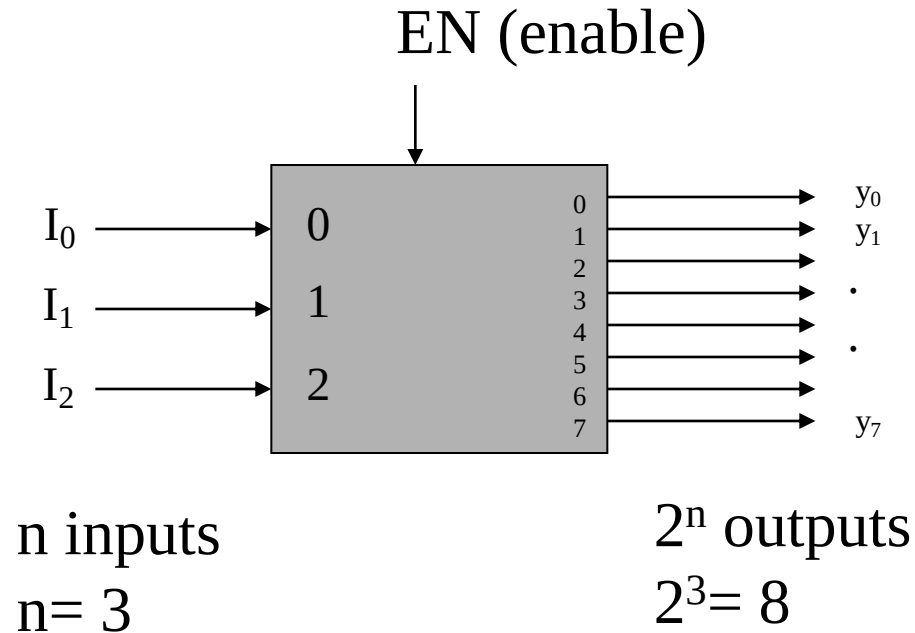


CSE 140

Discussion Session

Decoder

A digital module that converts a binary address to the assertion of the addressed device

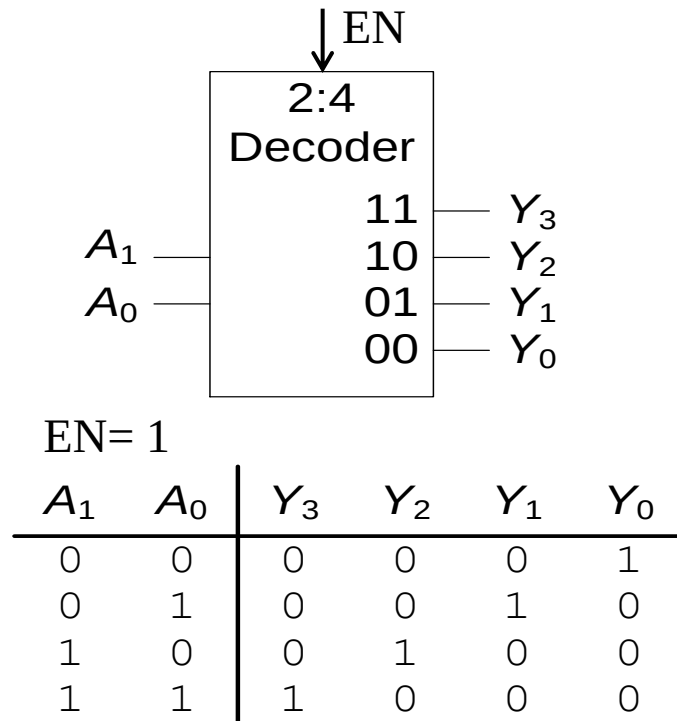


n to 2^n decoder
function:

$$y_i = 1 \text{ if } E_n = 1 \ \& \ (I_2, I_1, I_0) = i$$
$$y_i = 0 \text{ otherwise}$$

