Engineer On a Disk

Overview: This note set is part of a larger collection of materials available at http://claymore.engineer.gvsu.edu. You are welcome to use the material under the license provided at http://claymore.engineer.gvsu.edu/eod/global/copyrght.html. As always any feedback you can provide will be welcomed.

email: jackh@gvsu.edu
phone: (616) 771-6755
fax: (616) 336-7215

1. TABLE OF CONTENTS

2. COMPUTER HARDWARE

This section focuses on elements of computer hardware that are important to a designer/engineer.

3. A BRIEF OVERVIEW OF COMPUTER HARDWARE

3.1 BASIC COMPONENTS

• A computer has the basic properties,
  - stores data
  - processing data
  - Inputs and outputs data

3.1.1 The components of a computer are,

• The Central Processing Unit (CPU),
  - Control Unit - processes machine language instructions
  - Arithmetic Logic Unit - performs mathematical operations faster by using specialized hardware
  - Cache - most new computers incorporate a local block of memory which is faster than remote memory. A copy of remote memory is stored here so that the CPU may work faster

• Memory,
  - Random Access Memory (RAM) - memory used for temporary storage of programs and data while the computer is on.
  - Read Only Memory (ROM) - permanent software and data is stored in ROM, so that the CPU may always access this. A common use for this is a 'start-up' routine, which allows a computer to load the operating system off of a disk.
  - ROM (traditional) - a chip which is made to store the program, this may not be changed.
  - PROM - a chip which the program may be 'burned' into once.
  - EPROM - memory is 'burned' in, but may be erased using ultraviolet light, and rewritten.
  - EEPROM - similar to EPROM, but may be electrically erased.

• Input/Output (I/O),

  - Main functions
  • Load and Store programs and data,
    - optical and magnetic disk drives
    - magnetic tapes
    - etc.

  • Communicate with users
    - CRT driver
    - keyboard

- mouse
- printer

• Communicate with process
  - RS-232 data lines
  -

3.1.2 Some I/O Devices

• Teleprinters or Teletypes
  - this technology is becoming obsolete very fast, but may be useful when permanent records of sessions
  - inexpensive but slow
  - can be operated at a distance by telephone
  - allows user input by keyboard, and output on printed paper

• Cathode Ray Tube (CRT)
  - the preferred display device
  - very flexible and user friendly
  - Inexpensive and fast
  - Available in color, and varying sizes
  - used for graphical and text output
  - can be used with features such as touch screen input for harsh factory floor environments

• Card Readers and Card Punches
  - outdated technology which allowed permanent recording of user entry on punched cards

• Printers
  - Very popular method of permanent output
  - Inexpensive and fast
  - Can produce varying qualities of print
  - color printers available
  - letter size paper output most common

• Magnetic and Optical Disks and Tapes
  - Inexpensive methods for transferring and storing data and programs
  - storage capacity ranges from Kilobytes to Gigabytes, and capacities are increasing quickly
  - Some are read and write only
  - costs are very low per Megabyte, and are still dropping
  - Storage media is still somewhat fragile, and requires certain level of protection
  - Older tapes and systems are oriented towards backup, but newer methods allow random access.

• mouse/button boxes/dial boxes/tablets/etc
  - Various input peripherals make input easier
  - very inexpensive
  - most allow continuous input more suited to user
  - allows more complex programs for user interaction
  - the mouse has become the most popular input device
  - most of these devices are not suited for the factory floor

• keyboard
  - the time honored standard for input
  - now very inexpensive, used by almost all computers
  - can be used by all people, and is universally understood

• Special Application Boards
  - becoming very inexpensive, and common
  - A/D and D/A boards allow interfacing to applications which are monitored and/or controlled by analog signals (these are very inexpensive). These are the main source of interface to Sensors
  - Video processing - allows video camera inputs to be read into memory, and the still pictures may then be used for vision processing.

3.2 AN EXAMPLE OF A COMPUTER IN MANUFACTURING

• PLC's are an example of a microcomputer used for industrial control

• The basic components in a PLC are,
  - CPU
  - RAM to store working variables, statuses, etc.
  - ROM to store the operating system, and programs for interpreting ladder logic, programming, etc.
  - A/D, D/A - To input and output analog signals from the PLC, these I/O devices are used
  - PIA (Parallel Interface Adapter) - used to drive parallel data lines for digital I/O. These lines are used for detecting switch closures, driving relays for outputs, turning indicator lights on and off, etc.
  - ACIA (Asynchronous Communications Interface Adaptor) - used to drive RS-232 ports (or other serial ports) for interface.

3.3 COMMERCIAL COMPUTERS

3.3.1 Mainframes

  - Uses a large, and custom designed CPU, in big cabinets

- Large, expensive, and require special housing
- Require specialized personnel to run and program
- Ideal for systems which support many user sharing data and services, such as an accounting system, or database
- Software may be expensive and hard to maintain
- Graphical user interfaces are limited
- Approximate cost > $100,000
- Common example are VAX mainframes

3.3.2 Super Computers

- Completely customized hardware, optimized for numerical (primarily vectored and floating point) calculations
- Suited to huge numerical problems
- Often requires a front-end computer for job queueing, post-processing, and pre-processing
- Tremendous cost > $1,000,000
- One example is the CRAY line of computers
- Often uses parallel and distributed processors, along with nitrogen (or similar) cooling systems
- Very limited spectrum of users

3.3.3 Workstations

- Computers based on leading edge of microprocessor technology, using Enhanced graphical display capabilities.
- These have displaced minicomputers, and created a market for users who have small budgets but need reasonable computer power.
- Run faster than PC’s, had better price/performance ratio
- suited to networking, shared disks, and virtual memory
- Perform multiprocessing, thus suited to multiple users.
- Approximate cost $10,000 to $500,000
- Example are SUN, HP, and Silicon Graphics Workstations
- Can often find software intended for mainframes, and for Personal Computers which has been modified to run on Workstations.
- Has the widest collection of public domain software available.

3.3.4 Personal Computers

- Smaller, single user machines based on microprocessors, with some graphics capabilities.
- Very inexpensive from $500 to $10,000
- Some models are Apple Macintosh, IBM PC, etc
- Have very inexpensive software

- Style range such as desk-top, lap-top, factory hardened.
- Have the widest range of commercial software and hardware available.
- Excellent for real time control, and simple user interfaces.

3.3.5 Dedicated Computers

- A number of computers have been built into machines.
- These computers are dedicated to one operation, such as an NC controller, a PLC, A Plastic Injection Machine Controller, computer printer, etc.
- Very well debugged, and easy to use
- Requires some specialized user training.
- Cost is included in cost of equipment, and has effect of lowering costs over relay controlled systems.
- May have simple interfaces for connections to other computers.

3.3.6 Single Board Computers

- A computer with little or no user interface hardware (often only 1 RS-232 port) is provided.
- The previous standard for control.
- Very inexpensive, from $50 to $1000
- Requires Assembly Language Programming
- May come with I/O for digital, analog, serial, or other communication task.
- Hard to implement and debug

3.4 ARCHITECTURE AND BUSES

• All of the components are tied together by computer buses.

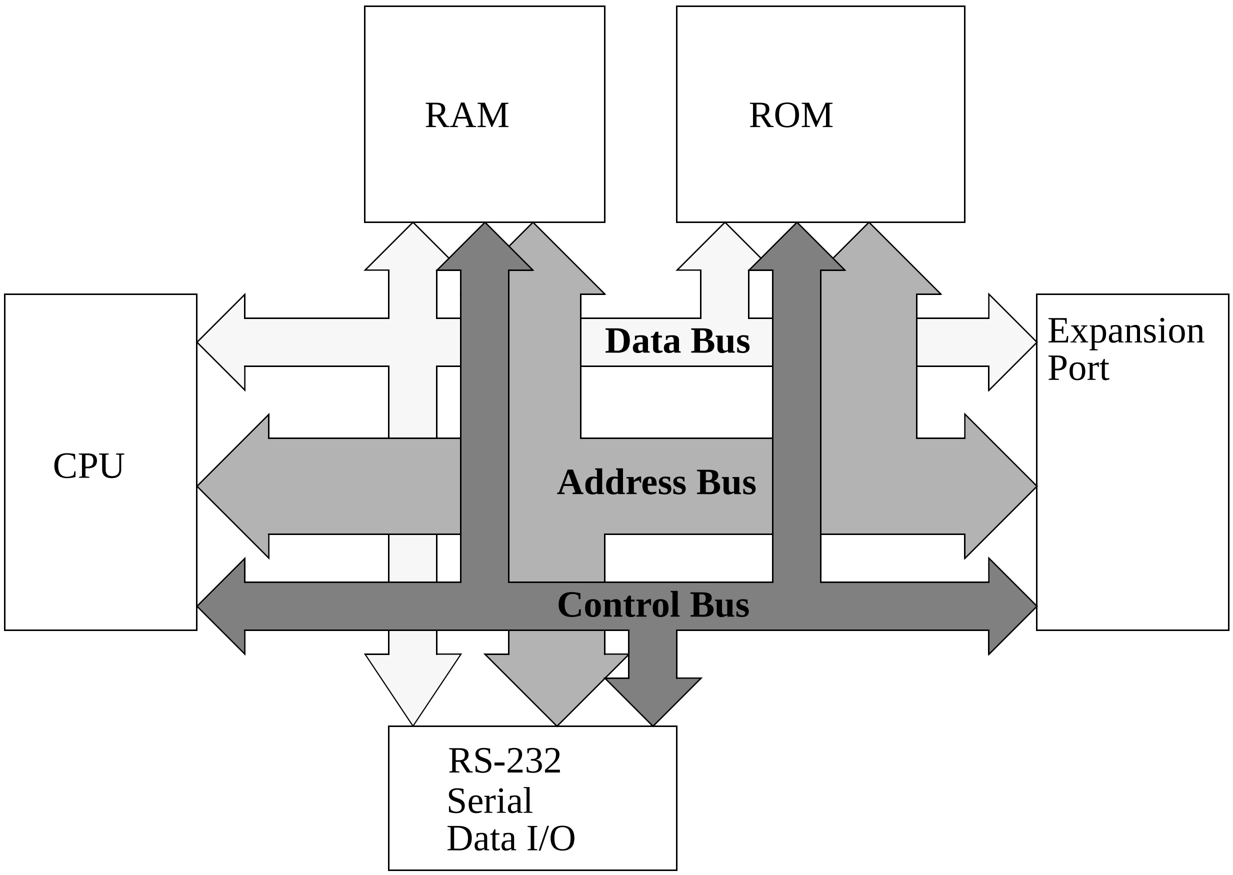
• There are a number of separate buses in a computer,

- • Address Bus,
    - carries locations from the CPU, so that the proper RAM, ROM, and I/O locations may be addressed
    - This is often 16 bits, but newer computers have moved to 32 an 64 bits.
    - The size of this bus limits immediately available memory.

- • Data Bus,
    - Carries data to and from the CPU and RAM, ROM, and I/O devices.
    - The size of this bus makes the difference when specifying whether the CPU is 8 bit, 16 bit, 32 bit, or 64 bit.
    - A wider data bus allows the CPU to run faster, but the hardware is more complex

• Control Bus,
- Runs between all devices, and is used to direct reads, writes, data ready, address ready, interrupts, and a number of other signals.

• The bus structure of a computer may be as below,



3.4.1 Clock Speed and the Buses

• Clock speed determined how often the CPU will process instructions

• Each instruction will use a variable number of clock cycles. For example a get from memory will take longer than a binary or operation.

• Different CPU's may use anywhere from 5 to a fraction of a clock cycle to process an instruction

• The bus speeds are limited by
- the physical layout of the board,
- memory speed
- CPU speed (which is related to clock speed)

- Each instruction for the CPU is made up of a variable number of bytes. These are loaded from memory during execution. In an 8 bit bus these are loaded 1 at a time. To speed this up wider buses are used. For example a 64 bit bus could probably load up to 8 instructions at one. This allows instructions to be stored and processed faster, and overcome the bus speed limitations of computers.

- Other Computers use a trick called caching to speed the computer. This uses a shorter bus to local memory which may be run at higher speeds, thus reducing the slower access to distant memory.

3.5 SOFTWARE

- The software drives the computer to perform some useful function

- Two main types of programs,

    - Interpreted,
        - Program is stored as text (or equivalent), and each instruction is read, and interpreted as the program is running.
        - Can be very slow
        - Easier to write operating systems
        - Suited to simple programming languages
        - BASIC is a popular interpreted language.

    - Compiled,
        - Programs are converted, and stored in machine language before execution.
        - Runs very fast
        - Can allow very advanced debugging tools
        - Programming can be complex
        - Easier to protect copyrights
        - FORTRAN and C are good examples of compiled languages.

- Programs may be divided into a number of various categories,

    - Operating Systems
        - Software which provides fundamental services for the computer, such as running printers, user accounting, etc.
        - Can be very large
        - Required for any computer to operate
        - Acts as a go between for the programs and the hardware
        - Popular Operating systems are UNIX, MS-DOS, VMS, NOS.

• Micro-code
  - This is a program which the engineers write to control the CPU when it is initially designed.
  - This is not seen by a user.
  - Tells CPU how to deal with instructions

• Machine Language
  - Written in the lowest level of language available to the user
  - These programs are very fast and efficient.
  - These programs are difficult to write.

• Graphical User Interface,
  - A program which makes extensive use of graphical abilities to interact with the user.
  - Requires a powerful computer
  - Can be slow, and use up to 90% of CPU time
  - Makes programs very easy to use by novices
  - Requires greatly increased development time for software
  - Good examples of these programs are Lotus 1-2-3, Autocad, Microsoft Windows, WordPerfect, SDRC Ideas, etc.

• Utility (for want of a better word),
  - Simple programs for individual tasks such as file format conversion.
  - small in size, and easy to use

• Scientific, Mathematical, etc.
  - Use large amounts of CPU time for mathematical calculations
  - Very problem specific
  - Requires specialized knowledge
  - Very expensive
  - Requires powerful computer
• Scripts, Batch Files, etc.
  - Often interpreted files which instruct the operating system in task such as installing software, running nightly backups, etc.
  - Written in ASCII with simple English like commands
  - Example of these are ‘autoexec.bat’ files in MS-DOS, and ‘.login’ files in UNIX.

• etc.

