ORNL/NRC/LTR-00/13

Program Title:	Qualification of Advanced Instrumentation and Control Systems (L1798)
Document Title:	Survey of Fieldbus Instrument Systems
Document Type:	Letter Report
Authors:	J. A. Mullens
Document Date:	November 2000
NRC Manager:	C. Antonescu
ORNLManager:	R. T. Wood, I&C Division

Prepared for
U.S. Nuclear Regulatory Commission
under
DOE Interagency Agreement 1886-N179-8L
NRC JCN No. L1798

Prepared by the
Instrumentation and Controls Division
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831-6010
managed by
UT-Battelle, LLC
for the
U. S. DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

CONTENTS

Survey of Fieldbus Instrument Systems

Purpose and Scope

This report documents an investigation of communications protocols that are candidates for use in safety-related digital systems. Topics of interest include noise rejection, reliability, and suitability for use in safety-related applications. Modeling tools that can be used to identify and evaluate performance characteristics of different protocols are also identified.

A limited set of protocols was selected for this study. Approximately 60 fieldbus-type communications standards are available by one count [**1**]. This report considers Foundation Fieldbus, ProfiSafe (Profibus), SafetyBUS, and some additional buses in contrast to these.

Background

Bus Evolution through Competition

The idea of standardized, open-specification communications buses for sensors goes back at least as far as Intel's BITBUS in the early 80's. In 1991, an international commission froze the de-facto BITBUS standard as IEEE-1118. Like many buses, BITBUS uses RS-485 physical media. Such buses typically allow multiple devices on a physical bus that can be several hundred meters in length, with data rates on the order of 1 Megabit per second (Mbps). Other early sensor buses include ARCNet (1977) and Interbus-S (1984). All of these buses are still supported today, along with many others. Today, there are a large number of digital communications standards for instrumentation networks. Appendix A lists sources of information for many of these systems.

These buses have a range of characteristics, which include the complexity of the communications performed, the level of information and control supported, the environment in which they can operate, and their costs. Some experts have placed these systems into three broad categories. The term *sensor bus* usually refers to a low-level network that essentially replaces analog signal wiring between simple sensors. The term *device bus* usually refers to a higher-level network that connects smart sensors and actuators, and provides diagnostic information. The term *fieldbus*, used in this context, refers to a higher-level network comparable to a distributed computing system. Used in this sense, fieldbus is characterized by a richer communication protocol and larger, more complex messages. Above this level lies the enterprise's general-purpose information network. Figure 1 shows another taxonomy [1].

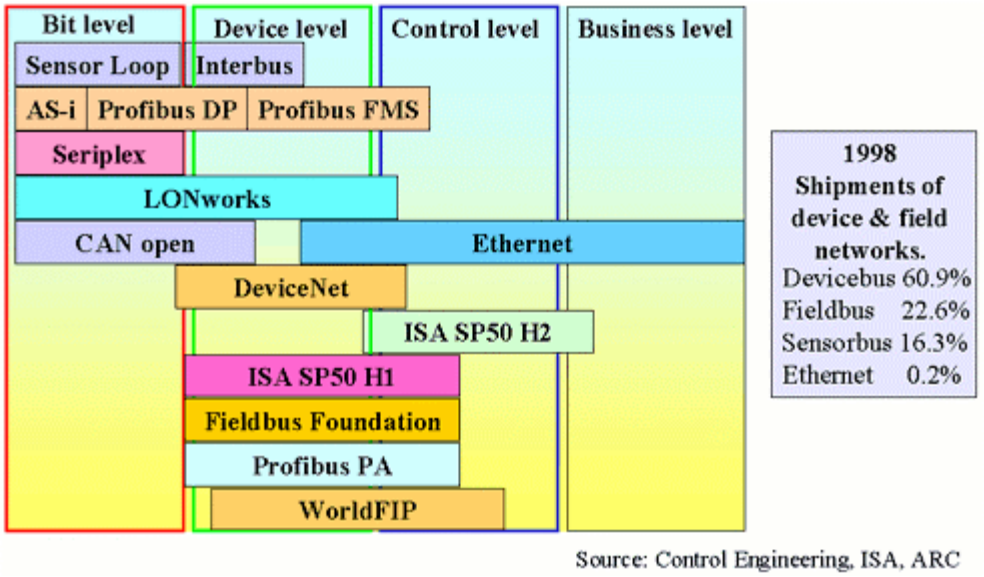

Figure 1. A classification of fieldbus systems using functional levels.

(from Control Engineering Online web site,
http://www.controleng.com/archives/2000/ctl0101.00/000100w1.htm)

It is difficult to classify these buses according to their characteristics. Most buses have a range of media, transmission speeds, bus topologies, protocol options, etc. There are few clear dividing lines separating the buses, and vendors' literature tends to be written to blur the distinctions.

An industrial process control bus has been the focus of an international development effort for some time now. In 1985 the Instrument Society of America (ISA), later joined by the International Electrotechnical Commission (IEC), started to develop a standard for two-way, multi-drop digital communications between field devices (instruments) and control systems for the process control and manufacturing control markets. The name *fieldbus* derives from this effort to define an international standard, and implies the subset of buses employing protocols based on the IEC 61158 standard (formerly IEC 1158). These buses support industrial control using existing wiring standards, intrinsic safety (explosive hazard), and reliable data transfer. The digital communications protocols have evolved to the point that some fieldbus systems can operate with measurement and control functions distributed among the field equipment, leaving only management functions to the control room equipment.

The IEC 61158 "standard" includes eight different fieldbus protocols as of January 2000:

1. Fieldbus Foundation,
2. ControlNet,
3. Profibus,
4. P-Net,
5. Fieldbus Foundation high-speed Ethernet,
6. SwiftNet,

7. WorldFIP, and
8. Interbus-S.

While it was the original intent of the standards committee to develop a single standard, the committee has added vendor-supported protocols to the group. The long delay in developing an IEC international fieldbus standard and the resulting confusion caused CENELEC[1] in 1993 to promote a European fieldbus standard. European Norm (EN) 50170 was approved in March 1996 and was based on three existing national standards: P-Net from Denmark, WorldFIP from France, and Profibus (-FMS and -DP) from Germany. Foundation Fieldbus is nominated as Part 4.

This report discusses two systems on this list (Fieldbus Foundation and Profibus) and two simpler systems (MIL-STD-1553 and SafetyBus).

There is now an industry-wide move to high-speed Ethernet (HSE) using TCP/IP and UDP protocols. This movement is fueled by the speed and decreasing costs of Ethernet. It seems likely that Ethernet and TCP/IP or UDP protocols will be used on the bus, encapsulating fieldbus messages, and running the higher-level fieldbus communications protocols. There are likely to be Ethernet versions of Foundation fieldbus, Profibus, DeviceNet, ControlNet, and LonWorks.

It seems that the industry is evolving through competition rather than standardization.

Contrast with Analog Systems

Advantages of bi-directional digital communications compared to 4-20 mA signals include:

- improved signal accuracy and reliability,
- multiple sensors per communications wire and multiple signals per sensor, and
- remote configuration and diagnostics of sensors.

Digital data transmission improves signal accuracy by eliminating some sources of noise. In addition, transmitted values can include information to indicate the quality of the measured value. A digital system can have fewer components and lower maintenance than an analog system. This in turn might result in lower probability of system failure. With continuous on-line diagnostics, the system might spend less time in an undetected failed state before repair, resulting in lower probability of failure on demand.

Some disadvantages of sensor bus protocols in comparison to 4-20 mA signals include:

- communication speed can be too low for closed loop control, depending on the protocol and bus media used,
- communication relies on software or complex hardware logic for some protocols, and
- the communications bus may be a source of common-mode failures, at least in some configurations.

[1] European Committee for Electrotechnical Standardization (web site http://www.cenelec.org).

Considerations for Nuclear Safety Applications

There are several potential applications for safety-related fieldbus in nuclear systems:

1. safety systems,
2. engineered safety systems, and
3. control systems with potential for ATWS scenarios.

The overall concern is the probability of failure on demand (PFD), which is the probability that the device will fail to perform its function when required. This can be further broken down into the PFD due to (1) the smart instrument, (2) the fieldbus network, or (3) the safety system's interface to the fieldbus. Complex fieldbus systems introduce potential faults due to their use of a communication bus.

