

Midterm 3 Review for CSE 140

Chester Holtz

Based on Winter '14 slides

June 7, 2021

Logistics

- ▶ Released on course website:
<http://cseweb.ucsd.edu/classes/sp21/cse140-a/>
- ▶ Full 48 hours, submission on gradescope
- ▶ Released Saturday 6/5 2:30 pm PST, due Monday 6/7 2:30 pm PST
- ▶ Format same as midterm 1 & 2—2 sections:
 - ▶ Section A: 90% question-solving style, roughly 80 minutes
 - ▶ Section B: 10% exploratory style, no penalties

Overview

- ▶ Combinational Modules: Decoders & Multiplexers
- ▶ System Design

Decoders

Definitions

- ▶ Digital module, N inputs, 2^N outputs
- ▶ One-hot outputs: only one output HIGH (1) at most

Decoders

Definitions

- ▶ Digital module, N inputs, 2^N outputs
- ▶ One-hot outputs: only one output HIGH (1) at most

Can be used to implement Boolean functions

- ▶ Decoder produces minterms when $E=1$.
- ▶ Can use an OR gate to collect the minterms to cover the On-set.
- ▶ For the Don't Care-Set, we can just ignore the terms

Decoders

Example

$$f(a, b, c, d) = \sum m(12, 15)$$

Implement f using a 4 – 16 decoder and OR gates

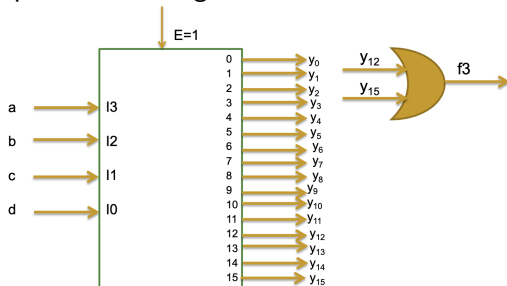
	0
	1
	2
I3	3
	4
I2	5
	6
	7
I1	8
	9
I0	10
	11
	12
	13
	14
	15

Decoders

Example

$$f(a, b, c, d) = \sum m(12, 15)$$

Implement f using a 4 – 16 decoder and OR gates



Decoders

Example

$$f(a, b, c, d) = \sum m(12, 15)$$

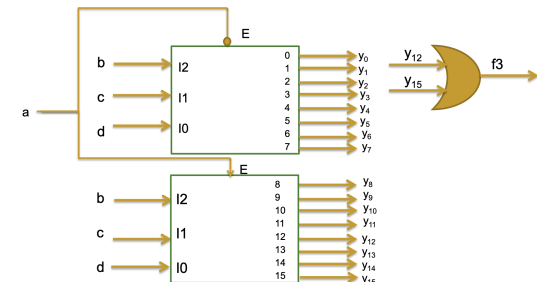
Implement f using 3 – 8 decoders and OR gates

Decoders

Example

$$f(a, b, c, d) = \sum m(12, 15)$$

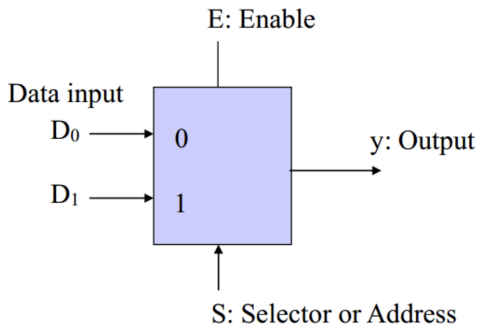
Implement f using 3 – 8 decoders and OR gates



Multiplexers

Definitions

- ▶ Selects one of N inputs to connect to the output
- ▶ $\log_2(N)$ -bit select input – control input



Multiplexers

Example

$$f(a, b, c) = \sum m(0, 3, 5, 7) + \sum d(6)$$

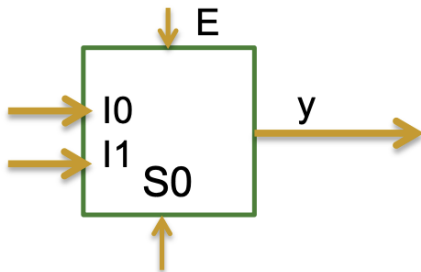
implement the function using a minimal network of 2:1 multiplexers.

