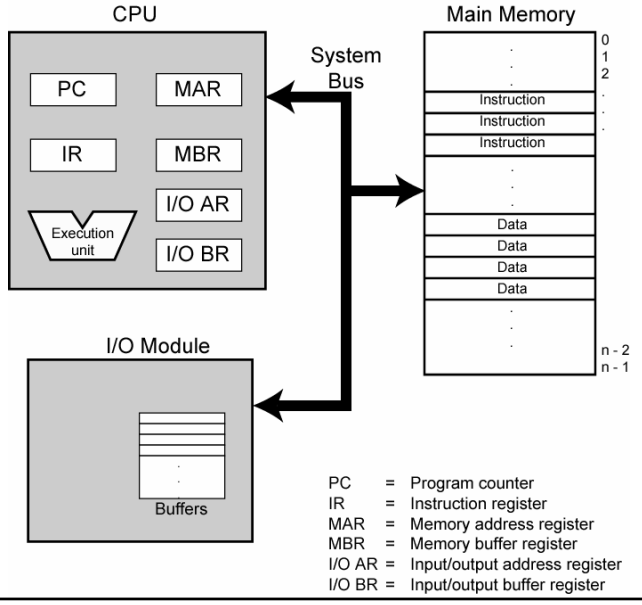
Mid Term Review

Lecture 10

Computer Components: Top Level View



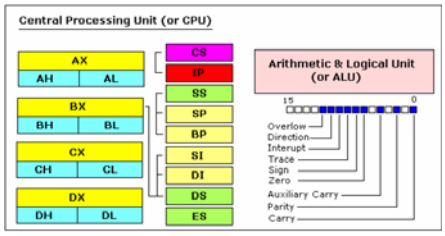
2

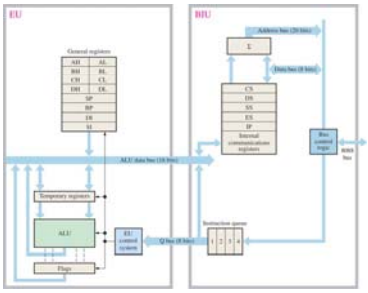
8086 Architecture

- The microprocessors functions as the CPU in the stored program model of the digital computer.
- The microprocessor also has a S/W function.
  - It must recognize, decode, and execute program instructions fetched from the memory unit.
    - Fetch – Decode – Execute – Store Cycle
  - This requires an Arithmetic-Logic Unit (ALU) within the CPU to perform arithmetic and logical (AND, OR, NOT, compare, etc) functions.
- The 8088/ 8086 CPUs are organized as two processing units, called the
  - Bus Interface Unit (BIU) and the Execution Unit (EU).
    - Each unit has a dedicated function and both operate at the same time

3

8086 Microprocessor



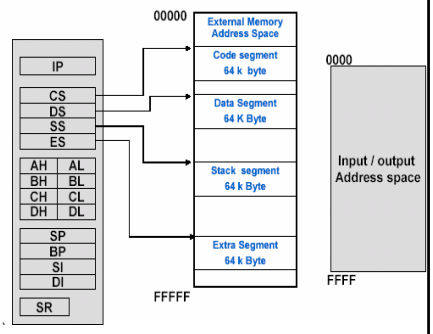


- 8086 is a 16 bit microprocessor
  - ALU
  - Internal registers and
  - Most of the instructions are of 16 bit binary word
- It has a 16 bit data bus- read or write 16 bits at a time to memory or to port (8bit bus for 8088)
- It has 20 bit address bus- can address 1 Mbyte of memory locations

4

Segment Registers

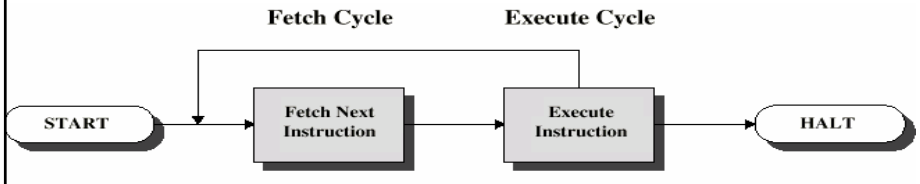
- These are unique to x86 microprocessor family, and are designed because
  - All Index and pointer registers are of 16 bit wide, and the memory is of 20-bit (1 M)
  - Requires 20 bit address
- Index and pointer registers are not wide enough to address directly to any memory location
  - To resolve memory is segmented
- Segment of memory
  - A block of 64 Kbytes of memory addressed by a special register called a “segment register”



5

Instruction fetch and execute

- At the beginning of each instruction cycle
  - Processor fetches an instruction from the memory
    - “PC”- program counter holds the address of the instruction to be fetched next
    - Processor always increment the PC after each instruction unless told otherwise
  - The fetched instruction is loaded into a register located in the processor – “ the instruction Register (IR)”
  - The instruction contains bits – to specify the action the processor is to take



