

CSE 140 Spring 2017: Final Solutions (Total 50 Points)

1. (Boolean Algebra) Prove the following Boolean theorem using Boolean laws only, i.e. no theorem is allowed for the proof. State the name of the law for each step of your proof.

$$x + 1 = 1 \tag{1}$$

Solution:

$$\begin{aligned} x + 1 &= (x + 1).1 && [\text{Identity Law}] \\ &= (x + 1).(x + x') && [\text{Complement Law}] \\ &= x + (1.x') && [\text{Distributive Law}] \\ &= x + x' && [\text{Identity Law}] \\ &= 1 && [\text{Complement Law}] \end{aligned}$$

OR

$$\begin{aligned} x + 1 &= x + (x + x') && [\text{Complement Law}] \\ &= (x + x) + x' && [\text{Associative Law}] \\ &= x + x' && [\text{Idempotent Law}] \\ &= 1 && [\text{Complement Law}] \end{aligned}$$

OR

Any other valid solution.

Rubrics:

5 points, if all the laws plus equations are correct and there are no missing steps.
Subtract 1 point for every incorrect law name.
Subtract 2 points for every missing step (missing equation plus missing law).
Subtract 2 points, if only Annulment Law is used to prove the theorem.
If there are more mistakes than the total points or if the entire proof is incorrect, then give 0 points.

2. (Switching Function) A set of switching variables a, b, c, d, e has the values defined by the following simultaneous switching equations.

$$a + b + c = 1 \tag{2}$$

$$b' + c' + d = 1 \tag{3}$$

$$c + d' + e = 1 \tag{4}$$

$$a + c + e' = 1 \tag{5}$$

$$a' + b' + d = 1 \tag{6}$$

Use a switching function $f(a, b, c, d, e)$ to cover the values of the variables as its On-Set F . In other words, a min-term m_i belongs to F if and only if the corresponding binary

code $i = (a, b, c, d, e)$ satisfies the above equations.

2.1 Check which of the following min-terms belong to On-Set F .

$$m_0, m_5, m_{10}, m_{15}, m_{30}. \quad (7)$$

Solution:

$$\mathbf{m}_0 = (0, 0, 0, 0, 0) :$$

$$(2) \ a + b + c = 0 + 0 + 0 = 0$$

m_0 does not belong to On-Set F .

$$\mathbf{m}_5 = (0, 0, 1, 0, 1) :$$

$$(2) \ a + b + c = 0 + 0 + 1 = 1$$
$$(3) \ b' + c' + d = 0' + 1' + 0 = 1$$
$$(4) \ c + d' + e = 1 + 0' + 1 = 1$$
$$(5) \ a + c + e' = 0 + 1 + 1' = 1$$
$$(5) \ a' + b' + d = 0' + 0' + 0 = 1$$

m_5 belongs to On-Set F .

$$\mathbf{m}_{10} = (0, 1, 0, 1, 0) :$$

$$(4) \ c + d' + e = 0 + 1' + 0 = 0$$

m_{10} does not belong to On-Set F .

$$\mathbf{m}_{15} = (0, 1, 1, 1, 1) :$$

$$(2) \ a + b + c = 0 + 1 + 1 = 1$$
$$(3) \ b' + c' + d = 1' + 1' + 1 = 1$$
$$(4) \ c + d' + e = 1 + 1' + 1 = 1$$
$$(5) \ a + c + e' = 0 + 1 + 1' = 1$$
$$(5) \ a' + b' + d = 0' + 1' + 1 = 1$$

m_{15} belongs to On-Set F .

$$\mathbf{m}_{30} = (1, 1, 1, 1, 0) :$$

$$(2) \ a + b + c = 1 + 1 + 1 = 1$$
$$(3) \ b' + c' + d = 1' + 1' + 1 = 1$$
$$(4) \ c + d' + e = 1 + 1' + 0 = 1$$
$$(5) \ a + c + e' = 1 + 1 + 0' = 1$$
$$(5) \ a' + b' + d = 1' + 1' + 1 = 1$$

m_{30} belongs to On-Set F .

m_5, m_{15}, m_{30} belong to On-Set F .

m_0, m_{10} do not belong to On-Set F .

Rubrics:

5 points, one point for every correct min-term answer i.e. whether it belongs or does not belong to the On-Set F.

2.2 Express function $f(a, b, c, d, e)$ in a minimal sum of products form. Describe the number of product terms and the number of literals.