The more sophisticated fieldbus instruments are programmable. The NRC has an additional requirement concerning the use of software in safety systems: it must be assumed that a software module will fail. This leads to requirements for diversity among safety channels in order to avoid common-mode software failures. Perhaps the underlying software issues are these:

- Any *function* complex enough to require "programming" in hardware or software is suspected of having subtle logic errors in its program design.
- Any *electronic device* complex enough to execute a program (a microprocessor, FPGA, or complex ASIC) is suspected of harboring subtle hardware errors because it cannot be exhaustively tested (the number of possible device states is too large to test).

There are examples of both types of errors occurring despite reasonable efforts to avoid them. At the heart of most smart instruments is either a communications ASIC or a general-purpose microprocessor.

The NRC also requires redundant, isolated safety channels. This would require that fieldbus devices in different safety channels not be on the same communications bus. This requirement provides protection against random failures, but the issue of common-mode (design) failures remains. A standard protection against common-mode failures is design diversity. Diversity for complex software is a reasonable requirement; diversity for fieldbus communications might also be required.

If a fieldbus technology were judged acceptable for safety applications, there would be additional issues in the mind of any vendor contemplating its use. Will there be a consensus fieldbus standard in the future? Could the nuclear certification process keep up with the evolution of fieldbus systems? How would the nuclear industry fare with adopted fieldbus systems as standards continue to evolve? How would open standards for network protocols be tested and certified for safety use?

The main topic of this report will be the safety implications of using a communications bus.

Descriptions of Selected Fieldbuses

This section briefly describes five systems. These were selected for one of two reasons:

1. they are typically used for industrial process control, or
2. they appear to be suited for real-time, safety-grade control.

Profibus DP is a popular protocol and a typical industrial fieldbus system. Its developing variant ProfiSafe is designed for safety-grade applications. Foundation Fieldbus is another industrial fieldbus system. It is intended to be the product of standards committees, rather than a vendor system adopted by a standards committee. SafetyBus is an adaptation of the CAN bus for safety-critical applications. (CAN is used in automotive, medical, and other systems). AS-Interface (Actuator Sensor Interface) is a simple bus which is developing a safety system capability. MIL-STD-1553 is a simple bus system developed for command and control applications in military aircraft. Its simplicity seems appropriate for safety systems as well, but it leaves many bus communications decisions up to the implementer.

Profibus and Foundation Fieldbus are included because their industrial process control heritage also makes them applicable to the balance of plant. Profibus's higher-level Process Automation (PA) profile and the Foundation Fieldbus are both designed for large-scale process control. However, at the level of simple devices like switches and valves, control does not require their sophisticated communications protocols: a low-level bus carrying simple commands might be all that is required. SafetyBus and MIL-STD 1553 are intermediate level buses. AS-i and Seriplex are examples of simple buses; Seriplex is briefly described at the end of this section.

Three reviewed systems are developing explicit safety-grade approaches: ProfiSafe, SafetyBus, and AS-Interface. They appear to be intended for factory machinery safety applications, with probability of failures on the order of 10^{-4}. Other systems can support safety applications through redundancy, failure isolation, and diversity.

Network topology is not discussed for individual buses. In the simplest configuration a single bus makes up a single safety channel, with a small number of instruments and/or actuators. There would be no bridging between buses, except that the safety channels must ultimately come together at the voting logic. Appropriate isolation is assumed for power supplies, EMI shielding, physical separation, etc. However, more complex topologies could be used to advantage. Redundancy can be used to increase reliability. Redundant power, redundant bus controllers, and redundant instruments might be used to protect against random faults. Some topologies also provide fault isolation.

Bus media is mentioned briefly. Wire is usually an option, with a special version (IEC 61158-2) if intrinsic (explosive) safety is required, but this restricts voltages and transmission speed. Optical fiber seems the logical choice when available, except for radiation environments. The media determines the bus bandwidth and therefore the response time of the fieldbus acting as a safety channel. The maximum transmission speed decreases with the bus length. There is a movement towards 100 Mbps / 1 Gbps

Ethernet [16]. In this case the speed of the network devices rather than the speed of the bus might determine the response time of the system.

The electronics hardware configuration will not be discussed for most systems. In general, three types of simple bus devices can be imagined:

1. Simple devices consisting of little more than a bus protocol ASIC attached to a traditional sensor, actuator, or other device.
2. Intelligent devices consisting of a bus protocol ASIC attached to a microprocessor, in turn attached to a sensor, actuator, or other device. The microprocessor firmware handles the more complex bus messages and/or processing for the sensor, actuator, etc.
3. Standard microprocessors handling the bus and its peripheral devices through firmware.

Many bus systems provide an ASIC to handle communications.

The following descriptions of fieldbus systems are a quick summaries intended only to identify general areas of future research for a variety of fieldbus systems. A detailed, in-depth review of any particular fieldbus system is beyond the scope of this report.

Profibus DP and ProfiSafe Profiles

Profibus, developed by Siemens[2] in 1994, is an international open fieldbus standard designed for machinery control, process control, and other applications [4]. Profibus-DP is a version of the network, intended for control applications, that provides time-critical communication between intelligent subsystems and distributed I/O on the lower levels of the network hierarchy. The DP profile also has optional "extended" functions that go beyond simple, cyclic data exchange on the bus. ProfiSafe is a variation of the application interface ("Application Profiles" in Profibus terminology) for safety applications [5]. It allows for failsafe devices on the network and extends the handling of communication errors and security mechanisms. It uses the DP communications profile but adds error-checking functionality at the application level (not the bus level). ProfiSafe applications can co-exist with standard DP applications on the same bus. The goal is safety level SIL3[3] (IEC61508), AK6 (German DIN V 19250); control category 4 (EN 954-1)[4].

Profibus can operate over three transmission media:

- RS-485, a standard useful in manufacturing;
- IEC 61158-2, an intrinsically safe (explosion proof) method for the process industry; and

[2] The Siemens web site has many useful documents on Profibus. A good starting point is http://www.sea.aut.siemens.com/pic/downloads-1.htm.

[3] SIL3 requires a probability of failure on demand of 10^{-4}. See ANSI/ISA-S84.01-1996 "Application of Safety Instrumented Systems for the Process Industry."

[4] Category 4 is the highest safety rating of EN954-1, a machinery control standard. It specifies that the safety-related parts of the control need to be designed so that a single fault does not compromise the safety function and will be detected at or before the next demand for the safety function. Alternatively, multiple faults must not lead to a loss of the safety function.

- Optical fiber, for improved interference immunity and large network distances.

Profibus may support 10 and 100 Mbps commercial Ethernet in the future, as will many other systems.

Protocols

Profibus DP has bus masters and slaves. Bus masters control communications while slaves are peripheral devices such as sensors and actuators. Bus masters use token passing to share bus access among themselves. Bus traffic has cyclic and acyclic messaging. Cyclic messages are typically data acquisition and control messages, and are under the control of a Class 1 Bus Master (DPM1). Acyclic messages are typically device configuration or engineering workstation messages, and are under the control of a Class 2 Bus Master (DPM2). Acyclic messages are passed on bus time not used by cyclic messages.

The timing of bus cycles synchronizes all device messaging operations. A mechanism for synchronizing the devices internal operations to a bus-wide clock (1 μs resolution) is also planned. Another synchronization mechanism is currently available: the Sync and Freeze functions. A DPM1 station broadcasts Sync and Freeze commands to tell devices to hold their current input or output values until the next command.

Network Management

Initialization consists of a parameterization and configuration phase in which the DPM1 and slave exchange data. The DPM1 sends configuration data to the slave, which compares the new configuration data with its actual configuration. Basic parameters must match.

DP extended functions allows configuring and reconfiguring devices on-the-fly, while the control loop is running normally.

Multiple DPM1s are possible and slaves can be assigned to a specific master. The slave's assigned master has direct access to the slave's inputs and outputs; other masters can read some values.

Error Detection and Handling

Bus messages have a Hamming distance of 4 due to compliance with the international standard IEC 870-5-1, special message start and end delimiters, slip-free synchronization, a parity bit, and a check byte.

Both DPM1 masters and slaves use a watchdog timer to detect failed communications. Upon failure the general reaction is to set outputs to failsafe status. Slaves that output to the plant (actuators) set their output to their failsafe value; masters that collect slave data set their data to failsafe values.

The ProfiSafe profile adds content to the DP cyclic messages at the user application level of network communications. These parameters include:

1. A status byte that indicates whether the device has recently had its parameters changed, has recognized an internal failure, has recognized a communications failure, or has switched to its failsafe state.
2. A message sequence number exchanged between devices in response and acknowledgement messages.
3. A 2/4 byte CRC value calculated over the safety data contained in the full message.

