Microcomputer Interfacing

SSP 3312

Lecturer : *Dr Abd Rahman Tamuri
Assoc. Prof Dr Yaacob Mat Daud
Mr Mohd Khalid Kasmim
Dr Abdul Khamim Ismail
Email : rahmantamuri@utm.my

Interfaces and Interfacing

- **Definitions** of “interface” from Webster’s Dictionary:
  - noun: the place at which independent systems meet and act or communicate with each other

    examples:

    human - machine interface (analogue-machine interface)
    terminal - network interface (TTL - CMOS interface)

    parallel or serial interface

- **Informal Definition**

  i) The physical, electrical and logical means of exchanging Information with a functional module

  ii) The process of enabling a computer to communicate with the external world through Software, Hardware and Protocols

- An interface is a device and/or set of rules to match the output of one device to send information to the input of another device
  - physical connection
  - the hardware
  - rules and procedures
  - the software
- Interfacing is the process of connecting devices together so that they can exchange information
- The process of reading input signals and sending output signals is called I/O
- I/O conventions
  - I/O direction is relative to the MCU
  - Input is data read by the MCU
  - Output is data sent out by the MCU

Why is computer interfacing important ?

1. The human - machine interface determines the ultimate success or failure of many computer- based systems

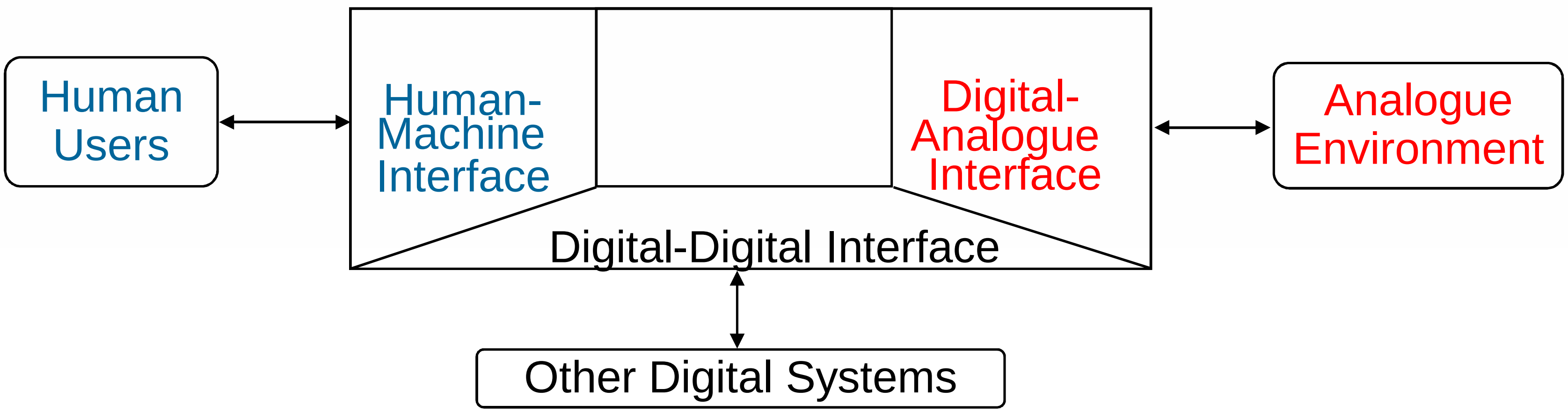
2. Digital systems exist within and must successfully interact with an analogue natural environment (Digital - analogue interfaces are unavoidable)

3. Rather than designing digital systems from elementary components, computer engineers more typically assemble new systems from existing subsystems

Typical Interfacing Activities

- Selecting software/hardware subsystems that can (at least potentially) interact well with each other
  - Appropriate D/A and A/D converters (speed, accuracy, …)
  - Serial vs. parallel communication
- Providing appropriate hardware connections
  - Selecting cabling, connectors, drivers, receivers, correct termination, etc.
- Resolving any hardware incompatibilities
  - CMOS with TTL
- Configuring hardware interfaces correctly using low-level software drivers
  - LCD, Keypads in embedded systems
- Interfacing software components correctly
  - Selecting compatible software versions
  - Calling the correct procedures in the correct sequence with the correct parameters

System-Level Interfaces



Human-machine interface

- Input devices: keyboard, mouse, microphone, camera
- Output devices: CRT, printer, light panel, audio amp

