Implementing a distributed control system and programmable controller interfaces in paper mill automation

P. Y. Keskar and Tim Marxen

Data and information exchange must take place at an early stage between the distributed control system and programmable controller groups, placing particular emphasis on the planning and design of the programmable controller subsystem.

While considering distributed control systems (DCS) for pulp and paper mill applications, it is a generally accepted practice to implement all analog control and monitoring loops as part of the DCS system, while the sequencing and motor control functions are performed in a separate programmable controller (PLC) subsystem which will be integrated into an overall DCS system. Supervisory control of the PLC subsystem is performed via the DCS consoles.

In the majority of applications, DCS and PLC hardware are furnished by different vendors. Also, the application programming and configuration of these systems are done by various parties, which may include one or more of the following: owner, DCS vendor, consultant firm, or PLC vendor. Since different hardware and programming groups are involved, careful planning and design is required to implement the DCS-PLC interface efficiently to minimize wasted man-hours during checkout and startup. Exchange of crucial information and data must take place at an early stage between the DCS and PLC groups, and each group must develop methodology for their application programming that keeps in mind the needs of the "other" group. We place heavy emphasis on the planning and design methodology of the PLC subsystem because it is the system that we believe needs the most attention.

Introduction

Figure 1 shows a simplified block diagram of a typical DCS system for paper mill applications. The analog inputs and outputs are handled directly by the DCS system, while the digital inputs and outputs, as well as sequencing, are implemented in the programmable controller subsystem. However, in most applications the supervisory control of the PLC subsystem is accomplished through the DCS consoles.

In other words, starting and stopping of motors and opening and closing of valves is done from the DCS consoles. Also, all status information on equipment controlled by the PLC subsystem, as well as the PLC diagnostic data, needs to be brought into the DCS system for use in the graphic displays, preparation of reports, and alarm summaries. The DCS console is usually the only operator interface for the PLC subsystem.

For these reasons the DCS-PLC interface is a very important part of any millwide DCS system. The efficient design and implementation of this interface is crucial to the on-time implementation and success of any DCS project. Careful planning and design of the DCS and the PLC subsystems becomes even more important in paper mill applications where the DCS and PLC hardware is supplied by different vendors and where the configuration and application programming of each subsystem may be implemented by different parties. This paper addresses the planning and design considerations that can lead to efficient implementation of the DCS-PLC interfaces.

The PLC subsystem

In this section several issues will be discussed that are important for efficient implementation of the DCS-PLC interface.

