Penn State Abington
CMPEN 271
Lecture Set #12
Multiplexers (MUXs), Demultiplexers (DMUXs), Magnitude Comparator

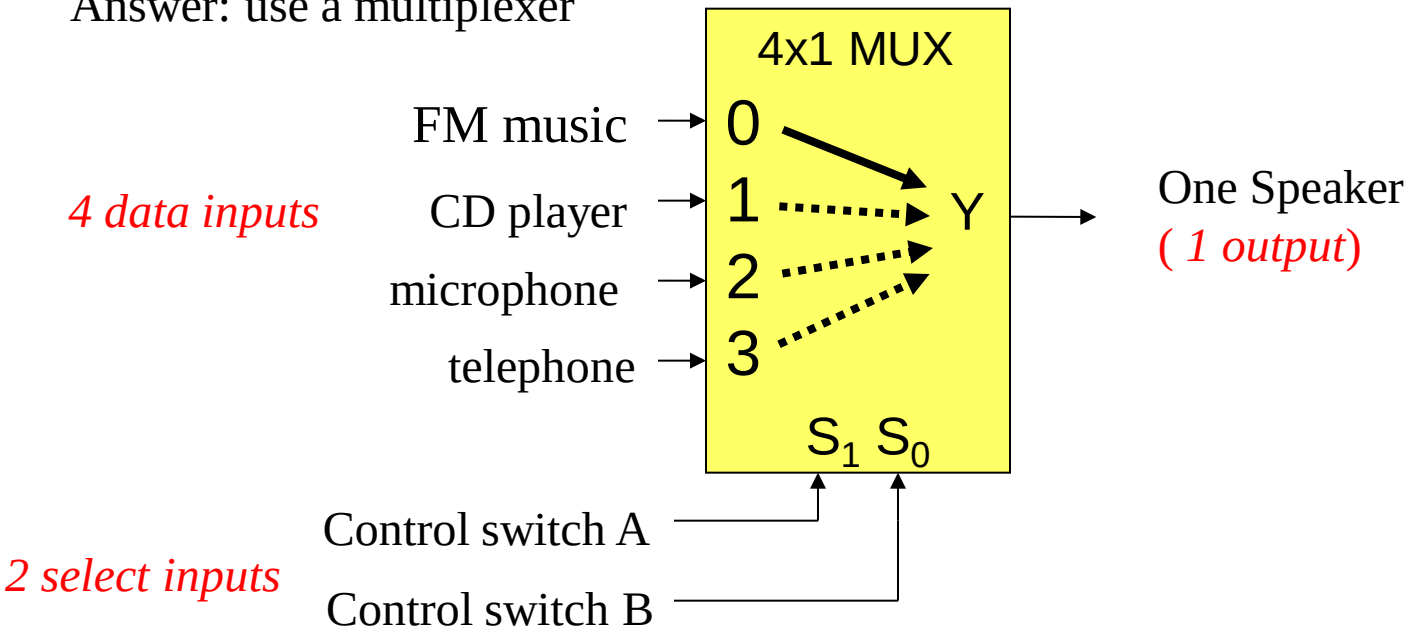
R. Avanzato © 2013-2015

Topics:

- Multiplexers (MUXs) — **Video part 1** ←
- MUX Applications — Video part 2
- Simulation of MUXs
- MUX IC Specs
- Demultiplexer (DMUX) — Video part 3
- DMUX Applications
- Magnitude Comparator (with design problem)
- HW #6 (BCD values and 7-segm display) — Video part 4
- Review Questions — Video part 5

What is a Multiplexer (MUX)?

Concept: Assume you have 4 audio sources (inputs) and only one speaker(output). How do you electronically select the one of the inputs to be connected to the output?
Answer: use a multiplexer

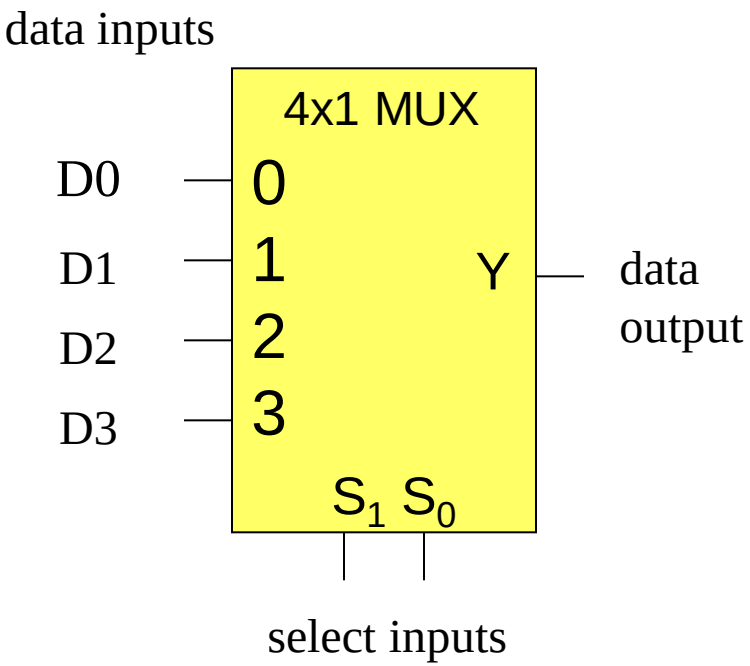


Abstract illustration -- set switches to select the which of the 4 input signals are to to be redirected to the speaker (output). For example, when switch A is 0 and switch B is 1, then the CD player signal would be switched to the speaker. (Note: digital MUXs do not work with analog systems, but the concept of multiplexing is found in many applications.) We will be dealing with digital inputs and outputs only.

What is a Multiplexer (MUX)?

- A **multiplexer (or MUX)** is an MSI (medium-scale integrated circuit) device
- There are **2 sets of inputs** for a MUX. One is a set of n "**data**" inputs, and the other set is a set of log n "**select**" inputs. For example, if there are 8 data inputs, then there are 3 select inputs. A MUX **has one data output**.
- Based on the **"select" inputs**, one of the "data" inputs is directed (connected) to the single output.
- A MUX is often referred to as a **"data selector"** or a "**switch**".
- MUXs exist in a variety of **sizes**: 2x1, 4x1, 8x1, 16x1, 32x1, etc.
- A **2x1 (or 2-to-1) MUX** has 2 data inputs, 1 select input, and one output.
- An **8x1 (or 8-to-1) MUX** has 8 data inputs, 3 select inputs, and one output.
- Mux's are used to **implement Boolean functions**, and to control the flow of data in a computer (e.g. multiplexed display)
- Multiplexers are commonly-used "**building blocks**" in digital design circuits.

4x1 Multiplexer



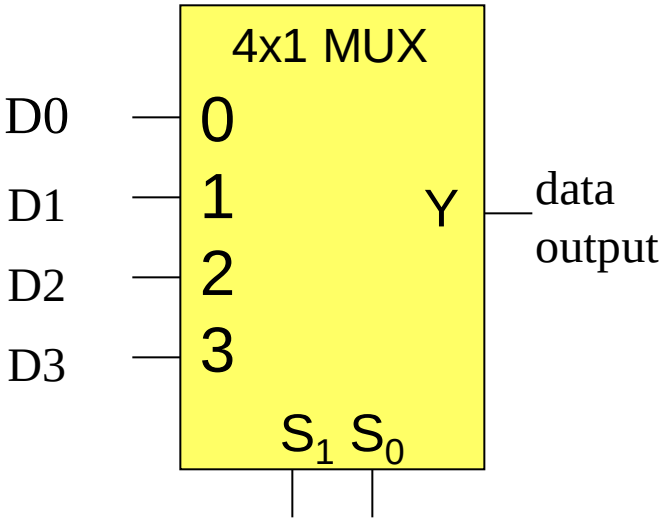
Block diagram

inputs		output
S_1	S_0	Y
0	0	D0
0	1	D1
1	0	D2
1	1	D3

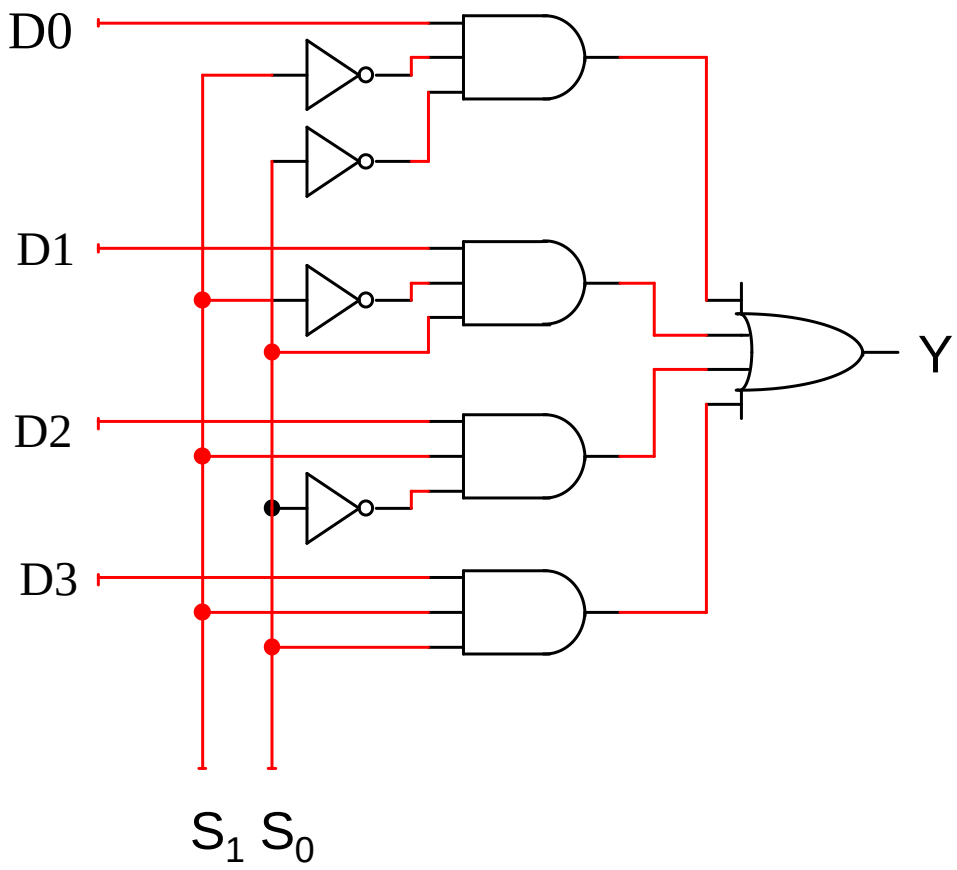
Truth table

A MUX is more of a “switch” or data selector.
Compare MUX to a decoder.