6

Fetch Execute - Cycle

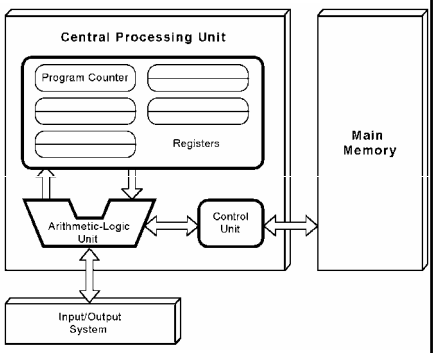
- **FETCH**
- Program Counter (PC) holds address of next instruction to fetch
- Processor fetches instruction from memory location pointed to by PC
- Increment PC
  - Unless told otherwise
- Instruction loaded into Instruction Register (IR)
- Processor interprets instruction and performs required actions

- **EXECUTE**
- Processor-memory
  - data transfer between CPU and main memory
- Processor I/O
  - Data transfer between CPU and I/O module
- Data processing
  - Some arithmetic or logical operation on data
- Control
  - An instruction may specifies the alteration of sequence of operations
    - e.g. jump

7

The von Neumann Model

- The computers employ a fetch-decode-execute cycle to run programs as follows:
  - The control unit fetches the next instruction from memory using the program counter to determine where the instruction is located
  - The instruction is decoded into a language that the ALU can understand.
  - Any data operands required to execute the instruction are fetched from memory and placed into registers within the CPU
  - The ALU executes the instruction and places results in registers or memory

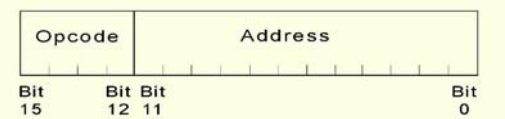


8

Example of Program Execution

Example

Add the contents of the memory at address 940 to the contents of the memory at address 941 and store the result in the next location



Internal CPU registers:

Program counter (PC): Address of next instruction

Instruction Register (IR): Current instruction

Accumulator (AC): Temporary Storage

Partial List of Opcodes:

0001 = (1h) = Load AC from Memory

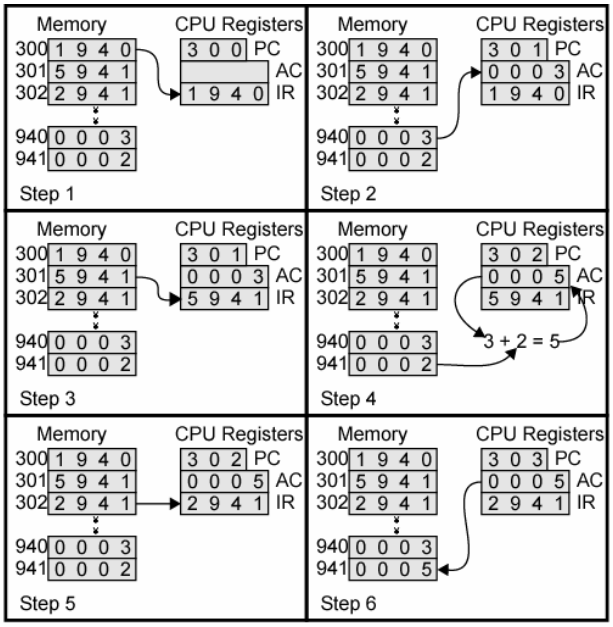
0010 = (2h) = Store AC to memory

0101 = (5h) = Add to AC from Memory

9

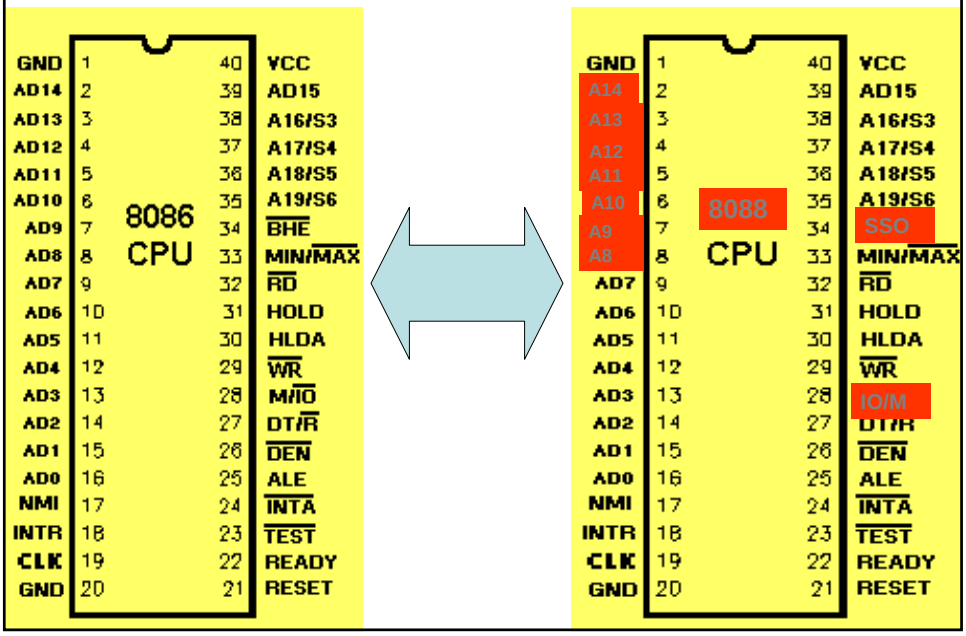
Add the contents of the memory at address 940 to the contents of the memory at address 941 and store the result in the next location

