5. NETWORKING

<TODO - get AB ethernet specs for MSG instruction>

<TODO - clean up internet materials>

Topics:
- Networks; topology, OSI model, hardware and design issues
- Network types; Devicenet, CANbus, Controlnet, Ethernet, and DH+
- Design case

Objectives:
- To understand network types and related issues
- Be able to network using Devicenet, Ethernet and DH+

5.1 INTRODUCTION

A computer with a single network interface can communicate with many other computers. This economy and flexibility has made networks the interface of choice, eclipsing point-to-point methods such as RS-232. Typical advantages of networks include resource sharing and ease of communication. But, networks do require more knowledge and understanding.

Small networks are often called Local Area Networks (LANs). These may connect a few hundred computers within a distance of hundreds of meters. These networks are inexpensive, often costing $100 or less per network node. Data can be transmitted at rates of millions of bits per second. Many controls system are using networks to communicate with other controllers and computers. Typical applications include;

- taking quality readings with a PLC and sending the data to a database computer.
- distributing recipes or special orders to batch processing equipment.
- remote monitoring of equipment.

Larger Wide Area Networks (WANs) are used for communicating over long distances between LANs. These are not common in controls applications, but might be needed for a very large scale process. An example might be an oil pipeline control system that is spread over thousands of miles.

5.1.1 Topology

The structure of a network is called the topology. Figure 59 shows the basic network topologies. The *Bus* and *Ring* topologies both share the same network wire. In the *Star* configuration each computer has a single wire that connects it to a central hub.

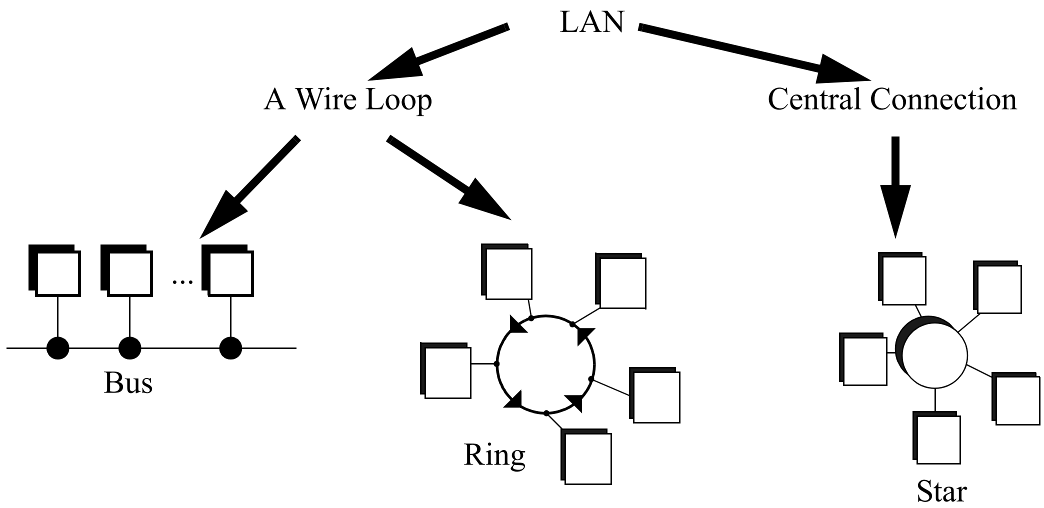


Figure 59 Network Topologies

In the *Ring* and *Bus* topologies the network control is distributed between all of the computers on the network. The wiring only uses a single loop or run of wire. But, because there is only one wire, the network will slow down significantly as traffic increases. This also requires more sophisticated network interfaces that can determine when a computer is allowed to transmit messages. It is also possible for a problem on the network wires to halt the entire network.

The *Star* topology requires more wire overall to connect each computer to an intelligent hub. But, the network interfaces in the computer become simpler, and the network becomes more reliable. Another term commonly used is that it is deterministic, this means that performance can be predicted. This can be important in critical applications.

For a factory environment the bus topology is popular. The large number of wires required for a star configuration can be expensive and confusing. The loop of wire required for a ring topology is also difficult to connect, and it can lead to ground loop problems. Figure 60 shows a tree topology that is constructed out of smaller bus networks. Repeaters are used to boost the signal strength and allow the network to be larger.