4x1 Multiplexer - 2

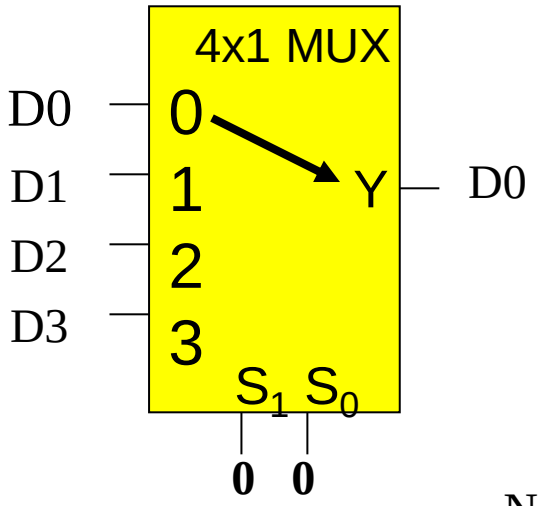


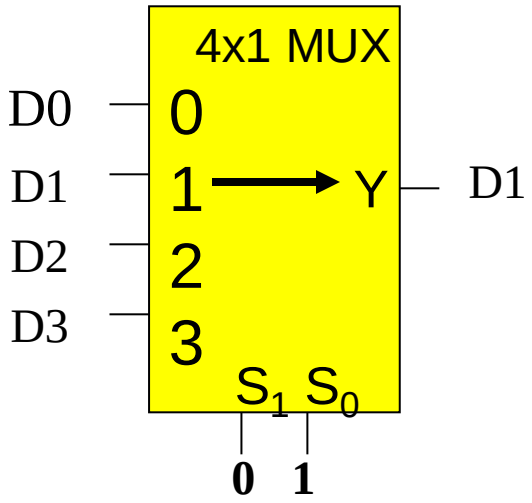
Block diagram



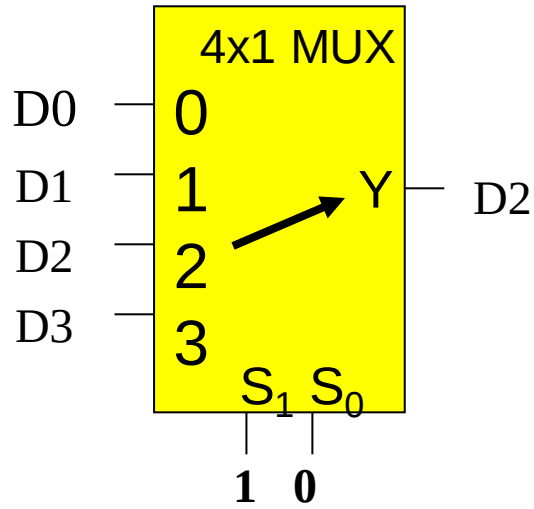
LogicWorks circuit

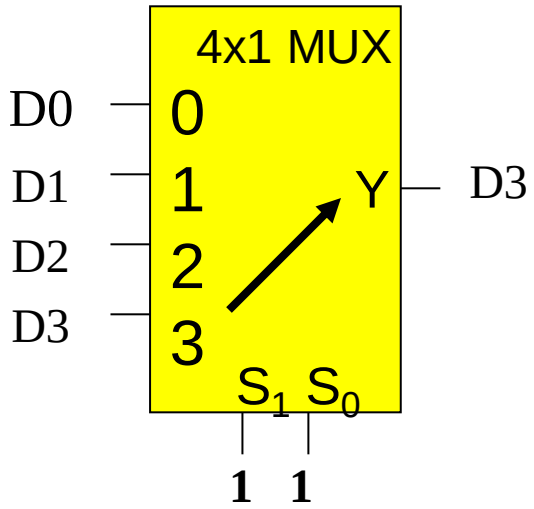
4x1 Multiplexer - 3



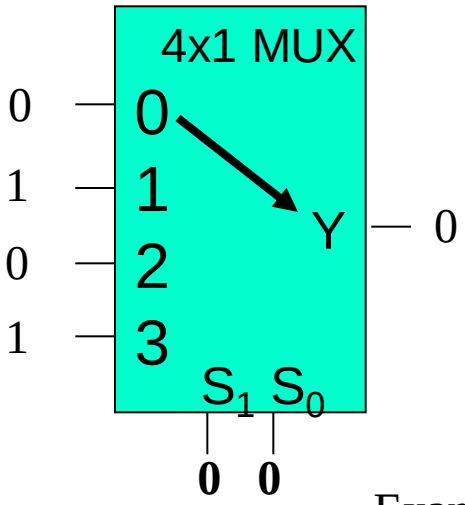


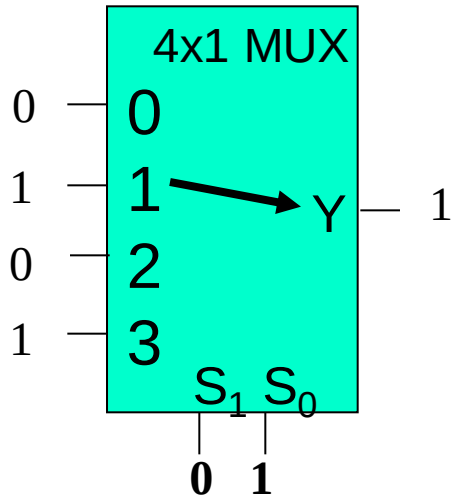
Notice select inputs



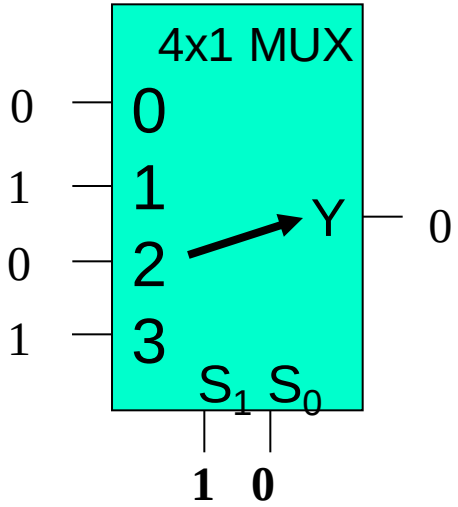


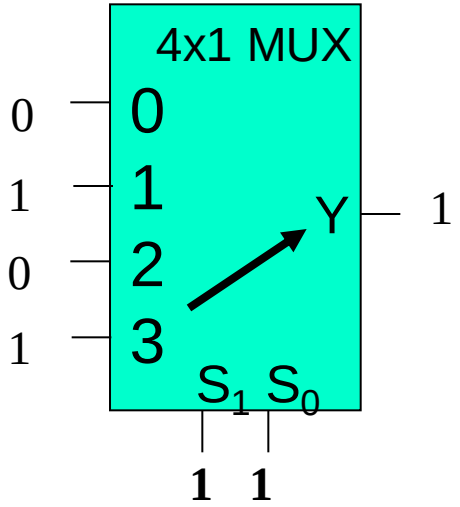
4x1 Multiplexer - 4





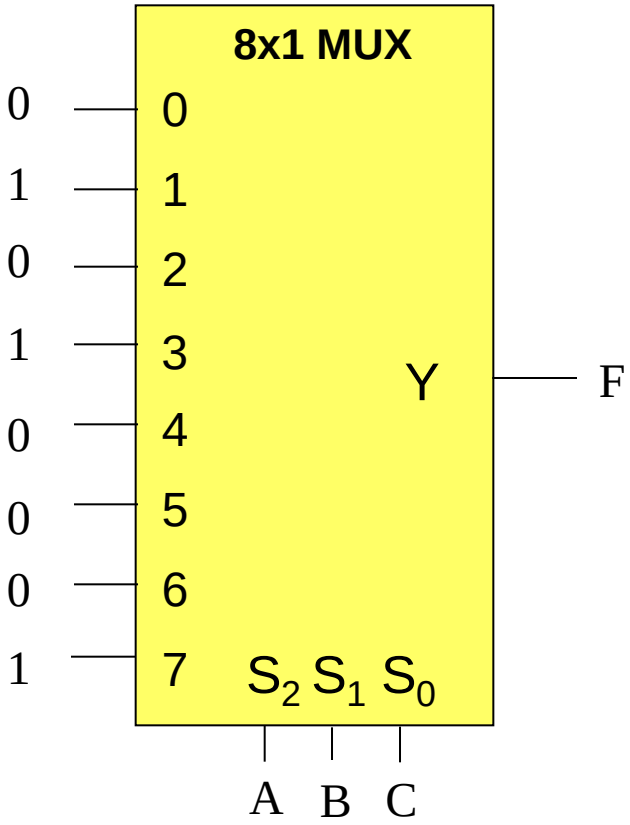
Example- arbitrary inputs.





Implementing a Boolean Function with a Multiplexer - 1

Example: Implement the function F(A,B,C) = Σm(1,3,7) on an 8x1 multiplexer (MUX) with an active-high output

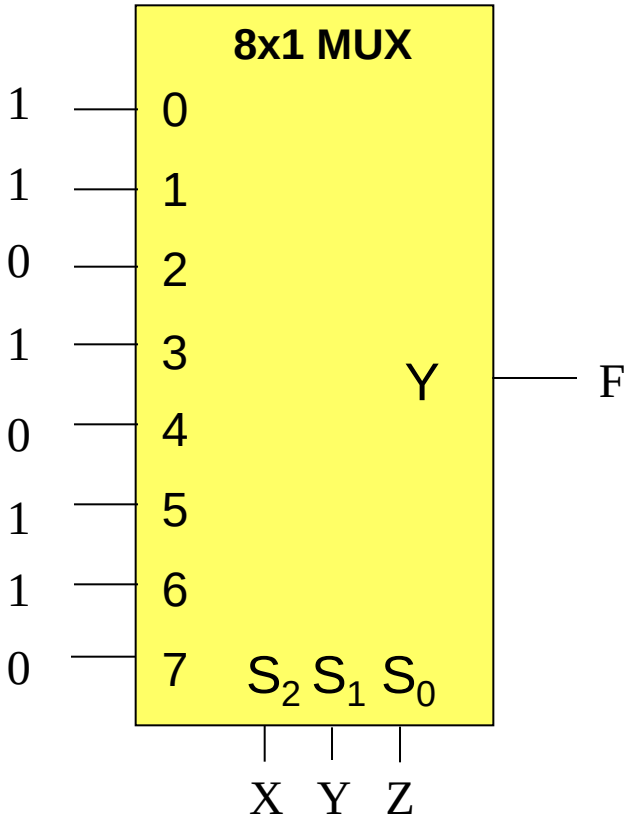


Notes:

- Set data inputs to 1's (+ 5 volts) or 0's (0 volts) based on truth table (minterms)
- Connect input variables (e.g. A,B,C) to select inputs of MUX (be careful with lsb)
- Set MUX output Y to output F
- Distinguish between internal labels (inside block diagram) and external labels (given in problem statement).
- Internal labels (variables) remain the same for every problem
- No external gates are necessary
- Generalize to any 3-variable function

Implementing a Boolean Function with a Multiplexer - 2

Example: Implement the function $F(X,Y,Z) = \Sigma m(0,1,3,5,6)$ on an 8x1 multiplexer (MUX) with an active-high output

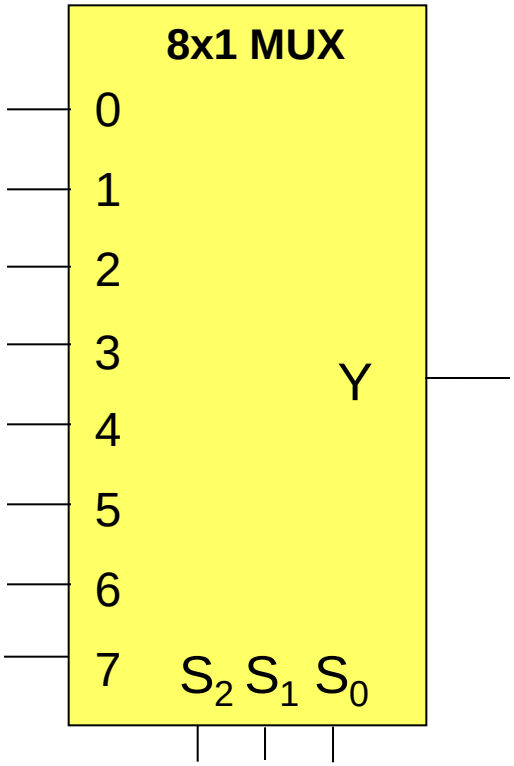


Notes:

- Set data inputs to 1's (+ 5 volts) or 0's (0 volts) based on truth table (minterms)
- Connect input variables (e.g. X,Y,Z) to select inputs of MUX (be careful with lsb)
- Set MUX output Y to output F
- Do not leave any inputs unconnected! (floating)
- It is ok to leave an output unconnected if it is unused.
- Generalize to any 3-variable function

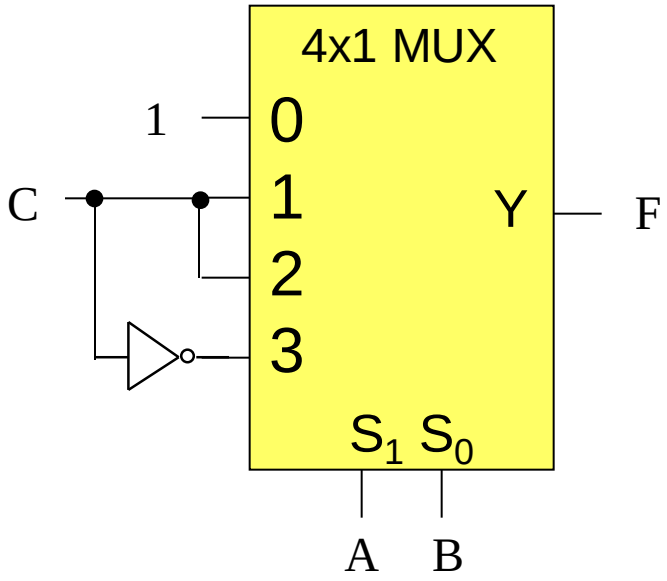
Implementing a Boolean Function with a Multiplexer - 3

Practice Exercise: Implement the function $F(A,B,C) = \Sigma m(2,6,7)$ on an 8x1 multiplexer (MUX) with an active-high output (default).



Implementing a Boolean Function with a Multiplexer - 4

Example: Implement the function F(A,B,C) = Σm(0,1,3,5,6) on a 4x1 multiplexer (MUX) with an active-high output. Observation: a 4x1 MUX has only 2 select inputs! I need 3. It can be done!

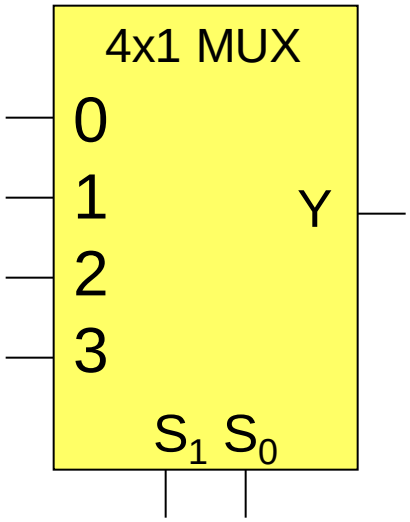


A	B	C	F	
0	0	0	1	F = 1
0	0	1	1	
0	1	0	0	F = C
0	1	1	1	
1	0	0	0	F = C
1	0	1	1	
1	1	0	1	F = C’
1	1	1	0	

Note: 4 possibilities: F=0, F=1, F=C, F=C’

Implementing a Boolean Function with a Multiplexer - 5

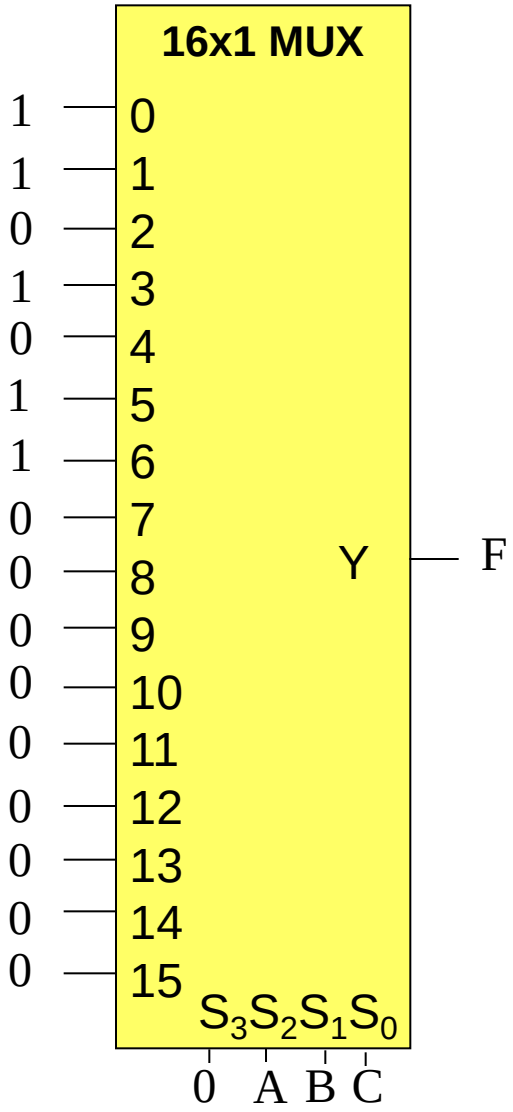
Practice Exercise: Implement the function F(A,B,C) = Σm(2,6,7) on a 4x1 multiplexer (MUX) with an active-high output. Observation: a 4x1 MUX has only 2 select inputs! I need 3. It can be done!



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

Note: 4 possibilities: F=0, F=1, F=C, F=C'

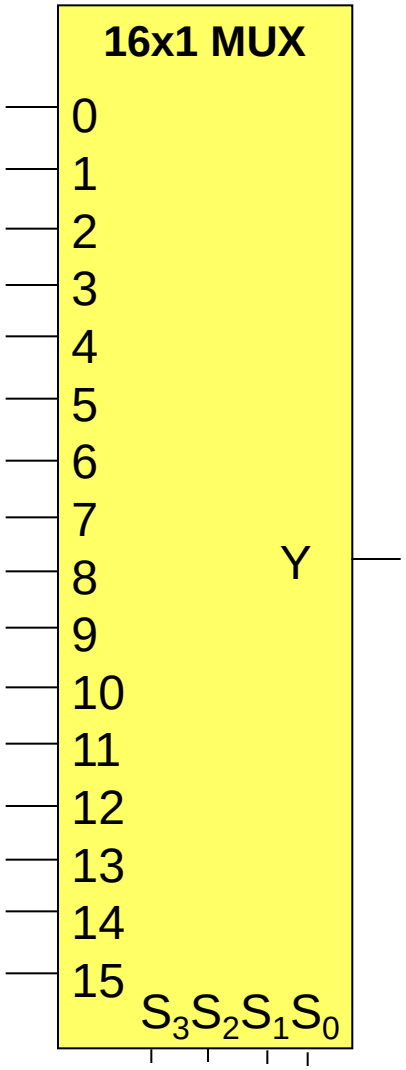
Implementing a Boolean Function with a Multiplexer - 6



Example: Implement the function F(A,B,C) = Σm(0,1,3,5,6) on a 16x1 multiplexer (MUX). Observation: a 16x1 MUX has 4 select inputs! I only need 3. It can be done!

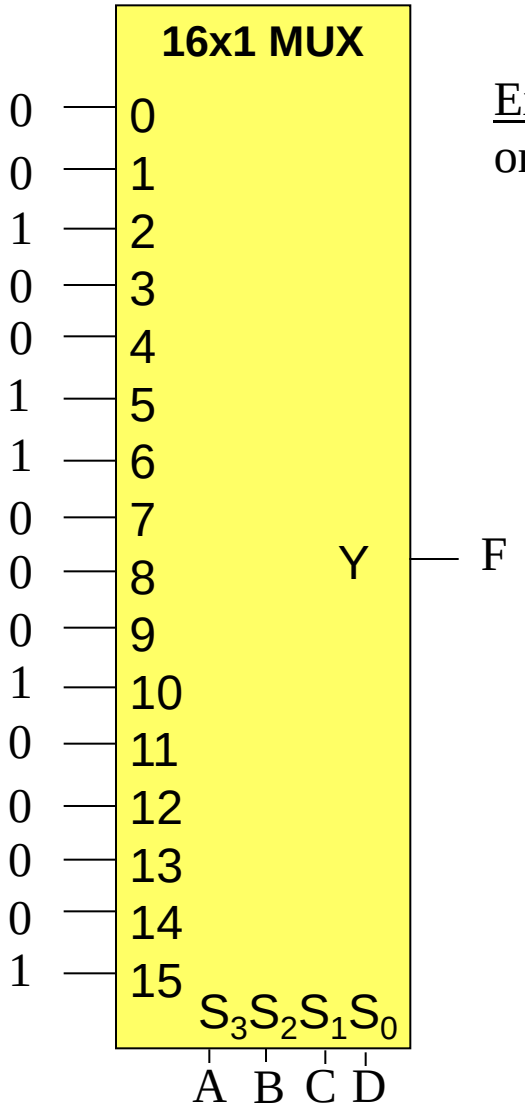
Notes:
- A,B,C connected to least significant select inputs
- msb of select input set to zero (do not leave unconnected)
- unused inputs (8-15) set to zero

Implementing a Boolean Function with a Multiplexer - 7



Practice Exercise: Implement the function F(A,B,C) = Σm(2,6,7) on a 16x1 multiplexer (MUX).

