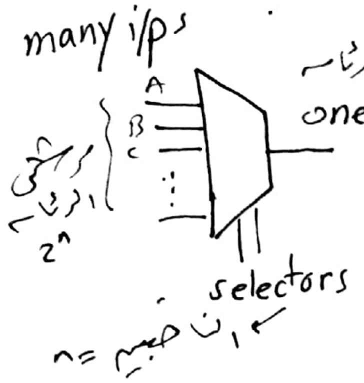
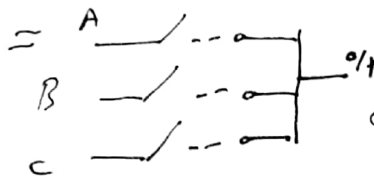
Lec (03)

منتقي = منتخب = Multiplexers [Combinational Circits]



دايره تنقي أحد الـ inputs وتوجه البيانات الموجودة عليه إلى فرع الدائرة الوحيد.

ويتم اختيار أحد المداخل inputs عن طريق خطوط الاختيار select lines

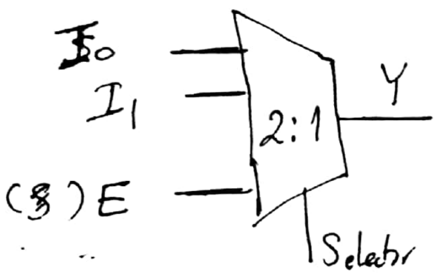


عن طريق اذا كان عدد (n) selectors فإن عدد الـ input Lines $= 2^n$

هدف MUX :- تقليل عدد الأسلاك اللي بتكلف تصميم دوائر كتير بيها

Types

(1) 2:1 MUX



E (High لازم)	S (selector)	Y
0	×	0
1	0	I_0
1	1	I_1

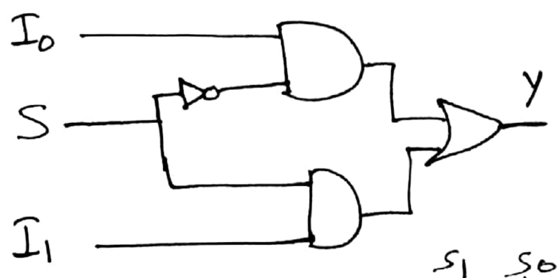
$\therefore y = I_0 \bar{S} E + I_1 S E$

مبدئياً لازم يكون Enable = 1 High

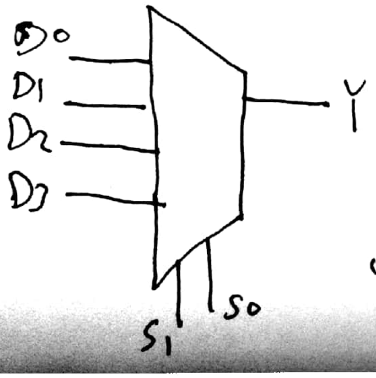
عندما تكون S = صفر ∴ الـ MUX يطلع I_0

وعندما تكون S = 1 ∴ ″ ″ I_1

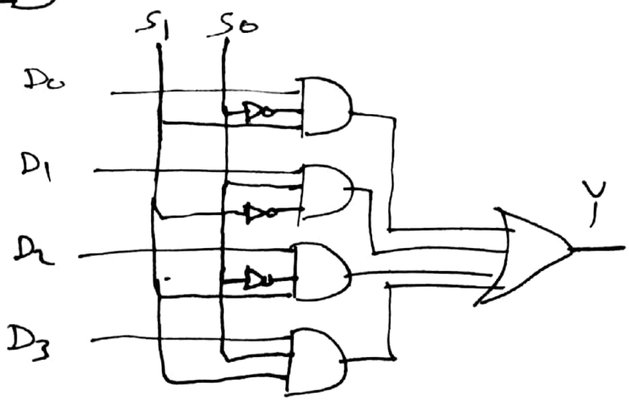
لاحظ بعض الـ IC's بيكون دخل الـ E منفي ($\bar{E}$) ∴ دخل الـ Enable صفر عشان يكون High



