EEE104 – Digital Electronics (I)

Lecture 12

Dr. Ming Xu

Dept of Electrical & Electronic Engineering

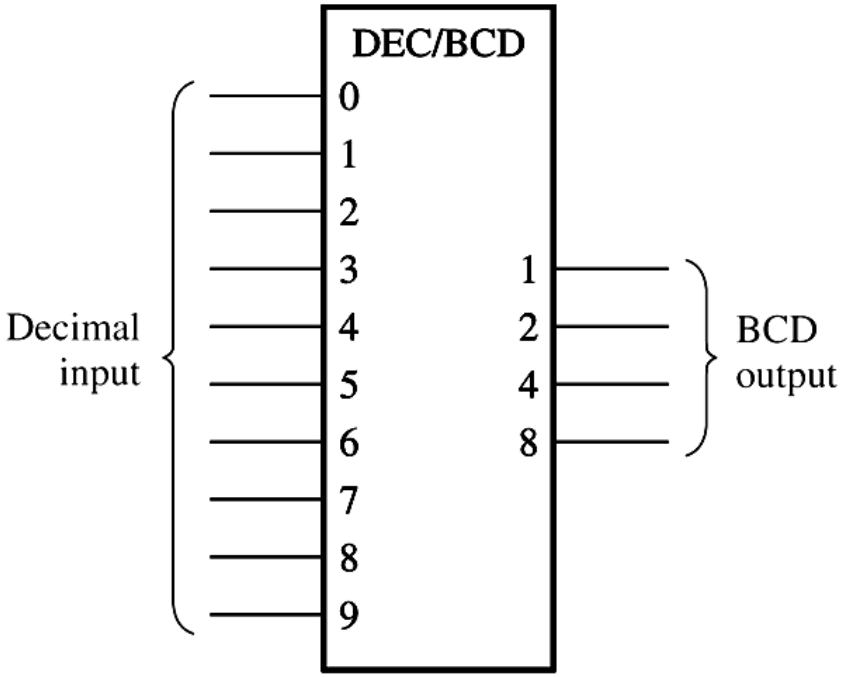
XJTLU

In This Session

- Functions of Combinational Logic Gates
  - Encoders
  - Multiplexers
  - Demultiplexers

Encoders

- *Encoding* is the process of converting from familiar symbols or numbers to a coded format.
- An encoder performs a "reverse" decoder function.



The Decimal-to-BCD Encoder:

If any input is high, it will output a BCD code for that decimal digit, e.g. 4 to 0100.

Encoders

The Decimal-to-BCD Encoder

DECIMAL DIGIT	BCD CODE A_3	A_2	A_1	A_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

$A_3 = 8 + 9$

$A_2 = 4 + 5 + 6 + 7$

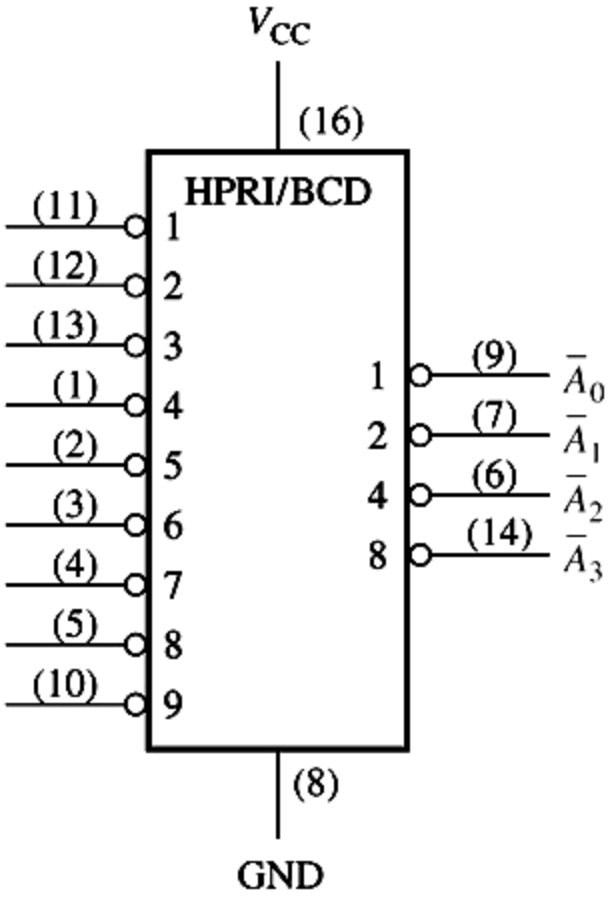
$A_1 = 2 + 3 + 6 + 7$

$A_0 = 1 + 3 + 5 + 7 + 9$

So an OR gate can be used for each output.