Implementing a Boolean Function with a Multiplexer - 8



Example: Implement the function F(A,B,C,D) = Σm(2,5,6,10,15) on a 16x1 multiplexer (MUX) with an active-high output.

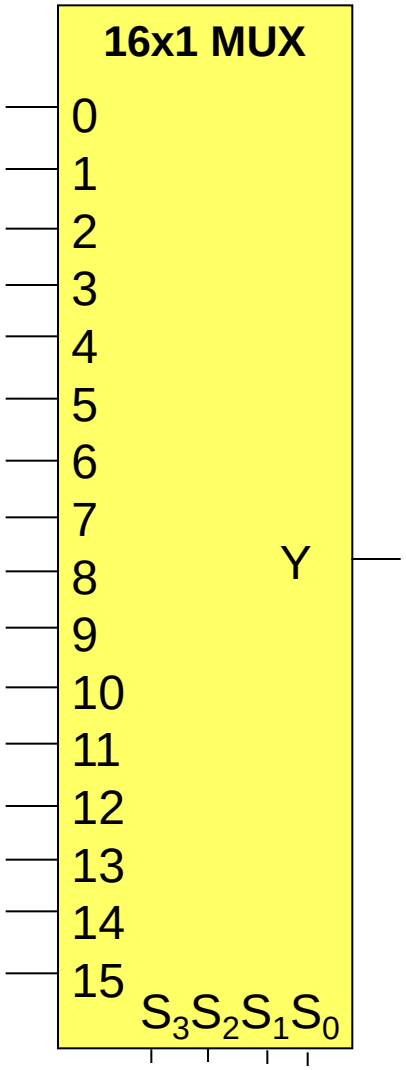
- Solution to part a)

- Observations?

- This circuit has 4 inputs A,B,C,D and one output, F.

- In this case, the # of variables = number of select inputs

Implementing a Boolean Function with a Multiplexer - 9

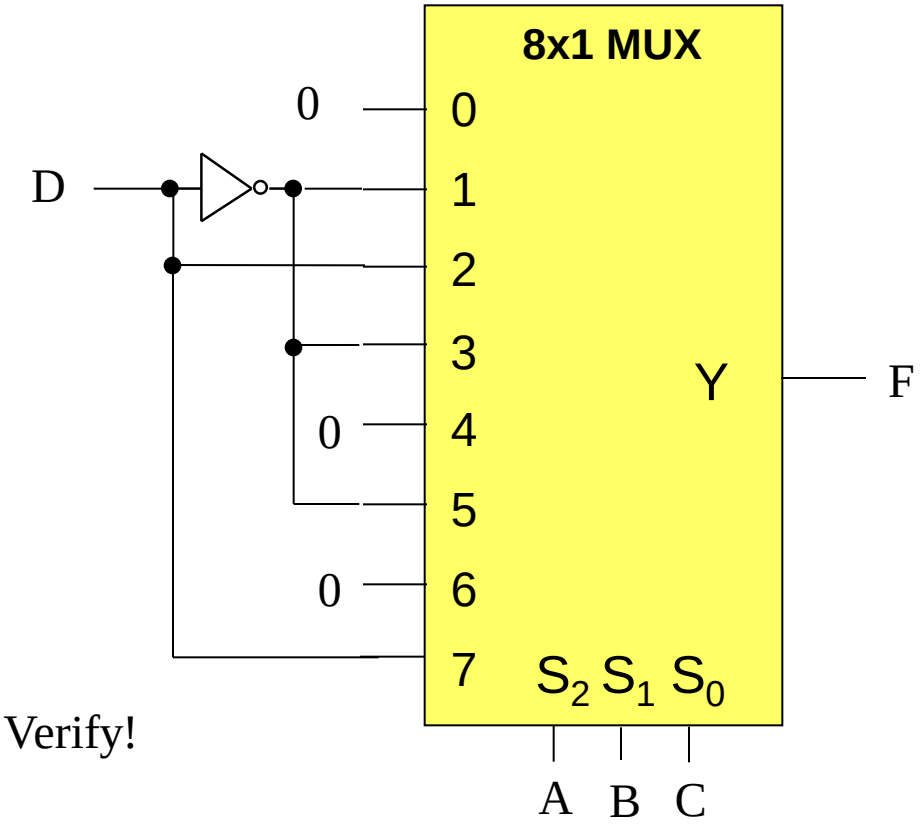


Practice Exercise: Implement the function F(A,B,C,D) = Σm(1,3,8,9,12) on a 16x1 multiplexer (MUX). Label all inputs and outputs.

Implementing a Boolean Function with a Multiplexer - 10

A	B	C	D	F	
0	0	0	0	0	F = 0
0	0	0	1	0	
0	0	1	0	1	F = D’
0	0	1	1	0	
0	1	0	0	0	F = D
0	1	0	1	1	
0	1	1	0	1	F = D’
0	1	1	1	0	
1	0	0	0	0	F = 0
1	0	0	1	0	
1	0	1	0	1	F = D’
1	0	1	1	0	
1	1	0	0	0	F = 0
1	1	0	1	0	
1	1	1	0	0	F = D
1	1	1	1	1	

Example: Implement F(A,B,C,D) = Σm(2,5,6,10,15) with an 8x1 multiplexer (MUX).



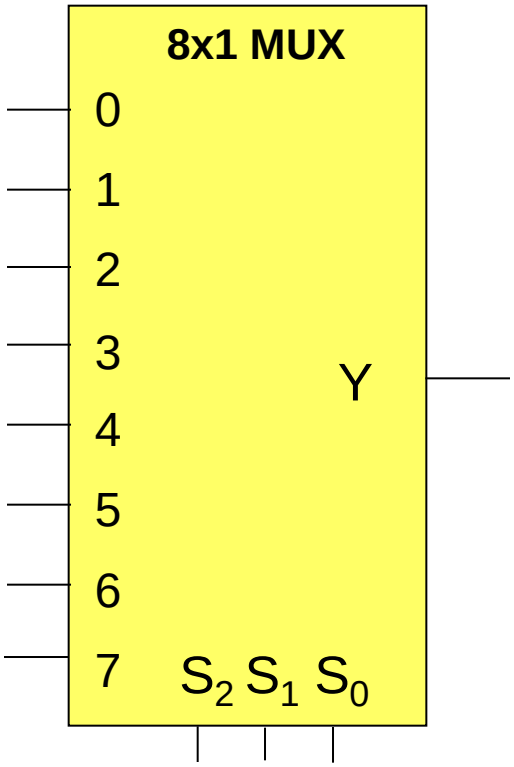
Verify!

- indicates a wire junction

Implementing a Boolean Function with a Multiplexer - 11

A	B	C	D	G
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

Practice Exercise: Implement G(A,B,C,D) = Σm(1,3,8,9,12) with an 8x1 multiplexer (MUX). Label all input and outputs.



Penn State Abington
CMPEN 271
Lecture Set #12
Multiplexers (MUXs), Demultiplexers (DMUXs), Magnitude Comparator

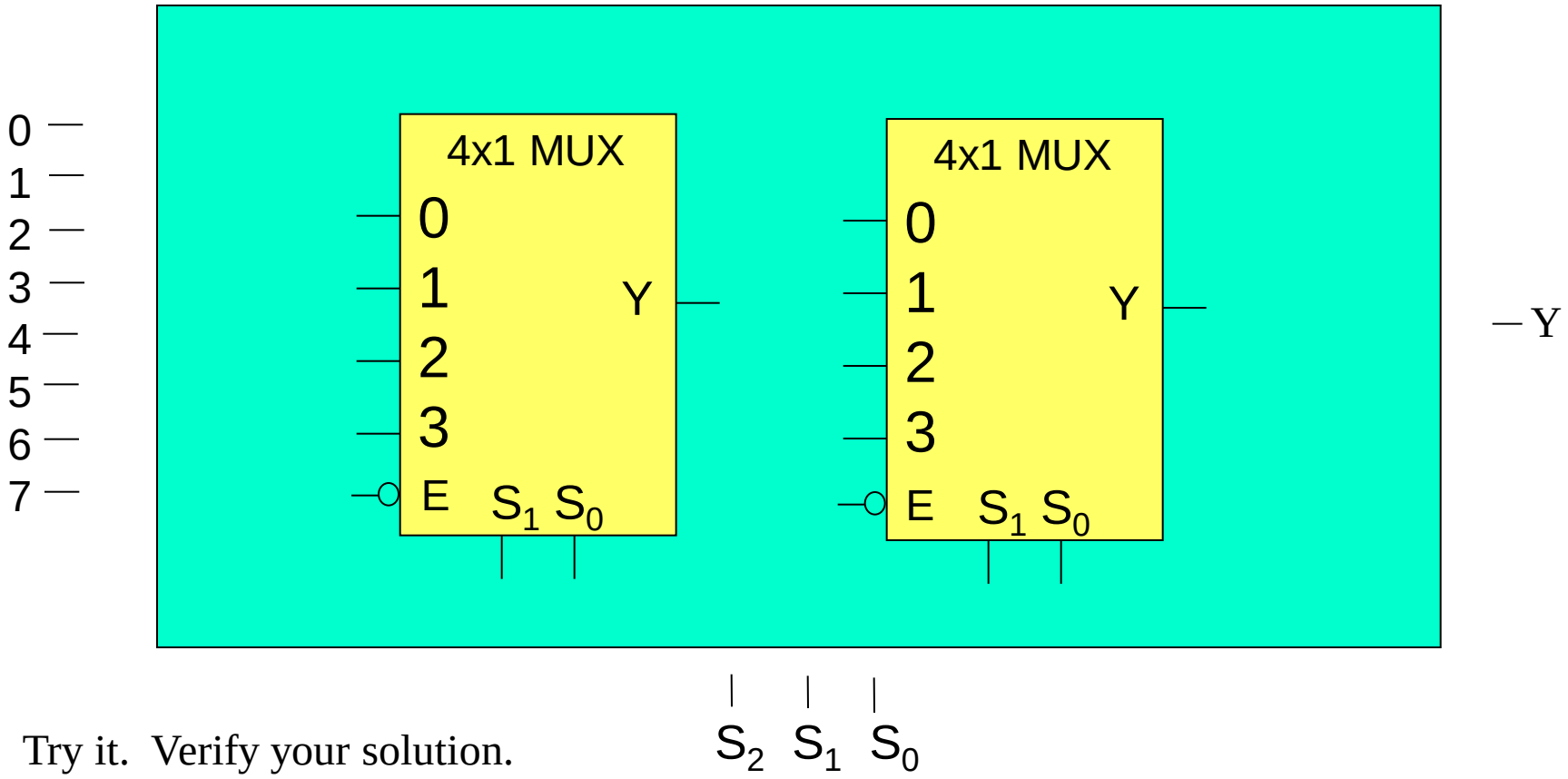
R. Avanzato ©

Topics:

- Multiplexers (MUXs) — Video part 1
- MUX Applications — **Video part 2 ←**
- Simulation of MUXs
- MUX IC Specs
- Demultiplexer (DMUX) — Video part 3
- DMUX Applications
- Magnitude Comparator (with design problem)
- HW #6 (BCD values and 7-segm display) — Video part 4
- Review Questions — Video part 5

Expanding Multiplexers - 1

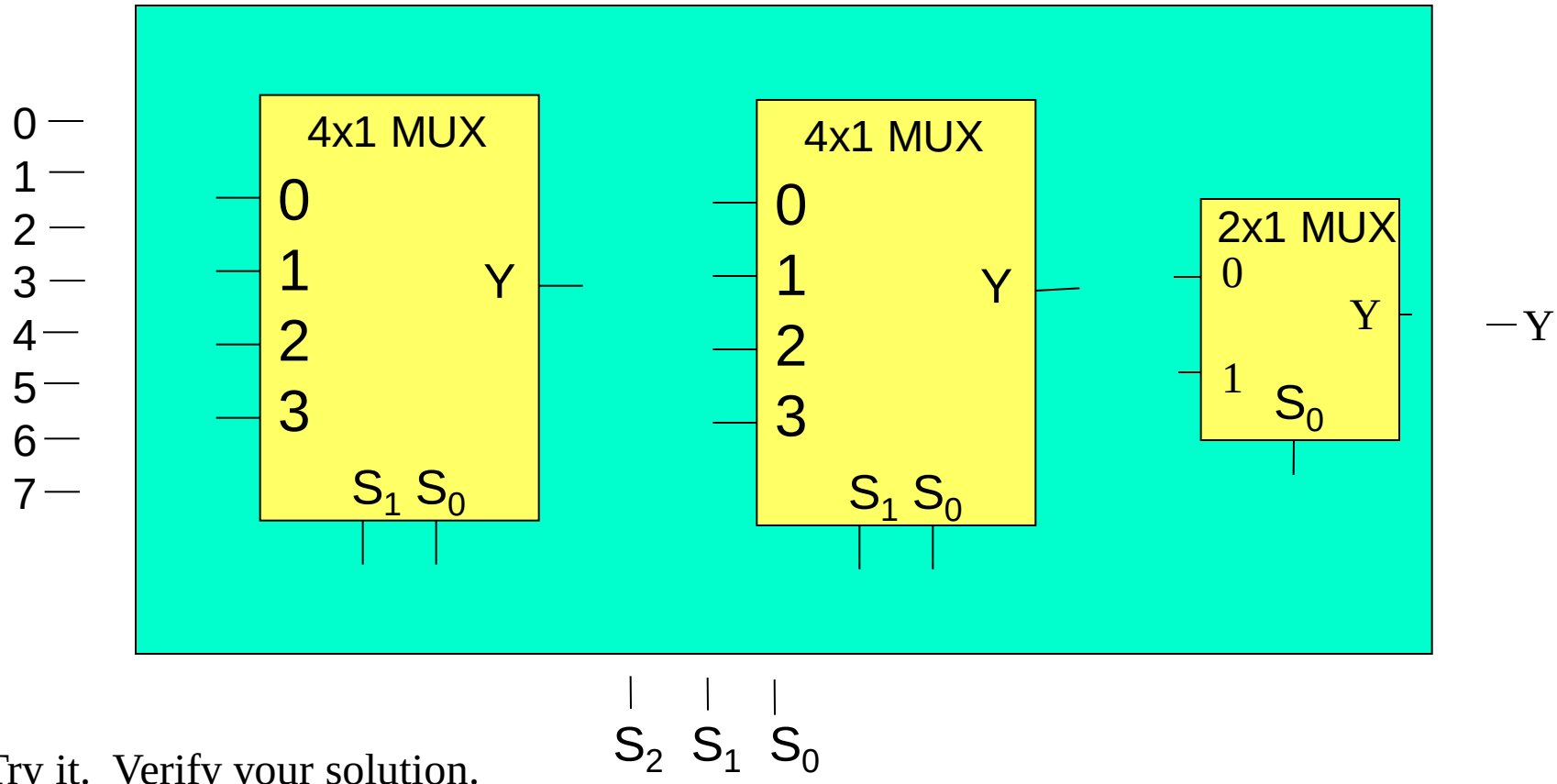
Exercise: Design a 8x1 multiplexer using 2, 4x1 multiplexers, each with an active-low enable. Use any external gates if necessary. Design a minimal circuit.



Try it. Verify your solution.

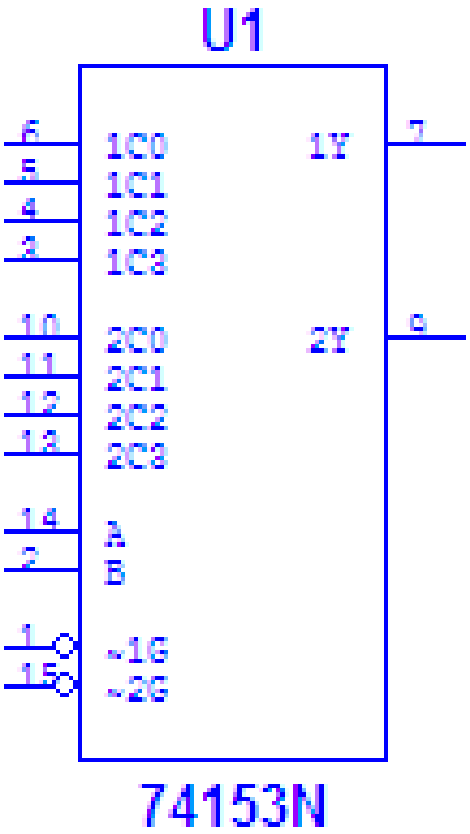
Expanding Multiplexers - 2

Exercise: Design a 8x1 multiplexer using 2, 4x1 multiplexers, both without any enables. We need an additional 2x1 MUX. Design a minimal circuit.