0001 = (1h) = Load AC from Memory
0010 = (2h) = Store AC to memory
0101 = (5h) = Add to AC from Memory



10

Pin layout 8086/8088



Clock Generator (8284A)

- 8086 requires clock signals from an external clock, to synchronize the internal operations, it provide fol main functions:
  - Clock Generation
  - RESET synchronization
  - READY synchronization
  - Peripheral Clock signal
    - CLK pin provide clock input signal to 8086 and other components
    - PCLK: provide clock signal to peripheral eqpt in the system
- 8284A has a 14.7456 MHz crystal connected to it
  - The clock signal is produced by Freq of the crystal divided by three (14.7456/3)
  - The actual freq for 8086 is 4.915 MHz
  - The PCLK freq is half of the clock freq so it is 2.45 MHz

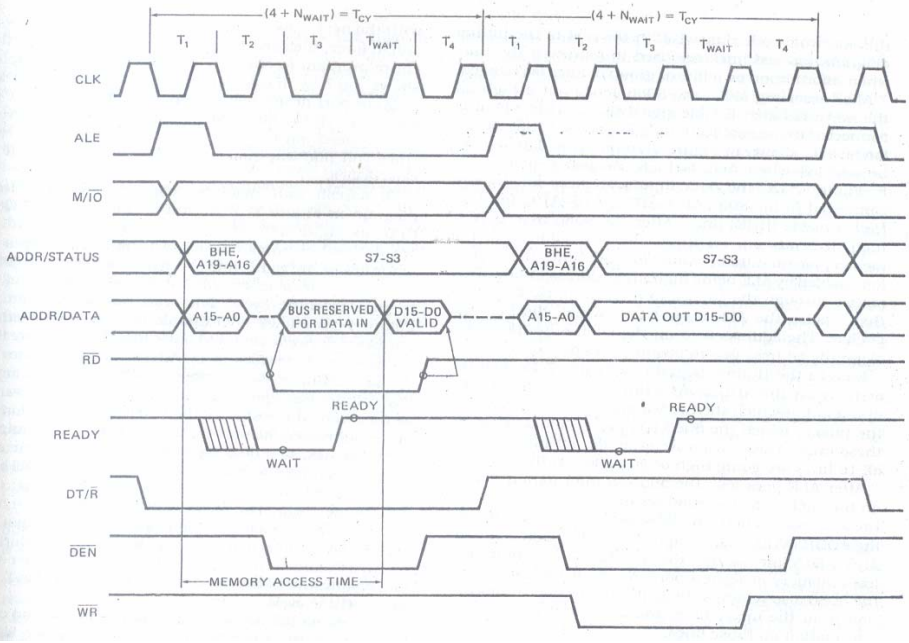
12

Timing

- One clock cycle is called a “State”
  - Measured from falling edge of one clock pulse to the falling edge of next clock pulse
  - Time for a state is determined by the frequency of the clock signal
- Basic microprocessor operation is called a “ Machine Cycle”
  - Machine cycle consists of several states
- The time a microprocessor requires to fetch and execute an entire instruction is referred as “instruction cycle”
  - Consists of one or more machine cycle

