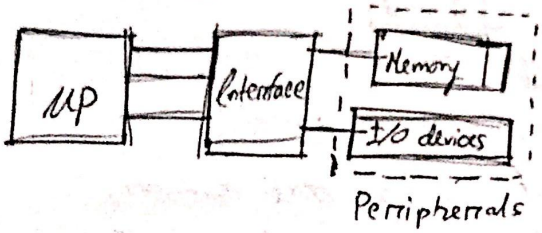
19-11-21
Friday

#1



Embedded devices → Incorporate multiple functionalities in a single device.

⊛ Interfaces are programmable

Interface ex- USB, HDMI, NIC, Fiber Channel, Firewire.
↳ (Wifi, Bluetooth, Ethernet)

#2

7 segment display → Pixel based display
(LED/LCD)

IN AL, 1712H → Port No.
OUT AB12H, DL

Use (Basic Input Output Subroutine)

[BIOS]

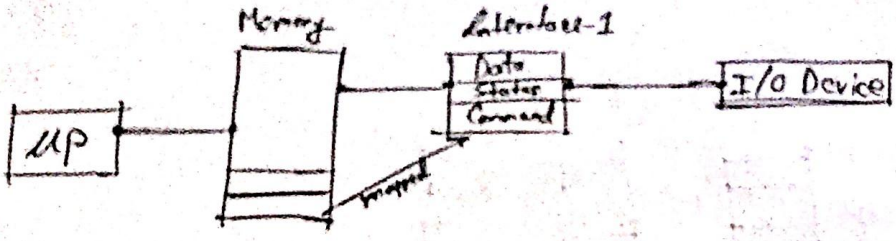
These are called BIOS functions.

Additionally, we have Interrupt-driven functions, DMA functions.
(DOS subroutine)

I/O controller registers — data register [buffer]
status "
command "

Q. Better than BIOS?

DOS functions → we don't have to memorize port numbers.
Instead we call a particular interrupt.



16 bit memory → 2^{16} = 64 kB used for port number.
from the memory [Usually first 64 kB]

This is called (memory mapped) I/O.

Disadvantage i) Allocated memory results in wastage of memory.

Isolated I/O

→ Seperate 64 KB memory from the main memory.
(support from Pentium processors)

Ways to Access → Memory Mapped
→ Isolated

Protocal to Communicate → Programmable I/O
→ Interrupt-driven I/O
→ DMA

i) Programmed I/O → (Polling)
µp always initiates the communication

ii) Interrupt-driven
I/O device sends interrupt

iii) DMA
→ Some memory operations (by I/O devices) don't require µp.

→ Direct Access of memory by I/O devices.

Adv: Efficient to transfer large volume of data.

Controllers → I/O controllers
→ Interrupt "
→ DMA "

#2

26-Nov-21
Friday

Microcontrollers

→ A small computer on a single chip
→ Has processor, primary, secondary memory and other required components

Microcontroller's advantages over µp
- Decreased pin count
- Cost, size, design time is decreased
- More reliable with easier upgrade path

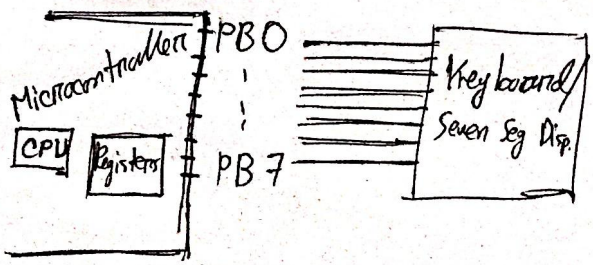
⊛ Microcontroller is more target specific than microprocessor.