Multiplexers

Example

$f(a, b, c)$

id	<i>a</i>	<i>b</i>	<i>c</i>	<i>f</i>
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	x
7	1	1	1	1



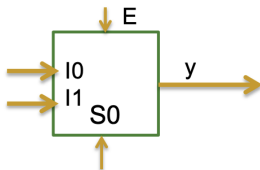
Multiplexers

Example

$$f(a, b, c) = \sum m(0, 3, 5, 7) + \sum d(6)$$

implement the function using a minimal network of 2:1 multiplexers.

id	a	b	c	f
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	x
7	1	1	1	1



bc	00	01	10	11	
a=0	1	0	0	1	$I0 = d0(bc)$
a=1	0	1	x	1	$I1 = d1(bc)$

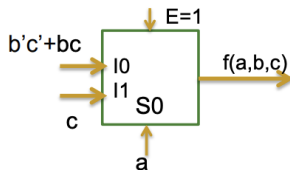
Multiplexers

Example

$$f(a, b, c) = \sum m(0, 3, 5, 7) + \sum d(6)$$

implement the function using a minimal network of 2:1 multiplexers.

bc	00	01	10	11	
a=0	1	0	0	1	$l0 = d0(bc)$
a=1	0	1	x	1	$l1 = d1(bc)$



c\b	0	1	c\b	0	1
0	1	0	0	0	x
1	0	1	1	1	1

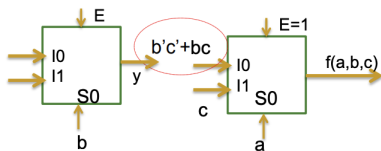
Multiplexers

Example

$$f(a, b, c) = \sum m(0, 3, 5, 7) + \sum d(6)$$

implement the function using a minimal network of 2:1 multiplexers.

c\b	0	1	c\b	0	1
0	1	0	0	0	x
1	0	1	1	1	1



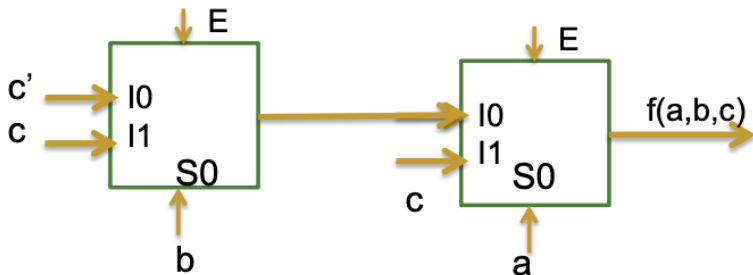
c	0	1	
b=0	1	0	I0 = c'
b=1	0	1	I1 = c

Multiplexers

Example

$$f(a, b, c) = \sum m(0, 3, 5, 7) + \sum d(6)$$

implement the function using a minimal network of 2:1 multiplexers.



System Design Overview

- ▶ Data Subsystem
 - ▶ Table of operations
 - ▶ Logic diagram
 - ▶ Control signals
- ▶ Control Subsystem
 - ▶ State diagram
 - ▶ Control signal for every state
 - ▶ One-hot state-machine

System Design: Data Subsystem

- ▶ S1: If start' goto S1 || done $\leftarrow$ 1
- ▶ S2: A $\leftarrow$ X || B $\leftarrow$ Y || C $\leftarrow$ (00000000) || done $\leftarrow$ 0;
- ▶ S3: A $\leftarrow$ Add(A, B) || C $\leftarrow$ Inc(C)
- ▶ S4: if A'[7] goto S3;
- ▶ S5: U $\leftarrow$ C || done $\leftarrow$ 1 || goto S1;