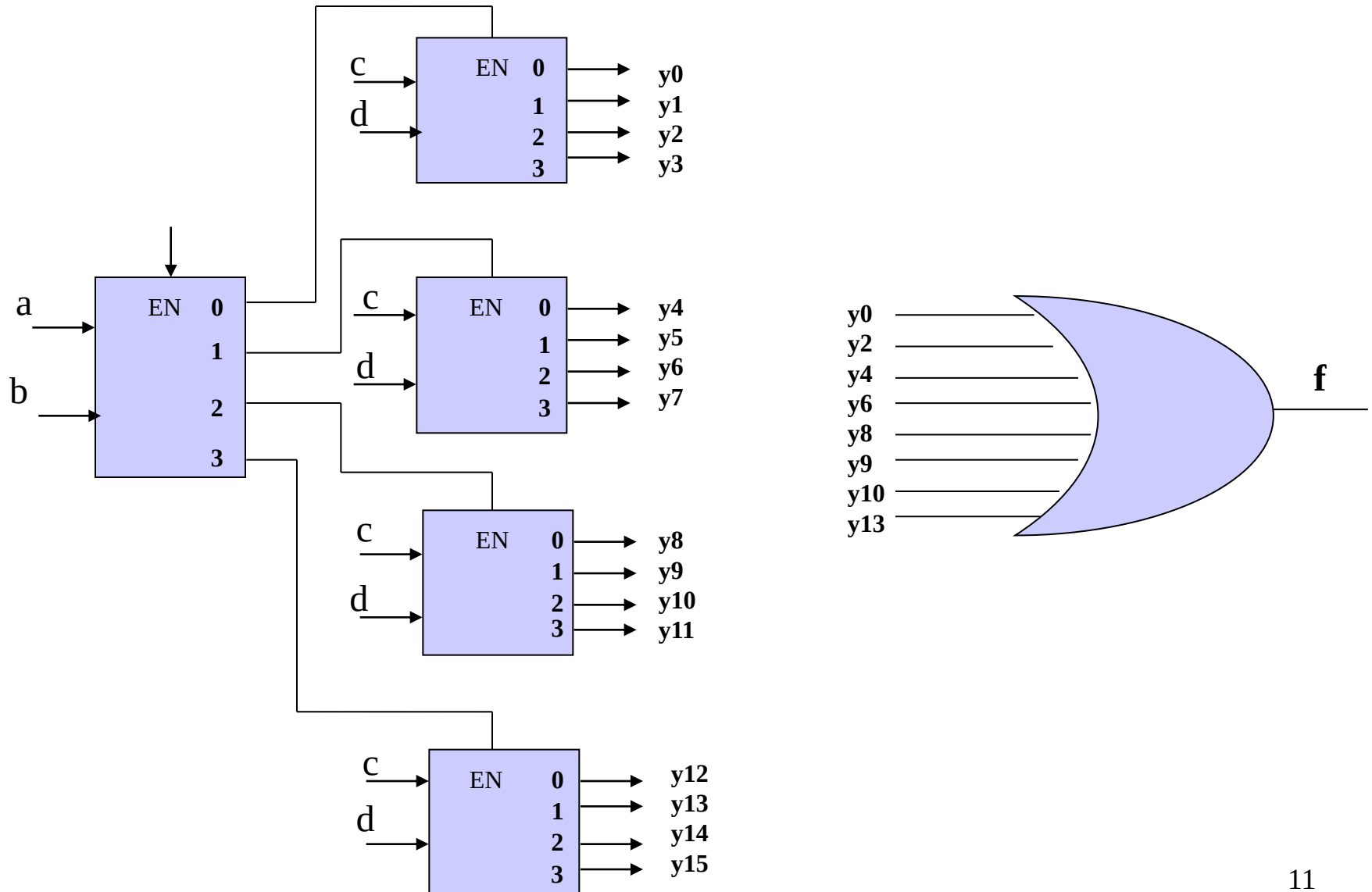
Decoder

- N inputs, 2^N outputs
- One-hot outputs: only one output HIGH at once



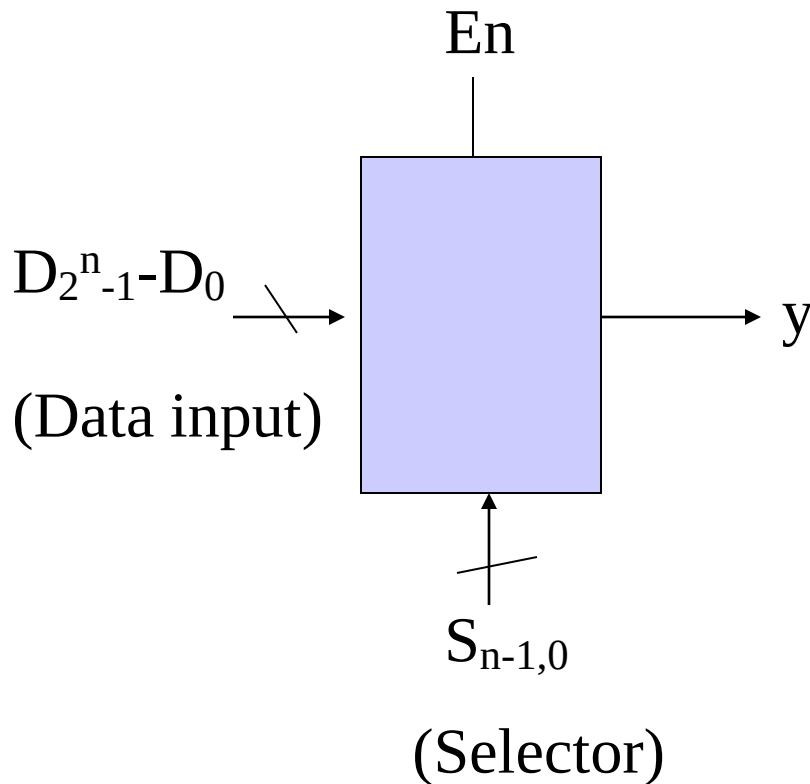
Decoder

Example 1: Given $f(a,b,c,d) = \sum m(0,2,4,6,8,9,10,13) + \sum d(3, 7, 12)$, implement the function using 2:4 decoders and OR gates



Multiplexer

