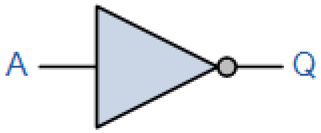
Decoder, Encoder

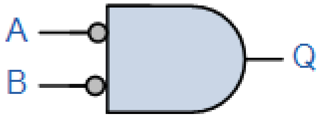
Active Inputs & Outputs

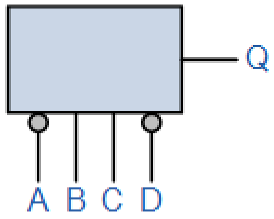
* A logic device input/output can be active HIGH or LOW

- Active HIGH is when a 1 activates a given input, or output
- Active LOW is when a 0 activates a given input, or output

* Negation bubble denotes active LOW input or output. If no bubble is present, input or output is active HIGH







Decoding

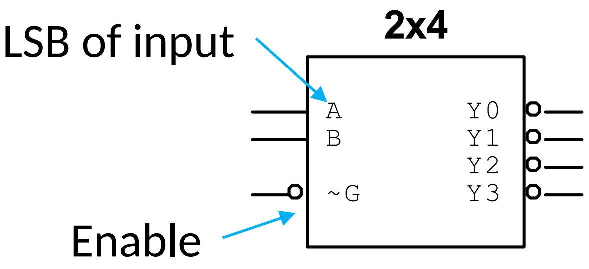
- Taking an input combination, or code, and translating it to one or more active outputs
- BCD decoder – Decodes the BCD input to various outputs
  - BCD to 7 – segment
- X of Y decoder (standard decoder) – Decodes the input X to activate one of Y outputs

Standard Decoder

- Only one output is active at a time
- Number of outputs is 2^n, where n is # of inputs
- Typical nomenclature is <*# of inputs*> to <*# of outputs*> (Not including enable input)
  - 3 to 8 decoder, 3 x 8 decoder
  - 2 to 4 decoder, 2 x 4 decoder
- Most decoders are active LOW outputs and use an active LOW enable

2 x 4 Decoder

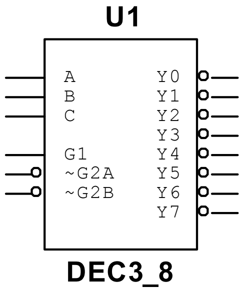
- If enable is **not** active, output is **inactive**
- Only one output is active at a time

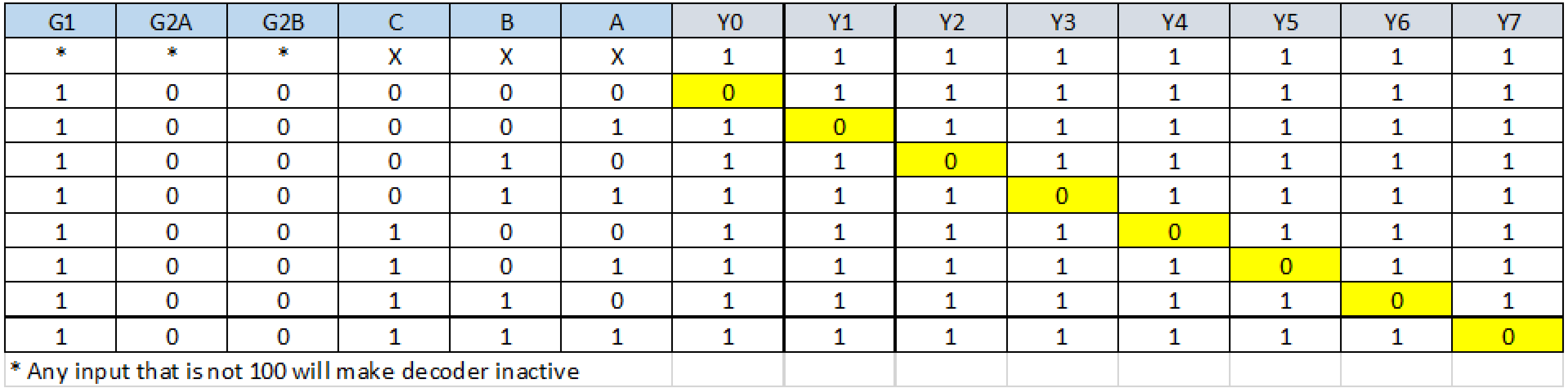


G	B	A	Y0	Y1	Y2	Y3
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

If input is decimal 0, then output 0 is active.
If input is decimal 1, then output 1 is active
If input is decimal 2, then output 2 is active, etc.

3 x 8 Decoder



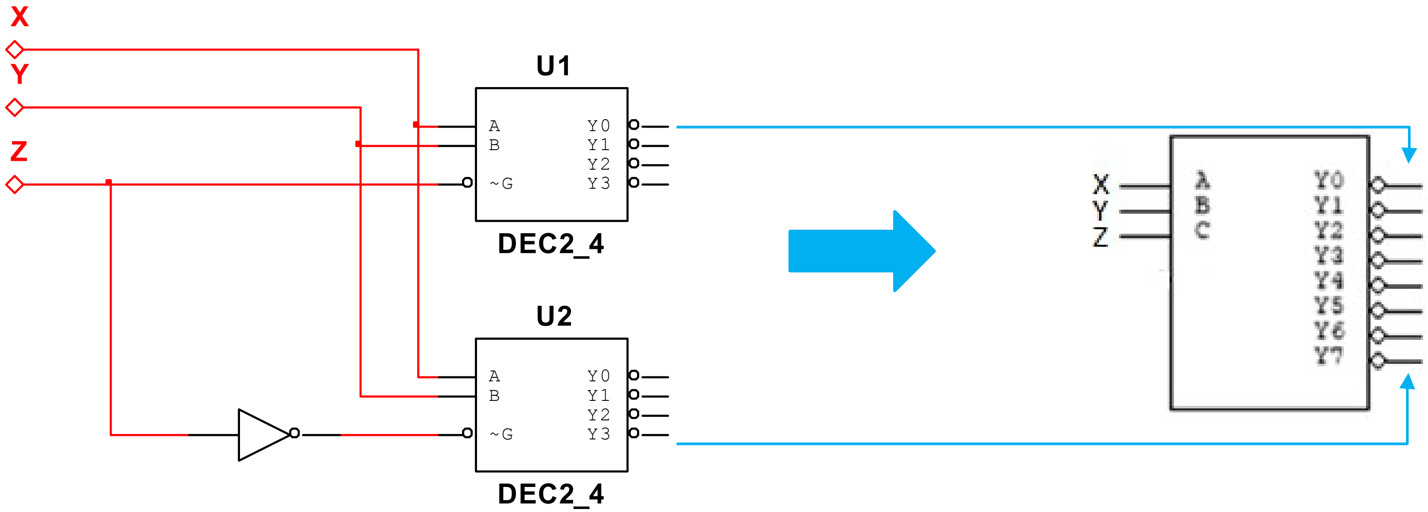


G1	G2A	G2B	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
*	*	*	X	X	X	1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	1	1	1	1	1	1	1
1	0	0	0	0	1	1	0	1	1	1	1	1	1
1	0	0	0	1	0	1	1	0	1	1	1	1	1
1	0	0	0	1	1	1	1	1	0	1	1	1	1
1	0	0	1	0	0	1	1	1	1	0	1	1	1
1	0	0	1	0	1	1	1	1	1	1	0	1	1
1	0	0	1	1	0	1	1	1	1	1	1	0	1
1	0	0	1	1	1	1	1	1	1	1	1	1	0