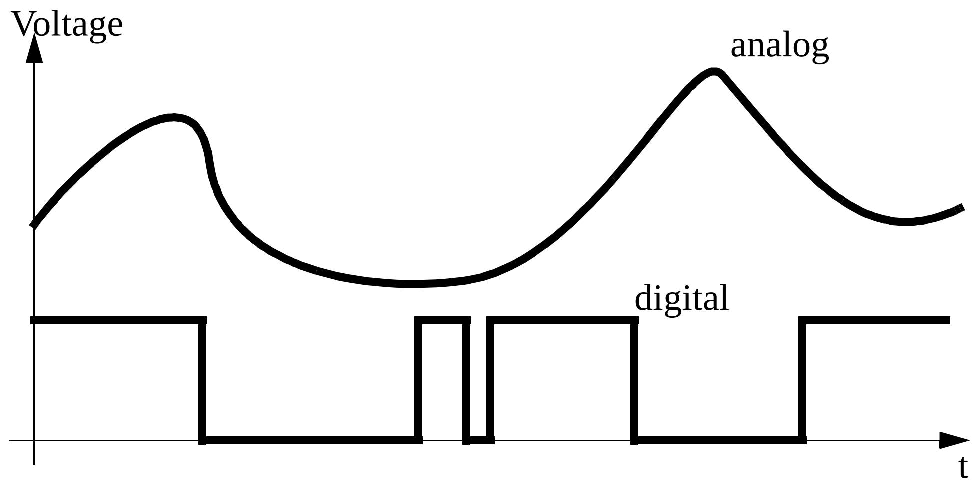
4. COMPUTER INTERFACING

• Interfacing for Acquisition of Signals from Sensors and Generation of Signals for Actuators

• Used by Computers, PLC’s, PID Controllers, etc.

• Computers are designed to handle both input and output (I/O) data

• Two main types of data I/O for computers
  • Analog
  • Digital



4.1 DIGITAL SIGNALS

• read by the computer, and interpreted as ON or OFF.

• Examples of Inputs
  - contact switches
  - proximity switches
  - pulse counters
  - pulse generators
  - optical sensors

• Examples of Outputs,
  - indicator lights
  - solid state relays (e.g. for power to motor)

• Within this signal type there are two categories of signal,
- Discrete binary - such as a switch which will be thrown once
- Pulse data - a continuous stream of pulses. The frequency, period, or duty cycle will control the signal value. A pulse from a tachometer for every revolution will indicate the motor speed.

• The electrical connections for these signals quite often require some additional circuitry. This includes,

latches - used to capture fast changing values (74163?)
schmitt triggers - pull a signal that may not be 0 or 5V to a reasonable logic level. It also uses a dead zone to reduce random switching. (7414?)
flip flops - we can toggle these (7421?)
inverters - gives the signal strength an buffers (7404?)
opto-isolators - eliminates electrical connections for circuit safety. These are basically an LED and phototransistor in one package.
triacs - allows DC signals to switch AC
relays - allows electrical switching that is electrically isolated with mechanical devices.
line drivers - converts low current signals to higher current (7485?)
op-amps - allow high impedance buffering between inputs and outputs. (741)
RC network - a resistor capacitor pair can be used to pass only high/low frequency signals, and filter out noise/DC.
pull-up resistors - can keep inputs/outputs from floating.

• Consider the example below,

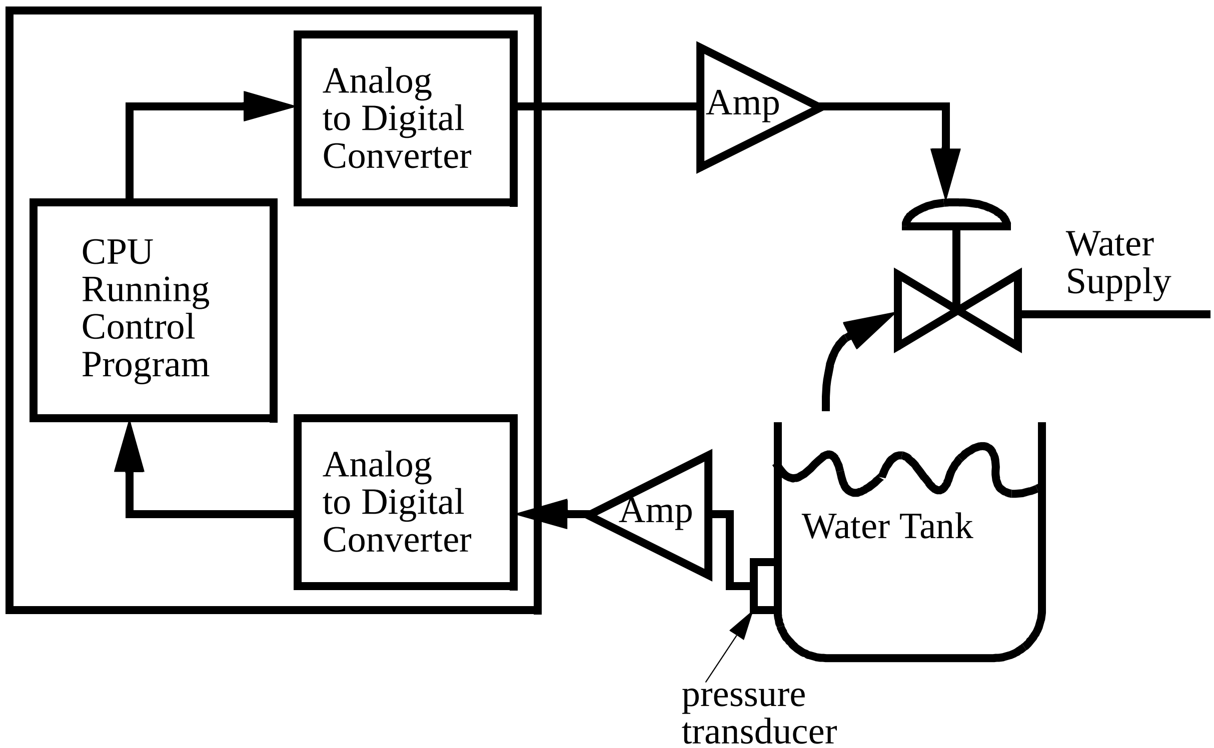
Design some circuitry for a fuel injection computer control unit that is to read pulses from a tachometer mounted on an engine, and then control fuel injection, and spark timing. The setpoint is determined by the gas pedal.

4.2 ANALOG SIGNALS

• A Continuous signal is sampled by the computer

• The computer uses approximation techniques to estimate the analog value during the sampling window.

• An example of an A/D, D/A control of a process is shown below



• Multiplexers are used when a number of signals are to be input to a single A/D converter. This allows each of a number of channels to be sampled, one at a time

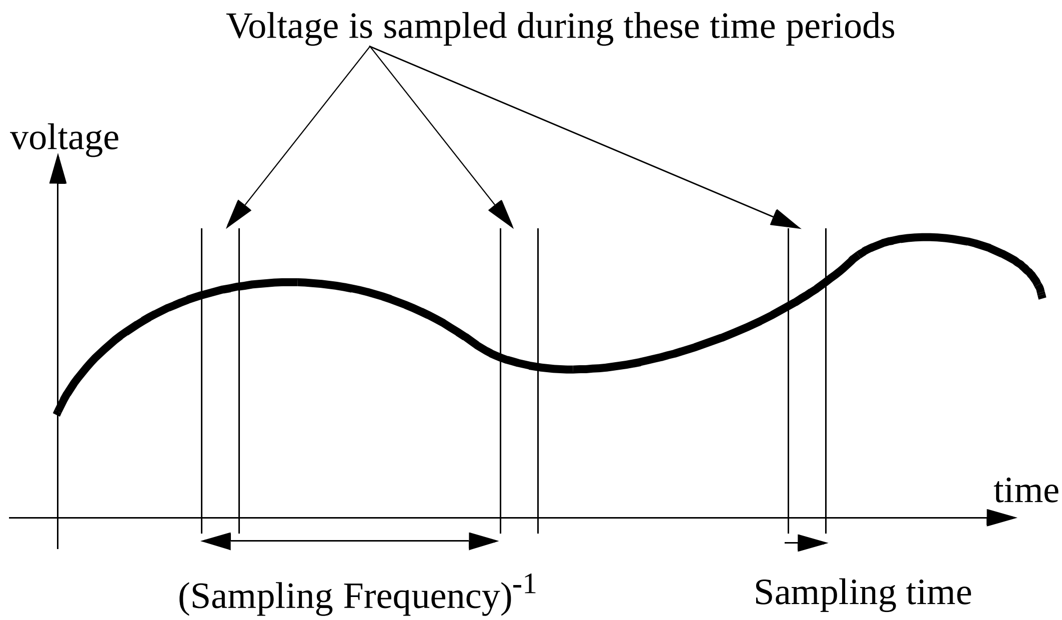
• Signal conditioners are often to amplify, or filter signals coming from transducers, before they are read by the A/D converter.

• Output drivers and amplifiers are often required to drive output devices when using D/A

• Sampling problems occur with A/D conversion. Because readings are taken periodically (not continually), the Nyquist criterion specifies that sampling frequencies should be twice the frequency of the signal being measured, otherwise aliasing will occur.

• Since the sampling window for a signal is short, noise will have added effect on the signal read. For example, a momentary voltage spike might result in a higher than normal reading.

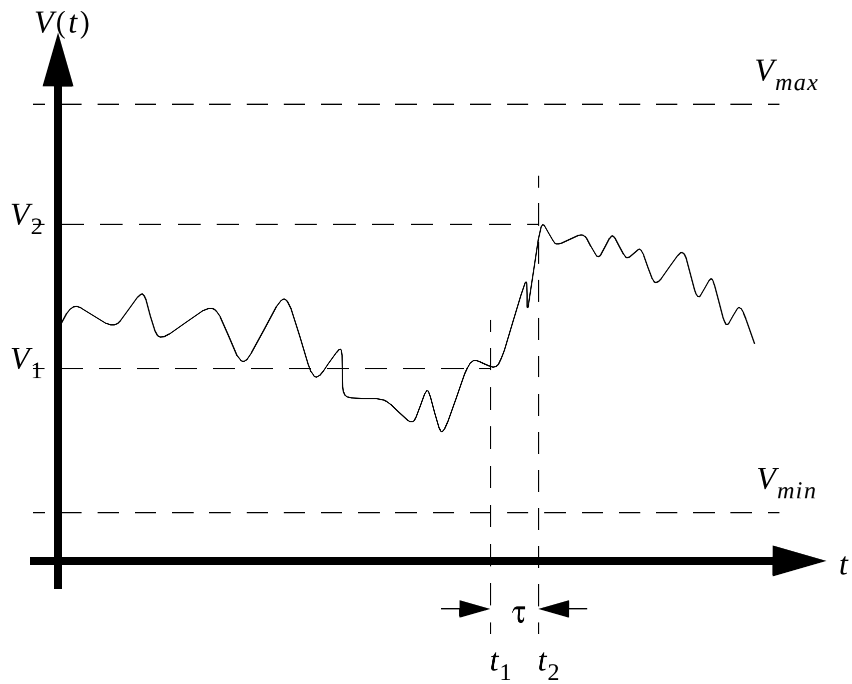
• When an analog value is converted to or from digital values, a quantization error is involved. The digital numbering scheme means that for an 8 bit A/D converter, there is a resolution of 256 values between maximum and minimum. This means that there is a round off error of approximately 0.4%.



4.2.1 Analog to Digital Conversion

• When there are analog values outside a computer, and we plan to read these to digital values, there are a variety of factors to consider,
  - when the sample is requested, a short period of time passes before the final sample value is obtained.
  - the sample value is ‘frozen’ after a sample interval.
  - after the sample is taken, the system may change
  - sample values can be very sensitive to noise
  - the continuous values of the signal loose some accuracy when conversion to a digital number

• Consider the conversion process pictured below,



where,

$V(t)$ = the actual voltage over time

τ = sample interval for A/D converter

t = time

t_1, t_2 = time at start,end of sample

V_1, V_2 = voltage at start, end of sample

V_{min}, V_{max} = input voltage range of A/D converter

N = number of bits in the A/D converter

• Once this signal is processes through a typical A/D converter we get the following relations (these may vary slightly for different types of A/D converters).

$$R = 2^N$$

$$V_D = INT\left[\left(\frac{V_2 - V_{min}}{V_{max} - V_{min}}\right)R - 0.5\right]$$

$$V_{ERROR} = \left(\frac{V_{max} - V_{min}}{2R}\right)$$

where,

R = resolution of A/D converter

V_D = the integer value of the voltage

V_{ERROR} = the maximum quantization error

• In most applications a sample is taken at regular intervals, with a period of ‘T’ seconds.

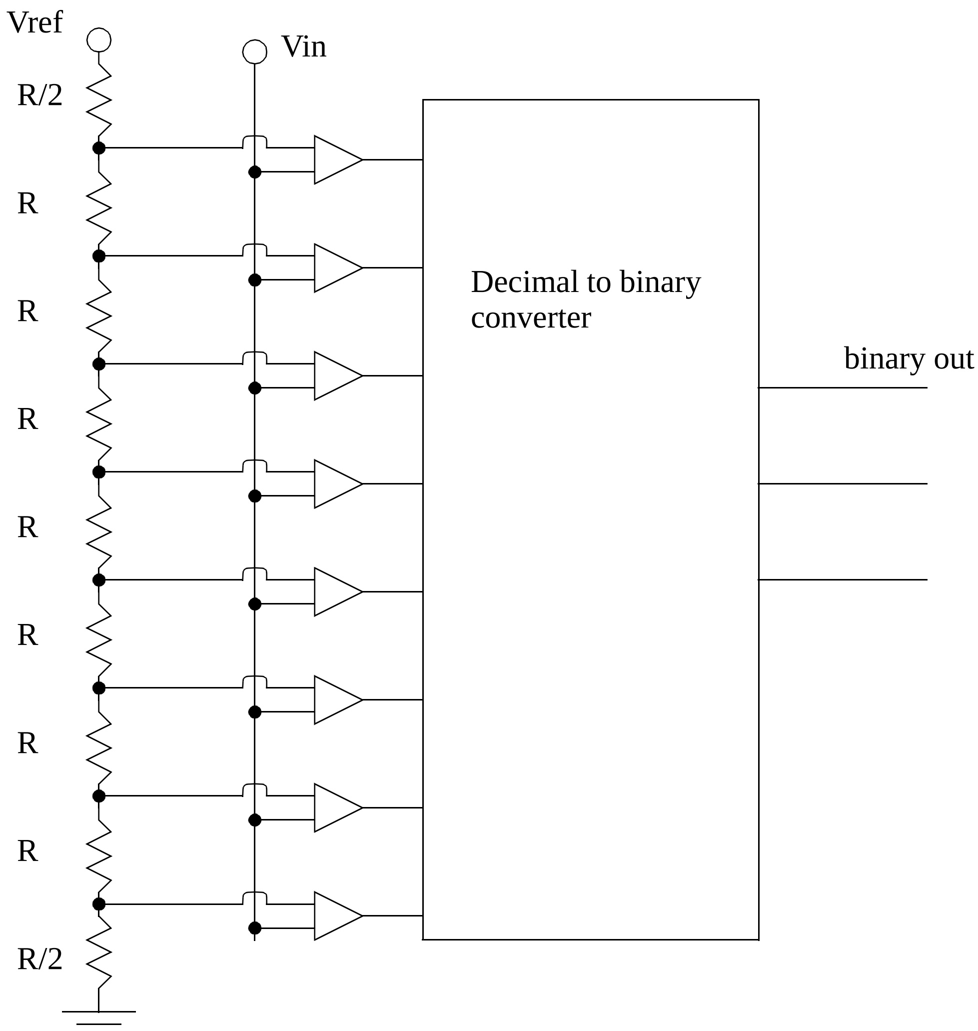
• In practice the sample interval is kept as small as possible. (i.e., tau << T)

• If we are sampling a periodic signal that changes near or faster that the sampling rate, there is a chance that we will get a signal that appears chaotic, or seems to be a lower frequency. This phenomenon is known as aliasing.

