Multiplexer & Demultiplexer

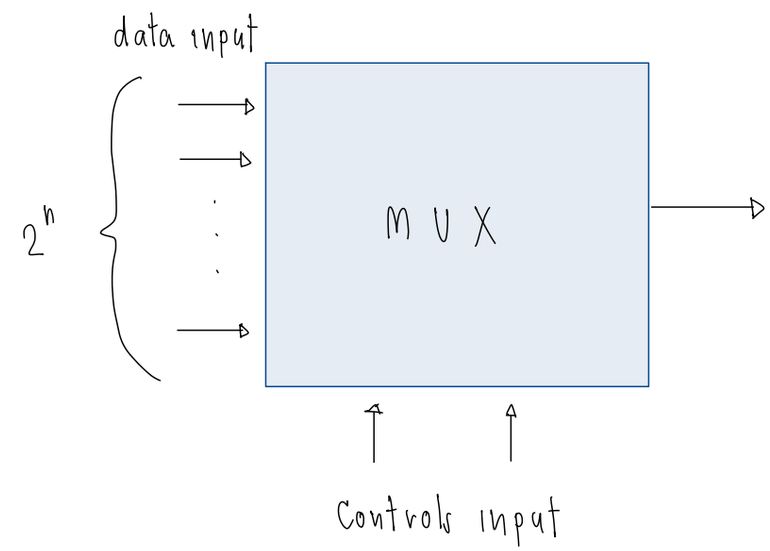
Lecture 7

Outline

- How multiplexers work
- Multiplexer as a logic building block
- How demultiplexers work

Multiplexer (MUX)

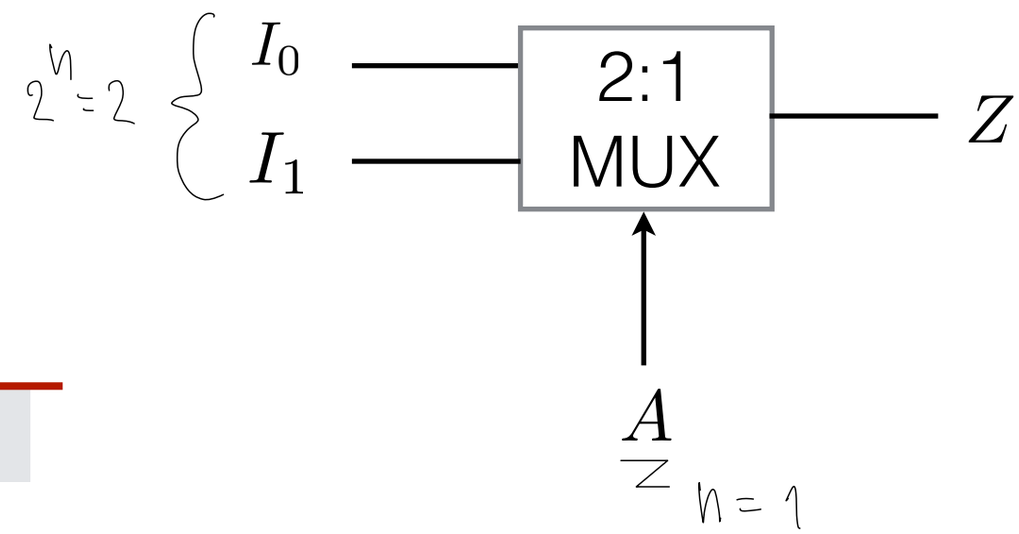
- เป็นวงจร Combinational Logic ที่ประกอบไปด้วย
  - 2^n Data Inputs
  - n Control Inputs
  - 1 Data Output



- ค่าของ Control Input จะเป็นตัวเลือกที่จะผ่านค่าของ Data Input เพียง 1 ตัวไปยัง Output
- เราอาจเรียก MUX ว่า Selector ก็ได้

2:1 MUX

- MUX อย่างย่อยที่สุดคือแบบ 2:1 MUX



A=0 → Output = I_0

A	Z
0	I_0
1	I_1

A=1 → Output = I_1

input

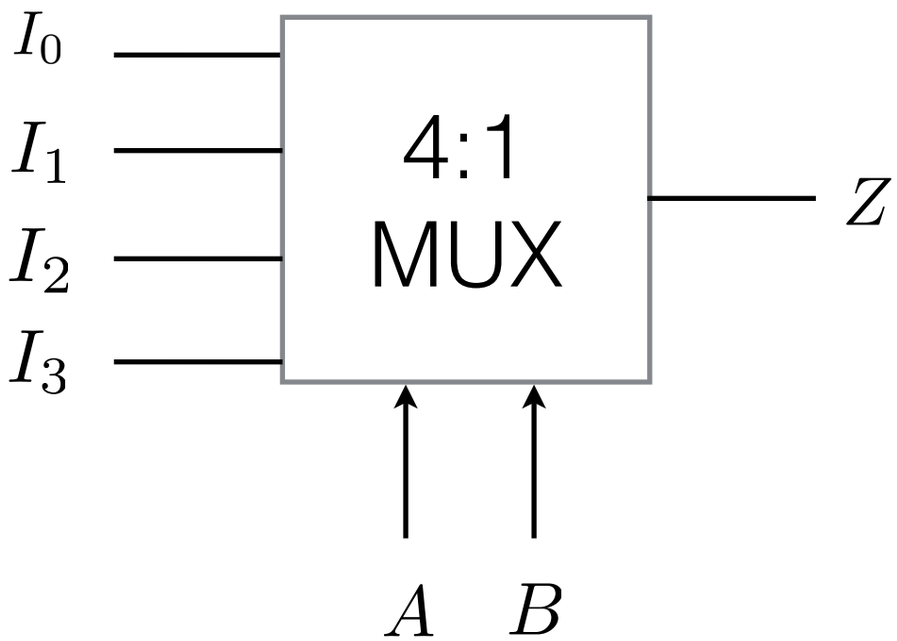
I_1	I_0	A	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

