BITS Pilani

Microprocessors & Interfacing

Memory Interface

Dr. Gargi Prabhu
Department of CS & IS

Address Decoding

- In order to attach a memory device to the microprocessor, it is necessary to decode the address sent from the microprocessor.

- Why decode memory?

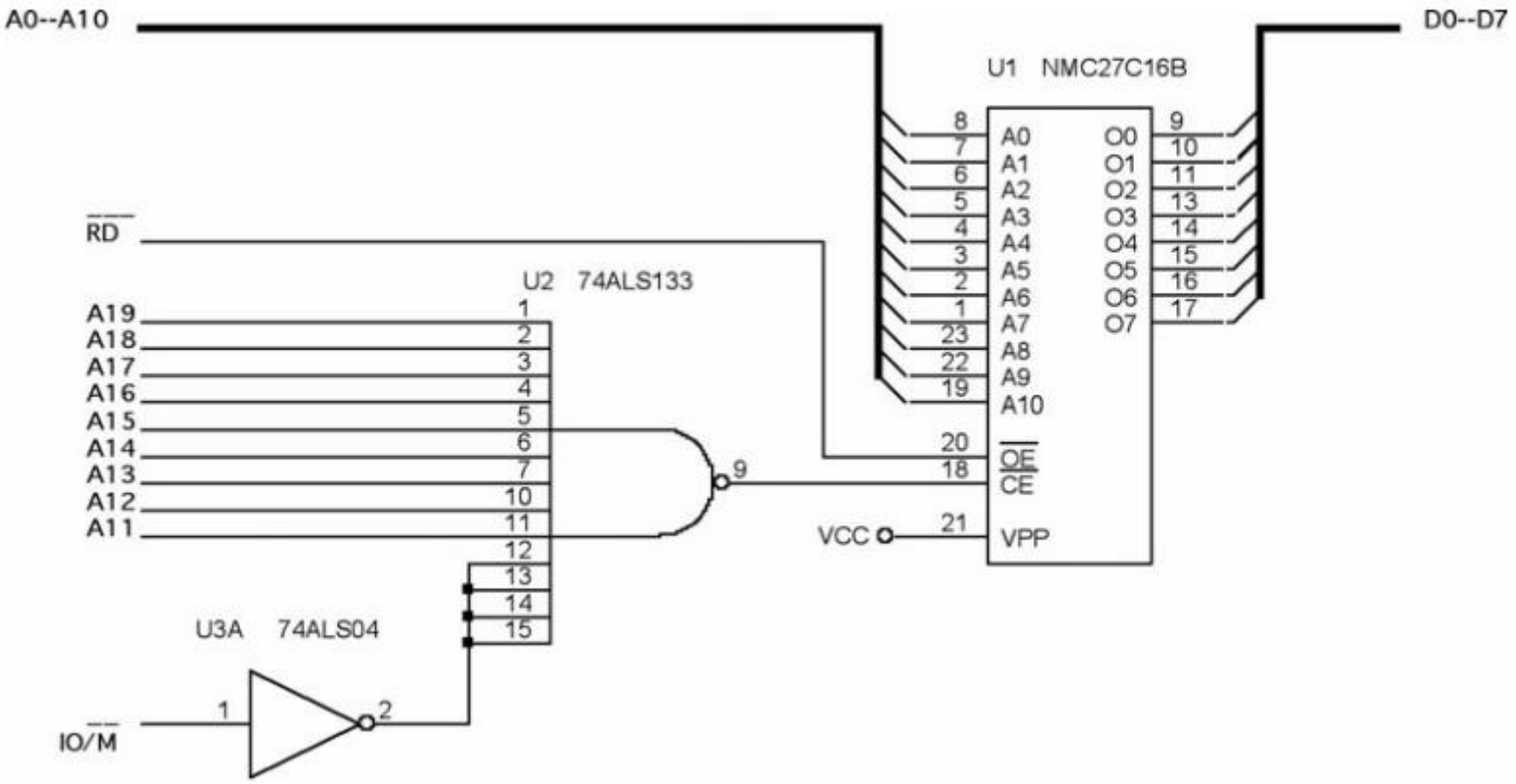
Simple NAND Gate Decoder

- When the 2K × 8 EPROM is used, address connections A10–A0 of the 8088 are connected to address inputs A10–A0 of the EPROM.
- The remaining nine address pins (A19–A11) are connected to the inputs of a NAND gate decoder.