• Quite often an A/D converter will multiplex between various inputs. As it switches the voltage will be sampled by a ‘sample and hold circuit’. This will then be converted to a digital value. The sample and hold circuits can be used before the multiplexer to collect data values at the same instant in time.

4.2.1.1 - Flash A/D Converter

• On type of A/D converter is the flash converter shown below,



• These converters are very fast, but they are hard to build for high resolutions.

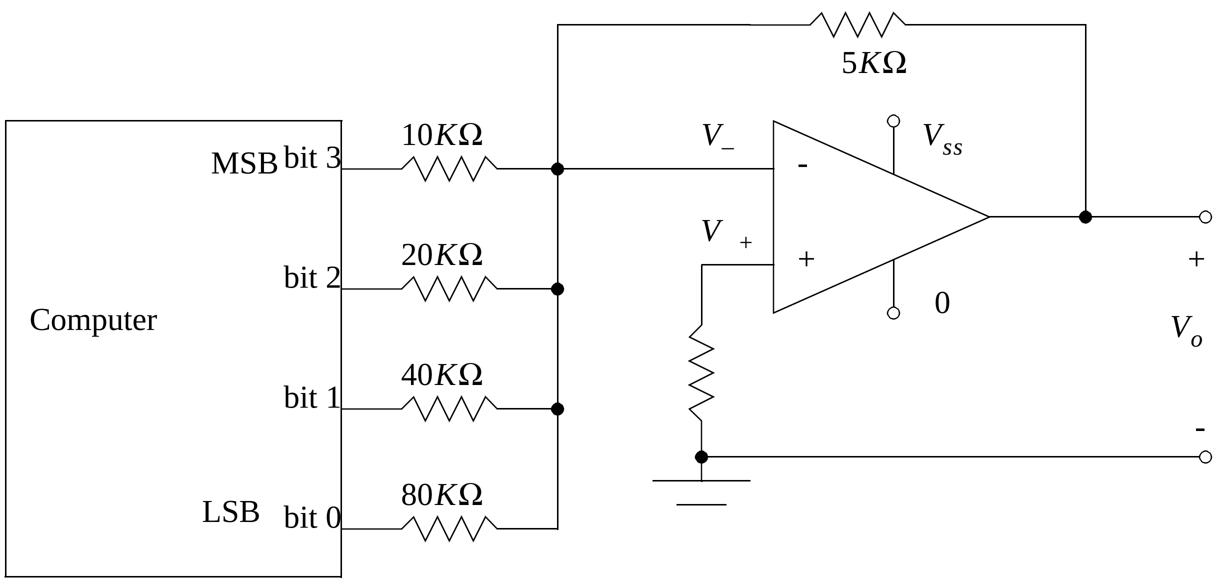
• The conversion rates for these devices are limited by inherent capacitance, and transistor switching times.

4.2.2 Digital to Analog Conversion

• After we have used a controller equation to estimate a value to put into our process, we must convert this from a digital value in the computers memory, to a physical voltage.

• This voltage is typically limited to 20mA in most computer board, and drawing near this current reduces accuracy and life of the board.

• A simple circuit is shown below for a simple digital to analog converter.



First we write the obvious,

$$V_+ = 0 = V_-$$

Next, sum the currents into the inverting input as a function of the output voltage and the input voltages from the computer,

$$\frac{V_{b_3}}{10K\Omega} + \frac{V_{b_2}}{20K\Omega} + \frac{V_{b_1}}{40K\Omega} + \frac{V_{b_0}}{80K\Omega} = \frac{V_o}{5K\Omega}$$

$$\therefore V_o = 0.5V_{b_3} + 0.25V_{b_2} + 0.125V_{b_1} + 0.0625V_{b_0}$$

Consider an example where the binary output is 1110, with 5V for on,

$$\therefore V_o = 0.5(5V) + 0.25(5V) + 0.125(5V) + 0.625(0V) = 4.375V$$

• The calculations for the A/D converter resolution and accuracy still apply.

• Consider the example below,

We need to select a digital to analog converter for an application. The output will vary from -5V to 10V DC, and we need to be able to specify the voltage to within 50mV. What resolution will be required? How many bits will this D/A converter need? What will the accuracy be?

4.3 TIMING

• Timing is significant in computer controlled processes. If we think of a simple example, a fixed sample period means that the computer program must arrange to read the analog value at a consistent time.

• Various schemes for timing include,
    - interrupts - fairly efficient, but requires some timing hardware inside
        • by computer clock - a regular interrupt for the operating system
        • programmable timer - a timer that is told to call an interrupt after some time
        • by external trigger - asynchronous timing, potentially unstable
        • by external clock - an external synchronous clock
    - delay loops - consumes a large number of clock cycles
        • wait for clock value - reasonable time, but prone to software problems
        • count cycles - quite stable, but the computer cannot do anything else
    - whenever possible - this ‘non-real-time’ approach can become unstable
    - hardware

- trigger and scan - a special input board can be set to read one or more values when timing or triggering conditions met

- Values are simply read from/written to locations in memory for both analog and digital values. When the process takes more than a single CPU cycle there will be a flag set to indicate when the operation is done. One example is reading analog values. This process may take hundreds of CPU cycles. We have three options,
  1. continually check the ‘done’ flag/bit in memory
  2. have the A/D converter trigger an interrupt when done
  3. have the program perform other tasks for a period of time longer than the longest scan time.

- There are three fundamental properties of interrupts used for timing,
  - Hardware/Software
    - Hardware - These are actual pins on the outside of the CPU. When a change occurs the CPU will save its current status, and go to a subroutine. When done the system status is restored, and the computer programs resumes where it was interrupted. The number of hardware interrupts is often severely limited, but extra hardware can be added to expand the hardware interrupts. A computer keyboard and mouse typically use hardware interrupts.
    - Software - These are similar to hardware interrupts, except they are started by a call in software. These are plentiful, and used in all aspects of modern operating systems. MS-Dos makes extensive use of these software interrupts.
  - Disabling
    - Maskable - These interrupts can be turned off by software. When running a critical process we may not want an interrupt to occur (and stop our program temporarily).
    - Non-Maskable - these interrupts cannot be disabled and will always cause the processor to stop.
  - Priority
    - We can set priorities for interrupts. If multiple interrupts occur at the same time the highest priority interrupt will be executed first. This allows critical processes to be addressed first.

4.3.1 Interrupts

4.3.2 Clocks and Timers

4.3.3 Watch Dog Timers

4.3.4 Polling

4.3.5 DMA

4.4 DISPLAYS

5. COMPUTER INTERFACE BOARDS

5.1 OVERVIEW

5.1.1 Types

5.2 REGISTER LEVEL PROGRAMMING

5.3 EXAMPLES

6. COMPUTER CONTROL OF PROCESSES

• Functions of a Process I/O Interface,

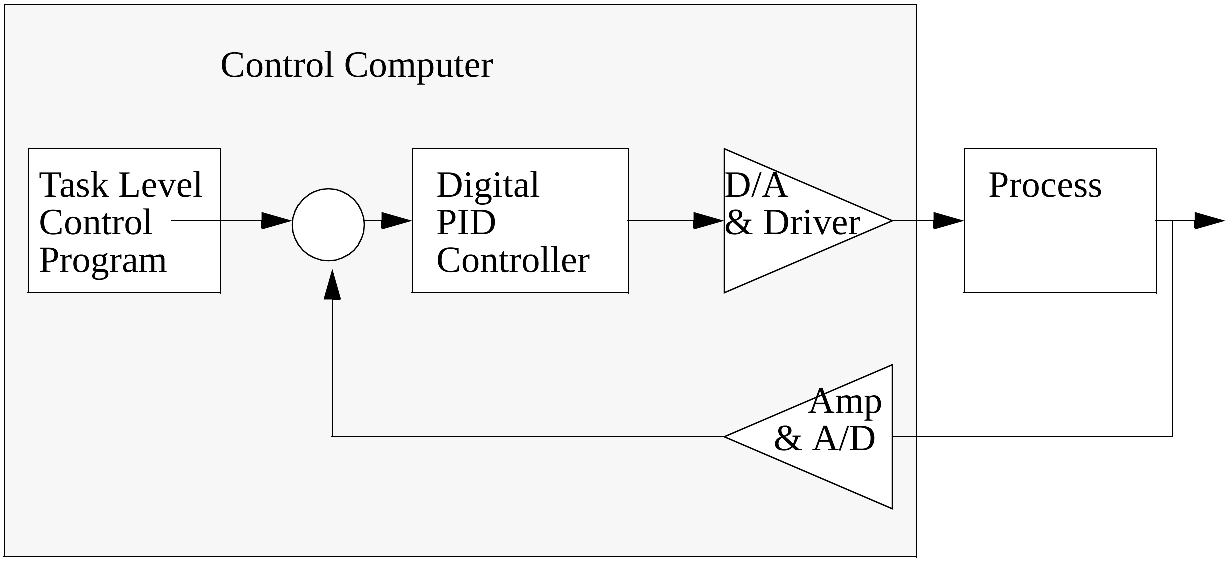
- Control the scanning and processing of all the inputs and outputs between the process and the computer.
- Provide conversions of signal levels, compensation for sensor range, and linearization of signals.
- Provide signal and sensor validity checks.
- Provide signal pre-processing and filtering.
- Provide measurements to the various other programming packages which have been converted to engineering units or interpreted in other ways.
- Provide alarm limit checks input-by-input.

• Some features found in computer control systems,
- timer initiated events - such as clock based sampling periods for a control loop
- Process initiated interrupts - Prioritized interrupt signals can be used to alert computers to standard notifications, or to initiate an emergency sequence when a critical failure has occurred.
- Control software - directs the operating points for the control system
- System and program initiated events - these are for system house keeping, and for communication to other computer systems in a network.
- Operator initiated events - requests for reports, changing of operating parameters, checking quality, etc.

• Interrupts
  • Each computer has the ability to process interrupts.
  • These interrupts can stop the microprocessor, and make it execute other subroutines.
  • If no interrupts are received, the computer will always run the same program
  • When interrupts are received, they are processed as urgent, or non-urgent.
  • When an interrupt is processed, a small subroutine is run, and then execution returns to the original point in the interrupted program.
  • If interrupts are not used, then the computer must continuously scan all of its inputs to see if any are critical. But this is not practical for real time processing.

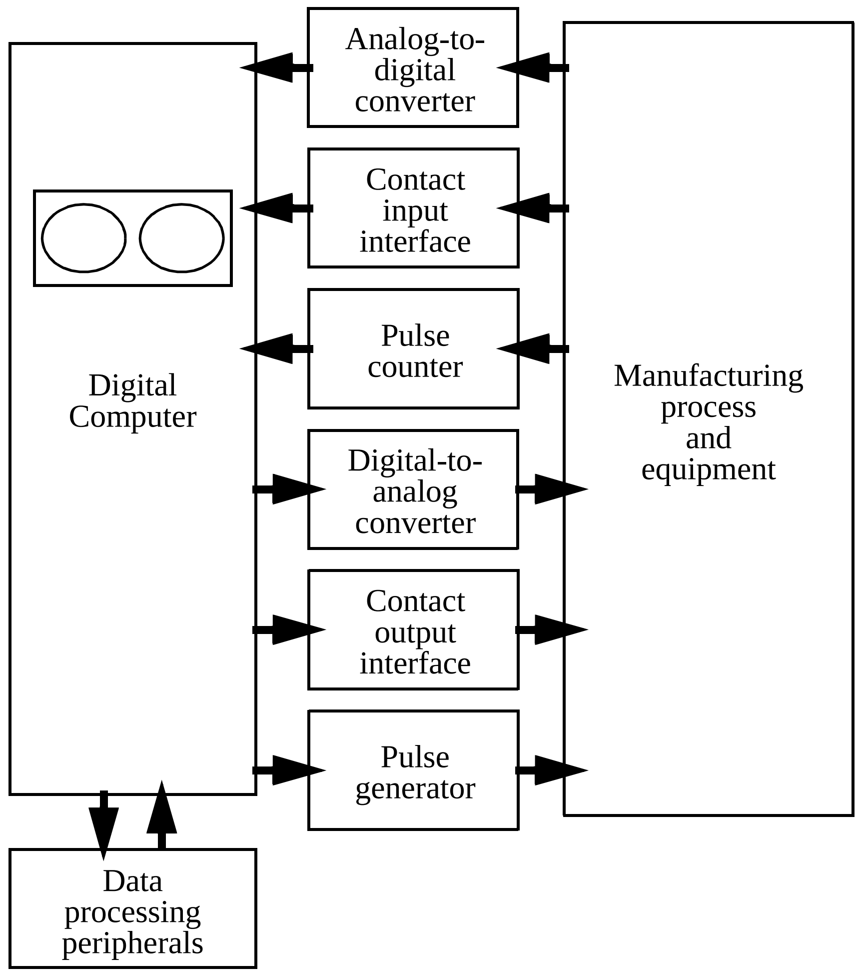
• Linear Computer Controllers,
  • Linear control theory can be converted into computer programs.
  • Canonical form (Laplace/frequency domain) control rules can be converted into a form suitable for computers by using ‘z-transforms’
  • State space control laws can be converted into computer code using matrix techniques, along with numerical integration.
  • For example the PID control law has been used in many computer control programs to date.

