SDS###: A CAN Protocol for Plant Floor Control

El.-Ing. HTL, A. Beck
Honeywell
Webergutstrasse 4, CH-3052 Zollikofen, Schweiz

ABSTRACT

A Controller Area Network (CAN) based communication protocol and control environment are presented for plant floor automation and control. This robust protocol allows for messaging between sensors, actuators, man-machine interfaces, controllers, and other plant floor control devices. This architecture provides a communications environment upon which a high-speed, real time centralized or distributed control platform can be created. In addition, the communication system provides the enabling technology for intelligent sensors, integration of advanced device diagnostics, and plant floor data-gathering, for communication to higher level networks. State of the art technology allows for the integration of the network interface into the smallest of plant floor devices.

This system provides benefits in a variety of areas. Several real applications are discussed, including a description of an installation at a General Motors Engine Plant in Australia. A description of the benefits achieved in each installation is provided. In particular, machine uptime is improved through the integration of more intelligent devices such as programmable, re-configurable sensors with integrated diagnostics. Labor costs are reduced, due to the simplified interconnect system, and the easy addition to and modification of existing installations. Also, reliability is improved through diagnostics and reduced system complexity, and maintenance is improved through advanced tools for device and system configuration, monitoring, and troubleshooting. Reduced wiring costs can also be achieved, due to the bussing of devices.

INTRODUCTION

Civilized societies, in an attempt to improve the human condition, are forever searching for ways to improve their control over their surroundings and creations. Technology advances are driven, in part, by this thirst for ever improving control, and the technological evolution has migrated from centralized control to distributed control. As mechanized transport evolved, the train and locomotive became insufficient for transportation needs; the automobile empowered individuals in a way that the covered wagon never could. As information technology progressed, the mainframe computer gave way to the desktop computer, networks of desktop computers, hand-held computers, and eventually, networks of hand-held computers.

Honeywell is a leader in this quest for control. In space and aviation, homes and buildings, and industry, Honeywell delivers control solutions to its customers. Distributed systems are an essential part of the solutions that Honeywell brings to its customers to meet their control needs. Distributed systems are a logical and evolutionary step in the development of new control strategies.

HONEYWELL/SDS###

An example of a distributed system is Honeywell's SDS### Smart Distributed System. This product is the focus of our development efforts at Honeywell to stimulate a new, open system approach to industrial control. The following is an in-depth look at the concepts behind SDS###.

A HIGH SPEED FIELDBUS

What are the characteristics of a device communications network for factory floor control? In the discrete world, speed is of the essence. The communications system must be tailored to provide high speed information exchange for real time exchange of process variables. PLC scan times of 10 milliseconds or below are increasingly common in the modern discrete control world. In order for the communication system to have little or no impact on the user's perception of control, the communication delay for process variable exchange should be less than 1 millisecond. In addition, a flexible, robust communication protocol is needed to support the requirements of intelligent, distributed field devices. Power delivery along the bus is essential for two reasons. All devices attached can make use of the delivered power to drive one side of an electrical isolation barrier. Furthermore, many types of devices (e.g. sensors, push buttons, etc.) have low power requirements, and their installation is greatly simplified by central power delivery. For devices running solely off of the bus-supplied power, the isolation barrier may not be needed. Of course, the system must be reliable and operational under conditions of industrial noise. Systems for use in the European market will have to meet the standards for electromagnetic compliance (EMC). Also, the added electronic complexity due to the communications interface should be as small as possible, to fit the widest range of devices, but scalable to meet the needs of complex devices. For bus-powered devices, power is limited, so the electronics should also have low power consumption. The new connection requirements present opportunities as well as challenges. There is no single interconnect solution which meets the needs of all customers, so a variety of choices are needed. These requirements taken together provide a useful method to select an appropriate technology for communications. There is a wide spectrum of communication technology available, from simple RS-232 serial communications up to Ethernet, FDDI, wireless, and beyond. CAN (Controller Area Network) is a relatively new technology which addresses these needs.