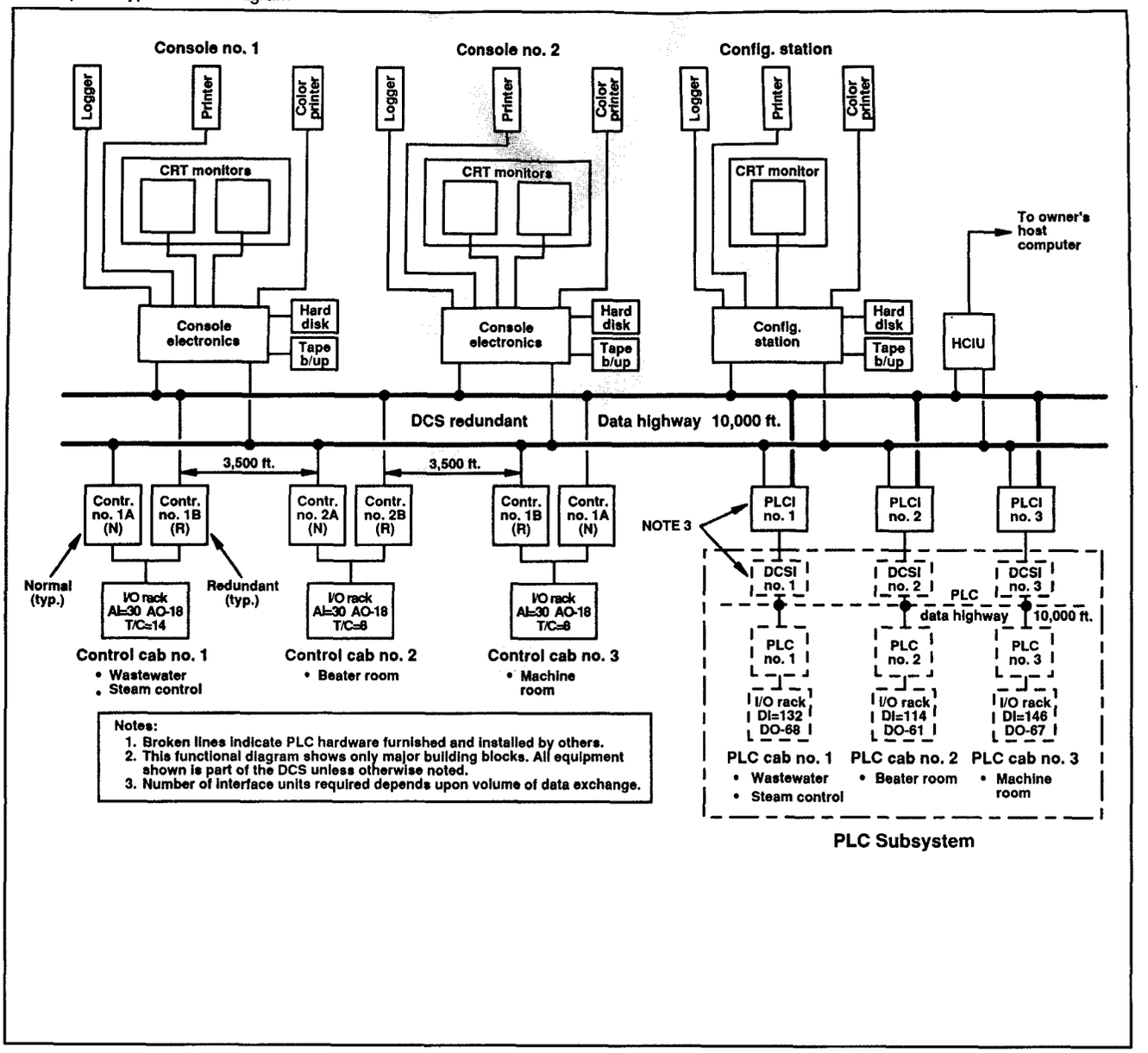
Coils and word assignments—the structured approach[1]

One of the major questions in implementing DCS-PLC interfaces concerns how much data needs to be exchanged between the two subsystems. Common sense and good engineering judgment should be used in optimizing the volume of data exchange. A PLC "coil," or bit, is the smallest unit of the data exchange. However, for convenience and speed, 16 coils (bits) are grouped together to form one digital word. The volume of data can then be measured

Keskar is head of the E & I Dept. at Sandwell, Inc., 9424 Baymeadows Rd., No. 100, Jacksonville, Fla. 32256. Marxen is senior project engineer at Georgia-Pacific, West 9th St., Brunswick, Ga. 31521.

[1]Keskar, P. Y., ISA Transactions 29(2): 47(1990).

