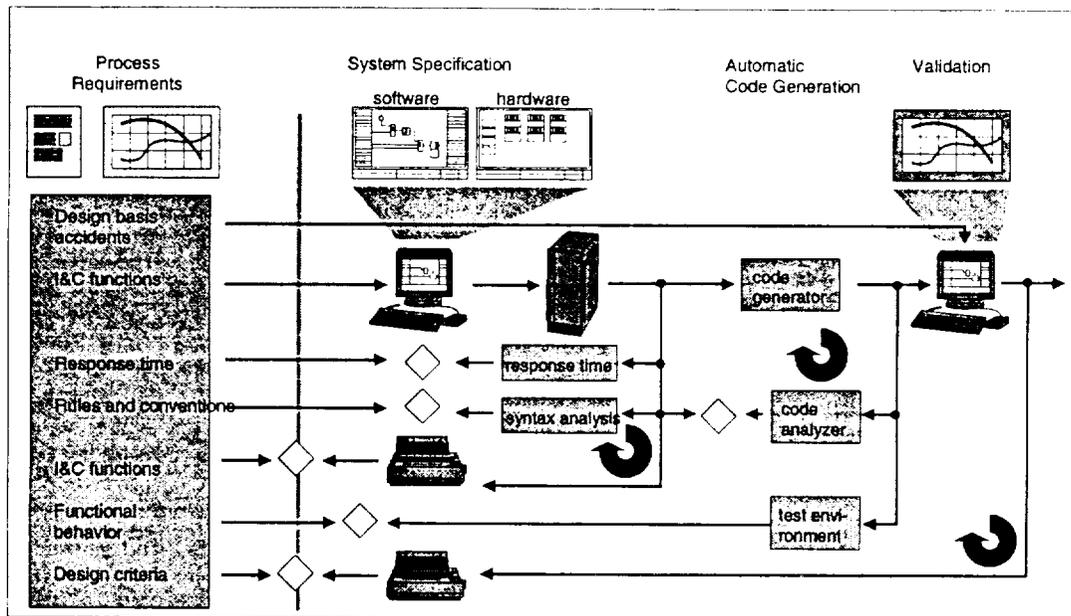


Figure 3.10 TELEPERM XS Software Production

1. Verification of Software Specification

Process engineers verify the function diagrams against the system requirements specification (see Figure 3.11). This includes safety and independence requirements. Additionally the system specification is verified to meet design rules, and conventions, completeness and consistency, by I&C engineers using automated tools.

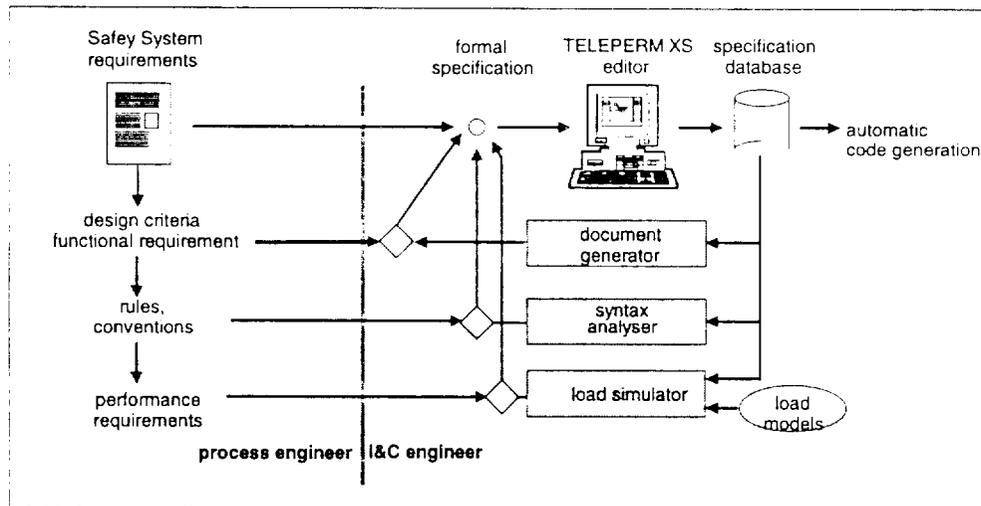
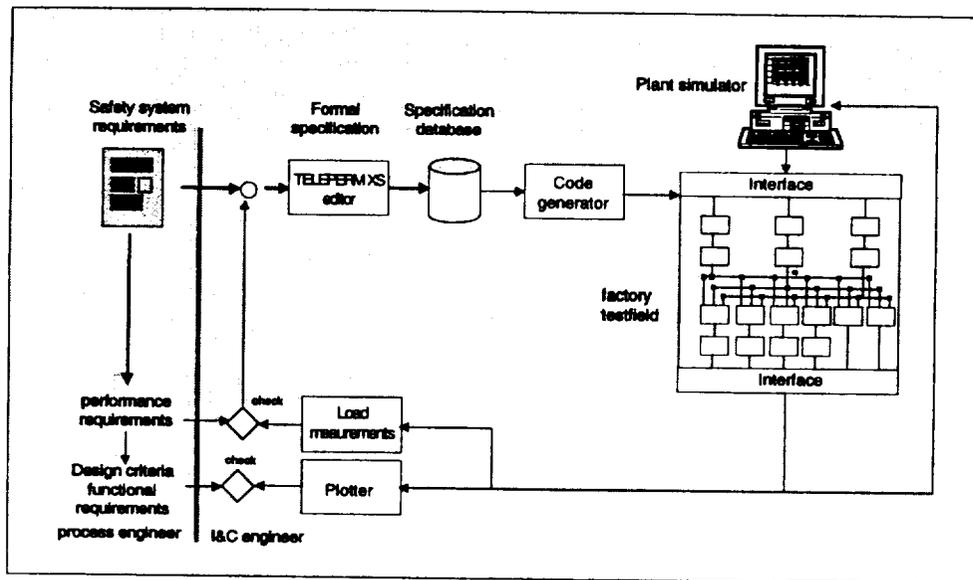


Figure 3.11 Verification of TELEPERM XS Software Specification

2. Verification of Code Generation

The generated code is then validated to meet the functional requirements using a simulator testbed (see Figure 3.12). Either closed loop or open loop simulations can be performed.





**Figure 3.13 Validation of a TELEPERM XS Application System**

### 3.1.7.3 Software Architecture and Software Development

The TELEPERM XS software architecture has an important impact on developing TELEPERM XS application systems. The use of high quality, prefabricated and type-tested software modules and a highly automated specification and code generation toolset allows cost effective generation of application systems. The high degree of reusability of the software components allows the costs for the high quality software development to be distributed over a large number of applications. The highly automated specification and code generation process allows the production of high quality application software at lower costs, as compared to conventional methods.

The specification database with its function diagrams and hardware specification diagrams is also used as forward documentation for a TELEPERM XS application system. It is used not only as software specification documentation, but also during commissioning and plant operation. Since the specification database is the single source of the application software, this documentation is always up to date.

During development of TELEPERM XS, the software architecture was also used to identify the subsystems and tools needed and for assigning work to the staff. The software architecture also helped to clarify the TELEPERM XS concept to managers and external assessors. The individual software modules identified were used as a basis for cost and time schedule planning, both for internal project management and for the external assessors. The software development process was decomposed into processes for each software component. Each software component was developed independently, with its own documentation, reviews, and unit test. Certification of the software by external assessors was also done based on the individual components (both hardware and software). For each individual component a type-test certificate was prepared.

### Testing

Both the module and the execution architectures explicitly address self supervision and testing. The fault-tolerant architecture facilitates on-line testing of any processing module (one or more) in a redundant system. The inputs (outputs) to (from) functional blocks are marked as being either normal, under testing, or faulty. This allows processing modules not under test to selectively ignore these signals, by using special validation or voting function blocks (for example 2nd maximum, 2nd minimum, 2-out-of-4 voting, etc.).

Separating the implementation into different categories of software allows them to be tested individually. Since much of the application software is automatically generated, it may be feasible to support automatic generation of the test cases for FDG-modules and FD-modules based on the test cases for individual FB-modules.

### Maintenance

The fault-tolerant architecture facilitates the on-line maintenance of the TELEPERM XS software. Using a formal language for the conceptual architecture, and the use of structure-preserving implementation techniques allows for on-line diagnosis of function diagrams. It is also possible to reverse-engineer the function diagrams from the code. Their comparison with original function diagrams gives yet another level of quality assurance.

### Adapting the Software Process

According to IEC 880, a standard phase model for the software development should be used, consisting of a software requirements specification phase, a software design phase, and a coding phase. Each phase result must be verified against its inputs. By using the formal graphical specification method described above, a modified phase model was defined for the safety application software. In this modified phase model, the software requirements specification phase and the software design phase are replaced by the formal software specification process using function diagrams. Verification of the software specification is done partly automatically based on rules (syntax conventions) and partly manually by verification with the system engineers. The coding phase is replaced by automatic code generation for the function diagrams. The results of the automatic code generation phase can be automatically verified by applying the inverse code generation rules and comparing the results with the original information stored in the specification database.

The automatically generated FD-modules are completely independent of the target system. They are coded in a portable high-level language. This allows simulation of the functional behavior of the system outside the target system, e.g., in a simulation environment on a workstation. Thus functional correctness of the software specification can be verified rather early in the system development process, before the target system hardware has been ordered and built up.

For executing the function diagrams in the target system, a hardware abstraction layer is used, the Runtime Environment. The Runtime Environment interfaces the function diagrams with the target system (operating system, communication software and system hardware). The Runtime Environment serves for the cyclically processing of the function diagrams, the sending and receiving of data messages, the processing of I/O modules and other services. It also implements protocols for error propagation barriers, system-inherent fault-tolerance

mechanisms, signaling of error conditions and an interface for external monitoring and control via the service unit. This interface is also used for recurrent testing of the safety I&C system.

In contrast to the function diagrams, the Runtime Environment is not automatically generated but hand-coded using standard SA/SD methodology. This had to be done only once, because the Runtime Environment is an universal software component, which can be configured for an application on a specific processing module via a configuration module. The configuration module contains all configuration data needed to adapt the Runtime Environment to the specific processor. The Runtime Environment's configuration data are also automatically generated by a code generator based on the information contained in the specification database. These configuration data can also be verified automatically by another independent tool, which transforms the automatically generated configuration data into a internal representation and compares it with the specification database.

### 3.1.8 Summary

TELEPERM XS has managed to completely separate I&C functionality from the complex protocols related to communication, fault tolerance, monitoring, and on-line testing. The execution architecture is separated from the rest of the software. This approach has made the introduction of digital technology into safety I&C of nuclear power plants both tractable and feasible.

Separation of the conceptual architecture facilitates the design, analysis, verification, reconfiguration, and update of I&C functions by process engineers and I&C engineers. The conceptual architecture is preserved and / or traceable in the code, supporting on-line diagnosis of function diagrams.

Separation of the complex protocols makes them amenable to formal modeling techniques, and simplifies their implementation and verification. These protocols were modeled using state charts and other appropriate notations. Separating out the also simplifies the manual implementation of functional blocks and the Runtime Environment, eliminating the code for these protocols from the application software. It also allows early and intensive unit testing, at a phase when errors can be detected more easily than during integration testing. Simple and standardized implementation techniques facilitate the automatic generation of application code, combining the conceptual architecture, the Runtime Environment, and the protocols.

Finally, separation of the execution architecture facilitates reconfiguration of function diagrams into FPGA modules and their processing module assignments. It also makes it easier to change the operating system software (e.g., operating system and communication) without affecting the application software.

## 3.2 ***Software Development Process Characteristics***

### 3.2.1 Software Type Test

The basic intention of type-tests is to separate out tests and inspections that are independent of a specific application from those that are specific to the safety needs of a particular power plant. Having type-tested components allows to rely on the correspondence of these components with the specification of their functional properties in the data sheet or in the software development documents. Moreover, tests and inspections can be performed independently from and in

advance of their deployment in a plant. Even more important is the fact that the type-testing procedure is executed only once. Each subsequent usage in an I&C system can refer to the type-tests, which have been performed successfully.

By order of the Bavarian licensing authority (BStMLU), GRS was contracted to perform the software type-tests as a third party assessment. GRS subcontracted ISTec and TÜV-Nord, in order to broaden the base of expert knowledge and to increase personnel. The assessment began in 1992 and was finished in 1997.

### 3.2.1.1 Rules and Standards

Since there is no KTA standard defining type-tests of software, the type-tests of the TELEPERM XS software components were performed in accordance with KTA-Standard 3503. From this standard came the principles of type-tests as well as the overall structure of the test activities. These were applied for the following points:

- separation in the theoretical and practical tests,
- institutions to be involved in type-tests,
- roles of these institutions for type-tests, and
- test documentation of type-tests.

The content of the theoretical and practical test is defined by the software standard DIN IEC 880, which was the most important base for the software development as well as for the type-tests.

KTA standards also require that the present state-of-the-art be taken into account during the qualification. In addition to KTA 1401, which defines criteria for quality assurances systems, the following software standards were applied and verified:

- ISO 9000-3, Management for Quality and Requirements of Quality Assurance,
- IEEE 830, Software Requirement Specifications,
- IEEE 828, Software Configuration Management Plan,
- IEEE 1012, Software Verification and Validation Plans,
- IEEE 829, Software Test Documentation,
- IEEE 1008, Software Unit Testing,
- IEEE1028, Software Reviews and Audits, and
- ANSI/ANS-10.4-, Verification and Validation of Scientific / Engineering Programs for the Nuclear Industry.

### Correspondence to U.S. Requirements

Among the standards referenced in the Standard Review Plan and the Branch Technical Positions, the requirements of IEEE 7-4.3.2 have some importance for the software type-tests.

IEEE 7-4.3.2 addresses specific requirements concerning the software development. Most of them are given by reference to the standards ASME NQA, IEEE 730, IEEE 828, IEEE 1012, and IEC 880. The requirements of ASME NQA are covered by KTA 1401 and the requirements of IEEE 730 are covered by those of ISO 9000-3. All other standards were directly applied in the development and evaluated in the type-tests.

### 3.2.1.2 Qualification Process

At the beginning of the TELEPERM XS software development, a set of project-specific engineering procedures was elaborated in line with the concepts described in the reports submitted for the concept review and in line with the underlying standards. These engineering procedures covered the whole software life cycle and defined the software development process with all verification and validation activities elaborated in detail. They supplement the Quality Assurance described in Siemens' QMH12E. All qualified software components had been developed according to these engineering procedures. Most components used in the online system were implemented using a subset of the C language of ANSI-C, while tools were implemented using C++. ANSI-Standard X3.159-1989 was used in together with Engineering Procedure FAW 2.1, "Guidelines for Programming." In a few cases, assembly language was used due to speed or hardware restrictions.

The engineering procedure 1.1 "Phase model" defined the individual phases that had to be applied in the development of each individual component and the associated verification and validation activities. According to this phase model, four refinement steps had to be applied:

- requirements specification,
- design specification,
- design description, and
- implementation description,

and two steps had to be applied for the component test:

- test specification, and the
- tests with test documentation.

The results of each phase had to be documented in a development report. At the end of the implementation phase, the complete source code and the complete object code had to be stored on a magnetic tape. The phase results had to be verified against each other in two different verification steps. One verification step focused on formal aspects. During this step, the consistency and the completeness of the phase results were verified. The results of this verification step were documented by check-lists. In a second verification step, the adequacy of the technical solution and the interfaces to other software components were verified by critical reviews. The results of the second, more extensive verification step were documented in review reports. The engineering procedure 1.1 implemented the most important requirements from the Standards

- DIN IEC 880,
- IEEE 1012, and

## TELEPERM XS: A Digital Reactor Protection System

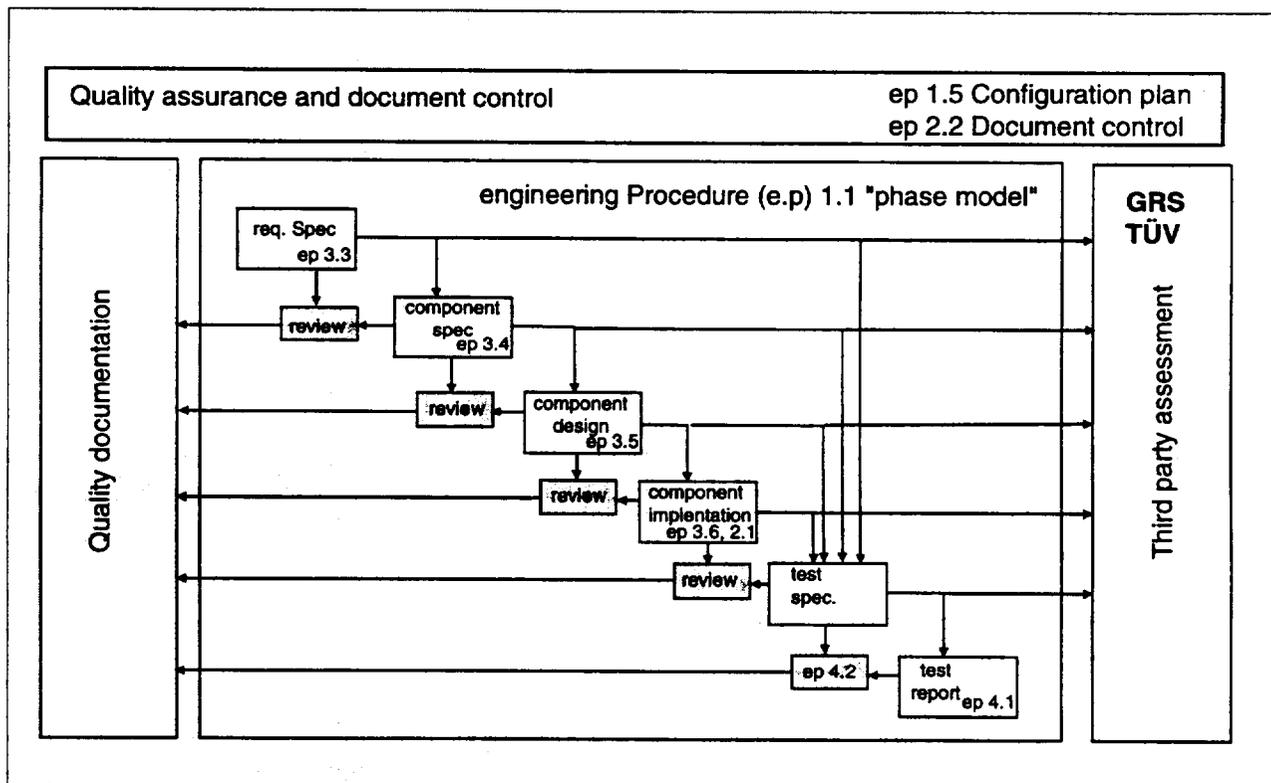
- ANSI/ANS-10.4- Verification and Validation of Scientific / Engineering Programs for the Nuclear Industry,

as well as some important requirement from

- ISO 9000-3.

For each of the development phases defined by the engineering procedure 1.1, a specific engineering procedure was elaborated, thus fixing the content and the structure of the documentation used to describe the phase results. The following engineering procedures were elaborated and applied:

- Engineering Procedure 3.3 "Inhaltsgliederung Lastenheft" (Requirement specification),
- Engineering Procedure 3.4 "Inhaltsgliederung Pflichtenheft" (Design specification),
- Engineering Procedure 3.5 "Inhaltsgliederung Designunterlage" (Design description),
- Engineering Procedure 3.6 "Inhaltsgliederung Implementierungsunterlage" (Implementation Description),
- Engineering Procedure 4.1 "Tests," and
- Engineering Procedure 4.2 "Reviews."



**Figure 3.14 Phase Model and Associated Engineering Procedures**

Figure 3.14 gives an overview of the phase model and the associated engineering procedures. The engineering procedure 3.3 implemented the most important requirements from the Standard

- IEEE 830

and the engineering procedures 4.1 implemented the most important requirements from the Standards

- IEEE 829, and
- IEEE 1008.

Additional engineering procedures were elaborated in order to:

- 1.0 define the coding rules (engineering procedure 2.1a),
- 2.0 control modification and manage configurations (engineering procedure 1.5),
- 3.0 perform and document critical reviews (engineering procedure 4.2), and
- 4.0 cover security aspects during the development (engineering procedure 1.7).

The software development of each individual TELEPERM XS component followed these engineering procedures. For the development of all components, the same compiler and linker was used with a restricted set of options. This was done in order to qualify the compiler and the linker during the development by good service records, because each piece of software was extensively tested to meet the IEC 880 requirements for c0 and c1 test coverage.

The development documentation together with the source code and the object code was then submitted to GRS for type-tests. The main issues in the type-tests of TELEPERM XS software were:

- evaluation of the software development process according to the software life-cycle, and
- evaluation of conformity to the coding recommendations of IEC 880, consideration of the state of art.

Following the principles set forth in KTA 3503, the software type-tests also contained a theoretical part and a practical part. In contrast to the theoretical test of hardware, where the documents describe the electrical module as a ready-made, fully developed product, the theoretical test of software is completely development oriented. Thus this step in the software type-tests was much more complex than the theoretical test for hardware. It ensured that the software development followed the agreed-upon development steps, and that the requirements of the underlying standards were met. Moreover, the independent verification of the development steps that were performed during the theoretical part assured the completeness, consistency, and correctness of the software development.

#### 3.2.1.2.1 Theoretical Part

The activities in the theoretical part of the software type-tests consisted of three separate verification steps. These steps were applied on the four software development documents, which are refinements of each other.

The three verification steps focused on

- form,
- content, and
- consistency of the respective documents.

#### The form check

The form check was an evaluation of whether the document contained all the elements required for the associated level of refinement. Correspondence to the structure and to the list of contents defined by the underlying engineering procedures was verified.

#### The content check

The content check was an evaluation of:

- functionality of the software modules,
- development process performed by the manufacturer,
- interfaces,
- accordance to the relevant standards, and
- description methods used (formal methods if possible).

An important issue for this step was to evaluate whether the software recommendations of IEC 880 were applied. These recommendations deal with:

- Code structures:
  - suitable modularization,
  - well structured,
  - simple control flow structures between the modules,
  - restriction to safe programming constructs and no programming tricks,
  - no recursive structures,
  - easy to understand due to sufficient, precise, and structured comments, and
  - generated code also contains comments.
- Data structures:
  - systematically and hierarchically structured,
  - correspondence to code structure, and
  - reduction of memory and computing time.
- Interfaces:
  - suitable interfaces,
  - clear cut,

- few parameters, and
- few global variables, clearly marked by specific naming conventions.
- Naming conventions:
  - formally defined,
  - consistently used, and
  - supporting the readability and understandability of the code.
- Control mechanisms:
  - version management and version check of on-line software during start up of the system,
  - self test program cyclically checking the integrity at runtime,
  - error-handling and return-codes of software components, and
  - management and checks of message-version, message-status and signal status.
- Comments.

In addition, static analyses were performed for selected components, both at the source code level and at the binary code level. This was done to verify certain features of the source code, and to verify compiler and linker features for error prone constructions.

#### The consistency Check

The consistency check was an evaluation of:

- the consistency of the document itself,
- the consistency between a document and its corresponding document in the preceding level of refinement, and
- the consistency of interfaces to other components.

These activities make it clear that software type-tests place great importance on the theoretical test, which verifies and evaluates the constructive elements of the software development process. In comparison to this, the pure functionality testing, performed during the practical part of software type-tests, can be interpreted as a comprehensive acknowledgment of the theoretical part of type-tests.

#### 3.2.1.2.2 Practical Part

In comparison to hardware, the practical test of software was much more compact, since it had only to test the functionality with respect to input-output behavior. All the mechanical, electrical and climatic aspects were irrelevant for the software.