The additions, along with the existing DP mechanisms, guard against several problems.

1. If an old message is repeated, the receiver's current data can be replaced with old and unsafe data. Since data transfers are cyclic, the unsafe data will be replaced on the next cycle. Applications can delay for one cycle before activating safety measures.
2. If a message is lost or inserted, or messages arrive in a different sequence than they were sent, safety actions can be lost, countermanded, or executed in an unsafe sequence. Lost, inserted, or out-of-sequence messages are detected through message sequence numbers.
3. If a message is corrupted unsafe actions may be taken. Corrupted messages are detected by a CRC check.
4. If a message is delayed due to heavy bus traffic or a device issuing bad messages, safety actions could occur too late. Watchdog timers detect late messages; requests and acknowledgements are linked through the message sequence number.
5. Safety-related and non-safety related messages, traveling on the shared bus, are mixed. Each safety-related message has a CRC that can only be encoded if the sender-receiver are the intended devices. Devices might also have their safety-related address fixed in memory or protected against over-writing to avoid a standard device assuming the identity of a safety device.

The safety portion of the network enters a failsafe state when the count of erroneous messages exceeds a threshold.

Foundation Fieldbus

Foundation Fieldbus (FF) is an open standard introduced about 1995 [6]. It is intended to encompass as much of the industrial controls community as possible. It is similar to Profibus, WorldFIP, and other fieldbus systems, but it is more oriented towards a providing a complete distributed computing control solution than other systems.

FF allows the control application to configure the processing performed by slave devices by selecting a series of slave "function blocks" to be executed. Readers can find information on this aspect of FF in [6] and other documents on the FF web site.

Surprisingly, there is less public information on the communications details of FF than other systems. One reason might be that copies of FF specifications are sold for amounts

of several thousand dollars. Another possible reason is that the public materials which are available must cover the application layer protocols (slave function blocks, etc.) as well as the bus data layer communications protocols. We have not researched it any farther because FF is on the high end of fieldbus functionality and possibly more than is needed for a safety system application. FF and Profibus-PA (Process Automation profile) have similarities due to their common roots in the Interoperable Systems Project [18].

FF can operate over wire with speeds ranging from 32 Kbps to 2.5 Mbps, depending on the bus length. 100 Mbps Ethernet (HSE) is being pursued as a backbone bus linking workstations and field networks.

Protocols

FF is similar to Profibus in its employment of multiple bus masters and slaves, and cyclic and acyclic messaging. Link Masters control bus communications; other devices are called Basic Devices. The current controller is called the Link Active Scheduler (LAS). The bus access right is passed among devices via a token. The LAS requests data according to a cyclic schedule; the data supplier then "publishes" the data on the bus for all devices that use it. Acyclic messages are passed when a device that wants data is passed the bus token. It may then send a data request to another device; the receiving device responds when it receives the bus token in its turn.

The timing of bus cycles synchronizes all device operations. Propagation of data between devices can be ordered (by the configuration) to minimize timing skews, caused by cyclic operation, for control loops implemented across devices. Time masters synchronize time for bus devices within 1 ms.

Error Detection and Handling

A two-byte "checksum" is added to messages to allow the receiver to check for message corruption.

If the LAS fails, another Link Master will assume control of bus communications. If a device fails to respond when passed the bus token, it is taken off of the list of "live devices" maintained by the bus controller. Time Masters can be included in the fieldbus to ensure continuous operation in the event of a fault or removal of the prime device.

Devices that detect plant problems can also perform their own automatic safety actions.

SafetyBUS p

SafetyBUS p® is a new, open bus system for serial transmission of safety-related data. The standard is being developed by companies based in Europe, and uses the Programmable Safety System (PSS) controller, developed by Pilz GmbH & Co. (Germany). The PSS is a PLC for safety applications. SafetyBUS p® is the first open, safe bus system that is approved for category 4 to EN 954-1.

The bus provides up to 500 Kbits at a bus length of 100 meters.

Protocols

SafetyBUS p® [7] is a multi-master system with linear bus topology based on the popular CAN bus system (Controller Area Network). The CAN bus is simpler than most fieldbus systems and is said to offer the following advantages:

1. Communications are event-driven instead of proceeding in bus cycles, so the system responds to plant process events more quickly. Bus access is granted based on the priority assigned to each device.
2. CAN interface chips are reliable. CAN chips have been available since 1989 and are used extensively in automobiles and other applications. Over 150 million have been installed.
3. CAN has high noise immunity.

The CAN bus is designed to carry small amounts of time-critical data between devices. (DeviceNet and SDS [Smart Distributed System] are also enhancements of CAN.) SafetyBus adds a network layer that contains safety measures and network management.

The data transfer is event-driven: devices access the bus when they want to send a message, according to the priority assigned to them. (The bus access method is called CSMA/CD+CR, for Carrier Sense, Multiple Access/Collision Detection + Collision Resolution. It allows a device to detect that its message has collided with that of another device, and yet the collision is not destructive so that there is no need to restart message transmission. The lower priority device switches over to receive mode without loss of data.) Priorities, which are also the message identifier, are assigned during system design. Compliant with CAN, 8 bytes are transferred less the safety information added by SafetyBus. Data structures up to 64 KB can be transferred by splitting it into these 8-byte messages.

A bus may have one Management Device (MD), several Logic Devices (LDs), and many Server Devices (SDs). The MD performs device configuration, cyclic polling of devices to confirm they are functioning, and error diagnostics. Only the MD has access to configuration and diagnostic information on the bus. The LDs manage a group of SDs (an I/O group) and perform any processing of their data required by the application. The SDs perform the I/O for the application; an SD may be a “virtual device” actually part of an LD.

Error Detection and Handling

A 15-bit CRC checksum is transmitted with each CAN message. Network devices receive all messages, check the CRC, and acknowledge acceptance, even if they do not use the message. Each device detects message errors, determines if it or some other device is the source of the error, and increments its error counters. The result is that devices causing errors are eventually turned off. Error counters can be read to give a measure of transmission quality.

A single-bit ACK field is set by any receiver and tells the sender that at least one device has received the message.

Additional measures were added by SafetyBus to ensure detection of message errors:

1. Receiver device address is included in the message to guarantee that only the intended receiver acts on the message.
2. Delayed and lost messages are detected by timing out the acknowledgement message.
3. Repeated and inserted messages are detected by comparing message sequence numbers in the message and the acknowledgement message.
4. Corrupted message data is detected by a 16-bit CRC checksum included with the 8 bytes of message data that CAN allows. This CRC is in addition to the CAN layer's 15-bit CRC, which is applied to the entire message.

Because of CAN's event-driven communications, there can be time to re-send messages before any safety-required response time is exceeded.

The failure of a single device disables its I/O group but other I/O groups on the bus will continue functioning. However, the MD is required for the bus to continue to operate.

Network Management

The network configuration steps requires all devices to register with the MD, know the master LD, receive their I/O group assignment and other parameters, and establish connections within the I/O group.

Access to data is controlled through access rights which are established when the bus is programmed. Only the MD can perform:

1. bus configuration,
2. maintenance,
3. reads of all device error stacks,
4. reads of the manufacturer's ID devices, and
5. reads the configuration list.

Each I/O group can have associated master and slave LDs. The master LD has read and write access to the I/O points of the devices in its group; slave LDs have read access only.

Actuator Sensor Interface (AS-Interface)

AS-Interface is a low-level sensor bus that is a simple alternative to hard wiring [9]. Developed by a European consortium in 1993, it is now being extended to support safety applications.[5]

Both power and communications are carried on an unshielded 2-wire bus up to 100 meters long. Its circuit design incorporates features to minimize its sensitivity to EMI.

[5] AS-i information is available through the web site http://www.as-interface.com .

Protocols

An AS-Interface network has one master. Messages are initiated by the master to a single slave, which responds immediately. The bus master polls devices at 167 kHz and achieves a latency of 5 ms or less on a fully-loaded (30 device) network.

The system does not require any programming by the user. The instrument or controller interfaces through an AS-Interface ASIC that does not contain a processor or software.

The system is designed with simple devices in mind (process data is exchanged in 4-bit sets, within 14-bit bus messages). A future version will support transmission of 16-bit analog values, transmitted in 7 consecutive messages.