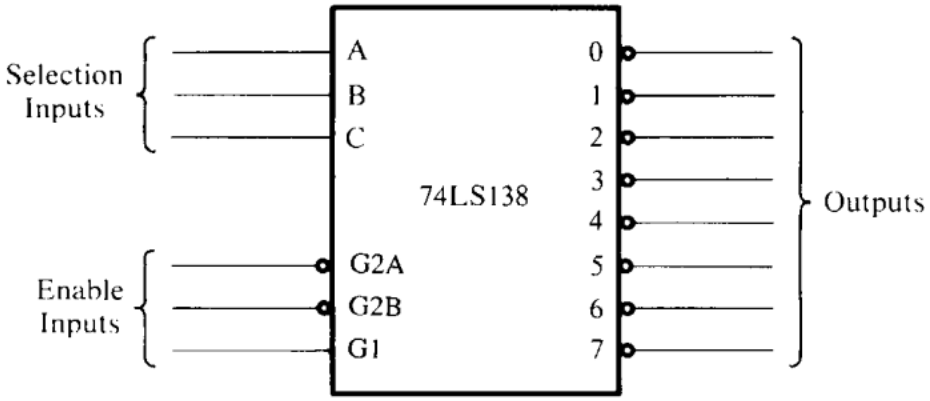
Example

```
1111 1111 1XXX XXXX XXXX

          or

1111 1111 1000 0000 0000 = FF800H
          to
1111 1111 1111 1111 1111 = FFFFFH
```