In accordance with the essential elements of type-tests (distribution of activities), the practical part was performed in the following steps:

- the manufacturer provided a test specification of the software components,

- the third party expert confirmed this test specification by checking it and asking for additional or different tests where appropriate,
- the manufacturer carried out the tests in line with the agreed-upon test specification, and summarized the results in a test report, and
- on the basis of this test report, the third party expert stated whether the software has successfully passed the practical part of the type-tests.

The most important evaluation criteria were:

- the completeness of the test (were all relevant features of the component covered by the test),
- the suitability of the test (was the selected test method adequate to demonstrate the features), and
- the test coverage.

In accordance with Appendix E of IEC 880 and depending on the software to be tested, a subset of the following test methods was applied to verify the features:

- statistical tests,
- black-box test,
- white-box test,
- path testing,
- coverage testing,
- execution time testing, and
- boundary test.

The software tests were primarily executed in a test-bay and were partially automated. Where it was advantageous, the tests were performed in a more typical software environment.

### 3.2.1.3 Scope of Components

Since the functionality of a TELEPERM XS safety system is given by the software running on the CPUs, the software is essential for the understanding of TELEPERM XS. The software consists of two main parts: the application software and the operating system.

The operating system is small and static. Its kernel has a size of approximately 1 KByte. It is identical for all the CPUs of an individual TELEPERM XS system. It is also independent of a specific application. The operating system is the bridge between the hardware and the application software.

The application software itself consists of three main modules:

- The runtime environment, which controls the execution of the code that implements the I&C functions.

- The function diagram group modules (FDG-modules), consisting of sets of function diagram modules (FD-modules). An FDG-module groups together all the FD-modules that are executed on the same processor in the same cyclic frequency. The FD/FDG-modules represent the pure functionality of the I&C functions. They are entirely application specific.
- The function block modules (FB-modules), which are basic software function primitives of a library. This library consists of the implementations of about 120 common I&C function elements.

These three modules of the application software are organized in a hierarchical manner. The runtime environment controls the execution of the FDG-modules by calling them cyclically in a fixed frequency. It triggers the FDG-modules to:

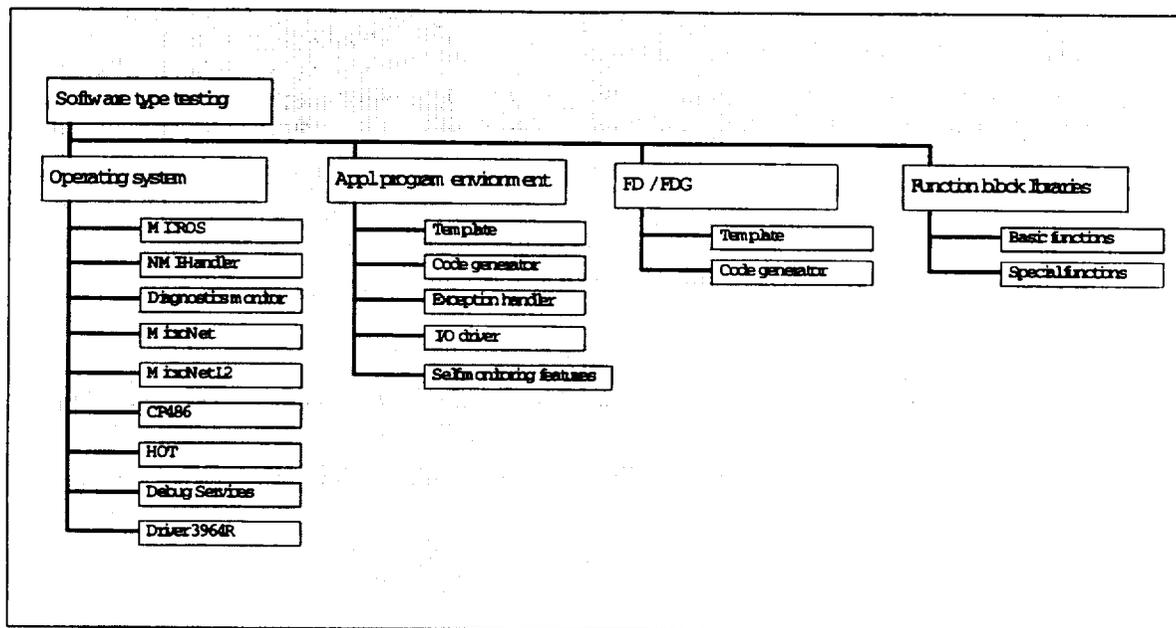
- receive groups of signals (messages) from the runtime environment to process these signals, and
- forward the resulting groups of signals (messages) to the runtime environment. The runtime environment sends these signals to, or receives them from other computers or devices via the communication processors.

Additional tasks of the runtime environment are exception handling and support of testing and maintenance. The runtime environment makes use of the services of the operating system.

The FD- and FDG-modules' task is to provide the functionality of the I&C functions that control the technical process.

Function block modules are used to compose the FD- and FDG-modules. They are the elementary software components of the application software and provide the basic logic and arithmetic functions (e.g., AND, OR, 2 out of 3, limit monitor, etc.). Each block has a graphical representation in the function diagrams. The function blocks can be thought of as the vocabulary of a formal specification language.

Type tests were performed for all reusable TELEPERM XS software components that have a high impact on safety. Thus in addition to type-testing the on-line components of TELEPERM XS, the tools used to automatically generate parts of the on-line software are type-tested. Figure 3.15 gives an overview of the type-tested software components.



**Figure 3.15 Type Tested Software Components of TELEPERM XS**

The figure includes all reusable software components of the on-line system:

- Operating system  
with its components:
  - MICROS.  
MICROS is the real-time operating system kernel of TELEPERM XS. It is small and static. In this context, the term static means that all the operating system objects are defined on startup and cannot be changed during run time.
  - MicroNET.  
MicroNET is responsible for the communication between different function computers, as well as between the communication processors and function computers within the same subrack.
  - MicroNET-L2 is an extension of the communication services that supports L2 communication.
  - CP486.  
Protocol handler implemented in the SCP1 communication processor, responsible for transmitting and receiving H1 messages.
  - HOT.  
The system component HOT (hardware organization tool) is responsible for the parameterization of the operating system on computer startup. HOT configures the communication memories and detects whether modules are assigned to the slots in a subrack. It also detects the module types.
  - NMI-Handler.  
The NMI handler, in cooperation with the exception handler, is responsible for handling abnormal conditions. In many cases, hardware failures are signaled by means of

- interrupts. Bus accesses to I/O modules are monitored for tolerable response times (time-outs) in this way, for example. If these times are exceeded, it is concluded that a failure has occurred and this is signaled by means of an interrupt. Both handlers are responsible for initiating the required measures in such cases. One important measure involves signaling the failure and returning the computer to the defined state (e.g., disabling of all outputs).
- **Debug-Services.**  
Services to support diagnostic services are mainly used during the system development and integration.
  - **Diagnostics monitor (monitor to support diagnostics).**  
The diagnostics monitor is a low-level debugging tool. If the computer fails, for example, it is possible to branch to the diagnostics monitor to analyze causes of failure that are difficult to diagnose after the computer has been returned to the defined state (outputs disabled).
  - **Driver 3964 R.**  
The 3964R driver constitutes the interface to the local V.24 interface which is used for loading programs and for the output of debug information.
  - **Runtime environment**  
with its components:
    - **Template of the runtime environment.**  
The application program environment is an intermediate layer between the operating system and the function diagram group modules. It permits the coordinated execution of up to two groups of function diagram modules with different processing cycles. A function diagram group module comprises all the function diagram modules which are computed on a common SVE1 in the same processing cycle. A function diagram module is the software implementation of the functionality specified by a function diagram and therefore constitutes the actual application function.  
The application program environment also implements the system features specific to safety I&C, such as services for communication between application functions, failure signaling mechanisms, interfaces for performing function, tests, etc. The application program environment is a template in the following sense: it is a fixed program created in the conventional manner, and requires only configuration data to tailor it to the specific application. This configuration data is produced with the aid of code generators.
    - **Exception Handler (high level failure handling).**
    - **I/O Drivers for peripheral devices.**  
The I/O drivers are responsible for data transfer to and from the analog and binary I/O modules.
    - **self-monitoring.**  
The purpose of self-monitoring is the detection of hidden failures on the processing computer. When failures are detected, they are signaled to the exception handler, which then takes appropriate measures.
  - **Tools** used to generate the specific application software, consisting of:

- Generator for runtime environment  
This generator generates a configuration data set for the runtime environment.
- Generator for FD/FDG (generates the application software).  
Function diagram modules and function diagram group modules implement the actual application functions. This generator generates the function diagram modules from the software specification, and relies on a library of function block modules for this purpose. From the point of view of programming, the function diagram modules are implemented as a linear sequence of function block modules that are linked by suitable data structures.
- Template for FD/FDG-modules  
This template defines only the structure of the function diagram modules software; it contains no software modules.
- Libraries of function block modules used by the FD/FDG generator to build the application software:
  - library of function block modules.  
Function block modules are the elementary components of the application functions. They are made available as a function block library and are called up in the function diagram modules.

Other tools of SPACE like the graphic editor or the test and verification tools are not type-tested because they have much less of an impact on the software in the safety system and therefore have less safety relevance.

Because of the particular nature of tools, the type-tests of the tools (generator for the runtime environment, FD/FDG generator) did not need to be as stringently held to the requirements of IEC 880. Certain requirements of IEC 880 that are important for online software are not relevant for tools and were therefore not applied to the tools.

#### 3.2.1.4 Submitted Documents and Results

The software type-tests were based on an agreed-upon phase model. In addition to the engineering procedures, all the development and test documentation was submitted to GRS for evaluation:

##### Reports submitted for the software type-tests

- Engineering procedures:
  - 1.1, Phasenmodell der Entwicklung Digitale SILT
  - 1.4 , Hardware QS-Plan
  - 1.5 , Konfigurationsmanagement-Plan
  - 1.7, Informationssicherheit
  - 2.1a, Programmierrichtlinien
  - 2.2a, Dokumentationsrichtlinien
  - 3.3, Inhaltsgliederung der Lastenhefte für SW- und HW Komponenten

- 3.4, Inhaltsgliederung der Pflichtenhefte für SW-Komponenten
  - 3.5, Inhaltsgliederung der Designunterlage für SW-Komponenten
  - 3.6a, Inhaltsgliederung der Implementierungsunterlagen für SW-Komponenten
  - 4.1, Tests
  - 4.2, Reviews.
- **Development and test documentation:**  
For each of the software components listed above, the entire development and test documents together with the source and object code were submitted to GRS for evaluation. Questions that arose during the evaluation and their corresponding answers were documented in a reviewable form.

### Evaluation Reports and Certificates

For each of the type-tested software components listed above, a certificate and evaluation report was prepared by ISTec/TÜV-Nord. Each certificate identifies the associated component with all development and test documents, and contains a short evaluation of whether or not the component has an acceptable quality for safety applications in nuclear power plants. It also contains a summary of the most important evaluation criteria. In addition, the evaluation report contains a more detailed description of the evaluation process and of all findings, including the correspondence between the manufacturer and the assessor written during the evaluation process. All components were accepted as having adequate quality for safety applications in nuclear power plants. The certificate of the CP486 contains a restriction that this component should be first applied in less critical applications to gain more service experience. The reason for this restriction was that the component contains a preexisting software module not specifically developed according to the above-mentioned standards, and it was not possible to evaluate the module thoroughly. At this time, the CP486 has gained some ten years of good service in the NPP "Unterweser" and it is expected that soon it will get the certificate without any restrictions.

In light of the correspondence between the KTA and IEEE standards described earlier, the evaluation basis covers the requirements of the U.S. standards.

The results of the software type-tests demonstrate that

- the software was developed, verified and validated in line with the relevant standards, and
- that each development step was verified additionally by a third party expert.

Furthermore they ensure that the software was tested using a test specification that was checked and accepted by an third party expert. For the development itself, the third party experts approved the quality aspects regarding:

- development documentation,
- code structures,
- data structures,
- interfaces,

- naming conventions,
- control mechanisms,
- standards, and
- specific aspects.

#### Summary of Deviations from IEC 880

Deviation from the underlying Standards are documented and assessed in the evaluation reports.

The following deviations were agreed upon with ISTec and TÜV-Nord at the beginning of the development:

- Use of the programming language C, although it does not have completely specified semantics.  
Potential problems have been ruled out by using programming guidelines to restrict the language.
- Absence of programming guidelines for Assembler code.  
Assembler coding was applied only to tasks where a high level language was not possible. Thus a limited number of lines of code were implemented in Assembler. They are clearly structured and easy to verify.

The following list contains the non-conformities to standard DIN IEC 880 that were agreed upon during the third party assessment.

SW Component	Agreed-upon deviations
Functional Block modules	Equivalence technique is used in two cases. This structuring method appears as the most clear in this place.
Functional Diagram Groups (FDG) template	Pointer function calls to the functional blocks. This results from the engineering procedure applied.
FDG Code Generator	None.
Runtime environment template	None.
Generator for the configuration data of the runtime environment	None.
Exception Handler	None.
I/O Driver	None.
Self-Supervision	None.
Operating System	
- MICROS	Use of system interrupts for timer and failure handling. This results from the engineering procedure applied.
- NMI Handler	None.
- Diagnostic Monitor	None.
- MicroNET	Use of interrupts for failure handling. This results from the engineering procedure applied.
- CP486 (Com Processor)	None.
- HOT (HW Organization Tool)	None.
- Debug Services	None.
- 3964R Driver	None.
- MicroNET L2	None.

**Figure 3.16 List of Agreed-Upon Non-Conformities to Standard DIN IEC 880**

### 3.2.2 Integration and System Test

The purpose of the integration and system test was to demonstrate and evaluate specific system features important to safety that could not be demonstrated at the component level. In accordance with the principles of type-tests, these features should be evaluated only once, and should not have to be re-evaluated for each safety application.

The integration and system test was performed by GRS as a third-party assessment, under orders of the Bavarian licensing authority. In order to gain the benefit of existing experience from the hardware and software type-tests, GRS subcontracted ISTec and TÜV-Nord. The total work was shared between both parties at the level of test goals. This means that the

development of one set of test goals was assessed by ISTec and another set was assessed by TÜV-Nord. The test was done using the test field with the original hardware and software of the first large TELEPERM XS application. This application was the limitation and control system for the nuclear power plant in Unterweser. It has a fourfold redundant architecture for the limitations as well as for the closed loop controls, and consists in total of 20 cabinets. This application was in the test field from November 1996 until May 1997. The integration and system test was performed from December 1996 until February 1997 based on a test specification previously agreed upon by all parties involved.

### 3.2.2.1 Rules and Standards

The integration and system test should be thought of as an extension of the component type-tests. Its purpose is to demonstrate system features that cannot be covered at the component level. Therefore the underlying standards were partly the same as those applied for component type-tests. Special attention was paid to:

- KTA 3503 with respect to the role of the different parties,
- KTA 3506 and IEEE 1008 which gives specific requirements on unit and system tests, and
- IEEE 829 with respect to the test documentation.

All system tests were performed in the test field under normal environmental conditions. This is because standards applied for the hardware component type-tests dealing with environmental conditions or electromagnetic interference have no relevance for the system and integration test.

### 3.2.2.2 Goals of the Integration and System Tests

The goal of the integration and system test was to verify the system features of TELEPERM XS that are relevant to reliability, using a representative system architecture as a basis. Such features include both the correct interaction of the individually qualified hardware and software components, and the typical features of safety I&C systems such as failure detection or fail-safe behavior.

The test objects were the safety-relevant system features of the integrated computer system composed of hardware and system software. The application software on which the integration test was based served as test equipment (the application functions and their input/output signals were used for creating the test cases). The software specification was the result of the process engineering requirements for the controls and limitation systems in the Unterweser nuclear power plant. Possible process engineering errors in the software specification had no effect on the results of the integration test. The evaluation of the functional response of the application function was therefore based on the formal software specification and not on the process engineering requirements.

The test goals were subdivided into two groups, i.e.:

- test goals concerning the correct interaction of all components, and
- test goals concerning TELEPERM XS system properties.

### Correct Interaction of all Components

The first group of test goals addressed integration tests that demonstrate the correct interaction of the various components. This group primarily involved interface tests that could not be performed within the framework of the component tests, or that could only be performed incompletely. The need for the tests arose from open points during component development or the software type-tests. The runtime environment was a particular focus: it is the central software component of the TELEPERM XS system, and has interfaces to system services, the application function, and corresponding startup environments. During the component tests, it was not possible to completely test the runtime environment's interface response. The interfaces of the startup environment therefore play a central role in the integration test.

The following interfaces of the runtime environment were verified in the scope of the integration test:

- to the Hardware of the function computer (front-plate LEDs and serial interface X4.2),
- to the operating system components (MICROS, MicroNET etc.),
- to the exception handler (management of and reaction to an event/exception),
- to the I/O drivers (reading in of input signals and issuing of output signals from and to the I/O modules),
- to the function diagram group (FDG) modules (calculation, transfer and acceptance of values),
- to the service unit (operator command interface), and
- to the code generator (configuration files) for configuration of the run-time environment.

During the software type-tests, two additional interfaces were identified as needing verification on the system level. These were the interfaces between the exception handler and the operating system, and interfaces between the self tests modules and the hardware.

The following interfaces of the exception handler were verified in the scope of the integration test:

- to the NMI handler,
- to the Diagnostics monitor,
- to the I/O drivers,
- to the hardware of the function computer (front-plate LEDs and serial interface X4.1),
- to the cabinet alarm system (via the signal for disable command output (BASP), and
- to the self tests modules.

The interfaces between the hardware components and the self test modules were also verified within the scope of the integration test.

### Demonstration of Safety Related System Features

The second group of test goals consisted of system tests to verify the relevant system properties of TELEPERM XS. The relevant system properties were described at the very beginning in terms of concepts and were evaluated during the concept review. Some of these properties could be evaluated during the component type-tests, but others could only be evaluated at the system level.

The following system properties had to be verified during the system test:

- Correct execution of application function.  
The functional response of the application function defined by the software specification must be fulfilled by the integrated system composed of hardware and software.
- Deterministic response.  
The consequence of cyclical system processing is that the task processing time and the communication load are set during configuring and are not affected by demands for system response. The system loads and the run-time response are therefore predictable and do not vary over time.
- Non-interaction of the application functions.  
Independent application functions must not affect each other even if they use common resources in time-sharing mode. This requirement is independent of the particular application function and independent of the particular hardware architecture.
- Independence of the system response from the process.  
One of the fundamental properties of TELEPERM XS is that the system response, i.e., all internal system operations, is completely independent of what is happening in the process. A direct consequence of this is the fact that the system response cannot be affected by input signals. Internal operations which can be supervised by means of monitoring mechanisms and special instrumentation are a special characteristic of the system response.
- Response time of application function.  
Worst-case reaction times are calculated for the system relative to the system design (workload, hardware architecture, coordination of different computers) within the framework of system design. The reaction times calculated in this way are used to verify the system design. They must be observed independently of demands for system response.
- Testability, maintainability and diagnostic capability.  
Testing and maintenance tasks require that individual computers be shut down for testing or repair and that they be subsequently reintegrated into the network without affecting the application functions. A prerequisite for this, of course, is that the selected hardware architecture features the appropriate redundancy.

In addition to the correct system behavior under normal conditions, the failure behavior plays an important role in safety related systems. Therefore some important system properties addressed the failure behavior and confinement areas of TELEPERM XS.

The following properties are of significant relevance and were verified in the scope of the system test:

- Failure detection and failure signaling by the appropriate routines.  
Failures of single components must be detected and signaled by the responsible system software routines. Downstream routines must mask the effects of these failures in accordance with specifications.
- Effect of failure barriers and failure tolerance features.  
Each individual failure must be covered by the redundant architecture of the system without any influence on the safety function. The failure barriers implemented on the application level must be able to assure that each failure may only influence safety functions within the designed confinement area.
- Failure effects and fail-safe behavior.  
The effects of any failure at the boundaries of a confinement area must comply with the specified behavior. Any failure in the interface to the switchgear that cannot be covered by the redundant system architecture has to lead to a fail-safe behavior of this interface.
- Actuation of the cabinet alarm system.  
Any failure within the system has to be signaled additionally by the cabinet alarm system.

#### 3.2.2.3 Qualification Process

Based on the above-listed test goals, a first draft of a test specification was elaborated by the manufacturer. This first draft contained the manufacturer's ideas about the test methods and test cases that should be applied to demonstrate the above features. This first draft was submitted to the third party assessors for remarks or modifications. After some modification cycles, the total scope of test cases was fixed. This scope of tests can be treated as a set that was accepted by all parties involved as giving sufficient confidence in the system properties of TELEPERM XS. Based on this set, the detailed test specification was elaborated and then submitted to the assessors for evaluation.

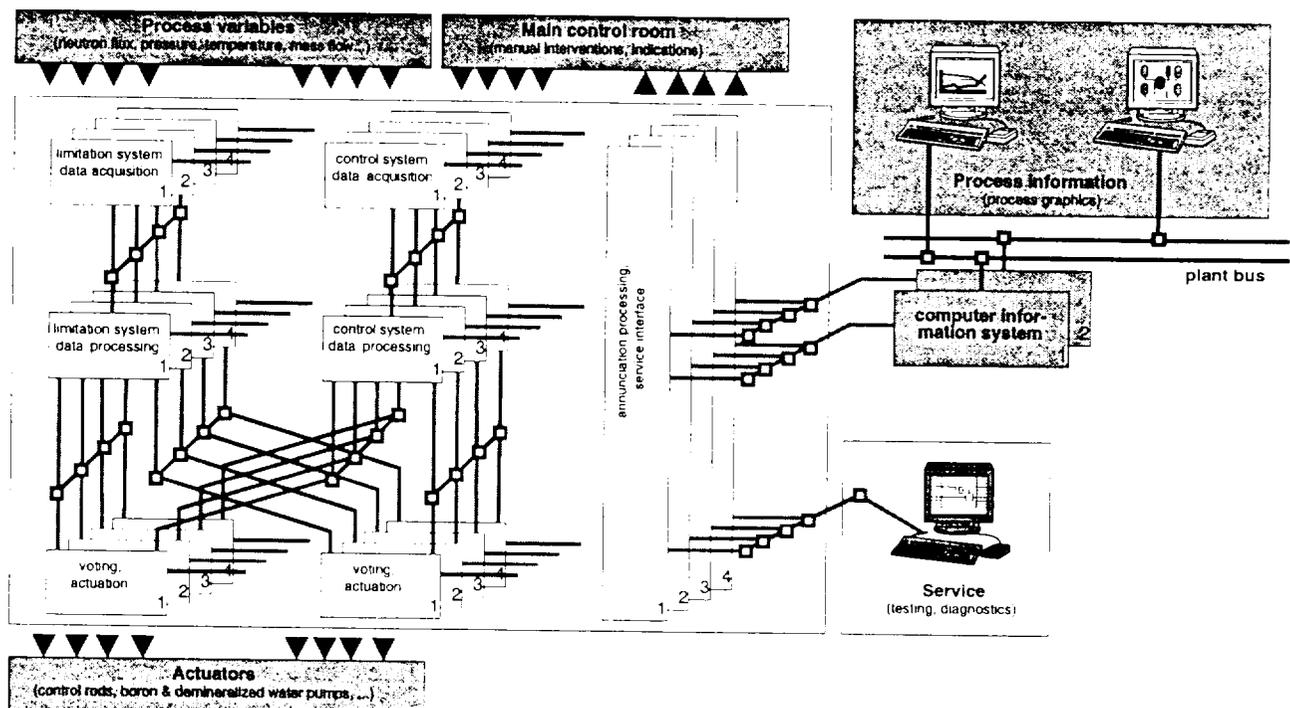
The integration and system test was performed on a real safety I&C system, which was designed for the replacement of the reactor controls and limitation systems together with the control assembly actuation system in the Unterweser nuclear power plant. This application was quite large. It consisted of 20 cabinets and included a variety of representative substructures suitable for demonstrating all relevant system features. The fact that a fully function-tested and electrically-tested computer system was used means that the design and commissioning aspects did not need to be considered in the integration and system test.