Network Management

The master has the following functions:

1. initialization of the network,
2. identification of devices,
3. acyclic setting of parameter values to the slaves,
4. diagnosis of bus and slave faults,
5. error messages to the host (application using the bus), and
6. setting of addresses in replaced slaves.

The slave's address can also be programmed with a simple hand-held device.

During operation the master uses part of each data cycle to poll unused addresses. If a device responds it is added to the active device list (it may have been unable to communicate for a brief period and dropped from the active list).

Error Detection and Handling

In all slaves and the master, each message is checked for possible transmission errors by means of six independent features (one parity bit and other unspecified features). The system can detect all single and double faults in any transmission with a certainty of 100 % and all threefold and fourfold faults with a certainty of 99.999 %. Incorrect messages are repeated during a time provided at the end of the cycle; an error during the repeat could cause the bus cycle time to be extended by 150 μs (a relatively small time when compared to the 5 ms cycle time). Repeated errors cause the master to flag a configuration error, which is passed on to the application using the bus for its own action.

Very severe levels of electro-magnetic noise (2 kV in burst test) might cause the system to stop working momentarily. The system specification requires an immediate recovery after such a condition, which requires that "lost" slaves be accepted automatically. This is tested as part of system certification.

Intelligent slaves which run self-diagnostics can signal their status through status bits. A future version will add information on peripheral faults such as short circuits, overloads, missing additional power, etc.

The specification is being enhanced with a 'Safety at Work' concept for safety applications. This specification adds a 'safety monitor' device to the bus. This device monitors safety device messages and triggers the 'emergency off' action within 35 ms if a safety device issues an alarm or the monitor detects an error within the safety device or its communications. Each message from safety devices has a field that is encoded according to an algorithm known to the safety monitor. If the encoding is not as expected, the monitor triggers the emergency stop. According to the vendor, applications to safety category 4 according to EN954-1 are realizable.

MIL-STD-1553

The MIL-STD-1553 bus standard [8] was originally developed as a command and control bus standard. It is intended to provide "very deterministic" data bus communications. It has evolved to become the predominant data bus standard for many military platforms and is being used increasingly in nonmilitary and space-based applications. The Society of Automotive Engineers (SAE) is responsible for providing maintenance and any future modifications to the standard.

The serial transmission bit rate of the bus is 1 Mbps.

Protocols

There is a single bus controller (BC) which initiates all traffic. Other devices on the bus are called bus terminals (RTs). The BC can send data to a RT, request data from a RT, or direct one RT to send data to another. Bus management commands include time synchronization, RT reset, and RT self-test. The BC can also initiate broadcast messages, requesting all RTs to receive data from the BC or another RT, or all RTs to respond to a management command. Because the bus is primarily a command and control bus, the protocol and message specifications allow for a maximum message data word package of only 64 bytes.

Error Detection and Handling

If a device does not respond to the bus controller within a specified time, the controller assumes no response will occur and initiates other traffic. Message integrity is checked via a parity bit on each 16-bit word transmitted. Other bus communications scheduling and error handling issues are decided by the BC. There is no fixed standard, so it is part of the design of the BC and the RTs.

Network Management

Network management is implementation specific. The developers of the BC and RTs must agree on the commands and status information the devices exchange.

Other Buses

Seriplex [10], which pre-dates AS-Interface, is also a simple serial bus intended to replace hard-wired connections. The bus has 2 wires for power and 2 wires for communication. It is designed with both analog and discrete (relay) devices in mind (data is exchanged in 4-bit sets). The bus can operate with or without a bus master. Latency as low as 1 ms is possible on a 30-device network. The instrument or controller interfaces through an Seriplex ASIC that does not contain a processor or software.

SwiftNet[6] is a reasonably simple serial bus that provides high capacity (85,000 16-bit data samples per second) and very accurate instrument synchronization via bus scheduling (50 us jitter). Bus interface is through an ASIC. It is one of the buses included in the IEC 61158 standard.

[6] Information available through Ship Star's web site http://www.shipstar.com.

General Discussion of Issues

Possible Safety System Configurations

The simplest fieldbus in a safety system would be used as a simple data acquisition system. Each safety channel would be a separate bus. It would deliver periodically sampled plant signals to a central safety computer, or a fault indication when it cannot deliver for any reason. Its failsafe state would be a vote for safety system actuation. If the bus failed to deliver a result on time (total failure), a watchdog timer would set its vote to failsafe. This watchdog result might be reset if the bus later resumes processing, allowing for occasional bus timing errors. The following configurations are variations on this basic idea.

1. Bus devices digitize plant signals in synchronization with the bus communication cycle. Therefore the bus speed limits the data acquisition rate and bus upsets can cause acquisition errors. The bus is simply a data acquisition system and virtually all other safety-related processing is done outside of the bus.

2. Same as (1) except that bus devices acquire and process data on their own at rates consistent with the plant signal bandwidth. However they periodically report results less frequently via the bus. For example, a bus device might detect a setpoint violation with a resolution of 1 milli-second and report that violation (with timestamp) every 0.01 second. Similarly, analog signals could be acquired at high rates, digitally filtered, and reported at the bus communications rate. In this case the devices' processing is safety related.

3. Same as (2) except the bus is event driven instead of cycle driven. This method of communications lends itself to events such as set point violations that only need to be reported when they happen. It is not so natural a solution for periodic acquisition of analog signals.

These configurations could be further enhanced by adding redundant, diverse components to each fieldbus safety channel.

Another possible configuration exchanges information between safety channels. In this configuration, each safety channel receives trip-related information from the other channels and decides for itself if the channels' votes should result in a trip (emergency stop action). All channels' decisions are then routed to a simple trip breaker that is activated when the prerequisite number of safety channels perceives a trip situation. This case differs from the previous cases in that:

1. safety channels are not entirely independent, though they are electrically isolated, and
2. the emergency action is initiated within the safety channels.

The latter difference is compatible with the fail-safe scheme used in several buses: bus failure leading to isolation of the trip device causes it to take action on its own (voting for a trip).

Digital Processing Devices

A general concern with digital devices is that the consequence of a single-bit error can be catastrophic, and there are a great many bits in data, hardware logic, and software. Consequently, error detection is employed wherever possible, diagnostics are run whenever possible, and external devices such as watchdog timers are employed to guard against other failures.

Smart devices present an additional problem when they perform complex functions using complex logic, timing, event sequencing, and interactions with external devices. Careful design is required if the device is to perform its normal function and handle all error conditions that arise. Exhaustive testing is usually not possible. Redundant, diverse safety channels are prescribed as a practical way to handle common mode failures such as logic flaws.

Naturally, fieldbus groups require vendors to submit their software and devices for certification testing. The universal proliferation of digital devices has sparked wide interest in certification for safety and mission critical software. For example, Underwriter's Laboratories has a program that reviews and certificates programmable systems that provide monitoring, control and protection in safety-related systems (Programmable Electronics/Software Safety); NASA has the Independent Verification and Validation Facility in Fairmont, West Virginia.

Bus Communication Errors

Device software errors, hardware failures, EMI, bus media degradation, and other failures can induce a variety of faults. To examine their effects on bus communications, the assumption is made that these faults result in a known set of communications errors. The ProfiSafe literature [5] lists the following as "all known possible errors that can occur during serial bus communication:"

1. repetition of a message,
2. loss of a message,
3. insertion of a message,
4. incorrect message sequence,
5. delay of a message,
6. masquerade (a message assumes the identity of another message), and
7. corrupted process data and erroneous addressing.

These errors and their treatment are discussed in the ProfiSafe description. The SafetyBus description [7] has a very similar list of concerns. This treatment seems to follow from considering the bus interface to be a fault isolation boundary. Rather than considering the details of every possible way that a bus message might be corrupted, every possible way the instrument processor/software might fail, and every result these failures might have, only the consequences at the bus interface are examined. That is,

device errors produce a limited, known set of communication errors because the device's bus interface limits the effects of faults; similarly, bus errors result in a known set of errors seen by devices. To make this fault isolation boundary effective, every effort is made to detect corrupted messages at the bus interface. The limited, known set of problems remaining is then handled through additional measures.

However, these systems also allow a mixture of safety and non-safety devices on the same bus. This is required to allow the user the advantage of a single bus for his equipment. The ProfiSafe documentation [5] refers to experience in the railway signaling technique [17]. This experience led to the ProfiSafe technique of using a standard, non-safety transmission system (Profibus-DP) with additional safety transmission functions. The requirement placed on the additional safety functions is to deterministically discover all possible faults / hazards that could be caused by the standard transmission system, or to keep the residual error (fault) probability under a certain limit.