1. Simplified typical block diagram



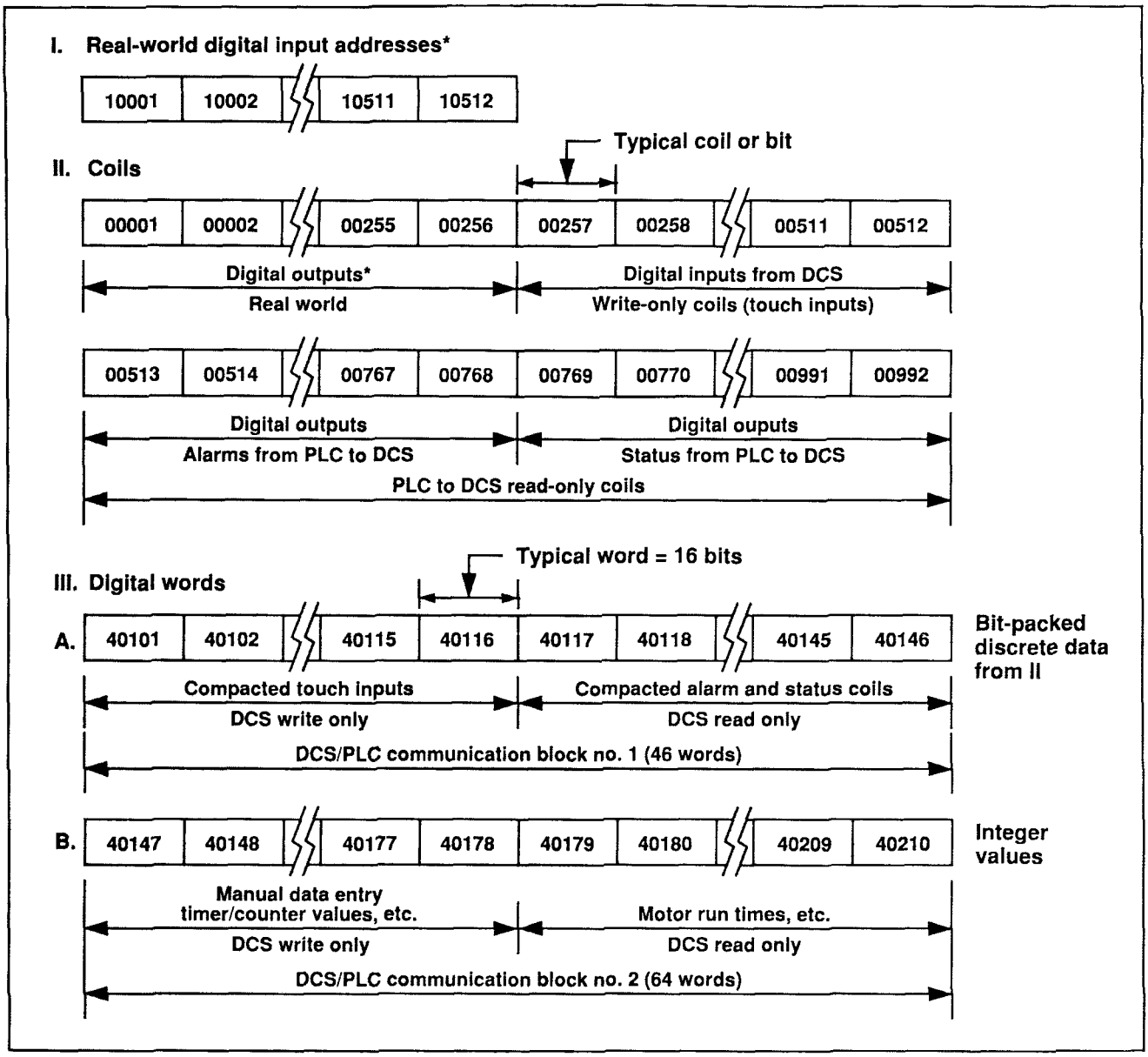
in terms of the number of "16-bit words" involved in the information exchange between the DCS and the PLC. In this paper, the term "bit packing" refers to the PLC instruction that allows the transfer of the status of a group of 16 coils into a 16-bit word. Besides storing the status of the coils, the digital words also store numerical values. The numerical values may be setpoints from the DCS transferred to the PLC or values computed by the PLC to be transferred to the DCS.

Coils (bits) and words that are involved in the data exchange can be classified into two broad categories: read only and write only. The read-only coils and words can be read by the DCS, but the DCS cannot modify or write into these coils or words. Status and alarm coils, words containing motor run times, and similar integer values are examples of read-only coils and words. The write-only coils and words are those in which the DCS can write new data. Coils for motor start and stop commands or words that store manually entered data (via the console) are examples of write-only coils and words.

Following is a suggested outline of input and output (I/O) coils and words classification and assignment (Note: "*" indicates items not involved in data exchange with the DCS.):

1. Real-world digital input addresses (hard-wired contact inputs)*
2. Coils
   Digital real-world outputs (hard-wired contact outputs)*
   Digital inputs from the DCS (write-only coils)
   Digital outputs to DCS (read-only coils)
   Alarms
   Status (run, open, close, etc.)
   Internal coils*
3. Digital words
   Inputs received from the DCS (write-only words)

2. PLC I/O coils and word assignments (* indicates items not included in the data exchange)



Timer values and similar manual data entries into the PLC system

Bit-packed write-only coils (i.e., motor start commands)

Outputs sent to the DCS (read-only words)

Bit-packed read-only coils

Motor run times and similar integer values to be read by the DCS

Internal words.*

We assume here that the PLC subsystem is used only for digital I/O and sequencing. Analog inputs and outputs are directly handled by the DCS. All read- and write-only coils are "bit packed" into a series of contiguous words for ease in communication with the DCS. The terms input and output are used with respect to the PLC subsystem.