System Design: Data Subsystem

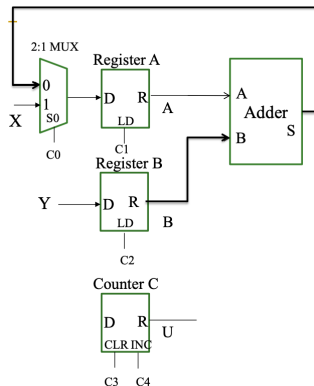
- ▶ S1: If start' goto S1 || done $\leftarrow$ 1
- ▶ S2: A $\leftarrow$ X || B $\leftarrow$ Y || C $\leftarrow$ (00000000) || done $\leftarrow$ 0;
- ▶ S3: A $\leftarrow$ Add(A, B) || C $\leftarrow$ Inc(C)
- ▶ S4: if A'[7] goto S3;
- ▶ S5: U $\leftarrow$ C || done $\leftarrow$ 1 || goto S1;

S	Stmt	Op	C
S2	A $\leftarrow$ X	A $\leftarrow$ Load(X)	
S2	B $\leftarrow$ Y	B $\leftarrow$ Load(Y)	
S2	C $\leftarrow$ 0	C $\leftarrow$ clear(C)	
S3	A $\leftarrow$ Add(A,B)	A $\leftarrow$ Add(A,B)	
S3	C $\leftarrow$ Inc(C)	C $\leftarrow$ Inc(C)	
S5	U $\leftarrow$ C	Wires	

System Design: Data Subsystem

- ▶ S1: If start' goto S1 || done $\leftarrow$ 1
- ▶ S2: A $\leftarrow$ X || B $\leftarrow$ Y || C $\leftarrow$ (00000000) || done $\leftarrow$ 0;
- ▶ S3: A $\leftarrow$ Add(A, B) || C $\leftarrow$ Inc(C)
- ▶ S4: if A'[7] goto S3;
- ▶ S5: U $\leftarrow$ C || done $\leftarrow$ 1 || goto S1;

S	Stmt	Op	C
S2	A $\leftarrow$ X	A $\leftarrow$ Load(X)	
S2	B $\leftarrow$ Y	B $\leftarrow$ Load(Y)	
S2	C $\leftarrow$ 0	C $\leftarrow$ clear(C)	
S3	A $\leftarrow$ Add(A,B)	A $\leftarrow$ Add(A,B)	
S3	C $\leftarrow$ Inc(C)	C $\leftarrow$ Inc(C)	
S5	U $\leftarrow$ C	Wires	



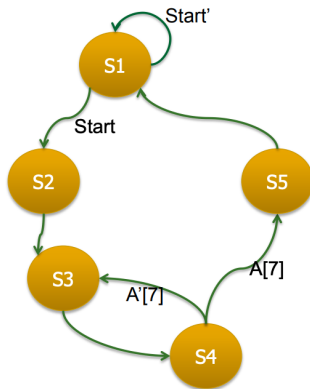
System Design: Control Subsystem

- ▶ S1: If start' goto S1 || done $\leftarrow$ 1
- ▶ S2: A $\leftarrow$ X || B $\leftarrow$ Y || C $\leftarrow$ (00000000) || done $\leftarrow$ 0;
- ▶ S3: A $\leftarrow$ Add(A, B) || C $\leftarrow$ Inc(C)
- ▶ S4: if A'[7] goto S3;
- ▶ S5: U $\leftarrow$ C || done $\leftarrow$ 1 || goto S1;

S	Stmt	Op	C
S2	A $\leftarrow$ X	A $\leftarrow$ Load(X)	C1=1, C0=1
S2	B $\leftarrow$ Y	B $\leftarrow$ Load(Y)	C2=1
S2	C $\leftarrow$ 0	C $\leftarrow$ clear(C)	C3=1
S3	A $\leftarrow$ Add(A,B)	A $\leftarrow$ Add(A,B)	C0=0, C1=1
S3	C $\leftarrow$ Inc(C)	C $\leftarrow$ Inc(C)	C4=1
S5	U $\leftarrow$ C	Wires	done