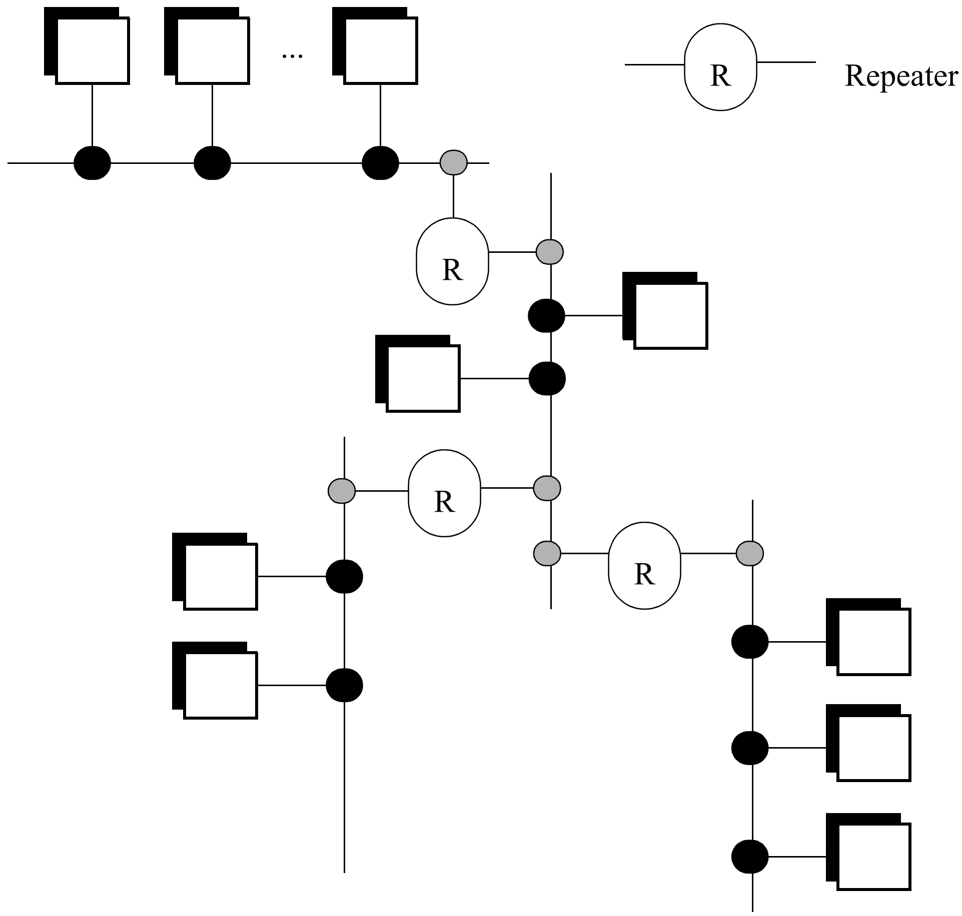
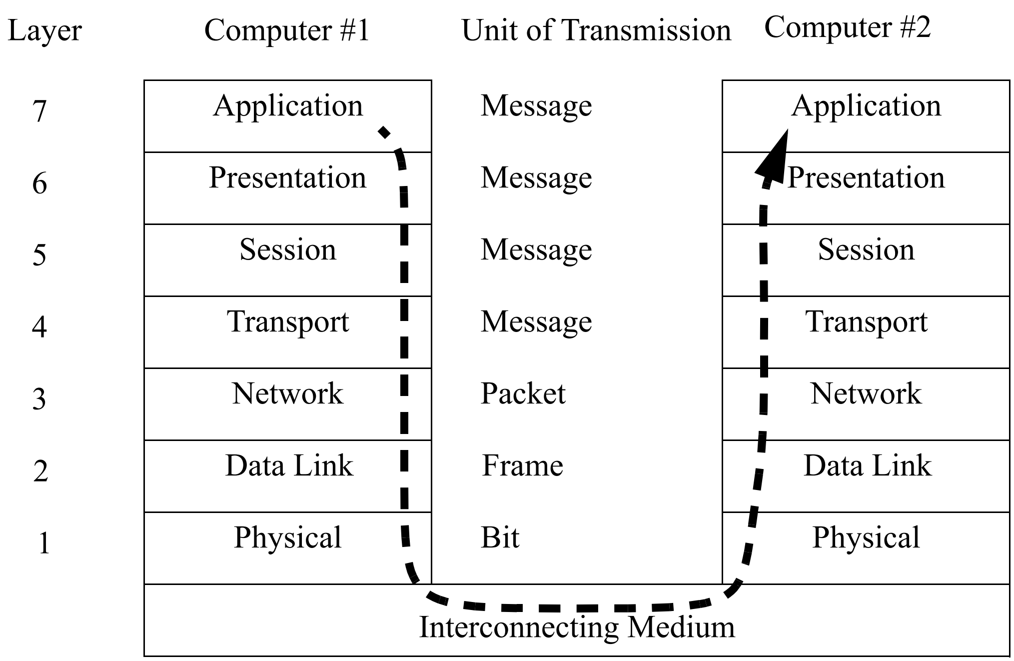


Figure 60 The Tree Topology

5.1.2 OSI Network Model

The Open System Interconnection (OSI) model in Figure 61 was developed as a tool to describe the various hardware and software parts found in a network system. It is most useful for educational purposes, and explaining the things that should happen for a successful network application. The model contains seven layers, with the hardware at the bottom, and the software at the top. The darkened arrow shows that a message originating in an application program in computer #1 must travel through all of the layers in both computers to arrive at the application in computer #2. This could be part of the process of reading email.



Application - This is high level software on the computer.
Presentation - Translates application requests into network operations.
Session - This deals with multiple interactions between computers.
Transport - Breaks up and recombines data to small packets.
Network - Network addresses and routing added to make frame.
Data Link - The encryption for many bits, including error correction added to a frame.
Physical - The voltage and timing for a single bit in a frame.
Interconnecting Medium - (not part of the standard) The wires or transmission medium of the network.

Figure 61 The OSI Network Model

The *Physical* layer describes items such as voltage levels and timing for the transmission of single bits. The *Data Link* layer deals with sending a small amount of data, such as a byte, and error correction. Together, these two layers would describe the serial byte shown in the previous chapter. The *Network* layer determines how to move the message through the network. If this were for an internet connection this layer would be responsible for adding the correct network address. The *Transport* layer will divide small amounts of data into smaller packets, or recombine them into one larger piece. This layer also checks for data integrity, often with a checksum. The *Session* layer will deal with issues that go beyond a single block of data. In particular it will deal with resuming transmission if it is interrupted or corrupted. The *Session* layer will often make long term connections to the remote machine. The *Presentation* layer acts as an application interface so that syntax, formats and codes are consistent between the two networked machines. For example this might convert '\' to '/' in HTML files. This layer also provides subroutines that the user may call to access network functions, and perform functions such as encryption and compression. The *Application* layer is where the user program resides. On a computer this might be a web browser, or a ladder logic program on a PLC.

Most products can be described with only a couple of layers. Some networking products may omit layers in the model.

5.1.3 Networking Hardware

The following is a description of most of the hardware that will be needed in the design of networks.

- Computer (or network enabled equipment)
- Network Interface Hardware - The network interface may already be built into the computer/PLC/sensor/etc. These may cost $15 to over $1000.
- The Media - The physical network connection between network nodes.
  - 10baseT (twisted pair) is the most popular. It is a pair of twisted copper wires terminated with an RJ-45 connector.
  - 10base2 (thin wire) is thin shielded coaxial cable with BNC connectors
  - 10baseF (fiber optic) is costly, but signal transmission and noise properties are very good.
- Repeaters (Physical Layer) - These accept signals and retransmit them so that longer networks can be built.
- Hub/Concentrator - A central connection point that network wires will be connected to. It will pass network packets to local computers, or to remote networks if they are available.
- Router (Network Layer) - Will isolate different networks, but redirect traffic to other LANs.
- Bridges (Data link layer) - These are intelligent devices that can convert data on one type of network, to data on another type of network. These can also be used to isolate two networks.
- Gateway (Application Layer) - A Gateway is a full computer that will direct traffic to different networks, and possibly screen packets. These are often used to create firewalls for security.

Figure 62 shows the basic OSI model equivalents for some of the networking hardware described before.

