Digital Circuit And Logic Design I

Lecture 6

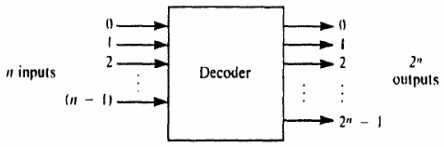
Outline

- Decoder and encoder
  1. decoder
  2. code converter
  3. encoder
  4. priority encoder
- Multiplexer and demultiplexer
  1. multiplexer
  2. demultiplexer

Decoder and encoder

1. Decoder

- A code is a string of several bits. With an n-bit code, it is possible to represent 2^n unique values
- The devices that translate the n-bit pattern into one of the 2^n possible values are called **decoders**

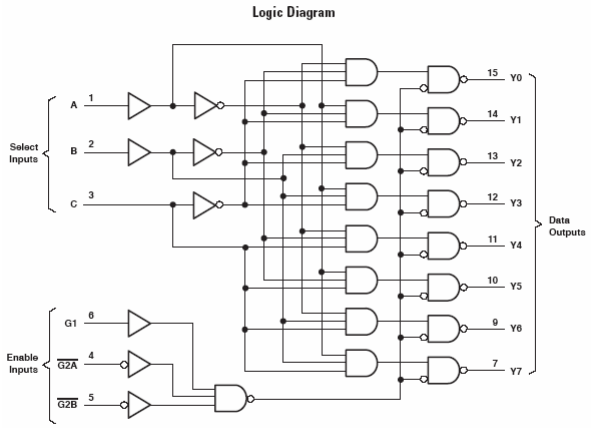


Inputs	Outputs
A B C	0 1 2 3 4 5 6 7
0 0 0	1 0 0 0 0 0 0 0
0 0 1	0 1 0 0 0 0 0 0
0 1 0	0 0 1 0 0 0 0 0
0 1 1	0 0 0 1 0 0 0 0
1 0 0	0 0 0 0 1 0 0 0
1 0 1	0 0 0 0 0 1 0 0
1 1 0	0 0 0 0 0 0 1 0
1 1 1	0 0 0 0 0 0 0 1

3-to-8 decoder truth table

Pictures from text book Introduction to Logic design

1. Decoder (cont.)



3-to-8 decoder logic diagram

'138 MSI component

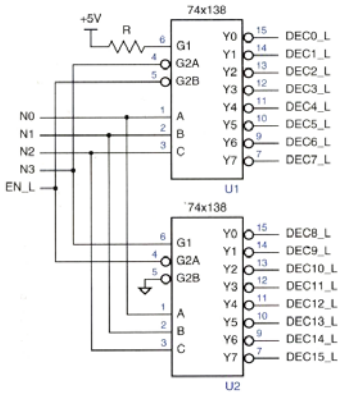
FUNCTION TABLE

INPUTS					OUTPUTS							
ENABLE		SELECT										
G1	$\overline{G2}$*	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	H	H	H	H	H	L	H	H	H	H
H	L	H	L	L	H	H	H	H	L	H	H	H
H	L	H	L	H	H	H	H	H	H	L	H	H
H	L	H	H	L	H	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	H	L

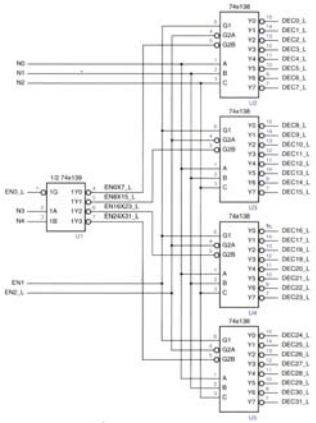
$\overline{G2}$* = $\overline{G2A}$+$\overline{G2B}$

Pictures from text book Digital Logic pocket data book

1. Decoder (cont.)



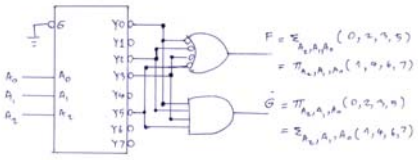
4-to-16 decoder implement by using 3-to-8 decoders



5-to-32 decoder implement by using 3-to-8 decoders and 2-to-4 decoder

1. Decoder (cont.)

- Decoders can be used to implement any function in canonical sum or canonical product form.
- All that is needed is external logic required to OR minterms for canonical sum expression or to AND maxterms for canonical product expression.