"Random" bus transmission errors still have some chance of going undetected and producing severe faults in processor-based systems. The ProfiSafe description [5] also includes the following probabilities for bit errors for transmission systems including bus drivers.

Bit Error Probability	**Transmission System**
$>10^{-3}$	Radio link
10^{-4}	Unshielded telephone cable
10^{-5}	shielded, "twisted-pair" telephone cable
10^{-6} - 10^{-7}	Digital telephone cable of Deutsche Telekom (ISDN)
10^{-9}	Coaxial cable in locally delimited applications
10^{-12}	Fiber optics cable transmission

Table 1. Probabilities of bit errors for transmission systems.

Table taken from Dieter Conrad's book, "Datenkommunikation," 3rd edition.

Probabilistic calculations are made from data such as this to establish the bus's safety rating (e.g., IEC/ISA/AIChE Safety Integrity Level 3). The ProfiSafe specification presents these calculations. Other factors used are:

1. the probability of hazardous faults in the bus device hardware,
2. the probability of the checksum algorithm detecting message corruption,[7]
3. the probability that a hardware fault will pass a message despite a bad checksum, and
4. the time interval during which the bus controller will tolerate device errors before switching to failsafe state.

[7] The bus specification often gives its "hamming distance." The distance is the number of bits which must be wrong in order for an invalid message to be passed as valid. Values of 3 - 4 are typical.

It is assumed that any undetected bit errors will cause the bus to fail to perform its safety function.

The CAN-in-Automation (CiA) users group has produced an interesting and detailed analysis of CAN serial bus errors and remedies [14].

Safety Actuators

If the safety system actuators are controlled from the bus, total bus failure requires them to go to their failsafe states (actuate safety measures) if the bus does not recover within a short timeout period. This is true whether the safety device is casting a vote for actuation or actuating safety measures itself.

Real-time Response

It is the intent of most bus designs to provide deterministic delivery of messages between the plant and the control system. The bus affects this determinism in two ways. First, it directs devices to acquire data at some time. Second, it directs devices to send their data at some time. Timing errors (jitter) in either step are errors for time-critical messages. The buses' ability to perform well is not generally questioned.[8] While there might not be a general study testing this issue, there would be individual studies conducted by fieldbus users for their own installations. Future enhancements planned for Profibus would reduce normal bus timing jitter to less than 1 micro-second.

Since the bus is shared, communication from one device blocks all other device communications for its duration. All systems therefore must allow for some delay, even for time critical messages such as safety system alarms. Some buses limit the maximum message length to help minimize this time; a high-speed bus also helps. Some systems try to provide quick bus access to devices, several systems schedule bus access in cycles, and some systems require the bus controller to request messages without specifying how that must be done. SafetyBus is an example of a quick access (event-based communication) bus: a device can send its message as soon as the bus is clear and higher priority devices are not waiting to send their transmissions. In cyclic scheduling (Profibus and others), routine periodic messages occupy a small part of the cycle and aperiodic messages fill another part. A safety-related alarm device would periodically send its alarm status to other devices, perhaps with additional detailed information in the aperiodic part of the cycle. Note that cyclic scheduling depends on devices to perform any processing they do within the time limits imposed by the cycle. Slices of cycle time are specified during configuration. The MIL-STD-1553 bus master simply polls all devices.

The possibility of bus transmission errors complicates the real-time response issue. All systems detect corrupted messages with some probability, but recovery varies. Cyclic buses generally discard the corrupted message from the current cycle and wait for the

[8] However, see Ship Star's performance tests at http://www.shipstar.com/bus-perf/bus-perf.html, also summarized in Appendix B.

next cycle's update. Event-based communication requires the receiver to request retransmission, assuming the receiver can determine the origin of the corrupted message.

The possibility of bus and bus device faults further complicates the real time response issue. Devices will have several failure modes and associated consequences. While the systems have recovery methods for such faults, these methods can require that the bus controller first identify the faulty device before it puts the system in failsafe status. A universal technique is that non-responsive devices are detected via a communications watchdog timer; more subtle failures are not so easily detected.

According to Profibus literature [15], 5% timing jitter is possible due to control functions; one transmission failure will be corrected within 500/100µs (1.5/12Mbaud); one station failure will add a jitter of 700/300µs.

For safety systems, the obvious configuration is one isolated fieldbus network for each safety channel. The consequence of late communication in any channel would logically be that the safety channel's vote should become "trip," at least until the channel properly updates its own vote. This could be implemented using a watchdog timer: if a safety channel does not update its vote within the required period, its vote becomes "trip." This provision might handle many of the concerns about complicated timing problems in fieldbus networks.

Event Time Stamps

Safety systems need to deliver precise alarm sequence information to the control room. If plant data is acquired only on synchronized bus cycles, time stamps are essentially common to all bus devices (with some allowance for bus media transmission delays). If plant events are time stamped independently by the bus device, time must be closely synchronized across the bus. Fieldbus systems are generally designed to synchronize time among the bus devices. For example, FF synchronizes devices to within 1 ms.

Diagnostic Information

It seems possible that smart sensors would be required, as part of their safety function, to deliver diagnostic information about themselves and the signals they monitor. An example is the data quality flag accompanying measured data in Foundation Fieldbus. A device could be required to report its internal diagnostics results or factors that may effect health such as environmental temperature, low power supply voltage, or noise contamination. Most, if not all, systems report on bus errors and device problems and this data would certainly be available to the maintenance engineers and control room. Concerns about a device's software failures might be eased by diagnostic reports. Similarly, additional information on the plant signal it monitors, such as noise content, rate of change, or range observed, might also validate the proper operation of the sensor. This leads to consideration of guidance on what should be reported, when it should be reported, and what actions to take in response to problems.

Redundancy and Fault Isolation

It is generally recognized that safety systems must use redundant hardware and fault isolation to deal with random failures. However, the most effective design must consider where failures are most likely to occur. One source [11] recommends redundant hardware in this rough order of priority for computer-based (PC) systems:

1. Process sensors and actuators
2. Analog I/O devices and field-mounted electronics. (modules, buss, wiring)
3. Digital buss and connections
4. Power Supplies
5. PC Motherboard and RAM
6. Network
7. Monitors, Keyboards, Pointing Devices, Audio, Data Storage Devices.

Smar, a fieldbus supplier, has published a good description of redundancy and fault isolation techniques for Foundation Fieldbus [13]. Measures include redundant bus power supplies, bus communications controllers, and instruments, as well as network topologies that isolate faults. All buses can take advantage of redundant sensors as long as the application software knows to switch instruments in the event of a failure. Profibus, if not other buses, allows redundant bus wiring (if fiber optic segments are used).

One common fault isolation technique is to employ network topologies that prevent bus faults on one segment from disrupting other segments. Another technique uses the bus master to detect bus device failures. A common detection method is communication time outs, either on normal communications or periodic polling of devices. The device's failed state is then marked for all users of its data. Another technique is to count bus transmission errors associated with each device; a device with excessive errors is isolated [7].

Network Configuration (System Management)

Part of the attraction of fieldbus is the ease of modification. For example, adding a new sensor can be as easy as attaching the sensor to the bus and modifying the plant's control configuration data base. Nuclear safety systems clearly do not need this flexibility and it could raise safety issues that would make qualification more difficult.

Systems generally perform a configuration phase when starting. This is process of discovering devices on the bus, initializing them, and setting up bus controller schedules for communications. Configuration through the bus raises the possibility of errors during a legitimate configuration phase, and the possibility of a communications error or device fault resulting in an accidental re-initialization during normal operation. For example, Profibus-DP's extended functions allow configuring and reconfiguring devices while the bus is running its normal cycles.

In order to prevent such problems, several buses allow "read-only" device parameter settings: switches to set devices addresses and read-only memory for parameter storage.

Most systems also have a system of "access rights," which restrict access to data and functions to bus masters or other devices allowed access as part of the bus configuration.

Fieldbus systems tend to use a vendor-independent plug-and-play approach to configuring the bus. The run-time configuration of a bus can be the end product of vendor-supplied data sheets, bus device internal settings, and the user's bus device I/O configuration choices. This information might be processed using off-line software to generate the run-time configuration information. For example, FF's Device Descriptions and Profibus's Electronic Device Data Sheets (GSD files) are part of this process. It seems unlikely that this entire process will be safety qualified. Qualification might start with its final product, the generated run-time configuration, instead.

