CompSci/ECE.350L.001.Sp21 Midterm 2

Rishi Salwi

TOTAL POINTS

98 / 100

QUESTION 1

1 5-Variable K-map and function 14 / 15

K-Map

✓ **+ 3 pts** *Correct entries on K-map*

+ 2 pts Mainly right but minority of cells wrong/transposed

+ 2 pts missed one entry

+ 1 pts badly transposed / many wrong

+ 2 pts right pattern but axes transposed

+ 1 pts left off don't cares, adds significantly to complexity

+ 1 pts Made the "don't cares" into 1s, adds significantly to complexity

Prime Implicants (given map used)

+ 6 pts all 9 prime implicants identified

✓ **+ 5 pts** *8 PI identified out of 9*

+ 4 pts 7 PI identified out of 9

+ 3 pts 6 PI identified out of 9

+ 2 pts 5 PI identified out of 9

+ 1 pts 4 PI identified out of 9

+ 0 pts 3 or fewer PI identified out of 9

+ 0 pts Not capturing

Function

✓ **+ 6 pts** *Plausible given map used (for correct map 7 terms x0'x2x3'+x1'x2'x3+x2'x3x4'+x2x3'x4'+x0'x1x2x4+x0x1x3'x4+(either x0x1'x3'x4' or x0x1'x2'x4'))*

+ 2 pts Unfinished or significantly incorrect

+ 5 pts 1 term too many

+ 4 pts 1 term too few

+ 0 pts Click here to replace this description.

QUESTION 2

Short Answer 20 pts

2.1 Structural Hazard 4 / 4

✓ **- 0 pts** *Correct*

- 1 pts pipelining actually exacerbates structural hazards (compared with a single-cycle machine) rather than solving them, but the pipeline structure overcomes this.

- 1 pts Pipelines are a source but not the only possible source of structural hazards - more generally any attempt to use the same resource twice

2.2 Data Hazard 3 / 4

- 0 pts Correct

- 1 *Point adjustment*

Not always on the same cycle, might not even be computed yet much less written back

2.3 Control Hazard 4 / 4

✓ **- 0 pts** *Correct*

2.4 Path Sensitization 4 / 4

✓ **- 0 pts** *Correct*

- 2 pts More specifically by setting other inputs so that a change of the input value on the current path of focus is reflected by a change in value on an observable output

- 1 pts important to include setting other inputs so that a change of the input on the selected path is reflected by a change of value on an observable output

2.5 Code Distance 4 / 4

✓ **- 0 pts** *Correct*

QUESTION 3

3 Pizza Emergency Button 40 / 40

✓ **- 0 pts** *[15 pts] Correct State Diagram (11 states, reset functionality)*

✓ **- 0 pts** *[9 pts] Correct State Table*

- 9 pts missing state table

✓ **- 0 pts** *[9 pts] Correct State Transition Table*

- 9 pts missing state transition table

- 3 pts missing output encodings in state transition table

✓ **- 0 pts** *[7 pts] Plausible Equations*

- 2 pts missing output equations

- 5 pts missing next state equations

QUESTION 4

4 FSM Minimization 25 / 25

✓ **- 0 pts** *Correct*

- 15 pts started

- 6 pts right initial table, incorrect processing

- 5 pts Correct grid, no final state table

1. Minimization

00010 01010 10010 | 01101
00011 01100 10011 | 11101
00100 01111 10100
00101 10000 11001
11010
11100

[15 points] Given

$f(x_0,x_1,x_2,x_3,x_4) = \sum m(2,3,4,5,10,12,15,16,18,19,20,25,26,28) + D(13,29)$

1 6789

Note carefully the labeling of the K-map. Use this 5-variable K-map to write a (possibly the) minimal SOP form of this function

$x_0=0$

x_1x_2 \ x_3x_4	00	01	11	10
00	0	0	1	1
01	1	1	0	0
11	1	d	1	0
10	0	0	0	1

$x_0=1$

x_1x_2 \ x_3x_4	00	01	11	10
00	1	0	1	1
01	1	0	0	0
11	1	d	0	0
10	0	1	0	1

Indicate prime implicants by circling them on the K-map, and write out the minimized function below