PHYSICAL LAYER

Referring to the OSI seven layer model for communications stacks, CAN is mostly a layer 2 definition (Data Link Layer). Other standards processes are moving to fill the gaps, particularly ISO/DIS 11898, a working discussion committee establishing Physical Layer standards for CAN communications. Honeywell has chosen to use these standards in our SDS### Smart Distributed System as the basis for our Physical Layer. We have developed a five-wire cabling strategy, including a twisted pair for differential data communications, a twisted pair for power delivery along the bus, and a drain wire attached to the cable shield(s). Up to 64 devices can be attached to a single bus in this fashion. In conjunction with cable suppliers, we have developed a set of cabling products for SDS### which allow the cabling and connection of the bus with no tools. These cables use industry standard four and five pin connectors, plus a specially developed cable "tee", to allow the installer to connect the main bus trunk and drops to devices using quick-disconnect, sealed connectors. (fig. 1)

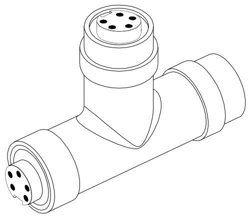

Figure 1 Quick Disconnect Cable Tee Sketch

For applications where a hard wired solution is preferred, we have also developed a set of pigtails, and reference notes, using traditional approaches with cut cable, terminal strips, etc. These two approaches span customer needs ranging from sealed connections, to conduit use, to ease of wiring and installation. Quick-disconnect sealed cable use by customers so far indicates the potential for an 80 percent reduction in installation time. Coupled with the reduction in cable length requirements afforded by the network communications approach, these translate into the potential for substantial benefits to the customer, in the areas of cost, installation time, maintenance, and system modification.

SDS### APPLICATION LAYER

The OSI communication stack includes seven layers to account for all communication functions needed to be able to generically describe any network communication system. Optimization choices inherent in CAN and necessary for a realizable high speed control-oriented network allow for the collapsing of all layers above layer 2 into a single layer called the Application Layer (layer 7 of the OSI model). (fig. 2)

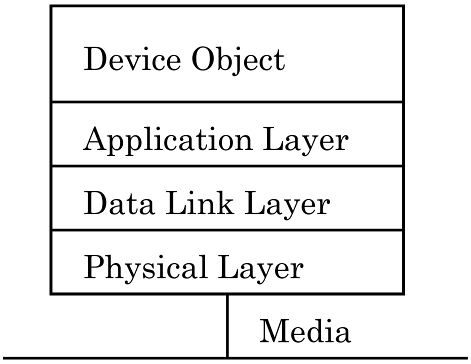

Figure 2 Device Model

This layer defines the services available to the user of the communication protocol, and manages the translation of user service requests into message objects suitable for the Data Link Layer. It also handles service requests coming up through the Data Link Layer from other devices. The SDS### Application Layer Protocol defines four generic service classes available to the user: READ, WRITE, ACTION, and EVENT. READ and WRITE operate on defined object **attributes** of a particular device. ACTION indicates that the device is to perform some specified function. EVENT provides for notification of asynchronous or other spontaneously generated events in the device, or sensed by the device. Each service can be used with or without an explicit Application Layer acknowledge of the service. For example, in a peer-to-peer environment, sensors may transmit their process variable data in broadcast fashion, without an explicit acknowledgment message returned to indicate reception. Note that the Data Link Layer still guarantees error-free transmission of the message under these circumstances. In a master/slave environment, the user of an Application Layer service may receive an acknowledgment message in response to the service request transmitted. This provides an additional layer of confirmation on top of the Data Link Layer. For services which require data to be returned as a result of the service request, the acknowledgment message provides the return data. Using these services, it is possible to perform a wide variety of functions in the control environment, ranging from reporting of process variables, to modifying controlled outputs, to accessing and changing device configuration. The combination of the Application Layer and CAN provide for a full-featured network suitable for a wide variety of control applications on the factory floor.