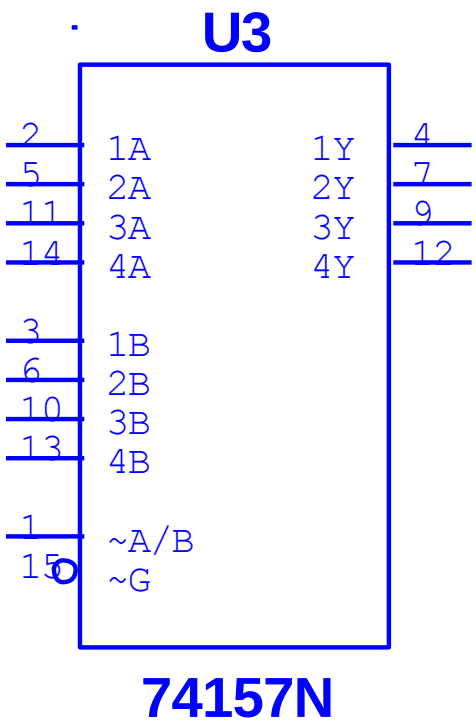


Try it. Verify your solution.

Simulation of Multiplexer (MultiSim)



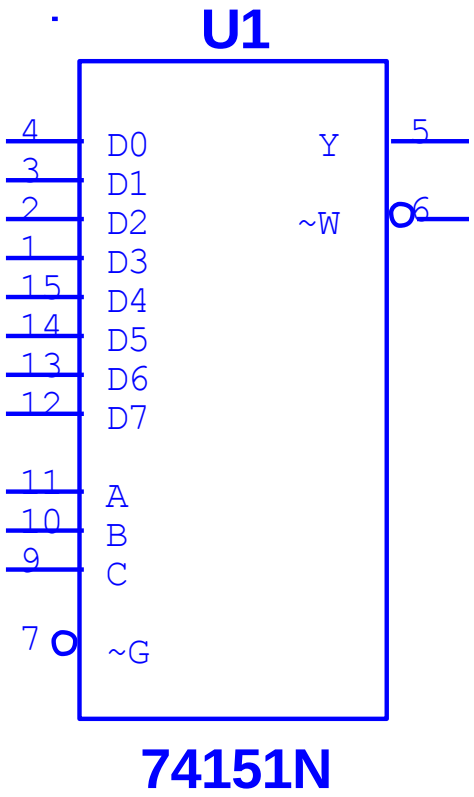
74153 (with pin #'s)
dual 4x1 MUX
(common A→S0, B → S1)
~1G, ~2G → active-low enables
(one enable for each 4x1 MUX)

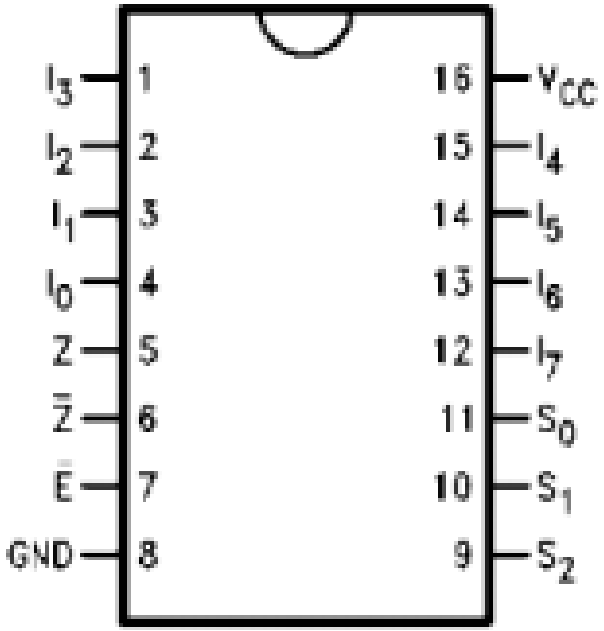


Try it!!

Quad 2x1 MUX
(common ~A/B → S0)
~G → active-low enable

Simulation of Multiplexer (MultiSim)

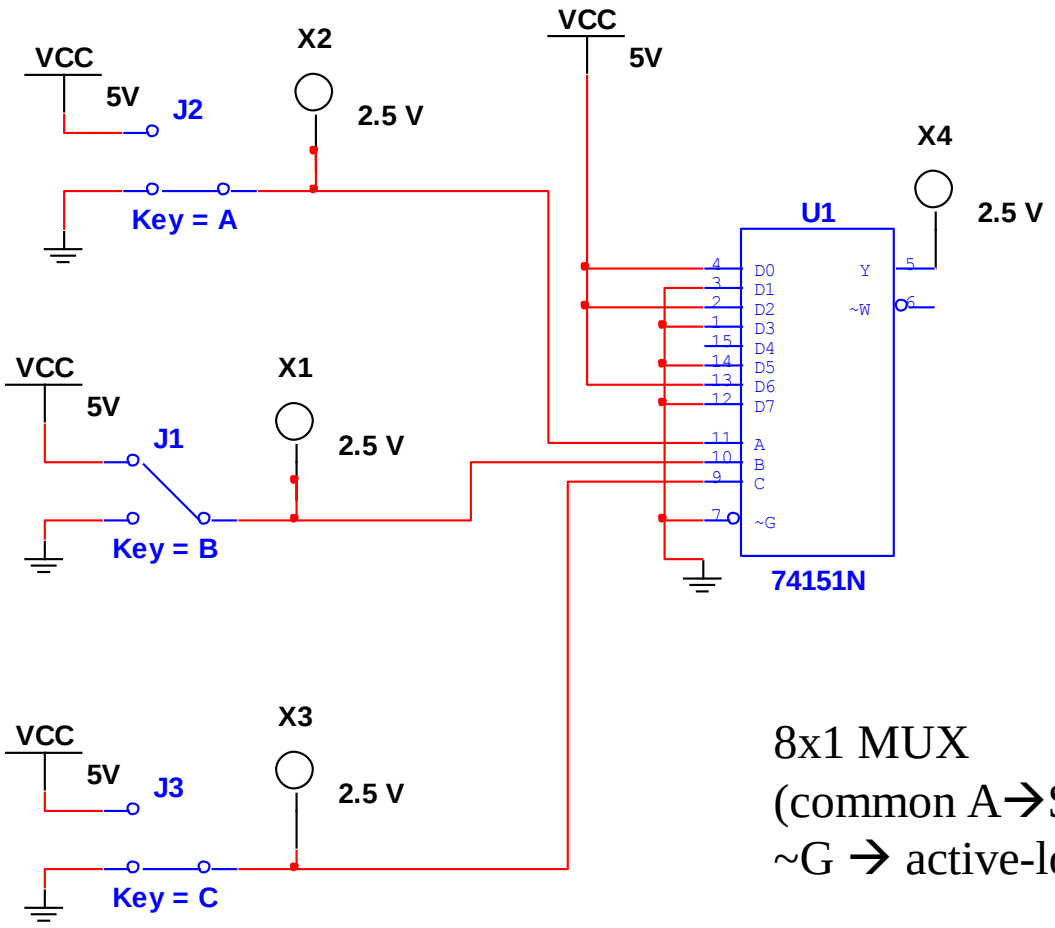




74151 (w/ pin #'s)
8x1 MUX
(common A→S0, B → S1, C→ S2)
~G → active-low enable

Try it!!

Simulation of Multiplexer (MultiSim)



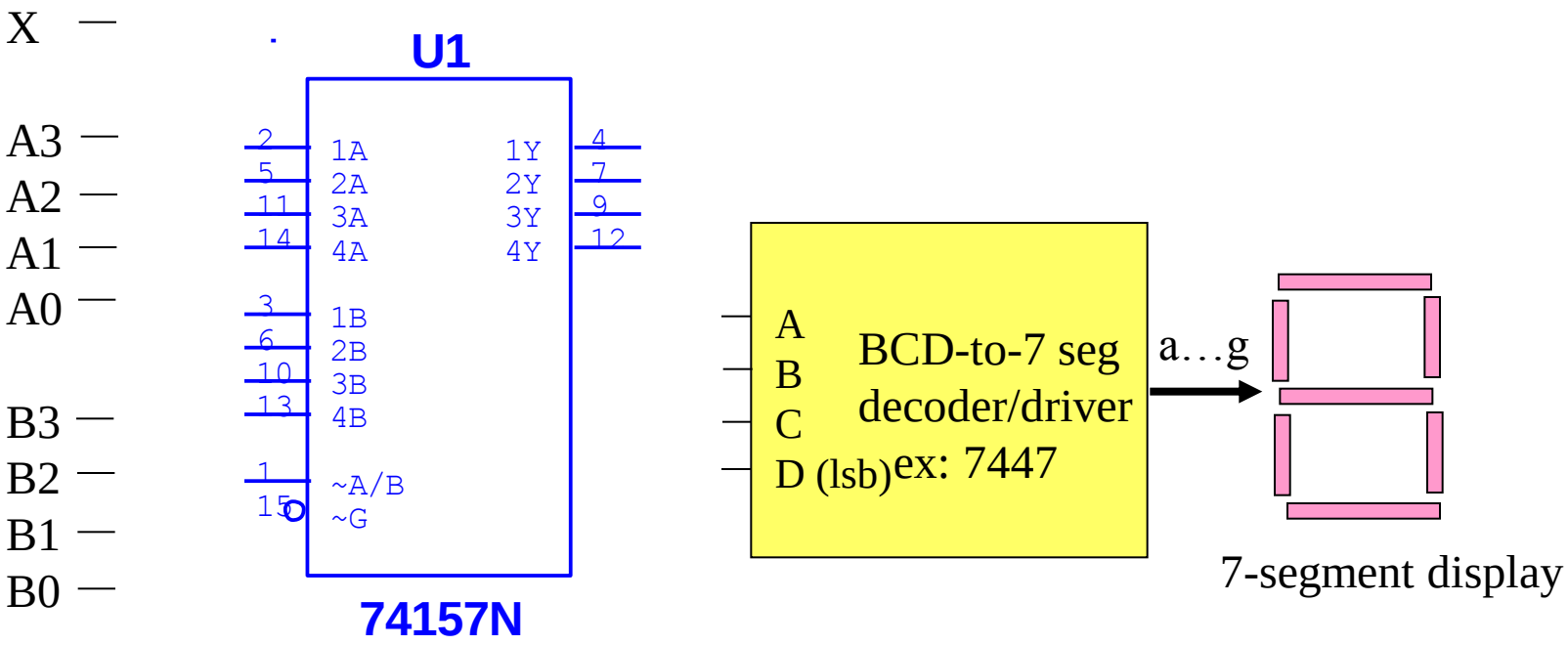
Try it!!

This Multisim file is available on Angel (ask instructor)

8x1 MUX
(common A→S0, B → S1, C→ S2)
~G → active-low enable

Multiplexed Display

Design Problem: Design a circuit that will allow 2, 4-bit BCD values to share a single 7-segment LCD display. Use a quad 2x1 MUX (74157). When the control input bit, X = 0, then display A3 A2 A1 A0. When the control input bit X = 1, display B3 B2 B1 B0. (Relate this to your digital alarm clock display - explain).

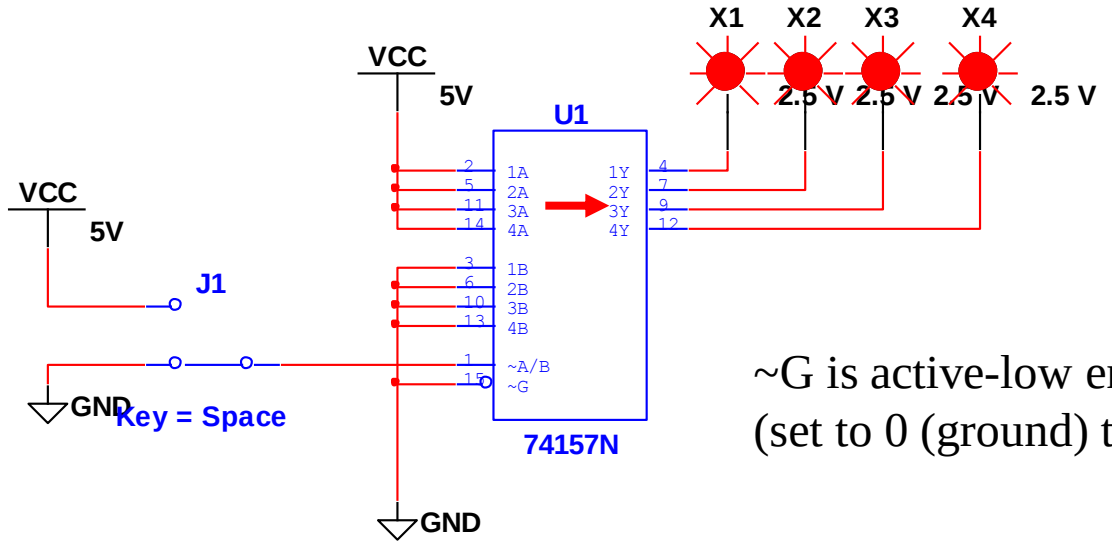


Multiplexed Display (Quad 2x1 MUX)

Fixed inputs for testing:
A = 1111; B = 00000

Case #1: ~A/B = 0

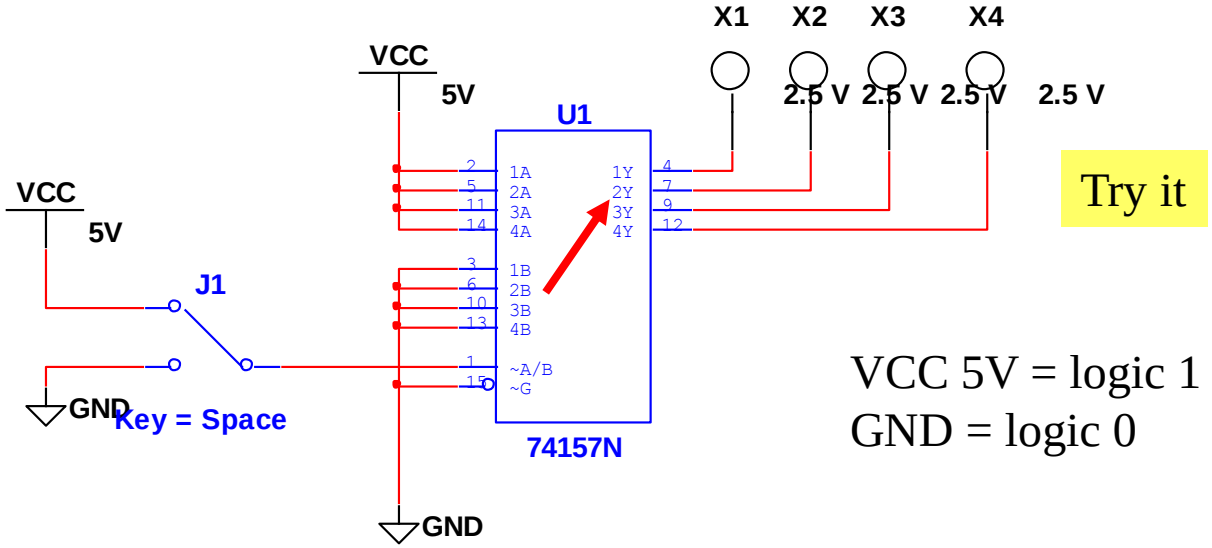
A is displayed at output



~G is active-low enable
(set to 0 (ground) to enable)

Case #2: ~A/B = 1 (5v)