Most of the tests required application functions whose correct processing was monitored under various conditions of operation, stress, and failure. The original I&C functions of the controls and limitation systems in the Unterweser nuclear power plant were used for this purpose.

The hardware architecture in the test field was not modified during the course of the integration test. Therefore the tests that required modifications were performed in a much smaller configuration in the laboratories of the manufacturer. All tests were basically performed using the same software configuration. However, certain test cases for selected function computers required the generation of new software in order to obtain the desired effect on the system.

Several manual actions were required during the test procedure, in order to simulate failures or faults. It proved necessary, for example to:

- isolate the L2 link between two nodes (SVE1),
- unplug I/O modules,
- restart the processing computer by pressing the RESET button,
- generate an exception by pressing the NMI button,
- temporarily change the jumper settings on a processing computer, and
- perform operations on the run-time environment via the service unit.



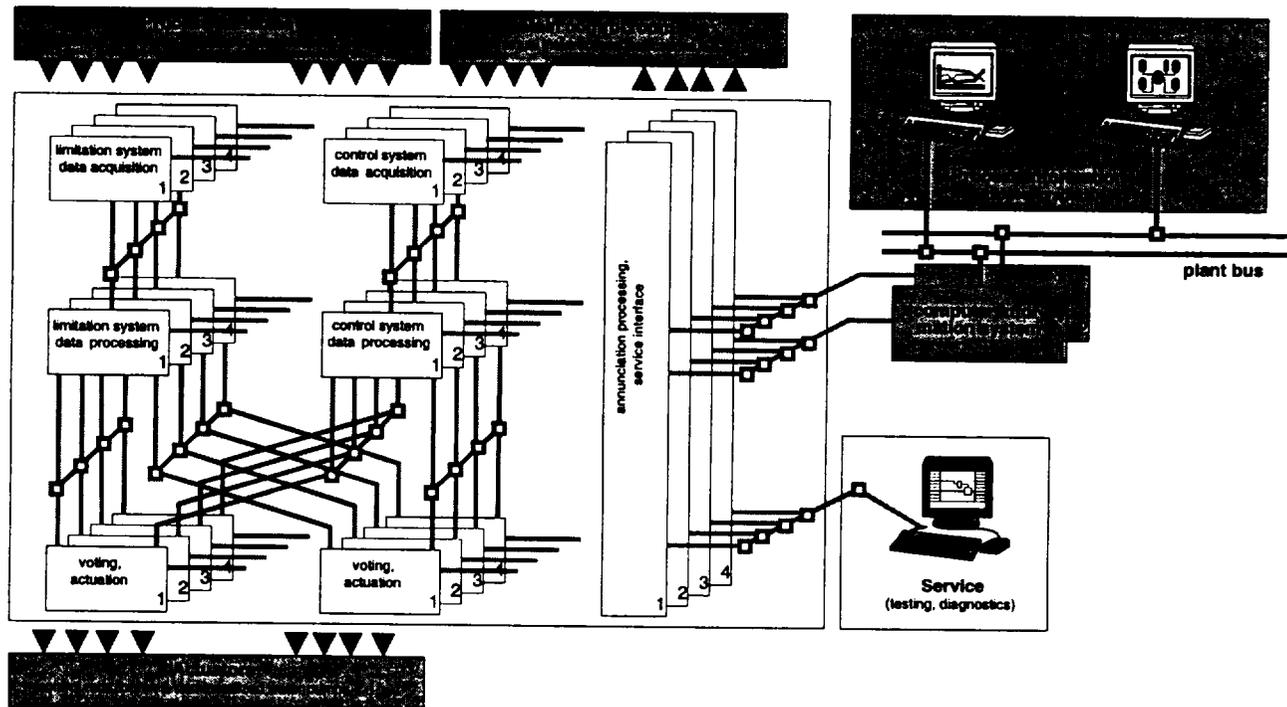
**Figure 3.17 Architecture of the Limitation and Control System at Unterweser**

The tests were basically performed as a black box test in which single-wire signals were generated with the aid of a test computer and fed into the peripheral input modules of the computer system. The system response was transferred to the test computer via peripheral output modules and documented. A subsequent comparison of the specified and the documented responses made it possible to conclude whether a test was successfully completed. In addition to the correct execution of the application function, system error messages and a number of internal variables, such as task processing time and bus loading, were also monitored in order to check the correct response.

All tests were performed by the manufacturer and supervised by the third party assessors. This means the assessors defined a set test cases that were of particular interest to them and they participated in these tests. All test steps were documented in test logs, which are included in the final test report. This test report was submitted to the assessors for evaluation.

TELEPERM XS: A Digital Reactor Protection System

- isolate the L2 link between two nodes (SVE1),
- unplug I/O modules,
- restart the processing computer by pressing the RESET button,
- generate an exception by pressing the NMI button,
- temporarily change the jumper settings on a processing computer, and
- perform operations on the run-time environment via the service unit.



**Figure 3.17 Architecture of the Limitation and Control System at Unterweser**

The tests were basically performed as a black box test in which single-wire signals were generated with the aid of a test computer and fed into the peripheral input modules of the computer system. The system response was transferred to the test computer via peripheral output modules and documented. A subsequent comparison of the specified and the documented responses made it possible to conclude whether a test was successfully completed. In addition to the correct execution of the application function, system error messages and a number of internal variables, such as task processing time and bus loading, were also monitored in order to check the correct response.

All tests were performed by the manufacturer and supervised by the third party assessors. This means the assessors defined a set test cases that were of particular interest to them and they participated in these tests. All test steps were documented in test logs, which are included in the final test report. This test report was submitted to the assessors for evaluation.

### 3.2.2.4 Submitted Documents and Results

#### Reports submitted for document review

The reports submitted for document review were the:

- Integration plan,
- Test specification, and
- Test report.

#### Evaluation report

The results of the evaluation will be documented by a certificate and the associated evaluation report. The certificates will confirm the properties of TELEPERM XS demonstrated by the system test. The evaluation of the test reports is still in progress, although all tests were successful completed in February 1997. The final evaluation report (TXS-AUST-0798-01) was completed in August, 1998.

The integration and system test ensures that:

- the hardware and software components cooperate correctly,
- the application software generated by the SPACE tools works as specified,
- the system behavior is completely defined by the specification and independent from the physical plant process (deterministic behavior) or other random effects,
- there is no interaction among independent safety functions even if they are processed on the same function computer,
- the response time of a safety function is within the limits calculated by the SPACE tools during the application system design phase,
- the system is testable and maintainable, and works correctly during periodic tests or startup,
- the failure barriers work correctly,
- the fault tolerant features and the fail-safe features work correctly, and
- failures are detected spontaneously and correctly signaled.

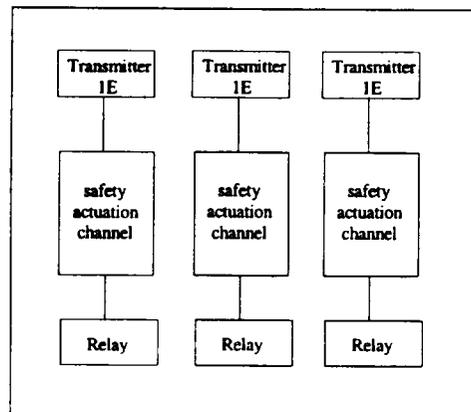
#### 4.0 Independence of Class 1E Equipment and Circuits

##### 4.1 Typical Architecture of Safety I&C Systems

The architecture of safety I&C systems is largely set by generic design requirements. Constraints result from rules and standards, from the postulated component failure combinations, from probabilistic requirements regarding loss of system function and spurious actuation, from test and repair requirements and from operational considerations such as space requirements.

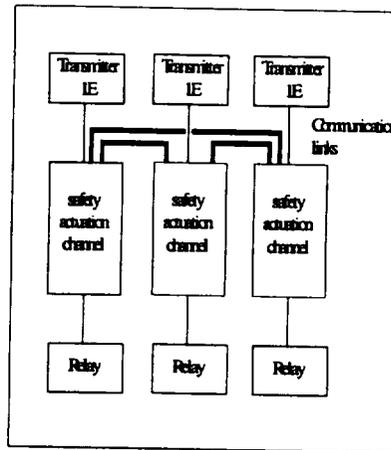
An architecture often used for safety I&C systems consists of three independent safety actuation channels, each of which monitors its own measured value and forms actuation signals that are then applied to 2-out-of-3 relay-voting in the switchgear (Figure 4.1).

In typical architectures, each safety actuation channel acquires the process variables via analog and digital input modules, processes the safety function and outputs the result of this processing via digital output modules to the switchgear where the signals are combined to provide actuation if certain logic is satisfied.



**Figure 4.1 Typical Architecture of an I&C Safety System**

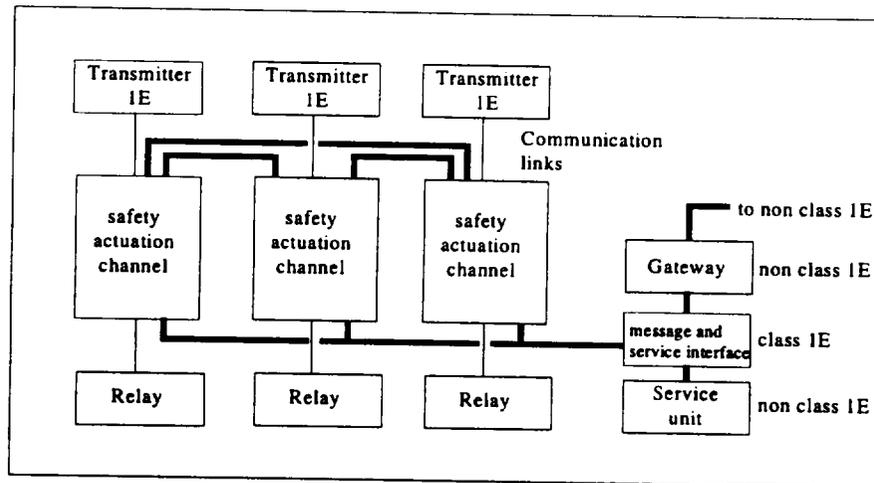
Because the process measuring sensors have a major effect on the unavailability of safety functions, screen architectures that provide early failure detection in the measuring sensors and their impact to preclude safety actuation are advantageous from the probabilistic point of view. This can be achieved by separating the whole safety task into two parts "Acquisition and distribution of measurement signals" and "Processing of the actuation signal". Serial communication links between the independent redundant channels enables cyclical signal exchange (Figure 4.2). The first part of the safety task acquires, filters and then distributes the measurement signals. The second part, i.e., the processing level then receives the three redundant measured process values. The true value is determined from these three measured values by on-line validation in a selection function block. The safety function is then executed with the true value. The formation of the true value includes both signal monitoring and screening of any faulted value detected.



**Figure 4.2 Architecture With Communication Links Crossing Channels**

If diagnostics and periodic tests are to be performed from a central service unit then additional communication processors have to be provided. Connections of the safety actuation channels with the service station via a class 1E MSI (message and service interface) computer (Figure 4.3) is provided in the basic system design. The MSI computer is required to isolate the class 1E safety actuation channels from the non class 1E service unit. By means of the MSI computer it can be ensured that any failure in the service unit will not prevent the capability of the safety I&C system to perform its safety functions.

MSI computers are physically separated from the equipment in the safety actuation channels.



**Figure 4.3 Architecture With Service Unit and Gateway**

If digital data transmission from the safety actuation channels to other non class 1E equipment is required it will be also be performed via a class 1E message and service interface computer and separate gateway. Busses connecting the MSI computers with equipment in the safety channel are fiber optics and separated from the buses connecting the MSI computer with the non class 1E gateway.

TELEPERM XS has the capability to provide isolated analog output signals to remote indicators and interfacing control systems.

#### **4.2 Design Principles**

The independence of non class 1E circuits from class 1E circuits and the independence of redundant 1E channels will be achieved by means of

- physical separation
- electrical isolation
- data flow separation and
- program flow separation.

Electrical isolation will be performed by class 1E isolation means such that the maximum credible voltage or current transient applied to the non-1E side will not degrade the operation of the circuit on the other side.

Data exchange between computer based equipment can cause dependencies even if the equipment are physically separated and electrically isolated. Data exchange between independent actuation channels is not necessary for all applications. If required independence will be achieved by data flow and by program flow separation. The communication principles applied for TELEPERM XS ensures this separation with the exception of the communication with the service unit under special conditions. This case will be explained later in detail.

Data flow separation is designed such that any failure in the data sending equipment can only affect the transmitted data itself but no other data in the data receiving equipment. As a necessary prerequisite independent communication means will be provided for all links which rely on data flow separation. Therefore links between class 1E and non class 1E equipment will be separated from links between redundant class 1E equipment. In addition links between redundant class 1E equipment will be used as end to end connections. Serial data transmission between class 1E equipment and non class 1E equipment will be performed via a class 1E qualified "message and service interface (MSI)" computer. The MSI computer is an isolation means for serial data communication to ensure that communication links connected to the class 1E equipment in the safety actuation channel can not be affected by any failure or data load on the other side.

Program flow separation ensures that the program flow in data receiving equipment is not influenced by the received data. Program flow separation is applied for all communication links between 1E equipment as well as for serial communication between 1E and non 1E equipment with one exception. The serial communication with the service unit under special conditions of operation constitutes the exception.

#### **4.3 Signal Transmission from Class 1E to non 1E Equipment and Circuits**

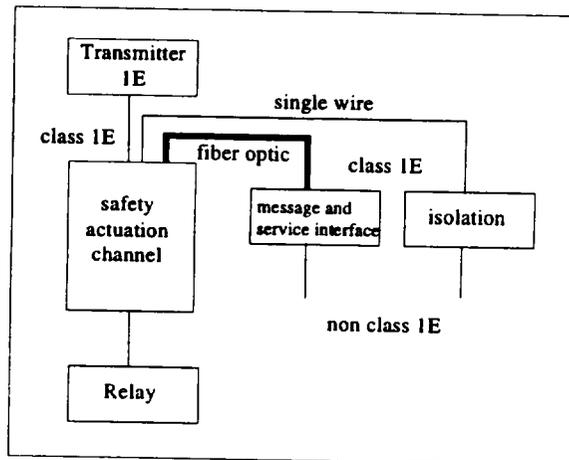
Class 1E signals may be transmitted to non class 1E circuits or equipment by single wires as current or voltage signals or by serial busses as digital data via a class 1E service and message interface. Single wire signal transmission is typically applied if class 1E transmitter signals or actuation signals have to be used as non class 1E circuits or control panels. Digital data

transmissions are typically used if class 1E signals have to be processed or stored by non class 1E data loggers or process computers.

#### 4.4 *Single Wire Signal Transmission*

Independence between class 1E circuits and non class 1E circuits will be achieved by one of the following ways (Figure 4.4).

- a. via a 1E electrical isolation device if the class 1E signal is electrically connected to any equipment in the safety actuation channel
- b. via the class 1E MSI computer without an additional isolation device



**Figure 4.4 Isolation of Single Wired Signal Transmission**

As the MSI computer is electrically isolated from the equipment in the safety actuation channel via fiber optics, additional isolation devices are not required. As the serial data transmission between the MSI computer and the equipment in the safety actuation channel are separated by program and data flow any degradation of the MSI computer can not affect the equipment in the safety actuation channel.

#### 4.5 *Digital Data Transmission via Serial Busses*

Any digital data transmission from the class 1E equipment in the safety actuation channel to non class 1E equipment will be performed via a class 1E message and service interface computer and a separate gateway (Figure 4.3). Busses connecting the MSI computers with equipment in the safety channel are fiber optics and are separated from the buses connecting the MSI computer with a non class 1E gateway. Within the MSI computer only a proprietary level 2 (level 2 of the OSI reference model) communication service is implemented which does not support any kind of routing. Data flow and program flow separation within the MSI computer ensures that any failure in the gateway can not influence the cyclical data exchange between the MSI computer and the equipment in the safety channel. In addition data flow and program flow separation in the equipment of the safety actuation channels ensures that any failure in the MSI computer can not influence the safety functions in equipment of the safety actuation channels.

Electrical isolation between the class 1E equipment and the non class 1E gateway is achieved by the fiber optic busses between the equipment of the safety actuation channel and the MSI computer. The communication means between the MSI computer and the gateway may be cooper or fiber optic busses.

#### 4.5.1 Communication With the Service Unit

A service unit will be applied if it is required to perform diagnostics or periodic tests from a central control panel. A service unit is a non class 1E device. Therefore it is connected to the class 1E equipment of the safety actuation channels via the MSI computer (Figure 4.3). Again electrical isolation is achieved by the fiber optic busses between the class 1E equipment in the safety actuation channels and the MSI computers. Communication between the service unit and the equipment in the safety actuation channels is performed by two kinds of messages:

- a. signaling messages transmitted from the equipment in the safety actuation channels to the service unit via the MSI computer
- b. command messages transmitted from the service unit to the equipment in the safety actuation channels

Each processing unit transmits within each processing cycle a signaling message of a fixed length to the service unit (if existing). This message consists of a header with status information and a data field which is empty most of time. The status information in the header contains information about the identification code of the sending processing unit, its cycle counter, the number of pending errors, the operating mode etc. The data field contains detailed information about the kind and the location of an error if a error has been newly detected within the processing cycle. This information is used in the service unit to monitor and to log the status of the safety system.

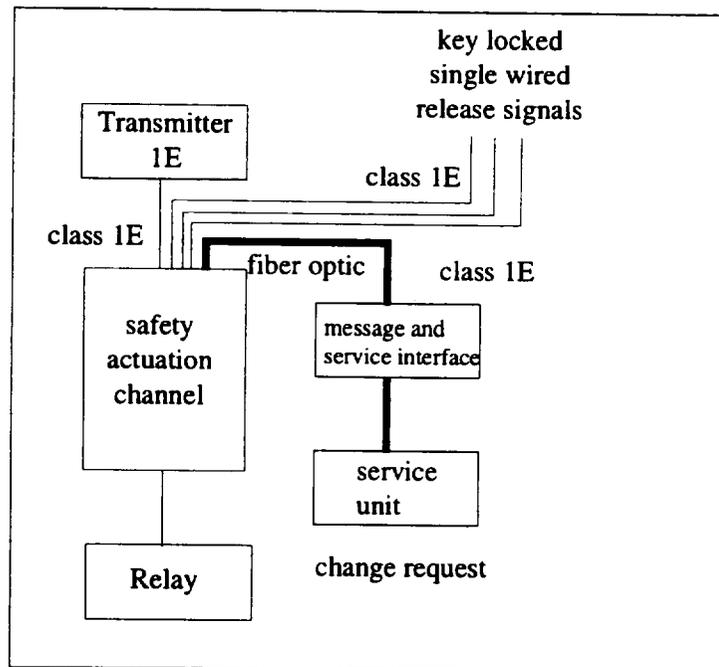
For test or diagnostic purposes it is possible to transmit commands from the service unit to the processing units in the safety actuation channels by means of the command messages. Command messages from the service to the MSI computers are transmitted on demand. The MSI computer checks the integrity of the message and copies its content into control messages of a fixed length which are transmitted cyclically between the MSI computer and the processing units in the safety actuation channels. By this way the communication load between MSI computer and the processing units is independent from the activities at the service unit.

Via these commands it is possible to influence data and program flow of the target processing unit. Because it could be possible to degrade the independence of redundant safety actuation channels by this linkage additional protection means are provided. The whole set of possible commands is subdivided into four categories. Category one contains only the command to read and to acknowledge error messages in the target processing unit. Commands of this category are not able to influence the safe processing of the safety functions. The second category includes additional commands to read other data and to change predefined parameters. These parameters can only be changed if they have been specified as changeable parameters during the system design phase. The third category includes additional commands to interrupt the cyclical processing and to simulate input or output data for test purposes. The fourth category includes additional commands to perform detailed diagnostics.

Corresponding to the four command categories the processing units of TELEPERM XS can be operated in four different modes. These are:

- a. cyclic processing
- b. parameter entry on definition
- c. function of test
- d. diagnosis

Within each operating mode only the corresponding set of commands are accepted by the software of the class 1E processing unit. That means within the mode "cyclic processing" only commands of the first category are accepted. To change the mode of operation from the "cyclic processing" to a higher mode two independent means are required. First it is necessary to set a mode specific release signal and second to transmit the associated request command from the service unit. Independence between redundant channels is ensured by the release signals. Each channel will be provided with independent single wired key-locked release signals. By adequate means (e.g., interlocks) it will be ensured that release signals can only be set for one channel at the same time. The set back of a release signal will cause to restart the processing unit immediately with the mode "cyclic processing" and to perform the safety functions. For security reasons it will also be ensured that the release signal can not be set by the same individual who sends the requests command to change the mode of operation from the service unit.



**Figure 4.5 Release Signals to Enable Change of Mode of Operation**

The means described ensure that any failure of the service unit or a human error of the individual operating the service unit can not affect processing units which are in the mode "cyclical operation". If a failure of the service unit or human error of the individual operating the service unit occurs while a processing unit is not in the mode "cyclical processing" the

consequences of the failure or error are restricted to this processing unit. In any case, the integrity of the 1E equipment can be verified.

#### **4.6 Transmitters**

Transmitters will be connected to input devices belonging to the same channel as the transmitters. The input devices perform galvanic isolation and analog to digital conversion. Additional 1E isolation devices are not required if the transmitter signals are only used within the 1E equipment of the same channel.

#### **4.7 Switchgear**

Actuation signals from the safety actuation channels to the switchgear will be electrically isolated using 1E isolation devices.

#### **4.8 Power Supply**

Power circuits of the class 1E safety actuation channel will be connected to the 1E power bars via fuses. Electrical isolation by fuses will be done on a subrack level.

#### **4.9 Signal Transmission Between Redundant Class 1E Channels**

Signal transmission between redundant class 1E channels may be required for availability or reliability reasons. If required it will be performed by serial fiber optic Profibusses in an end to end configuration (Figure 4.2). Electrical isolation is ensured by fiber optics. Data flow and program flow separation are ensured by the TELEPERM XS communication principles and by use of dedicated busses for each communication link. In general signal transmission between redundant class 1E channels is applied to provide each channel with the full redundant information. The redundant information is used by majority voting mechanisms to identify incorrect information, and exclude it from further processing.

## 5.0 Summary of Engineering Procedures and Project Instructions

### 5.1 *Engineering Procedure, Software Life - Cycle Process*

The Engineering Procedure 1.1 defines the software life-cycle-process for the TELEPERM XS product. It implements the life-cycle related recommendations of IEC 880 and IEEE 1074 (1991). The requirements of the procedure can be summarized as follows.

A safety I & C system shall be designed in a suitable engineering environment using matched component and high performance engineering tools. Within the pre-development process requirements from the fluid system engineer as well as from the I & C engineer shall be analyzed with respect to specify features, interfaces and performance data for the components and tools. According to this analyses components and tools shall be developed in the development process. In the post-development process the components and tools shall be applied to design safety systems as plant specific projects.

#### 5.1.1 Pre-Development Process

In the pre-development process requirements from the fluid system process shall to be analyzed with respect to device performance data for the components and to elaborate an adequate interface between fluid system engineers and I & C engineers. In addition requirements from the I & C systems itself shall be analyzed (testability, isolation) with respect to device performance data for the components.

#### 5.1.2 The Development Process

The development of each component and of each tool shall follow a development process, which consist of six phases:

1. Requirements Definition
2. Technical Design
3. Detailed Design
4. Implementation
5. Integration
6. Test

The Requirements Definition is based on an analysis of the User, the developer of the environment. Requirements shall be analyzed with respect to functionality, data and interfaces, performance and the implementation. Results of the analyses shall be described in the requirements specification.

