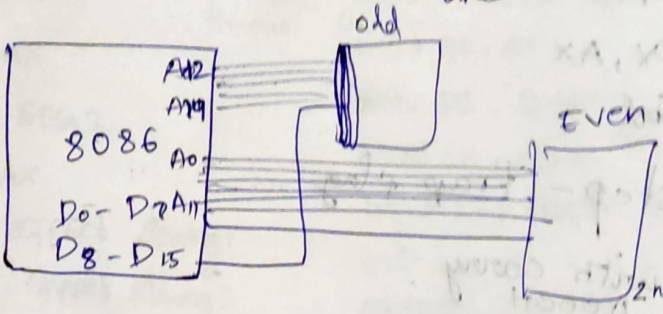
UNIT - III

Memory Interfacing.

4 KB – $A_0, A_1, \ldots A_{11}$ → 12 address lines

$2^{10} \times 2^2 = 2^{12}$ → 12 address lines



RAM → $\overline{MRD}$, $\overline{MWR}$

ROM → $\overline{MRD}$

1. arrange memory chips
2. req address lines
3. $D_0 - D_7$ → Even
   $D_8 - D_{15}$ → Odd
4. Along

RAM → FFFF0 to FFFFF 8 KB

Interface two 4 KB ROM and two 4 KB - RAM with suitable memory map.

2 → 4 KB → 8 KB → 2^{13}

	A_{19}	A_{18}	A_{17}	A_{16}	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
FE000 ROM	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
FFFFF RAM	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

FFFFF
FE000 – 1
00001
FDFFF

Semi Conductor Memory Interfacing

Static RAM Interfacing.

1. Arrange the available memory chips. in order to obtain 16 bit data lines
2. Connect available address lines and data lines (4 KB = 2^{12} → 12 lines)
3. Remaining address A_0 & $\overline{BHE}$, A_{13} to A_{19}. to generate Chip Select.

13 4 4 2 1 1 1

4K x 8 = 4KB

1. Interface two 4K X 8 EPROM & two 4K x 8 RAM chips with 8086. Select suitable map.

We must know that after reset IP & CS are initialized to form address FFFFFH hence this address must lie in EPROM. Address of RAM may be selected anywhere in the 1MB of 8086.

$8KB = 2^{13}$

EPROM

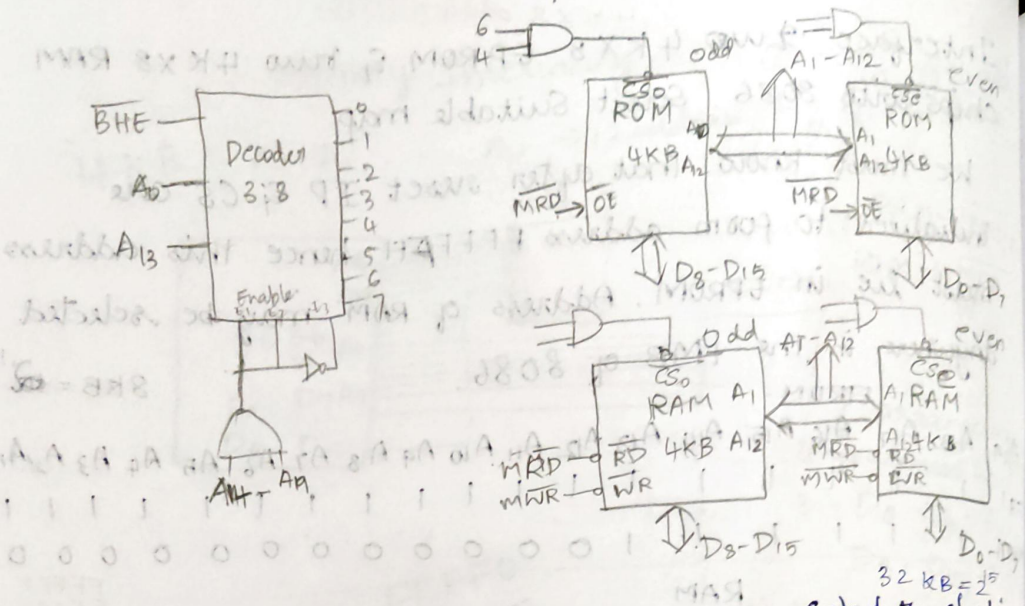
	A_{19}	A_{18}	A_{17}	A_{16}	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
FFFFH	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
E000H	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0