A digital module that selects one of data inputs according to the binary address of the selector.



Description

If $En = 1$

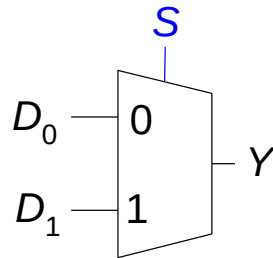
$y = D_i$ where $i = (S_{n-1}, \dots, S_0)$

Else

$y = 0$

Multiplexer

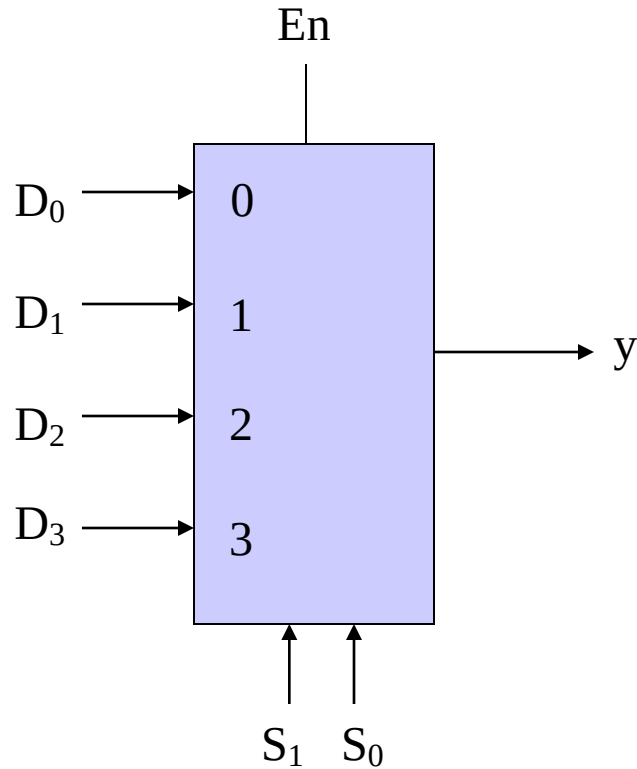
- Selects between one of N inputs to connect to the output.
- $\log_2 N$ -bit select input – control input
- Example:



2:1 Mux

S	D_1	D_0	Y	S	Y
0	0	0	0	0	D_0
0	0	1	1	1	D_1
0	1	0	0		
0	1	1	1		
1	0	0	0		
1	0	1	0		
1	1	0	1		
1	1	1	1		

Multiplexer Definition: Example



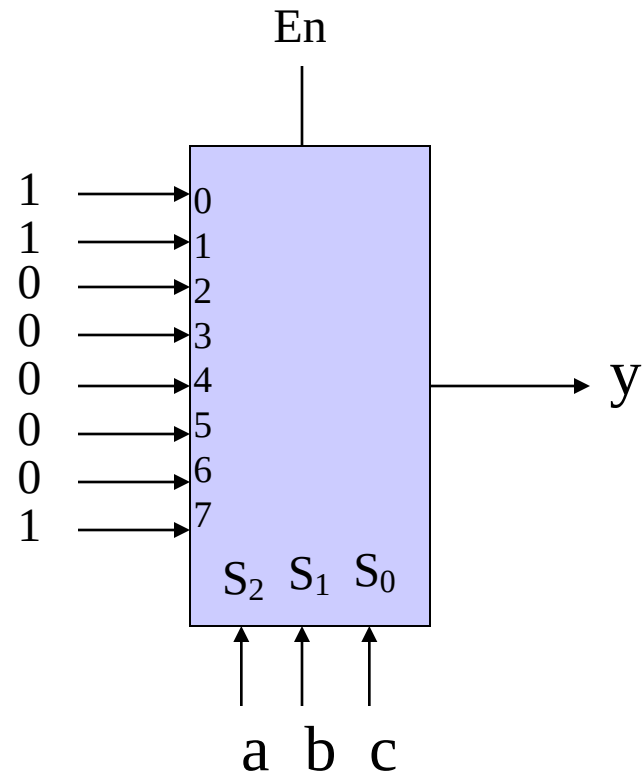
If $D_0 = 0$ and $S_1S_0 = 00 \Rightarrow y = 0$

If $D_0 = 1$ and $S_1S_0 = 00 \Rightarrow y = 1$

Multiplexer

Example 1: Given $f(a,b,c) = \sum m(0,1,7) + \sum d(2)$, implement with an 8-input Mux.

Id	a	b	c	f
0	0	0	0	1
1	0	0	1	1
2	0	1	0	X
3	0	1	1	0
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1



Multiplexer

Example 2: Given $f(a,b,c) = \sum m(0,1,7) + \sum d(2)$, implement with 2-input Muxes.

a	00	01	10	11	D (b,c)
0	1	1	X	0	$D_0(b,c)$
1	0	0	0	1	$D_1(b,c)$

KMAP

$$D_0(b,c) = b'$$

	1	X
c	1	0
	$\overline{b}$	

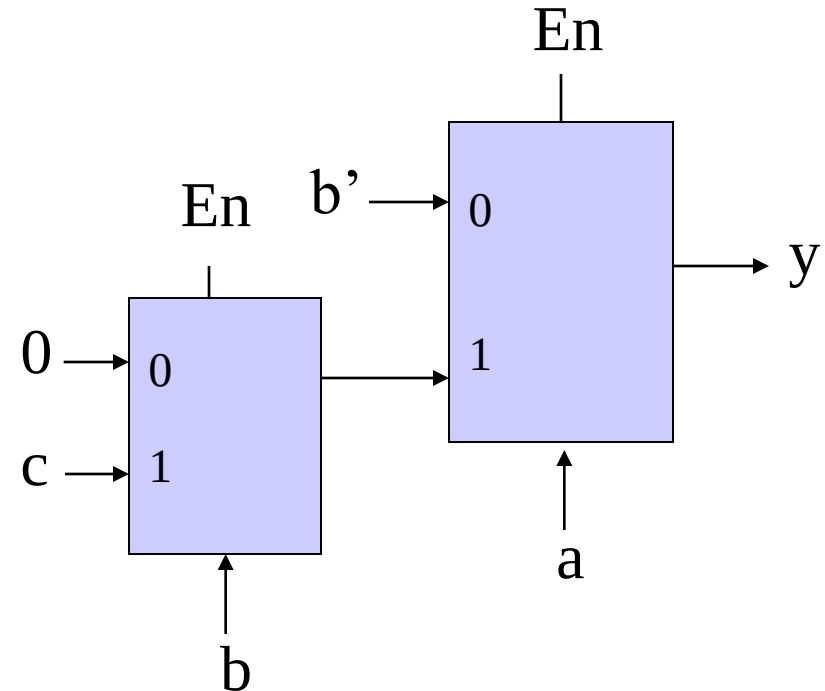
KMAP

$$D_1(b,c) = bc$$

	0	0
c	0	1
	$\overline{b}$	

$$D_1(b,c)$$

b	c = 0	c = 1	
0	0	0	$l_1(0) = 0$
1	0	1	$l_1(c) = c$



Multiplexer

Example 3: Given $f(a,b,c,d) = \sum m(0,2,4,6,8,9,10,13) + \sum d(3,7,12)$, implement with 2-input Muxes.