In the Technical Design Phase it shall be elaborated how the requirements specification will be met by the component. During this phase the design rules shall be specified, the structure of the component with its interfaces shall be specified, functional requirements shall be assigned to individual functions and the feasibility shall be evaluated. The results of this phase shall be described in the technical design specification.

In the Detailed Design Phase each function shall be specified in detail with all interfaces. The data model shall be specified and how the function have to be assigned to individual modules. If an object-oriented design is selected all objects and their attributes shall be identified and assigned to classes. The classes shall be specified. The results of the detailed design phase shall be described in the detailed design specification.

Within the Implementation Phase the internal structure of modules and classes shall be specified and translated into software code. The results of the implementation phase shall be described in the implementation specification.

In the Integration Phase the components shall be integrated according to an integration plan.

Test shall be performed on a module level as well as on a system level. The specification and the documentation of test shall be performed according to Engineering Procedure 1.6 "Tests".

#### 5.1.3 Post-Development Process

The design of safety systems on TELEPERM XS technology shall follow the following phases:

1. Clarification of the requirements safety function system requirements
2. Distribution of safety functions to independent equipment
3. Formal system specification
4. Code generation and hardware manufacturing
5. Implementation and Test
6. Maintenance

The distribution of the safety functions shall take into account, diversity requirements, independence requirements, safety categories, reliability and availability requirements. Distribution of safety functions shall be done using form sheets. The content of the form sheet is specified in a separate engineering procedure.

The formal specification shall be performed using the engineering tools of TELEPERM XS. The functional specification shall be structured in the hierarchical levels; overview level, group level and individual levels. All information is managed in a database system. The hardware specification defines the allocation of hardware modules in the subrack, the allocation of the subracks in the cabinets, and the assignment of measured signals to I/O modules.

The formal specification shall be used for automatic code generation using the engineering tools of TELEPERM XS. The code is loaded into read only memories of the target system. The integrated system shall be tested in the test field. Maintenance activities shall be defined on an application specific basis.

#### 5.1.4 Identification of Tools and Components

The following tools and components shall be developed during the development phase:

1. Function Block Libraries
2. Function Diagram Module and Function Diagram Group Module

3. Runtime Environment
4. Self-test Routines
5. Specification Tool
6. Code Generator
7. Verification Tools

## **5.2 *Engineering Procedure, Configuration Management Plan***

Engineering Procedure 1.5 gives requirements and procedures necessary for the configuration management activities of the project. It identifies the software configuration management requirements and establishes the methodology for generating configuration identifiers, controlling engineering changes and maintaining status accounting during the design and development of software configuration items. It is applied to all software and associated documentation of TELEPERM XS system. The content of the engineering procedure 1.5 is summarized as follows.

### **5.2.1 Management**

Configuration identification shall be applied to all software of the, both code and associated documentation. A software configuration shall be made up of software elements and the associated documentation. A configuration item is an identifiable element of a software configuration, which may be an individual document, or a whole software configuration.

The label of each configuration item has to be unique. A version number has to be assigned to each configuration item. Baselines shall be established for the control of design, product and engineering changes. Baselines are defined by the product authority. Throughout the development life cycle, at the discretion of the configuration control board, releases shall be performed. Releases of a software configuration shall be defined by a list that identifies all items of the configuration. The procedure to change configuration items consists of the following steps:

1. Change Request
2. Evaluation of the request by the development group
3. Proposal how to perform a change
4. Decision about the change
5. Performing the changes (including test and update of the documentation)

### **5.2.2 Configuration Control**

Software configuration management and change control shall be applied to all documents and code. Control shall be affected through the implementation of the configuration identification, the change control and status accounting functions. Changes shall be initiated by a formal change request. A change request shall include the following information:

1. Identification of the request, product, date, and author

2. Specification of the request including the reason
3. Identification of the configuration item
4. Category of the request
5. Behavior of the product as it is
6. Requested behavior of the product

Change requests shall be analyzed and evaluated. As a result a formal change proposal shall be elaborated. This proposal shall show possibilities how to perform the changes and the associated consequences. The change request forms the basis for the decisions taken by the project management. The decision is documented by a development order. After development, test, and updating of the product related documents, the change process shall be finished by a formal document which shall contains a short report about the changed configuration item.

#### 5.2.3 Tools

An integrated set of tools is used for configuration control and status accounting at the project.

#### 5.3 ***Engineering Procedure, Guideline for Documentation***

Engineering Procedure 2.2a provides requirements and recommendations for the following:

1. Identification, distribution and archiving
2. Form, verification and revision
3. Application of tools for generation and management

#### 5.4 ***Engineering Procedure, Structure of Contents of Requirements Specifications of Hardware and Software Components***

Engineering Procedure 3.3 provides requirements and recommendations for structuring hardware and software component requirements specification documents produced for the TELEPERM XS development project. The object and contents of the individual requirements specifications is determined by the system design specification which is a high level document used as reference for a modular development process.

The requirements specification is the milestone result of the first phase of product development. In this phase, the tasks of the required products are analyzed and documented. The requirement's specification provides constraints to be fulfilled by the product. As far as reasonable, a requirements specification may be subdivided into a document forming a frame, documents requiring individual products within the frame, and short documents or sheets for similar products (e.g., function blocks) for which common features are required. Parts of the requirement's specification, especially extensive explanation for some requirements - may be written as separate work report.

The Table of Contents for a requirement's specification is structured according to the following list:

- 0. Table of Contents
- 1. General
- 2. Requirements
  - 2.1 Outline and limits for requirements
  - 2.2 Requirements for the overall task and subdivision into partial tasks
  - 2.3 Relations between the overall task and the partial tasks
- 3. Quality Assurance
- 4. Further Subjects

In Section 1 "General," the following information is provided:

- a. Cause, motive
- b. Purpose, aim
- c. Economical data, future expectation
- d. Constraints and documentation

In Section 2.1, "Outline and limits of requirements," the following information is provided:

- a. Environment, hardware configuration
- b. Map of structure, scope of partial products
- c. Summarizing description of the partial tasks
- d. Technical data (e.g., response time, usage of resources)
- e. Quality properties (reliability, user friendliness, maintainability, portability)
- f. Guidelines/standards/laws

In Section 2.2, "Requirements for the overall task and subdivision into partial tasks," the following information is provided:

- a. Normal operation
- b. Disturbances
- c. Data saving and protection
- d. Start/restart
- e. Generation/Installation
- f. Maintenance/diagnose

In Section 2.3, "Relations between the overall task and the partial tasks," the following information is provided:

- a. Communication diagram
- b. Data catalog
- c. User interface

d. Data exchange

In Section 3, "Quality Assurance," the following information is provided:

- a. Quality properties
- b. Activities for Quality Assurance
- c. Documentation
- d. Procedures for releasing the development documents
- e. Sub-suppliers

In Section 4, "Further subjects," the following information is provided:

- a. Terms and abbreviations
- b. List of literature
- c. History of versions
- d. Configuration management

**5.5 *Engineering Procedure, Structure of Contents of Technical Design Specifications of Software Components***

Engineering Procedure 3.4 provides requirements and recommendations for structuring of technical design specification documents of software components produced for the TELEPERM XS development project. The relationship between the requirements specification (input document) and the detailed design and/or implementation description documents is explained.

At the beginning of the work on the software technical design specification the author (supplier of software), the reviewer, and the releasing person are assigned. The design specification is the milestone result of the second phase of product development. In this phase, an analysis is performed of tasks, and of the functional and non-functional requirements. From this, a basic design of the required product is derived and documented.

As far as reasonable, a technical design specification may be subdivided into a document forming a frame, documents describing individual products within the frame, and short documents or sheets for similar products (e.g., function blocks) for which common features are described in a document forming a frame. Parts of the design specification, especially extensive explanation for some design decisions, may be written as a separate work report.

The table of contents of a design specification shall be structured according to the following list:

- 0. Table of Contents
- 1. General
- 2. Requirements
  - 2.1 Outline and limits of the software system
  - 2.2 Detailed description of the software system and its modules
  - 2.3 Relations between the overall software system and its modules
  - 2.4 Description of interfaces
  - 2.5 Test strategy and methods

3. Quality Assurance
4. Further Subjects

In Section 1, "General," the following information is provided:

- a. Existing similar solutions
- b. Guidelines, standards, laws

In Section 2.1, "Outline and Limits of the Software System," the following information is provided:

- a. Environment, hardware configuration
- b. Analysis of requirements and of map of structure, scope of system and modules
- c. Summarizing description of the software modules
- d. Feasibility
- e. Technical data (e.g., response time, usage of resources)
- f. Quality properties (reliability, user friendliness, maintainability, portability)

In Section 2.2, "Detailed Description of the Software System and its Modules," the following information is provided:

- a. Normal operation of the system and the modules
- b. Disturbances
- c. Description of operation of the system and the modules (e.g., by data flow diagrams, control flow diagrams, state transition diagrams)
- d. Data description
- e. Data saving and protection
- f. Start/restart
- g. Generation/Installation
- h. Plant specific data configuration
- i. Maintenance/diagnose

In Section 2.3, "Relations Between the Overall Software System and its Modules," the following information is provided:

- a. Communication diagram
- b. Data catalog
- c. User interface
- d. Data exchange
- e. In Section 4 „Further subjects“ the following information shall be provided
- f. User documentation, manuals
- g. Terms and abbreviations

- h. List of literature
- i. History of versions
- j. Configuration management

#### 5.6 ***Engineering Procedure, Structure of the Contents of Detailed Design Specifications for Software Components***

Engineering Procedure 3.5 regulates the contents and the structure of detailed design specifications for TXS software components. It is implemented in accordance with the design recommendations of IEC Std. 880 (Clause 5; App. B1.a-e, B3.a-b and C).

Purpose and scope of application of the engineering procedure are given in its introduction together with responsibilities and competence for the preparation of a detailed design specification.

Some features of a detailed design specification and some rules on its preparation are stated before dealing with the actual structure of its' contents. The design specification describes how the solution specified in the technical design specification will be realized in a module structure suitable for implementation. The individual sections of a detailed design specification contain the following content:

Section 1, "General," part of a detailed design specification contains general conditions for and limitations to the entire component and the individual modules. This relates to consideration of already existing solutions, consideration of standards and laws, and possible passive and active protective rights.

Section 2, "Development," results of the detailed design specification are applied to the component and module specifications. There is a brief introductory description of the components and their boundary to the outside. The hierarchical structure is represented in a rough overview graphic. The structure defines the modules, their relations to each other, and also to the environment. In addition, the environment in which the component is to be integrated shall be examined. Technical data and features as well as quality features resulting from functional and non-functional requirements described in the technical design specification on the one side and superior boundary conditions and limitations for realization on the other shall also assessed and checked here. This shall be followed by a detailed description of a module structure suitable for data processing and the runtime behavior. The detailed static module structure shall completely cover the functionality specified in the technical design specification. Suitable methods shall be provided for the description of the runtime behavior during normal and incident operation, also considering dynamically aspects. The method of structured analysis (SA ) shall serve for static representation of communication relations including data flows and synchronization mechanisms; structure diagrams shall serve for representation of sequential characteristics; state transition diagrams represent the execution progress with reference to time, causality and dependencies on events. The data structures are comprised in a data catalog and are described in a formal way.

The components are linked with their environment via external interfaces. It is distinguished between user interfaces and programming interfaces with their description being oriented with regard to function and data by the requirements in the technical design specification.

In the descriptions explained above, the component as a whole is always looked at. Now the individual modules themselves are characterized with each of them being described separately. The following points are specified: structure, interfaces, function, resources and data structure.

Finally, information is provided on test strategy and test procedures. For example, coverage of the required functionality, module interfaces, and response times are re identified here. Possible specification of the procedure can depend on test targets, complexity, and structure of the test objects as well as the criteria for test termination. Test data for integration tests with other components can be specified dependent in accordance with the information available.

Chapter 3, " Quality assurance," provides conditions and rules to ensure product quality. These may be constructive quality assurance measures like design rules and methods for a uniform and fault-free procedure, maintenance requirements, specification of prerequisites for preparation of language programming guidelines, and release procedures.

### **5.7 *Engineering Procedure, Structure of the Contents of Implementation Specifications for Software Components***

Engineering Procedure 3.6a, regulates the contents and the structure of the implementation specifications for TXS software components. Together with the programming guidelines of Engineering Procedure 2.1, it specifies the implementation recommendations of IEC Std. 880. Engineering Procedure 3.6a covers Clause 5.2 and App. D of IEC Std. 880. Engineering Procedure 2.1 covers the coding sections of Clause 5, App. B2.a-f, App. B3.c-d, App. B4.a-g and App. B5.a-d.

Some features of an implementation specification and some rules on its elaboration are explained, before dealing with the actual structure of the contents. The implementation specification describes how the modules specified in the detailed design specification are to be implemented. The individual sections of the implementation specification have the structure and contents outlined below:

Section 1, "General" part of an implementation specification describes general conditions for and limitations to the entire component and the individual modules. These are implementation points such as, consideration of already existing solutions, consideration of standards and laws and possible passive and active protective rights.

Section 2, "Programming Language, Compiler, Linker etc." deals with the programming language, compiler to be used. It is particularly concerned with automatic fault-prevention automatic aids.

Section 3, "Development Results" provides specification of the individual modules. This is the main section of the document containing a detailed description subsection for each module. At first, task and purpose of the module shall be pointed out. This is followed by a description of the interface. In addition to the module's own call interface, active calls to other modules, to the operating system and to the process are described. Input and output data are specified and types of tests on data and responses to faulty are discussed. Then the actual realization is described. The algorithm is completely specified for standard operation and for operation in the case of a fault. Suitable methods are provided for representation of control and data flows (see Engineering Procedure 3.5). Information on data going beyond that included in the source code text is specified here. Implementation information not evident from the source code is written

down. This includes all compiler, linking, loading and installation procedures as well as references to source files. The source code is enclosed in the implementation specification as appendix. Finally, test means are specified to by the designer to facilitate the independent tests. These are specifications of the test conditions, test drivers, dummy modules, test data generators etc.

Section 4, "Quality Assurance," of an implementation specification provides conditions and measures to ensure product quality. These may be constructive QA measures like design rules and procedures for a uniform and fault-preventing process.

### 5.8 *Engineering Procedure, Tests*

Engineering Procedure 4.1, "Tests," provides requirements for the specification, execution and evaluation of tests including documentation and presentation of results. The Introduction provides purpose and scope information, together with responsibilities for the preparation of a test specification, test execution and test report.

Since testing is one of several inspection procedures in the life of a software product, the main features of testing are explained briefly. The main points of relevant test tasks, i.e., module testing, integration test and system test are described.

Testing includes six activities. The results of these activities are documented in a test specification or in a test report. The activities are:

#### 1. Planning the test.

At first, the aspired test targets are identified. Dependent on the application, the software product can be divided into sections of different critical importance and thus different test severity. As further test preparation, the test environment, conditions for normal and abnormal test termination, conditions for test interruption and restart are specified. The required resources are identified and a rough schedule is determined. The results of this planning are recorded in Section 1, "Test scheme" of the test specification.

#### 2. Designing the test.

The test objects (programs to be tested) as well as the test environment, tools and the executable program are assembled. Test cases are identified and the entire test scope is specified. The schedule is detailed and acceptance criteria are provided. The results of this specifying activity are recorded in Section 2, "Test Design," of the test specification.

#### 3. Identifying test cases.

The test cases identified in the course of test design are specified in detail. These details include input and expected output data, and are described in Section 3, "Test Case Specifications."

#### 4. Determining the test proceedings.

Test steps and their succession shall be determined here. They are written down in Section 4, "Test Proceedings Specification."

5. Executing and recording the test.

This activity involves installation of the test environment, execution of the tests. The activities and results are documented in the test report. Based on the test acceptance criteria, it is determined for each test case whether the test object has passed or failed. In case of a failure, a failure analysis is carried out. Information on failures is documented and a decision is made on how to proceed.

6. Evaluating the test.

This step comprises the evaluation of the test results on the basis of the test logs, the state of the test objects after the test, fault analysis and statistics. The evaluation is documented in the test report. It is whether or not the software meets the requirements. The test report allows for a complete diagnosis of design and processing deficiencies. Disposition of software deficiencies and test deviations that appeared during the test is also included. All test products are retained for the regression test.

The test documentation is, as evident above, structured according to the described activities and consists of the two documents "Test specification" and "Test report." All planning and specification activities carried out before the test are written down in the test specification. All accompanying activities during and evaluating activities after the test are also documented in the test report.

5.9 **Engineering Procedure, Reviews**

Engineering Procedure 4.2, "Reviews," specifies the process for implementation of reviews in the course of the development of TXS software components. Purpose and scope of the engineering procedure are given in its introduction together with responsibilities and competence regarding reviews. Participants in the review are the designer of the software component concerned, a review leader, and one or more competent checkers. Reviews take place at least at the end of each development phase based on the resultant documentation from each phase. The review is conducted in two steps:

1. Document review with written comments.

The document is thoroughly read by the checkers participating in the review and commented on in writing (errors, risks, hints, objections, suggestions, questions). The review leader takes on the role of administrator.

2. Review meeting.

The written comments are discussed in a meeting and results are recorded. There are two review methods possible:

- a. Walk-through, should mainly be applied to specifying documents, and
- b. Inspection, for program code.

Decisions are made for the treatment of open items. The review leader takes on the role of moderator. As the final step, a review report is prepared.

**5.10 Project Instruction No. 2, "Document Control for Software Development in the TELEPERM XS Project"**

Project Instruction No. 2, "Document Control for software development in the TELEPERM XS project requires a uniform handling of development specifications by all organizational units of different Siemens groups participating in the project. Development specifications for TELEPERM XS software components are the final documents of the phases according to the life cycle described in Engineering Procedure 1.1. Software development documents pass through the following stages:

**1. Preparation.**

The author prepares the first draft of the specification, for example a requirements specification in accordance with the instructions in the Engineering Procedure 3.3. He then signs in the author field of the form "Responsibilities" (2<sup>nd</sup> page of the specification). The number of the project-internal filing location is specified on the cover sheet.

**2. Consistency check.**

The draft is checked by at least one checker on content consistency and correctness of the interface. The result of this check is included in the review report (see below).

**3. Technical review.**

Each specification shall be reviewed in accordance with the Engineering Procedure 4.2. The participants in the review are chosen such that they are technically competent, but not involved in the preparation of the specification. When the document is corrected, they sign in the field for review participants on the form "Responsibilities". The results of the review and the consistency check are written down in the review report. The specification is now in the state "2<sup>nd</sup> draft".

**4. External check.**

In the course of type testing the TELEPERM XS system, each specification is sent to the external test institute (GRS) for a check. Each question, remark, change recommendation and fault identification has to be addressed by the author and the specification has to be changed in the respective way if necessary.

**5. External compliance check.**

The corrected specification is sent to the external test institute for a check. Necessary improvements can still be included. When the specification is accepted, it is considered to be qualified.

**6. Release.**

The internally and externally checked and corrected specification is released for application by signature of the project manager in the release field of the form "Responsibilities." A specification released for application is made known by the author using the standard distribution list of the project. The master copy shall be filed in the secretary's office and one copy each shall be deposited in the project-internal archive and the respective project filer. Each specification is part of a configuration unit. The changed configuration unit shall be

released by means of a release document. Limitations regarding the application of the configuration can also be stated in the release document.

#### **5.11 *Supplemental Engineering Procedures***

Additional Engineering Procedures used in Support of the TELEPERM XS safety system design process are as follows:

1. Engineering Procedure 1.1 – “Life Cycle Model for Safety System Software
2. Engineering Procedure 1.6 – “Verification and Validation Plan”
3. Engineering Procedure 1.7 – “Data Security”
4. Engineering Procedure 2.1 – “Coding Rules”

## 6.0 Industry Standards

The industry standards followed for TXS development and qualification are as follows:

### 6.1 IEC Standards

1. IEC 68-1 1988. "Environmental testing, General and guidance."
2. IEC 68-2-1 1990. "Environmental testing, Tests A: Cold."
3. IEC 68-2-2 1974. "Basic environmental testing procedures, Tests B: Dry heat."
4. IEC 68-2-3 1969. "Basic environmental testing procedures, Test Ca: Damp heat, steady state."
5. IEC 68-2-41 1983. "Basic environmental testing procedures, Test Z/BM: Combined dry heat/low air Pressure tests."
6. IEC 255-4 1976. "Electrical relays. Part 4: Single input energizing quantity measuring relays with dependent specified time."
7. IEC 801-1-1984. "Electromagnetic Compatibility for Industrial-Process Measurement and Control Equipment, General Introduction."
8. IEC 801-2-1991. "Electromagnetic Compatibility for Industrial-Process Measurement and Control Equipment, Electrostatic Discharge Requirements."
9. IEC 801-3-1993. "Electromagnetic Compatibility for Industrial-Process Measurement and Control Equipment, Immunity to Radiated Radio Frequency Electromagnetic Fields."
10. IEC 801-4-1988. "Electromagnetic Compatibility for Industrial-Process Measurement and Control Equipment, Electrical Fast Transient/Burst Requirements."
11. IEC 801-5-1993. "Electromagnetic Compatibility for Industrial-Process Measurement and Control Equipment, Surge Immunity Requirements."
12. IEC 801-6-1993. "Electromagnetic Compatibility for Industrial-Process Measurement and Control Equipment, Immunity to Conducted Disturbances Induced by Radio Frequency Fields."
13. IEC Std. 880-1986. "Software for Safety-Systems in Nuclear Power Stations."
14. (#) IEC 1000-4-2 1995. "Electromagnetic compatibility (EMC). Testing and measurement techniques. Electrostatic discharge immunity test. Basic EMC publication."
15. (#) IEC 1000-4-4 1995. "Electromagnetic compatibility (EMC). Testing and measurement techniques. Electrical fast transient/burst immunity test. Basic EMC publication."

16. (#) IEC 1000-4-5 1995. "Electromagnetic compatibility (EMC). Testing and measurement techniques. Surge immunity test."
17. (#) IEC 1000-4-6 1996. "Electromagnetic compatibility (EMC). Testing and measurement techniques. Immunity to conducted disturbances, induced by radio-frequency fields."
18. IEC Std. 1226 1995. "Nuclear power plants - Instrumentation and control systems important for safety – Categorization."
19. IEC 1131-1 1994. "Programmable controllers. General information."
20. IEC 1131-2 1995. "Programmable controllers. Equipment requirements and tests."
21. IEC 1131-3 1993. "Programmable controllers. Programming languages."
22. IEC 664 1980. "Insulation co-ordination within low-voltage systems including clearances and creepage distances for equipment."

## 6.2 **VDE Standards**

1. DIN VDE 0160 1988. "Electronic equipment for use in electrical power installations and their assembly into electrical power installations."
2. DIN VDE 0843-1 1987. "Electromagnetic compatibility for industrial-process measurement and control equipment; general introduction; identical with IEC 801-1, edition 1984."
3. DIN VDE 0843-2 1987. "Electromagnetic compatibility for industrial-process measurement and control equipment; electrostatic discharge requirements; identical with IEC 801-2, edition 1984."
4. DIN VDE 0843-3 1988. "Electromagnetic compatibility for industrial-process measurement and control equipment; radiated electromagnetic field requirements; identical with IEC 801-3, edition 1984."
5. DIN VDE 0843-4 1987. "Electromagnetic compatibility for industrial-process measurement and control equipment; electrical fast transient requirements; identical with IEC 65(Central Office)39."
6. DIN VDE 0843-5 1992. "Electromagnetic compatibility for electrical and electronic equipment; surge immunity requirements; identical with IEC 65A/77B(Secretariat)120/87."
7. DIN VDE 0843-6 1993. "Electromagnetic compatibility for electrical and electronic equipment; part 6: immunity to conducted disturbances induced by radio frequency fields (IEC 65A/77B(Secretariat)145/110:1993)."