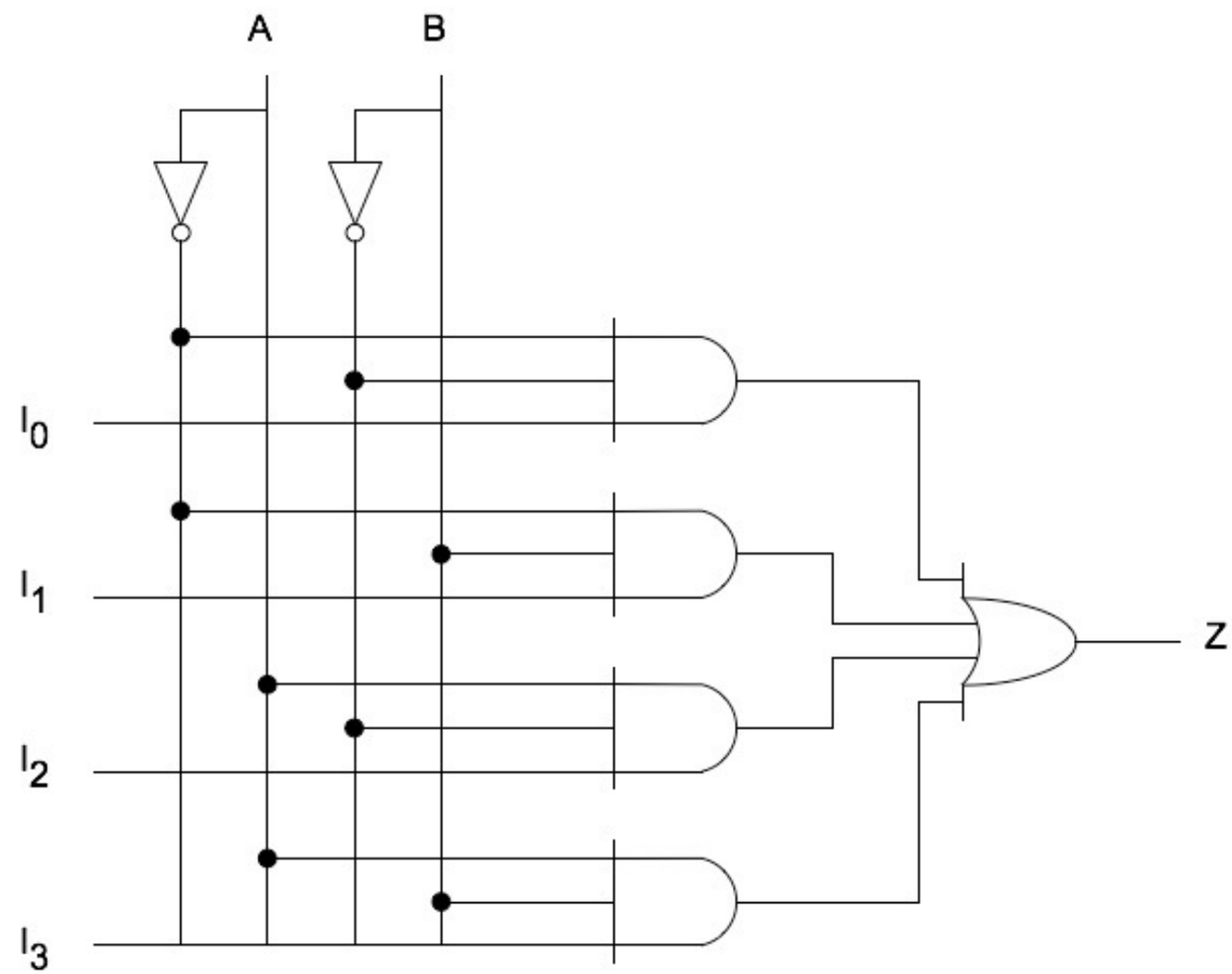
$0 I_0 + 1 I_1$

$$Z = \overline{A} I_0 + A I_1$$

$1 I_0 + \cancel{0 I_1}$

4:1 MUX



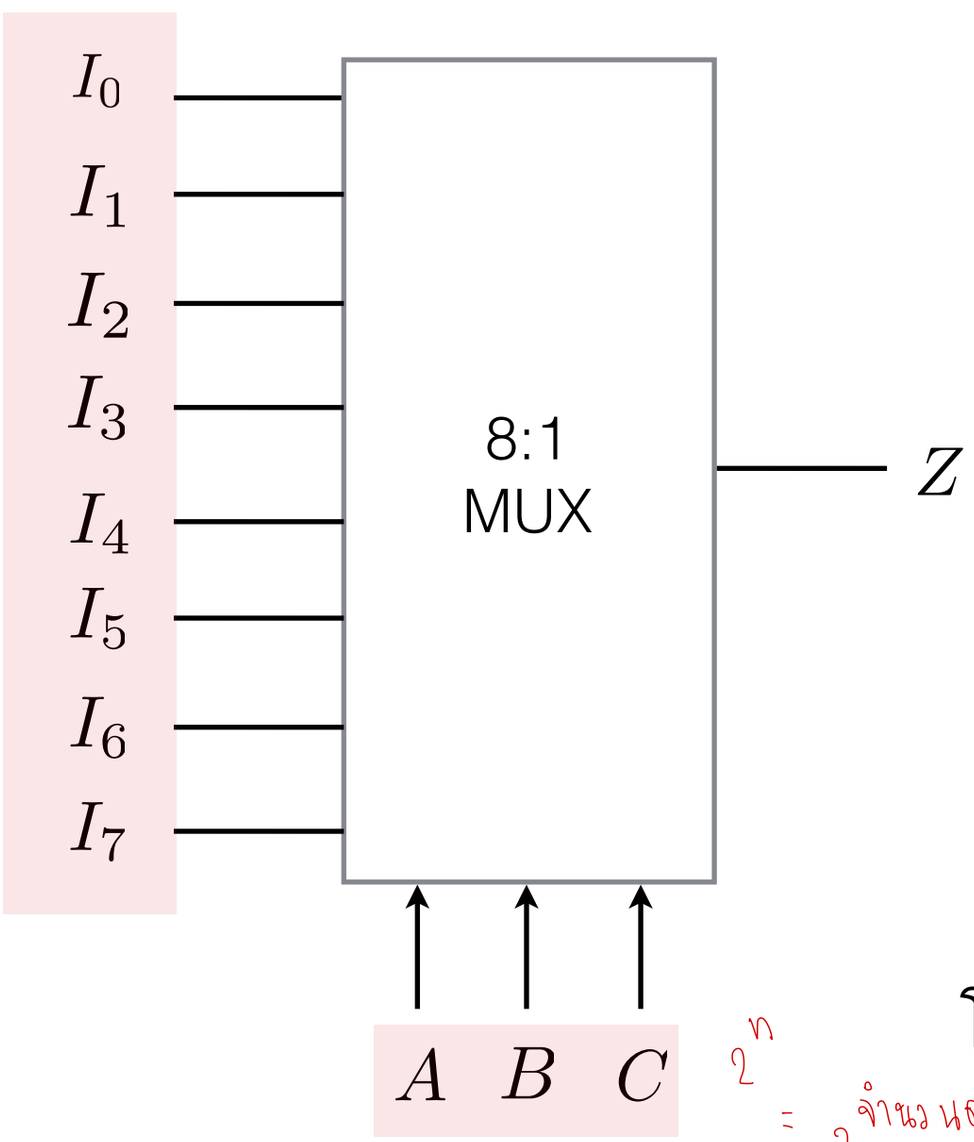


$$Z = \overline{A}\,\overline{B}I_0 + \overline{A}BI_1 + A\overline{B}I_2 + ABI_3$$

8:1 MUX

if ABC == 000:
    Z = I_0
elif ABC = 001:
    Z = I_1

ตาม Truth table



สมการบูลีนสำหรับ 8:1 MUX

$$Z = \overline{A}\,\overline{B}\,\overline{C}I_0 + \overline{A}\,\overline{B}CI_1 + \overline{A}B\overline{C}I_2 + \cdots + ABCI_7$$

สมการบูลีนสำหรับ 2^n:1 MUX

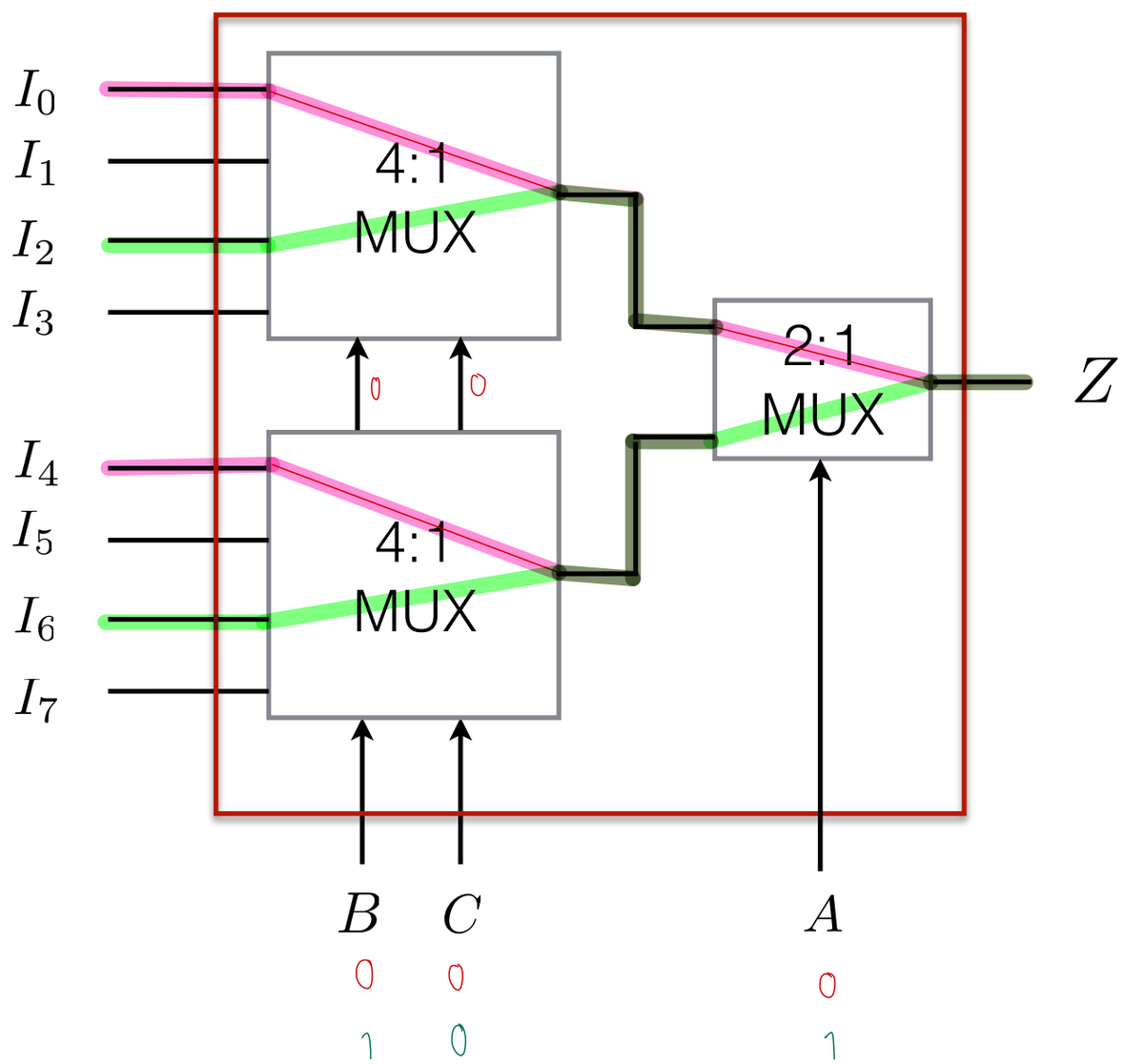
$$Z = \sum_{k=0}^{2^n-1} m_k \cdot I_k$$