2|4

$f(x_0,x_1,x_2,x_3,x_4) = \overline{x_1}\,\overline{x_2}\,x_3 + \overline{x_2}\,x_3\,\overline{x_4} + \overline{x_0}\,x_2\,\overline{x_3} + x_2\,\overline{x_3}\,\overline{x_4} + \overline{x_0}\,x_1\,x_2\,x_4 + x_0\,x_1\,\overline{x_3}\,x_4 + x_0\,\overline{x_1}\,\overline{x_3}\,\overline{x_4}$

1 5-Variable K-map and function **14 / 15**

K-Map

✓ **+ 3 pts** *Correct entries on K-map*

+ 2 pts Mainly right but minority of cells wrong/transposed

+ 2 pts missed one entry

+ 1 pts badly transposed / many wrong

+ 2 pts right pattern but axes transposed

+ 1 pts left off don't cares, adds significantly to complexity

+ 1 pts Made the "don't cares" into 1s, adds significantly to complexity

Prime Implicants (given map used)

+ 6 pts all 9 prime implicants identified

✓ **+ 5 pts** *8 PI identified out of 9*

+ 4 pts 7 PI identified out of 9

+ 3 pts 6 PI identified out of 9

+ 2 pts 5 PI identified out of 9

+ 1 pts 4 PI identified out of 9

+ 0 pts 3 or fewer PI identified out of 9

+ 0 pts Not capturing

Function

✓ **+ 6 pts** *Plausible given map used (for correct map 7 terms*

x0'x2x3'+x1'x2'x3+x2'x3x4'+x2x3'x4'+x0'x1x2x4+x0x1x3'x4+(either x0x1'x3'x4' or x0x1'x2'x4'))

+ 2 pts Unfinished or significantly incorrect

+ 5 pts 1 term too many

+ 4 pts 1 term too few

+ 0 pts Click here to replace this description.

2. Short Answer

In 1-2 sentences, and in the space available or on comparable space on your own paper, describe the concepts listed below

4 points - Structural Hazard This occurs when multiple instructions need the same resource such as the read/write port on the register, memory, or the ALU

4 points - Data Hazard . Data hazards are data dependencies in processors. They occour when you are trying to to read or writed from a register that is curently being read/written to

4 points - Control Hazard
Control hazards occur when you execute the wrong instructions after a branch because you don't know where to branch yet.

4 points - Path Sensitization It is an algorithm used to identify the minimal test set for a circuit. It shows what path or inputs need to be taken for a specific input to reach the output

4 points - Code Distance It is a term used in error correcting/detecting codes. It is the number of bits that need to change in a code to produce another valid code

2.1 Structural Hazard 4 / 4

✓ **- 0 pts** *Correct*

- 1 pts pipelining actually exacerbates structural hazards (compared with a single-cycle machine) rather than solving them, but the pipeline structure overcomes this.

- 1 pts Pipelines are a source but not the only possible source of structural hazards - more generally any attempt to use the same resource twice

2. Short Answer

In 1-2 sentences, and in the space available or on comparable space on your own paper, describe the concepts listed below

4 points - Structural Hazard This occurs when multiple instructions need the same resource such as the read/write port on the register, memory, or the ALU

4 points - Data Hazard . Data hazards are data dependencies in processors. They ocour when you are trying to to read or writed from a register that is curently being read/written to

4 points - Control Hazard
Control hazards occur when you execute the wrong instructions after a branch because you don't know where to branch yet.

4 points - Path Sensitization It is an algorithm used to identify the minimal test set for a circuit. It shows what path or inputs need to be taken for a specific input to reach the output

4 points - Code Distance It is a term used in error correcting/detecting codes. It is the number of bits that need to change in a code to produce another valid code

2.2 Data Hazard 3 / 4

- **- 0 pts** Correct

- 1 *Point adjustment*

Not always on the same cycle, might not even be computed yet much less written back

2. Short Answer

In 1-2 sentences, and in the space available or on comparable space on your own paper, describe the concepts listed below

4 points - Structural Hazard This occurs when multiple instructions need the same resource such as the read/write port on the register, memory, or the ALU

4 points - Data Hazard , Data hazards are data dependencies in processors. They ocour when you are trying to to read or writed from a register that is curently being read/written to

4 points - Control Hazard

Control hazards occur when you execute the wrong instructions after a branch because you don't Know where to branch yet.