Encoders

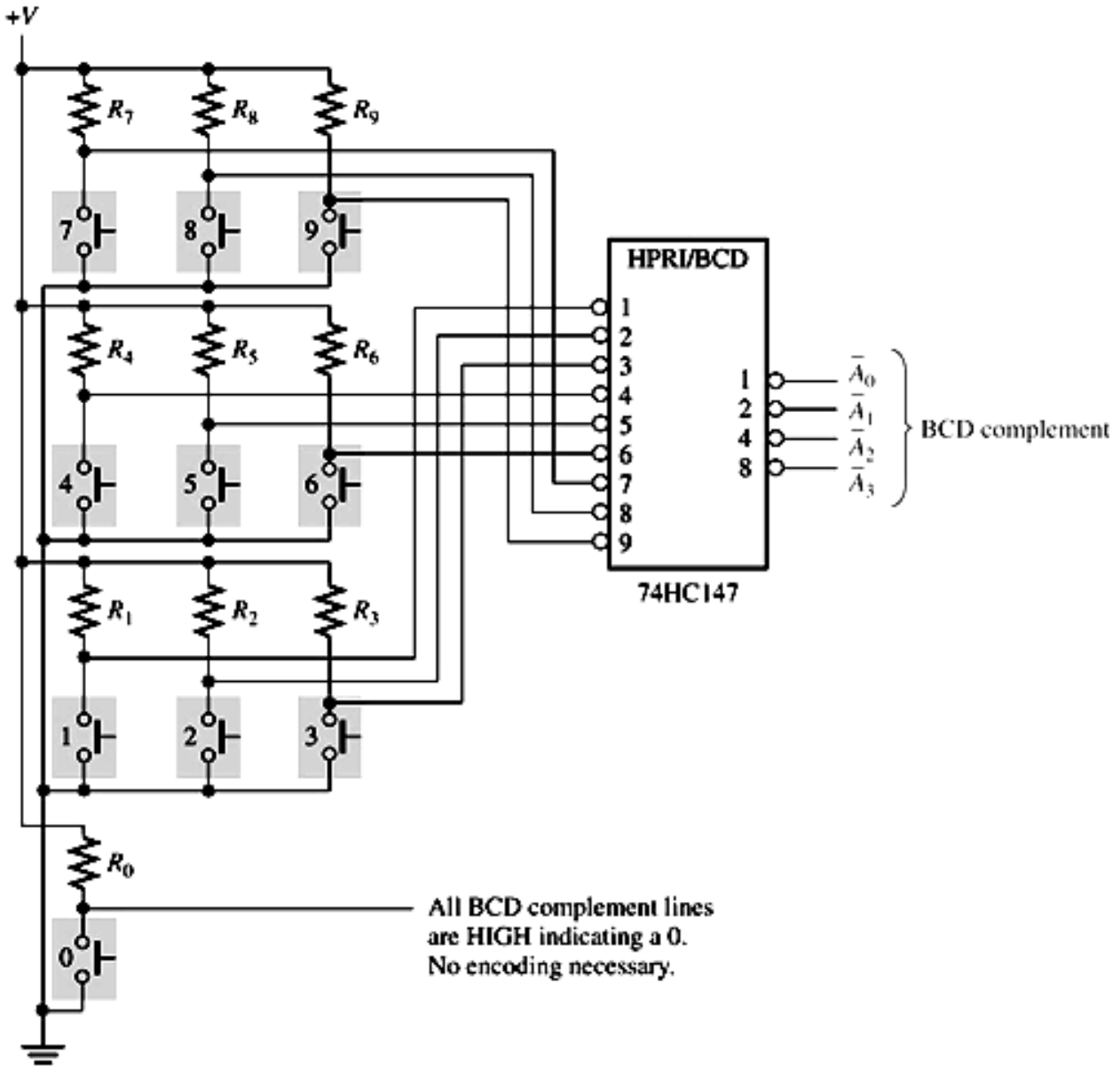
An MSI Decimal-to-BCD Encoder – 74HC147



- Active-LOW inputs and outputs.
- A *priority encoder*: when more than one inputs are active, the highest-order decimal digit input will be active.