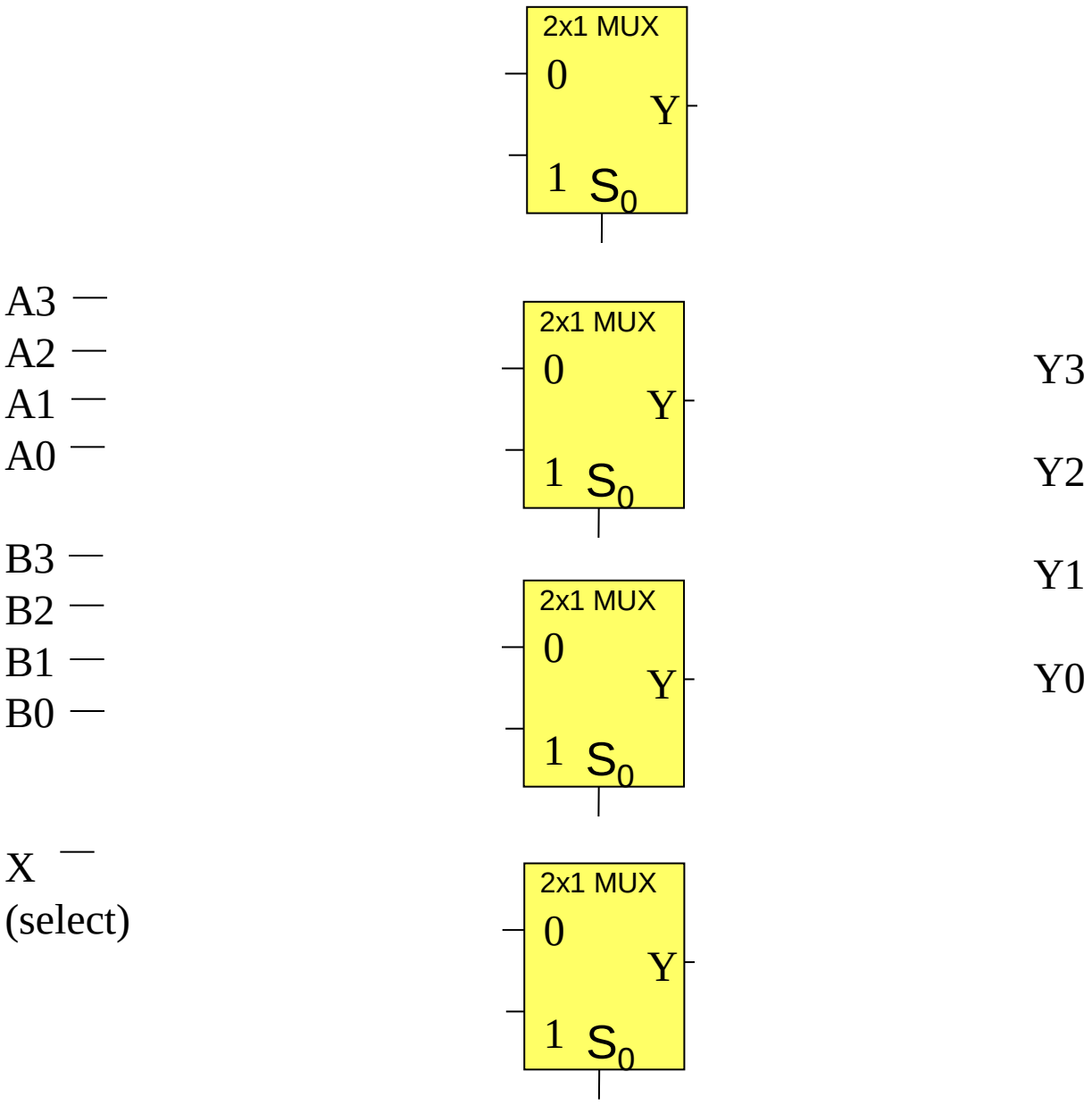
B is displayed at output



Try it

VCC 5V = logic 1
GND = logic 0

Quad 2x1 MUX operation (74157)



Sample MUX Datasheet (ww.ti.com)

SDLS055

SN54153, SN54LS153, SN54S153
SN74153, SN74LS153, SN74S153

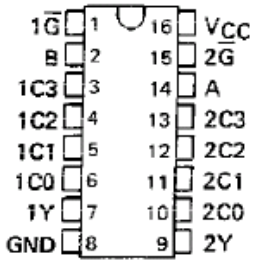
DUAL 4-LINE TO 1-LINE DATA SELECTORS/MULTIPLEXERS

DECEMBER 1972 — REVISED MARCH 1988

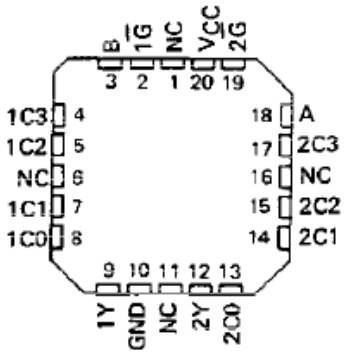
- Permits Multiplexing from N lines to 1 line
- Performs Parallel-to-Serial Conversion
- Strobe (Enable) Line Provided for Cascading (N lines to n lines)
- High-Fan-Out, Low-Impedance, Totem-Pole Outputs
- Fully Compatible with most TTL Circuits

TYPE	TYPICAL AVERAGE PROPAGATION DELAY TIMES			TYPICAL POWER DISSIPATION
	FROM DATA	FROM STROBE	FROM SELECT	
'153	14 ns	17 ns	22 ns	180 mW
'LS153	14 ns	19 ns	22 ns	31 mW
'S153	6 ns	9.5 ns	12 ns	225 mW

SN54153, SN54LS153, SN54S153 . . . J OR W PACKAGE
SN74153 . . . N PACKAGE
SN74LS153, SN74S153 . . . D OR N PACKAGE
(TOP VIEW)



SN54LS153, SN54S153 . . . FK PACKAGE
(TOP VIEW)



description

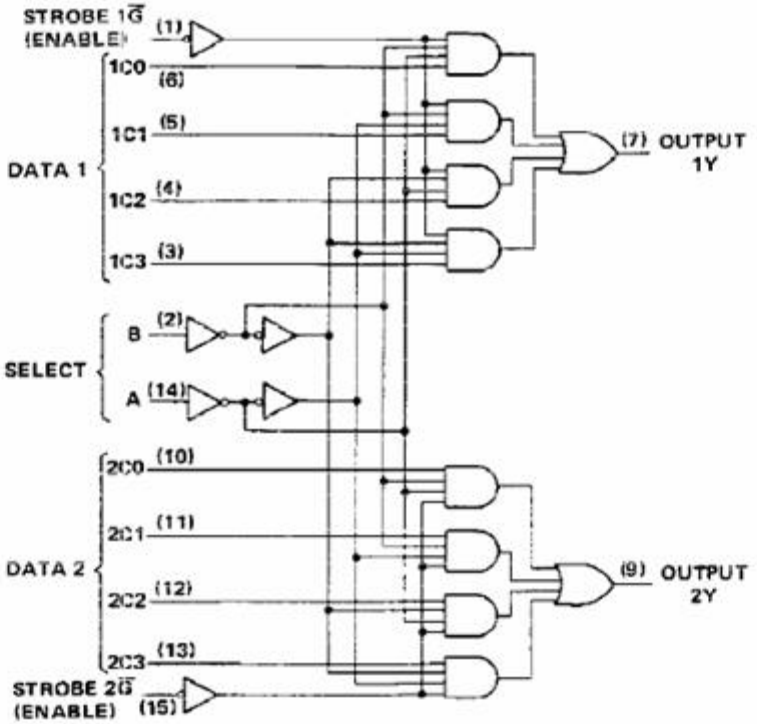
Each of these monolithic, data selectors/multiplexers contains inverters and drivers to supply fully complementary, on-chip, binary decoding data selection to the AND-OR gates. Separate strobe inputs are provided for each of the two four-line sections.

FUNCTION TABLE

SELECT			

Sample MUX Datasheet (ww.ti.com)

logic diagrams (positive logic)



Pin numbers shown are for D, J, N, and W packages.

TEXAS INSTRUMENTS
POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

What do you need to know about Multiplexers?

- Be able to construct **block diagrams** of various multiplexers including appropriate internal labels.
- Understand the **basic operation** of multiplexers, including relationship among data inputs, select inputs, enables, and the output.
- Be able to **implement any Boolean function** with different sized multiplexers. For example, a 3-variable function can be implemented on a mux of sizes 4x1, 8x1, 16x1, 32x1, and higher. A 4-variable function can be implemented on on a mux of size 8x1, 16x1, 32x1 and higher. (Typically, you would choose to implement a function on the smallest MUX available to save costs. There are times when the appropriate MUX is not available.)
- Be able to **construct a larger MUX from two, smaller MUX devices**. For example, be able to design a 4x1 MUX from two, 2x1 MUX's. Design a 8x1 mux from two, 4x1 mux's. Design a 16x1 MUX from two, 8x1 MUX's. In each of these cases, there are 2 possibilities -- the smaller mux's may or may not have enables. If the smaller mux's do not have enables, then an additional 2x1 mux may be necessary.
- Understand the **multiplexed display** circuit.

Sample Exercises (MUX)

1) Implement F(A,B,C) = Σm(0, 1, 5) on a 8x1 MUX with an active-low enable. Simulate.

2) Redo #1 above with a single 4x1 MUX with an active-low enable. Simulate.

3) Redo #1 above with a 16x1 MUX. Simulate.

4) Implement F(A,B,C,D) = Σm(1, 2, 14, 15) on a 16x1 MUX. Simulate.

5) Redo #4 above with a single 8x1 MUX with an active-low enable. Simulate.

6) a) Design a 16x1 MUX using two, 8x1 MUXs, both with active-low enables.
b) Use this circuit to implement F(A,B,C,D) = Σm(1, 2, 14, 15) . Simulate.

MUX and DEC Projects

Project #1: Redesign the BCD 7-segment driver circuit with DECs

- How many decoders would you need?
- Would other gates be needed?
- Would you need to do K-Maps?
- Compare complexity and IC chip count to prior solution

Project #2: Redesign the BCD 7-segment driver circuit with MUXs

- How many Muxs would you need?
- Would other gates be needed?
- Would you need to do K-Maps?
- Compare complexity and IC chip count to prior solution

Penn State Abington
CMPEN 271
Lecture Set #12
Multiplexers (MUXs), Demultiplexers (DMUXs), Magnitude Comparator

R. Avanzato ©

Topics:

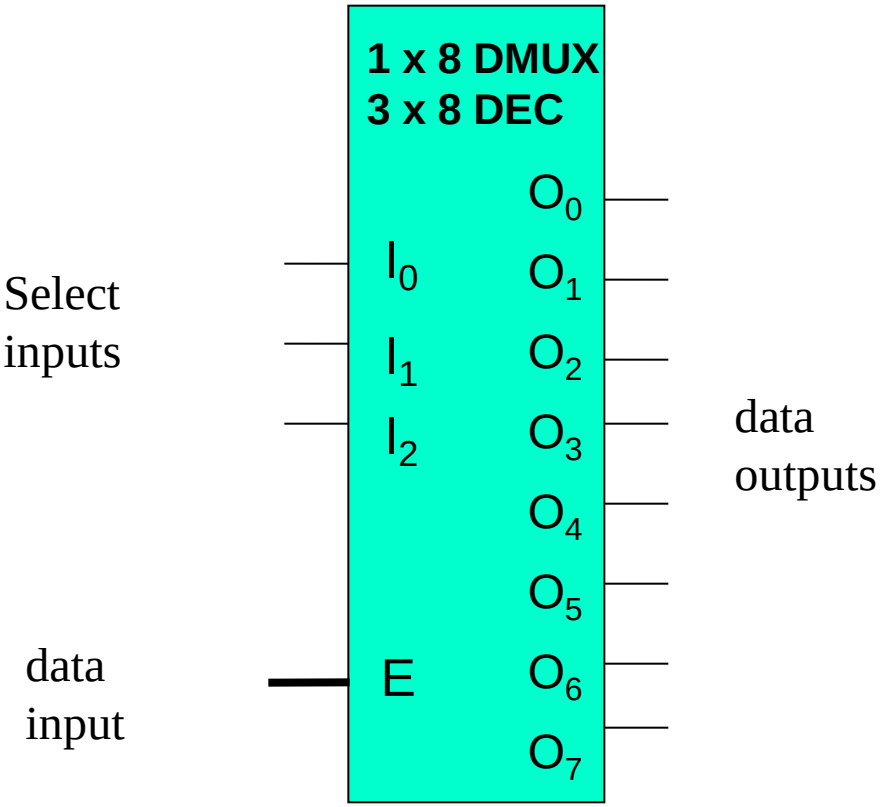
- Multiplexers (MUXs) — Video part 1
- MUX Applications — Video part 2
- Simulation of MUXs
- MUX IC Specs
- Demultiplexer (DMUX) — **Video part 3 ←**
- DMUX Applications
- Magnitude Comparator (with design problem)
- HW #6 (BCD values and 7-segm display) — Video part 4
- Review Questions — Video part 5

What is a Demultiplexer (DMUX)?

- A **demultiplexer** is an MSI device.
- A demultiplexer (DMUX) performs the **"inverse" operation** of a multiplexer
- A DMUX has **one "data" input**, n "select" input lines, and 2^n "data" outputs
- The input is **directed (connected) to one of the n outputs** based on the values of the select inputs
- Demultiplexers come in **sizes** such as 1x4, 1x8, 1x16
- A **1x8 demultiplexer** has one data input, 3 select inputs, and 8 data outputs
- Multiplexers and demultiplexers are sometimes used in **pairs in data communication** applications.
- A **demultiplexer** is in actuality the **same function as a decoder** (with an enable). The single data input to the demultiplexer will be the enable input of the decoder. So, typically, decoders will be called decoders/demultiplexers.

Demultiplexer - 1

Explain circuit.