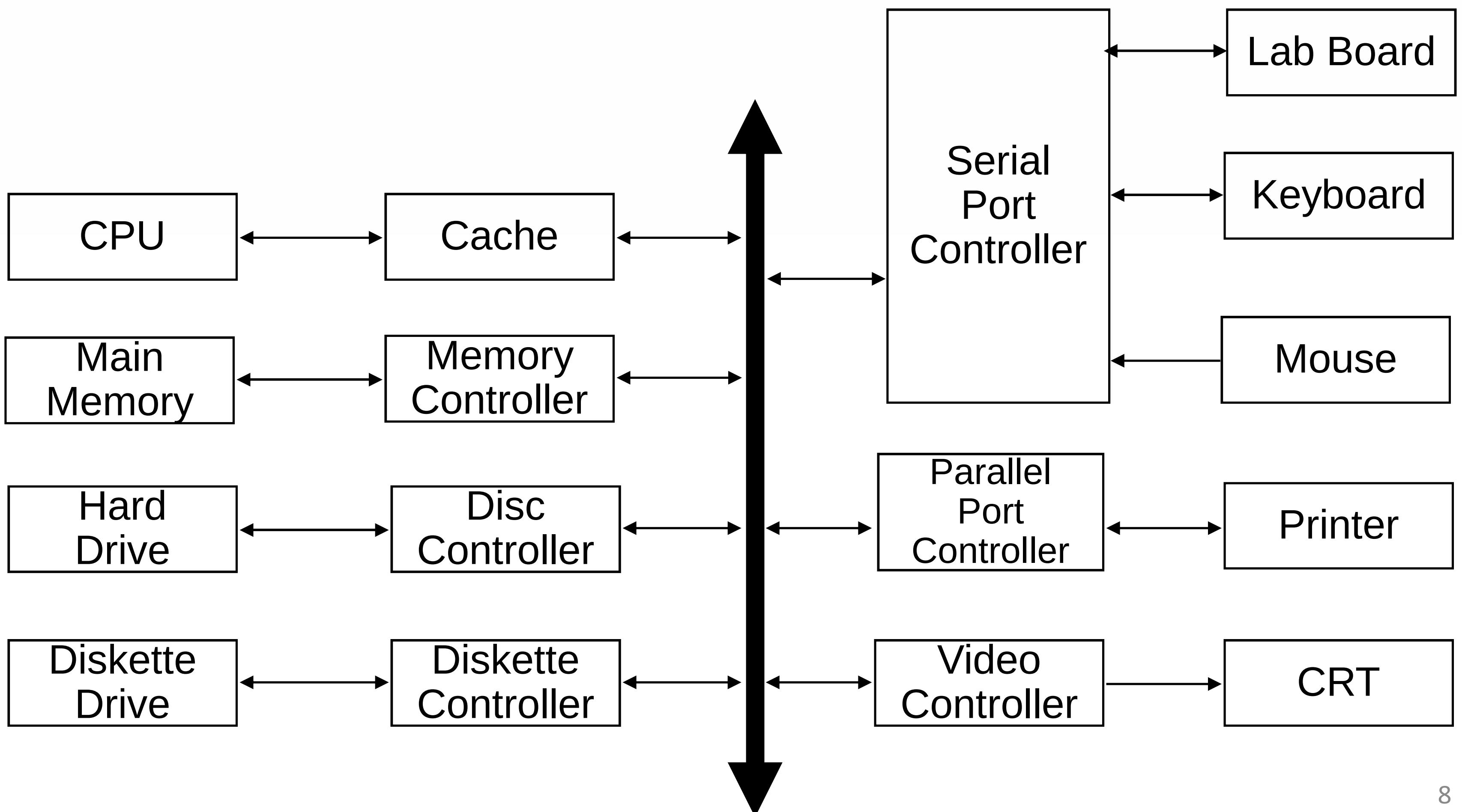
Digital - Analogue Interface

- Input devices: A/D converters, modems, sensors
- Output devices: D/A converters, modems, transducers actuators, stepper motors
- Control devices: switches, multiplexers, amplifiers attenuators

Digital - Digital Interface

- Connectors: wires, ribbon cable, coax, twisted pair, PCB
- I/O devices: buffers, level-shifters, synchronizers

Hardware Interfaces within a Personal Computer (PC)
(greatly simplified)



Standard Microcomputer Interfaces

- Parallel Interface
- Serial Interface
- Universal Serial Interface (USB)