* Any input that is not 100 will make decoder inactive

Cascading Decoders

- Cascading another decoder will add 1 more input, and double the output
- Cascading 2 x 4 decoders will create a 3 x 8 decoder
- Enable input becomes MSB of input of cascaded decoder

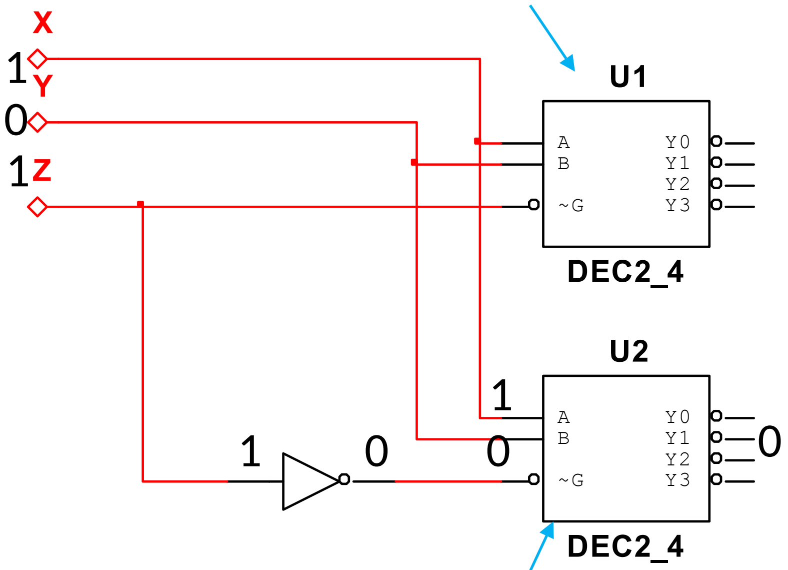


Z is MSB of 3 x 8 decoder

Cascading Decoders

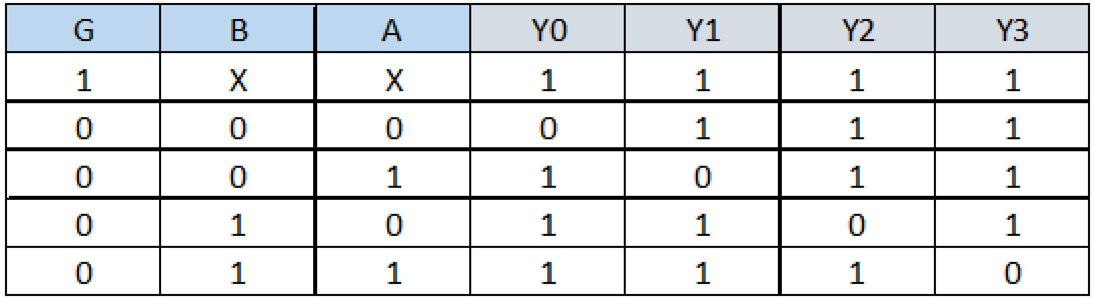
* Only 1 decoder is active at a time according to MSB input Z

top decoder inactive



Bottom decoder active

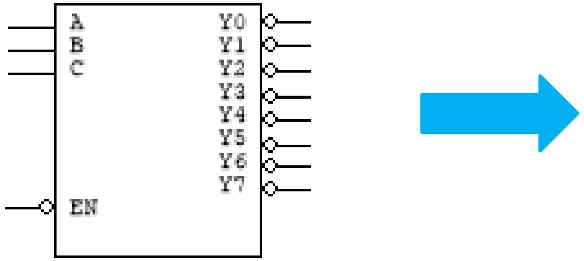
Using truth table, output Y1 is active on bottom decoder