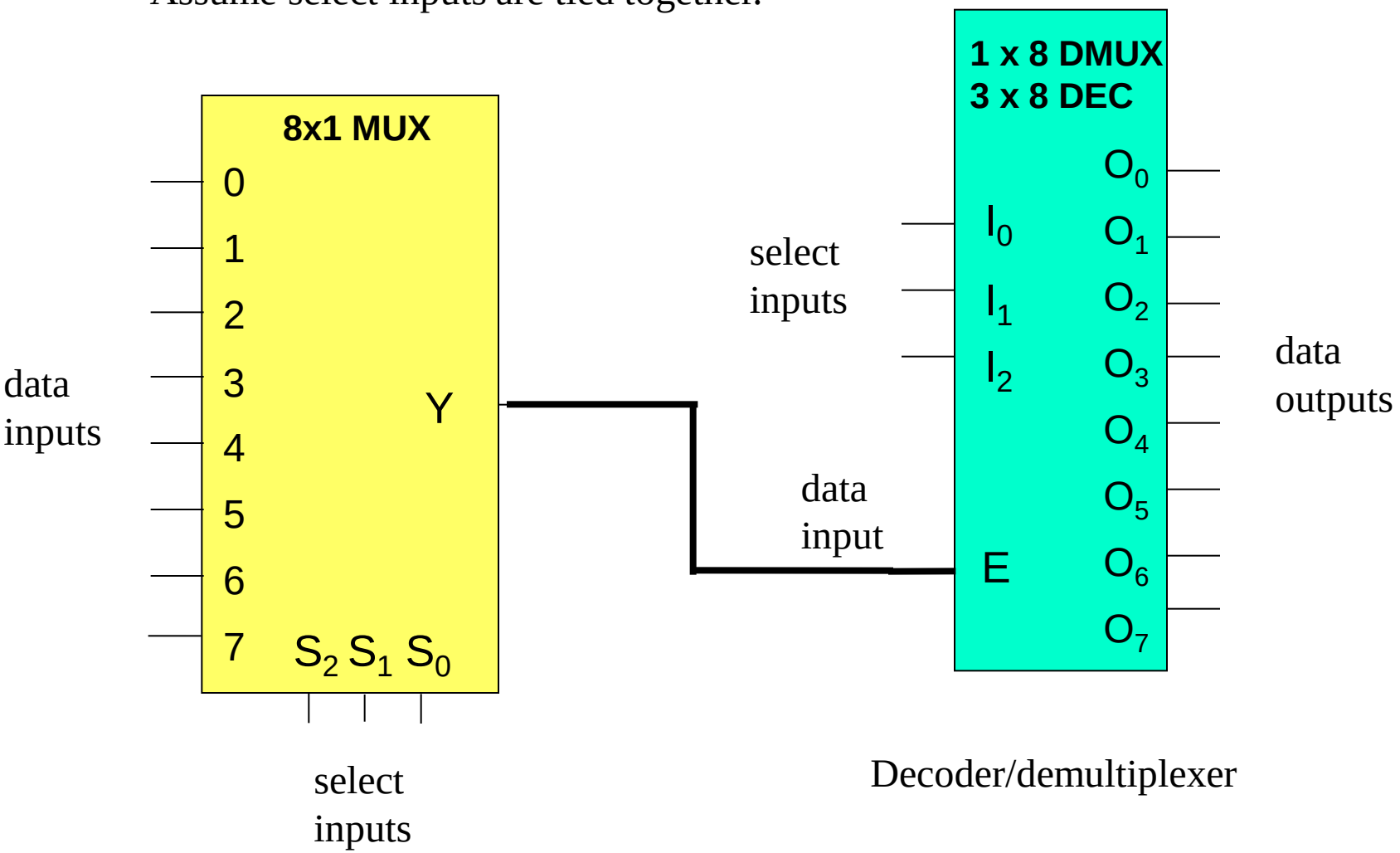


Explain operation

Decoder/demultiplexer

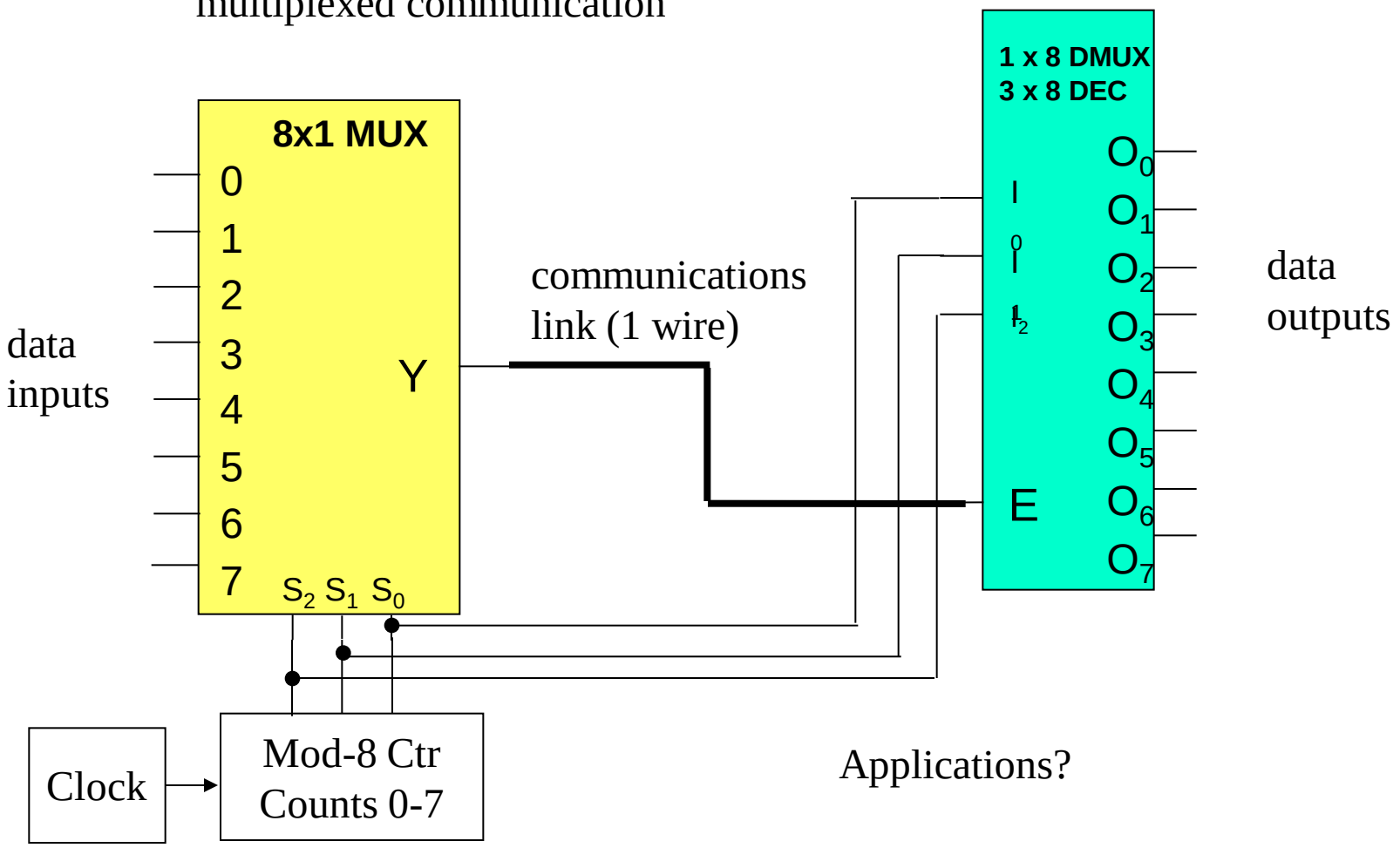
Demultiplexer - 1

Explain circuit. Combine a MUX and DEMUX for communication
Assume select inputs are tied together.



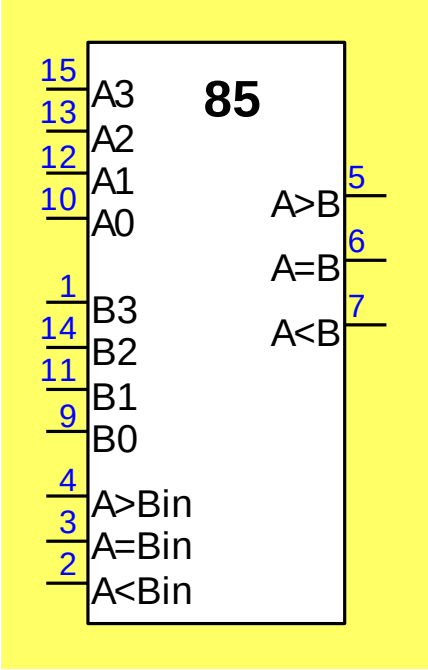
Demultiplexer - 2

Explain advantages and disadvantages of multiplexed communication



Magnitude Comparator - 1

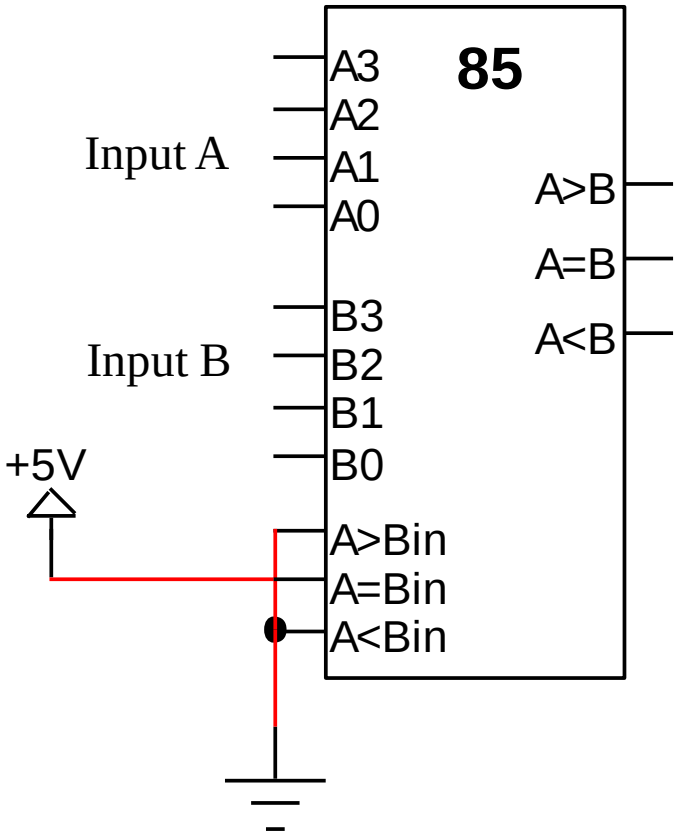
- A magnitude comparator is an MSI device used to compare two binary numbers.
- A magnitude comparator has 3 (active-high) outputs: A>B, A=B, A<B
- The 7485 IC is an example of a commercially available magnitude comparator IC



LogicWorks 4.0 - 7485 with pin numbers

Magnitude Comparator - 2

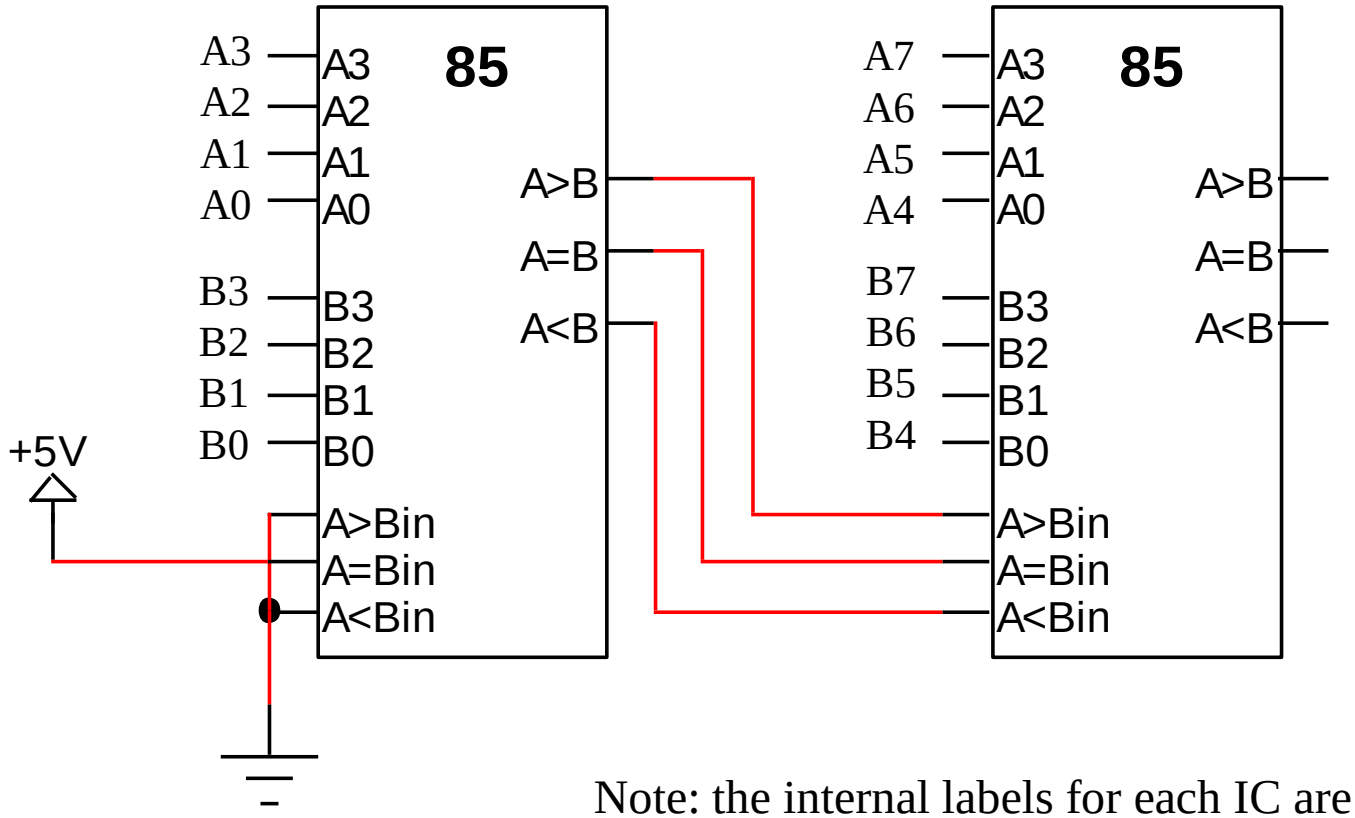
7485 setup for comparing two, 4-bit unsigned values



Applications?
- digital alarm clock
- digital thermostat
- event counter with alarm
- many others

Magnitude Comparator - 3

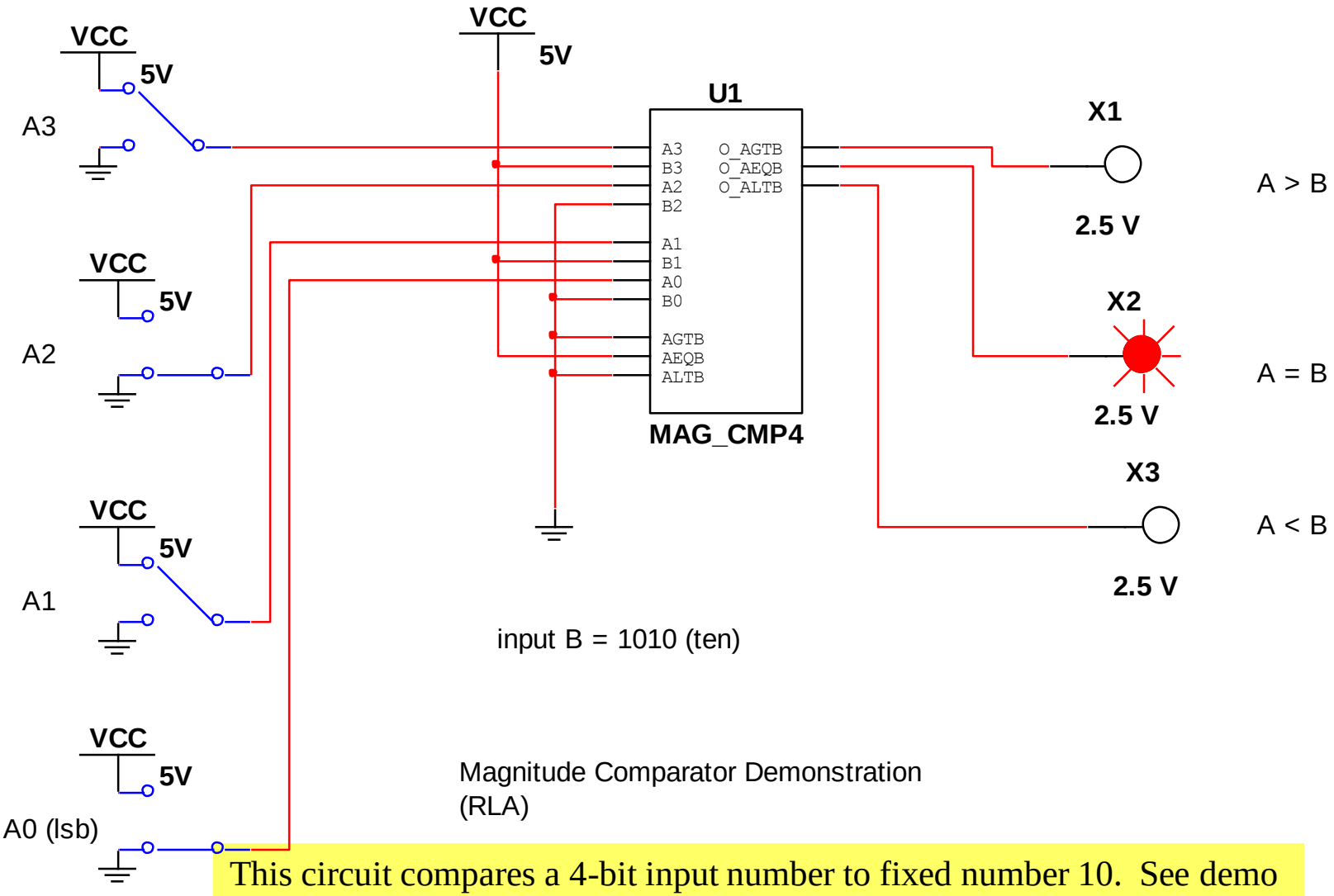
Cascading 7485s to compare two, 8-bit unsigned values:
A7…A0 and B7…B0 See data sheets www.ti.com for more details.