13

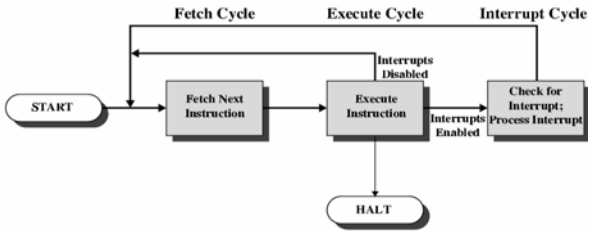
Basic 8086 Timings



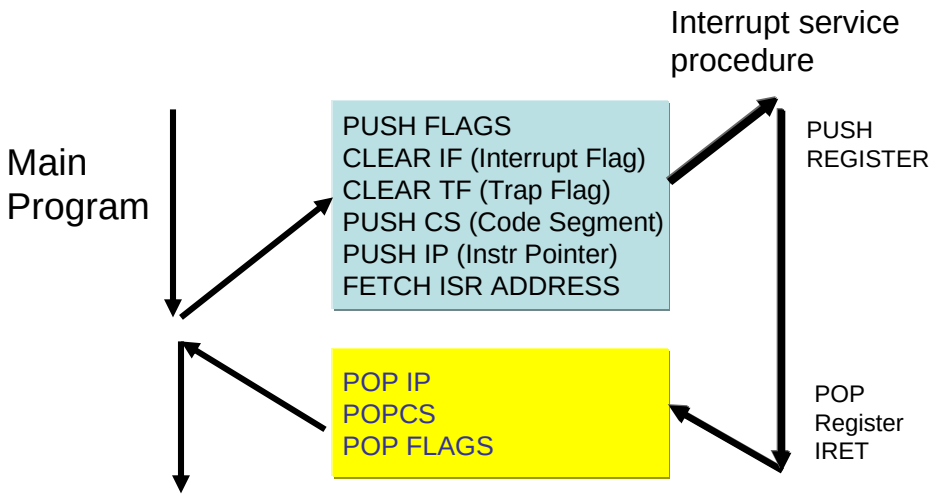
14

Interrupts

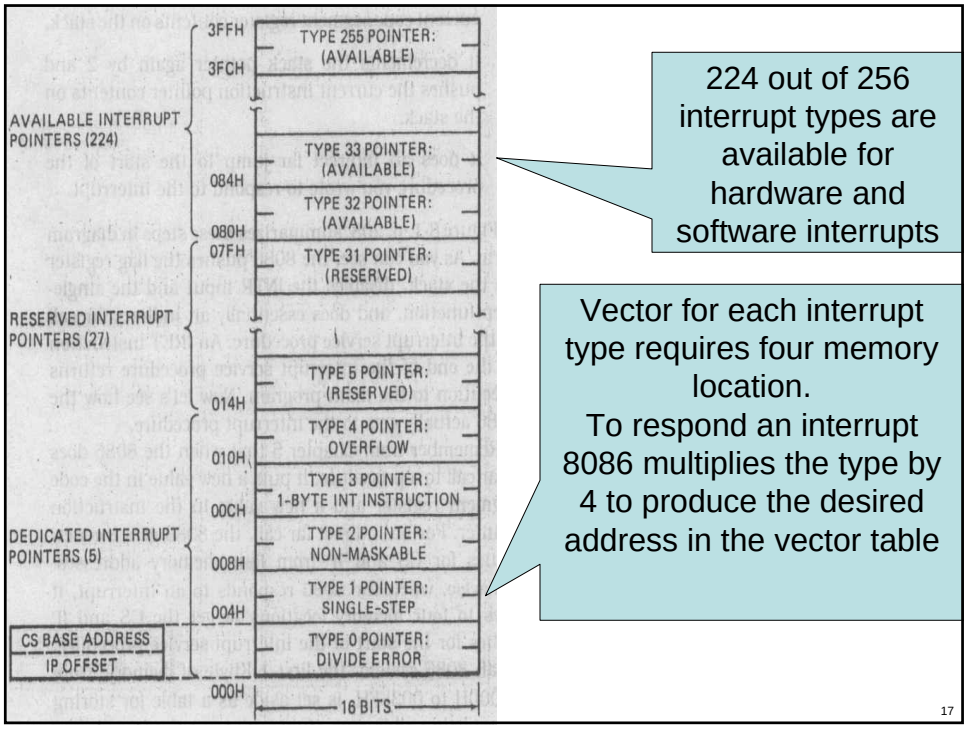
- All computers provide the mechanism by which other modules (e.g. I/O) may interrupt normal sequence of processing
- Most common classes of interrupts are:
  - Program: generated by some condition e.g. overflow, division by zero
  - Timer: Generated by internal processor timer, allow Operating system to perform certain functions at regular intervals
  - I/O: from I/O controller
  - Hardware failure: generated by a failure such as power



15



16

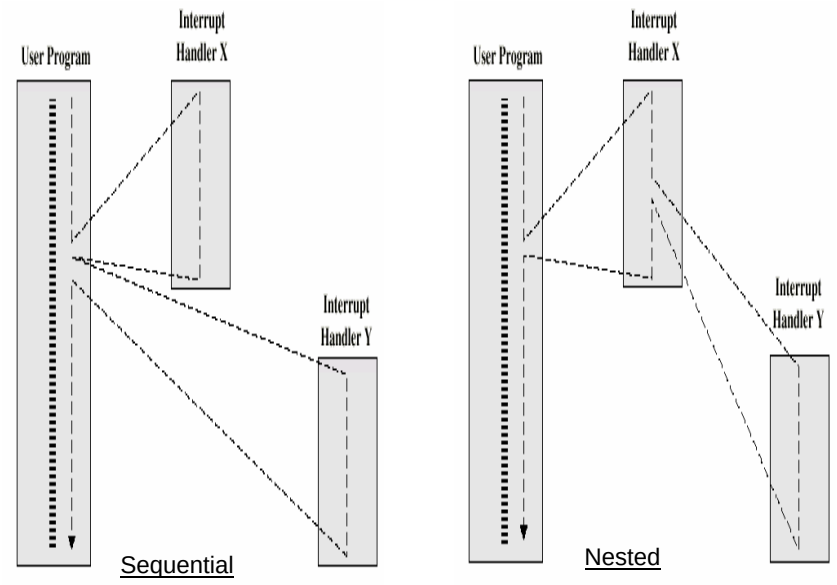


Multiple interrupts

- A typical system can support several to several dozen interrupts
  - How should the system respond if more than 1 interrupt occurs at the same time?
    - Systems prioritize the various interrupts
    - At the start of the interrupt cycle, the highest priority pending interrupt will be serviced
    - Remaining interrupt requests will be serviced in turn
- What if an interrupt occurs while an ISR is being executed (a result of a previous interrupt)
  - Ignore the second interrupt (by disabling interrupts) until the ISR completes
  - Recognize and service the interrupt only if it has a higher priority than the one