โดยที่ m_k คือ k-th minterm ของ control input

2^n = $2^{\text{จำนวนตัวแปร controls input}}$

Building Larger MUX

- เราสามารถสร้าง MUX ที่ใหญ่ขึ้นจาก MUX ขนาดย่อยได้

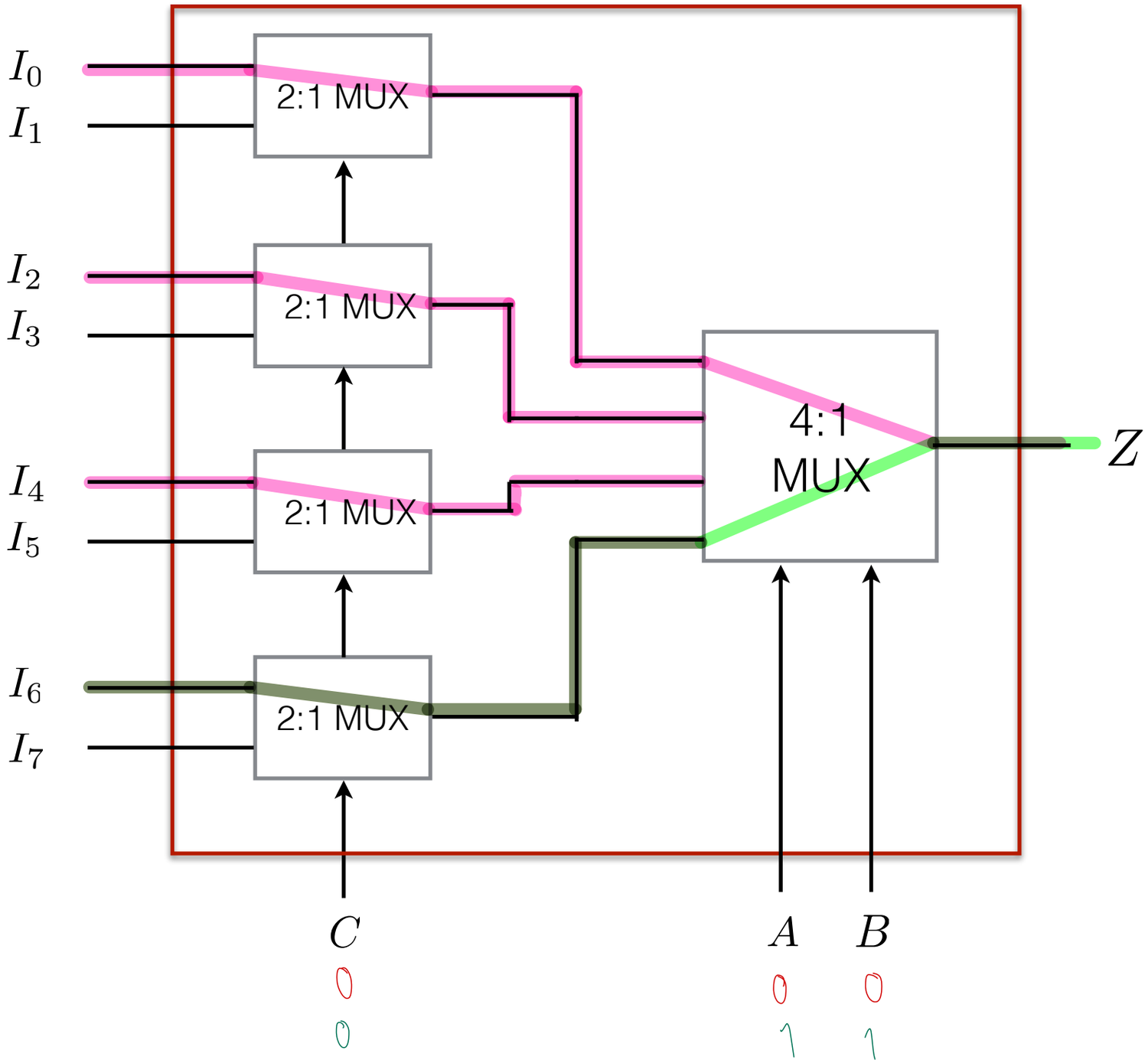


8:1 MUX

ABC = 000, Z = I_0

ABC = 110, Z = I_6

Building Larger MUX (2)