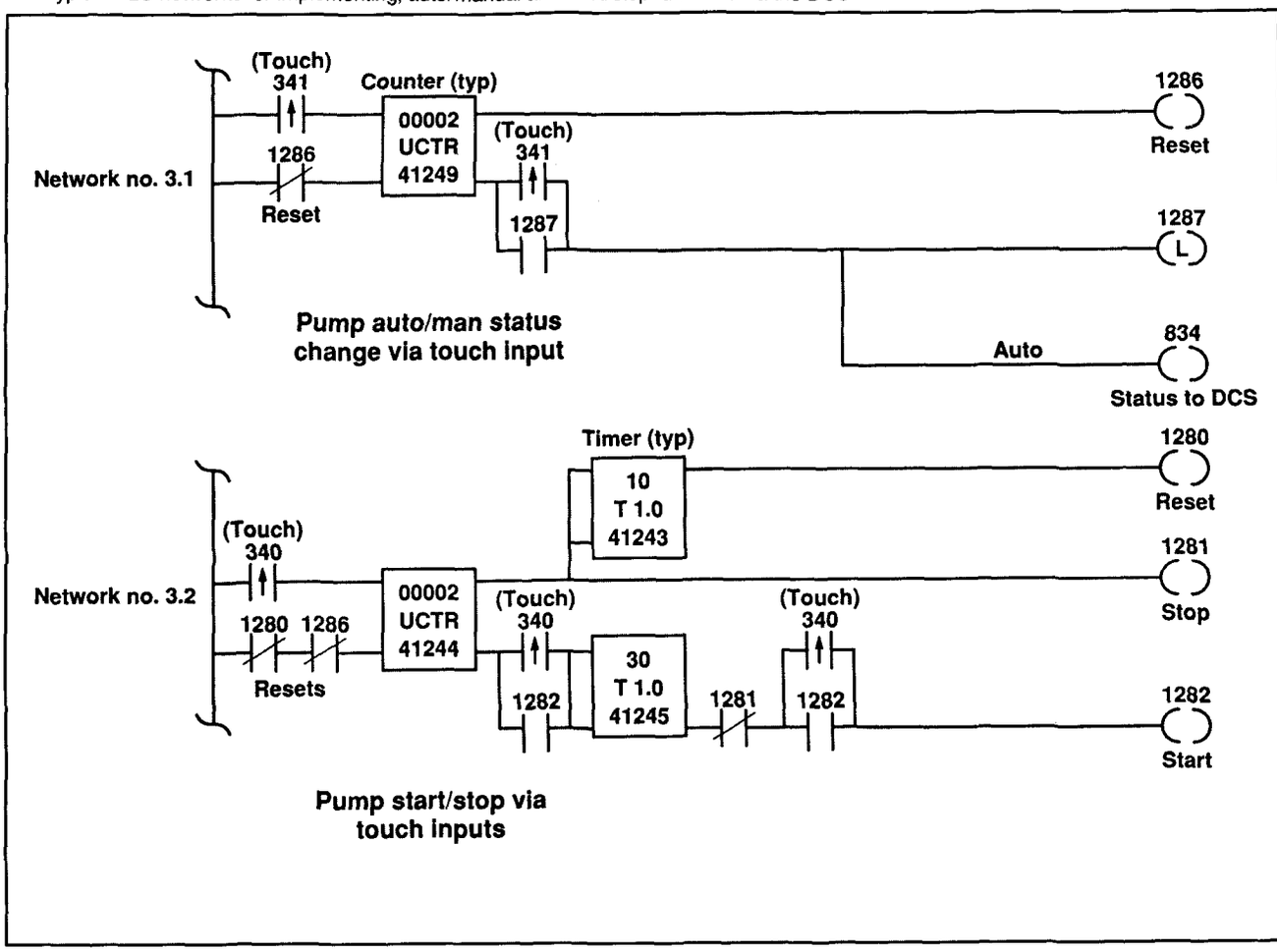
Figure 2 shows a hypothetical example of coils and word assignments for a typical PLC. The PLC coils have been classified into write-only and read-only coils. The write-only coils receive motor start and stop commands from the DCS (i.e., touch inputs from a touch screen), and the read-only coils are the alarm and status coils. All coils are finally bit packed, or copied, into read and write words shown in Group III. Note the large blocks of contiguous words assigned for efficient DCS-PLC communications.