FFFFF
FE000
01FFF

RAM

	A_{19}	A_{18}	A_{17}	A_{16}	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
DFFFH	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1
C000H	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Decoder I/P →	A_2	A_1	A_0	Selection Command
Address/$\overline{BHE}$ →	A_{13}	A_0	$\overline{BHE}$	
Word transfer on D_0-D_{15}	0	0	0	Even and odd address in RAM
Byte transfer on D_0-D_7	0	0	1	only even in RAM
D_8-D_{15}	0	1	0	only odd in RAM
Word transfer on D_0-D_{15}	1	0	0	Even and odd address in EPROM
Byte transfer on D_0-D_7	1	0	1	only even in EPROM
D_8-D_{15}	1	1	0	only odd in EPROM.

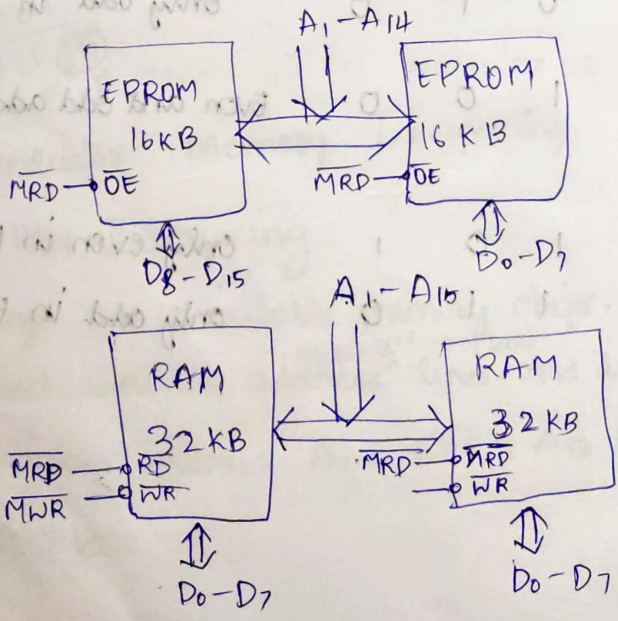


2 chips 16K × 8 and two 32K × 8. Select the starting address of EPROM

The RAM address of RAM is 00000H

16 KB = 2^{14}
= 2^{10}

Address	A_{19}	A_{18}	A_{17}	A_{16}	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1
FFFFFH	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
EPROM																			
F8000H	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RAM																			
00000H	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



It is required to interface 2 chips of 32K x 8 4 chips of 32K x 8 RAM according to the following map.

ROM 1 and 2 F0000H - FFFFFH

RAM 1 and 2 D0000H - DFFFFH

RAM 3 and 4 C0000H - EFFFFH

$32KB = 2^{5+10}$

$= 2$

Start MOV AL, 00H (sqre wave)

OUT C8, AL

CALL 1010

MOV AL, FF

OUT C8, AL

CALL delay

JMP Start

DELAY MOV CX, 05FF

LOOP L1

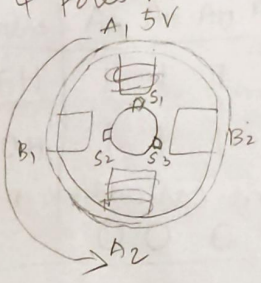
RET

mode
i/p port A
o/p port C

LOOP → Dec CX
↳ JNZ

STEPPER MOTOR INTERFACING (the movement will be step wise)

4 poles in stator, 3 pairs of rotors



Wave scheme (one coil is energized)
full step (2 coils are energized)
half step (mixed)

A_1	A_2	B_1	B_2	
1	0	0	0	→ A_1 is locked with S_1
1	0	1	0	→ B_1 is locked with S_2
0	1	0	0	→ A_2 is locked with S_3
0	0	0	1	→ B_2 is locked with S_1

full step:
adjacent coils are energised at a time

half step (mixed scheme)