Simple NAND Gate Decoder

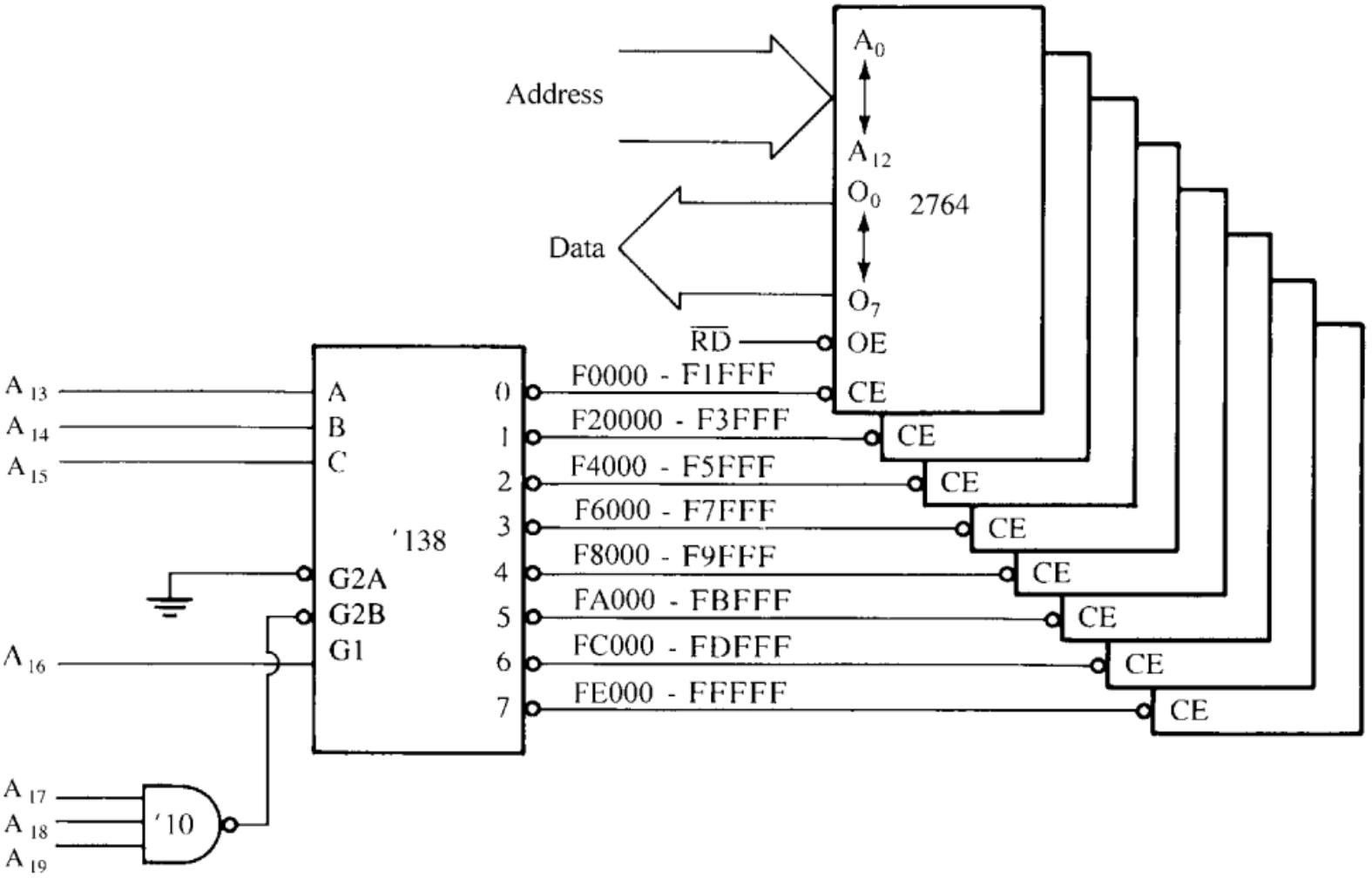


3 to 8 line Decoder (74LS138)



Inputs						Outputs							
Enable			Select										
$\overline{G2A}$	$\overline{G2B}$	G1	C	B	A	$\overline{0}$	$\overline{1}$	$\overline{2}$	$\overline{3}$	$\overline{4}$	$\overline{5}$	$\overline{6}$	7
1	X	X	X	X	X	1	1	1	1	1	1	1	1
X	1	X	X	X	X	1	1	1	1	1	1	1	1
X	X	0	X	X	X	1	1	1	1	1	1	1	1
0	0	1	0	0	0	0	1	1	1	1	1	1	1
0	0	1	0	0	1	1	0	1	1	1	1	1	1
0	0	1	0	1	0	1	1	0	1	1	1	1	1
0	0	1	0	1	1	1	1	1	0	1	1	1	1
0	0	1	1	0	0	1	1	1	1	0	1	1	1
0	0	1	1	0	1	1	1	1	1	1	0	1	1
0	0	1	1	1	0	1	1	1	1	1	1	0	1
0	0	1	1	1	1	1	1	1	1	1	1	1	0

2764 EPROMs for a 64K × 8 section of memory in an 8088 microprocessor-based system



Address Ranges: Example 1

```
1111 XXXX XXXX XXXX XXXX

          or

1111 0000 0000 0000 0000 = FOOOOH
          to
1111 1111 1111 1111 1111 = FFFFFH
```

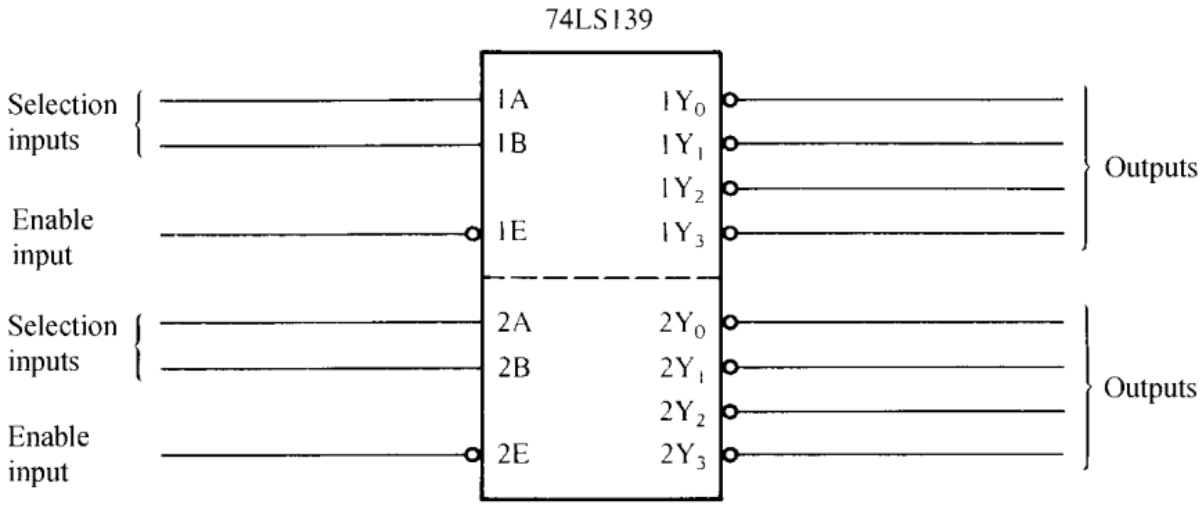
Address Ranges: Example 2

```
      CBA
1111 OOOX XXXX XXXX XXXX

          or

1111 0000 0000 0000 0000 = FOOOOH
          to
1111 0001 1111 1111 1111 = F1FFFH
```