8. DIN VDE 0871 B1 1991. "Radio interference suppression of radio frequency equipment; determination of limits for industrial, scientific and medical equipment; identical with CISPR 23, 1."
9. DIN VDE 0875 1977. "Specification for the radio interference suppression of appliances and systems."
10. DIN VDE 0878-2 1988. "Radio interference suppression of telecommunication equipment; equipment in telecommunication operating rooms."
11. DIN V VDE 0801 1990. "Grundsätze fuer Rechner in Systemen mit Sicherheitsaufgaben."

### 6.3 **IEEE Standards**

1. IEEE Std. 279-1971. "Criteria for Protection Systems for Nuclear Power Generating Stations."
2. IEEE Std. 308-1991. "Criteria for Class 1E Power Systems for Nuclear Power Generating Stations."
3. ANSI/IEEE Std. 323-1983. "IEEE Standard for Qualifying Class 1E Equipment for Nuclear Power Generating Stations."
4. IEEE Std. 338-1987. "Standard Criteria for the Periodic Surveillance Testing of Nuclear Power Generating Station Safety Systems."
5. IEEE Std. 344-1987. "Recommended Practice for Seismic Qualification of Class 1E Equipment for Nuclear Power Generating Stations."
6. IEEE Std. 379-1988. "Standard Application of the Single-Failure Criterion to Nuclear Power Generating Station Safety Systems."
7. IEEE Std. 384-1992. "IEEE Standard Criteria for Independence of Class 1E Equipment and Circuits."
8. IEEE Std. 603-1991. "IEEE Standard Criteria for Safety Systems for Nuclear Power Generating Stations."
9. IEEE Std. 665-1987. "Guide for Generation Station Grounding."
10. IEEE Std. 828-1990. "IEEE Standard for Software Configuration Management Plans."
11. IEEE Std. 829-1983. "IEEE Standard for Software Test Documentation."
12. IEEE Std. 830-1993. "IEEE Recommended Practice for Software Requirements Specifications."
13. IEEE Std. 1028-1988. "IEEE Standard for Software Reviews and Audits."

14. IEEE Std. 1042-1987. "IEEE Guide to Software Configuration Management."
15. IEEE Std. 1008-1987. "IEEE Standard for Software Unit Testing."
16. IEEE Std. 1012-1986. "IEEE Standard for Software Verification and Validation Plans."
17. (#) IEEE Std. 1050-1996. "IEEE Guide for Instrumentation and Control Equipment Grounding in Generating Stations."
18. IEEE Std. 1074-1991. "IEEE Standard for Developing Software Life Cycle Processes."
19. IEEE Std. 7-4.3.2-1993. "IEEE Standard for Digital Computers in Safety Systems of Nuclear Power Generating Stations."

#### 6.4 ***KTA Standards***

1. KTA 1401-1987. "General Requirements Regarding Quality Assurance."
2. KTA 3501-1978. "Reactor Protection System and Monitoring Equipment of the Safety System."
3. KTA 3503-1982. "Type Testing of Electrical Modules for the Reactor Protection System."
4. KTA 3506-1984. "Tests and Inspections of the Instrumentation and Control Equipment of the Safety System of Nuclear Power Plants."
5. KTA 3507-1986. "Factory Tests, Post-Repair Tests and Demonstration of Successful Service for the Instrumentation and Control Equipment of the Safety System."

#### 6.5 ***Code of Federal Regulations***

1. 10CFR50.49, "Environmental Qualification of Electric Equipment Important to Safety for Nuclear Power Plants."
2. 10CFR50.55a, "Codes and Standards."
3. 10CFR50.62, "Requirements for Reduction of Risk from ATWS Events for Light Water Cooled Nuclear Plants."
4. 10CFR50, Appendix A, "General Design Criteria."
5. 10CRF50, Appendix B, "Quality Assurance Criteria for Nuclear Power Plants and Fuel Reprocessing Plants."

#### 6.6 ***U.S. NRC Regulatory Guides***

1. Regulatory Guide 1.28. "Quality Assurance Program Requirements (Design and Construction)," 1985.

2. Regulatory Guide 1.70 - "Standard Format and Content of Safety Analysis Reports for Nuclear Power Plants." Office of Standards Development, U.S. Nuclear Regulatory Commission, November 1978.
  3. Regulatory Guide 1.89 - "Environmental Qualification of Certain Electric Equipment Important to Safety in Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, 1984.
  4. Regulatory Guide 1.97 – "Instrumentation for Light-Water-Cooled Nuclear Power Plants to Assess and Environs Conditions During and Following an Accident," 1983.
  5. (#) Regulatory Guide 1.152 - "Criteria for Digital Computers in Safety Systems of Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, 1996.
  6. (#) Regulatory Guide 1.153 - "Criteria for Power, Instrumentation, and Control Portions of Safety Systems." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, 1996
  7. (#) Regulatory Guide 1.168 - "Verification, Validation, Reviews and Audits for Digital Computer Software Used in Safety Systems of Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, 1997.
  8. (#) Regulatory Guide 1.169 - "Configuration Management Plans for Digital Computer Software Used in Safety Systems of Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, 1997.
  9. (#) Regulatory Guide 1.170 - "Software Test Documentation for Digital Computer Software Used in Safety Systems of Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, 1997.
  10. (#) Regulatory Guide 1.171 - "Software Unit Testing for Digital Computer Software Used in Safety Systems of Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, 1997.
  11. (#) Regulatory Guide 1.173 - "Developing Software Life Cycle Processes for Digital Computer Software Used in Safety Systems of Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, 1997.
  12. NUREG-0694 - "TMI-Related Requirements for New Operating Reactor Licenses," 1980.
  13. NUREG-0737 - "Clarification of TMI Action Plan Requirements," 1980.
  14. NUREG/CR-6303 - "Method for Performing Diversity and Defense-in-Depth Analyses of Reactor Protection Systems," 1994
- 6.7 **Miscellaneous Standards**
1. IAEA 50-C-QA (Rev. 1)-1988, "Quality Assurance for Safety in Nuclear Power Plants."

2. DIN EN ISO 9001-1994, "Model for quality assurance in design/development, production, installation and servicing."
3. DIN EN ISO 9000-3-1992, "Guidelines for the application of ISO 9001 to the development, supply, installation and maintenance of computer software."
4. SN29500-1996. "Failure Rates of Components."
5. ANS Std. 4.5 - "Criteria for Accident Monitoring Functions in Light Water Cooled Reactors."
6. ISA-S67.02-1980 - "Nuclear-Safety-Related Instrument Sensing Line Piping and Tubing Standards for Use in Nuclear Power Plants."
7. ISA-S67.04-1994 - "Setpoints for Nuclear Safety-Related Instrumentation Used in Nuclear Power Plants."
8. ASME Std. NQA-1-1994. "Quality Assurance Requirements for Nuclear Facility Applications."
9. ASME Std. NQA-2a-1990 Part 2.7. "Quality Assurance Requirements of Computer Software for Nuclear Facility Applications."
10. CISPR 11 1998. "Limits and methods of measurement of radio disturbance characteristics of industrial, scientific and medical (ISM) radio-frequency equipment."

#### 6.8 **EN Standards**

1. EN 55011 1998. "Limits and methods of measurement of radio disturbance characteristics of industrial, scientific and medical (ISM) radio-frequency equipment."
2. EN 61000-4-2 1995. "Electromagnetic compatibility (EMC). Testing and measurement techniques. Electrostatic discharge immunity test. Basic EMC," publication.
3. EN 61000-4-4 1995. "Electromagnetic compatibility (EMC). Testing and measurement techniques. Electrical fast transient/burst immunity test. Basic EMC," publication.
4. ENV 50140 1994. "Electromagnetic compatibility. Basic immunity standard. Radiated radio-frequency electromagnetic field. Immunity test."
5. ENV 50141 1993. "Electromagnetic compatibility; basic immunity standard; conducted disturbances induced by radio-frequency fields; immunity test."
6. ENV 50142 1994. "Electromagnetic compatibility - Basic immunity standard - Surge immunity tests."

(#) Implies this standard is in effect for current and future TELEPERM development, but was not available at the time of the original TXS development effort.

## 7.0 Conformance With IEEE Standards

The U.S. Code of Federal Regulations, 10 CFR 50.55a(h) requires protection systems to meet the requirements of ANSI/IEEE Std. 279. The criteria of ANSI/IEEE Std. 279, "Criteria for Protection Systems for Nuclear Power Generating Stations," address considerations such as design bases, redundancy, independence, single failures, qualification, bypasses, status indication, and test. IEEE Std. 603, "Criteria for Safety Systems for Nuclear Power Generating Stations," has since superseded ANSI/IEEE Std. 279. The guidance in IEEE Std. 603, as endorsed by Reg. Guide 1.153, "Criteria for Power, Instrumentation, and Control Portions of Safety Systems," incorporates the guidance of ANSI/IEEE Std. 279, and includes all I&C safety systems within its scope. The guidance described in IEEE Std. 603 is used in the evaluation of I&C safety systems.

IEEE Std. 603 does not directly discuss digital systems. It is supplemented by IEEE Std. 7-4.3.2, "IEEE Standard for Digital Computers in Safety Systems of Nuclear Power Generating Stations," which provides criteria for applying IEEE Std. 603 to computer systems. IEEE Std. 7-4.3.2 is endorsed by Reg. Guide 1.152, "Criteria for Digital Computers in Safety Systems of Nuclear Power Plants." References to IEEE Std. 603 in the remainder of this appendix should be read as including IEEE Std. 7-4.3.2, Reg. Guide 1.152, and Reg. Guide 1.153.

### 7.1 *Single-Failure Criterion*

**Criterion.** Single-Failure Criterion. The safety systems shall perform all safety functions required for a design basis event in the presence of: (1) any single detectable failure within the safety systems concurrent with all identifiable but non-detectable failures; (2) all failures caused by the single failure; and (3) all failures and spurious system actions that cause or are caused by the design basis event requiring the safety functions. The single-failure criterion applies to the safety systems whether control is by automatic or manual means.

**Compliance.** The TELEPERM XS safety system is designed so that a single failure within the protection system will not prevent the initiation or completion of a protective function at the system level. For U.S. nuclear power plant applications, the TELEPERM XS is applied to four redundant process channels and two trip logic trains for each Reactor Trip (RT) or Engineered Safety Features (ESF) actuation function. Redundancy is designed into the safety system to ensure system performance requirements are satisfied when subjected to system degradation by a single failure. The safety system provides for redundant channels originating at the sensing device (i.e., transmitter, flux input, etc.) through the signal processing and actuation electronics. These redundant channels and trains are electrically isolated and physically separated. Qualified isolation devices that have been tested to ensure functional operability, when subject to physical damage, short circuits, open circuits, or the application of credible fault voltages on the devices output terminals are used. The use of these isolation devices provides confidence that where protection signals are shared by non-safety-related systems, credible failures in the non-safety-related system cannot degrade the performance of the safety-related system or permit the fault to propagate back to the isolation device input terminals.

In addition, Siemens utilizes principles of Defense-in-Depth to provide several echelons of defense to challenges to plant safety, such that failures in equipment and human errors will not result in an undue threat to public safety. These principles are further discussed in Report No. 2267(P), "Siemens Power Corporation Methodology Report for Diversity and Defense-In-Depth. For protection and control system upgrades, Siemens will perform a detailed Defense-in-Depth

and diversity study on a plant specific application basis to address the potential for common-mode failures in digital computer-based systems. This study will be performed in accordance with NUREG/CR-6303, "Method for Performing Diversity and Defense-In-Depth Analyses of Reactor Protection Systems."

Additional details about TELEPERM XS compliance to Single-Failure Criterion are provided in Paragraph 2.7, "Fault Tolerant Features."

## 7.2 *Completion of Protective Action*

**Criterion.** The safety systems shall be designed so that, once initiated automatically or manually, the intended sequence of protective actions of the execute features shall continue until completion. Deliberate operator action shall be required to return the safety systems to normal. This requirement shall not preclude the use of equipment protective devices identified in 4.11 of the design basis or the provision for deliberate operator interventions. Seal-in of individual channels is not required.

**Compliance.** Once initiated with the TELEPERM XS safety system equipment, Reactor Trips and ESF actuations proceed to completion. Return to normal operation requires deliberate operator action to reset the reactor trip breakers. The reactor trip breakers cannot be reset while a reactor trip signal is present from the safety system. ESF actuations proceed to completion unless deliberate operator action is taken to terminate the function. This design is implemented consistent with plant specific functional logic to enable system-level protective actions to proceed to completion.

## 7.3 *Quality*

**Criterion.** Components and modules shall be of a quality that is consistent with minimum maintenance requirements and low failure rates. Safety system equipment shall be designed, manufactured, inspected, installed, tested, operated, and maintained in accordance with a prescribed quality assurance program.

**Compliance.** Components and modules of the TELEPERM XS have been evaluated to be of a quality consistent with minimum maintenance requirements and low failure rates. The TELEPERM XS safety system is designed, manufactured and tested in accordance with a rigorous quality assurance program that satisfies the requirements of 10CFR50, Appendix B. The TELEPERM XS software development process is implemented in accordance with the quality requirements of IEEE Std. 7-4.3.2 and BTP HICB-14. Qualification of software tools has been performed in accordance with the guidance presented in BTP HICB-14. Additional details about the TELEPERM XS software development and qualification process are presented in Paragraph 3.2, "Software Development Process Characteristics."

## 7.4 *Equipment Qualification*

**Criterion.** Safety system equipment shall be qualified by type test, previous operating experience, or analysis, or any combination of these three methods, to substantiate that it will be capable of meeting, on a continuing basis, the performance requirements as specified in the design basis.

**Compliance.** The TELEPERM XS safety system is environmentally and seismically qualified to ensure the system is capable of performing its designated safety functions while exposed to normal, abnormal, test, accident and post-accident environmental conditions. Mild environment qualification conforms to the guidance of IEEE Std. 323, "IEEE Standard for Qualifying Class 1E Equipment for Nuclear Power Generating Stations." EMI qualification is consistent with the guidance of EPRI TR-102323, "Guidelines for Electromagnetic Interference Testing in Power Plants." Additional details about the TELEPERM XS Equipment Qualification Program are provided in Paragraph 2.2, "Equipment Qualification."

### 7.5 *System Integrity*

**Criterion.** The safety systems shall be designed to accomplish their safety functions under the full range of applicable conditions enumerated in the design basis.

**Compliance.** The TELEPERM XS safety system has been designed and tested to confirm the equipment components and the system panels as a whole demonstrate system performance adequate to ensure completion of protective actions over the range of transient and steady-state conditions of both the energy supply and the environment. TXS system response times will be demonstrated to be consistent with plant specific accident analysis acceptance criteria. Failure modes are discussed in Paragraph 2.7, "Fault Tolerant Features."

### 7.6 *Independence*

**Criterion.**

1. Redundant portions of a safety system shall be independent of and physically separated from each other to the degree necessary to retain the capability to accomplish the safety function during and following any design basis event requiring that safety function.
2. Safety system equipment required to mitigate the consequences of a specific design basis event shall be independent of, and physically separated from, the effects of the design basis event.
3. The safety system design shall be such that credible failures in and consequential actions by other systems, shall not prevent the safety systems from meeting the requirements.

**Compliance.** The TELEPERM TXS safety system typical implementation provides for four separate and independent process channels together with two separate and independent trains of Reactor trip and ESF actuation. Separation of redundant process channels begins at the process sensors and is maintained in the field wiring, containment penetrations and the TELEPERM XS panels. Channels that provide signals for the same protective functions are located in different divisions, ensuring they will be physically separated and electrically isolated. Where redundant equipment communicates via data links, the TXS architecture has been designed to preserve independence between divisions. Electrical isolation devices are employed to preserve independence of the TXS from non-safety systems.

The TELEPERM XS safety system is designed to provide physical and electrical independence in accordance with the requirements of Regulatory Guide 1.75, "Physical Independence of Electrical Systems," and IEEE Std. 384, "IEEE Standard Criteria for Independence of Class 1E Equipment and Circuits." Communications independence is provided consistent with the guidance of IEEE Std. 7-4.3.2, Annex G.

Additional details with respect to TELEPERM XS physical, electrical, and communications independence are provided Paragraph 2.9, "Data Communications," and Section 4.0, "Independence."

### 7.7 *Capability for Test and Calibration*

**Criterion.** Capability for testing and calibration of safety system equipment shall be provided while retaining the capability of the safety systems to accomplish their safety functions. The capability for testing and calibration of safety system equipment shall be provided during power operation and shall duplicate, as closely as practicable, performance of the safety function. Testing of Class 1E systems shall be in accordance with the requirements of IEEE Std. 338-1987. Exceptions to testing and calibration during power operation are allowed where this capability cannot be provided without adversely affecting the safety or operability of the generating station. In this case:

1. appropriate justification shall be provided (for example, demonstration that no practical design exists),
2. acceptable reliability of equipment operation shall be otherwise demonstrated, and
3. the capability shall be provided while the generating station is shut down.

**Compliance.** The TELEPERM XS is designed to provide capability for test and calibration consistent with the requirements provided in Regulatory Guide 1.22, "Periodic Testing of Protection System Actuation Functions," Regulatory Guide 1.118, "Periodic Testing of Electric Power and Protection Systems," and IEEE Std. 338, "Standard Criteria for the Periodic Surveillance Testing of Nuclear Power Generating Station Safety Systems."

TXS periodic testing duplicates, as closely as practical, the overall performance required of the protection system. The capability exists to permit testing during power operation. The TXS design for testing, does not require disconnecting wires, installing jumpers, or other similar modifications of the installed equipment. Additional details are provided in Paragraph 2.5, "Testability."

The TELEPERM XS safety system is also designed to provide numerous self-diagnostic test capabilities. These self-diagnostic test capabilities are described in detail in Paragraph 2.7.1.1, "Inherent Mechanisms for Detecting and Signaling Failures," and Paragraph 2.7.1.2, "Configured Monitoring Mechanisms."

### 7.8 *Information Displays*

#### **Criterion.**

1. **Displays for Manually Controlled Actions.** The display instrumentation provided for manually controlled actions for which no automatic control is provided and that are required for the safety systems to accomplish their safety functions shall be part of the safety systems and shall meet the requirements of IEEE Std. 497-1981 [9]. The design shall minimize the possibility of ambiguous indications that could be confusing to the operator.
2. **System Status Indication.** Display instrumentation shall provide accurate, complete, and timely information pertinent to safety system status. This information shall include indication and identification of protective actions of the sense and command features and execute

features. The design shall minimize the possibility of ambiguous indications that could be confusing to the operator. The display instrumentation provided for safety system status indication need not be part of the safety systems.

3. **Indication of Bypasses.** If the protective actions of some part of a safety system have been bypassed or deliberately rendered inoperative for any purpose other than an operating bypass, continued indication of this fact for each affected safety group shall be provided in the control room.

**Compliance.** The TELEPERM XS safety system is designed to provide signals to display systems in accordance with plant specific functional logic diagrams. Outputs to non-safety displays or status indication devices are supplied through qualified isolation devices. If the TXS safety system is operated in a "Bypassed" mode, an output is provided to interface with a bypassed and inoperable status indication in accordance with the guidance of Regulatory Guide 1.47, "Bypassed and Inoperable Status Indication for Nuclear Power Plant Safety Systems."

### 7.9 *Control of Access*

**Criterion.** The design shall permit the administrative control of access to safety system equipment. These administrative controls shall be supported by provisions within the safety systems, by provision in the generating station design, or by a combination thereof.

**Compliance.** Access to the TELEPERM XS hardware is controlled via front and rear mounted cabinet doors. During normal operation, the cabinet doors are closed and locked. Door positions are monitored, allowing operators to investigate the reason for any open doors.

The service unit contains the central data of the I&C system. It is also the central means for interventions into the safety relevant software of the function processors. That's why the service unit is protected against non-authorized interventions. The installed control mechanisms assure that:

1. only authorized persons may access the service unit,
2. only the authorized interventions may be performed, and
3. protection system service unit access is restricted to a single redundancy (exceptions might be possible during plant outage).

Additional details are provided in Paragraph 2.6, "Control of System Access."

### 7.10 *Repair*

**Criterion.** The safety systems shall be designed to facilitate timely recognition, location, replacement, repair, and adjustment of malfunctioning equipment.

**Compliance.** The TELEPERM XS safety system is designed with many features to detect both hardware and software faults and assist in diagnostic and repair activities. The TXS self-test features are designed consistent with the guidance presented in BTP HICB-17, "Guidance on Self-Test and Surveillance Test Provisions." These self-diagnostic test capabilities are described in detail in Paragraph 2.7.1.1, "Inherent Mechanisms for Detecting and Signaling Failures," and Paragraph 2.7.1.2, "Configured Monitoring Mechanisms."

### 7.11 *Identification*

#### **Criterion.**

1. Safety system equipment shall be distinctly identified for each redundant portion of a safety system.
2. Components or modules mounted in equipment or assemblies that are clearly identified as being in a single redundant portion of a safety system do not themselves require identification.
3. Identification of safety system equipment shall be distinguishable from any identifying markings placed on equipment for other purposes (for example, identification of fire protection equipment, phase identification of power cables).
4. Identification of safety system equipment and its divisional assignment shall not require frequent use of reference material.

**Compliance.** The TELEPERM XS safety system can be distinctively identified in accordance with plant specific identification requirements. The preferred identification method for color coding of components, cables, and panels is available. Configuration management is used for maintaining the identification of computer software.

### 7.12 *Auxiliary Features*

**Criterion.** Auxiliary supporting features shall meet all requirements of IEEE Std. 603. Other auxiliary features that (1) perform a function that is not required for the safety systems to accomplish their safety functions, and (2) are part of the safety systems by association (that is, not isolated from the safety system) shall be designed to meet those criteria necessary to ensure that these components, equipment, and systems do not degrade the safety systems below an acceptable level.

**Compliance.** All auxiliary supporting features and other auxiliary features within the TELEPERM XS safety system panels are considered to be safety-related, and have been designed to satisfy all applicable criteria.

### 7.13 *Human Factors Considerations*

**Criterion.** Human factors shall be considered at the initial stages and throughout the design process to assure that the functions allocated in whole or in part to the human operator(s) and maintainer(s) can be successfully accomplished to meet the safety system design goals.

**Compliance.** Human factors considerations are evaluated on a plant specific basis in accordance with the individual applicant/licensee's commitments documented in Chapter 18 of the Safety Analysis Report.

### 7.14 *Reliability*

**Criterion.** For those systems for which either quantitative or qualitative reliability goals have been established, appropriate analysis of the design shall be performed in order to confirm that such goals have been achieved.

**Compliance.** The fundamental quality requirement for a safety I&C system is the reliability with which it performs its assigned safety functions. To assess this reliability, two mutually complementary methods are in standard use in Germany. These two methods are the probabilistic and the deterministic reliability analysis. Probabilistic analysis is used to quantify the reliability, with the "non-availability on demand" used as the standard measure of this. This term is defined as the probability of a given system not being able to perform its safety function when it is called upon to operate. This quantification of the quality characteristics is used as a yardstick for assessing different equipment designs.

Practical determination of reliability requires suitable modeling of the circumstances that could cause a system to lose its ability to perform the safety function demanded of it. Probabilistic analysis essentially assesses the physical and chemical aging phenomena which can lead to degradation of system characteristics with the passage of time. If the system characteristic affected is needed for performance of the safety function, the system is then no longer able to perform its intended function properly. Safety assessment of these "aging-related failures" requires qualitative analysis of the failure modes and effects, and a quantitative analysis of failure frequency. With respect to qualitative analysis, this report describes the system characteristics relevant to failure behavior. The fault and failure effects are then analyzed on the basis of the system characteristics. On the basis of this analysis it is then shown how the choice of suitable hardware architectures can have a positive impact on the response to faults and failures.