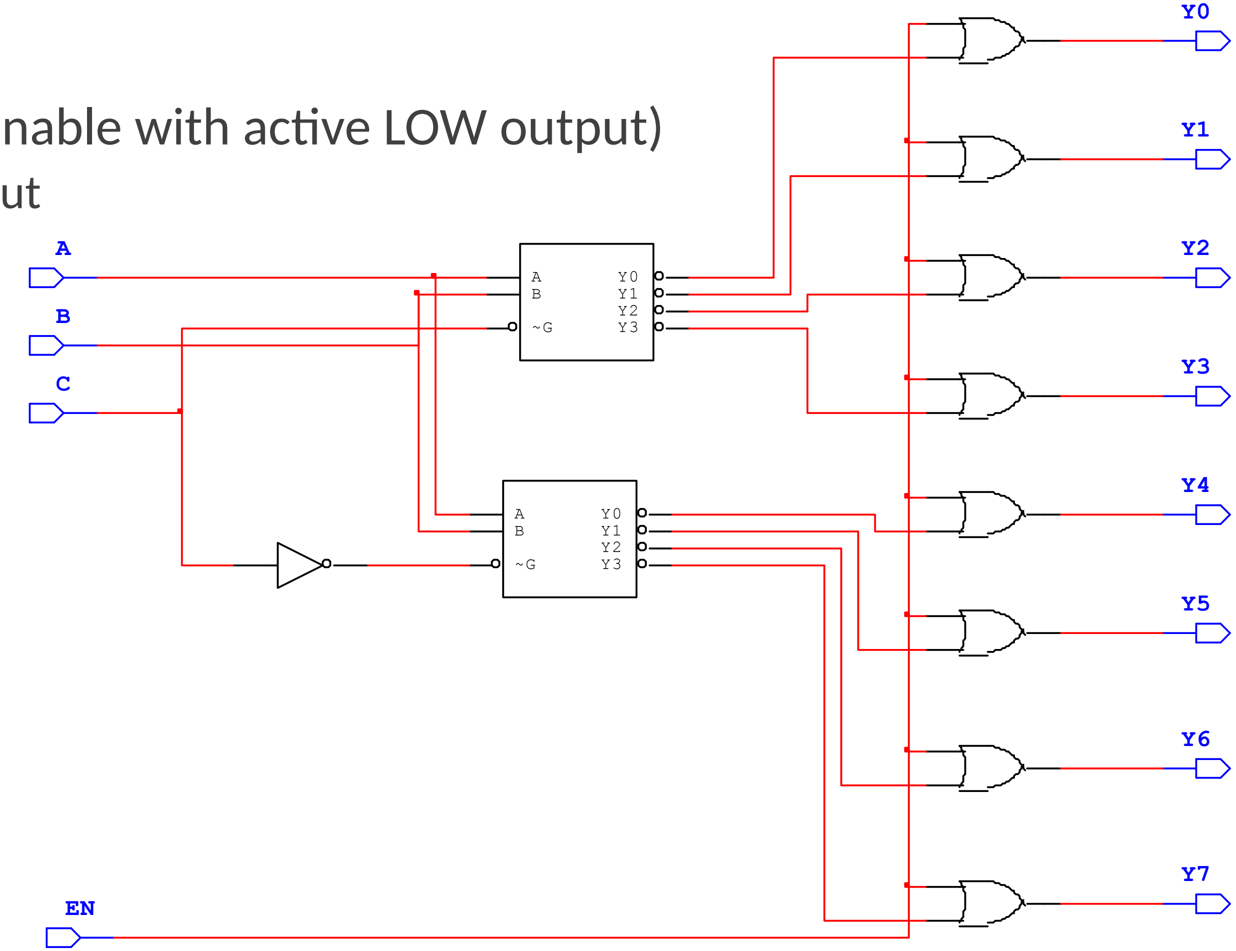


G	B	A	Y0	Y1	Y2	Y3
1	X	X	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

Adding an Enable

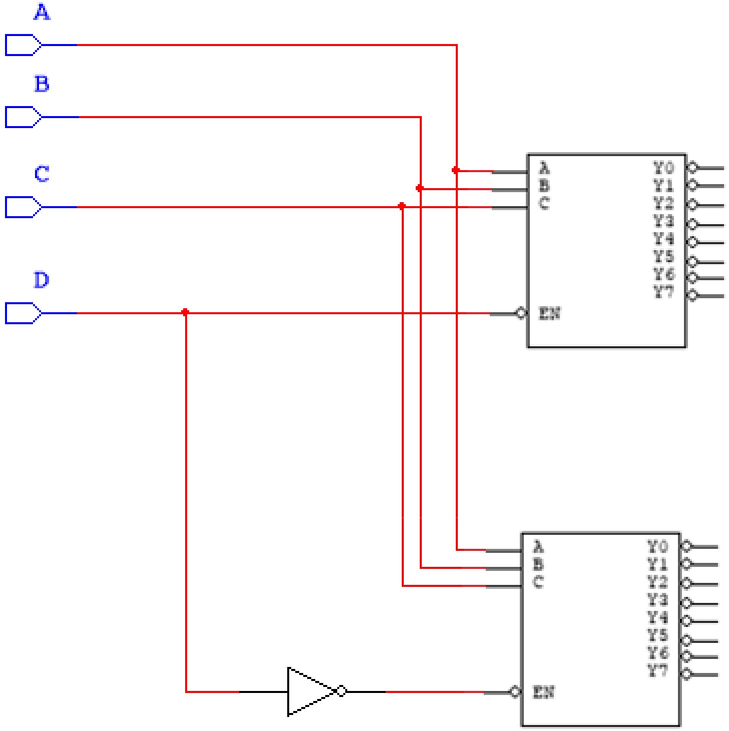
- Enable was used to create additional input
- Use OR gate to create enable pin (for active LOW enable with active LOW output)
  - AND gate for active HIGH enable with active HIGH output





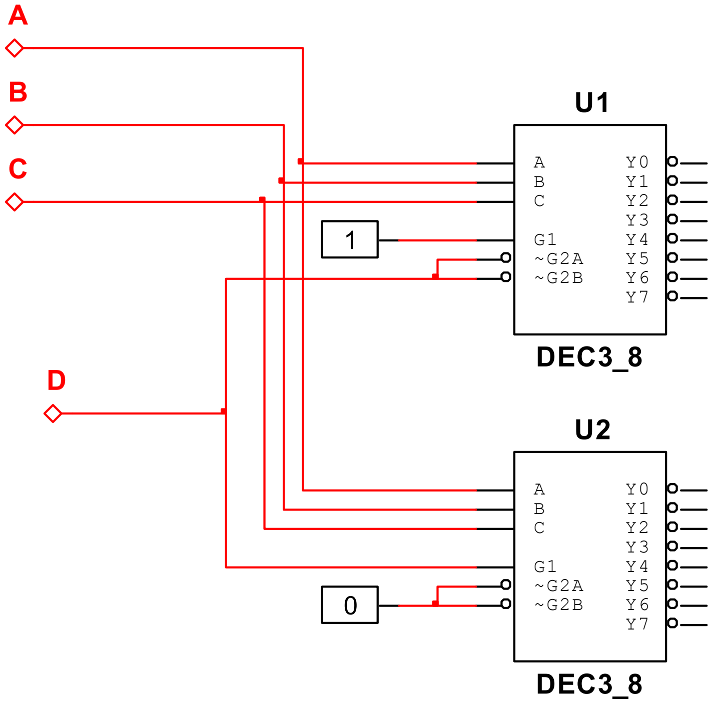
Creating 4 x 16 Decoder

- Using 3x8 created from 2 x 4 decoder example



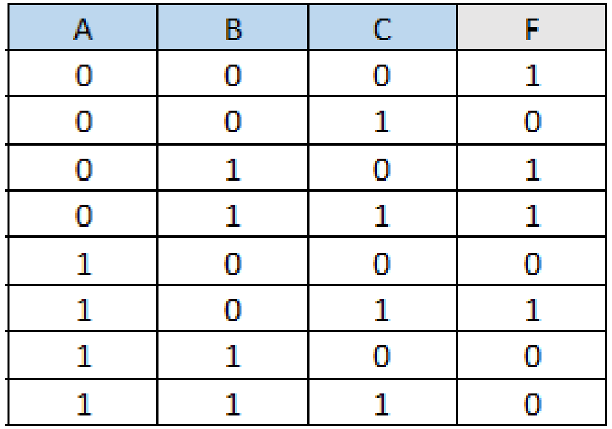
Creating 4 x 16 Decoder

- Top decoder is active for input 0 – 7, and bottom is active for 8 – 15.
- D is the MSB, A is the LSB of input