A_1
A_1, B_1
B_1
B_1, A_2
A_2
A_2, B_2
B_2
B_2, A_1

$$\theta = \frac{360}{N_S \times N_R} = \frac{360}{4 \times 3} = 30^\circ$$

No of stator core N_S.
No of pairs of rotor N_R

8051 → motor interface

8254 (PIT) Programmable interface Timer
P0, P1

Features

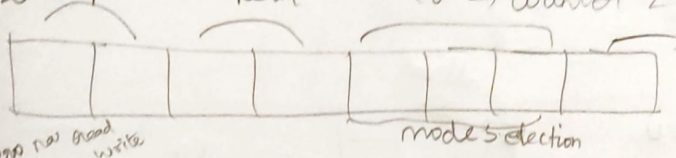
$2^{16} = 2^{10} \times 2^6$
$= 2$

→ all the counters are 16-bit
→ Clock input up to 10MHz
→ 4 internal address

binary → 0FFFFF
BCD → 0F9999

Control word format

00 → counter 0 is selected
01 → Counter 1 is selected
10 → Counter 2 is selected



6 modes → 3 bit

RD → zero no read
WR → zero no write

mode 0 → interrupt terminal count (10) → control word
mode 1 → Hardware Retriggerable one shot / mono shot
mode 2 → Rate generator (divide by N counter)
mode 3 → Square wave mode
mode 4 → Software triggered strobe
mode 5 → Hardware triggered strobe

for every N counter o/p pulse is generated

00 11 011

mode 0 → interrupt terminal count → control word (10)

mode 3 → if counter is even
(decrement by 2)
if odd → (decrement by 1) then decrement by 2 then ↓ 2

A0 → cannot be used

8279 (Special purpose
keyboard & display → 16 Byte
Segment Segment
debouncing
↓
delay

8251 USART (Universal Synchronous Asynchronous Receiver Transmitter).

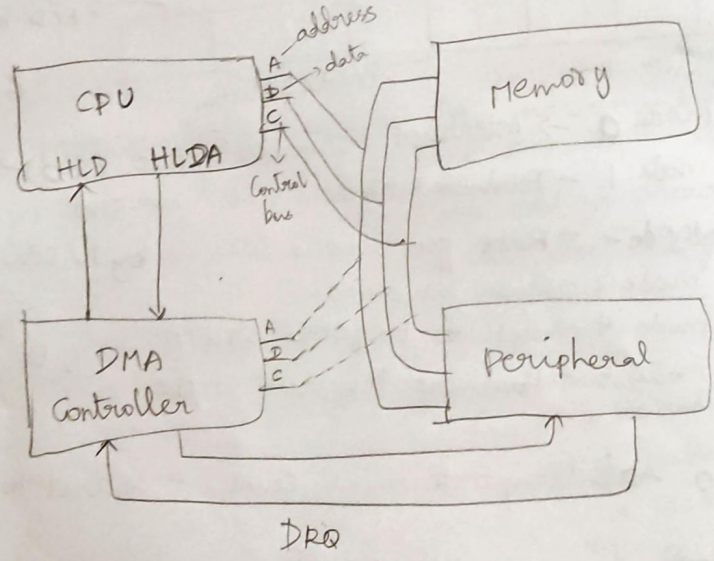
'framing'
Start bit stop bit

Baud rate. → $\overline{TX_c}$ $\overline{RX_c}$

8257 (DMA)

master → processor to peripheral
slave → peripheral to processor

Intel 4 channel each channel 2 register
total 10 register
to select 1 in 10 register 4 bits
address (16 bit register)
Counter register → 14 bits used
$2^{14} = 2^{10} + 2^4$
$= 16KB$
4 channels => 64KB can be transmitted



→ DMA mode of data transfer is the fastest among all the modes of data transfer
→ In this device may transfer data directly to memory or from memory without any interference
→ Address bus, data bus, Control bus so that the device may transfer data directly to or from memory
→ The DMA data transfer is initiated only after receiving HLDA signal from CPU

1. Design a programmable timer using 8254 and 8086 interface 8254 at an address 0040H for Counter 0. and write the following. ALPs.

8086 & 8254 run at 10MHz & 5MHz respectively

1. To generate square wave of period 1ms.
2. To interrupt the processor after 10 ms.
3. To drive a monoshot pulse with quasi state duration.

0 0 4 0
0000 0000 0100 0000

Counter 0 - 0040
1 - 0042
2 - 0042
CWR - 0046