4 points - Path Sensitization It is an algorithm used to identify the minimal test set for a circuit. It shows what path or inputs need to be taken for a specific input to reach the output

4 points - Code Distance It is a term used in error correcting/ detecting codes. It is the number of bits that need to change in a code to produce another valid code

Scanned with CamScanner

2.3 Control Hazard **4 / 4**

✓ **- 0 pts** *Correct*

2. Short Answer

In 1-2 sentences, and in the space available or on comparable space on your own paper, describe the concepts listed below

4 points - Structural Hazard This occurs when multiple instructions need the same resource such as the read/write port on the register, memory, or the ALU

4 points - Data Hazard , Data hazards are data dependencies in processors. They ocour when you are trying to to read or writed from a register that is curently being read/written to

4 points - Control Hazard

Control hazards occur when you execute the wrong instructions after a branch because you don't know where to branch yet.

4 points - Path Sensitization It is an algorithm used to identify the minimal test set for a circuit. It shows what path or inputs need to be taken for a specific input to reach the output

4 points - Code Distance It is a term used in error correcting/detecting codes. It is the number of bits that need to change in a code to produce another valid code

Scanned with CamScanner

2.4 Path Sensitization 4 / 4

✓ **- 0 pts** *Correct*

- 2 pts More specifically by setting other inputs so that a change of the input value on the current path of focus is reflected by a change in value on an observable output

- 1 pts important to include setting other inputs so that a change of the input on the selected path is reflected by a change of value on an observable output

2. Short Answer

In 1-2 sentences, and in the space available or on comparable space on your own paper, describe the concepts listed below

4 points - Structural Hazard This occurs when multiple instructions need the same resource such as the read/write port on the register, memory, or the ALU

4 points - Data Hazard Data hazards are data dependencies in processors. They occour when you are trying to to read or writed from a register that is curently being read/written to

4 points - Control Hazard
Control hazards occur when you execute the wrong instructions after a branch because you don't know where to branch yet.

4 points - Path Sensitization It is an algorithm used to identify the minimal test set for a circuit. It shows what path or inputs need to be taken for a specific input to reach the output