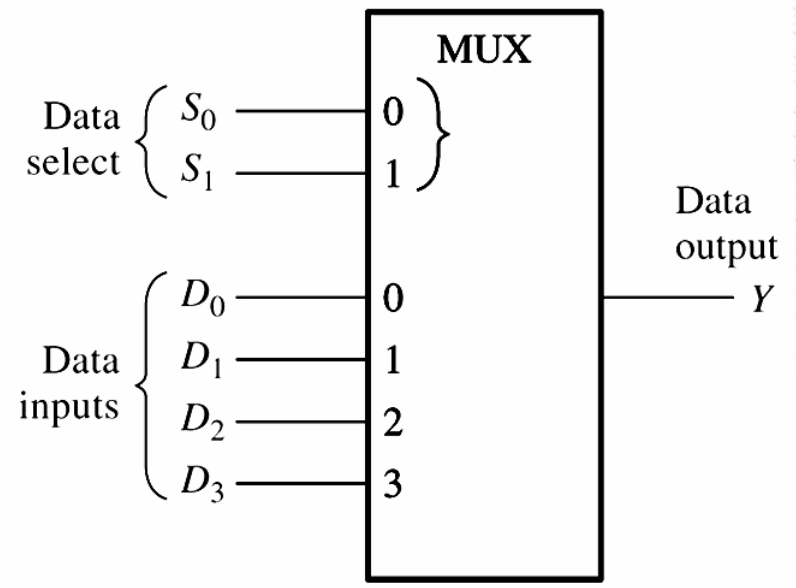
Encoders

Applications:
A keyboard encoder



Multiplexers

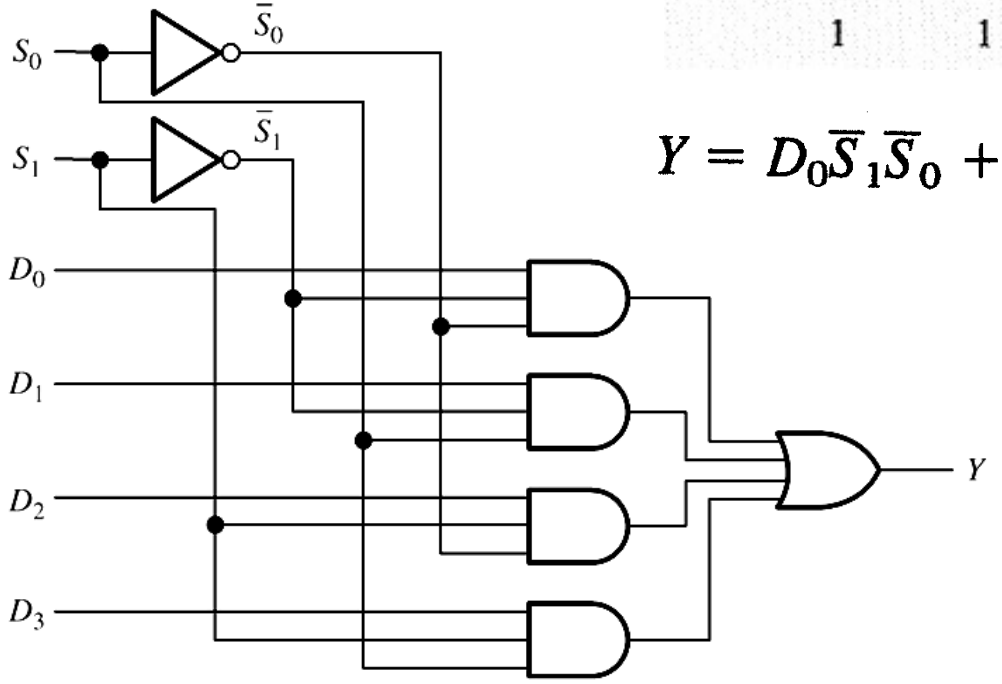
- A *multiplexer (MUX)*, also known as a *data selector*, outputs one of its multiple data inputs.
- The *data select* inputs will decide which data input is to be switched to the output line.