SDS### MESSAGE TYPES

The basic SDS### long form Application Protocol Data Unit (APDU) is diagrammed below:

	BITS								
	7	6	5	4	3	2	1	0	
	DIR {0,1}	Device Address {0 ... 125}							**CAN**
	PDU Type {0 ... 7}			0	Data Length Code > 0				**HEADER**
	Request/Response/Error		Fragment = 0	Reserved	Object/Group ID {0 ... 15}				
B	PDU Modifier (Attribute, Action, Event, {0 ... 255})								
Y	Variable Data (Byte 0)								**CAN**
T	Variable Data (Byte 1)								**DATA**
E	Variable Data (Byte 2)								**FIELD**
S	Variable Data (Byte 3)								
	Variable Data (Byte 4)								
	Variable Data (Byte 5)								

Figure 3 SDS™ APDU (long form) in a CAN Frame

The usage of all 10 bytes of the CAN message format are defined. The identifier field contains 3 parts. The Device Address occupies 7 bits, allowing up to 126 logical devices to be present on a single network. There is a direction bit which determines whether the Device Address is intended to be interpreted as either a source or destination address. Finally there is a PDU Type field, which identifies whether the message is a READ, WRITE, ACTION, or EVENT application layer service. For a long form message, the message always contains at least two defined bytes in the CAN data field, so the Data Length Code is always 2 or larger. The first byte of the CAN data field contains additional SDS### protocol related information, including whether the message is a service request, a service response, or a service error response, whether the message is a fragment part of a larger total message, and which of 16 possible objects within a logical device is targeted. The next byte in the CAN data field contains the PDU modifier, which is the number of the **Attribute** being read or written to, or the number of the ACTION or EVENT service involved. The remaining bytes (up to 6) are used for transferring data, if any.

Two additional messaging formats are defined. An extension of the long form APDU is the fragmented long form APDU:

	BITS								
	7	6	5	4	3	2	1	0	
	DIR {0,1}	Device Address {0 ... 125}							**CAN**
	PDU Type {0 ... 7}			0	Data Length Code > 0				**HEADER**
	Request/Response/Error		Fragment = 1	Reserved	Object/Group ID {0 ... 15}				
	PDU Modifier (Attribute, Action, Event, {0 ... 255})								
B	Fragment Number {0 ... 63}								

Y	Total Fragment bytes {0 ... 255}	CAN
T	Variable Data (Byte 0)	DATA
E	Variable Data (Byte 1)	FIELD
S	Variable Data(Byte 2)	
	Variable Data (Byte 3)	

Figure 4 SDS™ APDU (fragmented long form) in a CAN Frame

In the fragmented long form, the Fragment bit is set to one, indicating that this message is part of a larger, fragmented message sequence. This form is used automatically when the amount of data to be transferred is greater than the amount available (6 bytes) in a single long form APDU. The final message type is the short form APDU:

BITS								
7	6	5	4	3	2	1	0	
DIR {0,1}	Device Address {0 ... 125}							CAN
PDU Type {0 ... 7}			RTR	Data Length Code = 0				HEADER

Figure 5 SDS™ APDU (short form) in a CAN Frame

The Data Length Code is always zero in a short form APDU. Thus the message contains only two bytes. In this case, the PDU type is redefined to be specific to the device object model. Thus, different device types can use the same 8 PDU types to mean different things. In the case of a simple, single point digital input or output, the short form APDU is used to quickly transmit the information when the device turns on or off. For analog inputs or outputs, this message type can be used to quickly transmit the information when predefined set points are exceeded. A motor drive might use these messages to communicate that it is at the commanded speed. Other uses can be specific to the device in question.

DEVICE MODELS

The SDS### Application Layer Protocol is intended to be used in an object-oriented methodology. Devices are classified using an objected-oriented approach, including a device hierarchy with inheritance. All SDS###-compatible devices inherit (eventually) from a common device object which includes the minimum functions necessary for a device to communicate and be recognizable in an SDS### environment. The power of this approach becomes apparent in multi-vendor systems. In an open environment, two competing vendors may make different photoelectric sensors, which may have some features in common (such as the ability to indicate on/off) and some differentiating feature (e.g. diagnostics, etc.) These two devices can inherit from the same device object which specifies the basic sensor on/off functionality. Each device becomes a different type of object, but the common inheritance allows a controller to query the device to ascertain its inheritance and functionality. Thus the two devices can both appear to the controller as photoelectric sensors. If the controller also has specific knowledge of the advanced features of a device, it can use these also. Or, the advanced features may be appropriate only for access via a hand-held terminal or some other configuration management tool. Thus, the controller can utilize the process variable resident in the device, without having to know all its capabilities. A sample device model definition for a motor drive device is outlined below:

ATTRIBUTES			
ATTRIBUTE ID	ITEM	VARIABLE TYPE, SIZE, R/W	DATA RANGE
0	Reserved		
1	Reserved		
2	Device Type	String, 1-5 bytes, binary	(0...255)
3	Vendor ID Number	Unsigned, 8, R	(0...255)
4	Device Address	Unsigned, 8	(0...125)
5	Unused		
6	Un/Solicited Mode	Bit, R/W	0,1
7	Software Version	String, 32, R	ASCII
8	Diagnostic Error Counter	Unsigned, 8, R	(0...8)
9	Bus Diagnostic Flags	Unsigned, 8, R	(0...8)
10	Cyclical Timer	Unsigned, 16, R/W	(0...40963)
11	Serial Number	Unsigned, 32, R	(1...4,294,967,295)
12	Date Code	String, 4, R	ASCII
13	Catalog Listing	String, 32, R	ASCII
14	Vendor Name	String, 32, R	ASCII
15	Device Name	String, 32, R	ASCII
16	Reserved		
17	Reserved		
51 - 59	An-01 thru An-09	Unsigned, 16, R/W	(0...65535)
60 - 70	Available for future use		
71 - 82	bn-01 thru bn-12	Unsigned, 16, R/W	(0...65535)
83 - 90	Available for future use		
91 - 118	Sn-01 thru Sn-28	Unsigned, 16, R/W	(0...65535)
119 - 130	Available for future use		
131 - 172	Cn-01 thru Cn-42	Unsigned, 16, R/W	(0...65535)
173 - 255	Available for future use		

Figure 6 SDS### Drive Interface Device -- Attributes

ACTIONS					
ACTION #	ITEM	INPUT PARAMETERS	DATA RANGE	OUTPUT PARAMETERS	DATA RANGE
0	NOOP	None		None	
1	Change Address	Serial Number, Address	Unsigned 32, Unsigned 8	None	
2	Self Test	None		None	
3 - 5	Unused				
6	Clear All Errors	None		None	
7 - 56	Unused				
57	Password	Pass Number	Unsigned, 16, R	None	
58-60	Unused				
61	Operation Command	Bit Map	Bit 0,1 Unsigned 8	None	
62	Frequency Reference	Requested Value	Unsigned, 16, R/W		

Figure 7 SDS### Drive Interface Device -- Actions

EVENTS			
EVENT #	ITEM	OUTPUT PARAMETERS	DATA RANGE
0	Self Test Failure	Diagnostic Error Count ### 0	Unsigned, 8
1 - 50	Unused		
51	Multifunction Contact Output	Bit mapped contact status	Unsigned, 16
52	Multifunction Analog Output	Drive speed indication	Unsigned, 16
53 - 100	For Future Use		
101- 150	Fault Status	Fault Code	Unsigned, 16

Figure 8 SDS### Drive Interface Device -- Events

SMART DEVICES

The potential advantages in a communication environment to the device manufacturer are enormous. Honeywell is a leader in control solutions and already uses communications in their control strategies. By placing a microprocessor in our SDS###-compatible sensors to manage the communications protocol, we benefit from being able to exploit the microprocessor to manage other device features. For example, our SDS###-compatible sensors can all be configured to be normally on or normally off, directly from the communications interface, using a hand-held terminal, or other tools. We have photoelectric sensors which can indicate when the head is operating in a low-gain situation, perhaps due to a dusty lens, or out-of-alignment reflector. We have limit switches which can predict the end of their useful life, as a recommendation for preventive maintenance. These features can predict trouble spots for users before they develop into shutdown situations. In this way, machine or line uptime can be improved.

BENEFITS

The range of benefits that this system provides is huge. Hand-held terminals may be plugged into the system at any point to query the status of devices, force devices to test themselves, force machine outputs on or off, etc. The cabling system provides for a reduction in overall cabling needed, as well as vastly simplified installation. Smart devices can keep track of their own integrity, allow for configuration and re-configuration, and maintain special features that allow for differentiation, variety, and solutions in the marketplace. As users demand more and more open systems, communication strategies will have to be supported by a variety of vendors in order to be successful. As communications become a part of the designs, a variety of other benefits and life cycle cost reductions are achieved, including reduction in design time, and reduced maintenance costs.