• The figure below depicts a compute controlled process



• Advanced topics include,
- Feed forward controllers
- Adaptive controllers
- Optimal controllers
- Non-linear control systems
- Multi-input/Multi-output control systems

• An example of Computer-process interface



6.1 TEMPERTURE CONTROL

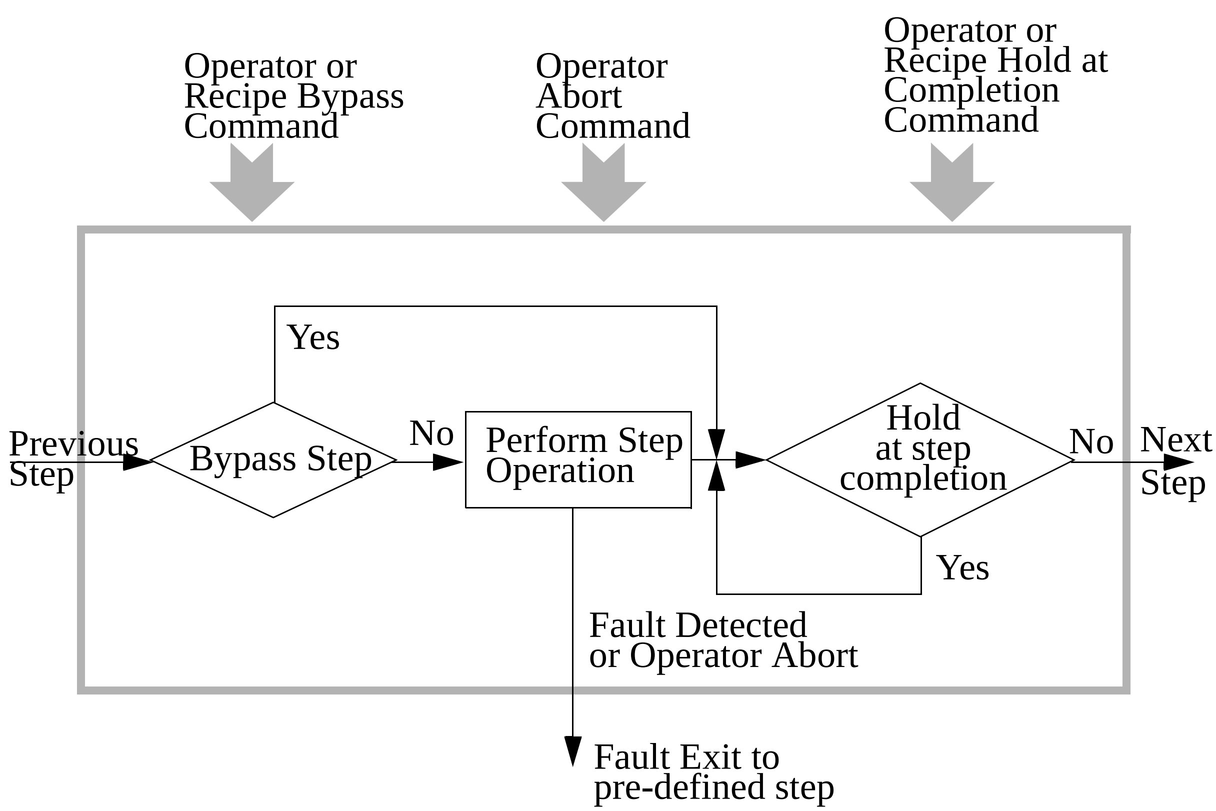
6.2 BATCH PROCESSING

• The nature of Batch processes,
- Batch processes deal with discrete quantities of raw materials or products.
- batch processes allow the tracking of these discrete quantities of materials or products
- batch processes allow more than one type of product to be processed simultaneously, as long as the products are separated by the equipment layout.

- Batch processes entail movement of discrete product from processing area to processing area
- Batch processes have recipes (or processing instructions) associated with each load of raw material to be processed into product.
- Batch processes have more complex logic associated with processing than is found in continuous processes
- Batch processes often include normal steps that can fail, and thus also include special steps to be taken in the event of a failure.

• The nature of steps in a batch process,
  - Each step can be simple or complex in nature, consisting of one or more operations
  - Generally, once a step is started it must be completed to be successful.
  - It is not uncommon to require some operator approval before leaving one step and starting the next.
  - There is frequently provision for non-normal exits to be taken because of operator intervention, equipment failure or the detection of hazardous conditions.
  - Depending on the recipe for the product being processed, a step may be bypassed for some products.
  - The processing operations for each step are generally under recipe control, but may be modified by operator override action.

• A typical process step



• Functions of all Process Control Systems

- There must be an I/O interface between the processes and the control. The I/O interface passes all the information moving between the two parts of the total system.
- The control system must provide continuous control functions to regulate appropriate portions of the process.
- The control system must provide sequential control functions. A properly designed batch control system should make it easy to describe the sequence of operations and the checks that must take place.
- The control system must provide displays and interfaces which the operator can use to monitor and direct process activity.

• Additional functions found only in Digital control systems
- An advanced control system should allow the process to handle a range of products, and not just a single product. The use of recipes is a concise and convenient method to describe the process steps for each product.
- An advanced control system should provide a multiprogramming environment in which each specific task can be programmed in a simple stand-alone fashion. The system should also allow more than one unit in the plant to be controlled at the same time.
- An advanced control system should provide displays which are oriented toward the total process rather than toward individual parameters.
- An advanced control system should automatically detect and correct process upsets and equipment failures.
- An advanced control system should provide for device operations which reduce the complexity of the logic that the user must deal with. There are many mundane and repeated operations and error checking functions which must be done in manipulating valves, pumps, motors, fans, and so on which can best be handled in device packages which standardize such operations.

6.3 COMPUTER PROCESS MONITORING

• Process Data,
  • can ensure dangerous situations are avoided
  • variables such as temperature, pressures, forces, concentrations of chemicals, etc.
  • Can help track process progress, and allow corrections

• Equipment Data,
  • Data related to equipment
  • Can avoid dangerous and costly operating conditions

• Product Data,
  • Allows on line monitoring of process output,
  • variables can be, quality, yield, production rate, etc.

6.4 IMPLEMENTING A COMPUTER CONTROL SYSTEM

• The steps below are typically essential for replacing an existing/new control system with a computer control system:
1. do an analysis of the existing/similar system.
2. clearly list control objectives
3. list outputs required to meet objectives
4. list inputs required to meet objectives
5. determine logical relationships between the inputs and outputs
6. write programs for relationships found in last step
7. document all aspects of the system

6.4.1 SCADA

• SCADA (System Control And Data Acquisition) systems are based around a central computer.

• The central computer communicates with remote computers/PLCs/sensors to issue control commands and collect data.

6.5 PRACTICE PROBLEMS

1. A video voltage signal is to be digitized by an A/D converter. The maximum voltage range is 15V. The A/D converter has a 6-bit capacity. Determine the number of quantizing levels the quantization level spacing, and the quantization error.

7. COMPUTER COMMUNICATIONS

• We need to be able to connect computers to transfer information.

7.1 COMPUTER COMMUNICATIONS CATEGORIES

• Basic Categories
- • Interface to Sensors and Actuators
    - - Uses readings from sensors to control devices in the environment
    - - very simple protocols, but often requires specialized hardware and software.
- • Interface to computers
    - - uses machine to machine connection for direct communication
    - - can appear to be like a local area network
    - - low speeds (eg. 300 baud, 1.2, 2.4 9.2 Kbaud)
- • Local Area Computer Networks
    - - measured in distances from meters to kilometers
    - - speed measured in Megabits/sec
    - - medium includes, wire, fibre-optics, radio frequency
    - - commonly available in all forms
    - - inexpensive
- • Wide Area Computer Networks
    - - from city wide, to global
    - - speed measured in Kilobytes/second
    - - medium includes, wire, fibre optics, satellite, microwave
    - - access is costly (some mix of initial and variable costs), and geared to large corporations, or special applications.

• Classification of Interconnected processors (by Physical Size)

Table 1:

<table>
<tr><th>Inter-processor Distance</th><th>Processors Located in Same</th><th>Example</th></tr>
<tr><td>0.1 m</td><td>Circuit Board</td><td>Data flow machine</td></tr>
<tr><td>1 m</td><td>System</td><td>Multi-processor</td></tr>
<tr><td>10 m</td><td>Room</td><td rowspan="3">Local network</td></tr>
<tr><td>100 m</td><td>Building</td></tr>
<tr><td>1 km</td><td>Campus</td></tr>
</table>

Table 1:

Inter-processor Distance	Processors Located in Same	Example
10 km	City	Long haul network
100 km	Country	
1000 km	Continent	Inter-connection of long haul networks
10000 km	Planet	

7.2 THE HISTORY

• Late 50's
  - Bell introduces data communication networks
  - Texaco uses remote monitoring and control for automating polymerization plant.
• Early 60's
  - World wide use of data phone system to link airports for maintenance
  - Many uses of terminals for inputting and displaying data.
• Mid 60's
  - Uses of computer for large scale test monitoring
  - NC part programming being done remotely on time sharing mainframes.
• 70's
  - Advent of Mini computers made it possible for companies to have multiple computers which had to communicate.
• 80's
  - Microprocessor drops cost of computers dramatically, and automated machines become very common
  - Personal computers, and automated machinery start a trend to network small computers networked with large computers
  -

7.3 SERIAL COMMUNICATIONS

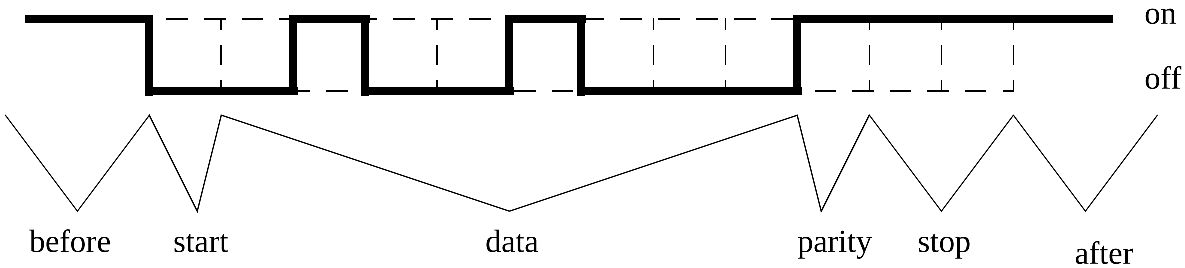
• Serial communication channels send data one bit at a time.

• These techniques are often slower, but are essential when transmitting using radio waves, telephone lines, or similar media.

• Serial lines have become one of the most common methods for transmitting data to instruments, and most personal computers now have two serial ports.

• Serial data can be transmitted two ways,
  synchronous - all of the data bits are timed between sender and receiver.
  asynchronous - the data bits arrive unannounced, without any coordination between sender and receiver. (this is the most common)
    RS-232/RS-232C - the most common standard using voltage levels
    RS-422 - a current loop standard well suited to noisy conditions
    RS-485 -

• A typical data byte looks like the one below. The voltage/current on the line is turned on/off. The width of the bits determines the possible bits per second (bps). The value shown before is used to transmit a single byte.

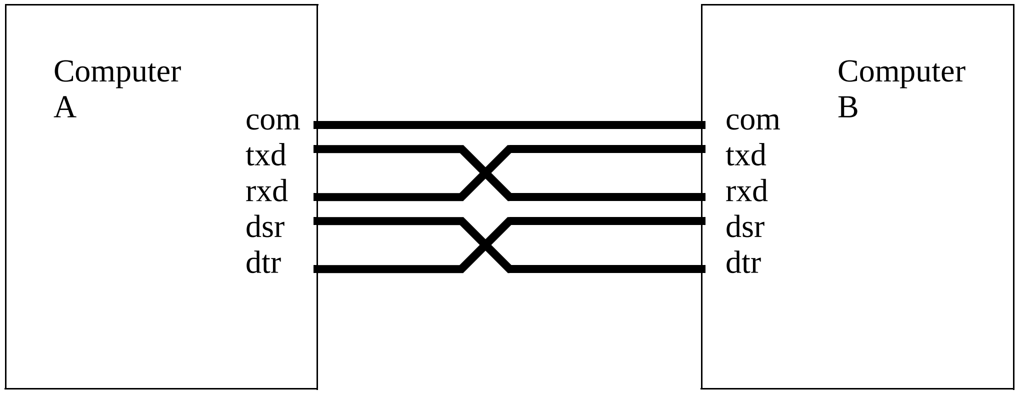


• The bits shown correspond to,
  before - this is a period where no bit is being sent and the line is high
  start - a single bit to help get the systems synchronized
  data - this byte could be 7 or 8 bits, but is almost always 8 now. The value shown here is 01001000 binary.
  parity - this lets us check to see if the byte was sent properly. The most common choices here are no parity bit (just drop it out), or even parity, or odd parity. In this case there are two bits set. If we are using even parity the bit would be set. If we are using odd parity the bit would be off
  stop - the stop bits allow form a natural pause at the end of the data.
  after - a period of time where the line is high before the next byte.

7.3.1 RS-232

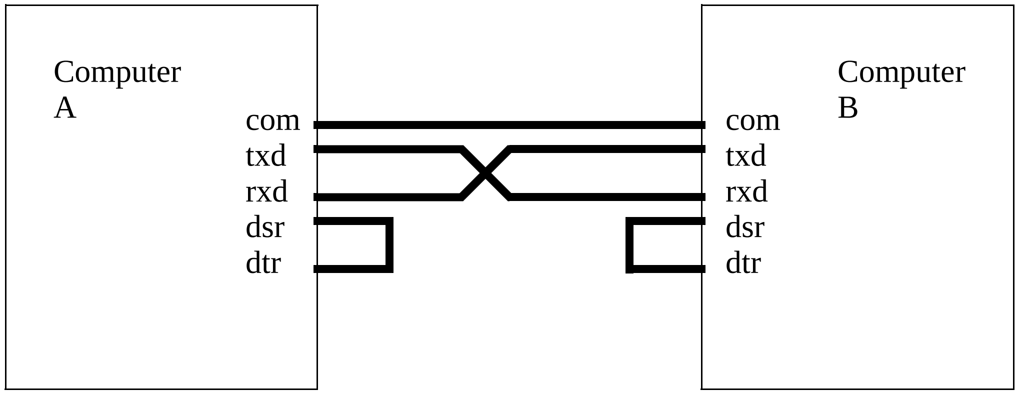
• This standard is based on a high being +3 to +15V, and an off being -3 to -15V (+/-12V is commonly used).

• The typical connection scheme is shown below. notice that the lines are crossed between connectors. Also note that in some cases the dsr/dtr lines are not used.

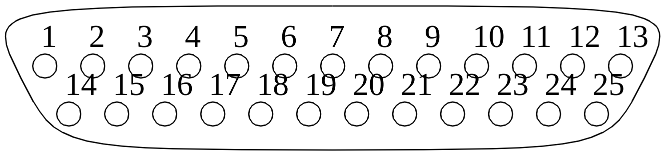


- The line names are,
    - TXD/RXD - (transmit data, receive data) these lines pass data one way from sender to receiver.
    - DSR/DTR - (data set ready, data terminal ready) these handshaking lines indicate when the remote machine is ready to receive data.
    - COM - a common ground to keep things from blowing up, etc.

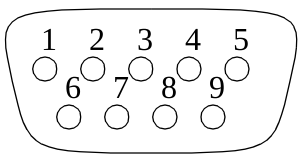
- A simpler wire is also used for some devices (in low noise situations) called a null modem cable. In this cable only the data and common lines are used.



- There are typically two connectors used for these devices. (Note: the connector have very fine numbers on them)



DB-25



DB-9

• The main pin assignments for the DB-25 are,

2 - TXD
3 - RXD
6 - DSR
7 - COM
20 - DTR

• The main pin assignments for the DB-25 are,

2 - TXD
3 - RXD
- COM
- DSR
- DTR

7.4 RS-422

7.5 RS-485

7.6 GPIB/IEEE-488

7.7 PARALLEL COMMUNICATIONS

- In parallel data schemes, all of the data bits are sent at the same time.

- This generally allows a higher rate of data transfer, but only shorter distances, and with a higher number of conductors in the cable.