Smart fieldbus devices provide a variety of run-time functions which plant operators and engineers can access during normal operation via a workstation. Devices can transmit additional information not sent during the routine data exchange cycle. Devices might be tuned via parameter settings, transmit its status, or even download new executable code. These are handled as low-priority requests so that the normal bus cycle is not interrupted. Many of these functions would be helpful, but some would raise concerns. If a complex fieldbus system were proposed, means of selectively disabling such functions would need to be investigated. The ProfiSafe documentation warns that access to such functions through the engineer's workstation must be administratively controlled [5]. This warning probably applies to other systems as well.

Modeling Tools

Inquiries about modeling and simulation tools found very little specifically for fieldbus systems. (Bus monitors and analyzers used for bus troubleshooting are common, however.) There are several general-purpose communications network analysis tools and at least one (OPNET) has been used to simulate an automotive bus (SAE J1850).

A safety-critical instrument bus would be not be designed and operated in the same way as a general-purpose LAN. The design and implementation of an instrument bus would provide more deterministic response and minimal queuing for bus access. Modeling in the sense of statistical analysis of the bus traffic on a LAN would be less of an issue than worst-case scenarios of bus traffic during plant (or automotive system) upsets. Modeling in the sense of discrete event simulation would be appropriate for worst-case scenario analysis.

A general-purpose discrete event simulation language would be helpful in simulating fieldbus systems when no existing simulation tool has fieldbus models. There are many discrete event simulation languages. Some of the more established are GPSS/H and SLX (Wolverine Software), MODSIM (CompuWare), and SIMSCRIPT (CACI Products).

The following simulation packages might be able to perform a fieldbus simulation, but would require some work on the user's part to model the bus and devices.

OPNET Modeller (from Opnet Technologies, formerly MIL 3) has been used by Chrysler to simulate the SAE J1850 communications protocol, an industry standard that provides bus communications for electronic modules within vehicles. SAE J1850 is comparable to CAN, although CAN is also used outside the automotive industry. According to the vendor, OPNET can be used to:

- predict network parameter effects such as message priority, latency and bus utilization,
- develop and optimize new communication protocols, and
- understand dynamic network behavior such as failure and recovery scenarios.

OPNET does not include any fieldbus models in its standard model libraries, but the J1850 model is included in its contributed models. All models include source code and the user may create his own models.

COMNET III (from CompuWare) simulates communications networks. It provides models for communications and computer networks used by telephone companies, cable television broadcasters, and computer networks. It is an established package that might be extendable to fieldbus systems. COMNET is written in MODSIM and the user can use MODSIM to extend the models available. (This package was formerly distributed by from CACI Products).

SES/*workbench* (from HyPerformix) is a simulation modeling tool for hardware architecture and other complex systems. It is a commercial, "industrial strength" modeling and simulation package that models large, complex systems. It provides a visual environment for model building and simulation execution for performance analysis and functional verification. It also allows the user to add procedural code in an internal language that is a superset of C.

Conclusions

Three fieldbus systems were found for safety-grade applications:

1. ProfiSafe, an adaptation of the Profibus system,
2. SafetyBUS, based on a safety-grade PLC and the CAN bus, and
3. the “Safety at Work” enhancement of the AS-Interface bus.

All of these systems have European origins.

Development Trends

There are approximately 60 fieldbus systems in use despite attempts to develop an international standard. It seems that there is no consensus yet on the "right way" to network instruments. Instead, consortia are developing their own solutions and letting the marketplace decide the winners. At the same time, systems are developing towards some common ground: communications ASICs are common, high-speed Ethernet is being implemented, and there is more effort to provide bus timing that is precise and deterministic in order to support control loops and safety actions. Given the current state of competitive pressure among the vendors, it seems inevitable that fieldbus systems will ride the wave of development taking place in general computer systems and communications. The end result might be that fieldbus systems adopt general computer industry standards for bus communications, simply adding their own higher layers of messaging to support control applications.

Safety-critical and time-critical control applications are demanding. One reason given to forge ahead into safety-critical applications is to dispense with conventional safety systems currently installed in parallel with fieldbus systems [5]. Several systems are being enhanced specifically to support safety-critical applications (this seems to be a trend among European systems). Another approach is to install a “standard” fieldbus system with enough redundancy, fault isolation, and diversity to achieve the reliability and availability required. In this case a fieldbus system must be chosen that allows such configurations. The remaining question is whether or not the strategies developed will be sufficient for the nuclear industry. It might be that vendors, in competition with each other, will move quickly to adopt the “good enough” enhancements that their competitors are marketing for other industries, but no system emerges that the nuclear industry can use.

Nuclear Industry Applications

The nuclear industry is vulnerable to equipment obsolescence: plants operate for a long time and are costly to certify. If the days of standardized current loop instruments are numbered, the days of stable instrumentation technology may also be over. At the least, the current climate of competition in the fieldbus market guarantees rapid developments and an eventual shake-out of systems.

Certainly plant designers will see economic advantages to using fieldbus in the balance of plant systems. Given the cost of plant equipment, it seems logical that designers would

take a conservative approach and adopt proven systems for the balance-of-plant. For safety systems, the logical choices seem to be either proven systems (if any exist) or very simple systems that could be most easily certified.

Specific Issues

One looming concern is the use of complex software and complex electronic devices (processors) in safety systems. A major concern is design errors leading to common mode failures. Certification requirements for vendors' fieldbus products offer some assurance. A requirement for diversity is another safeguard against problems. The more popular fieldbus systems offer some diversity through:

1. support for several physical bus media,
2. communications ASICs from multiple vendors,
3. communications stacks from multiple vendors, and
4. smart instruments and other bus devices from multiple vendors.

Another concern is whether a fieldbus system can acquire plant data on a precise schedule and report it in time to guarantee safety actions are taken. This must be done while handling the occasional bus communications error or delays due to shared usage of the bus. This is similar to a requirement that the bus be able to perform closed-loop, time-critical control (e.g., a PID algorithm) with the functions dispersed among several bus devices so that bus communication enters into the requirement. Vendors and user groups provide some information such matters. An independent assessment would need to analyze the bus design to determine worst case scenarios then test an actual system.

Most bus systems have elaborate checks for random bus communication errors. This is particularly true of those systems that have been enhanced for safety applications. Also, bus master devices monitor slave devices for detectable errors, e.g., excessive number of badly formatted messages, no response within time out period, and messages out of sequence. The system can then take a variety of remedial actions in order to isolate faults and substitute data from a redundant device. In the event of total bus failure, safety devices on the bus respond by going to their failsafe state (voting for safety actuation). However the analysis of error handling often started by stating lists of "all possible" communications errors which could result from failures. There may well already be extensive research behind these lists, but this could be investigated further. For example, have those buses that allow a mixture of safety and non-safety devices been fully analyzed for failure modes?

Equipment redundancy and fault isolation are major requirements for highly reliable safety systems. This is recognized and these systems generally support these goals through configuration options and network topology. Of course, the methods used vary and bus selection may depend on their differences.

The plug-and-play nature of fieldbus networks is a great benefit to users who want to routinely change the network. This feature extends to run-time reconfiguration of the network (hot-swapping of devices and restart of the network under control of the bus master). Of course, this raises a concern that some failure could result in run-time mis-

configuration of a device or the network, or that the network could be accidentally restarted. Measures are used to prevent such problems, e.g., an option to set device bus address via switches rather than through bus commands, and an option to use read-only-memory to store device configuration parameters. These are prudent measures to take for a safety-critical network and should be a factor when selecting a bus.

Summary

Three fieldbus groups were found to be making specific efforts to address safety-critical applications. Other fieldbus systems address safety issues through provisions for redundancy and fault isolation. Given the current interest in fieldbus designs, the possibility of new systems and new enhancements for safety-critical applications is a distinct possibility.

The techniques proposed to handle safety-critical applications are, at least, interesting. Their area of application seems to be closer to monitoring emergency stop buttons on machinery than nuclear systems. But if they gain general acceptance it is possible that similar techniques will be proposed for some applications in nuclear plants.[9] In any case, the broader issues raised by fieldbus systems are not that different from any digital computing application that might be proposed for nuclear plants. For example, all computers contain I/O buses and some, e.g. SCSI bus, run non-trivial communications protocols.