In Fig. 2, DCS-PLC communication block no. 1 consists of 46 words. These words consist of bit-packed discrete data (coils). Communication block no. 2 consists of 64 words. Each word represents an integer value. Similar PLC-DCS communication blocks can be designed for each PLC in the system. Figure 2 does not show the internal coils and words that are not part of the data exchange.

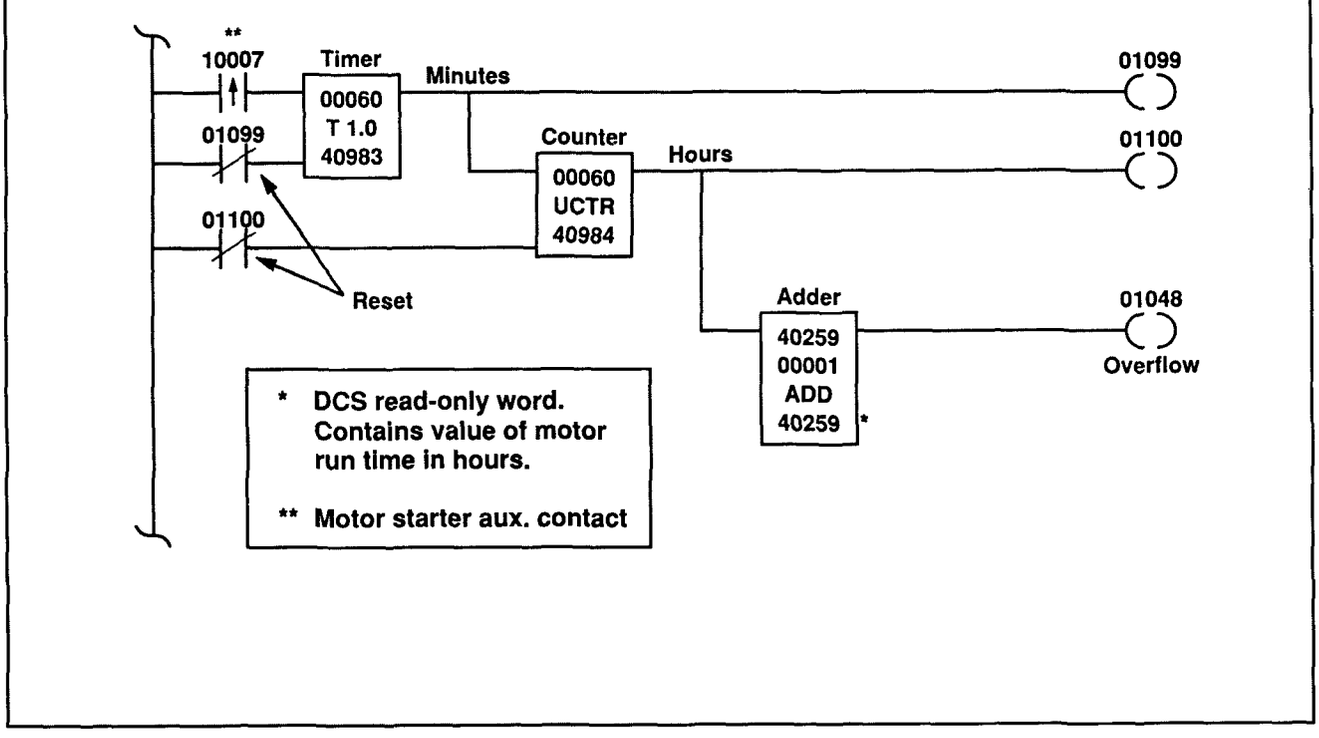
The PLC interface modules

The number of PLC interface modules (PLCI, Fig. 1) required in the DCS will depend on the number of words involved in the PLC- DCS communication blocks, and the