Deterministic analysis serves to assess design errors. German standards stipulate that equipment with high importance to safety must perform its task even if a plausible design error is assumed. This report shows the system characteristics that delimit the confinement area for a design error and the architecture features required for a safety I&C system to permit postulated design errors to be accommodated.

Further details are available in Paragraph 2.4, "Reliability."

### 7.15 *Automatic Control*

**Criterion.** Means shall be provided to automatically initiate and control all protective actions except for plant conditions during which manual control is permitted. The safety system design shall be such that the operator is not required to take any action prior to the time and plant conditions following the onset of each design basis event. At the option of the safety system designer, means may be provided to automatically initiate and control those protective actions required for each design basis event.

**Compliance.** The TELEPERM XS is designed to work in cooperation with the plant specific functional logic to automatically initiate and execute protective action, with precision and reliability, for the range of conditions specified. A plant specific evaluation is performed to ensure setpoints, margins, errors, and response times are enveloped by safety analysis assumptions.

## 7.16 **Manual Control**

### **Criterion.**

1. Means shall be provided in the control room to implement manual initiation at the division level of the automatically initiated protective actions. The means provided shall minimize the number of discrete operator manipulations and shall depend on the operation of a minimum of equipment.
2. Means shall be provided in the control room to implement manual initiation and control of the protective actions required to detect and mitigate a design basis event for those variables which were not selected for automatic control.
3. Means shall be provided to implement the manual actions necessary to maintain safe conditions after the protective actions are completed. The information provided to the operators, the actions required of these operators, and the quantity and location of associated displays and controls shall be appropriate for the time period within which the actions shall be accomplished and the number of available qualified operators. Such displays and controls shall be located in areas that are accessible, located in an environment suitable for the operator, and suitably arranged for operator surveillance and action.

**Compliance.** The TELEPERM XS is designed to work in cooperation with the plant specific functional logic requirements for manual controls. Manual controls are typically provided to enable the operator to initiate protective actions at the division or system level, as well as for individual components. Safety-related displays provide the operator with the information necessary to manually perform reactor trip, ESF actuation, post accident monitoring or safe shutdown functions. Failure in the automatic system does not prevent manual actuation of the protective functions.

## 7.17 **Interaction Between the Sense and Command Features and Other Systems**

**Criterion.** Where a single credible event, including all direct and consequential results of that event, can cause a non-safety system action that results in a condition requiring protective action and can concurrently prevent the protective action in those sense and command feature channels designated to provide principal protection against the condition, one of the following requirements shall be met:

1. Alternate channels not subject to failure resulting from the same single event shall be provided to limit the consequences of this event to a value specified by the design basis. Alternate channels shall be selected from the following:
  - a. Channels that sense a set of variables different from the principal channels.
  - b. Channels that use equipment different from that of the principal channels to sense the same variable.
  - c. Channels that sense a set of variables different from those of the principal channels using equipment different from that of the principal channels.
2. Equipment not subject to failure caused by the same single credible event shall be provided to detect the event and limit the consequences to a value specified by the design bases. Such equipment is considered a part of the safety system.

Provisions shall be included so that the requirements in stated above can be met in conjunction with the requirements of IEEE 603, Section 6.7, if a channel is in maintenance bypass. These provisions include reducing the required coincidence, defeating the non-safety system signals taken from the redundant channels, or initiating a protective action from the bypassed channel.

**Compliance.** The TELEPERM XS and TELEPERM XP I&C systems, together with the plant specific functional logic requirements, use a number of strategies to ensure a single credible failure, will not result in a non-safety system action causing a condition requiring protective action and concurrently prevent the protective action in those channels designated to provide protection against the condition. These strategies include the following:

1. Isolating the protection system from channel failure by providing additional redundancy.
2. Isolating the control system from channel failure by using data validation techniques to select a valid signal for control system actuation.
3. Electrical isolation techniques to prevent credible faults from propagating to redundant channels.

Additional details with respect to TELEPERM XS physical, electrical, and communications independence are provided Paragraph 2.9, "Data Communications," and Section 4.0, "Independence." Additional details on Failure modes are provided in Paragraph 2.7, "Fault Tolerant Features."

#### 7.18 *Derivation of System Inputs*

**Criterion.** To the extent feasible and practical, sense and command feature inputs shall be derived from signals that are direct measures of the desired variables as specified in the design basis.

**Compliance.** The TELEPERM XS safety system is designed to process input signals specified by plant specific functional logic diagrams. The system inputs are typically derived from signals that are direct measures (e.g., neutron flux, pressure, temperature, flow, and level) of the desired variables. Plant specific evaluations are performed to verify the characteristics (e.g., range, accuracy, resolution, response time, sample rate) of the instruments that produce the protection system inputs are consistent with the analysis provided in Chapter 15 of the Safety Analysis Report.

#### 7.19 *Operating Bypasses*

**Criterion.** Whenever the applicable permissive conditions are not met, a safety system shall automatically prevent the activation of an operating bypass or initiate the appropriate safety function(s). If plant conditions change so that an activated operating bypass is no longer permissible, the safety system shall automatically accomplish one of the following actions:

1. Remove the appropriate active operating bypass(es).
2. Restore plant conditions so that permissive conditions once again exist.
3. Initiate the appropriate safety function(s).

**Compliance.** The TELEPERM XS safety system is designed to implement requirements identified by plant specific functional logic diagrams. Typically, the functional logic diagrams

specify several operating bypasses that automatically block certain protective actions that would otherwise prevent certain modes of operation, such as startup. The operating bypasses are automatically removed when plant conditions change to an operating mode in which the protective actions are required to be operable, in order to mitigate consequences of a design basis event.

### 7.20 *Maintenance Bypass*

**Criterion. Maintenance Bypass.** Capability of a safety system to accomplish its safety function shall be retained while sense and command features equipment is in maintenance bypass. During such operation, the sense and command features shall continue to meet the requirements of 5.1 and 6.3.

**EXCEPTION:** One-out-of-two portions of the sense and command features are not required to meet Sections 5.1 and 6.3 of IEEE Std. 603, when one portion is rendered inoperable, provided that acceptable reliability of equipment operation is otherwise demonstrated (that is, that the period allowed for removal from service for maintenance bypass is sufficiently short to have no significantly detrimental effect on overall sense and command features availability).

**Compliance.** The TELEPERM XS safety system is designed with the capability to permit a single channel to be maintained, and tested, during power operation, without initiating a protective action at the system level. If a channel is bypassed for any reason, a signal is provided to facilitate continuous indication of this condition. Limiting conditions for maintenance and test bypass conditions are provided in plant specific Technical Specifications. This design is implemented in accordance with the requirements of Regulatory Guide 1.47.

### 7.21 *Setpoints*

**Criterion.** The allowance for uncertainties between the process analytical limit and the device setpoint shall be determined using a documented methodology. Where it is necessary to provide multiple setpoints for adequate protection for a particular mode of operation or set of operating conditions, the design shall provide positive means of ensuring that the more restrictive setpoint is used when required. The devices used to prevent improper use of less restrictive setpoints shall be part of the sense and command features.

**Compliance.** A plant specific safety system setpoint evaluation is performed. This evaluation confirms an adequate margin exists between operating limits and setpoints, such that there is a low probability for inadvertent actuation of the system. This analysis also confirms that an adequate margin exists between setpoints and safety limits, such that the system initiates protective actions before safety limits are exceeded. This is done by a statistical accounting for uncertainties such as sensor error, rack calibration error, temperature effects, rack drift, etc. This evaluation is performed consistent with the guidance of BTP-HICB-12, "Guidance on Establishing and Maintaining Instrument Setpoints."

**8.0 Qualification Documents**

**8.1 Software Type Test**

The different phases are documented with document types, which can be found in Figure 9.1 and Figure 9.2.

Phase	Document
Concept	Concept description
Requirements definition	Requirements Specification
System test	Test specification Test report

**Figure 8.1 Type of Documents for Overall System**

Phase	Document
Requirements definition	Requirements specification
Technical design	Technical design specification
Detailed design	Detailed design specification
Implementation	Implementation specification
Test	Test specification Test report

**Figure 8.2 Type of Documents for Each Component (Software Type Test)**

**8.1.1 Overall System**

The following list shows the record codes of these documents.

Title of Document	Records Code	
	Document	Review Report
<b>Overall-System (Gesamtsystem)</b>		
Digital Safety I&C, Concept Description Part 1: Safety Report (Digitale Sicherheits-Leittechnik, Konzeptbeschreibung Teil 1: Sicherheitsbericht)	B.02.01.031	--
Assessment on the Concept of the Digital Safety I&C of Siemens/KWU (Gutachten zum Konzept der Digitalen Sicherheitsleittechnik von Siemens/KWU)	B.02.01.034	--
Overall System Requirements Specification (Rahmenlastenheft)	B.02.03.029	B.09.03.020
Description of SPACE data pattern (SPACE-Datenmodell (Beschreibung))	H.02.02.001	--
Test specification of Plant independent System Test (Testspezifikation: Anlagenunabhängiger Systemtest)	B.02.06.006	B.09.03.099

## TELEPERM XS: A Digital Reactor Protection System

Overall-System (Gesamtsystem)		
Test report of Plant independent System Test (Testbericht: Anlagenunabhängiger Systemtest)	B.02.06.007	--
Technical test report on the plant independent system test of Siemens/KWU (Technischer Prüfbericht zum Anlagenunabhängigen Systemtest von Siemens/KWU)	C.06.06.020	--

General Quality Management (Allgemeines Qualitätsmanagement)		
Quality Management KWU N (Qualitätsmanagement KWU N)	QMH 12	--
IT-Manual (IT-Handbuch)	QMH N-01	--

8.1.2 Procedures and Instructions

Procedures and Instructions (Fach- und Projektanweisungen)		
FAW 1.1: Phase model (Phasenmodell)	A.01.13.007	--
FAW 1.4: Hardware QA Plan (Hardware-QS-Plan)	A.01.13.002	--
FAW 1.5: Configuration Management Plan (Konfigurationsmanagement-Plan)	A.01.13.005	--
FAW 1.7: Safety of Information (Informationssicherheit)	A.01.13.014	--
FAW 2.1: Guidelines for Programming (Programmierrichtlinien)	A.01.13.008	--
FAW 2.2: Guidelines for Documentation (Dokumentationsrichtlinien)	A.01.13.003	--
FAW 3.3: Structure of Requirements Specifications for SW- and HW components (Inhaltsgliederung der Lastenhefte für SW- und HW-Komponenten)	A.01.13.006	--
FAW 3.4: Structure of Technical Design Specifications for SW components (Inhaltsgliederung der Pflichtenhefte für Software-Komponenten)	A.01.13.010	--
FAW 3.5: Structure of Detailed Design Specifications for SW components (Inhaltsgliederung der Designunterlagen für Software-Komponenten)	A.01.13.011	--
FAW 3.6: Structure of Implementation Specifications for SW components (Inhaltsgliederung der Implementierungsunterlagen für Software-Komponenten)	A.01.13.013	--
FAW 4.1: Tests (Tests)	A.01.13.012	--
FAW 4.2: Reviews (Reviews)	A.01.13.004	--
PA Nr. 2: Document Control of the Software Development in the Project TELEPERM XS (Unterlagenlenkung der Software-Entwicklung im Projekt TELEPERM XS)	C.05.01.008	--

8.1.3 Function Blocks

Function Blocks V2.10 (Funktionsbausteine V2.10)		
Requirements specification: Function blocks (Lastenheft Funktionsbausteine)	B.05.01.014	B.09.03.022
Technical Design specification: Function blocks (Pflichtenheft Funktionsbausteine)	H.01.01.001	B.09.03.024

## TELEPERM XS: A Digital Reactor Protection System

<b>Function Blocks V2.10 (Funktionsbausteine V2.10)</b>		
Detailed Design specification: Database definition of function blocks (Designunterlage Datenbankdefinition der Funktionsbausteine)	H.01.02.002	B.09.03.026
Detailed Design specification: Function block modules (Designunterlage Funktionsbaustein-Module)	H.01.02.003	B.09.03.036
Implementation specification: Function block modules (Implementierungsunterlage Funktionsbaustein-Module)	H.01.02.005	B.09.03.028
Global test table entries of function blocks (Globale Prüftabelleneinträge für Funktionsbausteine)	H.01.02.001	--
Implementation specification: Global type definition of function blocks (Implementierungsunterlage globale Typdefinitionen für Funktionsbausteine)	H.01.02.004	B.09.03.015
Test specification: Database definition of function blocks (Testspezifikation Datenbankdefinition der Funktionsbausteine)	H.01.03.001, 005, 009, 013	B.09.03.027
Test report: Database definition of function blocks (Testbericht Datenbankdefinition der Funktionsbausteine)	H.01.03.002, 006, 010, 014	--
Test specification: Function block modules (Testspezifikation Funktionsbaustein-Module)	H.01.03.003, 007, 011, 015	B.09.03.023
Test report: Function block modules (Testbericht Funktionsbaustein-Module)	H.01.03.004, 008, 012, 016	--
Development documentation of function block ¼ (Entwicklungsdokumentation für Funktionsbaustein ¼)	H.01.02.006	B.09.03.102
Development documentation of function block MD-3/4 (Entwicklungsdokumentation für Funktionsbaustein MD-3/4)	H.01.02.007	B.09.03.102
Development documentation of function block 2/4 (Entwicklungsdokumentation für Funktionsbaustein 2/4)	H.01.02.008	B.09.03.102
Development documentation of function block MD-2/4 (Entwicklungsdokumentation für Funktionsbaustein MD-2/4)	H.01.02.009	B.09.03.102
Development documentation of function block FSSA (Entwicklungsdokumentation für Funktionsbaustein FSSA)	H.01.02.010	B.09.03.102
Development documentation of function block FSSB (Entwicklungsdokumentation für Funktionsbaustein FSSB)	H.01.02.011	B.09.03.102
Development documentation of function block AVERAGE (Entwicklungsdokumentation für Funktionsbaustein AVERAGE)	H.01.02.012	B.09.03.102
Development documentation of function block MD-AVERAGE (Entwicklungsdokumentation für Funktionsbaustein MD-AVERAGE)	H.01.02.013	B.09.03.102
Development documentation of function block A-MBU (Entwicklungsdokumentation für Funktionsbaustein A-MBU)	H.01.02.014	B.09.03.102
Development documentation of function block MD-A-MBU (Entwicklungsdokumentation für Funktionsbaustein MD-A-MBU)	H.01.02.015	B.09.03.102
Development documentation of function block AU-OUTPUT (Entwicklungsdokumentation für Funktionsbaustein AU-OUTPUT)	H.01.02.016	B.09.03.102
Development documentation of function block INTEGRAT (Entwicklungsdokumentation für Funktionsbaustein INTEGRAT)	H.01.02.017	B.09.03.102
Development documentation of function block TBA (Entwicklungsdokumentation für Funktionsbaustein TBA)	H.01.02.018	B.09.03.102
Development documentation of function block TBB (Entwicklungsdokumentation für Funktionsbaustein TBB)	H.01.02.019	B.09.03.102
Development documentation of function block KENNLIN (Entwicklungsdokumentation für Funktionsbaustein KENNLIN)	H.01.02.020	B.09.03.102
Development documentation of function block 2.MIN (Entwicklungsdokumentation für Funktionsbaustein 2.MIN)	H.01.02.024	B.09.03.102
Development documentation of function block 2.MAX (Entwicklungsdokumentation für Funktionsbaustein 2.MAX)	H.01.02.025	B.09.03.102

## TELEPERM XS: A Digital Reactor Protection System

<b>Function Blocks V2.10 (Funktionsbausteine V2.10)</b>		
Development documentation of function block STAB-AS (Entwicklungsdokumentation für Funktionsbaustein STAB-AS)	H.01.02.021	B.09.03.102
Development documentation of function block A-PARAM (Entwicklungsdokumentation für Funktionsbaustein A-PARAM)	H.01.02.026	B.09.03.102
Development documentation of function block COUNTER (Entwicklungsdokumentation für Funktionsbaustein COUNTER)	H.01.02.027	B.09.03.102
Development documentation of function block SOLLWERT (Entwicklungsdokumentation für Funktionsbaustein SOLLWERT)	H.01.02.028	B.09.03.102
Development documentation of function block B-PARAM (Entwicklungsdokumentation für Funktionsbaustein B-PARAM)	H.01.02.029	B.09.03.102
Development documentation of function block AU-INPUT (Entwicklungsdokumentation für Funktionsbaustein AU-INPUT)	H.01.02.030	B.09.03.102
Development documentation of function block A-MATRIX (Entwicklungsdokumentation für Funktionsbaustein A-MATRIX)	H.01.02.031	B.09.03.102
Development documentation of function block MD-A-MATRIX (Entwicklungsdokumentation für Funktionsbaustein MD-A-MATRIX)	H.01.02.032	B.09.03.102
Development documentation of function block B-MATRIX (Entwicklungsdokumentation für Funktionsbaustein B-MATRIX)	H.01.02.033	B.09.03.102
Development documentation of function block MD-B-MATRIX (Entwicklungsdokumentation für Funktionsbaustein MD-B-MATRIX)	H.01.02.034	B.09.03.102
Development documentation of function block SUM-BIN (Entwicklungsdokumentation für Funktionsbaustein SUM-BIN)	H.01.02.035	B.09.03.102
Development documentation of function block MD-SUM-BIN (Entwicklungsdokumentation für Funktionsbaustein MD-SUM-BIN)	H.01.02.036	B.09.03.102
Development documentation of function block A-SWITCH-3 (Entwicklungsdokumentation für Funktionsbaustein A-SWITCH-3)	H.01.02.037	B.09.03.102
Development documentation of function block MD-A-SWITCH-3 (Entwicklungsdokumentation für Funktionsbaustein MD-A-SWITCH-3)	H.01.02.038	B.09.03.102
Development documentation of function block B-SWITCH-3 (Entwicklungsdokumentation für Funktionsbaustein B-SWITCH-3)	H.01.02.039	B.09.03.102
Development documentation of function block MD-B-SWITCH-3 (Entwicklungsdokumentation für Funktionsbaustein MD-B-SWITCH-3)	H.01.02.040	B.09.03.102
Technical test report on the type test of the software component function blocks, version 2.10, for TELEPERM XS (Technischer Prüfbericht zur Typprüfung der Software-Komponente Funktionsbausteine Produktversion 2.10 für TELEPERM XS)	C.06.06.002	--

8.1.4 Function Diagrams

<b>Function Diagram and Function Diagram Group Modules Program Structure (Programmstruktur der Funktionsplan- und Funktionsplangruppen-Module)</b>		
Requirements specification: Program Structure of FD and FDG modules (Lastenheft Programmstruktur für Funktionsplan- und Funktionsplangruppen-Module)	H.04.01.001	B.09.03.035
Technical specification: Program Structure of FD and FDG modules (Pflichtenheft Programmstruktur für Funktionsplan- und Funktionsplangruppen-Module)	H.04.01.003	B.09.03.034
Detailed design specification: Program structure of FD and FDG modules (Designunterlage Programmstruktur für Funktionsplan- und Funktionsplangruppen-Module)	H.04.02.003	B.09.03.043
Technical test report on the type test of the program structure of FD and FDG modules for TELEPERM XS (Technischer Prüfbericht zur Typprüfung der Programmstruktur für Funktionsplan- und Funktionsplangruppen-Module für TELEPERM XS)	C.06.06.003	--

## TELEPERM XS: A Digital Reactor Protection System

<b>Function Diagram and Function Diagram Group Code Generator (Codegenerator für Funktionsplan- und Funktionsplangruppen-Module)</b>		
Requirements specification: Code Generator for FD and FDG modules (Lastenheft Codegenerator für Funktionsplan- und Funktionsplangruppen-Module)	H.04.01.004	B.09.03.014
Technical specification: Code Generator for FD and FDG modules (Pflichtenheft Codegenerator für Funktionsplan- und Funktionsplangruppen-Module)	H.04.01.005	B.09.03.025
Detailed design specification: Code Generator for FD and FDG modules (Designunterlage Codegenerator für Funktionsplan- und Funktionsplangruppen-Module)	H.04.02.004	B.09.03.047
Implementation specification: Code Generator for FD and FDG modules (Implm.unterlage Codegenerator für Funktionsplan- und Funktionsplangruppen-Module)	H.04.02.005	B.09.03.048
Test specification: Code Generator for FD and FDG modules (Testspezifikation Codegenerator für Funktionsplan- und Funktionsplangruppen-Module)	H.04.03.004	B.09.03.086
Test report: Code Generator for FD and FDG modules (Testbericht Codegenerator für Funktionsplan- und Funktionsplangruppen-Module)	H.04.03.005	--
Technical test report on the type test of the code generator for FD and FDG modules for TELEPERM XS (Technischer Prüfbericht zur Typprüfung des Codegenerators für Funktionsplan- und Funktionsplangruppen-Module für TELEPERM XS)	C.06.06.004	--

8.1.5 Program Structure

<b>Program Structure of Runtime Environment (Ablaufumgebung Programmstruktur)</b>		
Requirements specification: Runtime Environment regarding subsystem control (Lastenheft Ablaufumgebung Teilsystemsteuerung)	B.05.01.011	--
Requirements specification: Runtime Environment of monitoring and service interface (Lastenheft Ablaufumgebung im Meldeinterface)	B.05.01.024	B.09.03.040
Requirements specification: Runtime Environment in the voter level (Lastenheft Ablaufumgebung in der Voterebene)	B.05.01.021	B.09.03.039
Requirements specification: Runtime Environment in acquisition computers (Lastenheft Ablaufumgebung in den Erfassungsrechnern)	B.05.01.025	B.09.03.075
Technical specification: Program Structure of the Runtime Environment (Pflichtenheft Programmstruktur der Ablaufumgebung)	H.05.01.015	B.09.03.058
Detailed design specification: Program Structure of the Runtime Environment (Designunterlage Programmstruktur der Ablaufumgebung)	H.05.02.015	B.09.03.065
Implementation specification: Program Structure of the Runtime Environment (Implementierungsunterlage Programmstruktur der Ablaufumgebung)	H.05.02.016	B.09.03.064
Test specification: Program Structure of the Runtime Environment (Testspezifikation Programmstruktur der Ablaufumgebung)	H.05.03.011	B.09.03.073
Test report Program Structure of the Runtime Environment (Testbericht Programmstruktur der Ablaufumgebung)	H.05.03.012	--
Technical test report on the type test of the program structure of the Runtime Environment of TELEPERM XS (Technischer Prüfbericht zur Typprüfung der Programmstruktur der Ablaufumgebung TELEPERM XS)	C.06.06.005	--

8.1.6 Runtime Environment

<b>Runtime Environment Code Generator (Ablaufumgebung Codegenerator)</b>		
Requirements specification: Codegenerator for the Runtime Environment (Lastenheft Codegenerator der Ablaufumgebung)	H.05.01.008	B.09.03.059
Technical specification: Codegenerator for the Runtime Environment (Pflichtenheft Codegenerator der Ablaufumgebung)	H.05.02.013	B.09.03.061

TELEPERM XS: A Digital Reactor Protection System

<b>Runtime Environment Code Generator (Ablaufumgebung Codegenerator)</b>		
Detailed design specification: Codegenerator for the Runtime Environment (Designunterlage Codegenerator der Ablaufumgebung)	H.05.02.014	B.09.03.062
Implementation specification: Codegenerator for the Runtime Environment (Implementierungsunterlage Codegenerator der Ablaufumgebung)	H.05.02.008	B.09.03.066
Test specification: Codegenerator for the Runtime Environment (Testspezifikation Codegenerator der Ablaufumgebung)	H.05.03.003	B.09.03.095
Test report: Codegenerator for the Runtime Environment (Testbericht Codegenerator der Ablaufumgebung)	H.05.03.005	--
Technical test report on the type test of the code generator for the Runtime Environment of TELEPERM XS (Technischer Prüfbericht zur Typprüfung des Codegenerators der Ablaufumgebung TELEPERM XS)	C.06.06.006	--