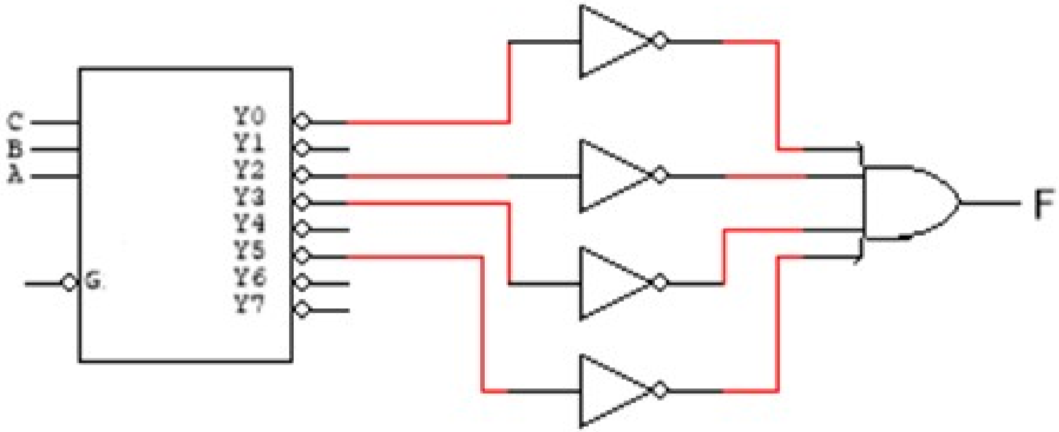


Using Decoders to Implement Minterms

* Decoders can implement a Boolean function as minterms



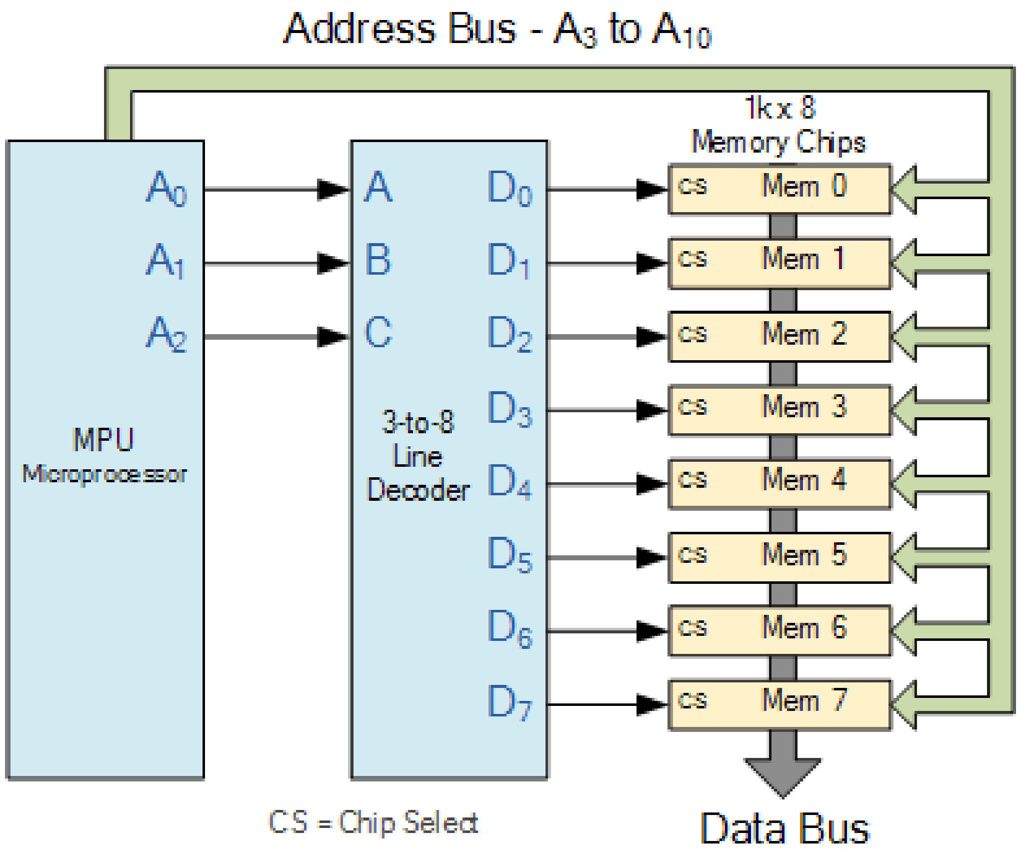
A	B	C	F
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0



The decoder is active LOW and needs NOT gate on output

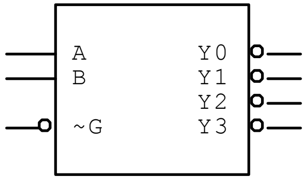
Where are Decoders Used?

- Memory addressing
- Selecting sensors
- Translating binary codes



Example

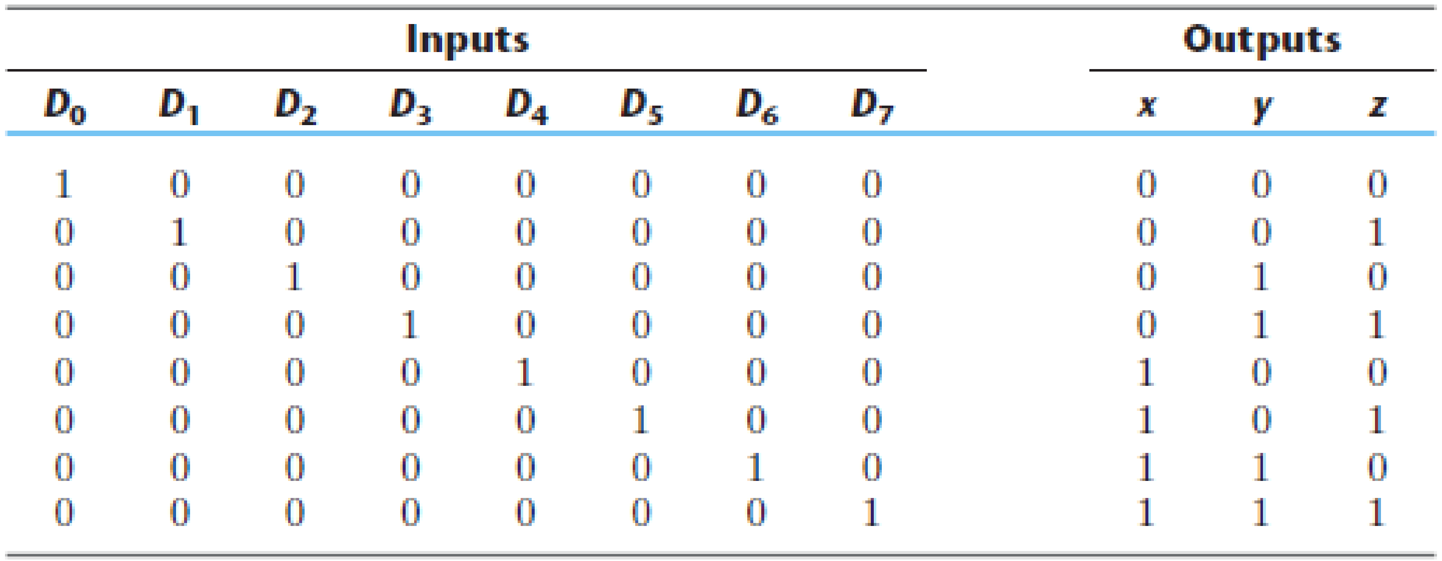
Implement the following truth table using 2 x 4 decoders