8:1 MUX

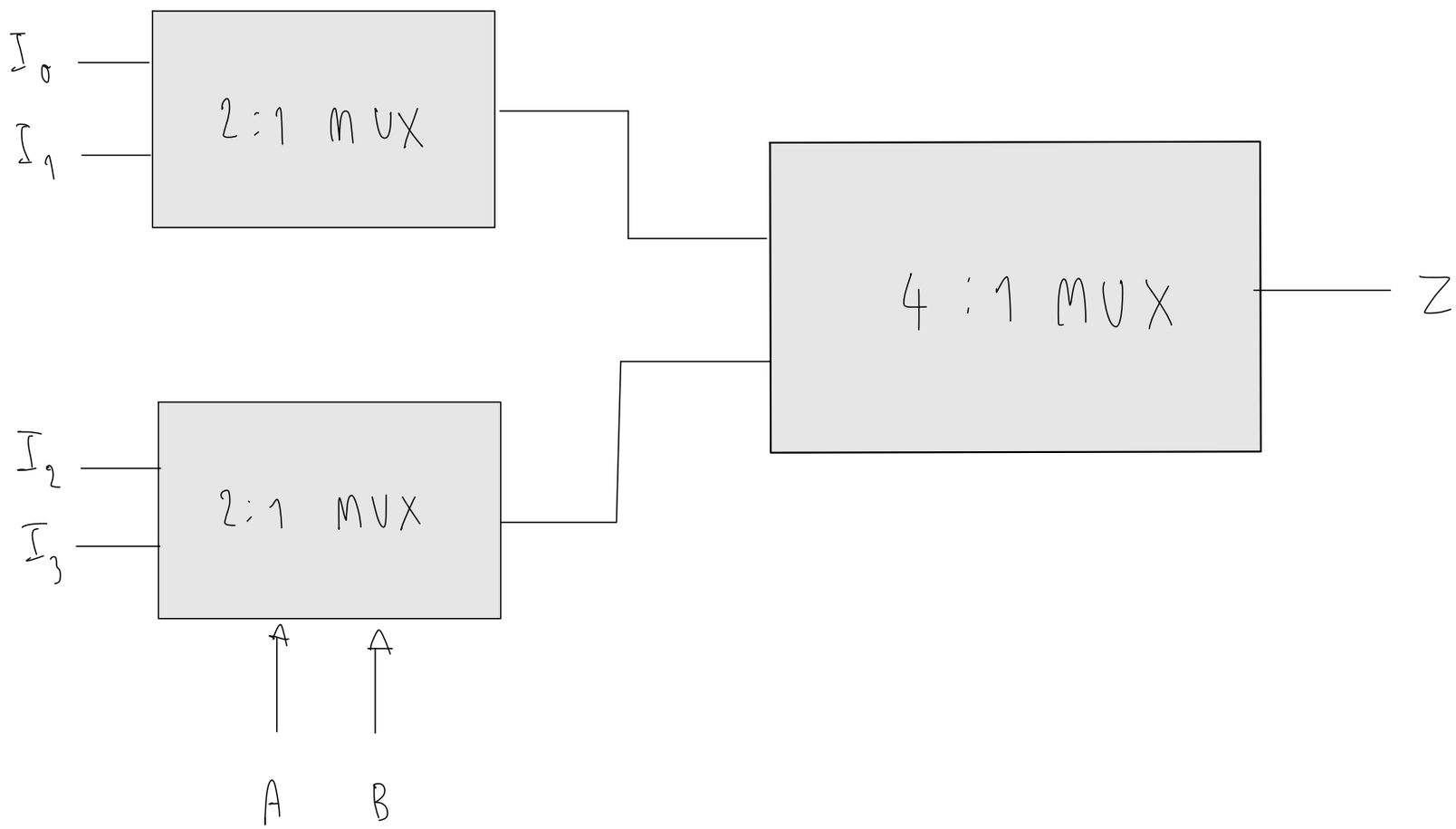
ABC = 000 , Z = I_0

ABC = 110 , Z = I_6

Example

- จงสร้าง 4:1 MUX จาก 2:1 MUX

A	B	Z
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

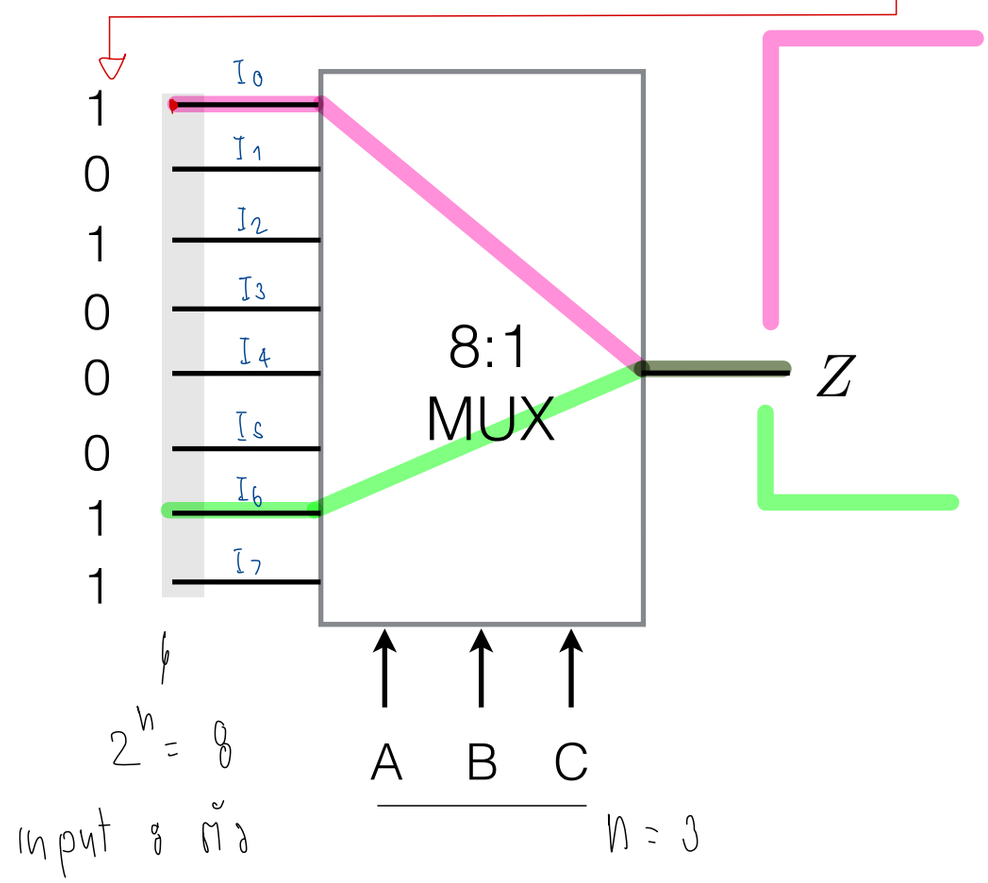


MUX as a Logic Building Block

- เราสามารถใช้ MUX สร้างฟังก์ชันตรรกต่างๆ ที่อยู่ในรูป SOP ได้
- โดยเชื่อม I_i ไปที่ลอจิก 1 ถ้าฟังก์ชันนั้นมี minterm m_i และเชื่อมอินพุทอื่นไปที่ ลอจิก 0
- เช่น

SOP

$$F(A,B,C) = m_0 + m_2 + m_6 + m_7$$



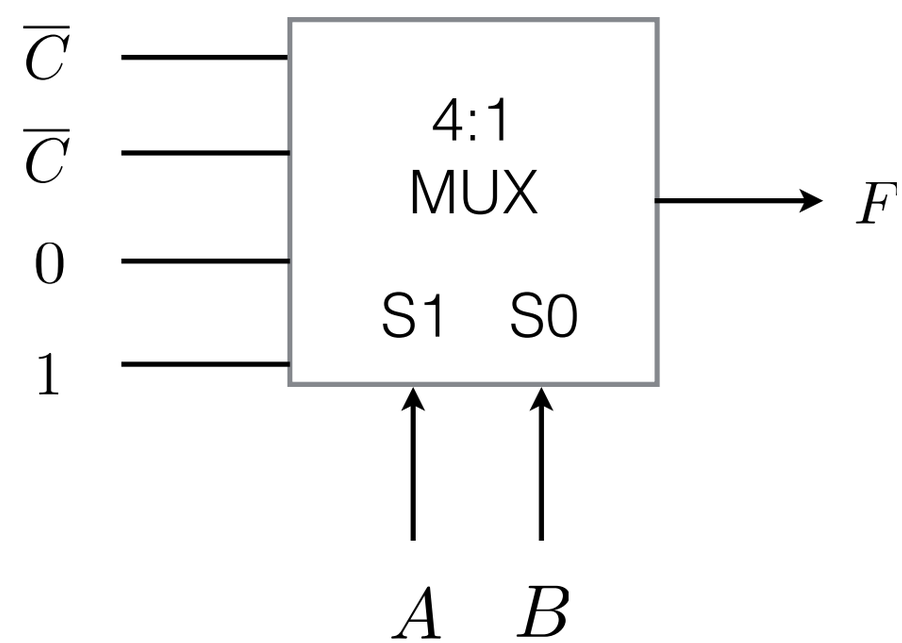
A	B	C	F
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Smaller Design