Other Interfaces

- General Purpose Interface Bus (GPIB)
- Custom Interfaces

Parallel Interface

- Originally known as Centronic parallel interface
- Centronic parallel interface is an older and still widely-used standard I/O interface for connecting printers and certain other devices to computers
- The interface typically includes a somewhat cumbersome cable and a 36- pin male and female connector at the printer or other device
- The cable plugs into a 25-pin parallel port on the computer
- Data flows in one direction only, from the computer to the printer or other device
- In addition to eight parallel data lines, other lines are used to read status information and send control signals
- Centronics Corporation designed the original Centronics parallel interface for dot matrix printers.
- When the Centronics parallel interface was first developed, the main peripheral was the printer
- Since then, portable disk drives, tape backup drives, and CD-ROM players are among devices that have adopted the parallel interface
- These new uses caused manufacturers to look at new ways to make the Centronics parallel interface better

- In 1991, Lexmark, IBM, Texas instruments, and others met to discuss a standard that would offer more speed and bi-directional communication
- Their effort and the sponsorship of the IEEE resulted in the IEEE 1284 committee. The IEEE 1284 standard was approved for release in March, 1994
- The IEEE 1284 standard specifies five modes of operation, each mode providing data transfer in either the forward direction (computer to peripheral), backward direction (peripheral to computer), or bi-directional (one direction at a time)
  - **Compatibility mode**
    - the original Centronics parallel interface and intended for use with dot matrix printers and older laser printers
    - The compatibility mode can be combined with the nibble mode for bi-directional data transfer.
  - **Nibble mode**
    - allows data transfer back to the computer
    - The nibble mode uses the status lines to send 2 nibble s (4-bit units) of data to the computer in two data transfer cycles
    - This mode is best used with printers.

- **Byte mode**
  - uses software driver s to disable the drivers that control the data lines in order for data to be sent from the printer to the computer
  - The data is sent at the same speed as when data is sent from the computer to the printer
  - One byte of data is transferred instead of the two data cycles required by the nibble mode.
- **ECP mode** (Enhanced Capability Port mode)
  - an advanced bi-directional mode for use with printers and scanners
  - It allows data compression for image s, FIFO (first in, first out) for items in queue s, and high-speed, bi-directional communication
  - Data transfer occurs at two to four megabytes per second. An advanced feature of ECP is channel addressing
  - This is used for multifunction devices such as printer/fax/modem devices
  - For example, if a printer/fax/modem device needs to print and send data over the modem at the same time, the channel address software driver of the ECP mode assigns a new channel to the modem so that both devices can work simultaneously.

- **EPP mode** (Enhanced Parallel Port mode)
  - designed by Intel, Xircom, and Zenith Data Systems to provide a high-performance parallel interface that could also be used with the standard interface
  - EPP mode was adopted as part of the IEEE 1284 standard
  - The EPP mode uses data cycles that transfer data between the computer and the peripheral and address cycles that assign address, channel, or command information
  - This allows data transfer speeds of 500 kilobytes to 2 megabytes per second, depending on the speed of the slowest interface
  - The EPP mode is bi-directional
  - It is suited for network adapters, data acquisition, portable hard drives, and other devices that need speed

Serial port

- Standard PC serial ports come in to versions: 9 pin and 25 pin one
- The functions of those both version are exactly the same, only different kind of connectors and different pinout
- PC serial port is nowadays usually used for interfacing PC to modem or mouse
- Original PC serial port was designed to operate up to 19.2 kbit/s (maximum speed defined in RS-232C standard) but nowadays they can typically go up to 115.2 kbit/s (some special cards can do even faster than that)
- PC serial port send and receives data in serial format
- In serial, asynchronous data transfer the individual bits which comprise each data byte are sent one after the other over a single line
- In this context, asynchronous means that the clock information is not included with the transmission, so that frequent re-synchronization using start/stop bits is required
- The maximum length specified by RS-232 is only 50 feet (around 15 meters), however much longer lengths are possible with proper shielding on the cable
- Generally you can run 9600 bps communication up to 250 feet (80 meters) over shielded data cable or unshielded twisted pair cable in good environment
- When using shielded cable and slower data rate longer lengths are possible (up to hundreds of meters in good conditions)

- The processing element of PC serial port is the UART
- UART is an integrated circuit used for serial communications, containing a transmitter (parallel-to-serial converter) and a receiver (serial-to-parallel converter), each clocked separately
- The parallel side of a UART is connected to computer bus (usually ISA bus)
- When the computer writes a byte to the UART's transmit data register (TDR), the UART will start to transmit it on the serial line
- The UART's status register contains a flag bit which the computer can read to see if the UART is ready to transmit another byte
- Another status register bit says whether the UART has received a byte from the serial line, in which case the computer should read it from the receive data register (RDR)
- If incorrectly formated data is received the UART may signal a "framing error" or "parity error“
- The UART may be set up to interrupt the computer when data is received or when ready to transmit more data
- Data on the serial line is formatted by the UART according to the setting of the UART's control register
- Those registers control the baud rate, number of data bits, number of stop bits and what kind of parity is used (odd, even or none).