Aside from the issues of bus communications, there is the associated issue of smart sensors. Such sensors provide additional information about the process they are monitoring and can perform self-diagnostic checks. It is clear that there are advantages here. For example, self-diagnostics reduces the amount of time a safety system spends in a failed state, increasing its availability, and increases the system's safety qualification rating (at least that is the case for the safety buses reviewed here). On the other hand, any safety performance credit claimed that is based on self-diagnostics must be evaluated.

Fieldbus is a developing technology that will be supporting industrial safety applications. Its progress could be monitored by:

- periodic surveys of fieldbus industry technology and applications,
- contacts with fieldbus groups developing safety-grade systems,
- contacts with the standards organizations evaluating the safety-grade systems, and
- gaining experience through an in-depth analysis of one or more fieldbus installations.

[9] Appendix B lists some applications of fieldbus systems.

Acronyms

ASI or AS-i	Actuator Sensor Interface
ASIC	Application Specific Integrated Circuit
ATWS	Anticipated Transient Without Scram
CAN	Controller Area Network
CRC	Cyclic Redundancy Check
DDC	Direct Digital Control
EMI	Electro-Magnetic Interference
FPGA	Field Programmable Gate Array
HMI	Human/Machine Interface
HSE	high-speed Ethernet
LAN	Local Area Network
MAP	Manufacturing Automation Protocol
MES	Manufacturing Execution System
MIS	Manufacturing Information System
MMS	Manufacturing Message Specification
ODBC	Open Data Base Connection
PC	Personal Computer
PCMCIA	Personal Computer Memory Card International Association
PFD	probability of failure on demand
PLC	Programmable Logic Controller
SCSI	Small Computer System Interface
SIL	safety integrity level
SIS	safety instrumented system
TCP/IP	Transmission Control Protocol/Internet Protocol
UDP	User Datagram Protocol

References

[1] Harrold, David, "Here We Go Again," Control Engineering, January 2000 online extra to article "Online extra to Next Generation Control System Technologies Promote Solutions," also at http://www.controleng.com/archives/2000/ctl0101.00/000100w1.htm.

[2] Hoske, Mark T., "Connect to the Benefits of Digital Industrial Networks," Control Engineering, January 1998.

[3] IEEE-SA Technical Report on Utility Communications Architecture (UCA TM) Version 2.0, IEEE-SA TR 1550-1999, The Institute of Electrical and Electronics Engineers, Inc. 3 Park Avenue, New York, NY 10016-5997, USA, 15 November 1999.

[4] PROFIBUS Technical Description, September 1999, PROFIBUS Nutzerorganisation e.V., Haid-und-Neu-Str. 7, D-76131 Karlsruhe.

[5] PROFIBUS Profile, Fail Safe with PROFIBUS (Draft), Revision 1.0, April 1999, PROFIBUS Nutzerorganisation e.V., Haid-und-Neu-Str. 7, D-76131 Karlsruhe.

[6] Foundation FieldBus Technical Overview, FD-043 Revision 2.0, Fieldbus Foundation, 9390 Research Boulevard, Suite II-250, Austin, Texas 78759.

[7] SafetyBUS p – Description, Version I, November 1999, SafetyBUS p Club International e.V. 2, Felix-Wankel-Str. 2, D-73760 Ostfildern.

[8] An Interpretation of MIL-STD-1553B, available through web site http://www.sbs.com/av_1553over.shtml.

[9] Actuator Sensor Interface Technical Overview Rev9810, Published by AS-i Trade Organization, 16101 N. 82nd Street, Suite 3B, Scottsdale, AZ 85260.

[10] Distributed, Intelligent I/O for Industrial Control and Data Acquisition … The SERIPLEX ® Control Bus, Bulletin No. 8310PD9501R4/97, Seriplex Technology Organization, Inc., Raleigh, NC, USA, April, 1997.

[11] Cawlfield, David W., "Achieving Fault-Tolerance with PC-Based Control," OMNX Open Control, Olin Corporation, Charleston, TN 37310-0248, published on web site http://www.omnx.com.

[12] Crowder, Robert S, "Fieldbus Performance," http://www.shipstar.com/bus-perf/bus-perf.html, SHIP STAR Associates, Newark, DE.

[13] Fayad, Claudio Aun and Pedro Anisio Biondo, "Reliability with Foundation Fieldbus," Tech. Papers of ISA. Networking and Communications on the Plant Floor. Technology Update LIV. ISA TECH 1999, vol 392, 10/99, pp 229-245.

[14] CAN Data Link Layer, CiA, Am Weichselgarten 26, D-91058 Erlangen, available through CiA website http://www.can-cia.de.

[15] PROFIBUS technical information II, a user FAQ list available at the Siemens web site http://www.sea.aut.siemens.com.

[16] McGilvreay, John (employed by Richard Hirschmann of America, Inc.), “Ethernet 101: Bringing the advantages of Ethernet to industrial automation,” published on http://www.plantautomation.com.

[17] European Standard prEN 50159-1 "Railway Aplications: Requirements for Safety-Related Communication in Closed Transmission Systems."

[18] Waterbury, Bob, “Fieldbus for Pragmatists,” Control, February 1999, pages 42-51.

Vendors

CACI Products Company
3333 N. Torrey Pines Court
La Jolla, CA 92037 USA
Phone: +1 (619) 457-9681

CENELEC
35 Rue de Stassart
B-1050
Brussels, Belgium
Tel : +32.(0)2.519.68.71
http://www.cenelec.org/

Compuware Corporation
31440 Northwestern Highway
Farmington Hills, MI 48334-2564
Phone: 248. 737.7300
800.521.9353
http://www.compuware.com

HyPerformix, Inc.
4301 Westbank Drive
Building A, Suite 300
Austin, TX 78746-6564
Toll Free: 800.759.6333
http://www.hyperformix.com

OPNET Technologies
3400 International Drive, NW
Washington, DC 20008
(202) 364-4700
http://www.opnet.com/

Pilz Automation Safety L.P. (U.S. Headquarters)
24850 Drake Rd.
Farmington Hills, MI 48335
Tel. (248) 473-1133
info@pilzusa.com

SHIP STAR Associates
36 Woodhill Drive
Suite 19
Newark, DE 19711-7017
(302) 738-7782

Smar Equipamentos Industriais Ltda.
Av. Dr. Antonio Furlan Jr., 1028
Sertãozinho, SP
Brazil 14160-000
Phone: +55 16 645-6455
di@smar.com.br

Synergetic Micro Systems, Inc.
2506 Wisconsin Avenue
Downers Grove, Illinois 60515
Phone:+1 800.600.0598
www.synergetic.com

Wolverine Software Corporation
2111 Eisenhower Avenue, Suite 404
Alexandria, VA 22314-4679
(800) 456-5671
(703) 535-6760
Fax: (703) 535-6763
mail@wolverinesoftware.com

Appendix A: Sources of Additional Information on Fieldbus Systems

Publications

Synergetic Micro Systems

Synergetic Micro Systems maintains fieldbus comparison charts that are widely referenced on the web [http://www.synergetic.com/compare.htm]. These charts include summaries of physical characteristics, transport mechanism, and performance.

Control Engineering Magazine

Control Engineering has published several articles about fieldbus systems. A brief summary of the following fieldbus systems appears in the January 1998 issue [2]:

- ARCNet,
- AS-Interface (AS-i),
- ControlNet,
- DeviceNet,
- FOUNDATION Fieldbus,
- LonWorks,
- Interbus,
- Profibus,
- Seriplex,
- SERCOS,
- Smart Distributed System (SDS), and
- WorldFIP.

This article is available on the web at
http://www.controleng.com/archives/1998/ctl0101.98/01c100.htm.