AVR (Advanced Virtual RISC)

- Tiny - Mega - Xmega
in (Event-triggered systems)
↳ Fire detection

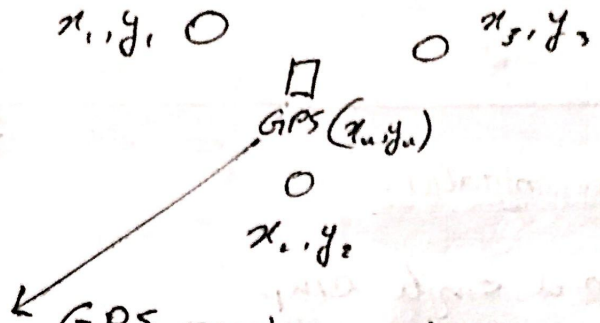
Company AVR
Pins of ATMEGA-16 (16KB)

→ Bi-directional I/O ports are directed integrated to microcontroller as pins unlike microprocessor.
(both Input and Output)



• Port A takes input for analog data to the ADC.

Trilateral Method



GPS receiver must be connected to at least three "known" devices.

#3

ADC and DAC

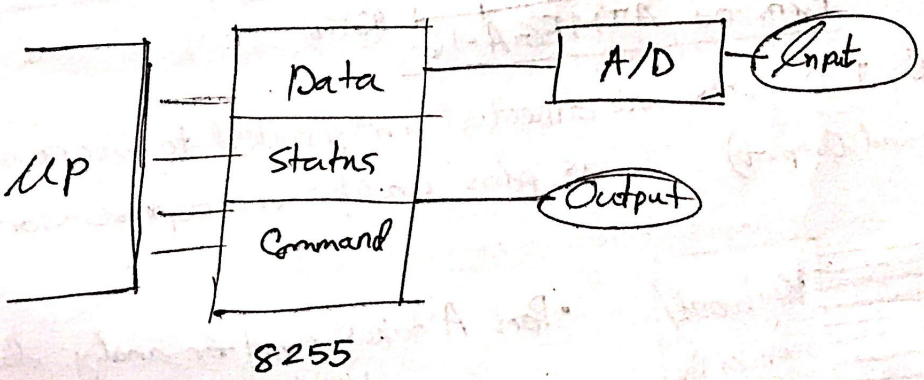
29-11-21
Monday

Transducer (Collects analog data) → ADC → Digital Logic → DAC → Control System

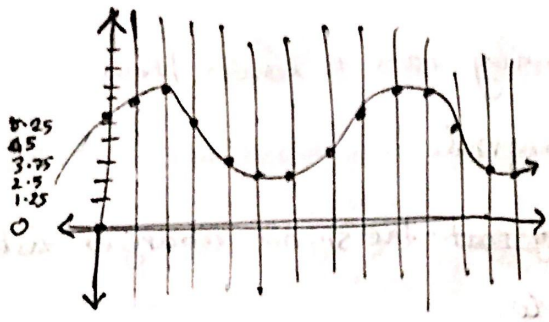
Why require modulation?

Ex - human voice (4-5) Hz which doesn't have enough frequency to penetrate the wall.

Combining with a higher frequency will make it travel longer distance.



ADC → Quantizing → i) Take some samples
ii) Map values with those samples
→ Encoding



← Dividing the signal into lines is sampling and getting those points

(only these values are possible)

Take the ~~ele~~ value ~~that~~ closet to the point

This is quantization.

Q- Why increment by 1.25.

(Resolution: No. of discrete values converter can output)

Possible states of output → 2^n (We chose this)

Here $n=3$, $2^3=8$.

$$\text{Increment} = \frac{Q_{max} - Q_{min}}{2^n} = \frac{(10v-0v)}{8} = 1.25$$

Analog Value	Output	Binary Equivalent
0 - 1.25	0	000
1.25 - 2.5	1	001
2.5 - 3.75	2	010
3.75 - 5	3	011
5 - 6.25	4	100
6.25 - 7.5	5	101
7.5 - 8.75	6	110
8.75 - 10	7	111

⊛ Improve Accuracy → i) ↑ Resolution
ii) ↑ Sampling Rate

Aliasing Problem → If sampling rate is lower than input signal.

i.e. it occurs when ~~sampling rate~~ the signal changes faster than we can sample.