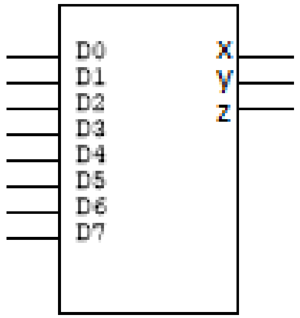
X	Y	Z	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Encoder

- “Reverse” of decoder. Accepts one active input and encodes it to a multiple binary value
- Same nomenclature as decoders <# inputs> x <# outputs>
  - 4 x 2, 8 x 3, 16 x 4



Inputs								Outputs		
D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	x	y	z
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

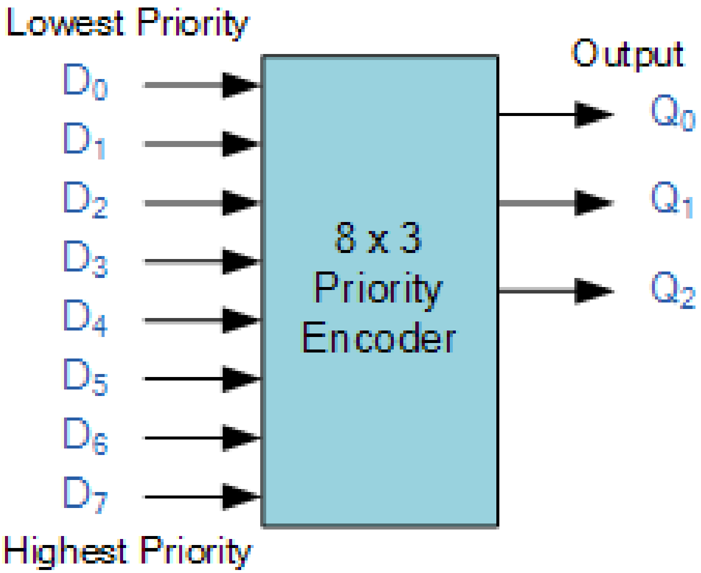


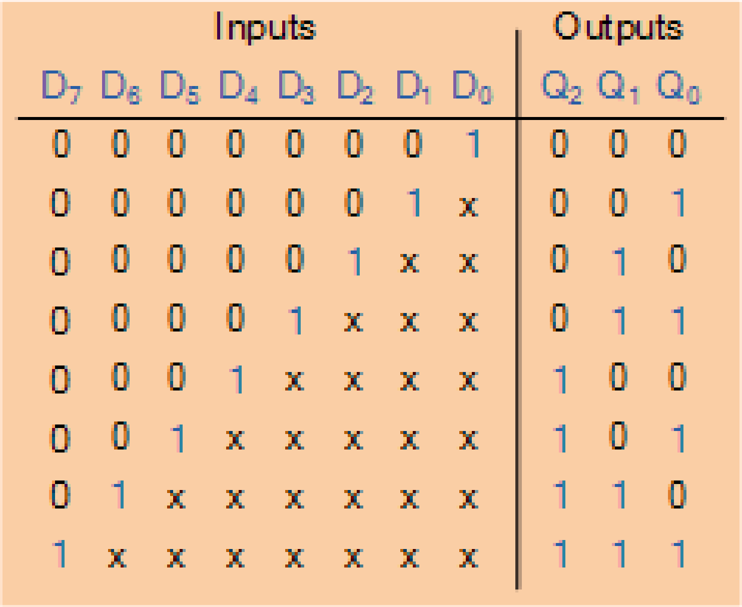
When input 0 (D0) is active, the output is binary 0. When input 6 (D6) is active, the output is binary 6

Priority Encoding

* What if two input values are active with an encoder?

* Priority encoder assigns priority to the MSB on truth table. Don’t care about bits other than MSB

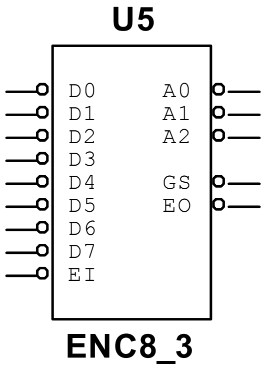




Inputs								Outputs		
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Q_2	Q_1	Q_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	x	0	0	1
0	0	0	0	0	1	x	x	0	1	0
0	0	0	0	1	x	x	x	0	1	1
0	0	0	1	x	x	x	x	1	0	0
0	0	1	x	x	x	x	x	1	0	1
0	1	x	x	x	x	x	x	1	1	0
1	x	x	x	x	x	x	x	1	1	1

Active Low Priority Encoder

E1	D7	D6	D5	D4	D3	D2	D1	D0	A2	A1	A0
1	X	X	X	X	X	X	X	X	1	1	1
0	0	1	1	1	1	1	1	1	0	0	0
0	1	0	1	1	1	1	1	1	0	0	1
0	1	1	0	1	1	1	1	1	0	1	0
0	1	1	1	0	1	1	1	1	0	1	1
0	1	1	1	1	0	1	1	1	1	0	0
0	1	1	1	1	1	0	1	1	1	0	1
0	1	1	1	1	1	1	0	1	1	1	0
0	1	1	1	1	1	1	1	0	1	1	1



Creates an active LOW code to represent the input. D7 would be binary 7 (111) if output is active HIGH. Active LOW NOTs the output. D7 is binary 0, or NOT (111). D0 is binary 7 or NOT (000).