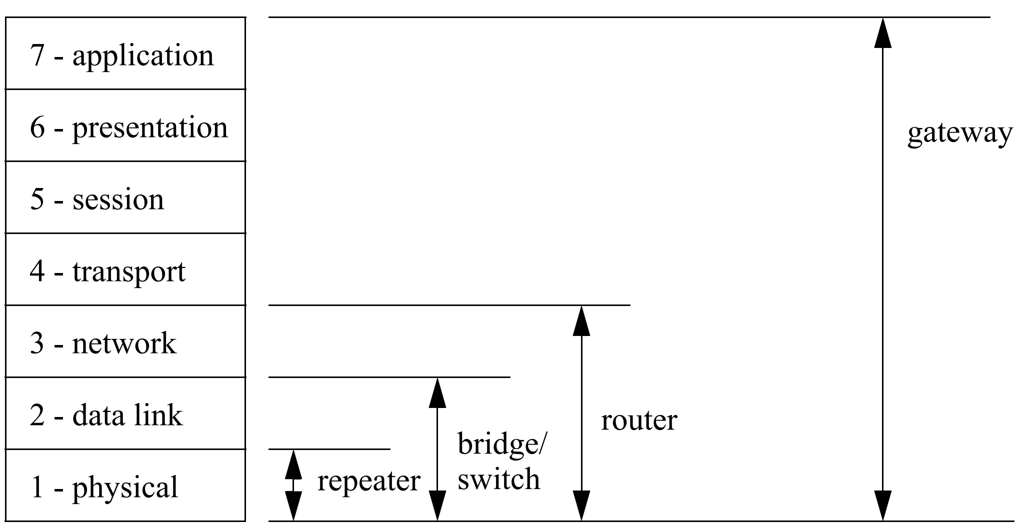


Figure 62 Network Devices and the OSI Model

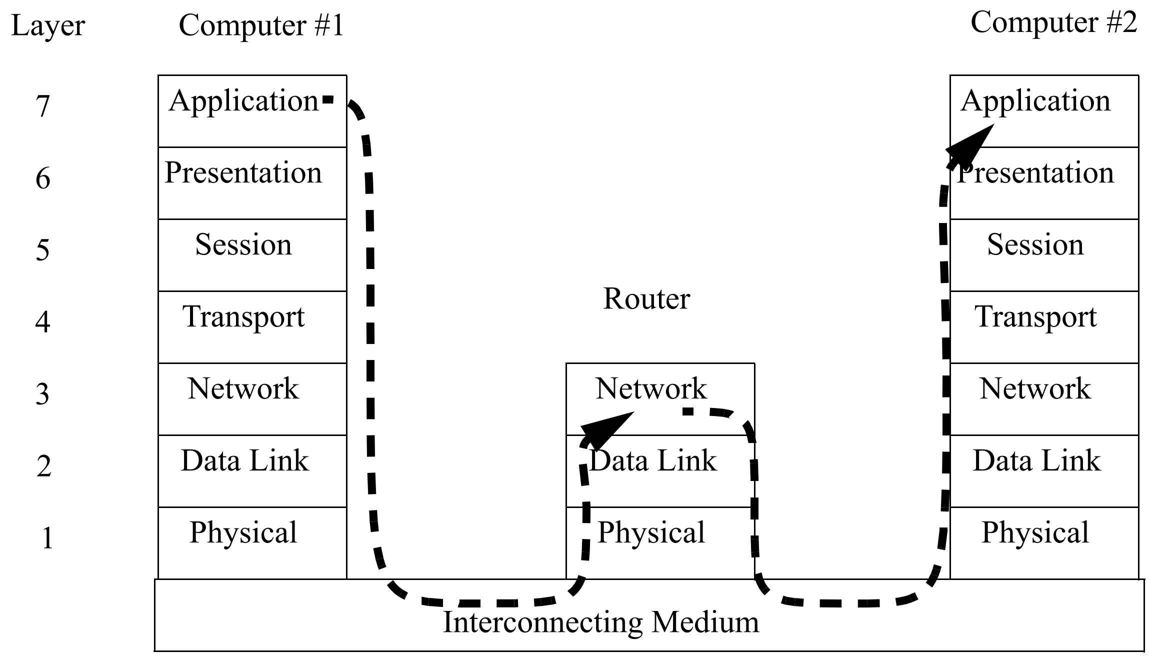


Figure 63 The OSI Network Model with a Router

5.1.4 Control Network Issues

A wide variety of networks are commercially available, and each has particular strengths and weaknesses. The differences arise from their basic designs. One simple issue is the use of the network to deliver power to the nodes. Some control networks will also supply enough power to drive some sensors and simple devices. This can eliminate separate power supplies, but it can reduce the data transmission rates on the network. The use of network taps or tees to connect to the network cable is also important. Some taps or tees are simple *passive* electrical connections, but others involve sophisticated *active* tees that are more costly, but allow longer networks.

The transmission type determines the communication speed and noise immunity. The simplest transmission method is baseband, where voltages are switched off and on to signal bit states. This method is subject to noise, and must operate at lower speeds. RS-232 is an example of baseband transmission. Carrierband transmission uses FSK (Frequency Shift Keying) that will switch a signal between two frequencies to indicate a true or false bit. This technique is very similar to FM (Frequency Modulation) radio where the frequency of the audio wave is transmitted by changing the frequency of a carrier frequency about 100MHz. This method allows higher transmission speeds, with reduced noise effects. Broadband networks transmit data over more than one channel by using multiple carrier frequencies on the same wire. This is similar to sending many cable television channels over the same wire. These networks can achieve very large transmission speeds, and can also be used to guarantee real time network access.

The bus network topology only uses a single transmission wire for all nodes. If all of the nodes decide to send messages simultaneously, the messages would be corrupted (a collision occurs). There are a variety of methods for dealing with network collisions, and arbitration.

CSMA/CD (Collision Sense Multiple Access/Collision Detection) - if two nodes start talking and detect a collision then they will stop, wait a random time, and then start again.
CSMA/BA (Collision Sense Multiple Access/Bitwise Arbitration) - if two nodes start talking at the same time the will stop and use their node addresses to determine which one goes first.
Master-Slave - one device one the network is the master and is the only one that may start communication. slave devices will only respond to requests from the master.
Token Passing - A token, or permission to talk, is passed sequentially around a network so that only one station may talk at a time.