8. INTERFACING COMPUTERS FOR DATA TRANSFER

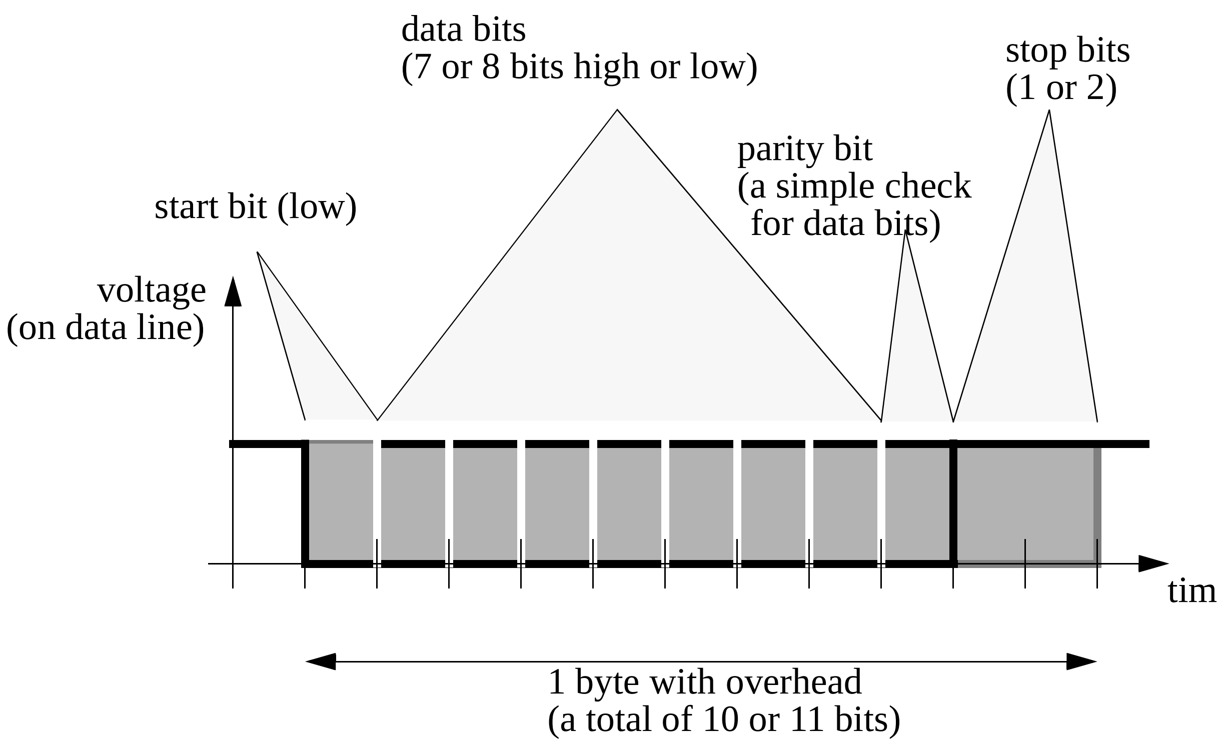
8.1 SERIAL DATA TRANSFER

• A single communication line links two computers, and allows communications one bit at a time.

• Typically strings are passed to and from a terminal, modem, mouse, etc.

• Each string is broken up and each byte is sent one at a time.

• Each byte is sent one bit at a time with the various framing options,
  - 1 or 2 stop bits
  - Even, Odd, or No parity bit
  - 7 or 8 data bits
  - 1 start bit



• Advantages,
  - Very inexpensive
  - Easy to hook up
  - A popular communication port found on almost all computer hardware, regardless of age.

- Can be robust in an industrial environment
- Standards are very clearly defined
- can use common phone lines

• Disadvantages,
  - Very slow

• Characteristics,
  • Speeds of 110, 150, 300, 600, 1200, 2400, 4800, 9600, 12000, 19200 baud (bits per second)
  • Can use as few as three wires for connection, or more as advanced handshaking lines are required.
  • Specialized chips, and modems are commercially available for using this interface

• Popular standards are RS (Recommended Standard)-XXX published by the Electronics Industries Association
  - RS-232 - available on almost all computers
  - RS-422 - a popular industrial standard which can be noise resistant then RS-232
  - RS-423
  - RS-449

8.2 PARALLEL DATA TRANSFER

• Instead of sending the bits in a byte one at a time, parallel buses send the bits in parallel, so that the entire byte arrives at once.

• Basic parallel interfaces connect only two computers (see next section for other case)

• Advantages,
  - Can be very fast, and reliable
  - Easy to create computer hardware to support
  - Chips exist for easy implementation of this scheme
  - The parallel port may be used for alternate form of digital I/O

• Disadvantages,
  - Cabling can be more expensive
  - Standards are not as wide spread as serial communications
  - parallel ports are not universally available on computers and peripherals

• These interfaces have been popular for,
  - printer, and disk interfaces because of their higher speeds, and low costs
  - as a basic digital Input/Output source to drive indicator lights, keyboards, displays, etc.

8.2.1 GPIB Bus (IEEE-488)

- A Parallel bus that has been enhanced to support a number of computers connected by the same cable.

- A Brief History,
    - In the early 70's there was a movement towards standard serial interfaces, but no clear development of a parallel interface standard. As a result Hewlett Packard (HP) set out to develop the GPIB (General Purpose Interface Bus).
    - The HP standard was accepted by both the IEEE and ANSI as standards in 1975.
    - By the early 80's the standard was available in small personal computers (e.g. Commodore Pet Computers).
    - Today many products, and chips are available for development and use of the standard.

- Advantages,
    - Low costs
    - Widely available for test instruments
    - Maximum speeds between 500 Khz and 1 MHz
    - Can replace up to 16 individual serial interfaces with a single interface on the main computer

- Disadvantages,
    - Not necessarily real time,
    - Can be difficult for beginners to learn the bus architecture, but users are often isolated from this.
    - This is often used as a high performance interface on specialized equipment, but is not available on commercial applications anymore.

- Some details,
    - Each device on a GPIB bus has its own address number.
    - A talker-listener protocol is used to resolve bus usage
    - The devices on the bus can be instructed to identify themselves.

9. COMPUTER NETWORKING

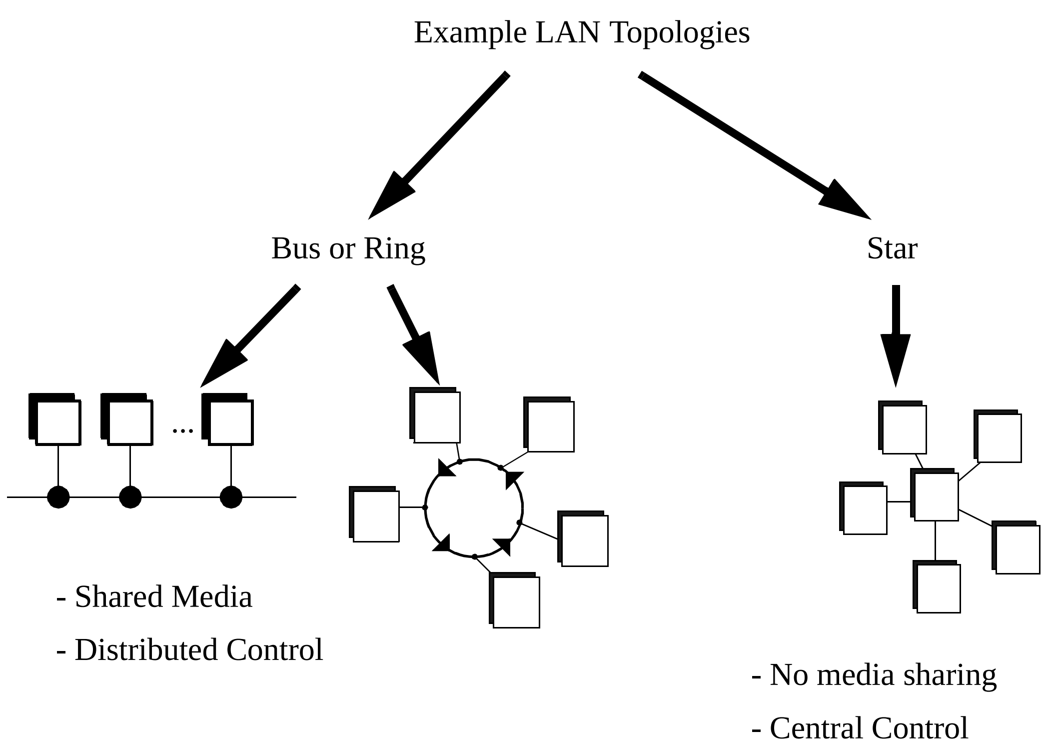
• A Network is a way of connecting different computers to allow data and message passing, and resource sharing.

• Early communication techniques included telegraphs (with morris code), telephones and tele-types.

• Advantages
  - allows shared computers and peripherals
  - provides communication and cooperation tools for people and computers
  - increases reliability by having alternate computers available
  - decreases isolation of many computers

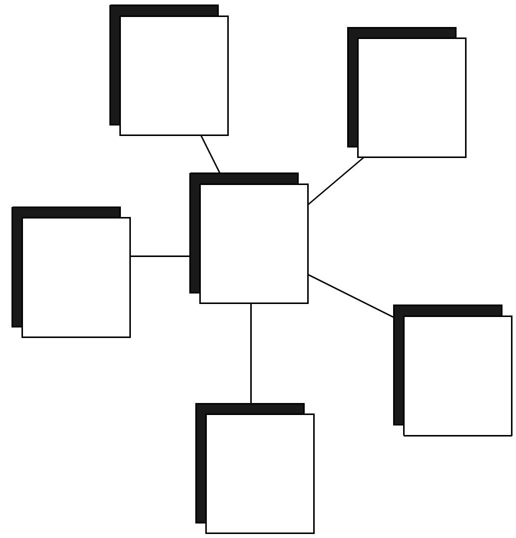
• Applications,
  - allows remote access to databases and libraries
  - electronic mail
  - distributed computational facilities
  - remote control of applications

• The relationship between various network types is shown below

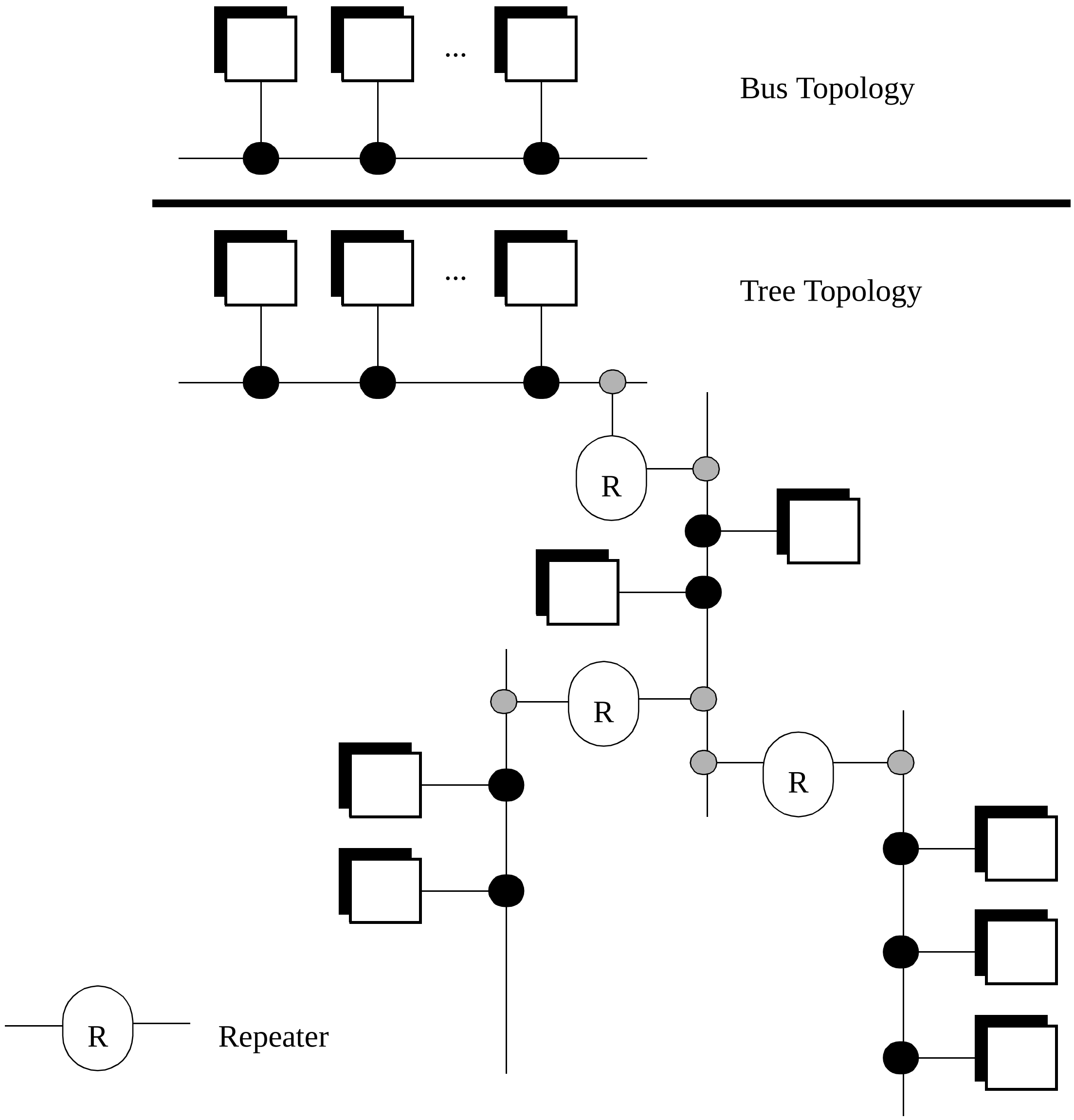


• The Star Network topology uses many connections to a central server

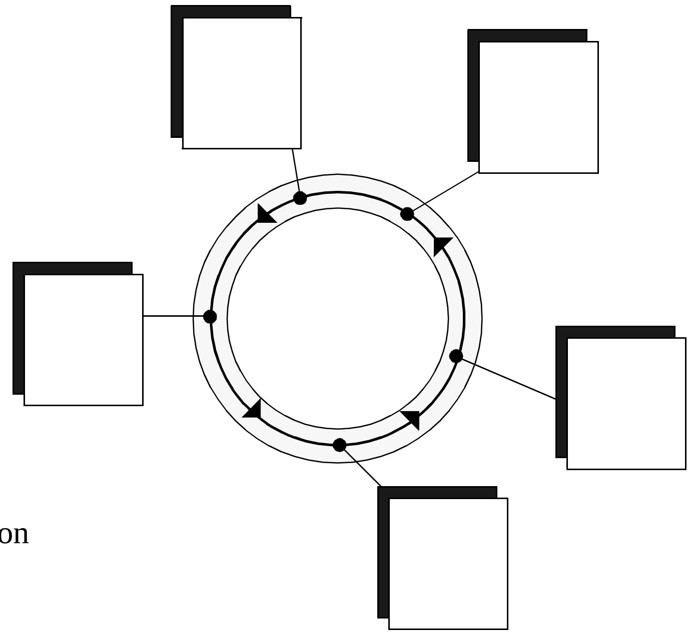
- Central control
- Dedicated communication channels
- Single point of failure
- Often compatible with telephone wiring (voice/data integration)
- Typical minicomputer and mainframe environments