18

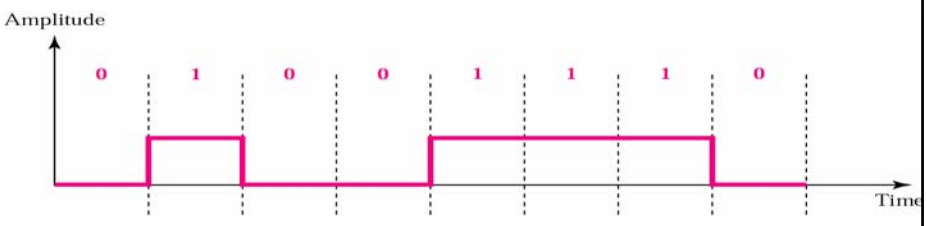
Multiple Interrupts



19

Encoding Scheme

- Unipolar: simplest, inexpensive to implement but obsolete in use
  - It provides the concept of encoding system
  - Most encoding scheme uses to send one voltage level for zero, and another for one
  - The polarity of the pulse decides whether it is posative or negative
- Unipolar scheme uses only one polarity, that is assigned to one of the two binary states, normally the 1
  - The other state is zero voltage

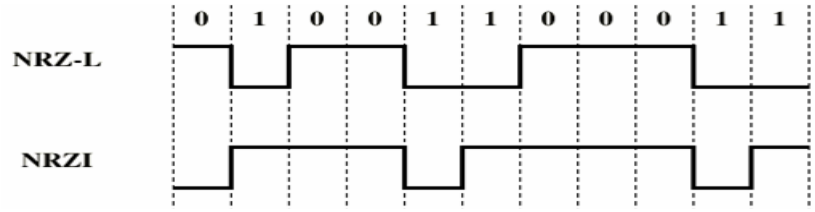


•presence of dc component (average amplitude is non zero
•Lack of synchronization (if data contains long 1's or 0's)

20

Polar: Nonreturn to Zero-Level & NRZ Invert (NRZ-L & NRZ-I)

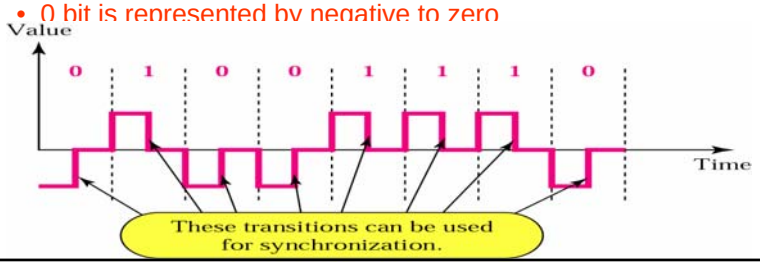
- Two different voltages (+ and – OR + and none) for 0 and 1 bits
  - Voltage constant during bit interval
    - no transition I.e. no return to zero voltage
    - e.g. Absence of voltage for zero, constant positive voltage for one
  - More often, negative voltage for one value and positive for the other



21

Polar: Return to Zero- RZ

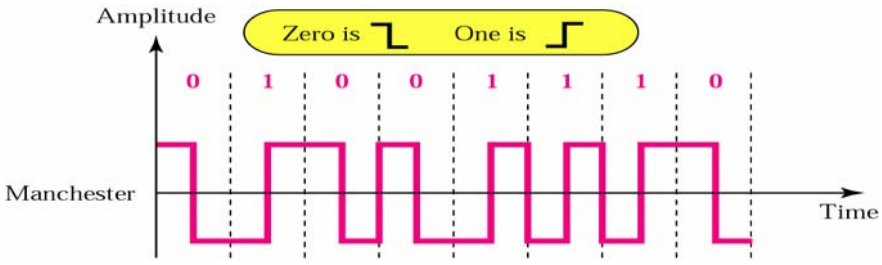
- With Return to Zero (RZ) encoding, the three values to the signal can be assigned
- With RZ signal changes not between bits but during each bit
  - Like NRZ-L, a positive voltage means a 1 and a negative voltage means 0
  - With RZ , half way through each bit interval, the signal return to zero
    - 1 bit is represented by positive to zero
    - 0 bit is represented by negative to zero



22

Polar: Manchester encoding

- It uses an inversion at the middle of each bit interval for synchronization and bit representation
  - A negative to positive transition represents binary 1
  - A positive to negative transition represents binary 0
- By using the signal transition for dual purpose, this encoding scheme has the same level of synchronization as of RZ, but with two levels of amplitude