The token passing method is deterministic, but it may require that a node with an urgent message wait to receive the token. The master-slave method will put a single machine in charge of sending and receiving. This can be restrictive if multiple controllers are to exist on the same network. The CSMA/CD and CSMA/BA methods will both allow nodes to talk when needed. But, as the number of collisions increase the network performance degrades quickly.

5.2 NETWORK STANDARDS

Bus types are listed below.

Low level busses - these are low level protocols that other networks are built upon.
RS-485, Bitbus, CAN bus, Lonworks, Arcnet
General open buses - these are complete network types with fully published standards.
ASI, Devicenet, Interbus-S, Profibus, Smart Distributed System (SDS), Seriplex
Specialty buses - these are buses that are proprietary.
Genius I/O, Sensoplex

5.2.1 Devicenet

Devicenet has become one of the most widely supported control networks. It is an open standard, so components from a variety of manufacturers can be used together in the same control system. It is supported and promoted by the Open Devicenet Vendors Association (ODVA) (see http://www.odva.org). This group includes members from all of the major controls manufacturers.

This network has been designed to be noise resistant and robust. One major change for the control engineer is that the PLC chassis can be eliminated and the network can be connected directly to the sensors and actuators. This will reduce the total amount of wiring by moving I/O points closer to the application point. This can also simplify the connection of complex devices, such as HMIs. Two way communications inputs and outputs allow diagnosis of network problems from the main controller.

Devicenet covers all seven layers of the OSI standard. The protocol has a limited number of network address, with very small data packets. But this also helps limit network traffic and ensure responsiveness. The length of the network cables will limit the maximum speed of the network. The basic features of are listed below.

• A single bus cable that delivers data and power.
• Up to 64 nodes on the network.
• Data packet size of 0-8 bytes.
• Lengths of 500m/250m/100m for speeds of 125kbps/250kbps/500kbps respectively.
• Devices can be added/removed while power is on.
• Based on the CANbus (Controller Area Network) protocol for OSI levels 1 and 2.
• Addressing includes peer-to-peer, multicast, master/slave, polling or change of state.

An example of a Devicenet network is shown in Figure 64. The dark black lines are the network cable. Terminators are required at the ends of the network cable to reduce electrical noise. In this case the PC would probably be running some sort of software based PLC program. The computer would have a card that can communicate with Devicenet devices. The *FlexIO rack* is a miniature rack that can hold various types of input and output modules. Power taps (or tees) split the signal to small side branches. In this case one of the taps connects a power supply, to provide the 24Vdc supply to the network. Another two taps are used to connect a *smart sensor* and another *FlexIO rack*. The *Smart sensor* uses power from the network, and contains enough logic so that it is one node on the network. The network uses *thin trunk line* and *thick trunk line* which may limit network performance.

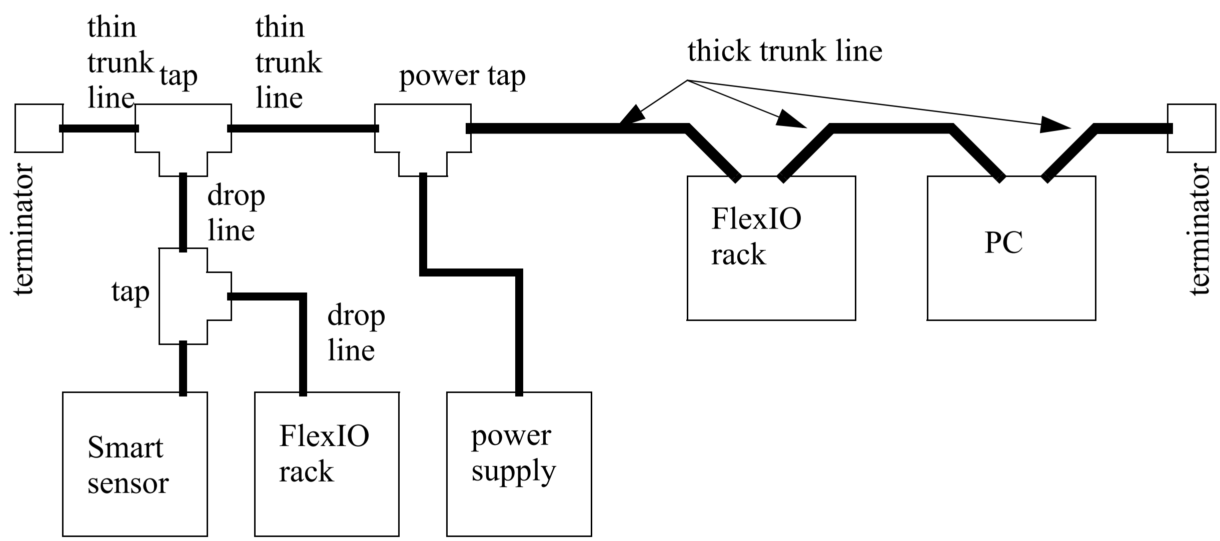


Figure 64 A Devicenet Network

The network cable is important for delivering power and data. Figure 65 shows a basic cable with two wires for data and two wires for the power. The cable is also shielded to reduce the effects of electrical noise. The two basic types are thick and thin trunk line. The cables may come with a variety of connections to devices.

• bare wires
• unsealed screw connector
• sealed mini connector
• sealed micro connector
• vampire taps