- เราสามารถสร้างวงจรที่ซับซ้อนน้อยลงได้โดยอาศัยการจัดกลุ่มที่ดี

$$F = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}B\overline{C} + A\overline{B}(0) + AB(1)$$

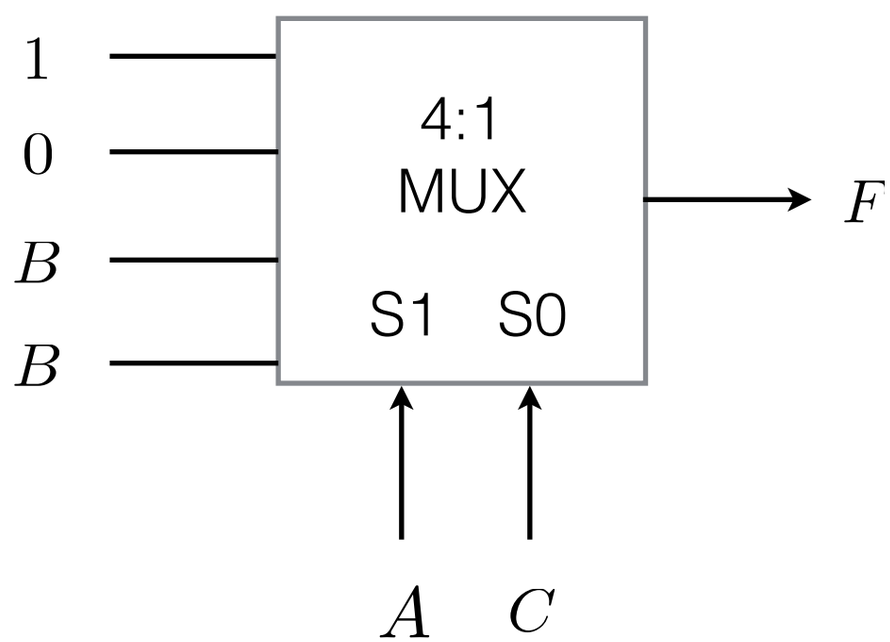
A	B	C	F	
0	0	0	1	$\overline{C}$
0	0	1	0	
0	1	0	1	$\overline{C}$
0	1	1	0	
1	0	0	0	0
1	0	1	0	
1	1	0	1	1
1	1	1	1	



Smaller Design (2)

- ไม่ต้องใช้ inverter

$$F = \overline{A}\,\overline{C}(1) + \overline{A}C(0) + A\overline{C}B + ACB$$



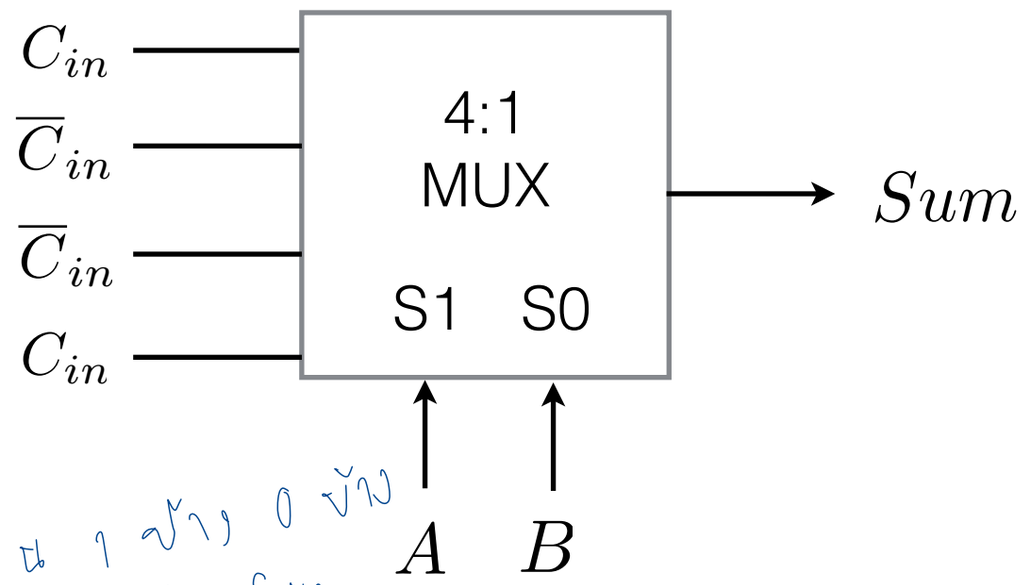
General Principles in Circuit Design with MUX

- สำหรับฟังก์ชันที่มีอินพุทจำนวน n ตัว เลือกอินพุทจำนวน n-1 ตัว มาใช้เป็น control inputs ของ MUX
- อินพุทอีก 1 ตัวที่เหลือของฟังก์ชันจะถูกใช้เป็น data input ของ MUX ร่วมกับ 0 และ 1