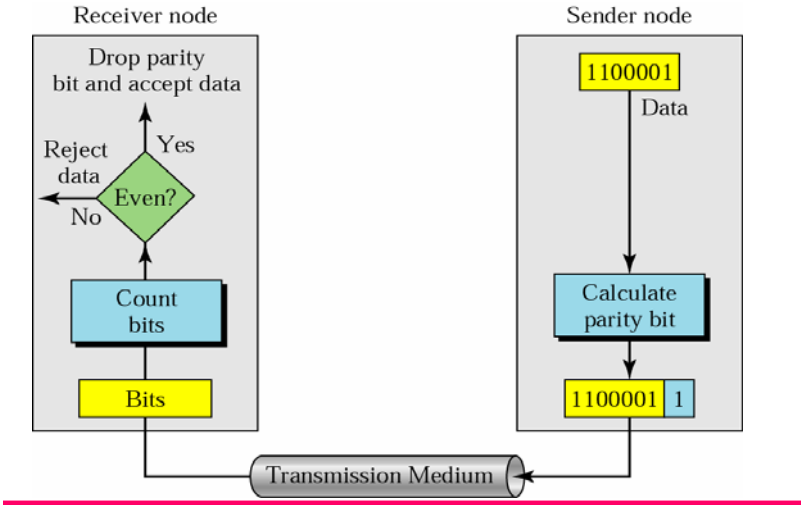
23

Error Detection and Correction

- It is physically impossible for any data recording or transmission medium to be 100% perfect 100% of the time over its entire expected useful life.
  - ***Data can be corrupted during transmission. For reliable communication, errors must be detected and corrected.***
- As more bits are packed onto a square centimeter of disk storage, as communications transmission speeds increase, the likelihood of error increases.
- Thus, error detection and correction is critical to accurate data transmission, storage and retrieval.
  - **Single-Bit Error**
  - **Burst Error**

24

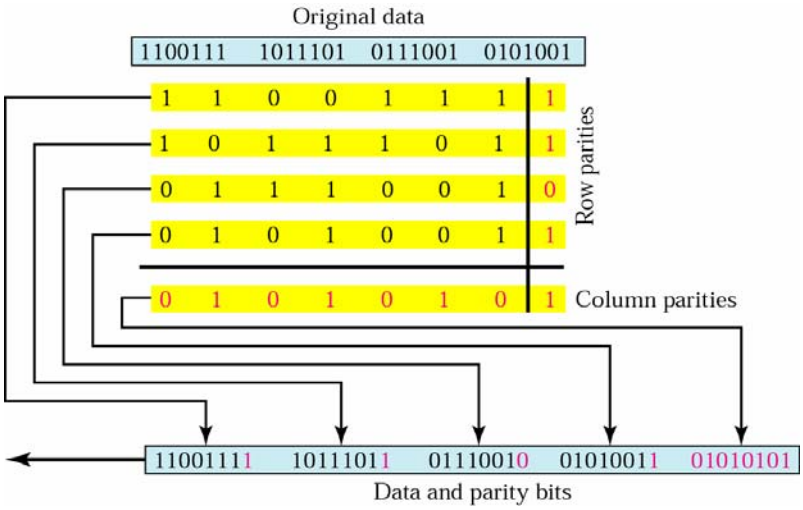
Even-parity concept



In parity check, a parity bit is added to every data unit so that the total number of 1s is even (or odd for odd-parity).

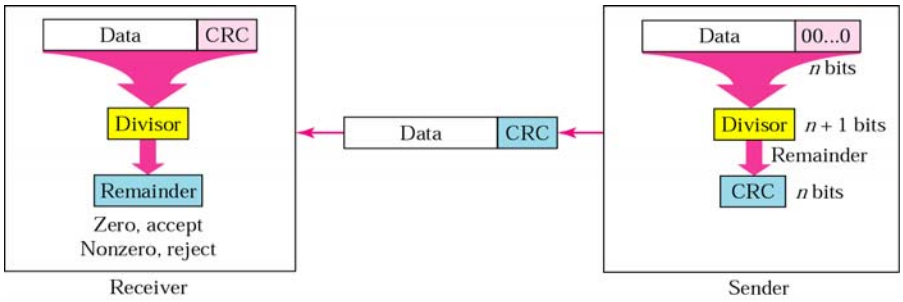
25

Two-dimensional parity



26

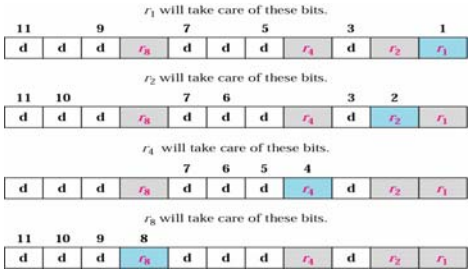
CRC generator and checker



27

Error Detection and Correction

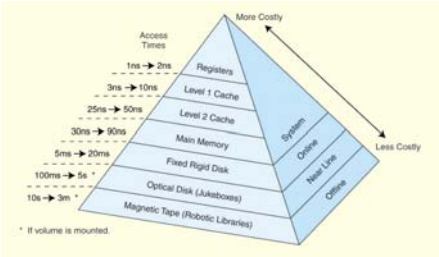
- Hamming codes:
  - Used in situations where random errors are likely to occur
    - use parity bits, also called *check bits* or *redundant bits*
    - The memory word itself consists of *m* bits, but *r* redundant bits are added to allow for error detection and/or correction
    - The final word, called a *code word*, is an *n*-bit unit containing *m* data bits and *r* check bits.
      - Code word = N bits = m bits + r bits
  - The number of bit positions in which two code words differ is called the *Hamming distance* of those two code words.