Using 3-to-8 decoder to implement logic function F and G

2. Code converter

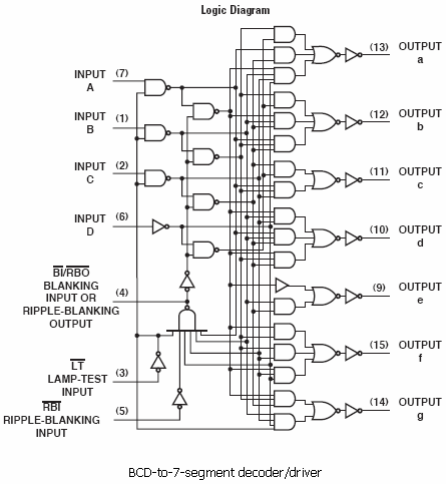
- A code converter is a combinational circuit that translates the input code word into the corresponding new code word
- For example, BCD-to-7-segment decoder and BCD-to-excess-3 code converter

	TTL	CMOS
BCD-to-7 segment decoders	7446	4495
(designed to drive various display types)	7447	4511
	7448	4513
Excess-3-to-decimal (one-of-ten) decoder	7443	
Excess-3-gray-to-decimal (one-of-ten) decoder	7444	

Pictures from text book Introduction to Logic design

Pictures from text book DDPP

2. Code converter (cont.)



BCD-to-7-segment decoder/driver

'47 MSI component

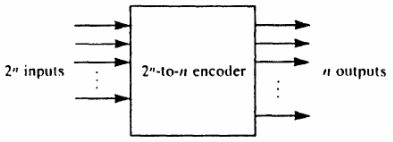
FUNCTION TABLE

No.	INPUTS						BI/RBO	OUTPUTS						
	LT	RBI	D	C	B	A		a	b	c	d	e	f	g
0	H	H	L	L	L	L	H	ON	ON	ON	ON	ON	ON	OFF
1	H	X	L	L	L	H	H	OFF	ON	ON	OFF	OFF	OFF	OFF
2	H	X	L	L	H	L	H	ON	ON	OFF	ON	ON	OFF	ON
3	H	X	L	L	H	H	H	ON	ON	ON	ON	OFF	OFF	ON
4	H	X	L	H	L	L	H	OFF	ON	ON	OFF	OFF	ON	ON
5	H	X	L	H	L	H	H	ON	OFF	ON	ON	OFF	ON	ON
6	H	X	L	H	H	L	H	OFF	OFF	ON	ON	ON	ON	ON
7	H	X	L	H	H	H	H	ON	ON	ON	OFF	OFF	OFF	OFF
8	H	X	H	L	L	L	H	ON	ON	ON	ON	ON	ON	ON
9	H	X	H	L	L	H	H	ON	ON	ON	OFF	OFF	ON	ON
10	H	X	H	L	H	L	H	OFF	OFF	OFF	ON	ON	OFF	ON
11	H	X	H	L	H	H	H	OFF	OFF	ON	ON	OFF	OFF	ON
12	H	X	H	H	L	L	H	OFF	ON	OFF	OFF	OFF	ON	ON
13	H	X	H	H	L	H	H	ON	OFF	OFF	ON	OFF	ON	ON
14	H	X	H	H	H	L	H	OFF	OFF	OFF	ON	ON	ON	ON
15	H	X	H	H	H	H	H	OFF	OFF	OFF	OFF	OFF	OFF	OFF
BI	X	X	X	X	X	X	L	OFF	OFF	OFF	OFF	OFF	OFF	OFF
RBI	H	L	L	L	L	L	L	OFF	OFF	OFF	OFF	OFF	OFF	OFF
LT	L	X	X	X	X	X	H	ON	ON	ON	ON	ON	ON	ON

Pictures from text book Digital Logic pocket data book

3. Encoder

- An **encoder** performs the reverse function of a decoder. The 2^n-to-n decoder generates an n-bit code word as a function of the combination of values on its 2^n inputs
- Usually only one of the inputs is 1 and all others are 0. The output is a binary code word corresponding to the input.
- Because encoders can be built easily, encoder ICs are not available as off-the-shelf components



Inputs				Outputs	
W	X	Y	Z	D_0	D_1
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1

A 4-to-2 encoder truth table

Pictures from text book Introduction to Logic design

4. Priority encoder

- If more than one input is 1 at any time, the encoder generally will not produce a valid output
- Nevertheless, there are exceptions. The so-called **priority encoders** allow more than one of their inputs to be active at any time.
- Each input has priority assigned to it. The code word produced as the output corresponds to that of the highest-priority input among all the inputs that are active