GS and EO Outputs

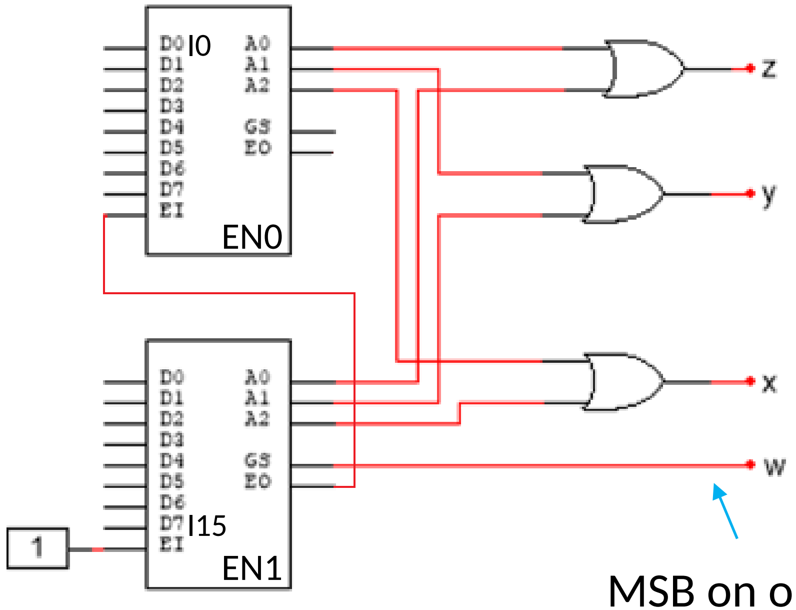
* E0 (enable output), and GS (group select)
  - Used to cascade for expanding input and output
  - If **no** input active on encoder, E0 is active
  - If **any** input is active, GS is active

Cascading Encoders

- Doubles the inputs, and adds one more output
  - Cascading 8 x 3 creates a 16 x 4
- Depending if the encoder is active HIGH or LOW, the outputs are ORed or NANDed together
- GS becomes MSB of output on cascaded encoder

Active HIGH Cascading

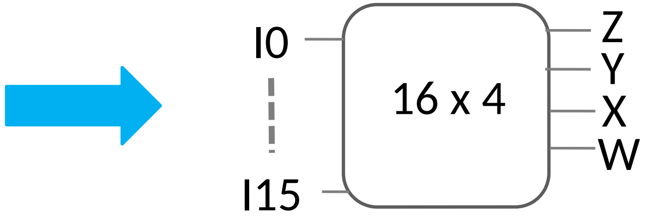
- If no input is present on encoder, E0 is active. If any input present on encoder, GS is active



MSB on output

EN0 will serve inputs I0 – I7

EN1 will serve inputs I8 – I15

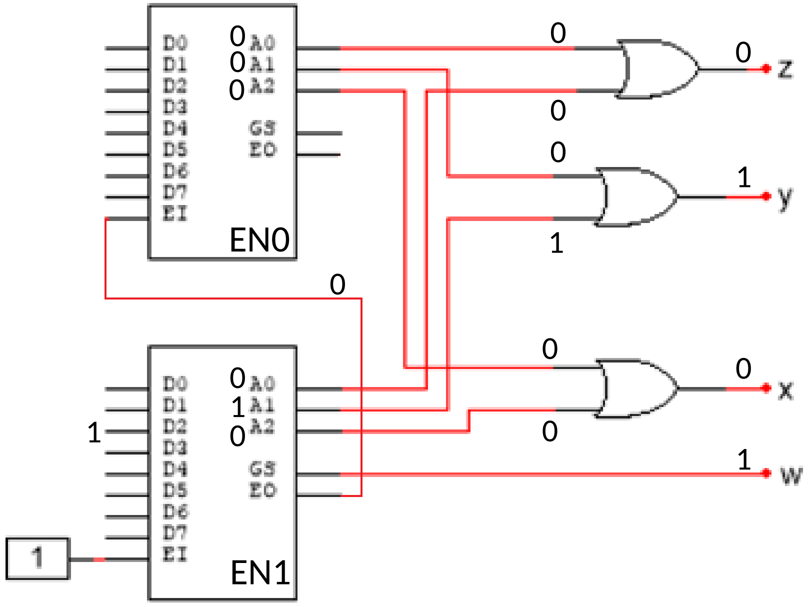


8 x 3 priority encoder truth table

E1	D7	D6	D5	D4	D3	D2	D1	D0	A2	A1	A0
0	x	x	x	x	x	x	x	x	0	0	0
1	1	x	x	x	x	x	x	x	1	1	1
1	0	1	x	x	x	x	x	x	1	1	0
1	0	0	1	x	x	x	x	x	1	0	1
1	0	0	0	1	x	x	x	x	1	0	0
1	0	0	0	0	1	x	x	x	0	1	1
1	0	0	0	0	0	1	x	x	0	1	0
1	0	0	0	0	0	0	1	x	0	0	1
1	0	0	0	0	0	0	0	1	0	0	0

Active HIGH Cascading

If a 1 is applied to input D2 on EN1, (input I10), the encoder output will be 1010 (decimal 10)

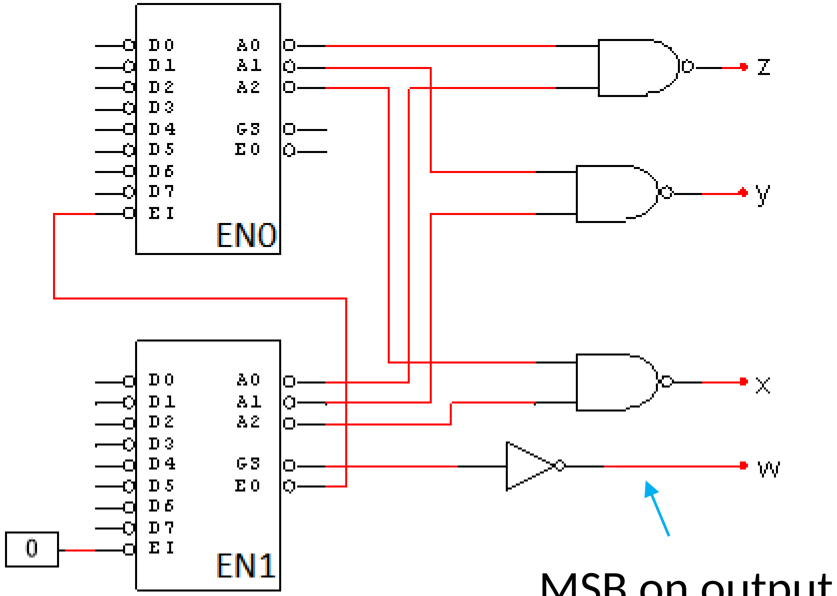


E1	D7	D6	D5	D4	D3	D2	D1	D0	A2	A1	A0
0	x	x	x	x	x	x	x	x	0	0	0
1	1	x	x	x	x	x	x	x	1	1	1
1	0	1	x	x	x	x	x	x	1	1	0
1	0	0	1	x	x	x	x	x	1	0	1
1	0	0	0	1	x	x	x	x	1	0	0
1	0	0	0	0	1	x	x	x	0	1	1
1	0	0	0	0	0	1	x	x	0	1	0
1	0	0	0	0	0	0	1	x	0	0	1
1	0	0	0	0	0	0	0	1	0	0	0

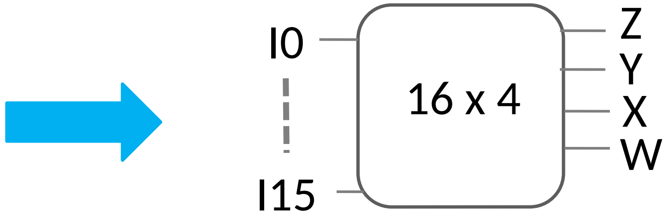
E0 is 0 because an input is active, EN0 is disabled. GS is 1 because an input is active