CASE STUDY: GM AUSTRALIA (HOLDEN), ENGINE LINE

Holden has designed a new section within the motor assembly line to allow for the new motors used in the latest model Holden Executive. The line is split such that as motors are moving down the assembly line, this section installs the top half of the motor, including the head. They are then given a bar-code number, and checked for completeness. The line checks if the motors are for existing or new model vehicles. The conveyor system then decides which direction the motor should be sent depending on the motor model. The system then switches the correct valves to divert the new model motors to a separate area of the engine test bay for a modified test procedure. The existing model motors go straight through to the existing engine test bay area.

The SDS### installation consists of:

1. Two ITS-Sensor Modules
2. One ITS-Actuator Module
3. 17 SDS### Limit Switches
4. 2 Univer Valves
5. 1 Handheld Activator (Programmer)
6. Hardwired cable assembly with Brad Harrison### pigtails.
7. Total trunk length 180 meters

HOLDEN ENGINE COMPANY

Overhead Conveyor

Assembly Line B To Engine Test Bay

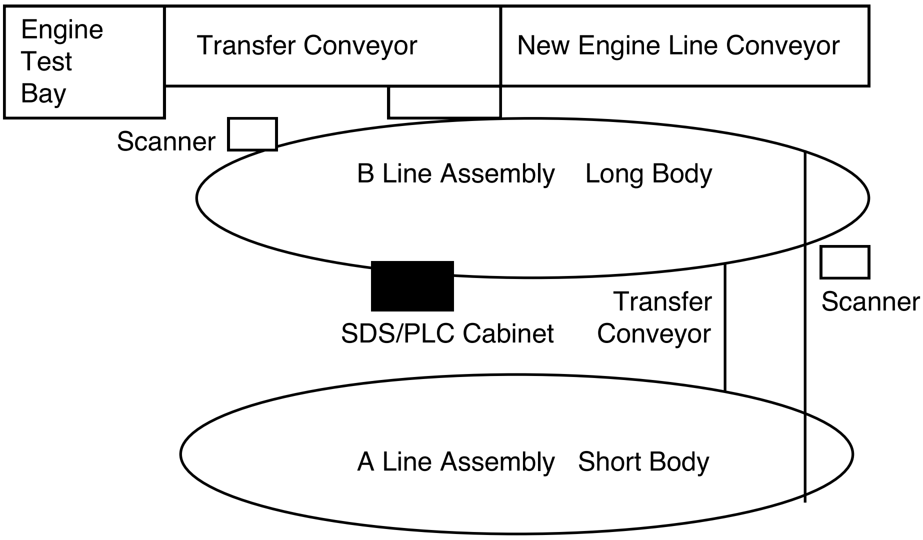

Figure 9 Holden Engine Line

HIGHLIGHTS

As a result of the problems in fault diagnosing with the system, and the limit switches and valves not being accessible at ground level, Holden decided to look at a system which would have improved diagnostic capabilities which would be accessible from a single point. The SDS### installation provided this. This saved them from having to climb up into difficult and dangerous areas to inspect switches if a problem existed. The Handheld Activator can be used for this purpose. The diagnostics on the ITS units also can be used to quickly diagnose faults. Additionally, the customer elected to use an additional diagnostics module on the PLC. Using the diagnostic capabilities of the SDS### installation, it could immediately report which device was the problem. This is an easy task with the SDS### installation, and the ability to ask the SDS### system for which device was the problem was a distinct advantage.

CUSTOMER BENEFITS/COMMENTS

This customer was particularly interested in the diagnostic capabilities of SDS### systems, and the ability to access this information easily from a remote location using a device such as a Hand Held Terminal. Holden is pleased with the application, so much so that they are considering expanding the project to include an extra 30 SDS### devices. The installation went smoothly, so much so that even though the contractors were behind schedule in installing the control system and field devices, they actually finished ahead of schedule due to the simplicity of the SDS### system installation.