Address Ranges: Example 3

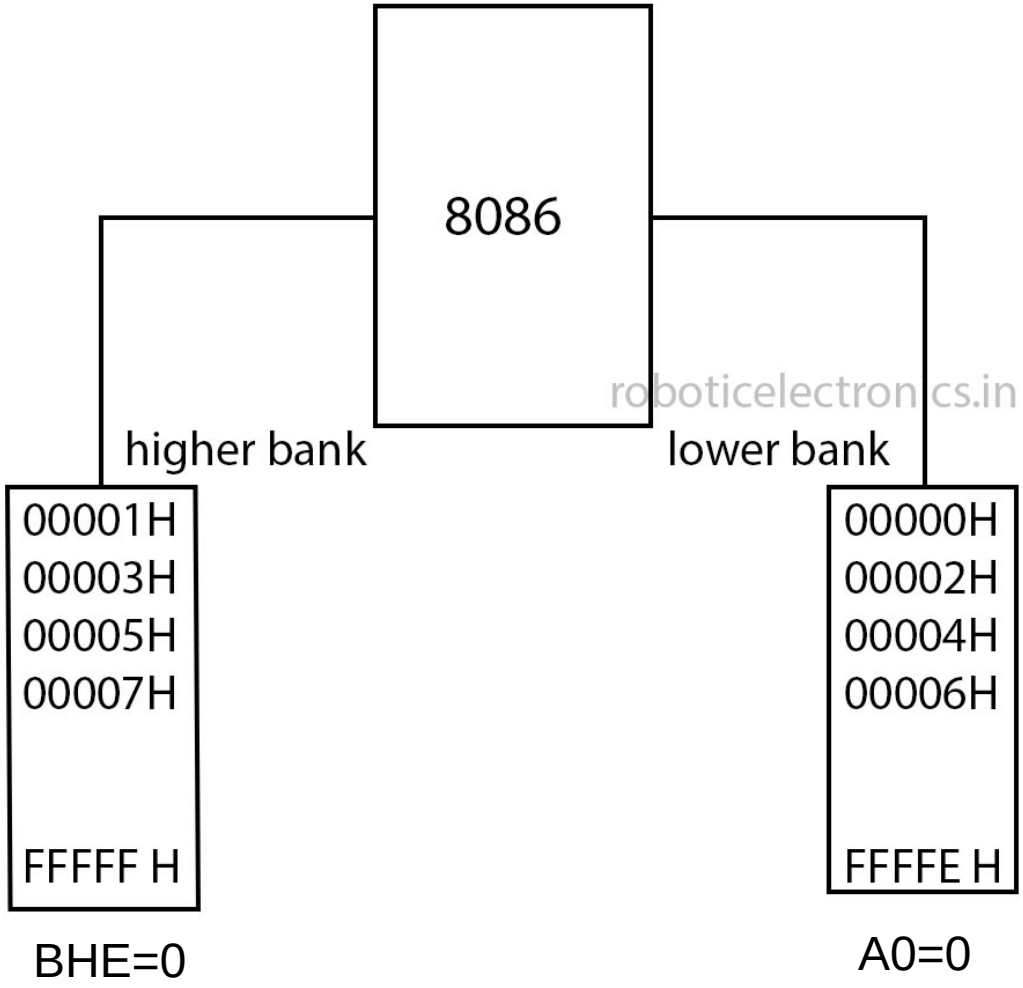
```
      CBA
1111 001x xxxx xxxx xxxx

          or

1111 0010 0000 0000 0000 = F2000H
          to
1111 0011 1111 1111 1111 = F3FFFH
```