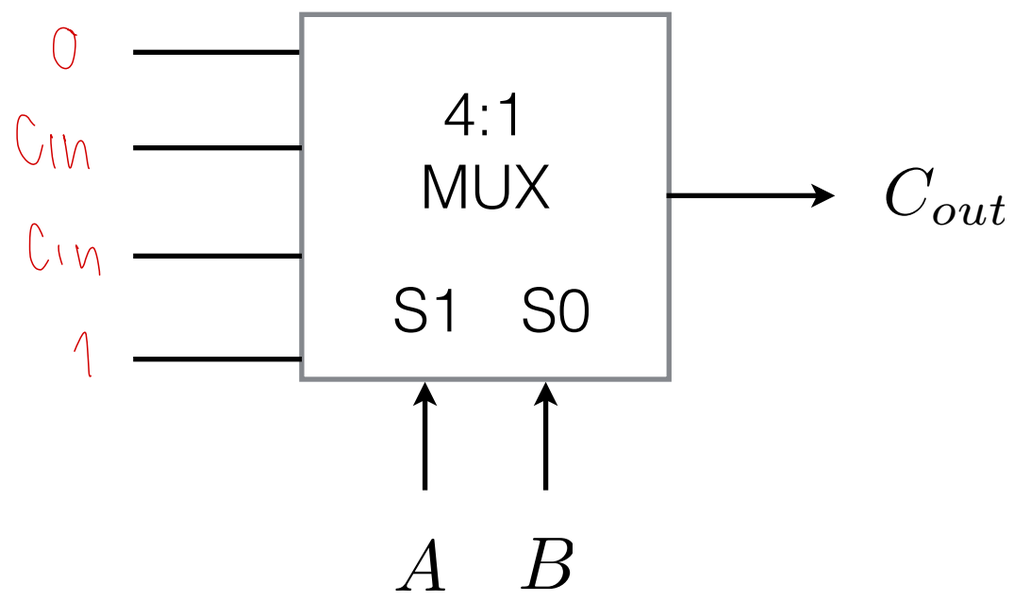
Example

- ใช้ MUX สร้าง Full Adder

A	B	Cin	Sum	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

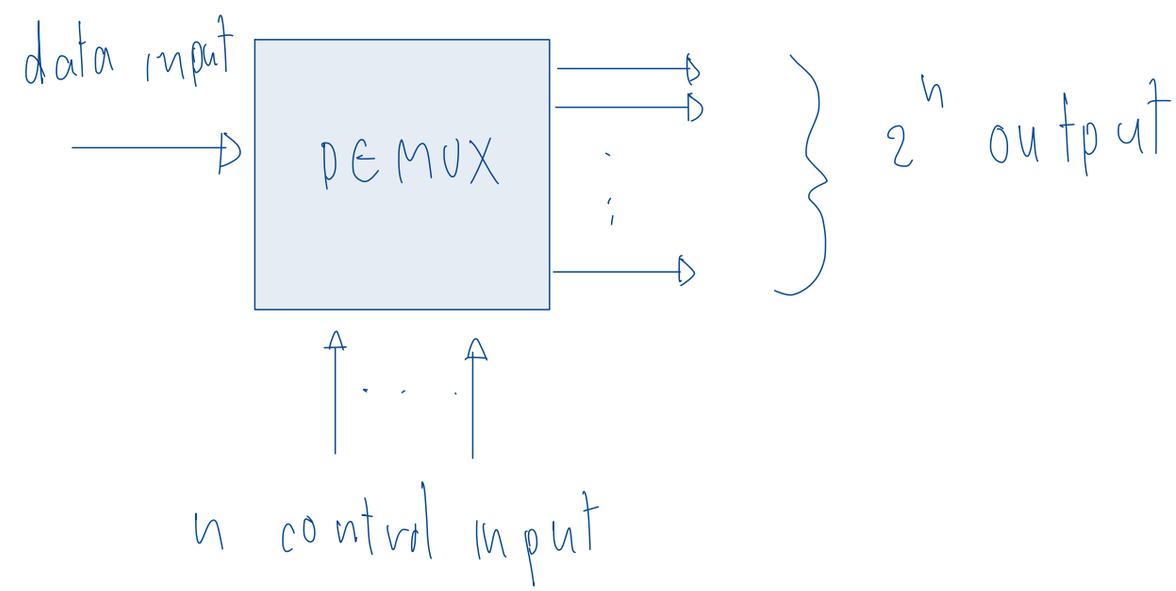


ถ้าเป็น 1 ข้าง 0 ข้าง
ให้ดู Cin



Demultiplexer (DEMUX)

- ทำหน้าที่ตรงข้ามกับ MUX
- ประกอบไปด้วย
  - 1 data input
  - n control signals
  - 2^n output lines