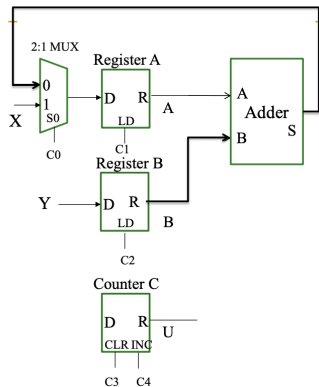
State Diagrams

- ▶ S1: If start' goto S1 || done $\leftarrow$ 1
- ▶ S2: A $\leftarrow$ X || B $\leftarrow$ Y || C $\leftarrow$ (00000000) || done $\leftarrow$ 0;
- ▶ S3: A $\leftarrow$ Add(A, B) || C $\leftarrow$ Inc(C)
- ▶ S4: if A'[7] goto S3;
- ▶ S5: U $\leftarrow$ C || done $\leftarrow$ 1 || goto S1;



Control Subsystem

- ▶ S1: If start' goto S1 || done $\leftarrow$ 1
- ▶ S2: A $\leftarrow$ X || B $\leftarrow$ Y || C $\leftarrow$ (00000000) || done $\leftarrow$ 0;
- ▶ S3: A $\leftarrow$ Add(A, B) || C $\leftarrow$ Inc(C)
- ▶ S4: if A'[7] goto S3;
- ▶ S5: U $\leftarrow$ C || done $\leftarrow$ 1 || goto S1;

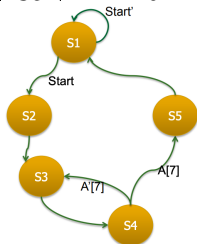


S	C0	C1	C2	C3	C4	done
S1	0	0	0	0	0	1
S2	1	1	1	1	0	0
S3	0	1	0	0	1	0
S4	0	0	0	0	0	0
S5	0	0	0	0	0	1

One-hot state machine

- ▶ S1: If start' goto S1 || done $\leftarrow$ 1
- ▶ S2: A $\leftarrow$ X || B $\leftarrow$ Y || C $\leftarrow$ (00000000) || done $\leftarrow$ 0;
- ▶ S3: A $\leftarrow$ Add(A, B) || C $\leftarrow$ Inc(C)
- ▶ S4: if A'[7] goto S3;
- ▶ S5: U $\leftarrow$ C || done $\leftarrow$ 1 || goto S1;

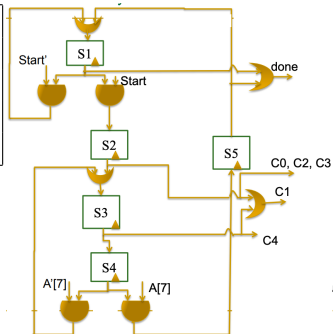
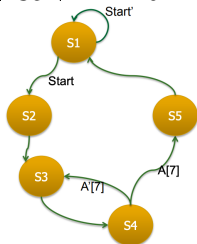
S	C0	C1	C2	C3	C4	done
S1	-	0	0	0	0	1
S2	1	1	1	1	0	0
S3	0	1	0	0	1	0
S4	-	0	0	0	0	0
S5	-	0	0	0	0	1



One-hot state machine

- ▶ S1: If start' goto S1 || done $\leftarrow$ 1
- ▶ S2: A $\leftarrow$ X || B $\leftarrow$ Y || C $\leftarrow$ (00000000) || done $\leftarrow$ 0;
- ▶ S3: A $\leftarrow$ Add(A, B) || C $\leftarrow$ Inc(C)
- ▶ S4: if A'[7] goto S3;
- ▶ S5: U $\leftarrow$ C || done $\leftarrow$ 1 || goto S1;

S	C0	C1	C2	C3	C4	done
S1	-	0	0	0	0	1
S2	1	1	1	1	0	0
S3	0	1	0	0	1	0
S4	-	0	0	0	0	0
S5	-	0	0	0	0	1



Some more tips

- ▶ Part A (problem-solving)
 - ▶ Similar to previous exams
 - ▶ 5 Subsections
 - ▶ Decoders & multiplexers
 - ▶ Sequential networks
 - ▶ System design
- ▶ Part B (exploratory)
 - ▶ 3 Subsections
 - ▶ Sequential networks
 - ▶ System design

Good Luck!