Active LOW Cascading

- Same setup as active HIGH cascading but NAND the outputs
  - The NAND will allow an output of decimal 11 to appear 1011 (binary 11) instead of 0100 (binary 4)



MSB on output



EN0 will serve inputs I0 – I7

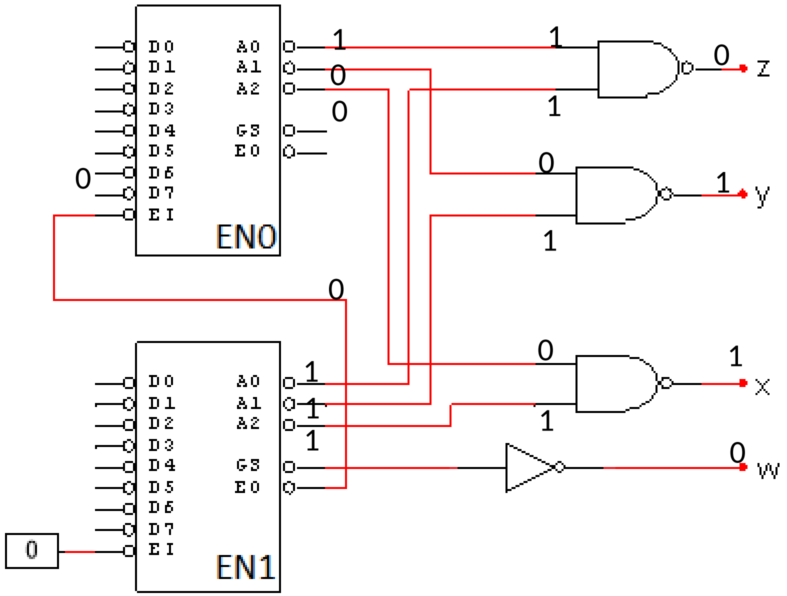
EN1 will serve inputs I8 – I15

8 x 3 priority encoder truth table

E1	D7	D6	D5	D4	D3	D2	D1	D0	A2	A1	A0
1	X	X	X	X	X	X	X	X	1	1	1
0	0	X	X	X	X	X	X	X	0	0	0
0	1	0	X	X	X	X	X	X	0	0	1
0	1	1	0	X	X	X	X	X	0	1	0
0	1	1	1	0	X	X	X	X	0	1	1
0	1	1	1	1	0	X	X	X	1	0	0
0	1	1	1	1	1	0	X	X	1	0	1
0	1	1	1	1	1	1	0	X	1	1	0
0	1	1	1	1	1	1	1	0	1	1	1

Active LOW Cascading

If a 0 is applied to input D6 on EN0, (input I6), the encoder output will be 0110 (decimal 6)



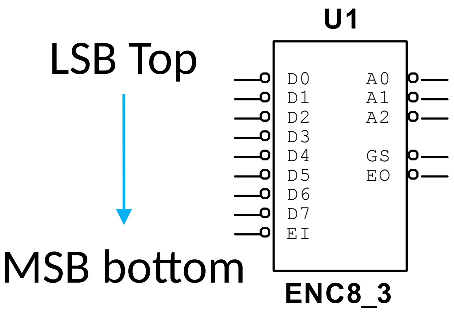
E1	D7	D6	D5	D4	D3	D2	D1	D0	A2	A1	A0
1	X	X	X	X	X	X	X	X	1	1	1
0	0	X	X	X	X	X	X	X	0	0	0
0	1	0	X	X	X	X	X	X	0	0	1
0	1	1	0	X	X	X	X	X	0	1	0
0	1	1	1	0	X	X	X	X	0	1	1
0	1	1	1	1	0	X	X	X	1	0	0
0	1	1	1	1	1	0	X	X	1	0	1
0	1	1	1	1	1	1	0	X	1	1	0
0	1	1	1	1	1	1	1	0	1	1	1

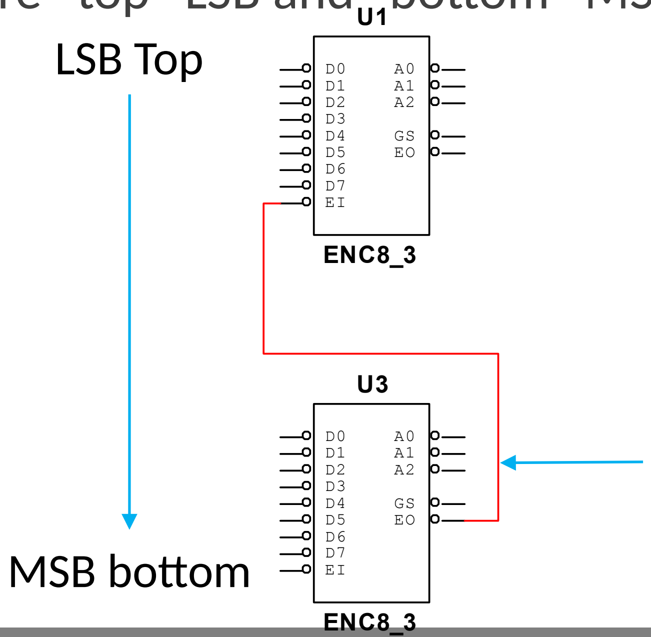
E0 is 1 because no inputs are active, EN0 is enabled. W is 0 because no inputs are active on EN1

Which Encoder is MSB?

* Looking at an individual encoder, the inputs start from LSB (D0) and go to MSB (D7). The outputs start from LSB (A0) and go to MSB (A2).

* When cascading, adding another encoder creates more MSB digits. The “top” encoder is the LSB encoder and the “bottom” encoder is the MSB encoder. This is because the inputs on the individual encoder are “top” LSB and “bottom” MSB.