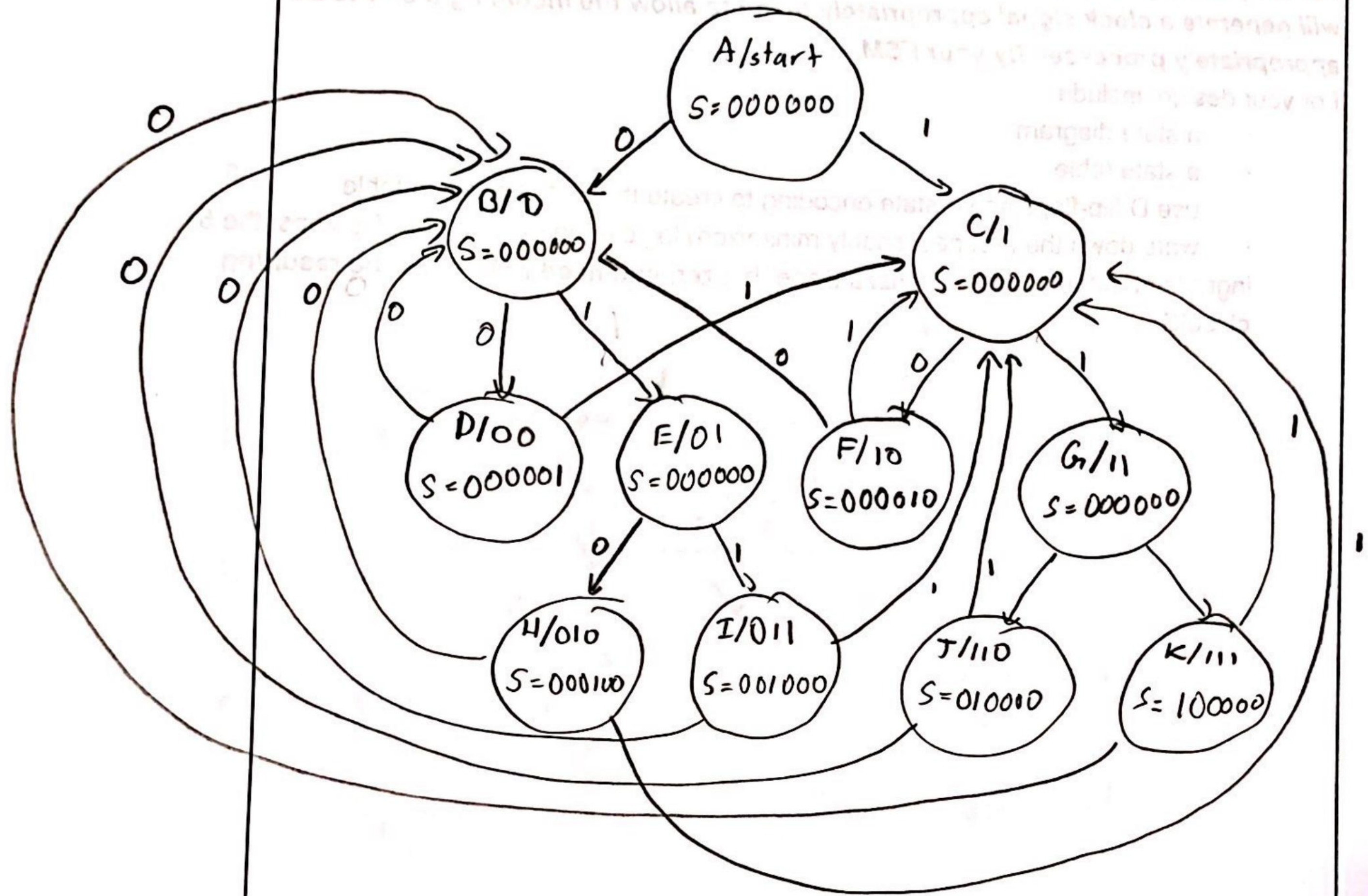
4 points - Code Distance It is a term used in error correcting/detecting codes. It is the number of bits that need to change in a code to produce another valid code

Scanned with CamScanner

2.5 Code Distance **4 / 4**

✓ **- 0 pts** *Correct*

State Diagram



pineapple
extra cheese

Output S = X X X X X X

end of pizza
onions
sausage
pepperoni

State Table (Symbolic)

Present State	Next State w=0	Next State w=1	Output	
A	B	C	000 000	nothing
B	D	E	000 000	nothing
C	F	G	000 000	nothing
D	B	C	000 001	extra cheese
E	H	I	000 000	nothing
F	B	C	000 010	pepperoni
G	J	K	000 000	nothing
H	B	C	000 100	sausage
I	B	C	001 000	pineapple
J	B	C	010 000	onions
K	B	C	100 000	end of pizza

State Transition Table (State encoding, D flip flops)

	Present State $y_3 y_2 y_1 y_0$	Next State w=0 $Y_3 Y_2 Y_1 Y_0$	Next State w=1 $Y_3 Y_2 Y_1 Y_0$	Output $S_5 S_4 S_3 S_2 S_1 S_0$
A	0000	0001	0010	000000
B	0001	0011	0100	000000
C	0010	0101	0110	000000
D	0011	0001	0010	000001
E	0100	0111	1000	000000
F	0101	0001	0010	000010
G	0110	1001	1010	000000
H	0111	0001	0010	000100
I	1000	0001	0010	001000
J	1001	0001	0010	010000
K	1010	0001	0010	100000

Equations

$Y_3 = D_3 = \overline{y_3}\, y_2\, y_1\, \overline{y_0} + \overline{y_3}\, y_2\, \overline{y_1}\, \overline{y_0}\, w$