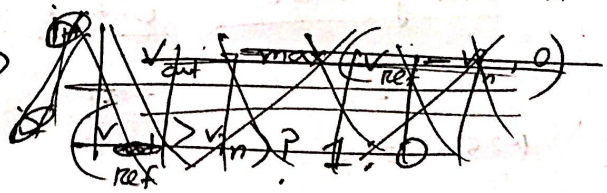
Solution → Sample at least twice (Nquist theorem)

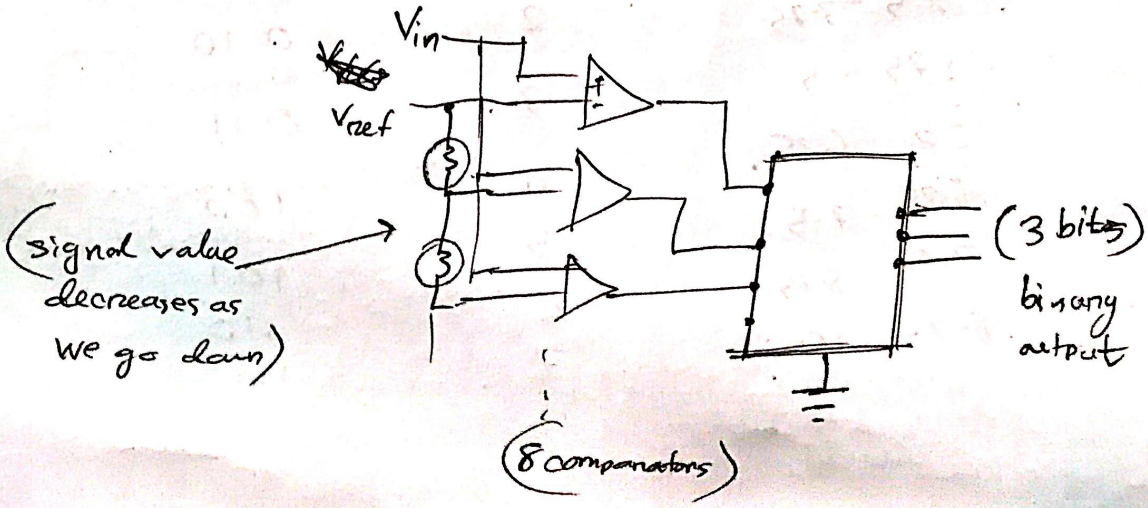
⊛ A/D Converters → Flash ADC
→ Delta-Sigma ADC
→ Successive Approximation ADC

Flash ADC

Comparator V_{in} V_{ref} — High if $V_{in} > V_{ref}$

$V_{out} = V_{in} > V_{ref} \; ? \; 1 : 0$





For Problem: For each additional output bit, the no. of comparators need to be doubled.
→ [Exponential]

Ex- for 8 bits, we need $2^8 = 256$ comparators

Hence, expensive af for higher resolution.

#4

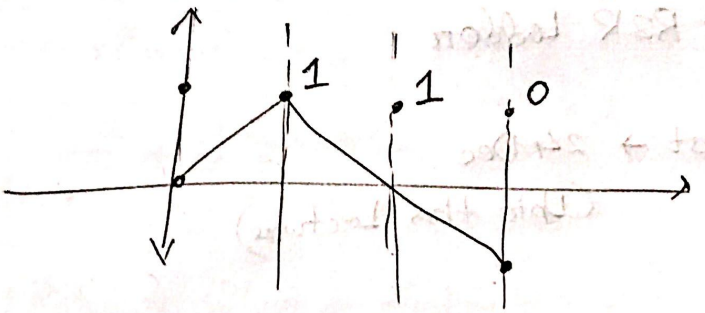
Analog to Digital Conversion

- Quantization
- Encoding

Delta-Sigma ADC

→ Sequential circuit: output is added to input; output emitted through lat

Advantage: Circuit design is not dependent on input size
Hence, can work with larger inputs.



(+) (i) High Resolution
(ii) No precision external components
(iii) High Accuracy

(-) (i) Slow due to oversampling.