(2) 4:1 MUX

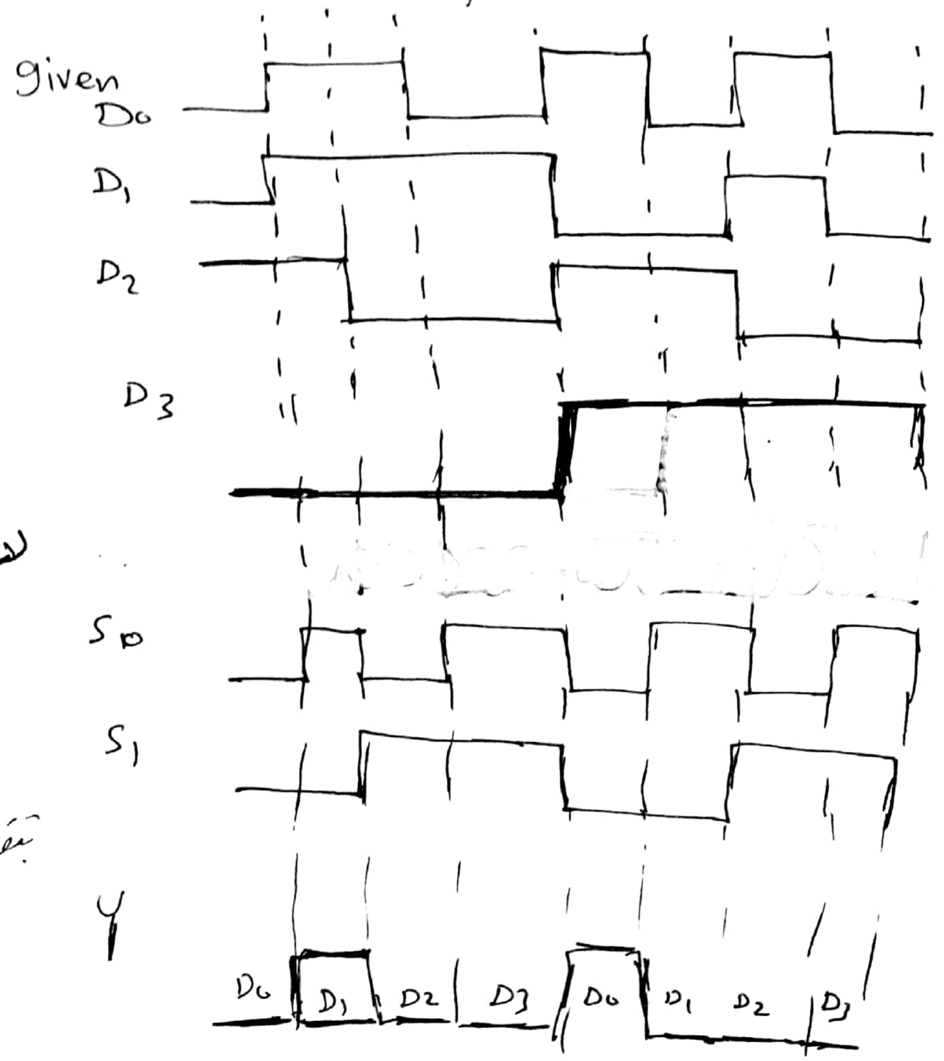


S_1	S_0	Y
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3



$$Y = D_0 \bar{S}_0 \bar{S}_1 + D_1 S_0 \bar{S}_1 + D_2 \bar{S}_0 S_1 + D_3 S_0 S_1$$

Example



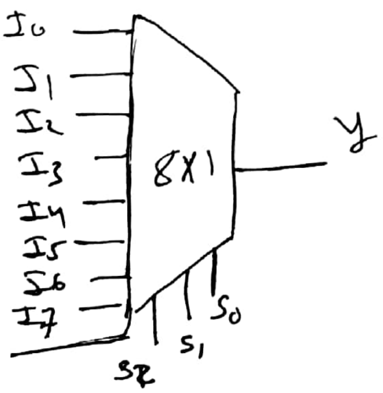
S_1	S_0	
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