Solution:

The given simultaneous equations can be represented in the POS form:
 $f(a, b, c, d, e) = (a + b + c).(b' + c' + d).(c + d' + e).(a + c + e').(a' + b' + d)$

N	a	b	c	d	e	$f(a, b, c, d, e)$
0	0	0	0	0	0	0
1	0	0	0	0	1	0
2	0	0	0	1	0	0
3	0	0	0	1	1	0
4	0	0	1	0	0	1
5	0	0	1	0	1	1
6	0	0	1	1	0	1
7	0	0	1	1	1	1
8	0	1	0	0	0	1
9	0	1	0	0	1	0
10	0	1	0	1	0	0
11	0	1	0	1	1	0
12	0	1	1	0	0	0
13	0	1	1	0	1	0
14	0	1	1	1	0	1
15	0	1	1	1	1	1
16	1	0	0	0	0	1
17	1	0	0	0	1	1
18	1	0	0	1	0	0
19	1	0	0	1	1	1
20	1	0	1	0	0	1
21	1	0	1	0	1	1
22	1	0	1	1	0	1
23	1	0	1	1	1	1
24	1	1	0	0	0	0
25	1	1	0	0	1	0
26	1	1	0	1	0	0
27	1	1	0	1	1	1
28	1	1	1	0	0	0
29	1	1	1	0	1	0
30	1	1	1	1	0	1
31	1	1	1	1	1	1

		a=0				a=1			
		bc				bc			
		00	01	11	10	00	01	11	10
		c				c			
de	00	0	1	0	1	1	1	0	0
	01	0	1	0	0	1	1	0	0
	11	0	1	1	0	1	1	1	1
	10	0	1	1	0	0	1	1	0
		b				b			
		e				e			
		d				d			
		a				a			

Minimal SOP form = $a'bc'd'e' + b'c + cd + ade + ab'd'$
Number of product terms = 5
Number of literals = 15

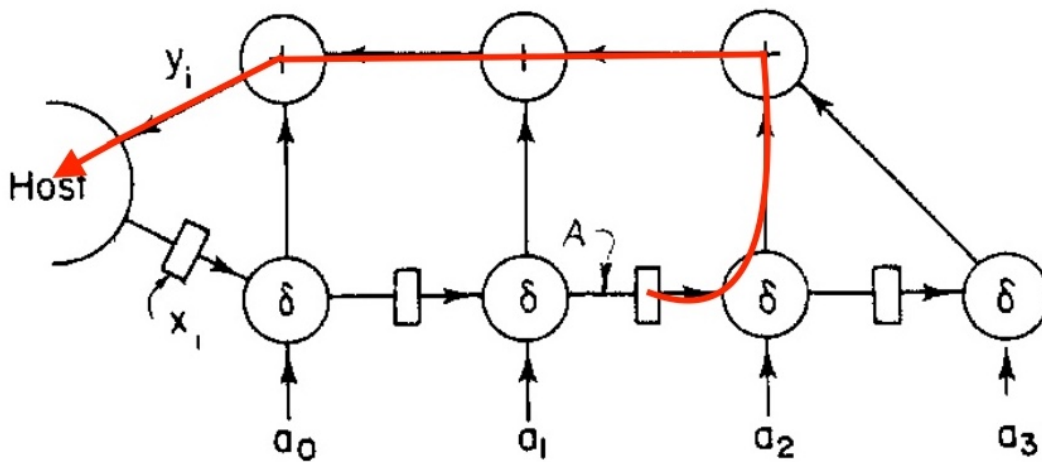
Rubrics:

- 10 points**, if correct Minimal SOP form, # of product terms and # of literals.
- Subtract 2 points, if incorrect # of product terms.
- Subtract 2 points, if incorrect # of literals.
- Subtract 2 points, if not minimal but correct SOP form (Eg. if only min-terms are specified).
- Subtract 4 points, if incorrect SOP form and some work is shown like Truth table or Kmap.
- If not attempted, then give 0 points.

3. (Retiming) Check the digital correlator circuit in paper, C. Leiserson and J. Saxe, "Retiming Synchronous Circuitry," Algorithmica, pp. 6:5-35, 1991. (The paper is posted with the exam. You are encouraged to read the approaches and references of the paper, but it is not required.) Suppose that each comparator propagation delay is changed to 10 picosecond and each adder + propagation delay is changed to 50 picosecond (changed from the numbers in the paper).

3.1 For the digital correlator circuit in figure 1 of the paper, what is the clock period of the circuit (the delay of a longest path of combinational rippling)?