Id	a	b	c	d	f
0	0	0	0	0	1
1	0	0	0	1	0
2	0	0	1	0	1
3	0	0	1	1	X
4	0	1	0	0	1
5	0	1	0	1	0
6	0	1	1	0	1
7	0	1	1	1	X
8	1	0	0	0	1
9	1	0	0	1	1
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	X
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	0

	bcd								
a	000	001	010	011	100	101	110	111	L(b,c,d)
0	1	0	1	X	1	0	1	X	$L_0(b,c,d)$
1	1	1	1	0	X	1	0	0	$L_1(b,c,d)$

KMAP

cd 00 01 11 10

b	0	1
0	1	0
1	1	0

$$L_0(b,c,d) = d'$$

cd 00 01 11 10

b	0	1
0	1	1
1	X	1

$$L_1(b,c,d) = c' + b'd'$$

Multiplexer

Example 3 (continued):

b	cd				M (c,d)
	00	01	10	11	
0	1	1	1	0	$M_0(c,d)$
1	X	1	0	0	$M_1(c,d)$

KMAP

$$M_0(c,d) = c' + d'$$

c	d	
	0	1
0	1	1
1	1	0

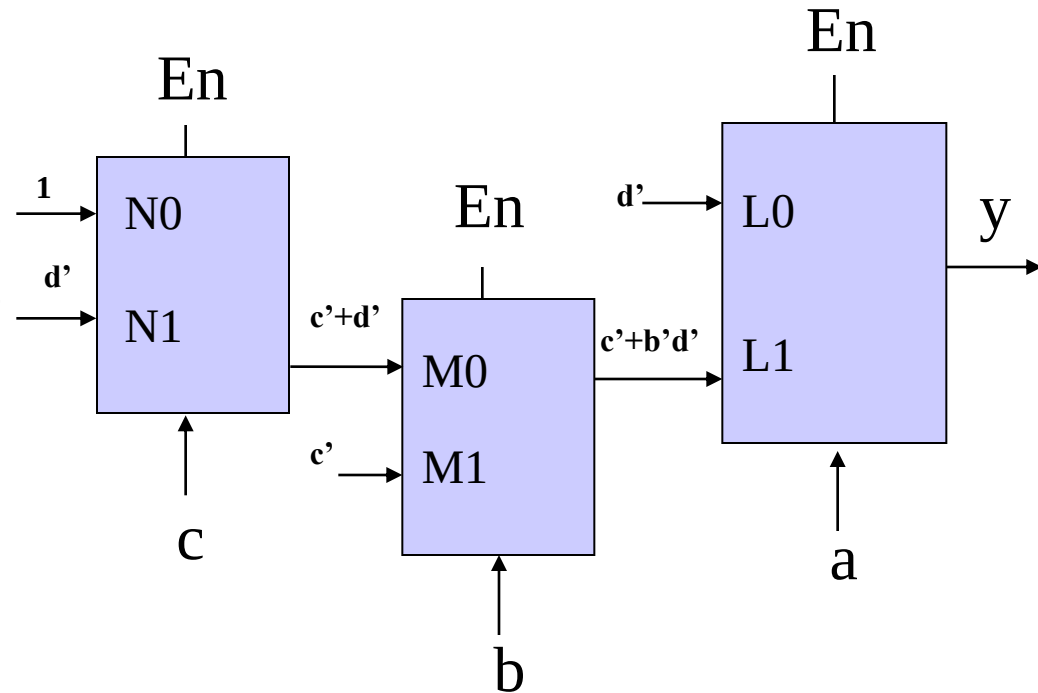
KMAP

$$M_1(c,d) = c'$$

c	d	
	0	1
0	X	1
1	0	0

$$M_0(c, d)$$

c	d = 0	d = 1	N (d)
0	1	1	$N_0(0) = 1$
1	1	0	$N_1(c) = d'$



CSE 140 Discussion

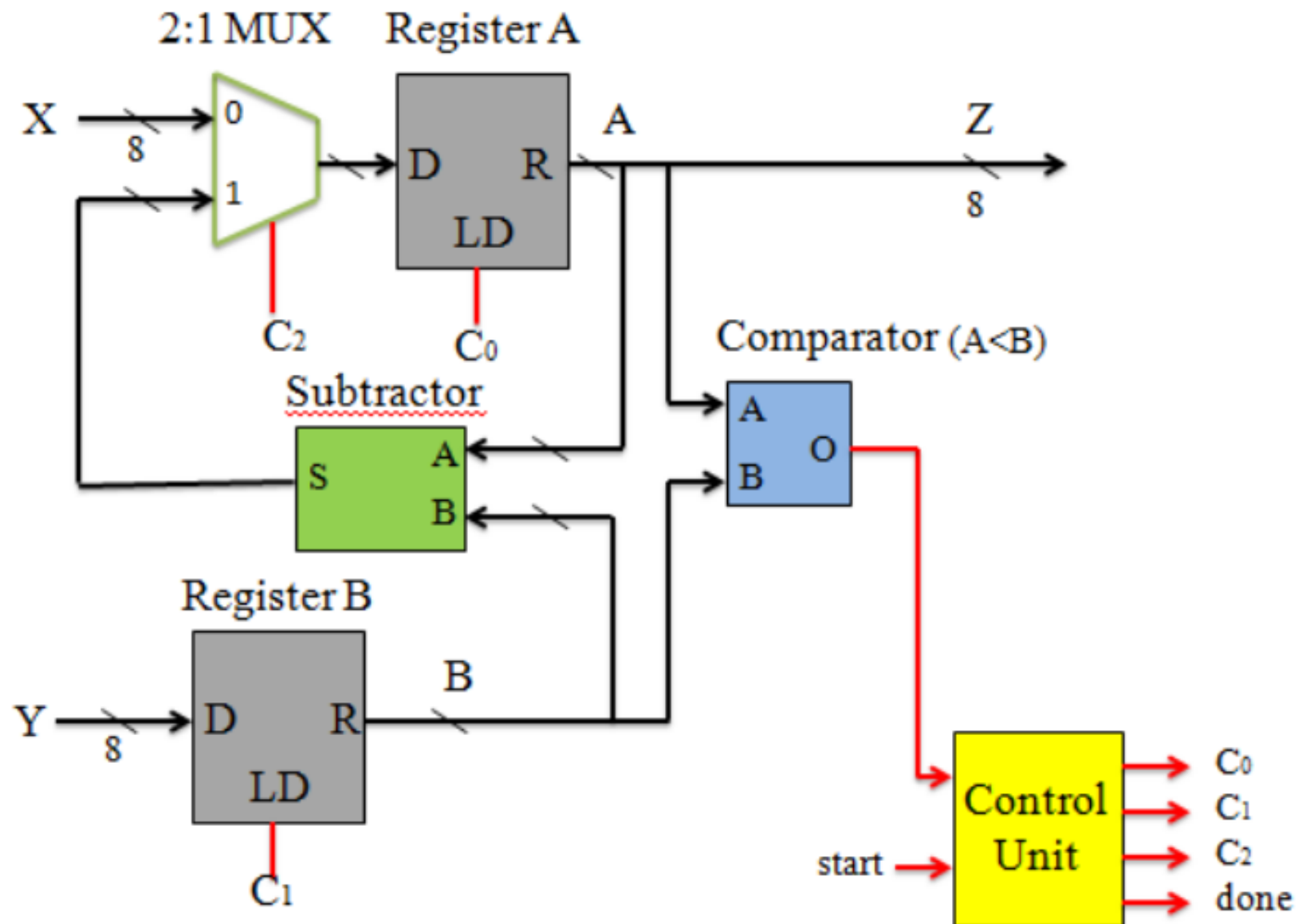
System Design

Example

- Design a data subsystem that is adequate to execute the algorithm and draw the schematic diagram

```
Alg(X, Y, Z, start, done)
Input X[0:7], Y[0:7], start;
Output Z[0:7], done;
Local-object A[0:7], B[0:7];
S1: If start' goto S1 ——— done <= 1;
S2: done <= 0 || A <= X || B <= Y;
S3: if (A < B) goto S5;
S4: A <= A - B || goto S3;
S5: Z <= A || done <= 1 || goto S1;
end Alg
```