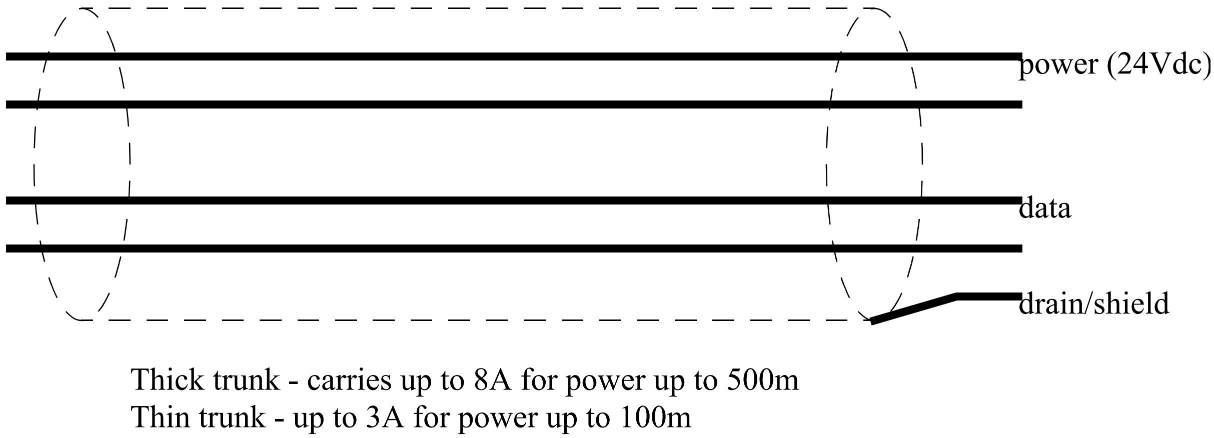
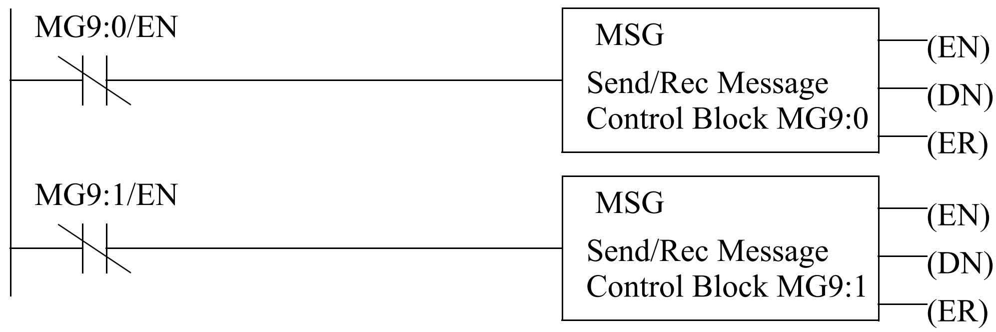


Figure 65 Shielded Network Cable

Some of the design issues for this network include;

- Power supplies are directly connected to the network power lines.
- Length to speed is 156m/78m/39m to 125Kbps/250Kbps/500Kbps respectively.
- A single drop is limited to 6m.
- Each node on the network will have its own address between 0 and 63.

If a PLC-5 was to be connected to Devicenet a scanner card would need to be placed in the rack. The ladder logic in Figure 66 would communicate with the sensors through a scanner card in slot 3. The read and write blocks would read and write the Devicenet input values to integer memory from *N7:40* to *N7:59*. The outputs would be copied from the integer memory between *N7:20* to *N7:39*. The ladder logic to process inputs and outputs would need to examine and set bits in integer memory.



MG9:0

Read/Write	Write
Data Table	N7:20
Size	20
Local/Remote	Remote
Remote Station	??
Link ID	??
Remote Link type	??
Local Node Addr.	N/A
Processor Type	????
Dest. Addr.	????

MG9:1

Read/Write	Read
Data Table	N7:40
Size	20
Local/Remote	Remote
Remote Station	??
Link ID	??
Remote Link type	??
Local Node Addr.	N/A
Processor Type	????
Dest. Addr.	????

Note: Get exact settings for these parametersXXXXXXXXXXXXXXXXX

Figure 66 Communicating with Devicenet Inputs and Outputs

On an Allen Bradley Softlogix PLC the I/O will be copied into blocks of integer memory. These blocks are selected by the user in setup software. The ladder logic would then using integer memory for inputs and outputs, as shown in Figure 67. Here the inputs are copied into N9 integer memory, and the outputs are set by copying the N10 block of memory back to the outputs.

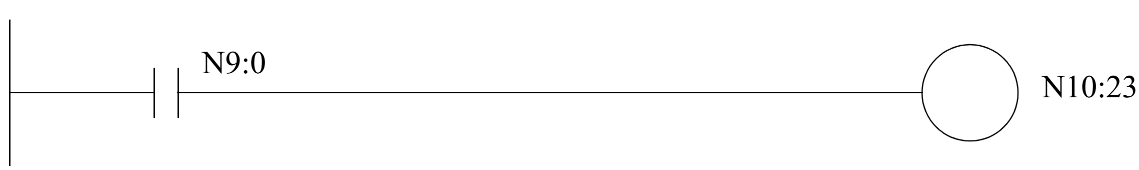


Figure 67 Devicenet Inputs and Outputs in Software Based PLCs

5.2.2 CANbus

The CANbus (Controller Area Network bus) standard is part of the Devicenet standard. Integrated circuits are now sold by many of the major vendors (Motorola, Intel, etc.) that support some, or all, of the standard on a single chip. This section will discuss many of the technical details of the standard.

CANbus covers the first two layers of the OSI model. The network has a bus topology and uses bit wise resolution for collisions on the network (i.e., the lower the network identifier, the higher the priority for sending). A data frame is shown in Figure 68. The frame is like a long serial byte, like that seen in the previous chapter. The frame begins with a start bit. This is then followed with a message identifier. For Devicenet this is a 5 bit address code (for up to 64 nodes) and a 6 bit command code. The *ready to receive it* bit will be set by the receiving machine. (Note: both the sender and listener share the same wire.) If the receiving machine does not set this bit the remainder of the message is aborted, and the message is resent later. While sending the first few bits, the sender monitors the bits to ensure that the bits send are heard the same way. If the bits do not agree, then another node on the network has tried to write a message at the same time - there was a collision. The two devices then wait a period of time, based on their identifier and then start to resend. The second node will then detect the message, and wait until it is done. The next 6 bits indicate the number of bytes to be sent, from 0 to 8. This is followed by two sets of bits for CRC (Cyclic Redundancy Check) error checking, this is a checksum of earlier bits. The next bit *ACK slot* is set by the receiving node if the data was received correctly. If there was a CRC error this bit would not be set, and the message would be resent. The remaining bits end the transmission. The *end of frame* bits are equivalent to stop bits. There must be a delay of at least 3 bits before the next message begins.

1 bit	start of frame	
11 bits	identifier	arbitration field
1 bit	ready to receive it	
6 bits	control field - contains number of data bytes	
0-8 bytes	data - the information to be passed	
15 bits	CRC sequence	
1 bit	CRC delimiter	
1 bit	ACK slot - other listeners turn this on to indicate frame received	
1 bit	ACK delimiter	
7 bits	end of frame	
>= 3 bits	delay before next frame	

Figure 68 A CANbus Data Frame

Because of the bitwise arbitration, the address with the lowest identifier will get the highest priority, and be able to send messages faster when there is a conflict. As a result the controller is normally put at address *0*. And, lower priority devices are put near the end of the address range.