← D_3 D_2 D_1 D_0

③ 8:1 Mux

3 selectors, 8 i/ps, one o/p

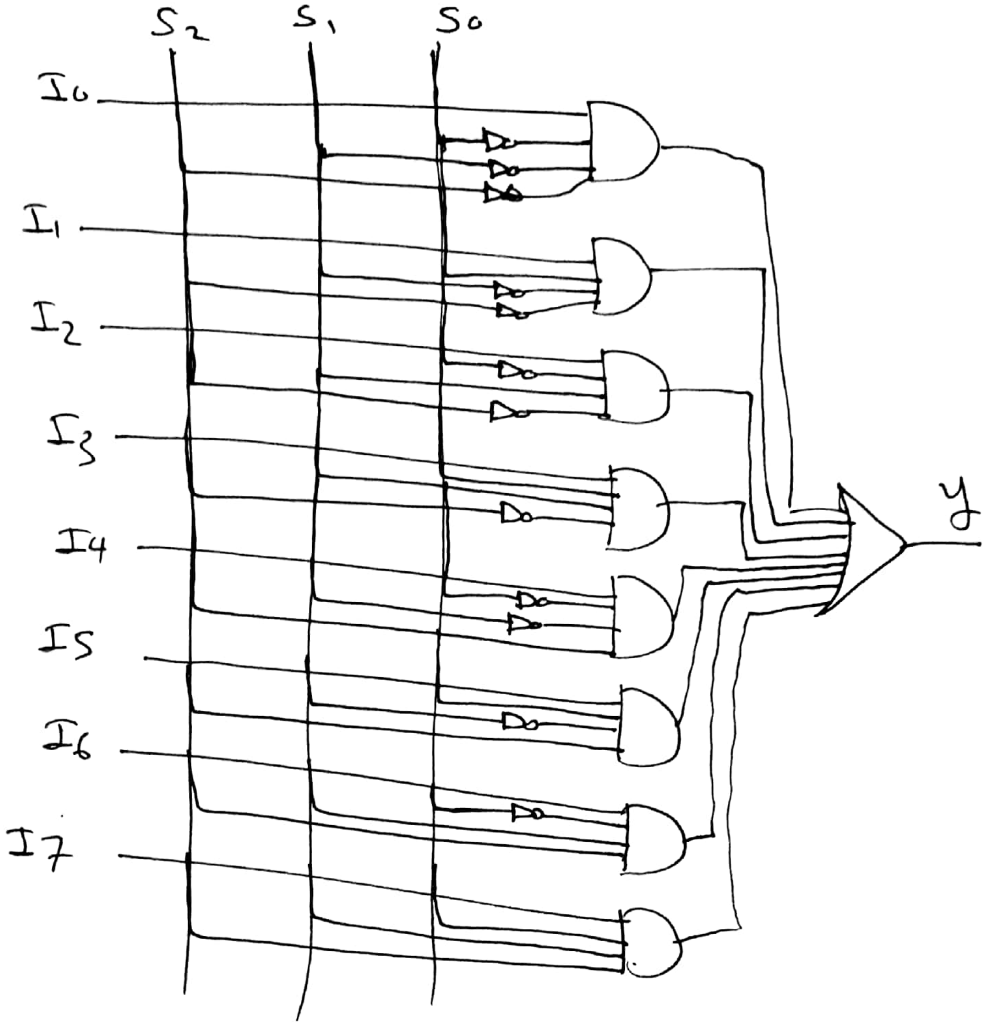
S_2	S_1	S_0	y
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	0	0	I_4
1	0	1	I_5
1	1	0	I_6
1	1	1	I_7