Solution:



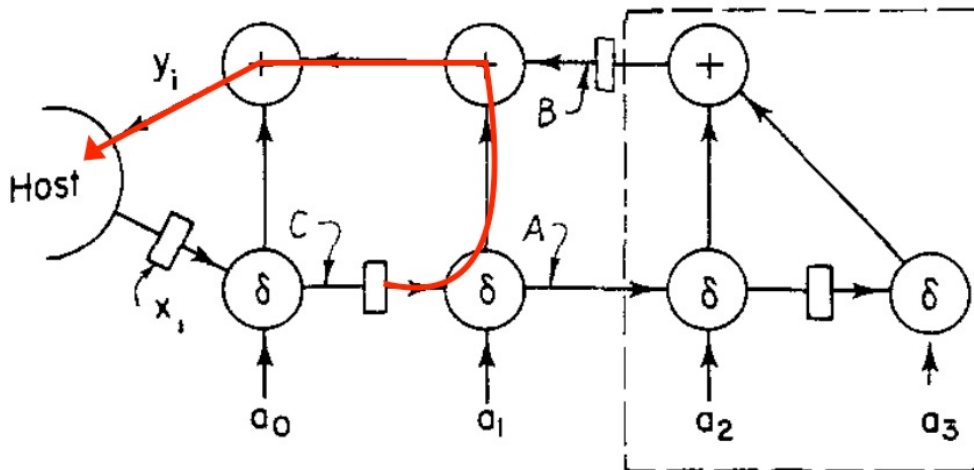
Clock Period = 1 Comparator Propagation Delay + 3 Adder Propagation Delay = $10 + (3 * 50) = 10 + 150 = 160$ ps.

Rubrics:

3 points, if correct answer. No partial credit.

3.2 For the circuit in figure 2 of the paper, what is the clock period of the circuit (the delay of a longest path of combinational rippling)?

Solution:



Clock Period = 1 Comparator Propagation Delay + 2 Adder Propagation Delay = $10 + (2 * 50) = 10 + 100 = 110$ ps.

Rubrics:

3 points, if correct answer. No partial credit.

3.3 For the same path between figure 1 and figure 2, that starts from and loops back to the Host, the loop traverses the same number of registers. Explain that the host should experience the same responses from the two circuits in figures 1 and 2. State your explanation in no more than four sentences.

Solution:

Consider the portion of the circuit surrounded by the dashed box in the figure. It communicates with the rest of the circuit only through connections A and B. When the register on A is removed, all input signals to this portion of the circuit arrive one clock tick earlier, and thus the boxed portion of Correlator 2 performs the same sequence of computations as in Correlator 1, but one clock tick earlier. Since the output from the boxed portion of Correlator 2 is delayed one clock tick by the new register on connection B, the remainder of the circuit sees the same behavior as in Correlator 1. Hence, both the figures are functionally equivalent and the host experiences the same responses from them.

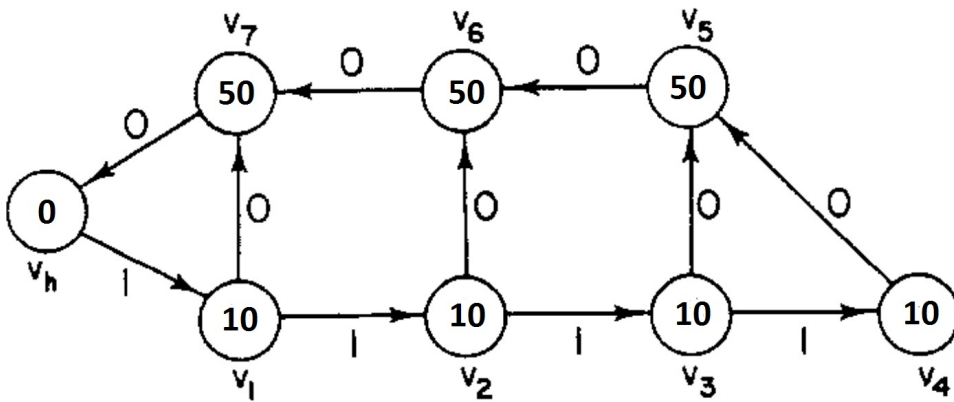
Rubrics:

1 point, for completion.

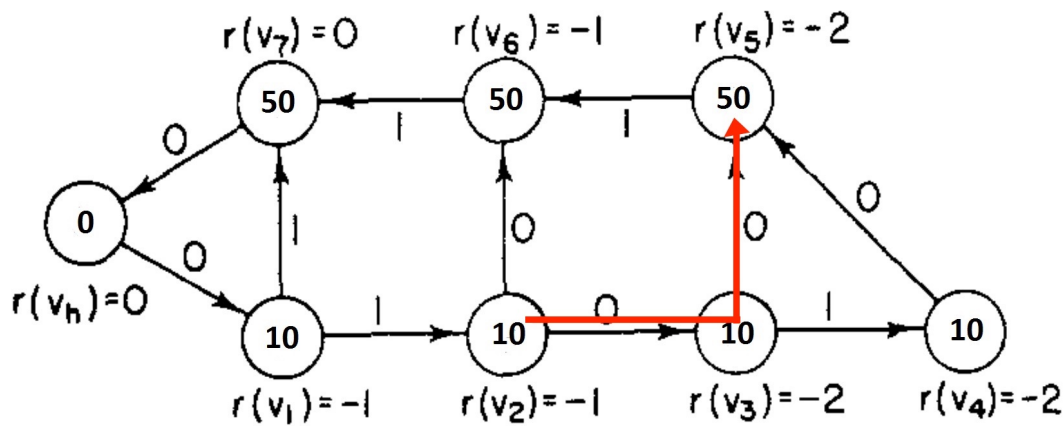
3.4 With aggressive retiming, what is the minimal clock period that can be achieved for the digital correlator in this paper? Hint: check figures 3 and 4 of the paper.