Plant Automation.com

Plant Automation.com is a web-hosted "community for industrial professionals." It has a variety of articles, news, and product information.

http://www.plantautomation.com

User Groups and Vendors

ARCNet

ARCNET ® Trade Association
8196 S. Cass Avenue, Darien, Illinois 60561
Phone: +1-630-964-4280
Fax: +1-630-724-0211
http://www.arcnet.com/

AS-Interface

S-International Association
Zum Taubengarten 52
63571 Gelnhausen (D)
Tel. +49-6051-473212
Email as-interface@t-online.de

AS-i Trade Organization.
16101 N. 82nd Street, Suite 3B
Scottsdale, AZ 85260
Ph: (602) 368-9091
www.as-interface.com

International web site
http://www.as-interface.com

BITBUS

BEUG e.V.
Theaterplatz 9
D-52062 Aachen
FAX +49-241-48480

BITBUS European User's Group
http://www.bitbus.org/

CAN

CANopen http://www.canopen.com/
CAN in Automation (CiA) www.can-cia.de

ControlNet

ControlNet™ International
William H. (Bill) Moss, Executive Director
PMB 315 - 20423 State Road 7 #F6
Boca Raton, FL 33498-6797 USA
(1) 561 477-7966 Phone

ControlNet International
8222 Wiles Rd., Suite 287
Coral Springs, FL 33067
954-340-5412
http://www.controlnet.org

DeviceNet

Open DeviceNet Vendor Association, Inc.
8222 Wiles Rd., Suite 287
Coral Springs, FL 33067
954-340-5412
http://www.odva.org

Ethernet

http://www.gigabit-ethernet.org/
http://www.iaopennetworking.com/
http://www.industrialethernet.com/

Foundation Fieldbus (FF)

Fieldbus Foundation
9390 Research Blvd., Suite II-250
Austin, TX 78759-9780
512-794-8890
http://www.fieldbus.org/information/

Interbus-S

INTERBUS Club - USA
Mailing Address:
P.O. Box 25141
Philadelphia, PA 19147
Phone: (888) 281-2871
http://www.ibsclub.com/

LonWorks

Echelon Corporation
415 Oakmead Parkway
Sunnyvale, CA 94085
1-888-ECHELON (324-3566)
lonworks@echelon.com
http://lonmark.org

MIL-STD-1553

http://www.sbs.com/av_1553over.shtml

ModBus

http://modicon.com/openmbus

P-Net

http://www.infoside.de/infida/index2uk.htm

Profibus (US User's Group)

Michael Bryant
PROFIBUS Trade Organization
16101 N. 82nd Street
Suite 3B
Scottsdale, AZ 85260
Tel: ++ 1 480 483 2456
http://www.profibus.com/

SafetyBUS

http://www.safetybus.com.

Seriplex

Seriplex Technology Organization, Inc.
P.O. Box 27445
Raleigh, NC 27611
800-775-9462
http://www.seriplex.org

SwiftNet

Robert S. (Bob) Crowder
SHIP STAR Associates
36 Woodhill Drive
Suite 19
Newark, DE 19711-7017
(302) 738-7782

http://www.shipstar.com/swiftnet.html

WorldFIP

http://www.worldfip.org/

23/25 Avenue Morane Saulnier,
92364 MEUDON-La-Foret
CEDEX, FRANCE

Appendix B: Selected Fieldbus Applications and Studies

Kola and Novovoronezh Nuclear Power Plants

Systems of two Russian nuclear power plants use Echelon's LonWorks networks and have received safety agency certification. The Gardia-2 Safety System, by DICS Intertrade of Sophia, Bulgaria, uses a triply redundant LonWorks network in its control and diagnostics of safety valves. The systems have been in use since early 1999 at Units 1 and 2 of the Kola Nuclear Power Plant and since June 1999 at Unit 4 of the Novovoronezh Nuclear Power Plant.

The systems were certified by Groupe SEBIM of France to meet Class E1/K3 (French RCCE code) and ANSI/IEEE 323 and 344 safety standards for nuclear electric power generating plants. The system uses special voting algorithms, redundant power supplies, and extensive diagnostic routines to ensure that the system remains on-line at all times.

Electric Utility Application

Hewlett Packard built a data acquisition hardware and software platform called Vantera that can provide electric utilities real time information about energy usage. It could incorporate information from smart sensors tied into the network, following several industry standards including IEEE-P1451 and OLE for process Control (OPC). ABB Power T&D Company An HP Vantera node is connected to the corporate Ethernet intranet and intelligent sensors, and places measurement values in its real-time data manager. In a related effort, Electric Power Research Institute (EPRI, Palo Alto, Calif.) investigated network communication options, conducted tests to verify findings, and defined a standardized communication protocol for electric utilities sharing information: supply, demand, and cost information in collected in 4 milli-seconds or less. The 4 milli-second interval was chosen so that functions such as tripping could occur over the LAN. EPRI tested 12 million bits per second (Mbps) Profibus DP against 10 Mbps Ethernet with switched hubs and 100 Mbps Fast Ethernet with hubs. Both Ethernet solutions bettered the 4-msec requirement. ["HP Vantera Helps Companies with Deregulation," Control Engineering, May 1999]. Reportedly, Profibus spent too much time passing the token to meet the 4 msec message delivery requirements. [“Ethernet meets requirements of deregulated electric industries,” Control Engineering, April 1999].

Network broadcast storms were avoided with a publish/subscribe messaging model developed by Tibco. A publisher node places a message on the network where subscriber nodes can read it then react or reply. The publish/subscribe messaging architecture eliminates the need for nodes to know where other nodes are located, and allows nodes to communicate peer to peer. (This general style of messaging, also known as producer/consumer, is used on several fieldbus systems.)

Electrical interference effects on Ethernet cables were also tested and found to cause failures in packet transmission or packet corruption in some situations. Fiber-optic media

are therefore recommended for critical, high-speed applications. ["HP Vantera Helps Companies with Deregulation," Control Engineering, May 1999.]

The Electric Power Research Institute (EPRI) has since created the Utility Communication Architecture (UCA) version 2.0 [3]. UCA specifies the communication architecture for the entire enterprise, based on open standards. On the field device level Manufacturing Message Specification (MMS) is used with object models developed for common devices.

SHIP STAR Associates

Ship Star Associates has published an evaluation of several fieldbus systems on their web site [12]. The evaluation was performed in 1996. The systems selected for evaluation were:

- Foundation Fieldbus, H1, 32.25 Kbs,
- Foundation Fieldbus, H2 (not Ethernet), 1 and 2.5 Mbs,
- PROFIBUS-DP, 12 Mbs and 1.5 Mbs,
- DeviceNet, 500 Kbs and 250 Kbs,
- WorldFIP, 2.5 Mbs and 1 Mbs,
- SwiftNet, 4 Mbps

These were selected by the author and Boeing as the best candidates for use in the flight control systems of commercial aircraft. However, SwiftNet was developed by Ship Star after phase 1 testing of other systems showed they did not meet the requirements. (SwiftNet is included as Type 6 in the IEC 61158 fieldbus standard.)

The application requirements were synchronous scanning of 12,800 samples per second with sample timing error less than 1.3% of the scan period. In addition to the cyclic scanning, alarms/messages were generated in the time scheduled between cycles by these systems.

SwiftNet met the requirements due to its high-efficiency bus protocol and strict control of timing jitter and cycle scheduling errors. Other interesting features are:

1. common bus time is available to all devices,
2. alarms and device errors are time stamped, and
3. communications are implemented in one ASIC.

Among the other systems, Profibus at 12 Mbps achieved the best sample rates by a wide margin. All of these systems had problems with timing: errors were on the order of 100% of the scan period when both cycle schedule errors and bus timing jitter are added together. However, no bus except SwiftNet was found to meet the requirements.

Deten Chemicals

Deten Chemicals S.A. produces a raw material used in detergents at its plant in Camacari, Bahia, Brazil. A Foundation Fieldbus system was installed by Smar to handle some

controls. Both redundant communications and redundant bus power systems were used for critical controls. This included fieldbus communications between PLCs and a Motor Control Center (MCC) that has its own PLC.

Triply redundant instrumentation was installed on critical control loops. This included three temperature transmitters that operate using a 2x3 voting scheme. Each transmitter was connected to a separate fieldbus process interface board. Redundant fieldbus instruments were used on all other control loops. Each loop had two temperature transmitters and one fieldbus-to-current (4-20 mA) converter. If the principle temperature transmitter stopped working, the secondary temperature transmitter would send its data to the PID control block. If one of the fieldbus interface boards developed a problem, the data will be handled by another. If both transmitters and both interface boards stopped working, a safety value would be sent by the software block to the PID control block.

Corning

Corning's Concord NC plant has over 600 analog devices and over 2000 discrete points running over a Foundation Fieldbus system. It is a fully redundant design with the aim of achieving high reliability and availability. Each bus segment is powered from both ends of the bus. On each end of the bus segments is a bus interface card attached to redundant I/O servers. The I/O servers have redundant Ethernet cards and send their data to redundant data servers. There are also redundant Ethernet fiber switches. The system includes 16 operator consoles.

INDEX