8.1.7 Exception Handler

<b>Exception Handler</b>		
Requirements specification: Exception Handler (Lastenheft Exception-Handler)	H.05.01.005	--
Technical specification: Exception Handler (Pflichtenheft Exception-Handler)	H.05.01.006	B.09.03.016
Detailed design specification: Exception Handler (Designunterlage Exception-Handler)	H.05.02.006	B.09.03.042
Implementation specification: Exception Handler (Implementierungsunterlage Exception-Handler)	H.05.02.007	B.09.03.103
Test specification: Exception Handler (Testspezifikation Exception-Handler)	H.05.03.001	B.09.03.041
Test report: Exception Handler (Testbericht Exception-Handler)	H.05.03.002	--
Technical test report on the type test of the software component Exception Handler of TELEPERM XS (Technischer Prüfbericht zur Typprüfung der Softwarekomponente Exception-Handler für TELEPERM XS)	C.06.06.007	--

8.1.8 I/O Driver

<b>I/O Drivers (E/A-Treiber)</b>		
Requirements specification: I/O drivers (Lastenheft E/A-Treiber)	H.05.01.011	B.09.03.044
Technical specification: I/O drivers (Pflichtenheft E/A-Treiber)	H.05.01.012	B.09.03.050
Detailed design specification: I/O drivers (Designunterlage E/A-Treiber)	H.05.01.013	B.09.03.049
Implementation specification: I/O drivers (Implementierungsunterlage E/A-Treiber)	H.05.01.014	B.09.03.082
Test specification: I/O drivers (Testspezifikation E/A-Treiber)	H.05.03.004	B.09.03.060
Test report: I/O drivers (Testbericht E/A-Treiber)	H.05.03.006	--
Technical test report on the type test of the software component I/O Drivers of TELEPERM XS (Technischer Prüfbericht zur Typprüfung der Softwarekomponente E/A-Treiber für TELEPERM XS)	C.06.06.008	--

**TELEPERM XS: A Digital Reactor Protection System****8.1.9 Self Monitoring**

<b>Self-monitoring (Selbstüberwachung)</b>		
Requirements specification: Self-monitoring (Lastenheft Selbstüberwachung)	B.02.03.030	B.09.03.037
Technical specification: Self-monitoring of safety I&C computers (Pflichtenheft Selbstüberwachung für Rechner in der digitalen Sicherheitsleittechnik)	H.05.02.009	B.09.03.089
Detailed design / implementation specification: Self-monitoring of safety I&C computers (Design-/Impl.unterlage Selbstüberwachung für Rechner in der digitalen Sicherheitsleittechnik)	H.05.02.012	B.09.03.083
Test specification: Self-monitoring of safety I&C computers (Testspezifikation Selbstüberwachung für Rechner in der digitalen Sicherheitsleittechnik)	H.05.03.007	B.09.03.084
Test report: Self-monitoring of safety I&C computers (Testbericht Selbstüberwachung für Rechner in der digitalen Sicherheitsleittechnik)	H.05.03.008	--
Technical test report on the type test of the self-monitoring of computers in the digital safety I&C system TELEPERM XS (Technischer Prüfbericht zur Typprüfung der Selbstüberwachung für Rechner in der Digitalen Sicherheitsleittechnik TELEPERM XS)	C.06.06.009	--

**8.1.10 Operating System**

<b>Operating System (Betriebssystem)</b>		
Requirements specification: TELEPERM XS operating system (Lastenheft TELEPERM XS Betriebssystem)	H.11.01.001	B.09.03.033
<b>MICROS TXS</b>		
Technical specification: MICROS (Pflichtenheft MICROS)	H.11.02.001	B.09.03.038
Detailed design specification: MICROS (Designunterlage MICROS)	H.11.02.003	B.09.03.051
Implementation specification: MICROS (Implementierungsunterlage MICROS)	H.11.02.006	B.09.03.053
Test specification: MICROS (Testspezifikation MICROS)	H.11.03.004	B.09.03.057
Test report: MICROS (Testbericht MICROS)	H.11.03.005	--
Technical test report on the type test of the operating system MICROS of TELEPERM XS (Technischer Prüfbericht zur Typprüfung des Betriebssystems MICROS für TELEPERM XS)	C.06.06.010	--

**8.1.11 NMI Handler**

<b>NMI Handler</b>		
Technical specification: NMI-Handler (Pflichtenheft NMI-Handler)	H.11.02.004	B.09.03.080
Detailed design / implementation specification.: NMI-Handler (Design-/Implementierungsunterlage NMI-Handler)	H.11.02.005	B.09.03.085
Test specification: NMI-Handler (Testunterlage NMI-Handler)	H.11.03.002	B.09.03.072
Technical test report on the type test of the TELEPERM XS operating system regarding the NMI Handler (Technischer Prüfbericht zur Typprüfung des TELEPERM XS Betriebssystems NMI-Handler)	C.06.06.011	--

TELEPERM XS: A Digital Reactor Protection System8.1.12 Diagnostic Monitor

<b>Diagnose Monitor</b>		
Technical specification: Diagnostic Monitoring (Pflichtenheft Diagnose Monitor)	H.11.02.007	B.09.03.080
Detailed design / implementation specification: Diagnostic Monitoring (Design-/Implementierungsunterlage Diagnose Monitor)	H.11.02.008	B.09.03.068
Test specification: Diagnostic Monitoring (Testspezifikation Diagnose Monitor)	H.11.03.012	B.09.03.069
Test report: Diagnostic Monitoring (Testbericht Diagnose Monitor)	H.11.03.013	--
Technical test report on the type test of the TELEPERM XS operating system regarding the Diagnostic Monitor (Technischer Prüfbericht zur Typprüfung des TELEPERM XS Betriebssystems Diagnose-Monitor)	C.06.06.012	--

8.1.13 MicroNET

<b>MicroNET</b>		
Technical specification: MicroNET (Pflichtenheft MicroNET)	H.11.02.002	B.09.03.054
Detailed design / implementation specification: MicroNET (Design-/Implementierungsunterlage MicroNET)	H.11.02.018	B.09.03.100
Test specification: MicroNET (Testspezifikation MicroNET)	H.11.03.014	B.09.03.092
Test report : MicroNET (Testbericht MicroNET)	H.11.03.015	--
Technical test report on the type test of the software component MicroNET of TELEPERM XS (Technischer Prüfbericht zur Typprüfung der Softwarekomponente MicroNET für TELEPERM XS)	C.06.06.013	--

<b>MicroNET L2</b>		
Requirements specification: MicroNET-L2 (L2-CP) (Lastenheft MicroNET-L2 (L2-CP))	H.11.01.002	B.09.03.070
Technical specification: MicroNET-L2 (L2-CP) (Pflichtenheft MicroNET-L2 (L2-CP))	H.11.02.019	B.09.03.071
Detailed design / implementation specification: MicroNET-L2 (L2-CP) (Design-/Implementierungsunterlage MicroNET-L2 (L2-CP))	H.11.02.021	B.09.03.101
Test specification: MicroNET-L2 (L2-CP) (Testspezifikation MicroNET-L2 (L2-CP))	H.11.03.018	B.09.03.096
Test report: MicroNET-L2 (L2-CP) (Testbericht MicroNET-L2 (L2-CP))	H.11.03.019	--
Technical test report on the type test of the software components MicroNET-L2 and MicroNET-L2-CP of TELEPERM XS (Technischer Prüfbericht zur Typprüfung der Softwarekomponenten MicroNET-L2 und MicroNET-L2-CP für TELEPERM XS)	C.06.06.014	--

8.1.14 SPC1

<b>SPC1</b>		
Test strategy for the qualification of the software of the CP486 (Prüfstrategie für die Qualifizierung der Software des CP486)	H.11.03.011	--
Technical specification: CP486 for MicroNET (Pflichtenheft CP486 für MicroNET)	H.11.02.010	B.09.03.054
Detailed design specification: CP486 for MicroNET (Designunterlage CP486 für MicroNET)	H.11.02.012	--

TELEPERM XS: A Digital Reactor Protection System

<b>SCP1</b>		
Test specification: CP486 for MicroNET (Prüfspezifikation CP486 für MicroNET)	H.11.03.006	B.09.03.063
Test report: CP486 for MicroNET (Testbericht CP486 für MicroNET)	H.11.03.007	--
Test report: Product test of the CP486 for MicroNET (repeated test) (Testbericht Produkttest CP486 für MicroNET (Wiederholungstest))	H.11.03.020	--
Technical test report on the type test of the software component Protocol Handler and Realtime Transport System RTS of the SCP1 of TELEPERM XS (Technischer Prüfbericht zur Typprüfung der Softwarekomponente Protocolhandler und Realzeit Transportsystem RTS des SCP1 für TELEPERM XS)	C.06.06.015	--

8.1.15 HOT

<b>HOT</b>		
Technical specification: Hardware Organization Tool HOT (Pflichtenheft Hardware Organization Tool HOT)	H.11.02.013	B.09.03.090
Detailed design / implementation specification: Hardware Organization Tool HOT (Design-/Implementierungsunterlage Hardware Organization Tool HOT)	H.11.02.020	B.09.03.081
Test specification: Hardware Organization Tool HOT (Testspezifikation Hardware Organization Tool HOT)	H.11.03.016	B.09.03.093
Test report: Hardware Organization Tool HOT (Testbericht Hardware Organization Tool HOT)	H.11.03.017	--
Technical test report on the type test of the Hardware Organization Tool HOT of TELEPERM XS (Technischer Prüfbericht zur Typprüfung des Hardware Organisation Tool HOT für TELEPERM XS)	C.06.06.016	--

<b>HOT-Wrapper</b>		
Development documentation: HOT Wrapper (Entwicklungsdokumentation HOT-Wrapper)	H.11.02.023	B.09.03.106
Test report: HOT-Wrapper (Testbericht: HOT-Wrapper)	H.11.03.03	--
Technical test report on the type test of the HOT Wrapper of TELEPERM XS (Technischer Prüfbericht zur Typprüfung des HOT-Wrapper für TELEPERM XS)	C.06.06.017	--

8.1.16 Debug Services

<b>Debug Services</b>		
Technical specification: Debug Services (Pflichtenheft Debug Services)	H.11.02.014	B.09.03.077
Detailed design / implementation specification: Debug Services (Design-/Implementierungsunterlage Debug Services)	H.11.02.015	B.09.03.078
Test specification: Debug Services (Testspezifikation Debug Services)	H.11.03.001	B.09.03.079
Test report :Debug Services (Testbericht Debug Services)	H.11.03.013	--
Technical test report on the type test of the software component Debug Services of TELEPERM XS (Technischer Prüfbericht zur Typprüfung der Softwarekomponente Debug-Services für TELEPERM XS)	C.06.06.018	--

TELEPERM XS: A Digital Reactor Protection System**8.1.17 Driver 3964R**

<b>Driver 3964R</b>		
Technical specification: Driver 3964R (Pflichtenheft Treiber 3964R)	H.11.02.011	B.09.03.052
Detailed design specification: Driver 3964R (Designunterlage Treiber 3964R)	H.11.02.016	B.09.03.094
Implementation specification: Driver 3964R (Implementierungsunterlage Treiber 3964R)	H.11.02.017	B.09.03.076
Test specification: Driver 3964R (Testspezifikation Treiber 3964R)	H.11.03.009	B.09.03.087
Test report: Driver 3964R (Testbericht Treiber 3964R)	H.11.03.010	--
Technical test report on the type test of the software component Driver 3964R of TELEPERM XS (Technischer Prüfbericht zur Typprüfung der Softwarekomponente Treiber 3964R für TELEPERM XS)	C.06.06.019	--

**8.2 Hardware Type Test****8.2.1 Type Test of TELEPERM XS**

The type test of the components of the TELEPERM XS system was preferably conducted as component test. This rendered possible to prove independence of the system components of specific specified system configurations.

Those test steps to be conducted in the course of the type test for which it was required to continuously monitor the functioning of the components under exposure to test conditions, for example climatic test, mechanical test, were conducted in a test environment typical for real TELEPERM XS systems. The documents describing this test installation are part of the general documentation of the type test.

Those documents that can directly be assigned to individual components as a test basis are listed in the module-specific documentation part of this paper for the respective module.

Documents for which the German title this given in brackets are only available in German.

**8.2.1.1 General Documents for the Type Test**

<b>Report</b>	<b>Report number</b>	<b>Date</b>	<b>Author</b>
Environment requirements specification TELEPERM XS (Lastenheft Umwelt TELEPERM XS)	KWU NL-R	26.01.94	Siemens KWU
Overall test specification (Rahmenprüfspezifikation)	KWU NLL1 E/94/010	25.10.94	Siemens KWU
System data TELEPERM XS (Systemdaten TELEPERM XS)	KWU NLL1-1008-76-V2.0	10.97	Siemens KWU
Test requirements specification General part PAS1 (Prüfanforderungsspezifikation Allgemeiner Teil PAS1)	TXS-950424-PAS1	24.04.95	TÜV-Nord

TELEPERM XS: A Digital Reactor Protection System

Report	Report number	Date	Author
Test requirements specification Module-specific part PAS2 (Prüfanforderungsspezifikation Baugruppenspezifischer Teil PAS2)	TXS-960108-PAS2	08.01.96	TÜV-Nord
Test requirements specification for the practical test General part (Prüfspezifikation für die praktische Prüfung. Allgemeiner Teil)	945/K 72900/95 Revision A	24.03.97	TÜV-Rheinland ISEB
Test requirements specification for the practical test. Intermediate functional test and function monitoring (Prüfspezifikation für die praktische Prüfung. Funktionszwischenprüfung und Funktionsüberwachung)	945/K 72930/97	24.03.97	TÜV-Rheinland ISEB
Failure rate analysis Calculation of MTBF (Ausfallratenanalyse MTBF-Berechnung)	KWU NLL1 / 96 / 5047b	28.11.97	Siemens KWU
Test report on failure rate analysis General part (Prüfbericht zur Ausfallratenanalyse. Allgemeiner Teil)	POARA	13.11.97	TÜV-Nord
Requirements specification Software, Function monitoring (Lastenheft Software Funktionsüberwachung)	KWU NLL1E/95/5601/b	28.03.96	Siemens KWU
User manual on the test program for function monitoring (Anwenderbeschreibung zum Testprogramm Funktionsüberwachung)	KWU NL-R/95/021b	04.04.96	Siemens KWU
Type test report EMC test for TELEPERM XS (Typprüfbericht: EMV-Messungen an TELEPERM XS)	AUT GT 6/ 96-M190	12.11.96	Siemens AUT
Summary of the test results of the TELEPERM XS system (Zusammenfassung der Prüfergebnisse des Systems TELEPERM XS)	945/K 72999/98	18.03.98	TÜV-Rheinland ISEB
(System document list: Hardware) System-Unterlagenverzeichnis Hardware	UV_SYS.xls : U_V3.2	Oct 97	Siemens KWU

8.2.2 Module-Specific Documents for the Type Test

8.2.2.1 Processing Module

Document	Version	Date	Author
<b>Development</b>			
Requirements specification Processing module with 32 bit processor 80486 (Lastenheft Verarbeitungseinheit mit 32-Bit-Prozessor 80486)	V 2.0	26.05.92	Siemens
Design Document Design specification VE 486 (Entwurfsunterlage)	V4.0	22.09.93	Siemens

TELEPERM XS: A Digital Reactor Protection System

<b>Document</b>	<b>Version</b>	<b>Date</b>	<b>Author</b>
Design-Spezifikation VE 486)			
Technical design specification VE 486 (Pflichtenheft Verarbeitungseinheit VE 486)	2	Nov. 92	Siemens
Specification Interface bus of the TP386 and I/O bus of the TP386/VE486 (Spezifikation Interfacebus der TP386 und E/A-Bus der TP386/VE486)	5	Nov 93	Siemens
Simulation of the processor module VE 486 (Simulation der Prozessorbaugruppe VE 486)	V 1.1	Dez. 93	Siemens
<b>Reliability</b>			
Limit load analysis Number of TÜV Nord report: PO04GBA (Grenzbelastungsanalyse TÜV Nord-Bericht-Nr. PO04GBA)		06.08.96	TÜV-Nord
Failure rate analysis Test report PO04ARA (Ausfallratenberechnung Bericht zur Prüfung PO04ARA)		14.11.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification Number of TÜV Nord report: VE4-960131-PAS3 (Prüfanforderungsspezifikation TÜV Nord-Bericht-Nr.VE4-960131-PAS3)		31.01.96	TÜV-Nord
Supplement to the test specification of the practical test Number of ISEB report: 945/K 72904/95 Revision B (Erg.Prüfspezifikation Prakt.Prüfung ISEB-Bericht-Nr. 945/K 72904/95 Revision B)		26.03.97	TÜV-Rheinland
Test certificate: TXS-980318-PZ04		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1009-76-V2.0/10.97 (Bedienhandbuch )	V 2.0	10.97	Siemens KWU
<b>Manufacturing documents for the component</b>			
List of manufacturing documents UV_SVE1.xls : F_V3.2 (Fertigungsunterlagenverzeichnis UV_SVE1.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

8.2.2.2 Communication Module LEBUS

<b>Document</b>	<b>Version</b>	<b>Date</b>	<b>Author</b>
<b>Development</b>			
Requirements specification I/O module TP-KOM for TP386 respec. KOM4 for VE 486 (Lastenheft E/A-Modulbaugruppe TP-KOM für TP386 bzw. KOM4 für VE 486)	V 1.1	16.12.91	Siemens
Technical design specification Communication unit TP-KOM/KOM4 for TP 386/VE 486 (Pflichtenheft Kommunikationseinheit TP-KOM/KOM4 für TP 386/VE 486)	3	March 92	Siemens
<b>Reliability</b>			
Limit load analysis Number of TÜV Nord report: PO07GBA (Grenzbelastungsanalyse TÜV Nord-Bericht-Nr. PO07GBA)		05.11.96	TÜV-Nord
Failure rate analysis Test report PO07ARA (Ausfallratenberechnung Bericht zur Prüfung PO07ARA)		21.11.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification Number of TÜV Nord report: KOM-950330-PAS3 (Prüfanforderungsspezifikation		30.03.95	TÜV-Nord

TELEPERM XS: A Digital Reactor Protection System

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
TÜV Nord-Bericht-Nr.KOM-950330-PAS3)			
Supplement to the test specification of the practical test Number of ISEB report: 945/K 72907/95 (Erg. Prüfspezifikation Prakt. Prüfung ISEB-Bericht-Nr.:945/K 72907/95)		21.03.95	TÜV-Rheinland
Test certificate TXS-980318-PZ07		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1012-76-V2.0/10.97 (Bedienhandbuch)	V 2.0	10.97	Siemens KWU
<b>Manufacturing documents for the component</b>			
List of the manufacturing documents UV_SKO1.xls : F_V3.2 (Fertigungsunterlagenverzeichnis UV_SKO1.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

8.2.2.3 Bus Interface Module LEBUS

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Development</b>			
Requirements specification Brief requirements specification for MMC BUK 2 (Lastenheft Kurzlastenheft für MMC BUK 2)		03.04.87	Siemens
Specification Description of LEBUS (Spezifikation LEBUS-Beschreibung)	A 1.0	06.04.95	Siemens
Requirements specification Interface module BUK 2 (Pflichtenheft Kopplungsbaugruppe BUK2)		Dec.88	Siemens
<b>Reliability</b>			
Limit load analysis Number of TÜV Nord report: PO08GBA (Grenzbelastungsanalyse TÜV Nord Bericht PO08GBA)		06.08.96	TÜV-Nord
Failure rate analysis: Test report PO08ARA (Ausfallratenberechnung Bericht zur Prüfung PO08ARA)		24.11.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification Number of TÜV Nord report: BUK-960129-PAS3 (Prüfanforderungsspezifikation TÜV Nord-Bericht-Nr. BUK-960129-PAS3)		29.01.96	TÜV-Nord
Supplement to the test specification of the practical test Number of ISEB report: 945/K 72908/95 (Erg. Prüfspezifikation Prakt. Prüfung ISEB-Bericht-Nr.:945/K 72908/95)	Revision A	27.03.97	TÜV-Rheinland ISEB
Test certificate TXS-980318-PZ08		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1013-76-V2.0/10.97	V2.0	Oct 97	Siemens KWU
<b>Manufacturing documents of the component</b>			
List of manufacturing documents: UV_SBU1.xls : F_V3.2 (Fertigungsunterlagenverzeichnis UV_SBU1.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

## 8.2.2.4 Digital Input Module

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Development</b>			
Requirements specification Digital compact I/O / new design (Lastenheft Digitale Kompaktperipherie / Renovierung)	4	15.12.83	Siemens
<b>Reliability</b>			
Limit load analysis (Grenzbelastungsanalyse)	Number of TÜV Nord report: GBA_430.doc TÜV Nord Bericht GBA_430.doc)	15.11.96	TÜV-Nord
Failure rate analysis (Ausfallratenberechnung)	Test report PO09ARA Bericht zur Prüfung PO09ARA)	29.10.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification TELEPERM XS 6ES5 430-4UA13 (Prüfanforderungsspezifikation TELEPERM XS 6ES5 430-4UA13)		15.08.94	TÜV-Nord
Supplement to the test specification of the practical test Number of ISEB report: 945/K729009/94 Ergänzende Prüfspezifikation Praktische Prüfung ISEB-Bericht-Nr: 945/K729009/94	Revision A	18.03.97	TÜV-Rheinland ISEB
Test certificate	TXS-980318-PZ09	18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual	KWU NLL1-1050-76-V2.0/10.97	V2.0	10.97
<b>Manufacturing documents of the component</b>			
List of manufacturing documents: (Fertigungsunterlagenverzeichnis)	UV_S430.xls : F_V3.2 UV_S430.xls : F_V3.2)	3.2	Oct 97

## 8.2.2.5 Digital Input Module

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Development</b>			
Requirements specification Modification of the Digital Input Module E455/93/529 (Lastenheft Modifizierung Digitaleingabebaugrup E455/93/529)		07.04.93	Siemens
Technical design specification (Pflichtenheft)	V 1.0	29.06.93	Siemens
<b>Reliability</b>			
Limit load analysis (Grenzbelastungsanalyse)	Number of TÜV Nord report: GBA_430.doc TÜV Nord Bericht GBA_430.doc)	15.11.96	TÜV-Nord
Failure rate analysis (Ausfallratenberechnung)	Test report PO10ARA Bericht zur Prüfung)	19.11.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification TELEPERM XS 6ES5 430-4UA13 1K (Prüfanforderungsspezifikation TELEPERM XS 6ES5 430-4UA13 1K)		13.05.94	TÜV-Nord

## TELEPERM XS: A Digital Reactor Protection System

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>	
Supplement to the test specification for the practical test Number of ISEB report: 945/K72910/94 (Ergänzende Prüfspezifikation Praktische Prüfung ISEB-Bericht-Nr: 945/K72910/94)	Revision A	18.03.97	TÜV-Rheinland ISEB	
Test certificate TXS-980318-PZ10		18.03.98	TÜV-Nord	
<b>Function</b>				
Operating manual KWU NLL1-1050-76-V2.0/10.97	V2.0	Oct 97	Siemens KWU	
<b>Manufacturing documents of the component</b>				
List of manufacturing documents: (Fertigungsunterlagenverzeichnis	UV_S431.xls : F_V3.2 UV_S431.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