• When all computers are connected to the same network cable, there are a number of structures which may be used. The bus topology below is simpler, and the tree topology is more flexible and versatile. This approach uses a single cable which is easy to add to



• The ring topology below shows a closed loop of cable which can be simple to implement, but limited in capabilities

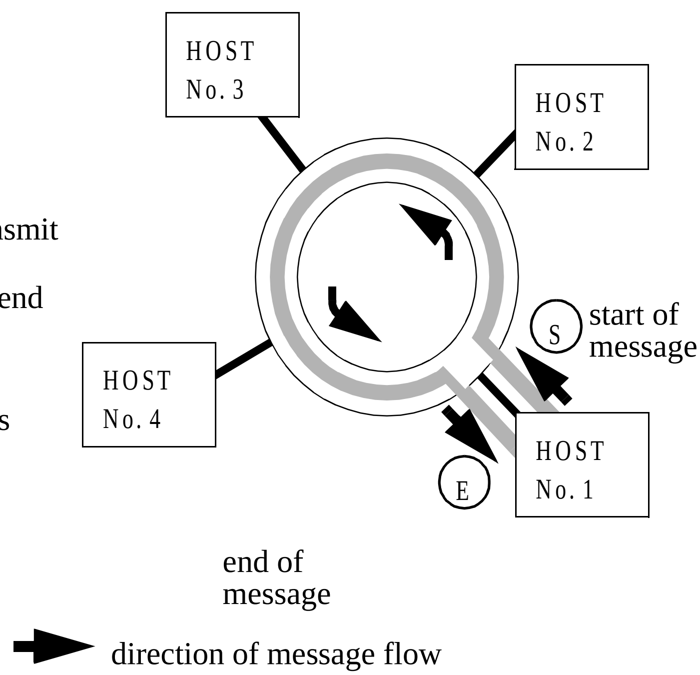


- Distributed control
- Entire communication capacity capacity for any station
- Vulnerable to loss of shared communication facilities
- Primary vendors: IBM & Proteon

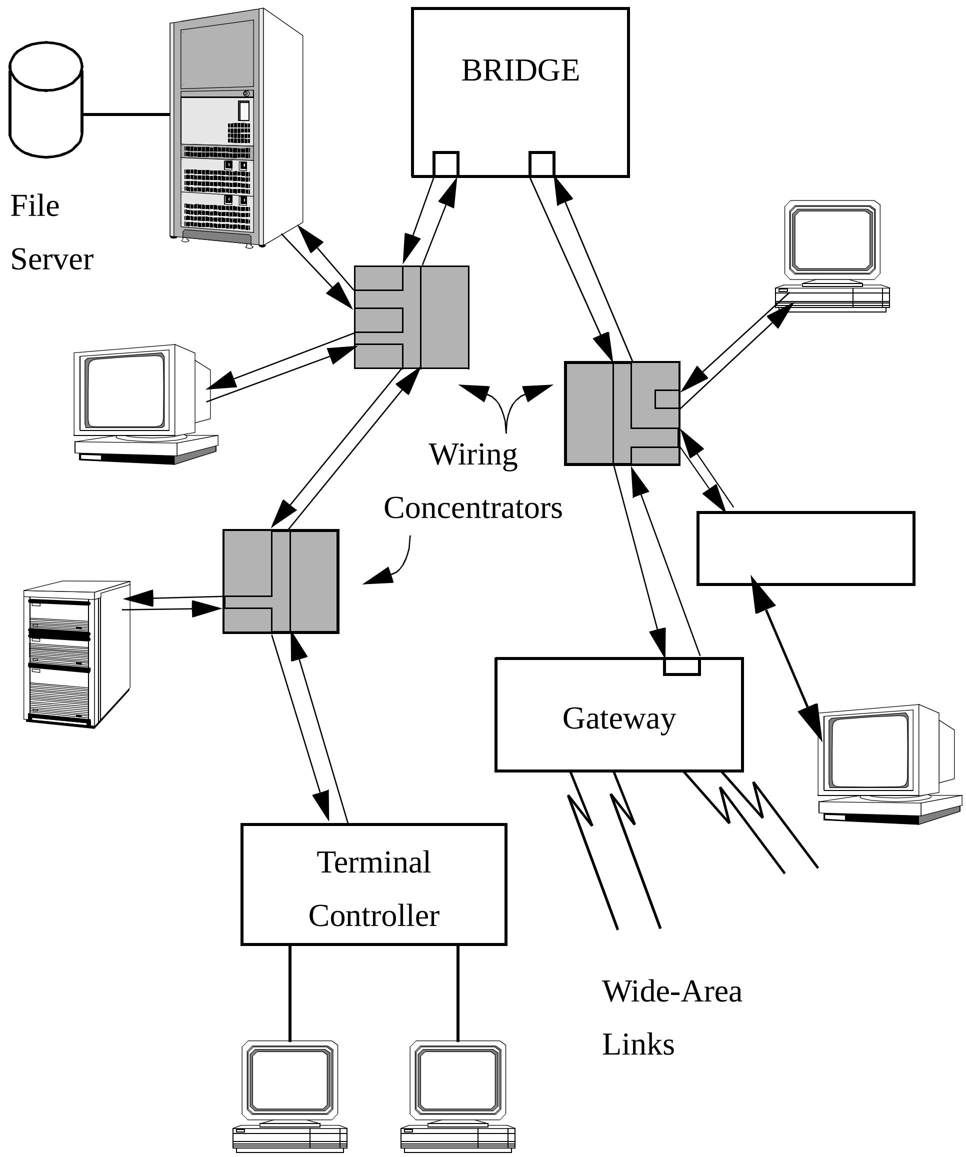
• An enhanced ring topology is pictured below

Characteristics

- Control token passed from node to node
- Only node with token can transmit
- Token replaced on ring at the end of message
- Permits variable size messages
- Only one message on the ring at a time



• The figure below shows two ring networks, with a few possible devices hooked to them.



9.1 OSI NETWORK MODEL

• Advantages
- Computers not necessarily from the same manufacturer
- Allow computers to communicate information.
- Sharing of equipment such as printers, disks, etc.
- Programs can run on multiple machines improving performance
- Access to machines with better/different resources.

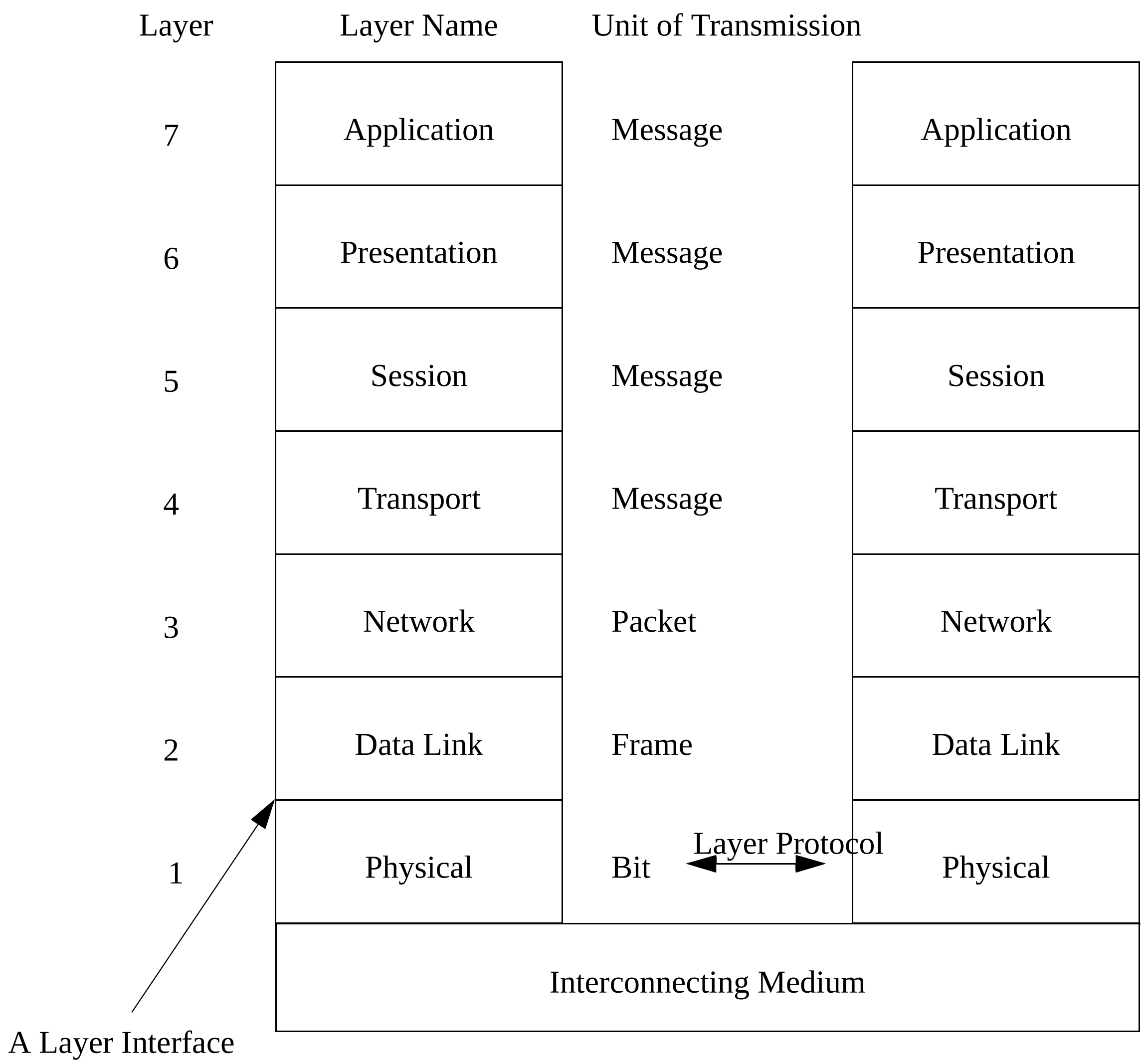
• Several types: Ring, Star, Linear, Point-to-Point

• Local Area Network (LAN), Wide Area Network (WAN)

9.1.1 Why Use A Network?

• Medium for Communication:
  - Satellite link
  - Phone line
  - High Speed (Multiplexed) phone lines
  - Coax Cable
  - Fibre Optical Cable
  - Twisted pair cable
  - (Theoretically anything can be used)

• General OSI Diagram contains seven layers



9.1.1.1 - Physical Layer

- Physical layer is concerned with transmission of raw bits over a physical circuit.
- Deals with voltages, timing, connections, etc.
- Responsible for bit synchronization and the identification of a signal element as either a 0 or a 1.
- Protocols: RS-232, RS-449, CCITT X.25 and X.21, IEEE 802

9.1.1.2 - Data Link Layer

• Data link layer breaks input data into “data frames” and processes acknowledgments.

• Object is to provide a error-free transmission line to the network layer.

• Responsible for the reliable delivery of information over a point-to-point or multipoint link.

• Supervises interchange of both link control data and user information

• Protocols: ANSI X3.28, HDLC, X.25, ISDN, IEEE 802

9.1.1.3 - Network Layer

• Network layer determines the interface between the computer and the intermediate system, how packets are routed.

• Chooses a route from the available data links that form the network.

• Object is to take messages, convert them to packets and send them towards the destination.

• Protocols: CCITT X.25, X.21, IP, CCITT Q.931, ISO 8473

9.1.1.4 - Transport Layer

• Transport layer takes data from the session layer, splits it up if necessary, and passes this to the network layer.

• Ensures that pieces all arrive correctly at the other end.

• Isolates the user from any concern for the actual movement of the information.

• Protocols: TCP, ISO 8073

9.1.1.5 - Session Layer

• Session layer defines a connection between users (presentation layer processes).

- Includes specification of the remote machine, authorization, options for the communication, and recovering from errors in broken transport connections.
- The set-up of communications is called binding.
- Protocols: ISO 8327, CCITT X.225, T.62, ECMA 75, NFS, RPC.

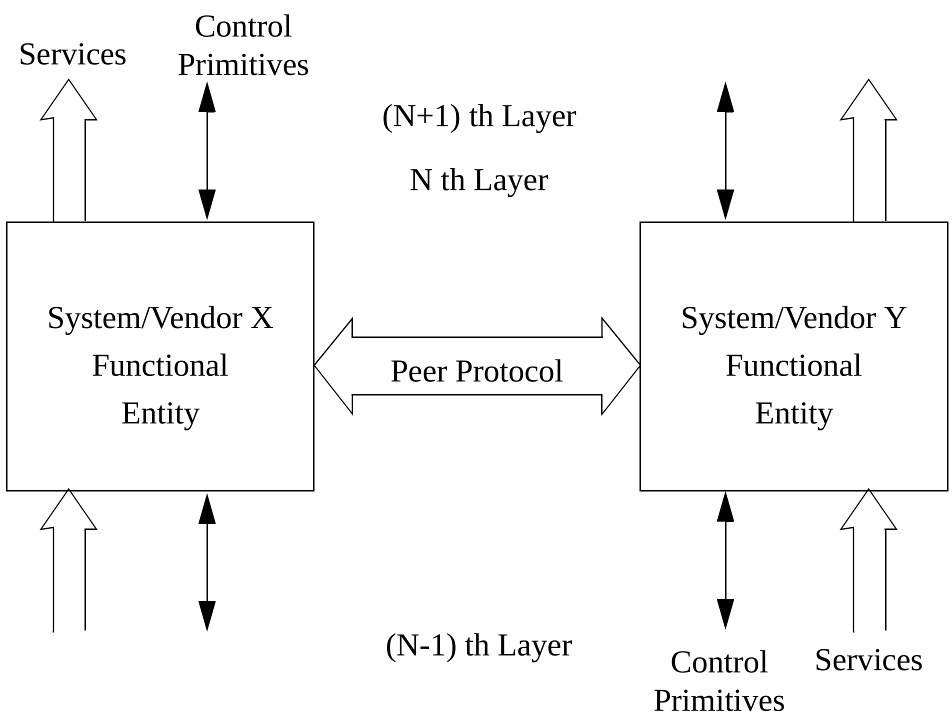
9.1.1.6 - Presentation Layer

- Ensures compatible syntax among the communicating processes by adjusting data structures, formats, and codes.
- Presentation layer is generally represented by library routines which the user accesses to perform network operations.
- This layer can also perform transformations such as compression and encryption.
- Protocols: DIS 8823, 8824, 8825, CCITT X.409, T.61

9.1.1.7 - Application Layer

- Application layer is written by the user, or is a program that performs some function for the user.
- Provides a window by which the user gains access to the communication services provided by the architecture.
- Protocols: DIS 8571, 8832, 9040