5.2.3 Controlnet

Controlnet is complimentary to Devicenet. It is also supported by a consortium of companies, (http://www.controlnet.org) and it conducts some projects in cooperation with the Devicenet group. The standard is designed for communication between controllers, and permits more complex messages than Devicenet. It is not suitable for communication with individual sensors and actuators, or with devices off the factory floor.

Controlnet is more complicated method than Devicenet. Some of the key features of this network include,

• Multiple controllers and I/O on one network
• Deterministic
• Data rates up to 5Mbps
• Multiple topologies (bus, star, tree)

• Multiple media (coax, fiber, etc.)
• Up to 99 nodes with addresses, up to 48 without a repeater
• Data packets up to 510 bytes
• Unlimited I/O points
• Maximum length examples
    1000m with coax at 5Mbps - 2 nodes
    250m with coax at 5Mbps - 48 nodes
    5000m with coax at 5Mbps with repeaters
    3000m with fiber at 5Mbps
    30Km with fiber at 5Mbps and repeaters
• 5 repeaters in series, 48 segments in parallel
• Devices powered individually (no network power)
• Devices can be removed while network is active

This control network is unique because it supports a real-time messaging scheme called Concurrent Time Domain Multiple Access (CTDMA). The network has a scheduled (high priority) and unscheduled (low priority) update. When collisions are detected, the system will wait a time of at least 2ms, for unscheduled messages. But, scheduled messages will be passed sooner, during a special time window.

5.2.4 Ethernet

Ethernet has become the predominate networking format. Version I was released in 1980 by a consortium of companies. In the 1980s various versions of ethernet frames were released. These include Version II and Novell Networking (IEEE 802.3). Most modern ethernet cards will support different types of frames.

The ethernet frame is shown in Figure 69. The first six bytes are the destination address for the message. If all of the bits in the bytes are set then any computer that receives the message will read it. The first three bytes of the address are specific to the card manufacturer, and the remaining bytes specify the remote address. The address is common for all versions of ethernet. The source address specifies the message sender. The first three bytes are specific to the card manufacturer. The remaining bytes include the source address. This is also identical in all versions of ethernet. The *ethernet type* identifies the frame as a Version II ethernet packet if the value is greater than 05DChex. The other ethernet types use these to bytes to indicate the datalength. The *data* can be between 46 to 1500 bytes in length. The frame concludes with a *checksum* that will be used to verify that the data has been transmitted correctly. When the end of the transmission is detected, the last four bytes are then used to verify that the frame was received correctly.

6 bytes	destination address
6 bytes	source address
2 bytes	ethernet type
46-1500 bytes	data
4 bytes	checksum

Figure 69 Ethernet Version II Frame

5.2.5 Profibus

Another control network that is popular in europe, but also available world wide. It is also promoted by a consortium of companies (http://www.profibus.com). General features include;

• A token passing between up to three masters
• Maximum of 126 nodes
• Straight bus topology
• Length from 9600m/9.6Kbps with 7 repeaters to 500m/12Mbps with 4 repeaters
• With fiber optic cable lengths can be over 80Km
• 2 data lines and shield
• Power needed at each station
• Uses RS-485, ethernet, fiber optics, etc.
• 2048 bits of I/O per network frame

5.2.6 Sercos

The SErial Real-time COmmunication System (SERCOS) is an open standard designed for multi-axis motion control systems. The motion controller and axes can be implemented separately and then connected using the SERCOS network. Many vendors offer cards that allow PLCs to act as clients and/or motion controllers.

• Deterministic with response times as small as a few nanoseconds
• Data rates of 2, 4, 8 and 16 Mbaud
• Documented with IEC 61491 in 1995 and 2002
• Uses a fiber optic rings, RS-485 and buses

5.3 PROPRIETARY NETWORKS

5.3.1 Data Highway

Allen-Bradley has developed the Data Highway II (DH+) network for passing data and programs between PLCs and to computers. This bus network allows up to 64 PLCs to be connected with a single twisted pair in a shielded cable. Token passing is used to control traffic on the network. Computers can also be connected to the DH+ network, with a network card to download programs and monitor the PLC. The network will support data rates of 57.6Kbps and 230 Kbps

The DH+ basic data frame is shown in Figure 70. The frame is byte oriented. The first byte is the *DLE* or delimiter byte, which is always $10. When this byte is received the PLC will interpret the next byte as a command. The *SOH* identifies the message as a DH+ message. The next byte indicates the destination station - each node one the network must have a unique number. This is followed by the *DLE* and *STX* bytes that identify the start of the data. The data follows, and its' length is determined by the command type - this will be discussed later. This is then followed by a *DLE* and *ETX* pair that mark the end of the message. The last byte transmitted is a checksum to determine the correctness of the message.

1 byte	DLE = 10H	header fields
1 byte	SOH = 01H	
1 byte	STN - the destination number	
1 byte	DLE = 10H	start fields
1 byte	STX = 02H	
	data	
1 byte	DLE = 10H	termination fields
1 byte	ETX = 03H	
1 byte	block check - a 2s compliment checksum of the DATA and STN values	

Figure 70 The Basic DH+ Data Frame

The general structure for the data is shown in Figure 71. This packet will change for different commands. The first two bytes indicate the destination, *DST*, and source, *SRC*, for the message. The next byte is the command, *CMD*, which will determine the action to be taken. Sometimes, the function, *FNC*, will be needed to modify the command. The transaction, *TNS*, field is a unique message identifier. The two address, *ADDR*, bytes identify a target memory location. The *DATA* fields contain the information to be passed. Finally, the *SIZE* of the data field is transmitted.