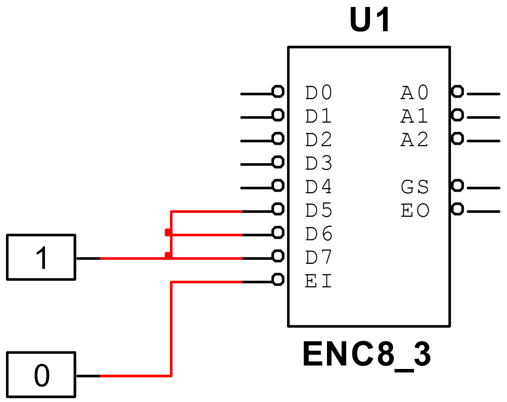




If no MSB inputs active, enable the LSB encoder to check for LSB inputs active

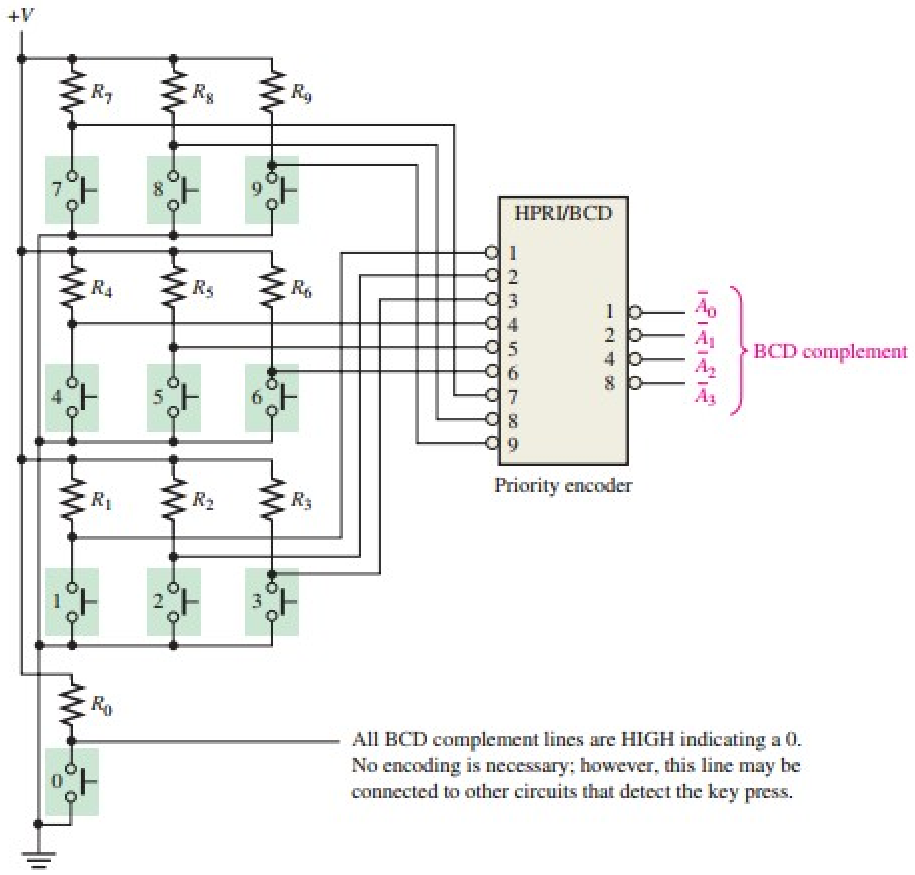
Unused Inputs

- Encoder inputs that are unused are made inactive.
  - Disable inputs starting from the MSB

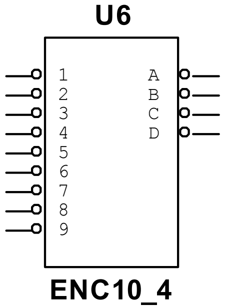


The following encoder is only using 5 of its 8 inputs

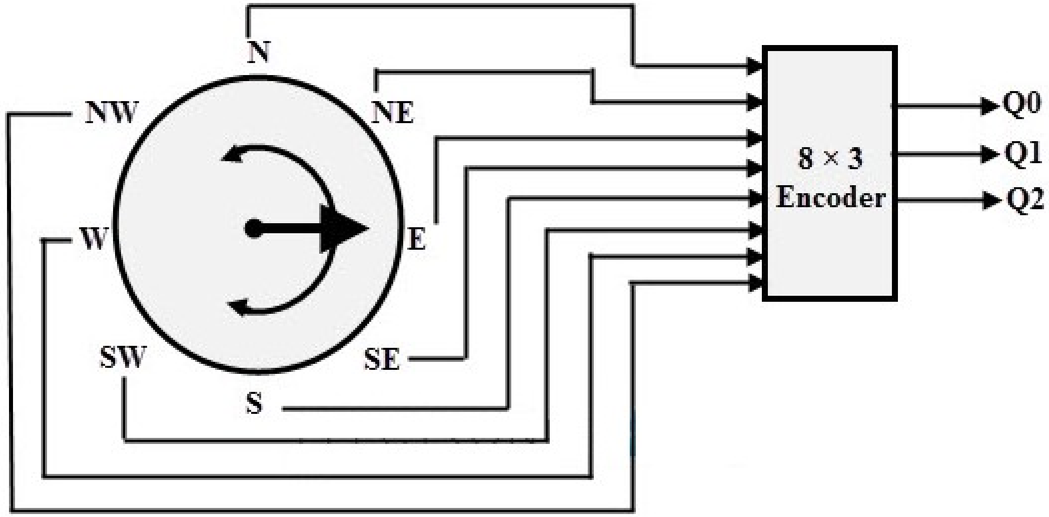
Encoder Application

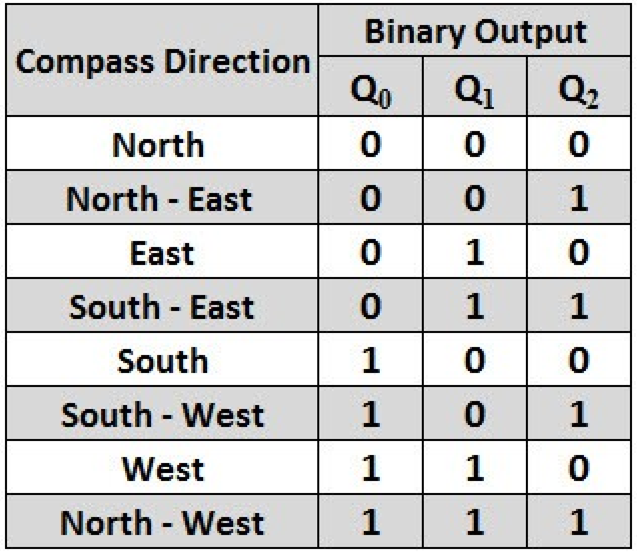


All BCD complement lines are HIGH indicating a 0. No encoding is necessary; however, this line may be connected to other circuits that detect the key press.



Encoder Applications



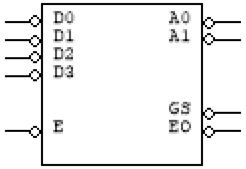


Compass Direction	Binary Output		
	Q_0	Q_1	Q_2
North	0	0	0
North - East	0	0	1
East	0	1	0
South - East	0	1	1
South	1	0	0
South - West	1	0	1
West	1	1	0
North - West	1	1	1

Example

Given the following 4 x 2 active LOW priority encoder (D0 is LSB on input, A0 is LSB on output)

- Cascade it to create a 8 x 3 encoder
- If the input 0xAF is applied to the cascaded encoder, what is the output?


D0
D1
D2
D3
E
A0
A1
GS
EO