Note: the internal labels for each IC are the same

Magnitude Comparator - 4

MultiSim (use 7485 or Magn Cmp4 component from library)



This circuit compares a 4-bit input number to fixed number 10. See demo

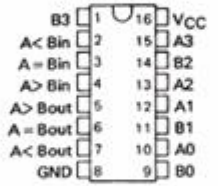
Check demo file on Angel

Sample 7485 IC Datasheet (www.ti.com)

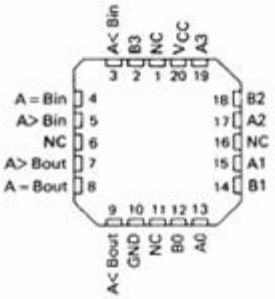
SN5485, SN54LS85, SN54S85
SN7485, SN74LS85, SN74S85
4-BIT MAGNITUDE COMPARATORS
SDLS123 – MARCH 1974 – REVISED MARCH 1988

TYPE	TYPICAL POWER DISSIPATION	TYPICAL DELAY (4-BIT WORDS)
'85	275 mW	23 ns
'LS85	52 mW	24 ns
'S85	365 mW	11 ns

SN5485, SN54LS85, SN54S85 . . . J OR W PACKAGE
SN7485 . . . N PACKAGE
SN74LS85, SN74S85 . . . D OR N PACKAGE
(TOP VIEW)



SN54LS85, SN54S85 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

description

These four-bit magnitude comparators perform comparison of straight binary and straight BCD (8-4-2-1) codes. Three fully decoded decisions about two 4-bit words (A, B) are made and are externally available at three outputs. These devices are fully expandable to any number of bits without external gates. Words of greater length may be compared by connecting comparators in cascade. The A > B, A < B, and A = B outputs of a stage handling less-significant bits are connected to the corresponding A > B, A < B, and A = B inputs of the next stage handling more-significant bits. The stage handling the least-significant bits must have a high-level voltage applied to the A = B input. The cascading paths of the '85, 'LS85, and 'S85 are implemented with only a two-gate-level delay to reduce overall comparison times for long words. An alternate method of cascading which further reduces the comparison time is shown in the typical application data.

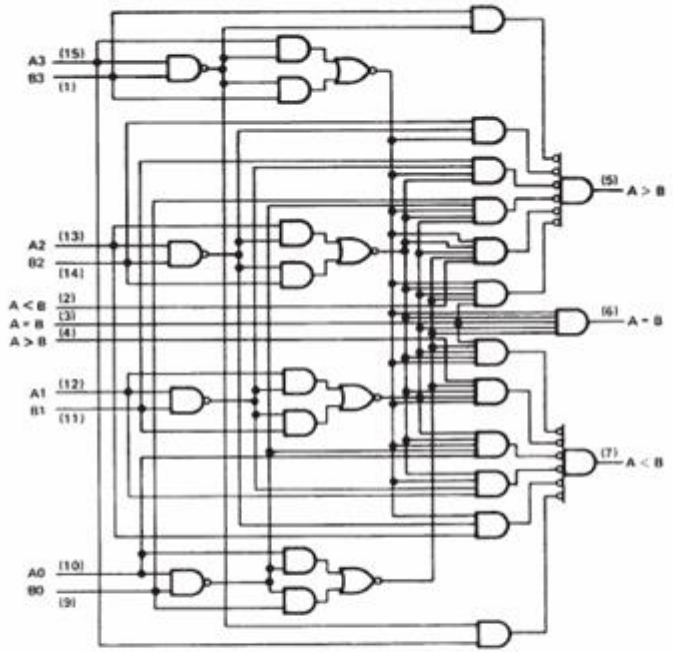
FUNCTION TABLE

COMPARING INPUTS				CASCADING INPUTS			OUTPUTS		
A3, B3	A2, B2	A1, B1	A0, B0	A > B	A < B	A = B	A > B	A < B	A = B
A3 > B3	X	X	X	X	X	X	H	L	L
A3 < B3	X	X	X	X	X	X	L	H	L
A3 = B3	A2 > B2	X	X	X	X	X	H	L	L
A3 = B3	A2 < B2	X	X	X	X	X	L	H	L
A3 = B2	A2 = B2	A1 > B1	X	X	X	X	H	L	L
A3 = B3	A2 = B2	A1 < B1	X	X	X	X	L	H	L
A2 = B3	A2 = B2	A1 = B1	A0 > B0	X	X	X	H	L	L
A3 = B3	A2 = B2	A1 = B1	A0 < B0	X	X	X	L	H	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	H	L	L	H	L	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	L	H	L	L	H	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	X	X	H	L	L	H
A3 = B3	A2 = B2	A1 = B1	A0 = B0	H	H	L	L	L	L
A3 = B3	A2 = B2	A1 = B1	A0 = B0	L	L	L	H	H	L

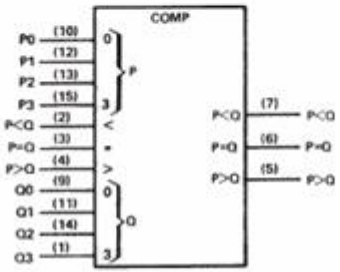
Sample 7485 IC Datasheet (www.ti.com)

SN5485, SN54LS85, SN54S85
SN7485, SN74LS85, SN74S85
4-BIT MAGNITUDE COMPARATORS
SDLS123 – MARCH 1974 – REVISED MARCH 1988

logic diagrams (positive logic)

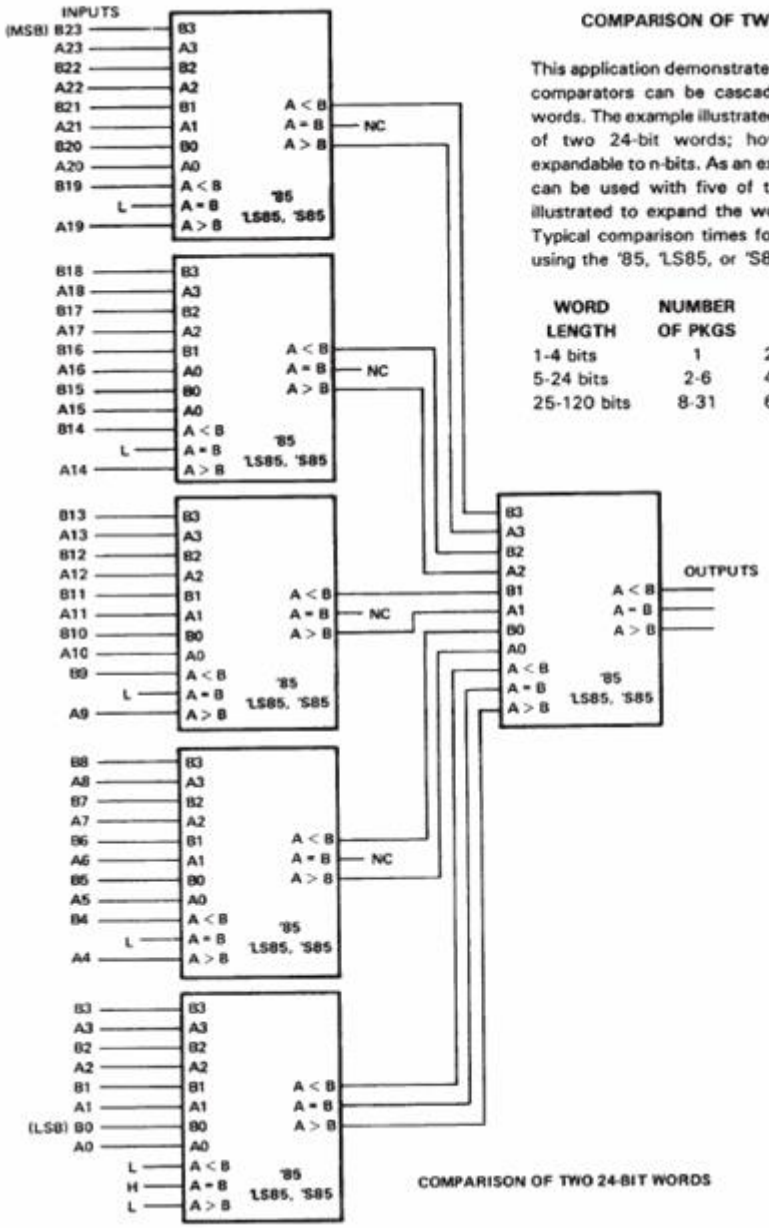


logic symbol†



Sample 7485 IC Datasheet (www.ti.com)

TYPICAL APPLICATION DATA



COMPARISON OF TWO N-BIT WORDS

This application demonstrates how these magnitude comparators can be cascaded to compare longer words. The example illustrated shows the comparison of two 24-bit words; however, the design is expandable to n-bits. As an example, one comparator can be used with five of the 24-bit comparators illustrated to expand the word length to 120-bits. Typical comparison times for various word lengths using the '85, 'LS85, or 'S85 are:

WORD LENGTH	NUMBER OF PKGS	'85	'LS85	'S85
1-4 bits	1	23 ns	24 ns	11 ns
5-24 bits	2-6	46 ns	48 ns	22 ns
25-120 bits	8-31	69 ns	72 ns	33 ns

COMPARISON OF TWO 24-BIT WORDS

Penn State Abington
CMPEN 271
Lecture Set #12
Multiplexers (MUXs), Demultiplexers (DMUXs), Magnitude Comparator

R. Avanzato ©

Topics:

- Multiplexers (MUXs) — **Vid**eo part **1**
- MUX Applications — Video part 2
- Simulation of MUXs
- MUX IC Specs
- Demultiplexer (DMUX) — Video part 3
- DMUX Applications
- Magnitude Comparator (with design problem)
- HW #6 (BCD values and 7-segm display) — **Video part 4** ←
- Review Questions — Video part 5

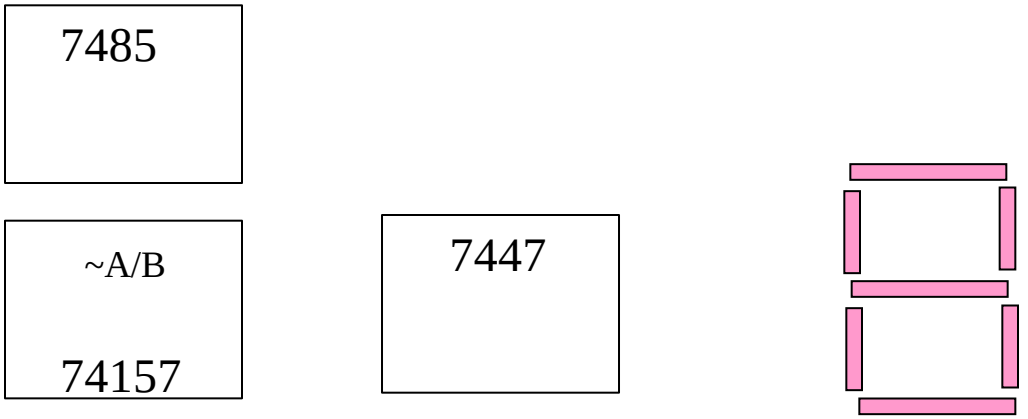
Design Problem (Homework #6)

Design Problem: Design a circuit that accepts 2, 4-bit unsigned BCD values as inputs A3 A2 A1 A0, and B3 B2 B1 B0. The circuit will numerically **compare the two BCD values**, and display the greater value on a single 7-segment display. **Use MSI devices when possible**. Simulate circuit to verify operation. Include appropriate test cases,

Could we solve this problem with a truth table and K-Maps?

What MSI devices should we use?

What is general architecture of the solution?



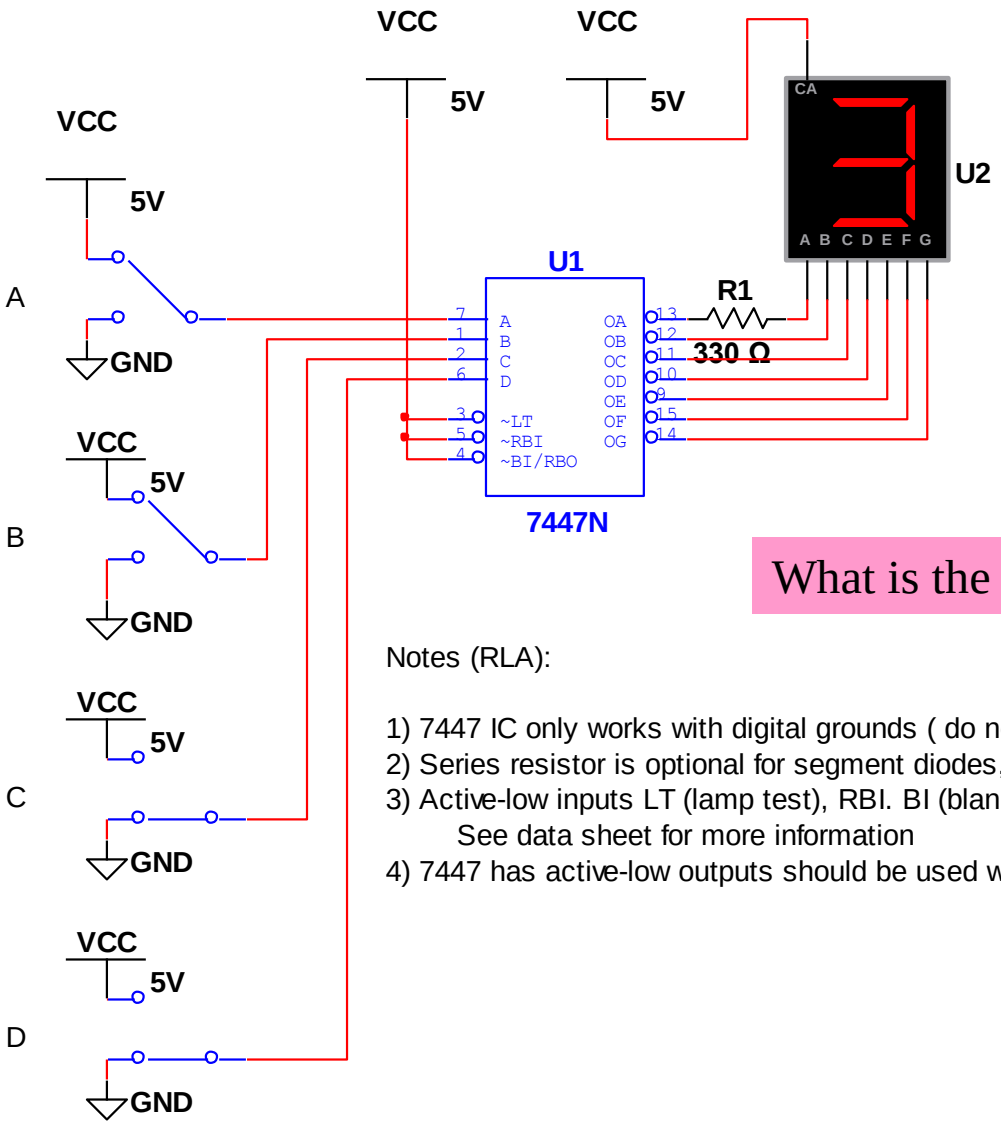
HW#6 (see schedule for due date)

- **Design Problem**: Design a circuit that accepts 2, 4-bit unsigned BCD values as inputs A3 A2 A1 A0, and B3 B2 B1 B0. The circuit will numerically compare the two BCD values, and display the greater value on a single 7-segment display. Use MSI devices when possible. Simulate circuit to verify operation.

- NOTE: **use 7447 BCD-to-7segment drivers** may be used (active low outputs) and common-anode 7-segment display. Use 74157 quad 2x1 multiplexer chip. Use 7485 (or MagCmp4) chip.
  1) 7447 IC only works with **digital grounds** in MultiSim (do not use conventional ground)
  2) Series resistor is optional for segment diodes, but resistance must be 470ohms or less (you may use resistors, since they would be required in a real world circuit.)
  3) Active-low inputs LT (lamp test), RBI. BI (blanking) should be set to +5 (VCC).
     See data sheet for more information
  4) 7447 has active-low outputs should be used with a common anode segment display
     (The common anode CA pin of the 7-segment display should be connected to +5v = VCC)

- Submit solution (including simulation) to Angel (1 Word or PDF document only)
- Use MultiSIM (or equivalent) simulation of all problems. **Include appropriate test cases**.
- Use standard policies and format for homework
- Total of 1 design problem due.