## 8.2.2.6 Digital Output Module

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>	
<b>Development</b>				
Requirements specification Digital compact I/O / new design (Lastenheft Digitale Kompaktperipherie / Renovierung)	4	15.12.83	Siemens	
<b>Reliability</b>				
Limit load analysis TÜV Nord report: TXS\58gba451.doc (Grenzbelastungsanalyse TÜV Nord Bericht TXS\58gba451.doc)		23.08.95	TÜV-Nord	
Failure rate analysis Test report PO11ARA (Ausfallratenberechnung Bericht zur Prüfung PO11ARA)		29.10.97	TÜV-Nord	
<b>Type test</b>				
Test requirements specification TELEPERM XS 6ES5 451-4UA13 (Prüfanforderungsspezifikation TELEPERM XS 6ES5 451-4UA13)		04.08.94	TÜV-Nord	
Supplement to the test specification of the practical test Number of ISEB report: 945/K729011/94 (Ergänzende Prüfspezifikation Praktische Prüfung ISEB-Bericht-Nr: 945/K729011/94)	Revision A	19.03.97	TÜV-Rheinland ISEB	
Test certificate TXS-980318-PZ11		18.03.98	TÜV-Nord	
<b>Function</b>				
Operating manual KWU NLL1-1051-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU	
<b>Manufacturing documents of the component</b>				
List of manufacturing components (Fertigungsunterlagenverzeichnis	UV_S451.xls : F_V3.2 UV_S451.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

## 8.2.2.7 Digital Output Module, Relays

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Development</b>			
Requirements specification Digital compact I/O / new design (Lastenheft Digitale Kompaktperipherie / Renovierung)	Stand 4	15.12.83	Siemens
<b>Reliability</b>			
Limit load analysis TÜV Nord report TXS\58gba458.doc		23.08.95	TÜV-Nord

<b>Document</b>	<b>Version</b>	<b>Date</b>	<b>Author</b>
(Grenzbelastungsanalyse TÜV Nord Bericht TXS\58gba458.doc)			
Failure rate analysis (Ausfallratenberechnung Test report PO12ARA Bericht zur Prüfung PO12ARA)		29.10.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification TELEPERM XS 6ES5 458-4UC11 (Prüfanforderungsspezifikation TELEPERM XS 6ES5 458-4UC11)		04.08.94	TÜV-Nord
Supplement to the test specification of the practical test Number of ISEB report: 945/K72912/94 (Ergänzende Prüfspezifikation Praktische Prüfung ISEB-Bericht-Nr: 945/K72912/94)	Revision A	19.03.97	TÜV-Rheinland ISEB
Test certificate TXS-980318-PZ11		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1052-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU
<b>Manufacturing documents of the component</b>			
List of manufacturing documents (Fertigungsunterlagenverzeichnis UV_S458.xls : F_V3.2 UV_S458.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

## 8.2.2.8 Analog Input Module, Integrating

<b>Document</b>	<b>Version</b>	<b>Date</b>	<b>Author</b>
<b>Development</b>			
Requirements specification Analog compact I/O / new design (Lastenheft Analoge Kompaktperipherie / Renovierung)	2	13.12.83	Siemens
<b>Reliability</b>			
Limit load analysis TÜV Nord report GBA_460.DOC (Grenzbelastungsanalyse TÜV Nord Bericht GBA_460.DOC)		15.11.96	TÜV-Nord
Failure rate analysis Test report PO13ARA (Ausfallratenberechnung Bericht zur Prüfung PO13ARA)		19.11.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification TELEPERM XS 6ES5 460-4UA13 (Prüfanforderungsspezifikation TELEPERM XS 6ES5 460-4UA13)		27.07.94	TÜV-Nord
Supplement to the test specification of the practical test Number of ISEB report 945/K72913/94 (Ergänzende Prüfspezifikation Praktische Prüfung ISEB-Bericht-Nr: 945/K72913/94)	Revision A	03.04.97	TÜV-Rheinland ISEB
Test certificate TXS-980318-PZ13		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1053-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU
<b>Manufacturing documents of the component</b>			
List of manufacturing documents: (Fertigungsunterlagenverzeichnis UV_S460.xls : F_V3.2 UV_S460.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

TELEPERM XS: A Digital Reactor Protection System

8.2.2.9 Analog Input Module

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Development</b>			
Requirements specification High speed analog input (Lastenheft High Speed Analogeingabe)	4	26.01.89	Siemens
Technical design specification Module 6ES5 466-3LA11 (Pflichtenheft Baugruppe 6ES5 466-3LA11)	V1.0	10.10.88	Siemens
<b>Reliability</b>			
Limit load analysis TÜV Nord report GBA_466.doc (Grenzbelastungsanalyse TÜV Nord Bericht GBA_466.doc)		15.11.96	TÜV-Nord
Failure rate analysis Test report PO14ARA (Ausfallratenberechnung Bericht zur Prüfung PO14ARA)		26.10.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification TELEPERM XS 6ES5 466-3LA11 (Prüfanforderungsspezifikation TELEPERM XS 6ES5 466-3LA11)		27.07.94	TÜV-Nord
Supplement to the test requirements specification of the practical test Number of ISEB report: 945/K72914/94 (Ergänzende Prüfspezifikation Praktische Prüfung ISEB-Bericht-Nr: 945/K72914/94)	Revision A	04.04.97	TÜV-Rheinland ISEB
Test certificate TXS-980318-PZ14		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1054-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU
<b>Manufacturing documents of the components</b>			
List of manufacturing documents UV_S466.xls : F_V3.2 (Fertigungsunterlagenverzeichnis UV_S466.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

8.2.2.10 Analog Output Module

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Development</b>			
Requirements specification Analog compact I/O, new design (Lastenheft Analoge Kompaktperipherie / Renovierung)	2	13.12.83	Siemens
<b>Reliability</b>			
Limit load analysis TÜV Nord report GBA_470.doc (Grenzbelastungsanalyse TÜV Nord Bericht GBA_470.doc)		15.11.96	TÜV-Nord
Failure rate analysis Test report PO15ARA (Ausfallratenberechnung Bericht zur Prüfung PO15ARA)		01.12.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification TELEPERM XS 6ES5 470-4UA12 (Prüfanforderungsspezifikation TELEPERM XS 6ES5 470-4UA12)		09.08.94	TÜV-Nord

TELEPERM XS: A Digital Reactor Protection System

<b>Document</b>	<b>Version</b>	<b>Date</b>	<b>Author</b>
Supplement to the test specification of the practical test Number of ISEB report 945/K72915/94 (Ergänzende Prüfspezifikation Praktische Prüfung ISEB-Bericht-Nr: 945/K72915/94)	Revision A	08.04.97	TÜV-Rheinland ISEB
Test certificate TXS-980318-PZ15		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1055-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU
<b>Manufacturing list of the component</b>			
List of manufacturing documents UV_S470.xls : F_V3.2 (Fertigungsunterlagenverzeichnis UV_S470.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

8.2.2.11 Counter Module

<b>Document</b>	<b>Version</b>	<b>Date</b>	<b>Author</b>
<b>Development</b>			
Requirements specification WF706 with analog outputs (Lastenheft WF706 mit Analogausgängen)		23.08.94	Siemens
Technical design specification WF706C Positioning Module (Pflichtenheft WF706C Positionierbaugruppe)	2.1	17.01.95	Siemens
<b>Reliability</b>			
Limit load analysis Report number: 945/K 73403/97 (Grenzbelastungsanalyse Bericht Nr.: 945/K 73403/97)		02.09.97	TÜV-Rheinland ISEB
Failure rate analysis Test report Positioning Module WF706C (Ausfallratenberechnung Prüfprotokoll Positionierbaugruppe WF706C)	1.0	08.02.96	Siemens
<b>Type test</b>			
Documentation of the theoretical test Number of ISEB report: 945/K 73400/97 (Dokumentation der theoretischen Prüfung ISEB-Bericht-Nr: 945/K 73400/97)		21.11.97	TÜV-Rheinland ISEB
Test specification of the practical test Number of ISEB report 945/K 73401/97 (Prüfspezifikation der praktischen Prüfung ISEB-Bericht-Nr: 945/K 73401/97)		10.10.97	TÜV-Rheinland ISEB
Documentation of the practical test Number of ISEB report: 945/K 73402/97 (Dokumentation der praktischen Prüfung ISEB-Bericht-Nr: 945/K 73402/97)		21.11.97	TÜV-Rheinland ISEB
Test certificate 945/K 734/97		20.01.98	TÜV-Rheinland ISEB
<b>Function</b>			
Operating manual KWU NLL1-1041-76-V1.0/10.97	V 1.0	Oct 97	Siemens KWU
<b>Manufacturing documents of the components</b>			
List of manufacturing documents UV_S706.xls : F_V2.0 (Fertigungsunterlagenverzeichnis UV_S706.xls : F_V2.0)	2.0	Nov 97	Siemens KWU

8.2.2.12 Relay Module

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Development</b>			
Requirements specification (Lastenheft) Relay Module Koppelrelaisbaugruppe)	V 1.1	02.02.96	Siemens KWU
Technical design specification (Pflichtenheft) Relay Module Koppelrelaisbaugruppe)	V 1.2	24.07.96	Siemens
<b>Reliability</b>			
Limit load analysis (Grenzbelastungsanalyse)	V 1.0	19.12.96	Siemens
Failure rate analysis (Ausfallratenberechnung)	V 1.0	24.07.96	Siemens
<b>Type test</b>			
Documentation of the theoretical test Number of ISEB report 945/K 73600/97 (Dokumentation der theoretischen Prüfung ISEB-Bericht-Nr: 945/K 73600/97)		01.07.97	TÜV-Rheinland ISEB
Test specification of the practical test Number of ISEB report 945/K 73601/97 (Prüfspezifikation der praktischen Prüfung ISEB-Bericht-Nr: 945/K 73601/97)		10.10.96	TÜV-Rheinland ISEB
Documentation of the practical test Number of ISEB report: 945/K 73602/97 (Dokumentation der praktischen Prüfung ISEB-Bericht-Nr: 945/K 73602/97)		01.07.97	TÜV-Rheinland ISEB
Test certificate 945/K 736/97		01.07.96	TÜV-Rheinland ISEB
<b>Function</b>			
Data sheet	V 1.4	27.06.97	Siemens KWU
<b>Manufacturing documents of the component</b>			
List of manufacturing documents (Fertigungsunterlagenverzeichnis)	5	27.06.97	Siemens KWU

8.2.2.13 Communication Processor

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Development</b>			
Requirements specification Interface Module LAx for the VE 486/TP-MSR (Lastenheft Ankoppelbaugruppe LAx für VE 486/TP-MSR)		Aug. 92	Siemens
Technical design specification Interface Module SINEC H1 LAx for VE486 and TP-LAx for TP-MSR Pflichtenheft Anschaltung SINEC H1 LAx für VE486 und TP-LAx für TP-MSR)		Okt. 92	Siemens
<b>Reliability</b>			
Limit load analysis (Grenzbelastungsanalyse) Number of TÜV Nord report: PO05GBA TÜV Nord Berichts-Nr PO05GBA)		08.10.96	TÜV-Nord
Failure rate analysis (Ausfallratenberechnung) Test report PO05ARA Bericht zur Prüfung PO05ARA)		26.10.97	TÜV-Nord

TELEPERM XS: A Digital Reactor Protection System

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Type test</b>			
Test requirements specification Number of TÜV Nord report CP4-960129-PAS3 (Prüfanforderungsspezifikation TÜV Nord-Bericht-Nr. CP4-960129-PAS3)		29.01.96	TÜV-Nord
Supplement to the test specification of the practical test Number of ISEB report: 945/K 72905/95 (Erg. Prüfspezifikation der Prakt. Prüfung ISEB-Bericht-Nr.: 945/K 72905/95)	Rev. B	26.03.97	TÜV-Rheinland ISEB
Test certificate TXS-980318-PZ05		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1010-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU
<b>Manufacturing documents of the component</b>			
List of manufacturing documents UV_SCP1.xls : F_V3.2 Fertigungsunterlagenverzeichnis )	3.2	Oct 97	Siemens KWU

8.2.2.14 Twin-Transceiver

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Reliability</b>			
Limit load analysis (Grenzbelastungsanalyse) Number of TÜV Nord report PO19GBA TÜV Nord Berichts-Nr PO19GBA)		06.08.96	TÜV-Nord
Failure rate analysis (Ausfallratenberechnung) Test report PO19ARA Bericht zur Prüfung PO19ARA)		24.11.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification Number of TÜV Nord report ECA-960130-PAS3 (Prüfanforderungsspezifikation TÜV Nord-Bericht-Nr ECA-960130-PAS3)		30.01.96	TÜV-Nord
Test specification of the practical test Number of ISEB report 945/K 72919/95 (Prüfspezifikation der Prakt. Prüfung ISEB-Bericht-Nr. 945/K 72919/95)	Revision A	11.04.97	TÜV-Rheinland ISEB
Test certificate TXS-980318-PZ19		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1017-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU
<b>Manufacturing documents of the component</b>			
List of manufacturing components UV_SCP1.xls : F_V3.2 (Fertigungsunterlagenverzeichnis UV_SCP1.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

8.2.2.15 Optical Transceiver

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Reliability</b>			
Limit load analysis (Grenzbelastungsanalyse) Number of TÜV Nord report PO21GBA TÜV Nord Berichts-Nr PO21GBA)		06.08.96	TÜV-Nord
Failure rate analysis (Ausfallratenberechnung) Test report PO21ARA Bericht zur Prüfung PO21ARA)		26.11.97	TÜV-Nord

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Type test</b>			
Test requirements specification Number of TÜV Nord report OTD-960130-PAS3 (Prüfanforderungsspezifikation TÜV Nord-Berichts-Nr. OTD-960130-PAS3)		30.01.96	TÜV-Nord
Test specification of the practical test Number of ISEB report 945/K 72921/95 (Prüfspezifikation Praktische Prüfung ISEB-Bericht-Nr. 945/K 72921/95)	Revision A	14.04.97	TÜV-Rheinland ISEB
Test certificate TXS-980318-PZ21		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1018-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU
<b>Manufacturing documents of the component</b>			
List of manufacturing documents (Fertigungsunterlagenverzeichnis UV_SHO1.xls : F_V3.2 UV_SHO1.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

## 8.2.2.16 Fiber-Optic Transceiver

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Reliability</b>			
Grenzbelastungsanalyse TÜV Nord Berichts-Nr PO20GBA		06.08.96	TÜV-Nord
Ausfallratenberechnung Bericht zur Prüfung PO20ARA		25.11.97	TÜV-Nord
<b>Type test</b>			
Prüfanforderungsspezifikation TÜV Nord-Berichts-Nr OYD-960130-PAS3		30.01.96	TÜV-Nord
Prüfspezifikation der praktischen Prüfung ISEB-Bericht-Nr.945/K 72920/95	Revision A	14.04.97	TÜV-Rheinland ISEB
Prüfzeugnis TXS-980318-PZ20		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1018-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU
<b>Manufacturing documents of the component</b>			
List of manufacturing documents (Fertigungsunterlagenverzeichnis UV_SHO2.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

## 8.2.2.17 Active Star Coupler

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Reliability</b>			
Limit load analysis (Grenzbelastungsanalyse Number of TÜV Nord report: PO17GBA TÜV Nord Berichts-Nr PO17GBA)		14.11.96	TÜV-Nord
Failure rate analysis (Ausfallratenberechnung Test report PO17ARA Bericht zur Prüfung PO17ARA)		27.10.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification TÜV Nord report A2D-960131-PAS3 (Prüfanforderungsspezifikation TÜV Nord-Bericht A2D-960131-PAS3)		31.01.96	TÜV-Nord

TELEPERM XS: A Digital Reactor Protection System

<b>Document</b>	<b>Version</b>	<b>Date</b>	<b>Author</b>
Supplement to the test specification of the practical test Number of ISEB report 945/K 72917/95 (Ergänzende Prüfspezifikation Praktische Prüfung ISEB-Bericht-Nr. 945/K 72917/95)	Revision A	04.04.97	TÜV-Rheinland ISEB
Test certificate TXS-980318-PZ17A		24.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1022-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU
<b>Manufacturing documents of the component</b>			
List of manufacturing documents (Fertigungsunterlagenverzeichnis UV_SHO2.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

8.2.2.18 Kommunikationsmodule L2

<b>Document</b>	<b>Version</b>	<b>Date</b>	<b>Author</b>
<b>Development</b>			
Requirements specification Communication processor module for TP-MSR2/VE 486 (Lastenheft Kommunikationsprozessormodul für TP-MSR2/VE 486)	V 1.2	04.05.93	Siemens
Technical design specification Communication processor module for TP-MSR2/VE 486 (Pflichtenheft Kommunikationsprozessormodul für TP-MSR2/VE 486)	V 1.0	13.05.93	Siemens
<b>Reliability</b>			
Limit load analysis (Grenzbelastungsanalyse Number of TÜV Nord report: PO06GBA TÜV Nord Bericht PO06GBA)		12.11.96	TÜV-Nord
Failure rate analysis (Ausfallratenberechnung Test report PO06ARA Bericht zur Prüfung PO06ARA)		13.11.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification Number of TÜV Nord report TPL-960117-PAS3 (Prüfanforderungsspezifikation TÜV Nord-Bericht-Nr. TPL-960117-PAS3)		17.01.96	TÜV-Nord
Supplement to the test specification of the practical test Number of ISEB report: 945/K 72906/95 (Erg. Prüfspezifikation Prakt. Prüfung ISEB-Bericht-Nr.: 945/K 72906/95)	Revision B	21.03.97	TÜV-Rheinland ISEB
Test certificate TXS-980318-PZ06		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1011-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU
<b>Manufacturing documents of the component</b>			
List of manufacturing documents (Fertigungsunterlagenverzeichnis UV_SL21.xls : F_V3.2 UV_SL21.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

TELEPERM XS: A Digital Reactor Protection System

8.2.2.19 SLLM L2 Link Module

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Development</b>			
Requirements specification Optical PROFIBUS systems (Lastenheft Optische PROFIBUS-Systeme)	4	20.08.93	Siemens
Technical design specification Optical PROFIBUS systems / OZD Profi xxx-a (Pflichtenheft Optische PROFIBUS-Systeme / OZD Profi xxx-a)	Version 2.1	02.08.95	Siemens
<b>Reliability</b>			
Limit load analysis (Grenzbelastungsanalyse	Report number: 945/K 73503/97 Bericht Nr.: 945/K 73503/97)	02.09.97	TÜV-Rheinland ISEB
Limit load analysis (Ausfallratenanalyse	Work report KWU NLL1/96/5048 Arbeitsbericht KWU NLL1/96/5048)	15.10.96	Siemens KWU
<b>Type test</b>			
Documentation of the theoretical test Number of ISEB report: 945/K 73500/97 (Dokumentation der theoretischen Prüfung ISEB-Bericht-Nr: 945/K 73500/97)		21.11.97	TÜV-Rheinland ISEB
Test specification of the practical test Number of ISEB report: 945/K 73501/97 (Prüfspezifikation der praktischen Prüfung ISEB-Bericht-Nr: 945/K 73501/97)		24.11.97	TÜV-Rheinland ISEB
Documentation of the practical test Number of the ISEB report 945/K 73502/97 (Dokumentation der praktischen Prüfung ISEB-Bericht-Nr: 945/K 73502/97)		21.11.97	TÜV-Rheinland ISEB
Test certificate	945/K 735/97	26.11.97	TÜV-Rheinland ISEB
<b>Function</b>			
Operating manual	KWU NLL1-1021-76-V2.0/10.97	V 2.0	Oct 97
<b>Manufacturing documents of the component</b>			
List of manufacturing components (Fertigungsunterlagenverzeichnis	UV_SLLM.xls : F_V2.0 UV_SLLM.xls : F_V2.0)	2.0	Nov 97

8.2.2.20 Subrack

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Development</b>			
Requirements specification for subrack BGT 8-1 (Lastenheft Baugruppenträger BGT 8-1)	1	28.07.92	Siemens
Requirements specification for the communication bus 32 bit for the VE486 (Lastenheft Kommunikationsbus 32 bit für VE486)	V 1.0	19.08.92	Siemens
Technical design specification for the communication bus 32 bit for the VE486 (Pflichtenheft Kommunikationsbus32 bit für VE486)	V 1.4	04.01.93	Siemens

## TELEPERM XS: A Digital Reactor Protection System

<b>Document</b>	<b>Version</b>	<b>Date</b>	<b>Author</b>
<b>Reliability</b>			
Limit load analysis (Grenzbelastungsanalyse) TÜV Nord report PO01GBA TÜV Nord Bericht)		16.11.96	TÜV-Nord
Failure rate analysis (Ausfallratenanalyse) Test report PO01ARA Bericht zur Prüfung PO01ARA)		11.11.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification Number of TÜV Nord report B3D-960112-PAS3 (Prüfanforderungsspezifikation TÜV-Nord-Bericht-Nr. B3D-960112-PAS3)		12.01.96	TÜV-Nord
Supplement to the test specification of the practical test Number of ISEB report 945/K 72901/95 (Erg. Prüfspezifikation Prakt. Prüfung ISEB-Bericht-Nr. 945/K 72901/95)	Revision A	20.03.97	TÜV-Rheinland ISEB
Test certificate TXS-980318-PZ01		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1015-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU
<b>Manufacturing documents of the component</b>			
List of manufacturing documents (Fertigungsunterlagenverzeichnis) UV_SBG1.xls : F_V3.2 UV_SBG1.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

## 8.2.2.21 Subrack

<b>Document</b>	<b>Version</b>	<b>Date</b>	<b>Author</b>
<b>Development</b>			
Requirements specification for subrack BGT 8-1 (Lastenheft Baugruppenträger BGT 8-1)	1	28.07.92	Siemens
Requirements specification for the communication bus 32 bit for the VE486 (Lastenheft Kommunikationsbus 32 bit für VE486)	V 1.0	19.08.92	Siemens
Technical design specification for the communication bus 32 bit for the VE486 (Pflichtenheft Kommunikationsbus 32 bit für VE486)	V 1.4	04.01.93	Siemens
<b>Reliability</b>			
Limit load analysis (Grenzbelastungsanalyse) TÜV Nord report PO01GBA TÜV Nord Bericht PO01GBA)		16.11.96	TÜV-Nord
Failure rate analysis (Ausfallratenanalyse) Test report PO03ARA Bericht zur Prüfung PO03ARA)		10.11.97	TÜV-Nord
<b>Type test</b>			
Test requirements specification Number of TÜV Nord report B5D-960115-PAS3 (Prüfanforderungsspezifikation TÜV-Nord-Bericht-Nr. B5D-960115-PAS3)		15.01.96	TÜV-Nord
Supplement to the test specification of the practical test Number of ISEB report 945/K 72903/95 (Erg. Prüfspezifikation Prakt. Prüfung ISEB-Bericht-Nr. 945/K 72903/95)	Revision A	20.03.97	TÜV-Rheinland ISEB
Test certificate TXS-980318-PZ02		18.03.98	TÜV-Nord
<b>Function</b>			
Operating manual KWU NLL1-1015-76-V2.0/10.97	V 2.0	Oct 97	Siemens KWU

TELEPERM XS: A Digital Reactor Protection System

---

<i>Document</i>	<i>Version</i>	<i>Date</i>	<i>Author</i>
<b>Manufacturing documents of the component</b>			
List of manufacturing documents (Fertigungsunterlagenverzeichnis)      UV_SBG2.xls : F_V3.2 UV_SBG2.xls : F_V3.2)	3.2	Oct 97	Siemens KWU

## Distribution

### Controlled Distribution

\* e-mail notification only

\*\* hard copy

#### Richland

D. J. Denver\*  
D. A. Nauman\*\*  
J. H. Nordahl\*  
C. M. Powers\*

#### Pittsburgh

W. J. Catullo\*\*  
L. E. Erin\*\*

#### Bellevue

J. M. Burnett\*

#### Roswell

W. Theis\*\*  
M. Winkler\*\*

#### Erlangen

Dr. A. Graf\*\*  
K-H Lochner\*\*

#### Offenbach

Dr. M. Stimler\*\*