DATA-SELECT INPUTS S_1	S_0	INPUT SELECTED
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

Multiplexers

DATA-SELECT INPUTS S_1	S_0	INPUT SELECTED
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3



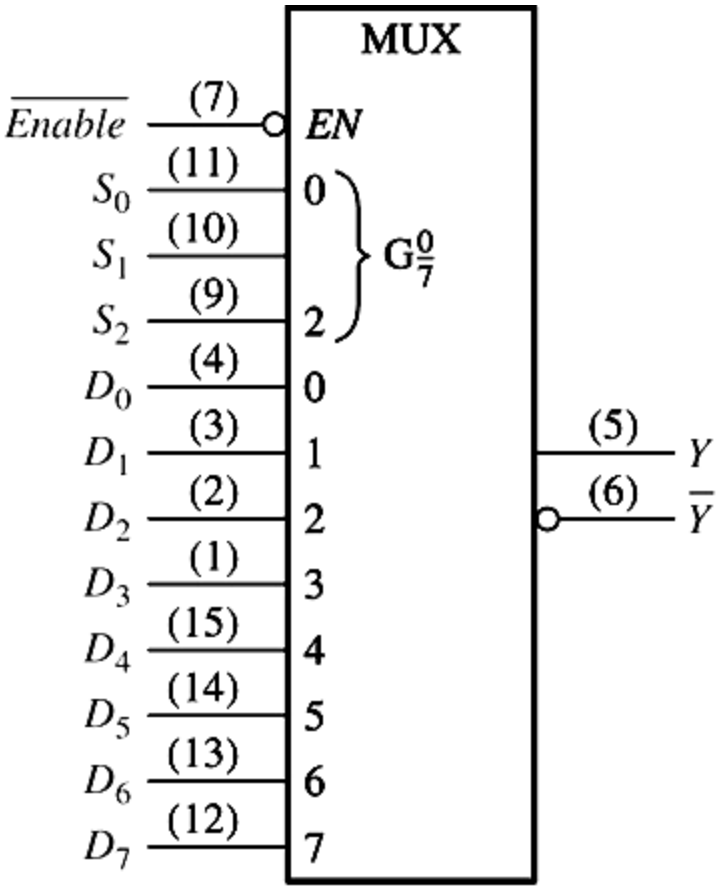
$$Y = D_0\overline{S}_1\overline{S}_0 + D_1\overline{S}_1S_0 + D_2S_1\overline{S}_0 + D_3S_1S_0$$

The expression is in SOP form, so AND-OR logic.

Multiplexers

MSI 8-Input Multiplexers

- When EN is LOW, the selected data input appears in Y.
- When EN is HIGH, Y is LOW and /Y is HIGH.