(… Enable … High …) E

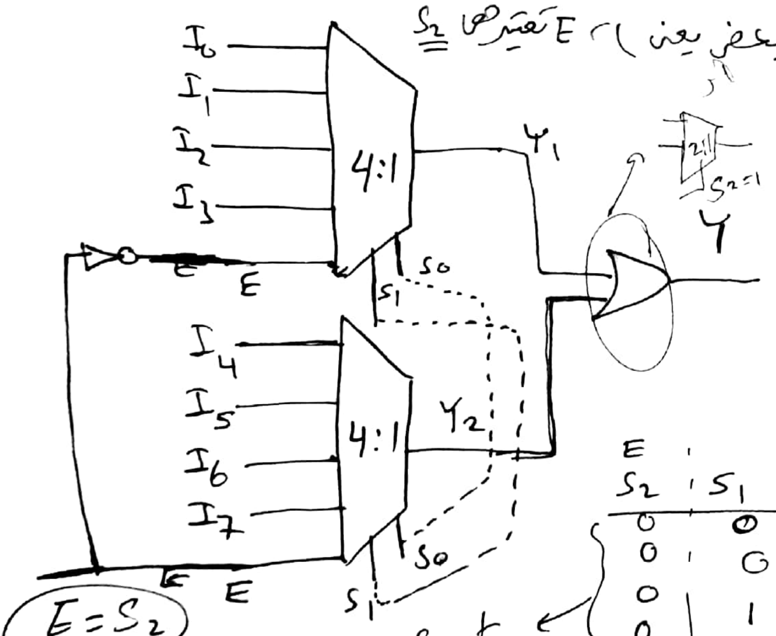
$$Y = I_0\overline{S_0}\,\overline{S_1}\,\overline{S_2} + I_1 S_0\overline{S_1}\,\overline{S_2} + I_2\overline{S_0}S_1\overline{S_2} + I_3 S_0 S_1\overline{S_2}$$
$$+ I_4\overline{S_0}\,\overline{S_1}S_2 + I_5 S_0\overline{S_1}S_2 + I_6\overline{S_0}S_1S_2 + I_7 S_0S_1S_2$$

8:1 Mux
logic Diagram



④ Implementation of 8:1 using Two 4:1

لاحظ so، s1 هم هم ذا ال 2 Ic (موصلين ببعض يعني)، E تعتبرها S_2



if $E=0$ $(S_2=0)$ ∴ first Mux on, 2nd Mux off

if $S_0=0, S_1=0, \underset{S_2}{E}=0$ ∴ $I_0 = Y_1$, $Y_2=0$ ∴ $Y = I_0$

if $S_0=1, S_1=0, \underset{S_2}{E}=0$ ∴ $Y_1 = I_1, Y_2=0$

وهكذا

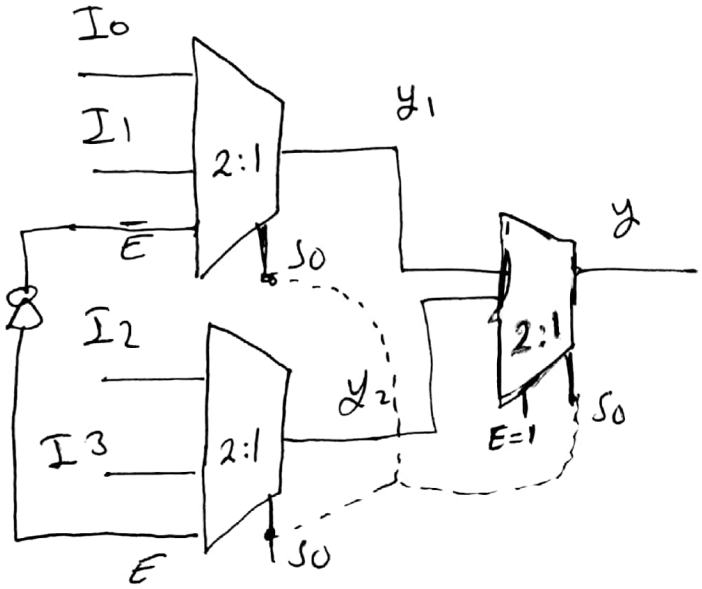
E / S_2	S_1	S_0	Y_1	Y_2	Y
0	0	0	I_0	0	I_0
0	0	1	I_1	0	I_1
0	1	0	I_2	0	I_2
0	1	1	I_3	0	I_3
1	0	0	0	I_4	I_4
1	0	1	0	I_5	I_5
1	1	0	0	I_6	I_6
1	1	1	0	I_7	I_7

first Mux on (rows 1–4)

2nd Mux on (rows 5–8)