	1 byte	DST - destination node for the message
	1 byte	SRC - the node that sent the message
	1 byte	CMD - network command - sometime FNC is required
	1 byte	STS - message send/receive status
	2 byte	TNS - transaction field (a unique message ID)
optional	1 byte	FNC may be required with some CMD values
optional	2 byte	ADDR - a memory location
optional	variable	DATA - a variable length set of data
optional	1 byte	SIZE - size of a data field

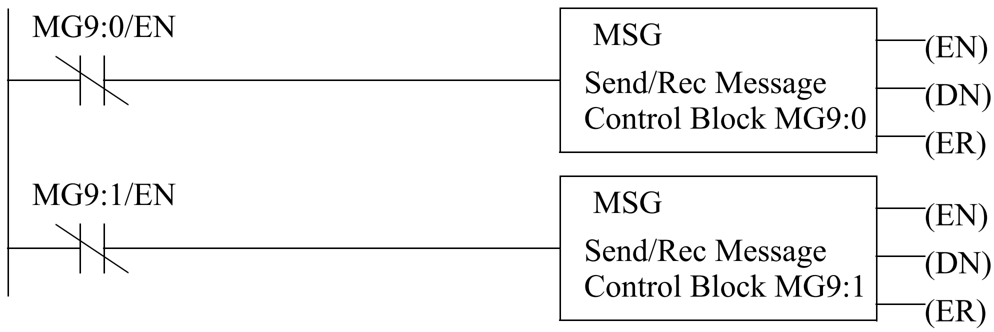
Figure 71 Data Filed Values

Examples of commands are shown in Figure 72. These focus on moving memory and status information between the PLC, and remote programming software, and other PLCs. More details can be found in the Allen-Bradley DH+ manuals.

CMD	FNC	Description
00		Protected write
01		Unprotected read
02		Protected bit write
05		Unprotected bit write
06	00	Echo
06	01	Read diagnostic counters
06	02	Set variables
06	03	Diagnostic status
06	04	Set timeout
06	05	Set NAKs
06	06	Set ENQs
06	07	Read diagnostic counters
08		Unprotected write
0F	00	Word range write
0F	01	Word range read
0F	02	Bit write
0F	11	Get edit resource
0F	17	Read bytes physical
0F	18	Write bits physical
0F	26	Read-modify-write
0F	29	Read section size
0F	3A	Set CPU mode
0F	41	Disable forces
0F	50	Download all request
0F	52	Download completed
0F	53	Upload all request
0F	55	Upload completed
0F	57	Initialize memory
0F	5E	Modify PLC-2 compatibility file
0F	67	typed write
0F	68	typed read
0F	A2	Protected logical read - 3 address fields
0F	AA	Protected logical write - 3 addr. fields

Figure 72 DH+ Commands for a PLC-5 (all numbers are hexadecimal)

The ladder logic in Figure 73 can be used to copy data from the memory of one PLC to another. Unlike other networking schemes, there are no *login* procedures. In this example the first MSG instruction will write the message from the local memory *N7:20 - N7:39* to the remote PLC-5 (node 2) into its memory from *N7:40* to *N7:59*. The second MSG instruction will copy the memory from the remote PLC-5 memory *N7:40* to *N7:59* to the remote PLC-5 memory *N7:20* to *N7:39*. This transfer will require many scans of ladder logic, so the *EN* bits will prevent a read or write instruction from restarting until the previous *MSG* instruction is complete.



MG9:0

Read/Write	Write
Data Table	N7:20
Size	20
Local/Remote	Local
Remote Station	N/A
Link ID	N/A
Remote Link type	N/A
Local Node Addr.	2
Processor Type	PLC-5
Dest. Addr.	N7:40

MG9:1

Read/Write	Read
Data Table	N7:40
Size	20
Local/Remote	Local
Remote Station	N/A
Link ID	N/A
Remote Link type	N/A
Local Node Addr.	2
Processor Type	PLC-5
Dest. Addr.	N7:20

Figure 73 Ladder Logic for Reading and Writing to PLC Memory

The DH+ data packets can be transmitted over other data links, including ethernet and RS-232.

5.4 NETWORK COMPARISONS

Table 1: Network Comparison

Network	topology	addresses	length	speed	packet size
Bluetooth	wireless	8	10	64Kbps	continuous
CANopen	bus	127	25m-1000m	1Mbps-10Kbps	8 bytes
ControlNet	bus or star	99	250m-1000m wire, 3-30km fiber	5Mbps	0-510 bytes

Table 1: Network Comparison

Network	topology	addresses	length	speed	packet size
Devicenet	bus	64	500m	125-500Kbps	8 bytes
Ethernet	bus, star	1024	85m coax, 100m twisted pair, 400m-50km fiber	10-1000Gbps	46-1500bytes
Foundation Fieldbus	star	unlimited	100m twisted pair, 2km fiber	100Mbps	<=1500 bytes
Interbus	bus	512	12.8km with 400m segments	500-2000 Kbps	0-246 bytes
Lonworks	bus, ring, star	32,000	<=2km	78Kbps-1.25Mbps	228 bytes
Modbus	bus, star	250	350m	300bps-38.4Kbps	0-254 bytes
Profibus	bus, star, ring	126	100-1900m	9.6Kbps-12Mbps	0-244bytes
Sercos	rings	254	800m	2-16Mbps	32bits
USB	star	127	5m	>100Mbps	1-1000bytes

5.5 DESIGN CASES

5.5.1 Devicenet

Problem: A robot will be loading parts into a box until the box reaches a prescribed weight. A PLC will feed parts into a pickup fixture when it is empty. The PLC will tell the robot when to pick up a part and load it using Devicenet.