$Y_2 = D_2 = \overline{y_3}\, \overline{y_2}\, y_1\, \overline{y_0}\, \overline{w} + \overline{y_3}\, y_2\, \overline{y_1}\, \overline{y_0}\, \overline{w} + \overline{y_3}\, \overline{y_2}\, \overline{y_1}\, y_0\, w + \overline{y_3}\, \overline{y_2}\, y_1\, \overline{y_0}\, w$

$Y_1 = D_1 = \overline{y_3}\, \overline{y_2}\, \overline{y_1}\, y_0\, \overline{w} + \overline{y_3}\, y_2\, \overline{y_1}\, \overline{y_0}\, \overline{w} + \overline{y_3}\, \overline{y_2}\, \overline{y_1}\, y_0\, w + \overline{y_3}\, y_2\, \overline{y_1}\, \overline{y_0}\, w$

$Y_0 = D_0 = \overline{w}$

S_0 = extra cheese = $\overline{y_3}\, \overline{y_2}\, y_1\, y_0$

S_1 = pepperoni = $\overline{y_3}\, y_2\, \overline{y_1}\, y_0$

S_2 = sausage = $\overline{y_3}\, y_2\, y_1\, y_0$

S_3 = pineapple = $y_3\, \overline{y_2}\, \overline{y_1}\, \overline{y_0}$

S_4 = onions = $y_3\, \overline{y_2}\, \overline{y_1}\, y_0$

S_5 = end of pizza = $y_3\, \overline{y_2}\, y_1\, \overline{y_0}$

3 Pizza Emergency Button **40 / 40**

✓ **- 0 pts** *[15 pts] Correct State Diagram (11 states, reset functionality)*

✓ **- 0 pts** *[9 pts] Correct State Table*

- 9 pts missing state table

✓ **- 0 pts** *[9 pts] Correct State Transition Table*

- 9 pts missing state transition table

- 3 pts missing output encodings in state transition table

✓ **- 0 pts** *[7 pts] Plausible Equations*

- 2 pts missing output equations

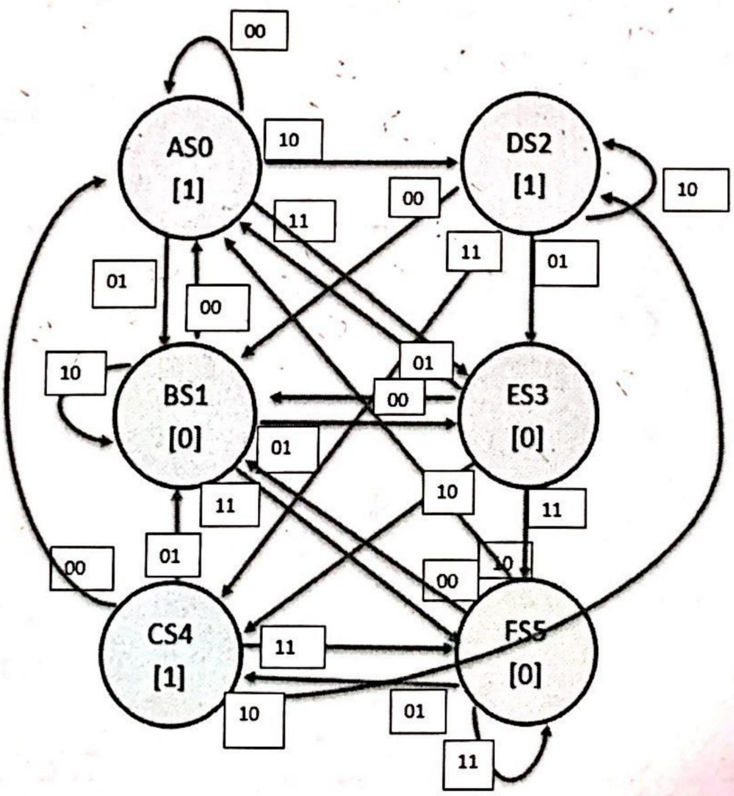
- 5 pts missing next state equations

4. FSM Minimization

[25 pts] Given the following finite state machine (the state diagram is admittedly confusing so go by the state table for the "official" definition), use the implication chart method to see if it can be reduced to an equivalent machine with fewer states. Give the reduced state table if so. A blank implication chart is provided on the next page. Score will be based on completion of the implication chart and the final state table.