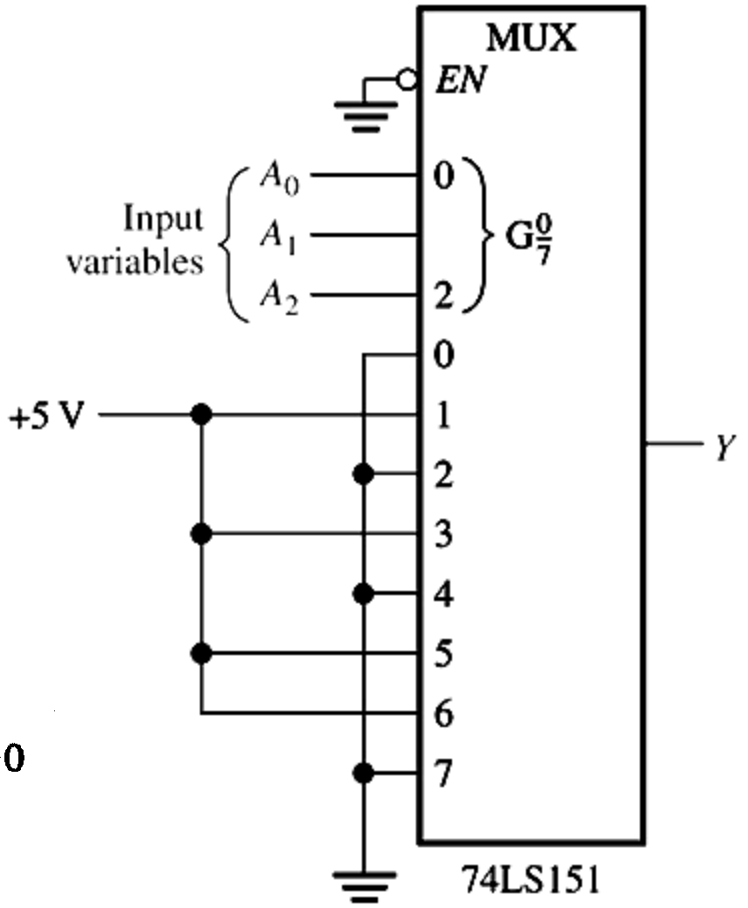


Multiplexers

Application Examples: A Logic Function Generator

Inputs A_2	A_1	A_0	Output Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

$$Y = \bar{A}_2\bar{A}_1A_0 + \bar{A}_2A_1A_0 + A_2\bar{A}_1A_0 + A_2A_1\bar{A}_0$$



Multiplexers

Decimal Digit	Inputs A_3	A_2	A_1	A_0	Output Y
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	0
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	1

Application Examples:

A 4-Variable Logic Function Generator

The $A_3A_2A_1$ are used as data select inputs.

For each pair of rows:

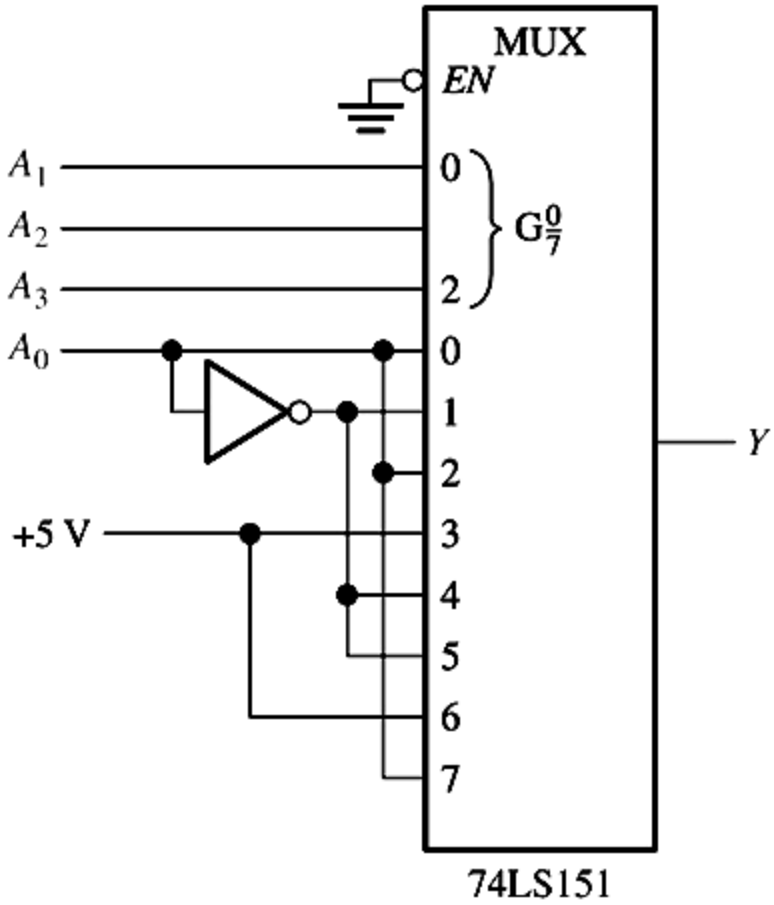
1. A_0 01 Y 00 : Y = 0
2. A_0 01 Y 11 : Y = 1
3. A_0 01 Y 01 : Y = A_0
4. A_0 01 Y 10 : Y = $\overline{A_0}$

Multiplexers

Application Examples:

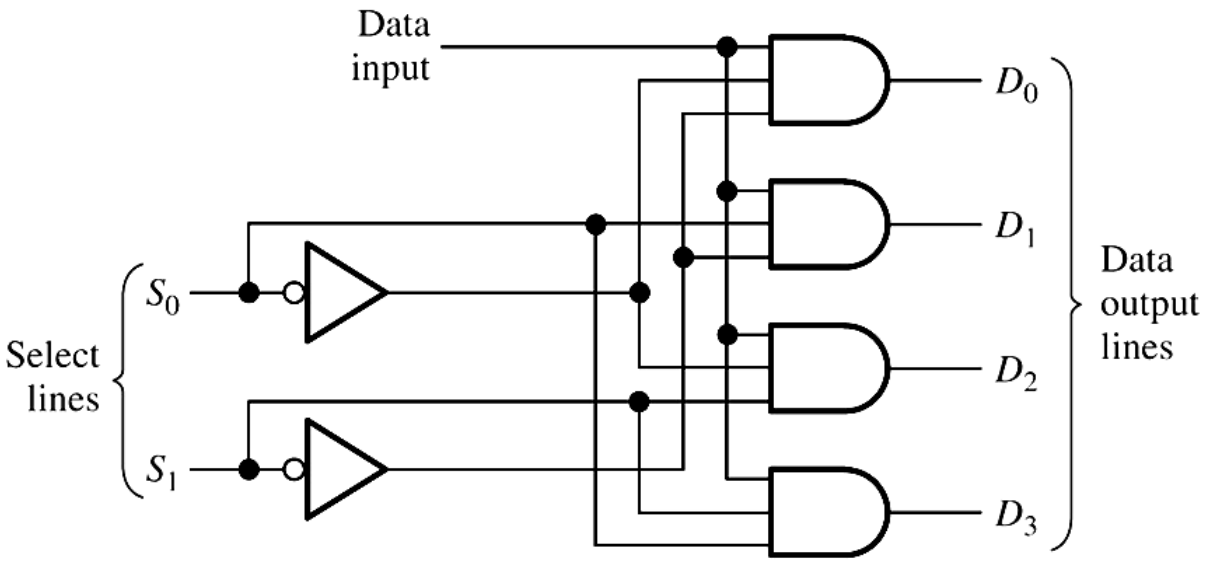
4-Variable Logic Function Generator

$A_3A_2A_1$	Y
000	A_0
001	$\overline{A_0}$
010	A_0
011	1
100	$\overline{A_0}$
101	$\overline{A_0}$
110	1
111	A_0

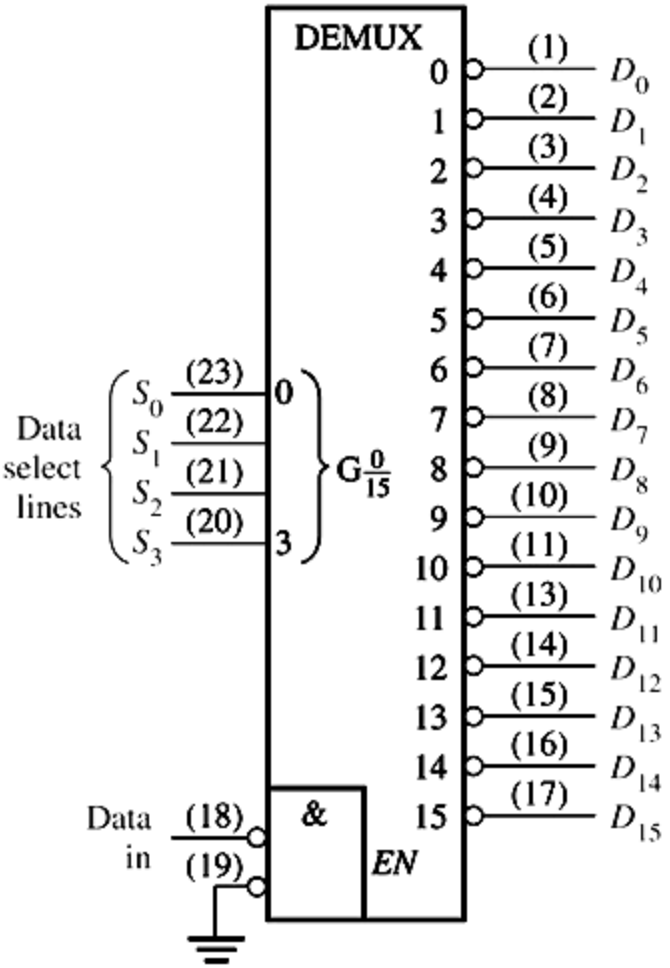


Demultiplexers

- A *demultiplexer* (*DEMUX*) takes data from one line and distributes them to one of the output lines.
- It reverses the multiplexering function.



Demultiplexers



74HC154 (a 4-line-to-16-line decoder) can also be used as an MSI demultiplexer.

The data is input to chip select pins.

Demultiplexers

- MUXs and DEMUXs are often used when data from *multiple sources* are to be transmitted *over one line* and redictributed to *multiple destinations*.