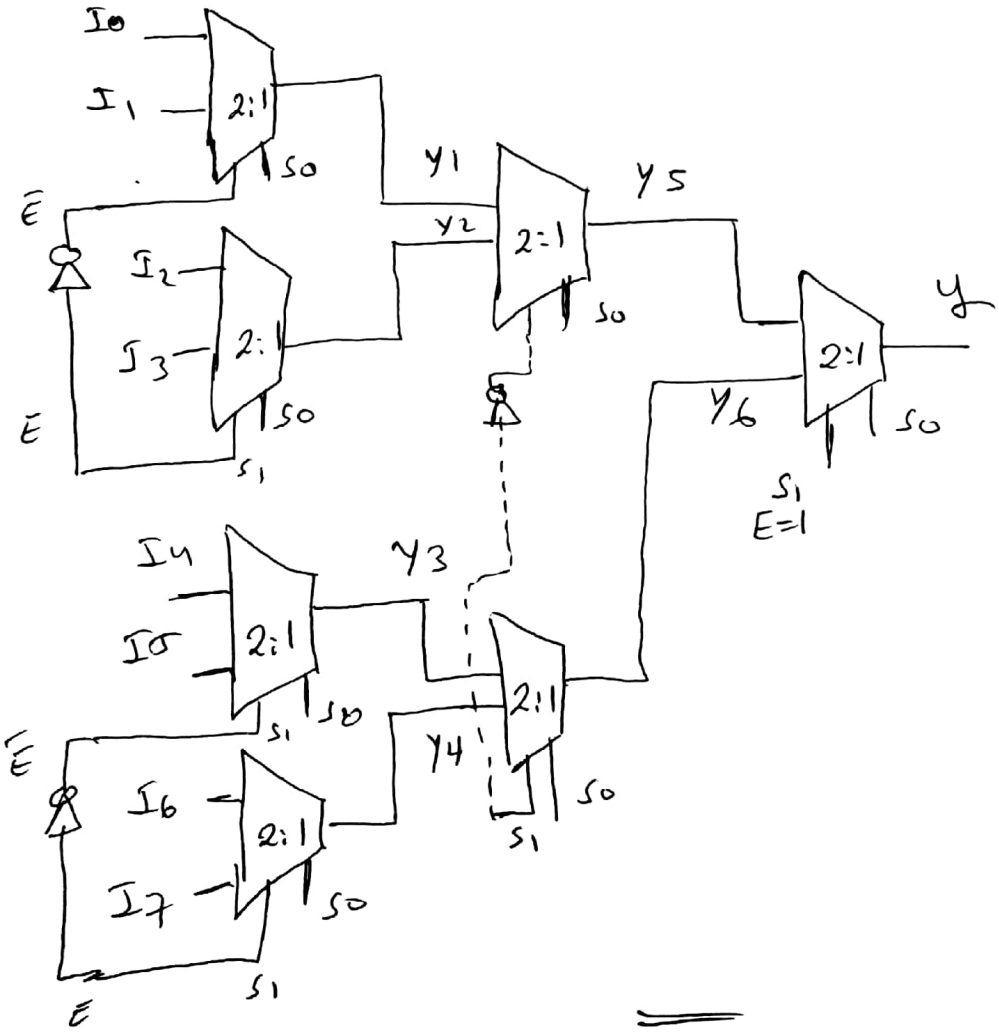
ممكن استبدال ال OR ب 2:1 Mux

(5) Implementation of 4:1 MUX using 2:1



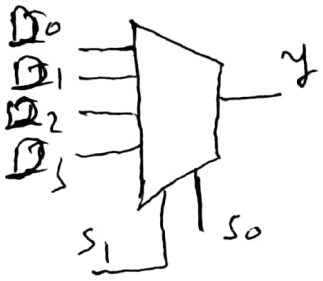
---

(6) Implementation of 8:1 MUX using 2:1



---

(7) Implementation of XOR, Not, AND, OR using 4:1 MUX



$y = D_0 \bar{S_0} \bar{S_1} + D_1 S_0 \bar{S_1} + D_2 \bar{S_0} S_1 + D_3 S_0 S_1$

* for $D_0 = D_1 = D_2 = 0$, $D_3 = 1$

$\therefore y = S_0 S_1$ (AND)

* for $D_1 = D_2 = D_3 = 1$, $D_0 = 0$

$y = S_0 \bar{S_1} + \bar{S_0} S_1 + S_0 S_1 = S_0 \bar{S_1} + S_1 (S_0 + \bar{S_0})$, $(S_0 + \bar{S_0}) \to 1$

$\therefore y = S_0 + S_1$ (OR)

* for $D_0 = D_2 = 1$ & $D_1 = D_3 = 0$

$\therefore y = \bar{S_0} \bar{S_1} + \bar{S_0} S_1 = \bar{S_0} (\bar{S_1} + S_1) = \bar{S_0}$ (Not)