3. Typical PLC networks for implementing, auto/manual and start/stop functions via the DCS



4. A PLC network for motor run time totalization



capacity of each PLCI module. Usually PLC interface module capacity is expressed in terms of (a) number of words of discrete data that can be handled and (b) number of words of integer values that can be handled (each integer value being placed in the individual word). For example, let's assume that a typical PLC interface card can handle 30 words of discrete data (480 coils) or 120 integer values (120 words). Then, for the example PLC shown in Fig. 2, we will need three PLC interface modules. Two PLCIs can be assigned to handle 46 words of discrete data, and the third PLCI can be assigned to handle 64 words of integer values. This arrangement leaves enough room for future expansion. In other words, each PLC with the data communication requirements shown in Fig. 2 will require three PLCI modules. A total of nine PLCI modules will be required for the three PLCs shown in Fig. 1, assuming they have similar data exchange requirements. Both coil data and integer value words can be mixed in a PLC interface.

Different vendors use terms such as data highway processor or programmable controller interface unit to define the PLC interface module, but the basic concept is the same. The latest concept in the data communication area is the gateway. A gateway is much faster than the earlier interface devices, generally communicates directly with the DCS man-machine interface, and is used exclusively to exchange real-time supervisory control related data. A gateway can handle 1500 to 3000 tags, depending on the manufacturer, and is equivalent to several PLC interface modules of earlier design.

Philosophy of push-button and selector-switch implementations via the DCS system

The philosophy of the push-button and selector-switch implementations via the DCS system[1] should be established very early in the project. The PLC implementation group should ask the following questions:

- Is the DCS a touch-screen based system?
- Should the start and stop or auto/manual switch network be designed based on receiving momentary contact closure input from the DCS, or should it be based on receiving maintained contact type inputs from the DCS?

Figure 3 shows typical network implementations of auto/manual switch and start/stop push-button based on receiving touch inputs from a touch screen based graphic display part of the DCS system.

Network 3.1 in Fig. 3 shows the implementation of an auto/manual switch, via the DCS touch screen. The switch is part of the graphic display. When the switch is touched once (coil no. 341), then coil no. 1287 is set to 1, putting the switch in auto mode. Status coil no. 834 is also set to 1, reporting the auto status back to the DCS. When the switch is touched again, then coil no. 1287 changes to manual mode, likewise status coil no. 834 changes status to "0". Note that coil no. 1287 is a latching type, to preserve the status of the coil following a power failure.

Network 3.2 in Fig. 3 shows the implementation of a start/stop push-button via a DCS touch screen. When the pump symbol is touched once (coil no. 340), then coil no. 1282 is set to 1 for 30 seconds, issuing a pulse to the pump start network. When the pump symbol is touched again, coil no. 1281 is set to 1 for 10 seconds, issuing a stop pulse to the pump network. Note the presence of contact no. 1286 in Network 3.2. This contact ensures that the start/stop counters in Network 3.2 are reset to zero whenever the pump is put in manual mode using Network 3.1.

Minimizing load on the DCS

To minimize load on the DCS as far as is practical, all computations related to the equipment controlled by the PLC subsystem should be performed within the PLC. This reduces computational burden on the DCS, and in some cases may also reduce traffic on the data highway, improving overall system response.

Computation of motor run time is one such example. **Figure 4** shows the PLC network implementation of motor run time totalization. The four-digit run time value is stored in a DCS read-only word.

Preparation of the DCS-PLC interface configuration data

The PLC design group should prepare the data that needs to be supplied to the DCS supplier so that the supplier can configure the DCS operator interface for the PLC subsystem. **Tables I and II** show typical organization of the data. Table I summarizes all words containing discrete data coils, and Table II summarizes all words containing integer values. Only those words involved in the information exchange between the DCS and PLC subsystems need to be listed in these tables.

The DCS subsystem

Once the DCS-PLC interface configuration data is supplied by the PLC design group (in the form outlined in Tables I and II), the DCS supplier should be able to design a look-up table that translates the PLC addresses to the DCS addresses and vice-versa. These look-up tables can be very useful in configuration, debugging, and testing of the DCS-PLC system. Tables I and II should be given to the DCS supplier as soon as possible to avoid delays in the project implementation. It is the responsibility of the DCS group to transfer the information contained in Tables I and II to the proper places in the graphic displays, alarms, and status summaries.

The test plan

Last but not least, a comprehensive, well written test plan is the key to the success of any DCS project. The test plan should be jointly written and mutually reviewed by the DCS and PLC groups.