9.2 OPEN SYSTEMS



- Open Systems Principles
    - Entity is a group of functions for a particular task. Accept input, and produce output.
    - Layer is a group of functions designed to provide a set of services related to the mission of that layer
    - Entities provide a set of services to the layer above. In doing this they use the services of the underlying layers.
    - Each layer isolates the layer above from the details of the underlying layers, and thus a lower layers characteristics can change without affecting the rest of the system.
    - Primitives control the layer's services and data flow. Primitives fall into 4 groups: Request, Indicate, Response, and Confirm

- We can do a simple comparison of networking models

<table>
<tr><th>ISO</th><th>Internet</th><th>SNA</th><th>DECNET</th></tr>
<tr><td rowspan="2">Application</td><td rowspan="2">User</td><td rowspan="2">Application</td><td rowspan="4">Application</td></tr>
<tr></tr>
<tr><td rowspan="2">Presentation</td><td rowspan="2">Telnet, Ftp</td><td rowspan="2">NAU services</td></tr>
<tr></tr>
<tr><td rowspan="2">Session</td><td rowspan="2">(None)</td><td>Data flow control</td><td rowspan="2">(None)</td></tr>
<tr><td rowspan="2">Transmission control</td></tr>
<tr><td rowspan="2">Transport</td><td>Host-Host</td><td rowspan="2">Network services</td></tr>
<tr><td rowspan="2">Source to destination IMP</td><td rowspan="3">Path Control</td></tr>
<tr><td rowspan="2">Network</td><td rowspan="2">Transport</td></tr>
<tr><td rowspan="3">IMP-IMP</td></tr>
<tr><td rowspan="2">Data Link</td><td rowspan="2">Data link control</td><td rowspan="2">Data link control</td></tr>
<tr></tr>
<tr><td rowspan="2">Physical</td><td rowspan="2">Physical</td><td rowspan="2">Physical</td><td rowspan="2">Physical</td></tr>
<tr></tr>
</table>

9.3 NETWORKING HARDWARE

• A number of basic components are required for networks

• Computer (or dedicated computing equipment)
  - An obvious must. Without the computer there is no need for networking
  - Approximate cost $500 to $1,000,000

• Network Interface Hardware
  - Some computers come with built in network interfaces, If these are for the wrong network types, or there is no network interface, a networking interface must be purchased.
  - examples are ethernet, latticenet, fibre, etc.
  - Approximate Cost $200.00 and up

• The Media,

- This is the cabling which will connect the computers
- 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.
- 10baseT (twisted pair) is the most popular. It is a pair of twisted copper wires terminated with an RJ-45 connector.
- Approximate cost for thick net $40 + $5/station, for thin net $10 + $1/station
- 10base2

• Transceiver
- A device which is sometimes included between the network media, and the network interface hardware.
- Allows easy changes to media type.
- Not used for thin cable computer interfaces.
- Approximate costs $100 to $200

• Hub/Concentrator
- Connects separate wires and will route local traffic to local wires and remote through external connections.
- approximate cost $50/wire

• Bridges, (Data link layer)
- Used when hooking one network type to another, or isolating one part of a network from another.
- A bridge can be used for hooking an ethernet cable to an optical backbone cable.
- Has one input, one output, and will only handle sorting by network address (like ethernet address of machine).
- Approximate cost $2,000 to $5,000
- Same functionality as router.

• Routers, (Network Layer)
- When there are too many networks on the same network, a router can be used to isolate small loops of the network from traffic which does not involve it.
- More intelligent than a bridge, multiple inputs/outputs.
- Sorts Network packets by IP address.
- Approximate cost $5,000 to $10,000.

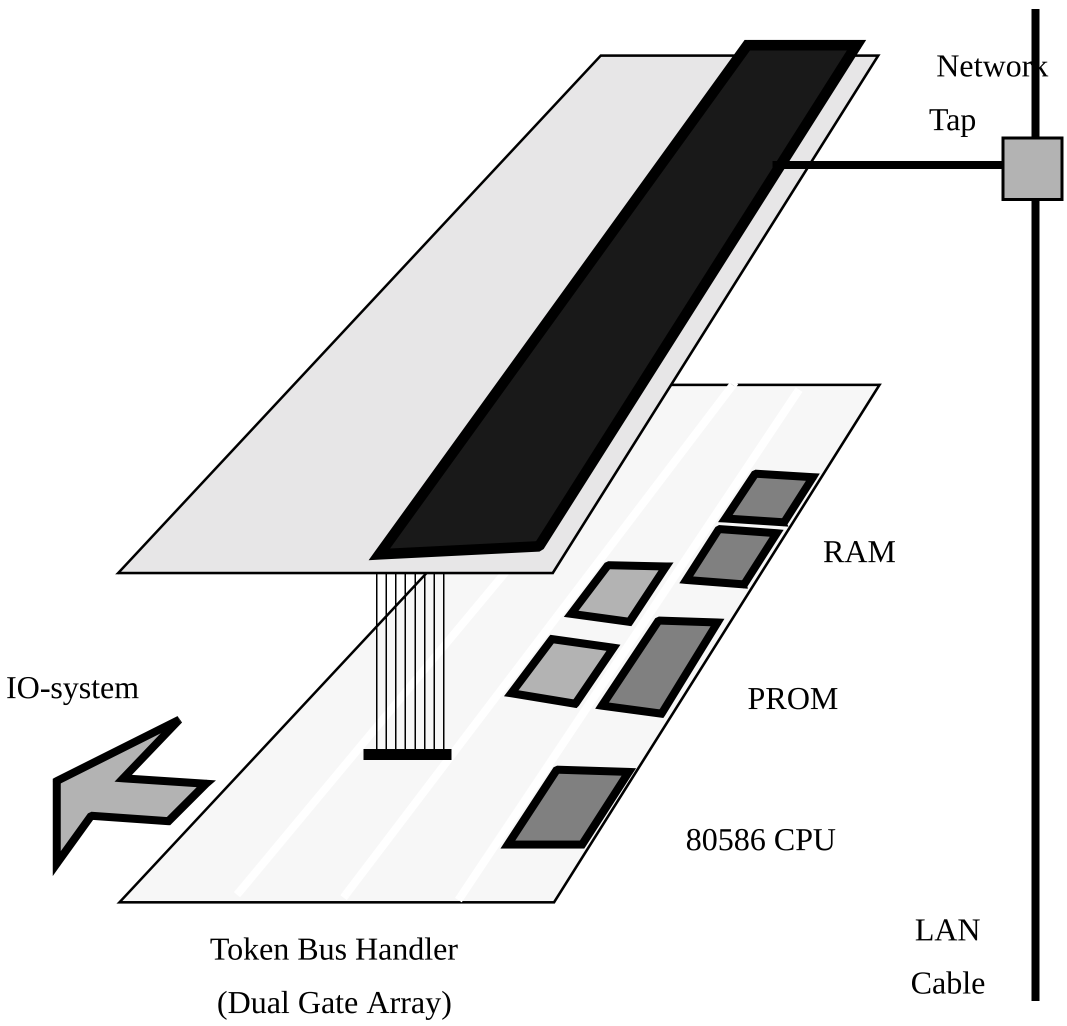
• Repeaters, (Physical Layer)
- If network cables are too long, the repeaters will boost the signal strength so that it may complete it’s journey.
- Also allows media to be changed

• Networking Software / Networking Management Software
- Handles networking interface control for,
  - receiving / sending data or files
  - electronic mail tools

- messaging systems
- etc
- Examples of this are NFS (Network File Server), SNMP (Simple Network Management Protocol)
- Approximate costs >$1,000, unless provided with system

• Gateways (Application, Presentation, Session and Transport layers)

• An example of a micro computer with a networked interface



10. MANUFACTURING AUTOMATION PROTOCOL (MAP)

10.1 OVERVIEW

• Specifically developed for computer communications in a factory environment

• 1980 GM decides to begin development of a networking protocol for the high data rates expected, while improving noise immunity. This scheme was expected to provide a common standard for all equipment to simplify integration.

• Problems,
    • Difficulties have arisen getting Countries and Vendors to agree on specific standards
    • Standards are so broad that they have become very complex and hard to develop hardware and software for, thus driving up the costs
    • Versions 2.1 and 2.2 left the application layer under-defined, but a new effort in version 3.0 to define the application layer. This has lead to even higher levels of disagreement.
    • The MAP Protocol is not clearly defined in a single document

• In 1985 GM realized that it couldn't meet its needs for CIM. Only 15% of the 40,000 programmable devices could communicate.

• No single vendor could meet all the needs, and multiple vendors caused communication incompatibilities.

10.2 DETAILS

10.2.1 Physical Alternatives (Layer 1 OSI Model),

• Networks,
    • Broadband
        - many simultaneous communications on same wire at same time, using varied frequencies for transmit channels and receive channels.
        - uses a head end remodulator to retransmit the signal
        - each station needs MAP hardware
        - for very large factories
        - uses a 1 inch (2.5 cm) thick cable
        - maximum 10 Megabits/sec. per channel

- Carrierband
    - less expensive
    - uses smaller cables
    - rate of 5 to 10 Megabits second (single channel)

- Other Equipment,

    - Bridge
        - allows connections between different network types (such as a connection between Broadband and Carrierband)

10.2.2 Data Link Layer (Layer 2 OSI Model),

- Real time response was required, so token passing is used.

- Token passing allows a ‘token’ to be passed between computers, whichever computer has the token is given control of the network.

- The token is continually passed to each machine on a network.

- If a token passing ring is broken, the computers will wait, and eventually each half of the broken ring will have tokens being passed.

- Extensive algorithms have been developed for control of token passing, and resolution of conflicts.

- Preferred Standards.

- Preferred LAN is IEEE 802.4 token-passing bus.

- Uses standard ISO network, transport, session, and presentation layers.

10.2.3 Application Layer:

- Some specifics,
    - FTAM (File Transfer Access Method) for file transfer
    - MMS (Manufacturing Message System) for content of messages bound for robotic equipment
    - ACSE (association Control for Service Elements) for program to program communication.
    - Set of protocols should make communication between heterogeneous devices, not only

in moving data around, but also in communication the meaning of that data.
- MAP has been delayed because of changes to the standard between 2.1 (1985) and 3.0 (1988)
- A version of MAP, EPA (Enhanced Performance Architecture) has been developed, and is supported by the process control industry, and CNMA (Communication Networks for Manufacturing Applications)
- CNMA has lobbied for the inclusion of Ethernet (IEE 802.3) as an alternative for the standard.

• MAP Layers Compared with the ISO/OSI Reference Model

<table>
<tr><th>ISO/OSI
Reference
Model</th><th>MAP Version 3.0</th></tr>
<tr><td>Application</td><td>Directory Services
Network Management
MMS FTAM ACSE</td></tr>
<tr><td>Presentation</td><td>Presentation</td></tr>
<tr><td>Session</td><td>Session</td></tr>
<tr><td>Transport</td><td>Transport Class 4</td></tr>
<tr><td>Network</td><td>CLNS

ES-IS</td></tr>
<tr><td>Data Link</td><td rowspan="2">IEEE 802.2

IEEE 802.4 10M-byte
Broadband
or
5M-byte Carrier Band</td></tr>
<tr><td>Physical</td></tr>
</table>

Notes:

ACSE	Association control service element
CLNS	Connectionless network service
ES	End system
FTAM	File transfer and access method
IS	Intermediate system
MMS	Manufacturing message specification

10.3 DETAILS FOR TOP

• TOP (Technical and Office Protocols) is designed for the office environment.

• Preferred LAN is IEEE 802.3 Ethernet, But allows for Token Bus (IEEE 802.4) and Token Ring (IEEE 802.5). Fibre optical media will be added when they are defined.

• uses same ISO standards for network, transport, session and presentation layers that MAP uses

10.3.1 Application Layer:

• ACSE and FTAM are the same as MAP, MMS is not included

• (MHS) CCITT X.400 message handling service provides electronic mail transfers of all types of documents

• ISO Virtual terminal protocol to allow a terminal on any machine to log onto any other machine.

• ODA/ODIF (ISO Office Document Architecture and Office Document Interchange Format) To permit formatted documents to be transmitted independently of the package used to create them. (Text, and Raster or Line graphics)

• CGM (ISO Computer Graphics Metafile) to permit the exchange of geometrical graphics

• IGES (ANSI Initial Graphics Exchange Standard) to exchange product design information among CAD/CAM workstations.

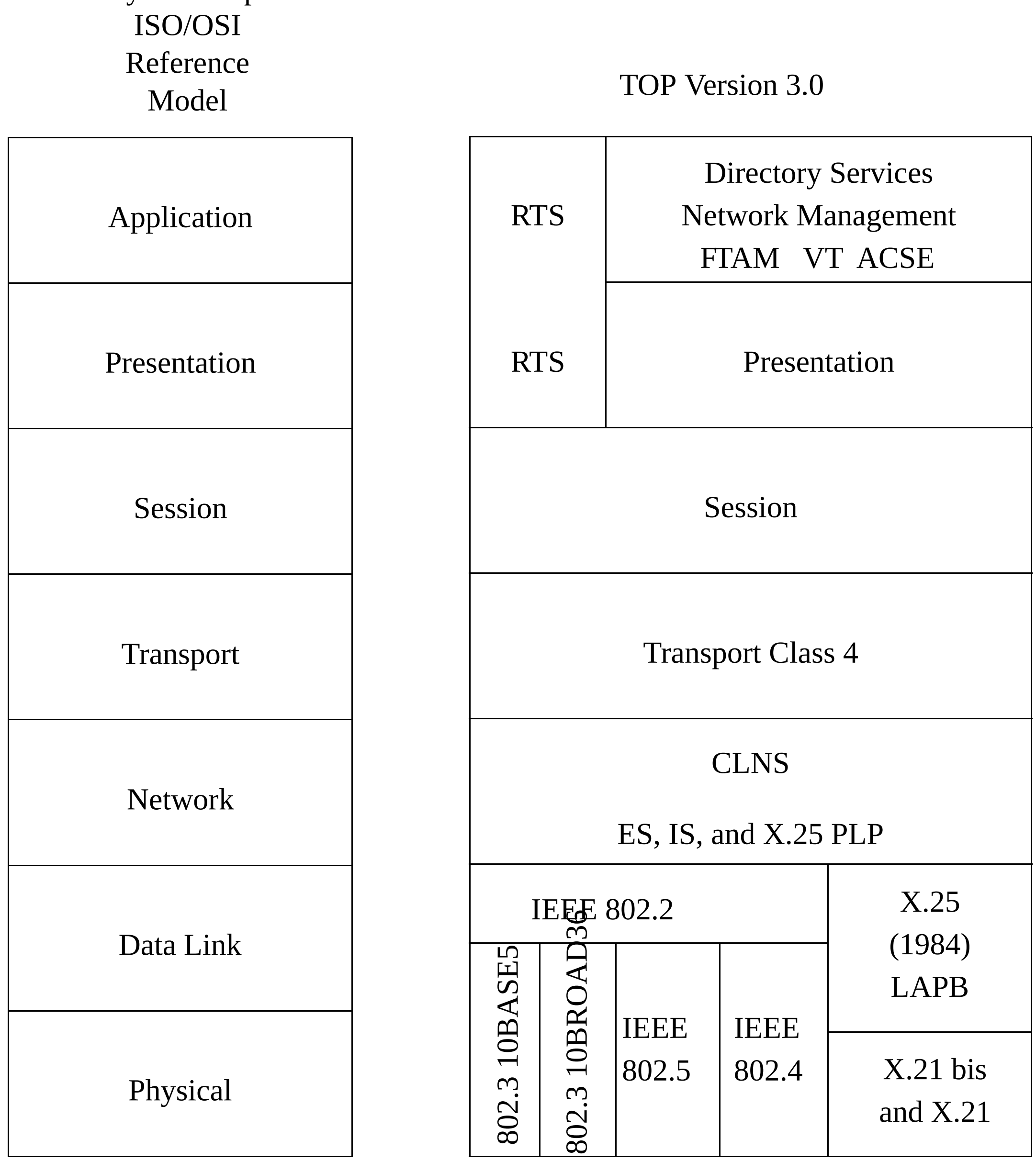
10.4 MAP & TOP

• Suggested that the two protocols MAP for CAM, and TOP for CAD be separated at the database level.