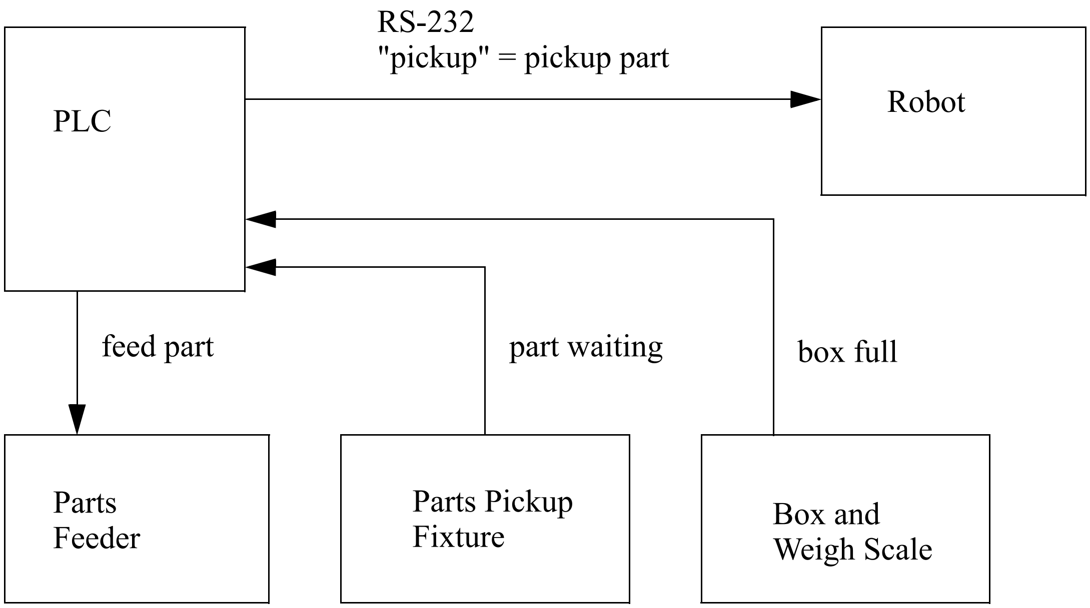


Figure 74 Box Loading System

Solution: The following ladder logic will implement part of the control system for the system in Figure 74.

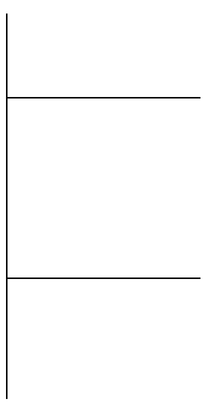


Figure 75 A Box Loading System

5.6 SUMMARY

• Networks come in a variety of topologies, but buses are most common on factory floors.
• The OSI model can help when describing network related hardware and software.
• Networks can be connected with a variety of routers, bridges, gateways, etc.
• Devicenet is designed for interfacing to a few inputs and outputs.
• Controlnet is designed for interfacing between controllers.

- Controlnet and devicenet are based on CANbus.
- Ethernet is common, and can be used for high speed communication.
- Profibus is another control network.

5.7 PRACTICE PROBLEMS

1. Explain why networks are important in manufacturing controls.

2. We will use a PLC to control a cereal box filling machine. For single runs the quantities of cereal types are controlled using timers. There are 6 different timers that control flow, and these result in different ratios of product. The values for the timer presets will be downloaded from another PLC using the DH+ network. Write the ladder logic for the PLC.

3.

a) We are developing ladder logic for an oven to be used in a baking facility. A PLC is controlling the temperature of an oven using an analog voltage output. The oven must be started with a push button and can be stopped at any time with a stop push button. A recipe is used to control the times at each temperature (this is written into the PLC memory by another PLC). When idle, the output voltage should be 0V, and during heating the output voltages, in sequence, are 5V, 7.5V, 9V. The timer preset values, in sequence, are in N7:0, N7:1, N7:2. When the oven is on, a value of 1 should be stored in N7:3, and when the oven is off, a value of 0 should be stored in N7:3. Draw a state diagram and write the ladder logic for this station.

b) We are using a PLC as a master controller in a baking facility. It will update recipes in remote PLCs using DH+. The master station is #1, the remote stations are #2 and #3. When an operator pushes one of three buttons, it will change the recipes in two remote PLCs if both of the remote PLCs are idle. While the remote PLCs are running they will change words in their internal memories (N7:3=0 means idle and N7:3=1 means active). The new recipe values will be written to the remote PLCs using DH+. The table below shows the values for each PLC. Write the ladder logic for the master controller.

	button A	button B	button C
PLC #2	13 690 45	17 235 75	14 745 34
PLC #3	76 345 987 345 764 87	72 234 12 34 456 67	56 645 23 456 568 8

4. A controls network is to be 1500m long. Suggest three different types of networks that would meet the specifications.

5 How many data bytes (maximum) could be transferred in one second with DH+?

6. Is the OSI model able to describe all networked systems?

7. What are the different methods for resolving collisions on a bus network?

5.8 ASSIGNMENT PROBLEMS

1. Describe an application for DH networking.

2. The response times of hydraulic switches is being tested in a PLC controlled station. When the units arrive a 'part present' sensor turns on. The part is then clamped in place by turning on a 'clamp' output. 1 second after clamping, a 'flow' output is turned on to start the test. The response time is the delay between when 'flow' is turned on, and the 'engaged' input turns on. When the unit has responded, up to 10 seconds later, the 'flow' output is turned off, and the system is allowed to sit for 5 seconds to discharge before unclamping. The result of the test is written to one of the memory locations from F8:0 to F8:39, for a total of 40 separate tests. When 40 tests have been done, the memory block from F8:0 to F8:39 is sent to another PLC using DH+, and the process starts again. Write the ladder logic to control the station.

3. a) Controls are to be developed for a machine that packages golf tees. Each container will normally hold 1000 tees filled from three different hoppers, each containing a different color. For marketing purposes the ratio of colors is changed frequently. To make the controller easy to reconfigure, the number of tees from each hopper are stored in the memory locations N7:0, N7:1 and N7:2. The process is activated when an empty package arrives, activating a PRESENT input. When filling the package, the machine opens a single hopper with a solenoid, and counts the tees with an optical sensor, until the specified count has been surpassed. It then repeats the operation with the two other hoppers. When done, it activates a SEAL for 2 seconds to advance a heated ram that seals the package. After that, the DONE output is turned on until the PRESENT sensor turns off. Write the ladder logic for this process.

b) Write a ladder logic program that will read and parse values from an RS-232 input. The format of the input will be an eleven character line with three integer numbers separated by commas. The integers will be padded to three characters by padding with zeros. The line will be terminated with a CR and a LF. The three integers are to be parsed and stored in the memory locations N7:0, N7:1 and N7:2 to be used in a golf tee packaging machine.

4. A master PLC is located at the top of a mine shaft and controls an elevator system. A second PLC is located half a mile below to monitor the bottom of the elevator shaft. At the top of the mine shaft the PLC has inputs for the door (D), a top limit switch (T), and start (G) and stop (S) pushbuttons. The PLC has two outputs to apply power (P) to the motor, or reverse (R) the motor direction. The PLC at the bottom of the elevator shaft checks a bottom limit switch (B) and a door closed (C) sensor. The two PLCs are connected using DH+. Write ladder logic for both PLCs and indicate the communication settings. Use structured design techniques.