28

Types of Memory

- There are two kinds of main memory:
  - *Random Access Memory, RAM, and Read-only-Memory, ROM*.
- There are two types of RAM:
  - Dynamic RAM (DRAM) and Static RAM (SRAM).
    - Dynamic RAM consists of capacitors that slowly leak their charge over time. Thus they must be refreshed every few milliseconds to prevent data loss.
  - SRAM consists of circuits similar to the D flip-flop
    - SRAM is very fast memory and it doesn't need to be refreshed like DRAM does. It is used to build cache memory



29

Memory organization

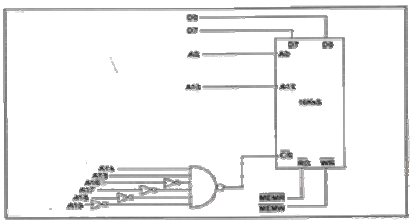
- Total number of bits that a memory chip can store is equal to the number of locations times the number of data bits per location.
- To summarize:

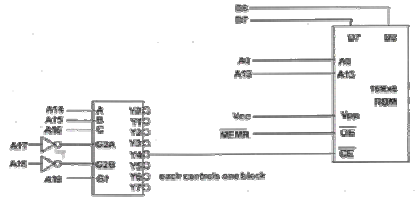
1. Each memory chip contains 2^x locations, where *x is* the number of address pins on the chip.
2. Each location contains *y* bits, where *y* is the number of data pins on the chip.
3. The entire chip will contain $2^x \times y$ bits, where *x* is the number of address pins a*nd y* is the number of data pins on the chip.

30

Simple logic gate as address decoder

- The simplest method of decoding circuitry is the use of NAND or other gates.
  - The fact that the output of the NAND gate is active low and that CS is also active low makes them a perfect match.
- In cases where the CS input is active high, an AND gate must be used.
- Using a combination of NAND and inverters, one can decode any address range.





31

Memory Organization

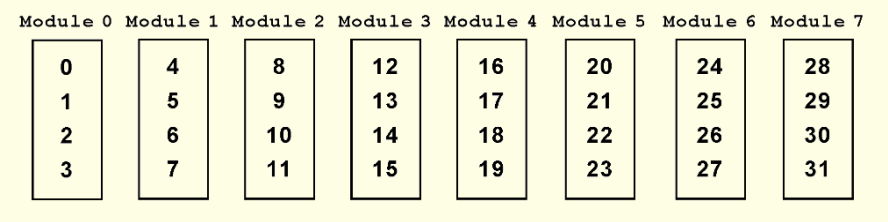
- Physical memory usually consists of more than one RAM chip.
  - Access is more efficient when memory is organized into banks of chips with the addresses interleaved across the chips
- With low-order interleaving, the low order bits of the address specify which memory bank contains the address of interest.

Module 0	Module 1	Module 2	Module 3	Module 4	Module 5	Module 6	Module 7
0	1	2	3	4	5	6	7
8	9	10	11	12	13	14	15
16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31

32

Memory Organization

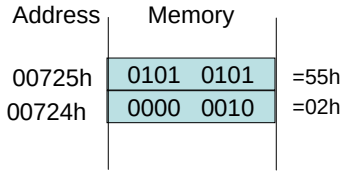
- With high-order interleaving, the high order address bits specify the memory bank.



33

Storing a word of data

- To permit efficient use of memory
  - The word of data (16 bits) can be stored at even or odd addressed word boundaries
  - The least significant bit of the address determine the type of word boundary
    - If the bit is 0, the word is at an even address boundary corresponds to two consecutive byte located at even address



Collectively represents the word =5502h or

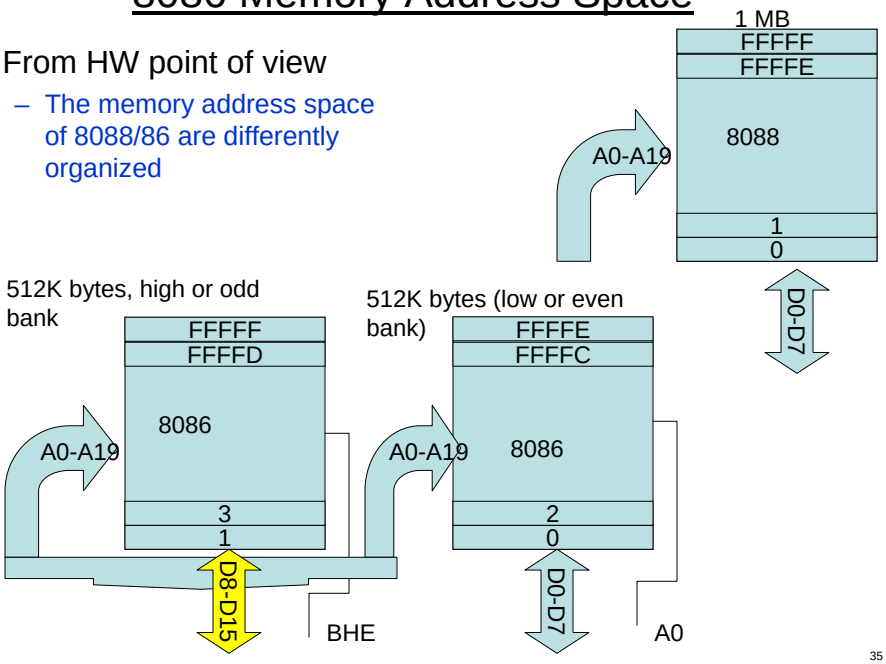
= (**0101010100000010**)**2**

The Least Significant byte of the word is stored at **00724**h, so it an even address boundary