SAR

Step-1

$V = V_{ref}/2$ → Compare → 0 if $V_{in} \geq V$
→ 1 if $V_{in} < V$

Step-2

$V = 1 \times \frac{V_{ref}}{2} + \frac{V_{ref}}{4}$ → compare → 0
→ 1

Step-3

$V = 1 \times \frac{V_{ref}}{2} + 0 \times \frac{V_{ref}}{4} + \frac{V_{ref}}{8}$ → compare

Adv → Good tradeoff between speed and accuracy

Flash (+) Very Fast (-) Expensive
~~(-) Less Accuracy~~

Delta-sigma (+) Cheap (-) Slow (oversampling)
(+) Very Accurate

SAR (+) Medium Fast
(+) Low Cost

Digital to Analog Converters

→ Weighted Sum
→ R2R Ladder

✱ Class Test → 24 Dec
(Upto this Lecture).

#5

Interfacing Data Transfer Models

13-Dec-21
Monday

- Maximum transferable units for motherboard → 8-bit/16-bit/32-bit/64-bit

(*) Models of data transfer / transmission → Parallel, Serial

→ Disadv of Parallel → i) (No. of pins of sender and receiver must match and is higher) → Very Expensive
ii) (Alternation of 1 bit needs whole set of bits to be resend.) → Limited to small distance
iii) Bits must be synchronized.

Use: Keyboard

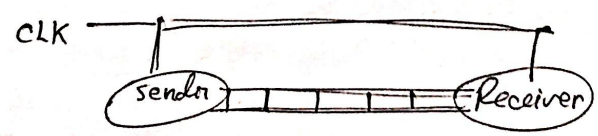
→ Serial Mode Disadv
→ Complicated as machine needs to decompose and reconstruct each bit.

Use: Printer, USB.

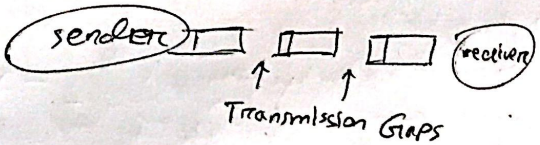
→ Cheaper but slower

(*) Serial Communication → i) Synchronous, ii) Asynchronous

Sync → i) Rate dependent on clock rate
ii) Less overhead



Async → i) Each byte is encoded (start - stop bits)



Ex:- Send 250 (8-bit) characters.

Async → $250 \times (8+2) = 250 \times 10 = 2500$ bits

↳ start and stop

Sync → Sync after 50 characters

No. of syncs $= \frac{250}{50} = 5$ syncs

$(250 + 5 \text{ sync char}) \times 8 = 2040$ bits

20-12-21

#6

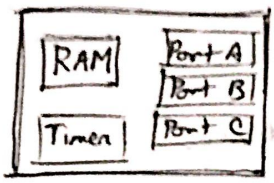
Lecture-6

Internal Interfacing → Interfacing with internal peripherals (ex - memory, built-in graphics)

PPI → Programmable Peripheral Interface (8155, 8255)

8155 → Has static RAM which acts as data buffer.

→ PORT C (6 bit) is used for handshaking



(*) Command Register Format (Imp.)

D0 - PA if { 0 input device is connected in Port-A, 1 output device is connected in Port-B }

D1 - PB

(For Port-C) D3 D2 ⇒ All Input (all 6 pins of port-C)

(0 0) → All Input

(0 1) → All Output

(1 0) → 3 OUT, 3 Port-A Handshaking / control signals

(1 1) → 3 Port-B handshaking, 3 Port-A handshaking

D4 → PA intrpt enable { 0 disable, 1 enable }

D5 → PB intrpt enable { 0 disable, 1 enable }

D7, D8 → for Timer

Memory Mapped I/O → I/O Read/Write operations directly occure on memory

[An address value of the 'µp' memory is stored in the interface's command register. For ex - 20H is stored. Then if any change is done to 21H, the mapped Port-A will be updated.

Ex - Given, 20H in command reg. { 21H → Port-A, 22H → (mapped) Port-B }

Handshaking

PC-(Port-C) Strobe, IBF (~~Interrupt~~ Buffer Full)
Input
PC2 PC1

INT → PC0

(Input) ~~Device sends~~ 8155 receives strobe signal from input device.
8155 sends IBF signal.

(Output) 8155 sends OBF signal.
8155 receives ACK from output device.

Class Test-1
24 Dec (Friday)
Lec - (2,3,4)