Implication chart method to minimize this Moore FSM

	Next State: w0 w1				
Present State	00	01	10	11	Output
A	A	B	D	E	1
B	A	E	B	F	0
C	A	B	D	F	1
D	B	E	D	C	1
E	B	A	C	F	0
F	B	C	A	F	0



B	X				
C	A – A B – B D – D E – F	X			
D	~~A – B~~ ~~B – E~~ ~~D – D~~ ~~E – C~~	X	~~A – B~~ ~~B – E~~ ~~D – D~~ ~~F – C~~		
E	X	~~A – B~~ ~~E – A~~ ~~B – C~~ ~~F – F~~	X	X	
F	X	~~A – B~~ ~~E – C~~ ~~B – A~~ ~~F – F~~	X	X	B – B A – C C – A F – F
	A	B	C	D	E

A=C

E=F

Final State Table

Present State	Next state $w_0 w_1$ 00	01	10	11	Output
A, C	A	B	D	E	1
B	A	E	B	E	0
D	B	E	D	A	1
E, F	B	A	A	E	0

4 FSM Minimization 25 / 25

✓ **- 0 pts** *Correct*

- 15 pts started

- 6 pts right initial table, incorrect processing

- 5 pts Correct grid, no final state table

Name

Rishi Salmi

CompSci/ECE350 Spring 2021 Midterm Examination 2 – 4/15/21

This is an open book, open notes, open internet exam designed to take 75 minutes, but you have 90 minutes to work (including time needed to deal with the logistics of accessing and submitting this exam). Please answer all 4 questions, and note that question 3 is worth more than the others (about double). If you can print and scan your exam, you can use the spaces provided (augmented with scratch paper), but you may also just answer on your own paper and submit your responses by scanning those instead. You may/should use extra pages to show your work; scan those too!

Please, if accurate, sign this statement before turning in your exam[1]: (or summarize on your own paper)

I have neither given nor received any unauthorized help on this examination.

I will be on the usual class lecture Zoom during the exam windows to answer any mid test questions. If you want to ask me a question over Zoom, I will move you to a breakout room as we did for the first midterm.

[1] If the statement is not accurate, we need to talk!

3. FSM Design

[40 points] A number of years ago (I am not making this up!), some MIT students designed a pizza-shaped Emergency Pizza Button which, when pressed, placed a pizza order on the early internet. Since there is just one button and not everyone wants the same pizza, they employ a Huffman code to encode desired ingredients, with more popular ingredients taking fewer button presses than less popular ones. Huffman codes have the property that, even though codewords are of different lengths, each can be unambiguously decoded. I presume the switch is a simple 2 position switch that can reliably send "0" or "1" (pressing the Pizza button down or up).

Given the following Huffman code:

00	extra cheese
10	pepperoni
010	sausage
011	pineapple
110	onions
111	EndOfPizza

a Pizza Emergency Code of 0010010111 would mean "send a pizza with extra cheese, pepperoni, and sausage;" note so also would 0100010111 as by construction all the shorter codes are distinct from all the longer codes so the sequence is unambiguous. The EndOfPizza code must be appended to the end of each message so we know when a pizza is complete. Sending the same code twice (or more) for the same pizza is ok, ..1010... will cause double pepperoni, and will have caused your FSM to fire the Pepperoni output twice.

Design a Moore-style finite state machine to process incoming pizza emergency messages, signaling via output signals which ingredients to include on the pizza. (Note a pizza code of 111 would indicate a plain pizza with no special ingredients, i.e. no ingredient output signals active.) Hint: You will need 6 separate 1-bit outputs, one for each ingredient, and "Pizza Done".

For simplicity, you may assume the system's clocking "just works," each button press will generate a clock signal appropriately timed to allow the incoming 0 or 1 to be appropriately processed by your FSM.

For your design, include

- a state diagram
- a state table
- use D flip-flops and a state encoding to create the state transition table
- write down the (not necessarily minimized) logic equations for your flip-flops, the 5 ingredient outputs, and the "Pizza Done" buzzer; **you need NOT draw the resulting circuit!**