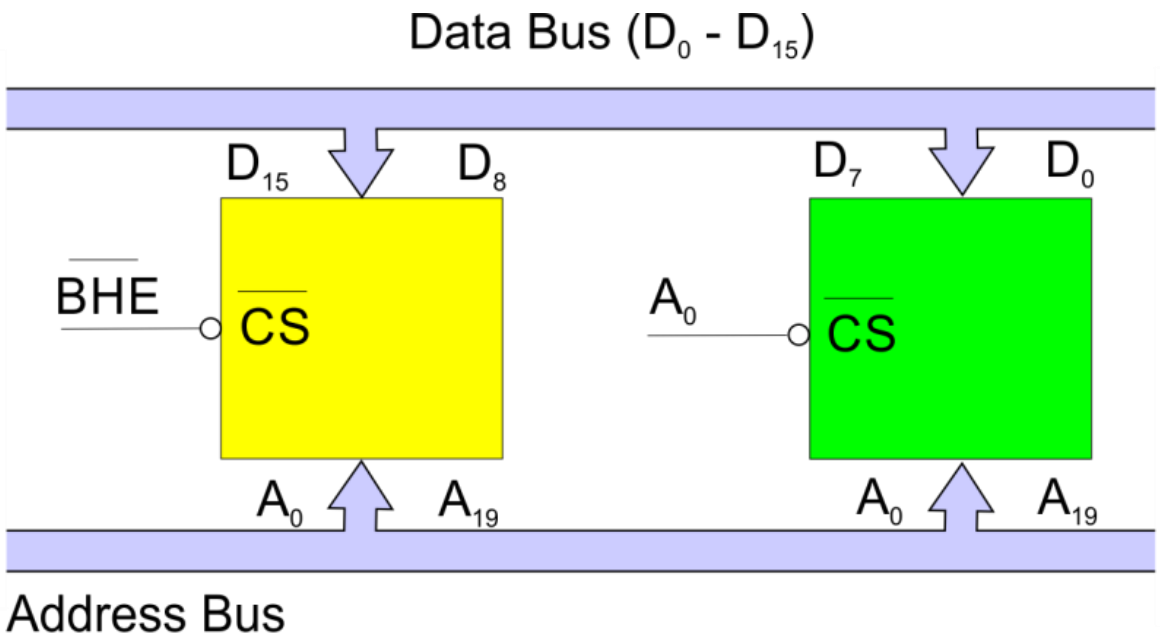
Dual 2 to 4 Line Decoder (74LS139)



Inputs			Outputs			
$\overline{E}$	A	B	$\overline{Y_0}$	$\overline{Y_1}$	$\overline{Y_2}$	$\overline{Y_3}$
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0
1	X	X	1	1	1	1

Odd and Even Banks of Memory




Data Bus (D_0 - D_{15})
D_{15}
D_8
D_7
D_0
BHE
$\overline{CS}$
A_0
$\overline{CS}$
A_0
A_{19}
A_0
A_{19}
Address Bus
Odd Addressed Memory Bank
Even Addressed Memory Bank

Odd and Even Banks of Memory

	Operation	$\overline{BHE}$	A_0	Data Lines Used
1	Read/ Write byte at an even address	1	0	$D_7 - D_0$
2	Read/ Write byte at an odd address	0	1	$D_{15} - D_8$
3	Read/ Write word at an even address	0	0	$D_{15} - D_0$
4	Read/ Write word at an odd address	0	1	$D_{15} - D_0$ in first operation byte from odd bank is transferred
		1	0	$D_7 - D_0$ in first operation byte from odd bank is transferred

Memory organization in 8086

- Available memory space = EPROM + RAM
- Allot equal address space in odd and even bank for both EPROM and RAM
- Can be implemented in two IC's (one for even and other for odd) or in multiple IC's

innovate achieve lead

BITS Pilani

Pilani Campus

Thank You

Memory interfacing

It is required to interface two chips of 32K × 8 ROM and four chips of 32K × 8 RAM with 8086, according to the following map.

ROM 1 and 2 F0000H - FFFFFH, RAM 1 and 2 D0000H - DFFFFH
RAM 3 and 4 E0000H - EFFFFH

Show the implementation of this memory system.

Address	A_{19}	A_{18}	A_{17}	A_{16}	A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_{09}	A_{08}	A_{07}	A_{06}	A_{05}	A_{04}	A_{03}	A_{02}	A_{01}	A_{0}
F0000H	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ROM 1and2											64K									
FFFFFH	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
D0000H	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RAM 1and2											64K									
DFFFFH	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
E0000H	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RAM 3and4											64K									
EFFFFH	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0