34

8086 Memory Address Space

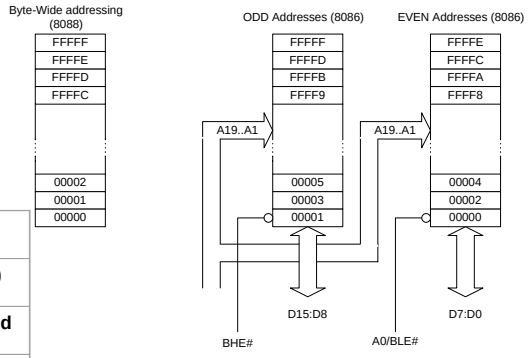
- From HW point of view
  - The memory address space of 8088/86 are differently organized



35

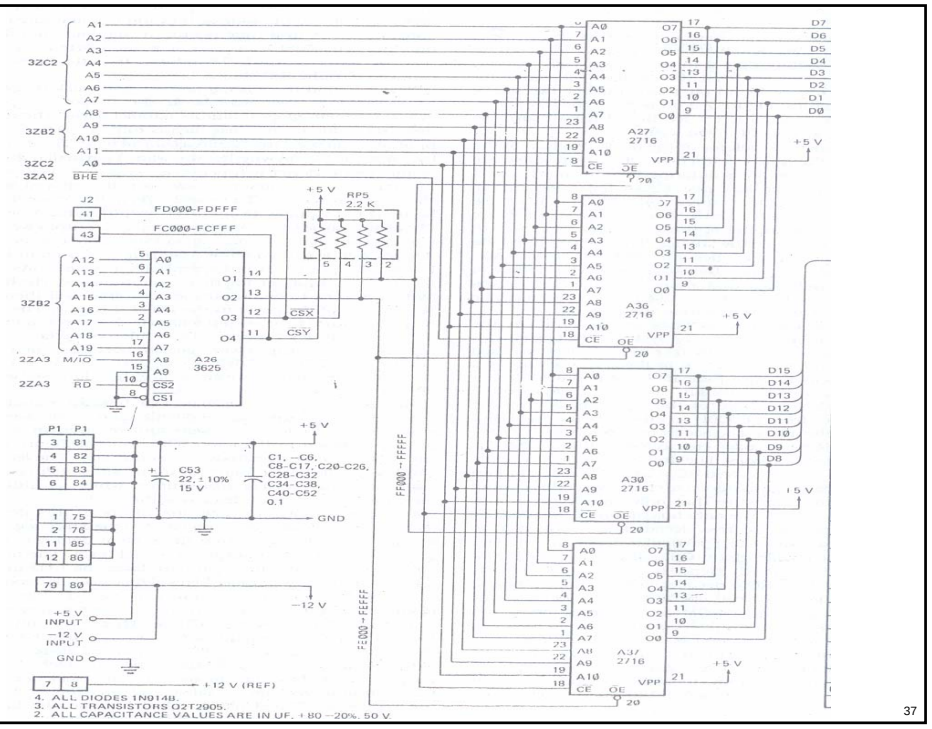
BHE Signal (8086 Only)

- The 8086 processor can address memory a byte at a time
  - Its data bus is 16b wide
  - It uses the BHE# signal and A0 (sometimes called BLE#) to address bytes using its 16b bus
- Use of BHE#/A0



BHE	A0	Selection
0	0	Whole word (16-bits)
0	1	High byte to/from odd address
1	0	Low byte to/from even address
1	1	No selection

36



37

PROM Decoder truth table

PROM INPUT				PROM OUTPUT				PROM Address Block Selected
M/IO	A14-A19	A13	A12	04	03	02	01	
1	1 1 1 1	1	1	1	1	1	0	FF000H-FFFFFH
1		1	0	1	1	0	1	FE000H-FEFFFH
1		0	1	1	0	1	1	FD000H-FDFFFH
1		0	0	0	1	1	1	FC000H-FCFFFH
All Other States				1	1	1	1	NONE

38

Quiz # 2

15 Mins

10 Marks

Quiz # 2

- Time 15 Mins → Marks 10
- Consider following word addressable memories, find out the address and data lines requires to access word length of data.
  - 2K × 16
  - 4M × 16
- A massage 1101 was sent as hamming code and received as 1111.
  - Construct the hamming code word to be sent
  - Find out the error and using the hamming code technique correct the error

40