SCHEMATIC DIAGRAM FOR DATA SUBSYSTEM



Statement	Operation	Control
$A \leq B$	$A < - \text{Load}(X)$	$C_0 = 1$ and $C_2 = 0$
$B \leq Y$	$B < - \text{Load}(Y)$	$C_1 = 1$
$A < B$	$\text{Comp}(A, B)$	
$A \leq A - B$	$A < - \text{Sub}(A, B)$	$C_0 = 1$ and $C_2 = 1$
$Z \leq A$	Wires	done

Table 1: Relationship between statement, operation and control

STATE DIAGRAM

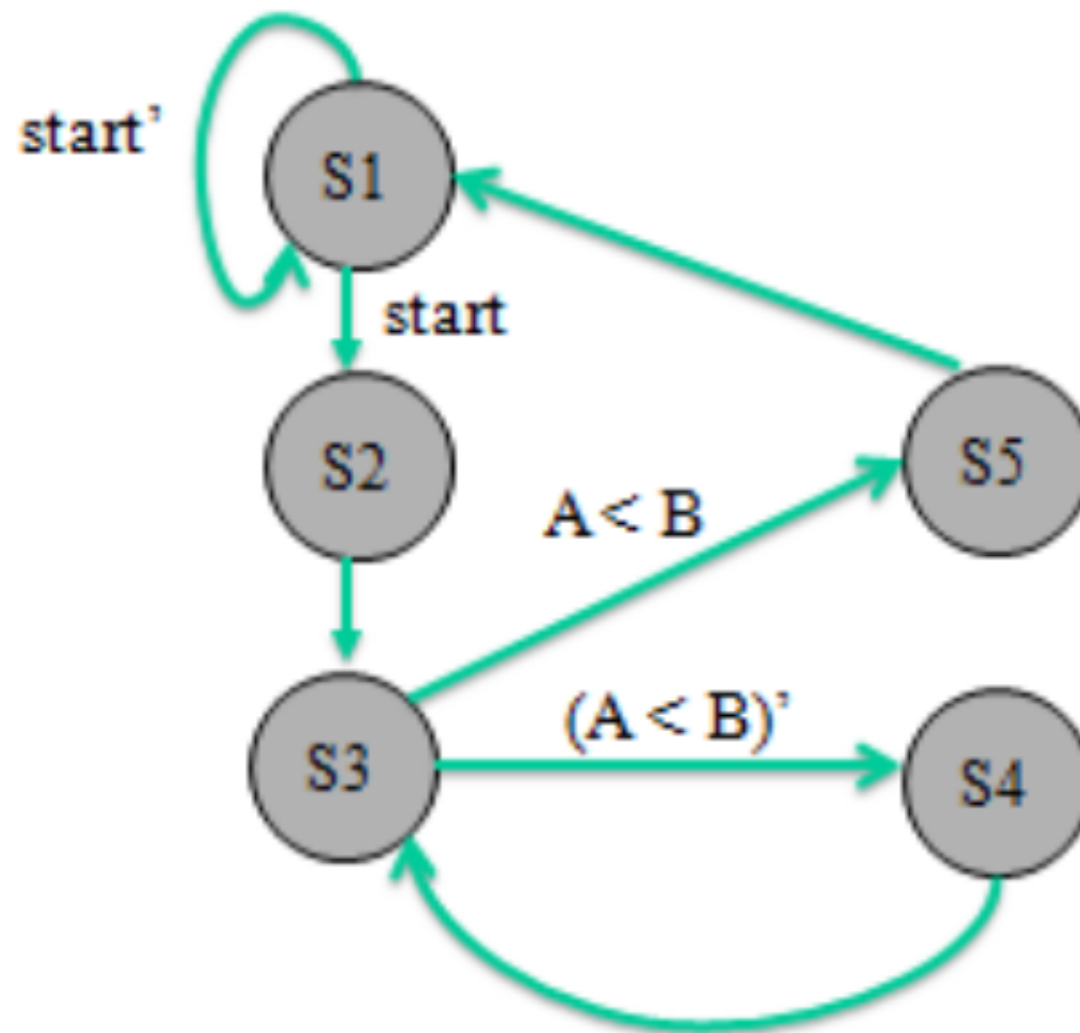


TABLE FOR CONTROL SUBSYSTEM

	C0	C1	C2 (mux)	done
S1	0	0	X	1
S2	1	1	0	0
S3	0	0	X	0
S4	1	0	1	0
S5	0	0	X	1

ONE-HOT STATE MACHINE