- The original UART chip shipped with the IBM personal computer was the 8250
- This chip was limited to 9600 bps maximum rate
- It was replaced with the 16450 which had the same architecture as the 8250 but has a higher maximum bps specification (115200 bps)
- Both of the chips only have a one byte FIFO/buffer
- The newer 16550 UART contains a 16-byte buffer which helps the computer in the communications and makes faster communication speeds without lost characters possible
- A 16-byte FIFO allows up to 16 characters to be received before the computer has to service the interrupt
- There are also other UARTs with longer FIFOs
- When operating under DOS at speeds below 9600 bps the 16450 should provide satisfactory performance
- When operating under any Windows or other multitasking operating system, a 16450 will be limited to about 1200 or 2400 bps reliable communication 16550 or similar UART will work on multitasking enviroment at high speeds very nicely.

- Having FIFO in the UART means that the computer need to have interrupts less often, which means less processing power wasted on the serial communications and less likely that data is lost if it takes some time from interrupt request that actual interrupt happens
- The UART's serial connections go via separate line driver and line receiver integrated circuits which provide the power and voltages (+12V and -12V) required to drive the serial line and give some protection against noise on the line.
- Most modern PC serial cards and built-in serial ports are designed for maximum 115200 bps data rate
- There are also some special serial port card which can handlealso higher data rates (quadruple the clock speed (4X) will allow transmission speeds up to 460,800 bps and a card with 8X setting will allow speeds up to 921,400 bps).

Universal Serial Bus (USB)

- USB, or the Universal Serial Bus Interface is now well established as an interface for computer communications
- In many areas it has completely overtaken RS232 and the parallel or Centronics interface for printers, and it is also widely used for memory sticks, computer mice, keyboards and for many other functions
- One of the advantages of USB is its flexibility: another is the speed that USB provides
- USB provides a sufficiently fast serial data transfer mechanism for data communications, however it is also possible to obtain power through the connector and this has further added to the popularity of USB as many small computer peripherals may be powered via this
- From memory and disk drives to other applications such as small fans and coffee cup warmers, the USB port on computers can be used for a variety of tasks

USB Evolution

- The Universal Serial Bus (USB) was born out of the frustration PC users experience trying to connect an incredibly diverse range of peripherals to their computers
- It was developed as a result of the need for a communications interface that was convenient to use and one that would support the higher data rates being required within the computer and peripherals industries
- Universal Serial Bus came into life when a group of 7 companies : Compaq, Digital Equipment, IBM, Intel, Microsoft and Northern Telecom decides to form a specifications to merge legacy connectivity such as RS232, Printer port, PS2 port into a single common connector to the Personal Computer
- The first proper release of a USB specification was Version 0.7 of the specification. This occurred in November 1994. This was followed in January 1996 by USB 1.0. USB 1.0 was widely adopted and became the standard on many PCs as well as many printers using the standard

- The motivation of differentiating two transfer speed was to maintain the low-cost implementation of computer peripherals such as keyboards and mice, and, still allow higher speeds devices such that printers and scanners to be able to use the same serial bus
- USB was agreed to as a standard by Microsoft, Compaq and many other large names in PC industry
- USB 1.0 standard was finally issued in 1996
- It was later modified to USB 1.1. Version USB1.1 was delivered on 23 September 1998
- The 1.1 specifications clarified many timing parameters which were grey-areas in the past. However, no huge "functional" improvements were given

- In addition to this a variety of other peripherals adopted the USB interface, with small memory sticks starting to appear as a convenient way for transferring or temporarily storing data
- With USB 1.0 well established, faster data transfer rates were required, and accordingly a new specification, USB 2 was released
- With the importance of USB already established it did not take long for the new standard to be adopted
- With USB defining its place in the market, other developments of the standard were investigated
- With the need for mobility in many areas of the electronics industry taking off, the next obvious move for USB was to adopt a wireless interface
- In doing this wireless USB would need to retain the same flexible approach that provided the success for the wired interface
- In addition to this the wireless USB interface needs to be able to transfer data at rates which will be higher than those currently attainable with the wired USB 2 connections
- To achieve this ultra-wideband UWB technology is used

THANK YOU