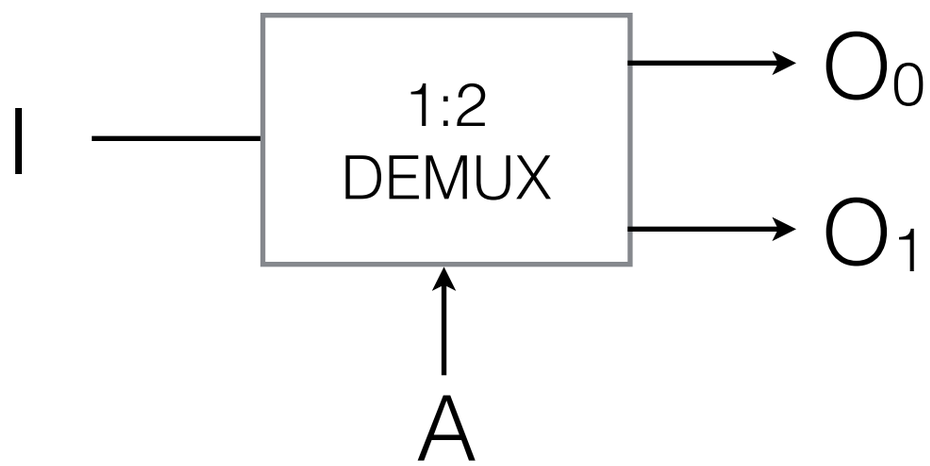


1:2 DEMUX

- เมื่อ I = 0, Output ทั้งคู่จะเป็น 0
- เมื่อ I = 1, Output จะขึ้นอยู่กับ A

$$O_0 = \bar{A}I$$

$$O_1 = AI$$



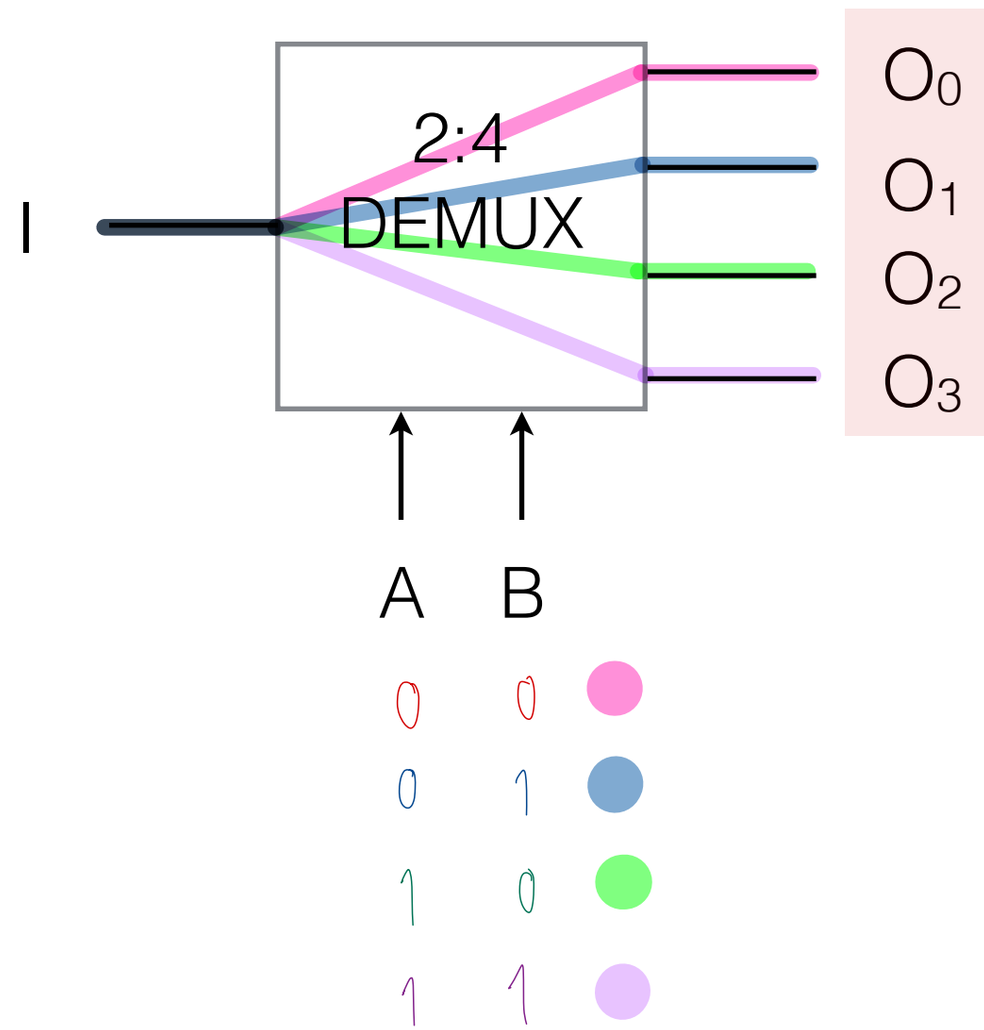
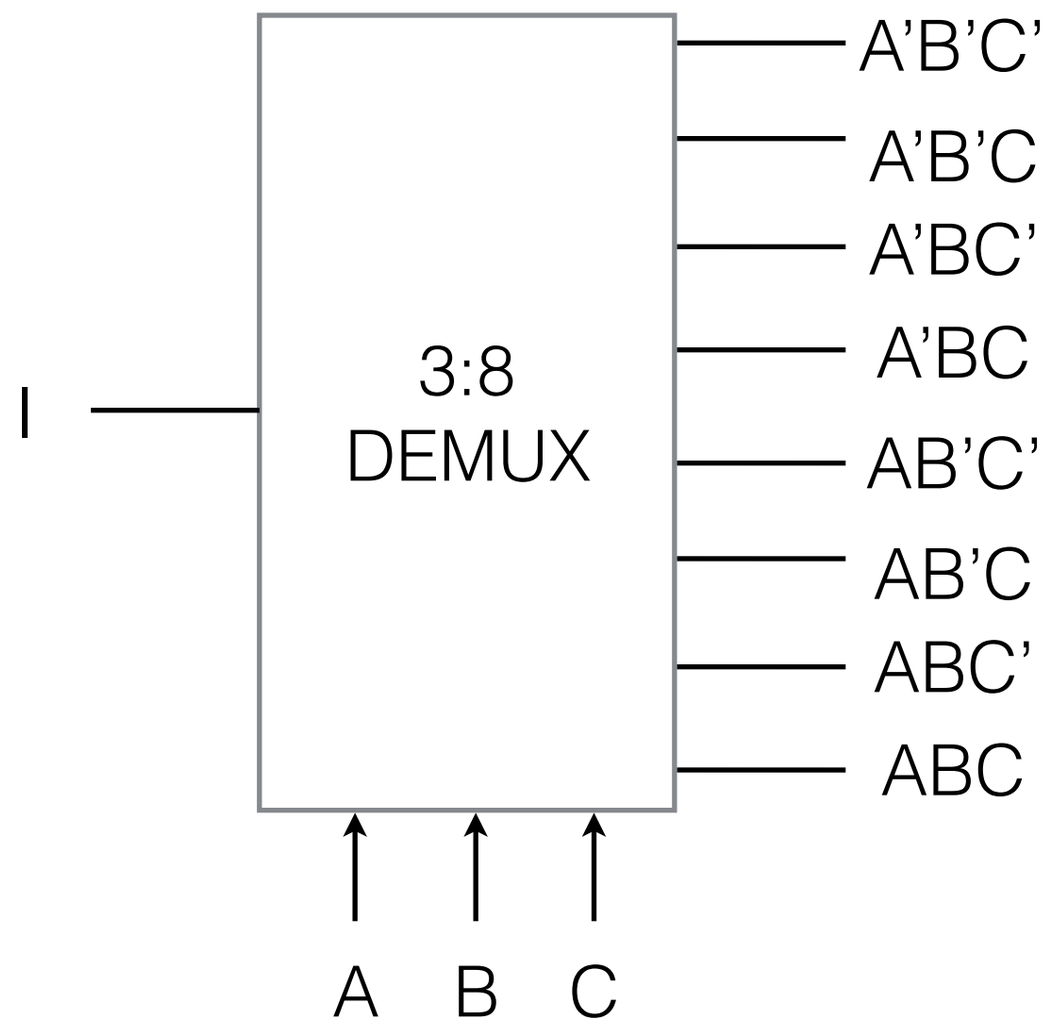
2:4 DEMUX

$$O_0 = \overline{A}\,\overline{B}I$$

$$O_1 = \overline{A}BI$$

$$O_2 = A\overline{B}I$$

$$O_3 = ABI$$



3:8 DEMUX