• STEP (Standard for the Exchange of Product Data) based on PDES, is an international standard to facilitate the transfer of data in CAD/CAM.

• GOSIP (Government Open Systems Interconnection Profile) has been developed by the U.S. Government (NIST) to provide an open systems standard fro their computers. GOSIP is almost a subset of TOP 3.0

• TOP Protocol Layers Compared with the ISO/OSI Reference Model

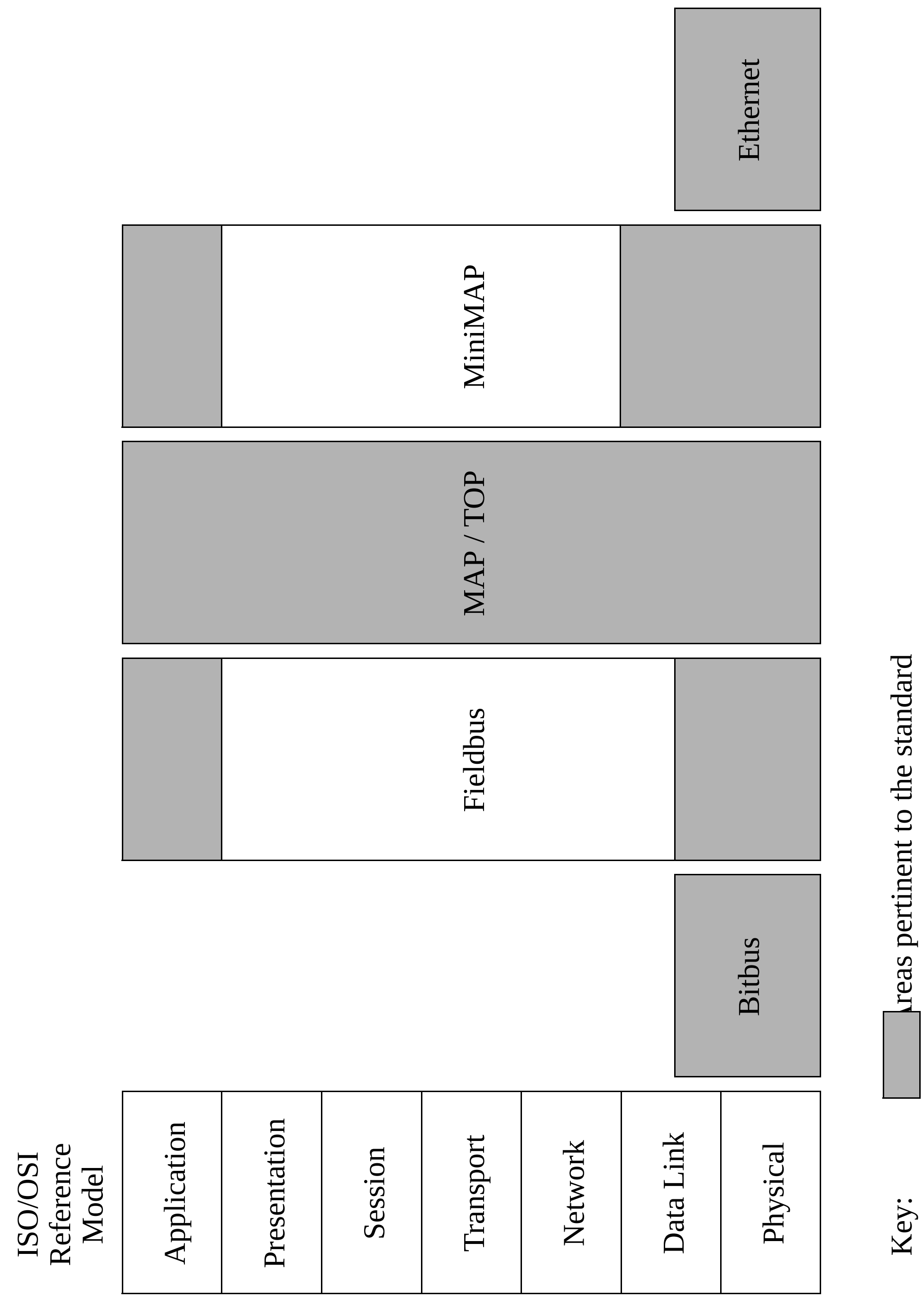


Notes:

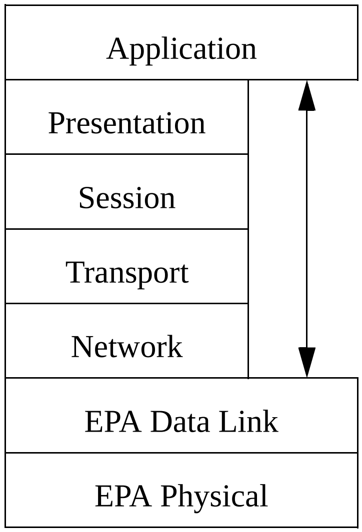
ACSE	Association control service element
CLNS	Connectionless network service
ES	End system
FTAM	File transfer and access method
IS	Intermediate system
LAPB	Link access protocol-balanced
MHS	Message-Handling System
PLP	Physical layer protocol
RTS	Reliable transfer system
VT	Virtual Terminal

10.5 MAP AND OTHER STANDARDS

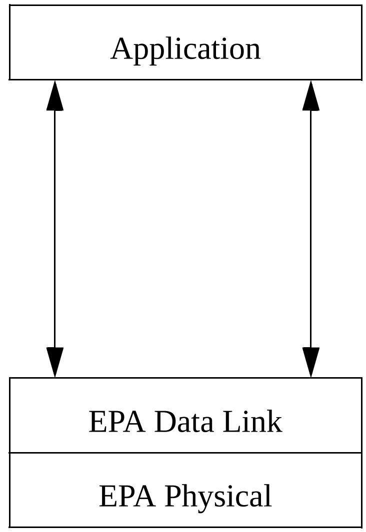
• ISO/OSI Reference Model Compared with Factory Communication Standards



• MAP Enhanced Performance Architecture (EPA) and MiniMAP Protocol Stacks



a. MAP/EPA System



b MiniMAP System

10.6 AN EXAMPLE OF A MAP IMPLEMENTATION

• One proposed MAP-compliant approach to computer integration is the introduction of an FMS into the plant-wide data communications network)

• All seven MAP layers reside and run on each board set of this board-level MPA interface

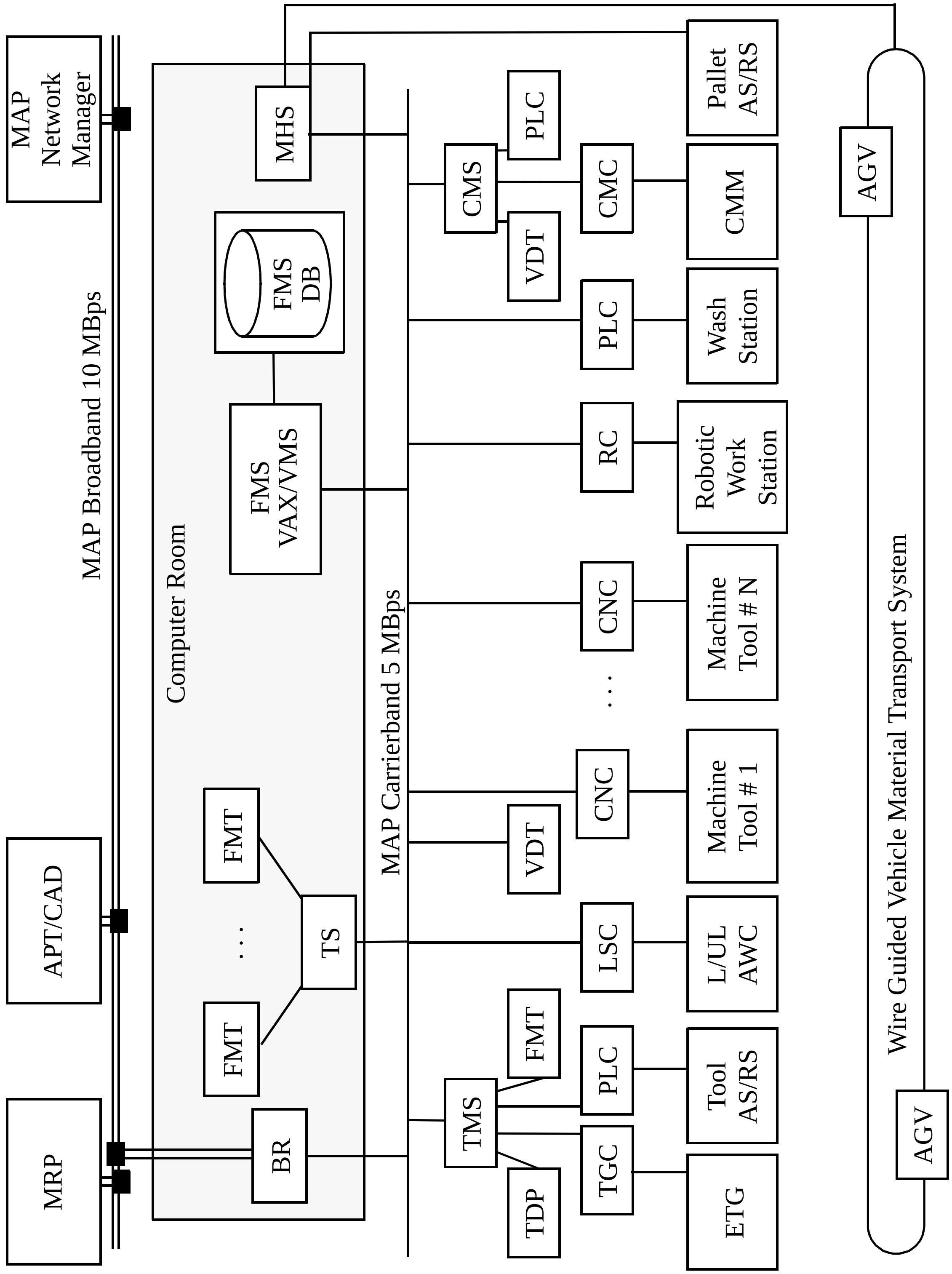


Table 2:

Legend	
MRP	Material Requirements Planning
APT/CAD	Automatic Programmed Tool/ Computer-Aided Design
BR	Bridge
FMT	FMS Terminal
TS	Terminal Server
FMS DB	FMS Database
MHS	Material Handling System
TMS	Tool Management System
CMS	Coordinate Measuring System
TDP	Tool Data Panel
TGC	Tool Gage Control
PLC	Programmable Logic Control
LSC	Load Station Controller
VDT	Video Display Terminal
CNC	Computer Numerical Controller
RC	Robot Controller
CMC	Coordinate Measuring Control
ETG	Electronic Tool Gage
AS/RS	Automatic Storage/Retrieval System
L/UL	Load/Unload
AWC	Automatic Work Changer
CMM	Coordinate Measuring Machine

10.7 ETHERNET

10.7.1 Internet

10.7.2 SLIP/PPP

10.8 DATA HIGHWAY+

10.9 REMOTE PLC I/O

10.10 DEVICENET

10.11 OTHER STUFF

10.12 Network Facts

• Speeds often measured in bps (bits per second) but we are more interested in the bytes per second. There is approximately an 11:1 ratio.
T1 is 1.544 Mbps
T3 is 44.54 Mbps

• Network transmission,
baseband - single frequency- only one station may transmit at any time (Ethernet, ARC-

net, Token ring). These must be in discrete digital packets.

boradband - multiple frequencies allow multiple simultaneous data streams. This network can reach higher speeds by multiplexing and can have continuous and/or analog channels (ISDN, CATV,ATM)

• Networks are some mix of clients and servers.

server - will share reseaources (eg, files) with many remote computers

client - a computer that typically talks to a limited number of servers

• Token based networks rotate talk permissions between machines by passing a "token"

• ARCnet - a token passing network that is relatively immune to noise

• IBM token ring - a star topology token passing network

• FDDI - fiber based token passing rings that rotate in opposite directions

• Ethernet - both bus and star topologies

11. DATABASE TECHNOLOGY

11.1 DISTRIBUTED DATABASE SYSTEMS

11.1.1 Relational database systems

•Information is stored in 2D-tables

•Rows in a table are records

•Columns in a record are fields

•A key, consisting of one or more fields, that uniquely identifies a record

•A dictionary is a table that describes all the tables

•Fully-partitioned: if each table is stored at exactly one physical site

•Fully-replicated: if each table is stored at all physical sites

•Natural distribution: data are kept at the local site

•Query operations:
- Select - picking records
- Project - picking fields
- Join - merging of tables

•Predicate: a condition between fields used to manipulate the queries

11.1.2 Issues for distributed database systems

• Distribution of tables to sites

• Natural distribution of data at various sites

• Fully partition of systems

• Fully replicated systems

• Important factors

• Replicates the dictionary at every site

• Frequency of request to a table from a site

• Storage capacity at each site

• Communication costs between sites

11.1.2.1 - Query processing

• Query response time (for interactive applications)

• Total bandwidth consumed (for batch applications)

• Approach
  - Optimizing specific query based upon specific statistical conditions
  - The query site will either “estimate” or request the related sites to report the related time and cost of moving the data before deciding on an actual query sequence

11.1.2.2 - Concurrency control

• Maximize the amount of parallel activity while maintaining the semantic integrity of the data

• Approach
  - A transaction: a set of reads, followed by some processing, and then a set of writes
  - A log is the time ordered sequence of reads and writes performed on the database
  - A log is serial if each read is immediately followed by an appropriate write
  - There is no known algorithm that allows serial logs, all serializable logs, and all other logs that leave the database consistent
  - Most algorithms achieve serializable logs by allowing transaction to lock part of the database
  - The lock could be applied on the full database, or some tables, records, fields, and physical sectors
  - Deadlock occurs when two queries want to lock certain resources that have already been locked by each other