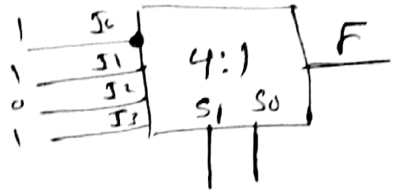
* for $D_0 = D_3 = 0$, $D_1 = D_2 = 1$

$\therefore y = S_0 \bar{S_1} + \bar{S_0} S_1$ (XOR)

8- Implement the Function $F(x,y) = \Sigma(0,1,3)$ using suitable MUX

Solution

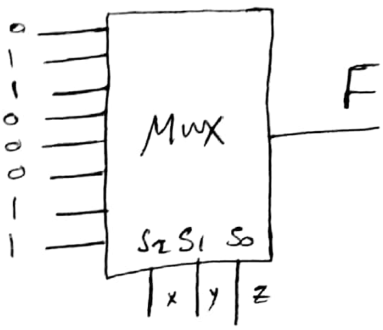
x	y	F
0	0	1
0	1	1
1	0	0
1	1	1



We selected 4:1 Mux due to the largest Input number is 3 (∴ 4 line inputs & require 2 select lines)

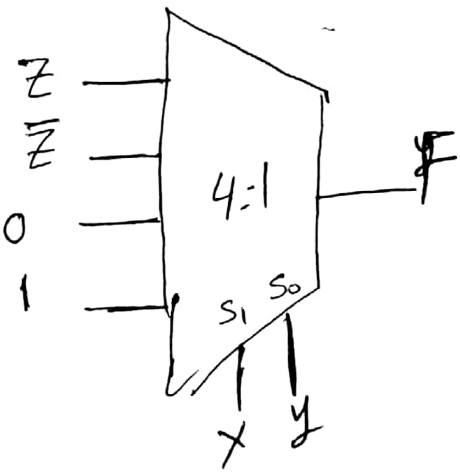
9- Implement $F(x,y,z) = \Sigma(1,2,6,7)$

x	y	z	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1



10- Implement $F(x,y,z) = \Sigma(1,2,6,7)$ using (one) 4:1 MUX

x	y	z	F	
0	0	0	0	} F = z
0	0	1	1	
0	1	0	1	} $F = \bar{z}$
0	1	1	0	
1	0	0	0	} F = 0
1	0	1	0	
1	1	0	1	} F = 1
1	1	1	1	

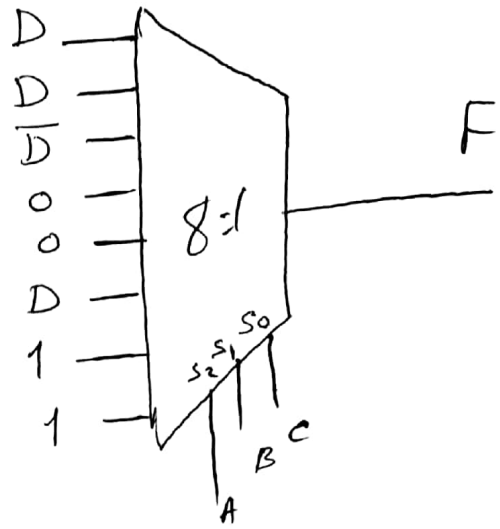


when xy = 00 o/p F = z < if z=0 f=0, if z=1 f=1

وهكذا الباقي

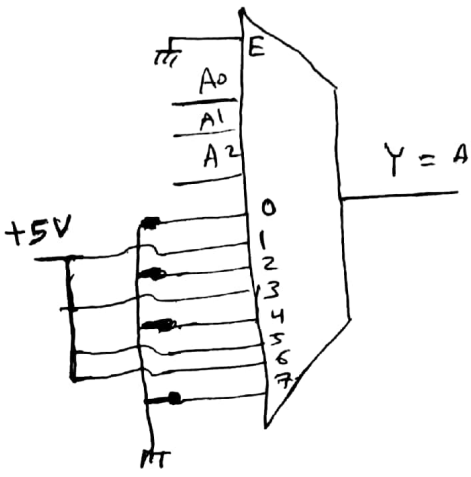
11- Example, $F(A,B,C,D) = \Sigma(1,3,4,11,12,13,14,15)$
using 8-to-1 Mux