It is not our intent to outline a detailed test plan in this paper; however, as a minimum, the following items related to the testing of the DCS-PLC interface should be included in any test plan:

- Test the auto/manual switches and start/stop push-button functions from the DCS touch screens by verifying that proper write-only coils change status in the PLC, including the corresponding internal coils and real-world outputs.
- Perform manual data entry at the DCS consoles (i.e., timer and counter values), and verify that corresponding write-only words receive the proper values and that the

I. PLCI configuration data for DCS supplier (words containing discrete data)

DCS read-only words and coils alarm and status				*DCS write-only words and coils start and stop commands*			
Word no./bit. no.	*Coil no.*	*Tag no.*	*Description*	*Word no./bit no.*	*Coil no.*	*Tag no.*	*Description*
40117/0	00513	EAH-127	Agit. no. 1 overload	40101/0	00257	M127	Agit. no. 1 start/stop
40117/1	00514	EAH-128	Agit. no. 2 overload	40101/1	00258	M128	Agit. no. 2 start/stop
.	.			.	.	.	
.	.			.	.	.	
.	.			.	.	.	
.	.			.	.	.	
40117/14	00527	EAH-141	FP no. 1 overload	40101/14	00271	M141	Feedpump no. 1 start/stop
40117/15	00528	EAH-142	FP no. 2 overload	40101/15	00272	M142	Feedpump no. 2 start/stop
.	.			.	.	.	
.	.			.	.	.	
.	.			.	.	.	
40146/0	00977	ELH-367	Compr. no. 1 run	40116/0	00497	M367	Compr. no. 1 start/stop
40146/1	00978	ELH-368	Compr. no. 2 run	40116/1	00498	M368	Compr. no. 2 start/stop
.	.			.	.	.	
.	.			.	.	.	
.	.			.	.	.	
40146/14	00991	ELH-381	Caust. P1 run	40116/14	00511	M381	Caust. P1 start/stop
40146/15	00992	ELH-382	Caust. P2 run	40116/15	00512	M382	Caust. P2 start/stop

II. PLCI configuration data for DCS supplier (words containing integer values)

DCS read-only words motor run time, etc.			*DCS write-only words manual data entry*		
Word no.	*Tag no.*	*Description*	*Word no.*	*Tag no.*	*Description*
40179	KQ127	Agitator no. 1 run time	40147	...	B/W interval
40180	KQ128	Agitator no. 2 run time	40148	...	B/W duration
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
40209	KQ381	Caust. P1 run time	40177	...	Agit. no. 1 run interval
40210	KQ382	Caust. P2 run time	40178	...	Agit. no. 1 run duration

values are ultimately copied into the timers or counter numerical data storage words of the proper network.

- In the PLC, force the status of the DCS read-only coils (alarm/status) and verify the status changes on the DCS graphic displays or alarm summaries. This test should include motor auxiliary contacts, limit switches, and other related field inputs.
- In the PLC, load the DCS read-only words with simulated data and verify that the same data is shown on the DCS operator interface.
- Evaluate the overall response of the system. One simple way to look at the response time is to measure the elapsed time between a motor start command issued at the touch screen and the change in the symbol color at the graphic display indicating the closure of the corresponding motor starter auxiliary contact. This response time will vary depending on the system conditions (i.e., the data highway traffic, number of consoles simultaneously being used by the operators, or multiple operators trying to access the same PLC). A response time of less than two seconds under light load and five seconds under loaded conditions should be considered adequate. Similar tests could be applied to numerical values entered at the DCS and observed at the PLC or vice-versa.

These tests are designed to test the DCS read-only and write-only coils and words. Tables I and II and the look-up table developed by the DCS group should be very useful for these tests. Again, we have discussed only the tests related to the DCS-PLC interface. For other DCS related testing, readers are referred to the literature.[2]

Conclusion

We have tried to outline important design considerations that need to be taken into account for efficient implementation of DCS-PLC interfaces. Major emphasis is placed on the PLC subsystem because this is the subsystem that needs more attention during the design and planning stages.□

Received for review May 31, 1990.

Accepted Aug. 13, 1990.

Presented at the TAPPI 1990 Engineering Conference.

[2]Keskar, P. Y., ISA/86 International Conference Proceedings, Instrument Society of America, Research Triangle Park, N.C., 1986, p. 975.