'147 MSI component

FUNCTION TABLE

INPUTS									OUTPUTS			
1	2	3	4	5	6	7	8	9	D	C	B	A
H	H	H	H	H	H	H	H	H	H	H	H	H
X	X	X	X	X	X	X	X	L	L	H	H	L
X	X	X	X	X	X	X	L	H	L	H	H	H
X	X	X	X	X	X	L	H	H	H	L	L	L
X	X	X	X	X	L	H	H	H	H	L	L	H
X	X	X	X	L	H	H	H	H	H	L	H	L
X	X	X	L	H	H	H	H	H	H	L	H	H
X	X	L	H	H	H	H	H	H	H	H	L	L
X	L	H	H	H	H	H	H	H	H	H	L	H
L	H	H	H	H	H	H	H	H	H	H	H	L

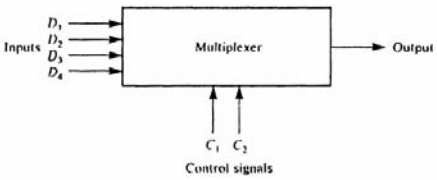
10-to-4 lines priority encoder

Pictures from text book Digital Logic pocket data book

Multiplexer and demultiplexer

1. Multiplexer

- Multiplexing is the process of channeling information from one of several sources to a single destination
- A multiplexer (selector) is thus a switch connecting one of its several inputs to the output. A set of n control inputs is needed to select one of the 2^n inputs

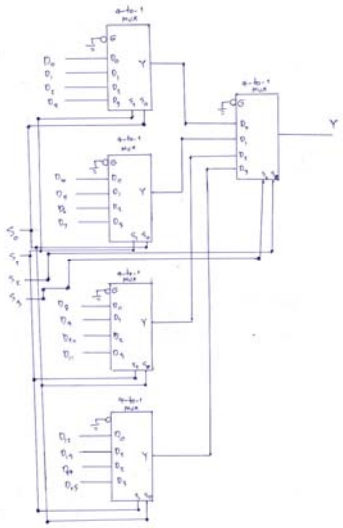


C_1	C_2	Output
0	0	D_1
0	1	D_2
1	0	D_3
1	1	D_4

Pictures from text book Introduction to Logic design

1. Multiplexer (cont.)

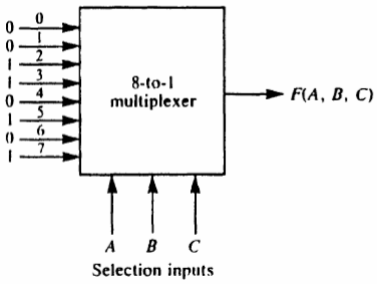
- Multiplexers can be stacked to produce 2^n-to-1multiplexer.



16-to-4 multiplexer implement by using 4-to-1 multiplexer

1. Multiplexer (cont.)

- Multiplexer can be used to realize any logic function F
- For example, F = $\Sigma_{A,B,C}(2,3,5,7)$
- It is possible to reduce the number of input on the multiplexer to implement this function
- We can partition the truth table based on the values of B and C (this is often called folded-table method)

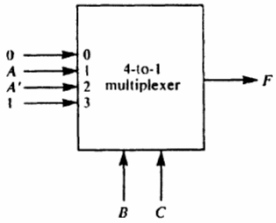


Pictures from text book Introduction to Logic design

1. Multiplexer (cont.)

Minterm	ABC	F
0	000	0
1	001	0
2	010	1
3	011	1
4	100	0
5	101	1
6	110	0
7	111	1

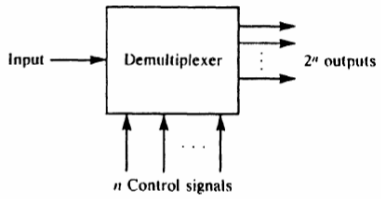
BC	A	Minterm	F	
00	0	0	0	F = 0
	1	4	0	
01	0	1	0	F = A
	1	5	1	
10	0	2	1	F = A′
	1	6	0	
11	0	3	1	F = 1
	1	7	1	

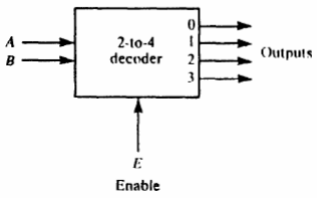


Pictures from text book Introduction to Logic design

2. Demultiplexer

- Demultiplexing is the reverse of multiplexing. That is,A demultiplexer (distributor) distributes its input signal to one of its several outputs
- In fact, a decoder with an enable input can be used demultiplexer.





E	AB	0 1 2 3
0	00	0 0 0 0
0	01	0 0 0 0
0	10	0 0 0 0
0	11	0 0 0 0
1	00	1 0 0 0
1	01	0 1 0 0
1	10	0 0 1 0
1	11	0 0 0 1

Pictures from Introduction to Logic design