A	B	C	D	F		
0	0	0	0	0	0	F = D
0	0	0	1	1	1	
0	0	1	0	0	2	F = D
0	0	1	1	1	3	
0	1	0	0	1	4	$F = \bar{D}$
0	1	0	1	0	5	
0	1	1	0	0	6	F = 0
0	1	1	1	0	7	
1	0	0	0	0	8	F = 0
1	0	0	1	0	9	
1	0	1	0	0	10	F = D
1	0	1	1	1	11	
1	1	0	0	1	12	F = 1
1	1	0	1	1	13	
1	1	1	0	1	14	F = 1
1	1	1	1	1	15	



(12) function Use Truth table to design using 8:1

A_2	A_1	A_0	Y	
0	0	0	0	
0	0	1	1	1
0	1	0	0	
0	1	1	1	3
1	0	0	0	
1	0	1	1	5
1	1	0	1	6
1	1	1	0	



$Y = A_0\bar{A}_1\bar{A}_2 + A_0A_1\bar{A}_2 + A_0\bar{A}_1A_2 + \bar{A}_0A_1A_2$

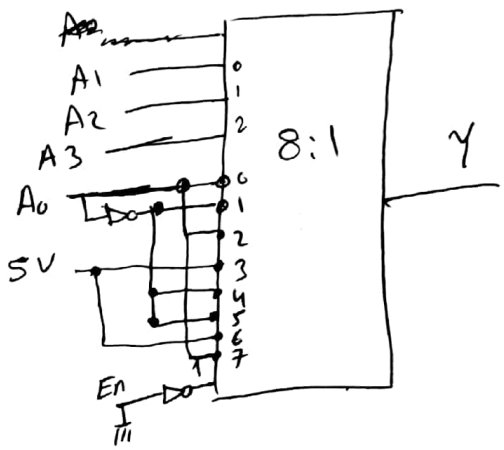
(13)- Implement next Truth table using 8:1 MUX

A_3	A_2	A_1	A_0	Y	
0	0	0	0	0	$Y = A_0$
0	0	0	1	1	$A_0 \bar{A_1} \bar{A_2} \bar{A_3}$
0	0	1	0	1	$\bar{A_0} A_1 \bar{A_2} \bar{A_3}$
0	0	1	1	0	$Y = \bar{A_0}$
					$Y = A_0$
0	1	0	0	0	
0	1	0	1	1	$A_0 \bar{A_1} A_2 \bar{A_3}$
0	1	1	0	1	$\bar{A_0} A_1 A_2 \bar{A_3}$
0	1	1	1	1	$A_0 A_1 A_2 \bar{A_3}$ → $Y = 1$
1	0	0	0	1	$\bar{A_0} \bar{A_1} \bar{A_2} A_3$ → $Y = \bar{A_0}$
1	0	0	1	0	
1	0	1	0	1	$\bar{A_0} A_1 \bar{A_2} A_3$ → $Y = \bar{A_0}$
1	0	1	1	0	
1	1	0	0	1	$\bar{A_0} \bar{A_1} A_2 A_3$ → $Y = 1$
1	1	0	1	1	$\bar{A_0} A_1 A_2 A_3$
1	1	1	0	0	$Y = A_0$
1	1	1	1	1	$A_0 A_1 A_2 A_3$

$$Y = A_0 \bar{A_1} \bar{A_2} \bar{A_3} + \bar{A_0} A_1 \bar{A_2} \bar{A_3} + A_0 \bar{A_1} A_2 \bar{A_3} + \bar{A_0} A_1 A_2 \bar{A_3} + A_0 A_1 A_2 \bar{A_3}$$
$$+ \bar{A_0} A_1 \bar{A_2} A_3 + \bar{A_0} \bar{A_1} A_2 A_3 + A_0 A_1 A_2 A_3$$

Pins

0,1 → A_0 ⟶ 0 ✓
2,3 = $\bar{A_0}$ ⟶ 1 ✓
4,5 = A_0 ⟶ 2 ✓
6,7 = 1 ⟶ 3 ✓
8,9 = $\bar{A_0}$ ⟶ 4 ✓
10,11 = $\bar{A_0}$ ⟶ 5 ✓
12,13 = 1 ⟶ 6 ✓
14,15 = A_0 ⟶ 7 ✓



Next lecture → Encoders
→ Demultiplexers