HW#6 (see schedule for due date)



Try This
Circuit is on Angel

What is the lsb on the 7447?

Notes (RLA):

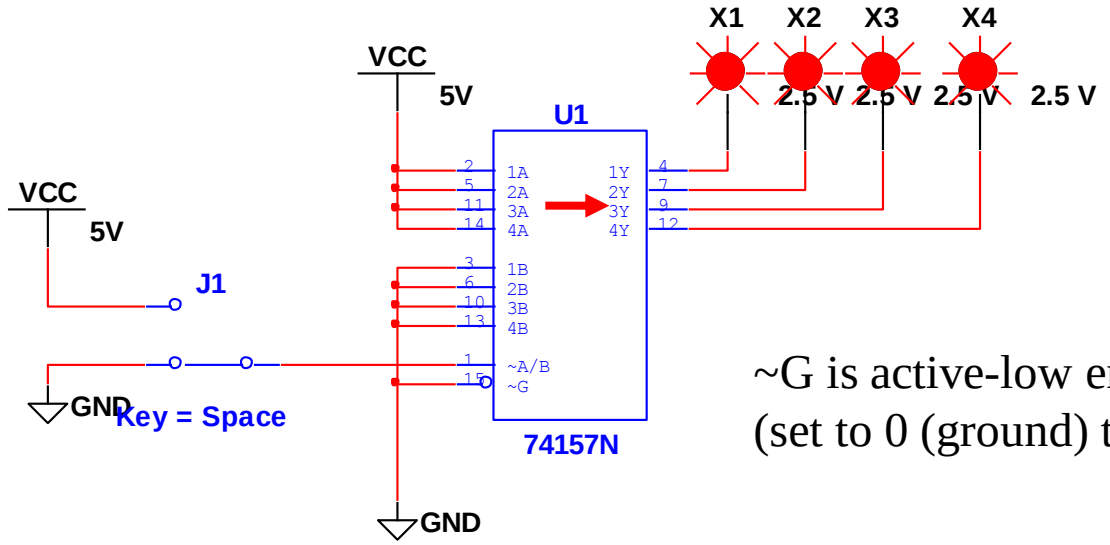
1) 7447 IC only works with digital grounds (do not use conventional ground)
2) Series resistor is optional for segment diodes, but resistance must be 470ohms or less
3) Active-low inputs LT (lamp test), RBI. BI (blanking) should be set to +5 (VCC).
   See data sheet for more information
4) 7447 has active-low outputs should be used with a common anode segment display

HW #6 Quad 2x1 MUX (74157)

Fixed inputs for testing:
A = 1111; B = 00000

Case #1: ~A/B = 0

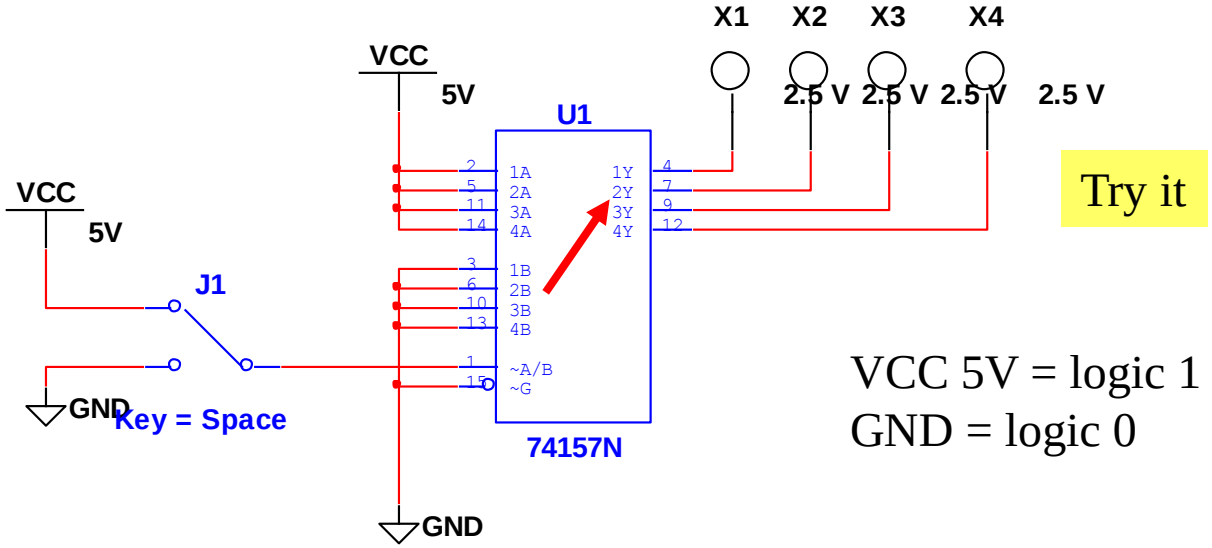
A is displayed at output



~G is active-low enable
(set to 0 (ground) to enable)

Case #2: ~A/B = 1 (5v)

B is displayed at output



Try it

VCC 5V = logic 1
GND = logic 0

Summary

- A multiplexer is an MSI device that is used as a building block to create more complex digital circuits.
- A multiplexer is often referred to as a "data switch" because it redirects one of its many inputs to its single output. The particular input that is switched to the output is controlled by the input control wires.
- A multiplexer has n data input lines, one data output, and log2 n control input lines. For example, an 8x1 multiplexer has 8 data inputs, one data output and 3 input select lines.
- Multiplexers commonly appear in sizes 2 x1, 4 x 1, 8 x 1, 16 x 1, etc. (Notes: 4 x 1 means 4 inputs and 1 output; All multiplexers have one output.)
- Multiplexers can be used to implement Boolean functions (no algebra or K-Maps are needed when you use a multiplexer).
- Multiplexers can be used to redirect input to more than one output device such as a 7-segment LED display.
- Multiplexers can be combined to form larger multiplexers. For example, two 4 x 1 multiplexers can be combined to form on 8 x 1 multiplexer.
- Multiplexers are represented as block diagrams showing internal labels for data inputs, select inputs and the one output. There might also be an optional active-low enable or an active-high enable.
- A demultiplexer performs the "inverse" operation of a multiplexer. A demultiplexer has one input and n outputs. The signal at the input will be sent to one of the outputs based on the select lines.
- A demultiplexer is actually a decoder. The enable of the decoder is used as the single input of the demultiplexer.

Further Reading

- Mano,Kime, Logic and Computer Design Fundamentals, 2nd edition, Prentice Hall (Chapter 3).

- Wakerly, John F., Digital Design: Principles and Practices.

Penn State Abington
CMPEN 271
Lecture Set #12
Multiplexers (MUXs), Demultiplexers (DMUXs), Magnitude Comparator

R. Avanzato ©

Topics:

- Multiplexers (MUXs) — Video part 1
- MUX Applications — Video part 2
- Simulation of MUXs
- MUX IC Specs
- Demultiplexer (DMUX) — Video part 3
- DMUX Applications
- Magnitude Comparator (with design problem)
- HW #6 (BCD values and 7-segm display) — Video part 4
- Review Questions — Video part 5 ←

Sample Questions

#1.- **A 4x1 multiplexer has how many data inputs?**

a) 1 b) 2 c) 3 d) 4

#2.- **A 4x1 multiplexer has how many data outputs?**

a) 4 b) 3 c) 2 d) 1

#3.- **A 4x1 multiplexer has how many select/control inputs?**

a) 1 b) 2 c) 4 d) 8

#4.- **A 1x4 demultiplexer has how many data outputs?**

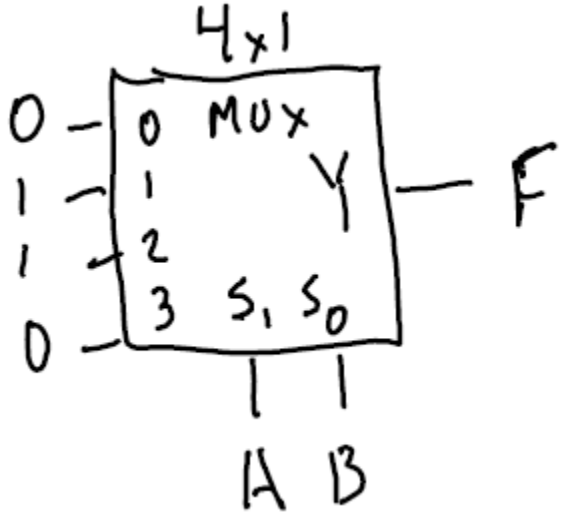
a) 1 b) 2 c)3 d) 4

#5.- **A 1x4 demultiplexer has how many select/control inputs?**

a) 1 b) 2 c) 4 d) 8

Sample Questions

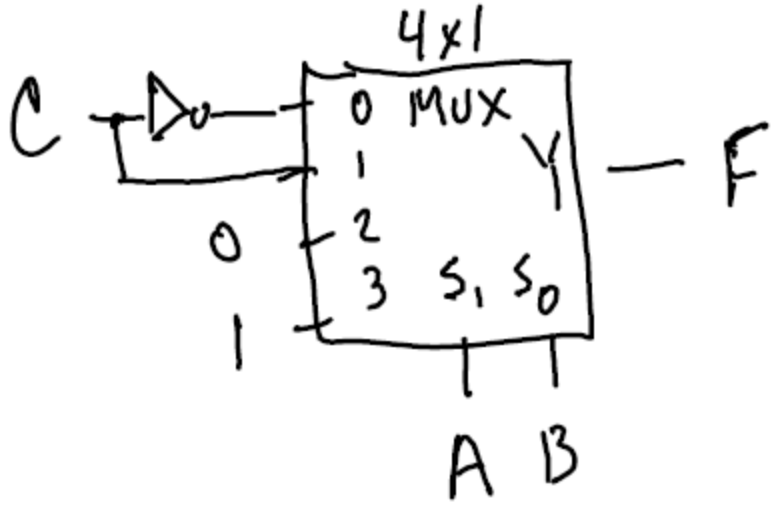
#6.- What function is implemented by the multiplexer circuit below?



a) F(A,B) = ∑m (0, 3) b) F(A,B) = ∑m (1, 2) c) F = 0 d) F = 1

Sample Questions

#7.- What function is implemented by the circuit below?



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

a) $F(A,B,C) = \Sigma m\ (0,3,6,7)$

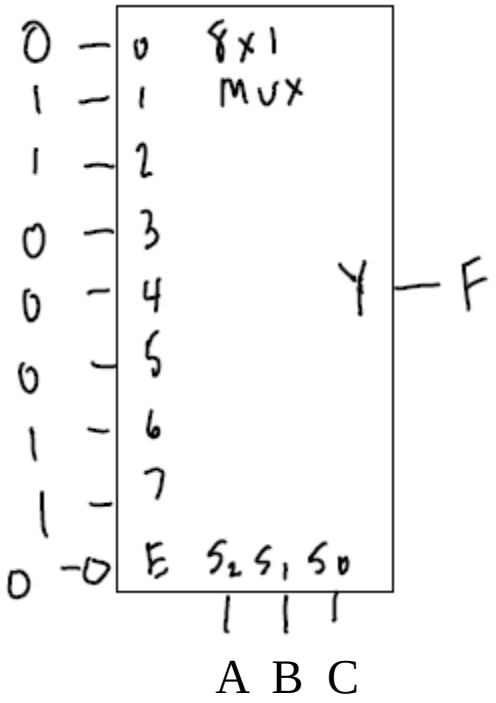
b) $F(A,B,C) = \Sigma m\ (0, 1)$

c) F = 0

d) F = 1

Sample Questions

- #8.- What function is implemented by the circuit below?



a) F(A,B,C) = Σm (0,1,2)

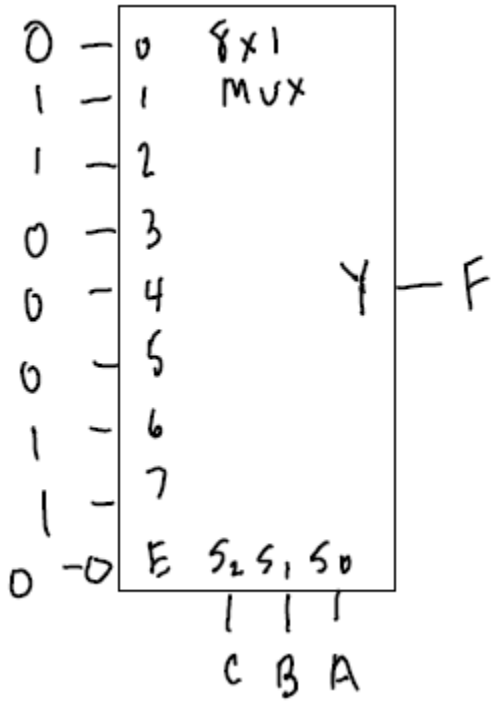
b) F(A,B,C) = Σm (1,2,6,7)

c) F = 0

d) F = 1

Sample Questions

- #9.- What function F(A,B,C) is implemented by the circuit below?



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

a) F(A,B,C) = Σm (2,3,4,7)

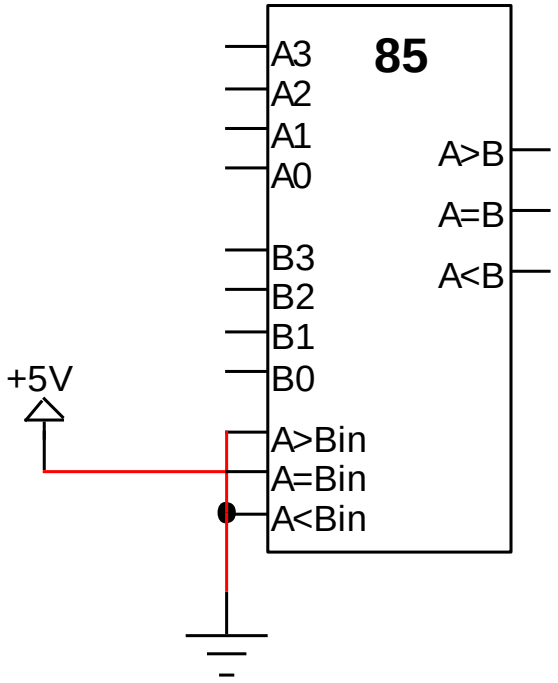
b) F(A,B,C) = Σm (1,2,6,7)

c) F = 0

d) F = 1

Sample Questions

#10.- What are the outputs of the magnitude comparator circuit below?

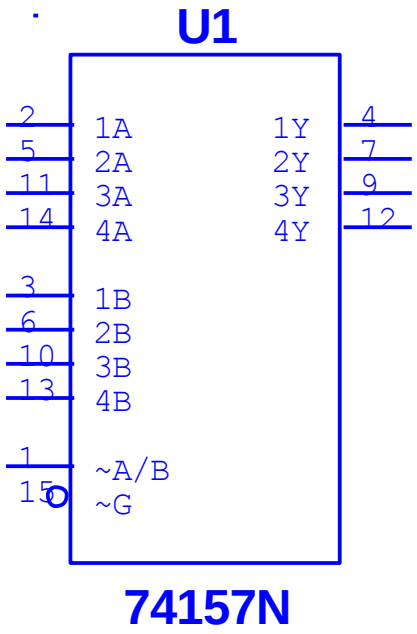


a) F1=1, F2=0, F3=0 b) F1=0, F2=1, F3=0 c) F1=0, F2=0, F3=1

d) F1=0, F2=1, F3=1

Sample Questions

#11.- What are the outputs of the 74157 quad 2x1 multiplexer circuit below?



a) 1Y=0, 2Y=0, 3Y=0, 4Y=0

b) 1Y=0, 2Y=0, 3Y=1, 4Y=1

c) 1Y=1, 2Y=0, 3Y=1, 4Y=0

d) 1Y=1, 2Y=1, 3Y=1, 4Y=1