CASE STUDY: CROWN CORK AND SEAL, USA, CANNING LINE

Crown Cork and Seal is setting up a new facility in Chicago, Illinois, to make 3 piece steel aerosol cans. SDS### is slated for use in several application areas. The first application is on a portion of the process which puts the dome cap on the can, and sends the completed cans to the palletizing equipment. There are several of these machines, each of which handles 400 cans per minute. The SDS### devices are used in a variety of applications, including can counting, jam detection, and line sequencing.

The SDS### installation consists of:

1. Two Koyo DL405### PLC Direct### SDS### Interfaces
2. Three ITS-Remote I/O Modules
3. 30 SDS### Mini-base Photos and Prox Switches
4. 25 SDS### Remote I/O Industrial I/O Subbases for pushbutton stations and miscellaneous I/O
5. 5 SDS### Limit Switches
6. 1 Handheld Activator (Programmer)
6. Hard wired cable assembly with Molex### pigtails and Molex### quick connect cordsets.
7. Total trunk length 150 meters (each of two busses)

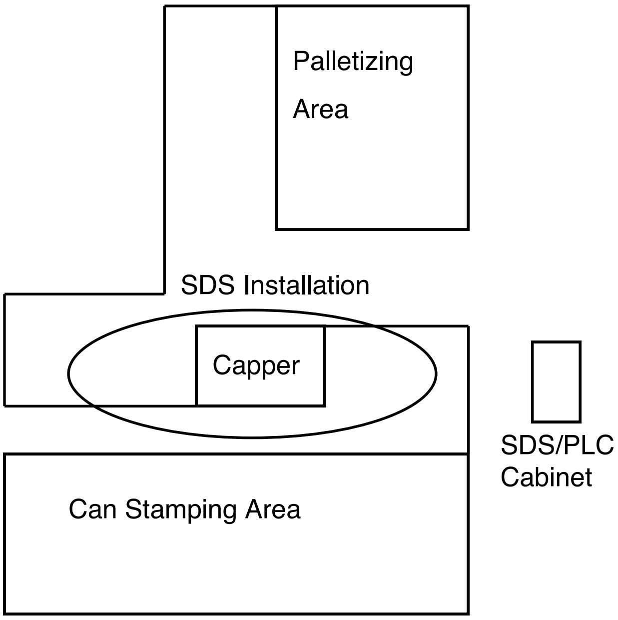

Figure 10 Crown Cork and Seal Canning Line

HIGHLIGHTS

The customer is primarily interested in ease of installation and system configuration. The SDS### system drastically simplified and reduced the total machine wiring solution, and provided direct access to a variety of device configuration parameters, such as normally open/normally closed or output inversion, on delays, off delays, and high speed counting. All of these features are accessible from a single point using the Handheld Activator. Furthermore, the customer is using a direct PLC interface for SDS### to the PLC Direct### (by Koyo) line. These direct interfaces allow for much higher density I/O, reducing costs and cabinet space, and drastically simplifying control cabinet wiring. The customer also used a mixture of hardwired, conduit wiring, and sealed quick-disconnect wiring, depending on the area of the machine.

CUSTOMER BENEFITS/COMMENTS

The main benefits to the customer were simplified installation and reduced overall costs, while providing flexible and easy system configuration and maintenance. The customer is pleased with the installation, and is evaluating the remaining application areas for use of SDS### systems. Approximately 10 additional lines are expected to use SDS### systems this year. In particular, the customer was pleased with the ability to quickly and easily add devices to the SDS### system anywhere along the line just by tapping into the bus, without pulling any new cables. Very often, the best way to solve a particular application problem is not apparent until the line is up and running. With the SDS### system, the application engineer was able to add devices to the beginning and end of each piece of equipment to control can jamming better and speed up the overall throughput of the line.

THE FUTURE

What does the future hold? Clearly, the way industrial control systems are created, used, and maintained will undergo significant change as communications proliferate to all levels of control environments. As industrial computers have a larger impact on industrial systems, the marriage of industrial computers and networked, distributed devices is the next milestone on the path to distributed systems.