Solution:

Original Digital Correlator:



After Aggressive Retiming:



Minimal Clock Period = 2 Comparator Propagation Delay + 1 Adder Propagation Delay = $(2 * 10) + 50 = 20 + 50 = 70$ ps.

Rubrics:

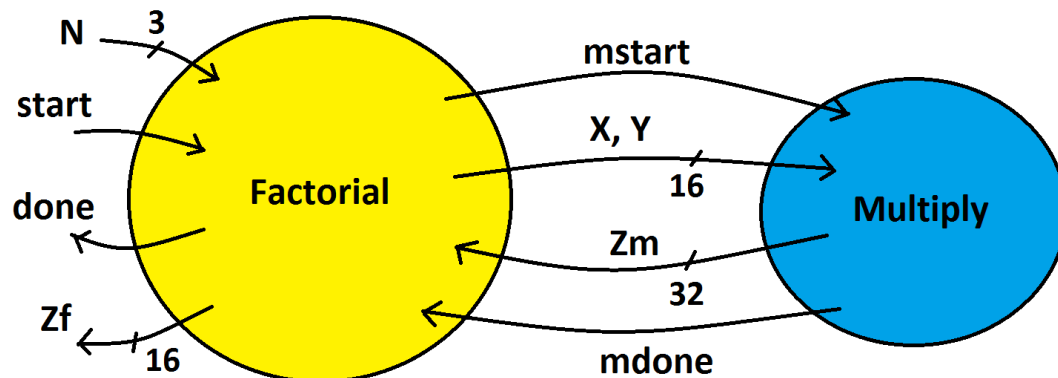
3 points, if correct answer. No partial credit.

4. (System Designs) Follow the style of the routine *Multiply* in slide 5 of lecture 14. Suppose that module *Multiply*(*X*, *Y*, *Z*, *start*, *done*) is used for the multiplication operation. Design a system that inputs a 3-bit binary number *n* and outputs its factorial *n!*.

4.1. Write the program. Explain the interface with module *Multiply*(*X*, *Y*, *Z*, *start*, *done*).

Solution:

Interface:



Program:

```
Factorial(N, Zf, start, done, X, Y, Zm, mstart, mdone)
{
  Input: N[2:0] type bit-vector, start type boolean, Zm[31:0] type bit-vector,
  mdone type boolean;
  Local-Object: F[15:0] type bit-vector, i[15:0] type bit-vector;
  Output: Zf[15:0] type bit-vector, done type boolean, mstart type boolean,
  X[15:0] type bit-vector, Y[15:0] type bit-vector;

  S0: if start' goto S0 || done = 1;
  S1: F = 1 || i = 1 || done = 0 || if N == 0, goto S8;
  S2: mstart = 0 || if mdone', goto S2;
  S3: if i == 1, goto S5
  S4: F = Zm[15:0]
  S5: if i >= N, goto S8;
  S6: i = i + 1 || X = F;
  S7: Y = i || mstart = 1 || if mdone, goto S7 else goto S2;
  S8: Zf = F || done = 1 || goto S0;
}
```

The module Multiply is executed every time we want to perform a multiplication when computing the factorial. F and i are connected to X and Y inputs of Multiply. mstart is connected to the input start of Multiply. The output Z of Multiply is connected to Zm. The output done of Multiply is connected to mdone. Initially mstart is set to 0, and we wait until mdone becomes 1. Then X and Y, are updated with the new values of F and i, and mstart is set to 1. We then wait for mdone to become 0, when this happens we know that Multiply has started execution. We then set mstart to 0 and wait for mdone to be 1. When this happens we know that Multiply has finished execution and that the multiplication result has been reflected in the input Zm, which we load to F. We then continue our program to compute the factorial.

Rubrics:

6 points, if the program is correct with explanation for the interface.

Subtract 1 point, if the interface explanation is missing.

Subtract 1 point, if there is a logical mistake in factorial calculation.

Subtract 0.5 points, if the check for mdone == 1 is missing before calling Multiply.

Subtract 0.5 points, if mdone == 0 is not checked and mstart = 0 is not set.

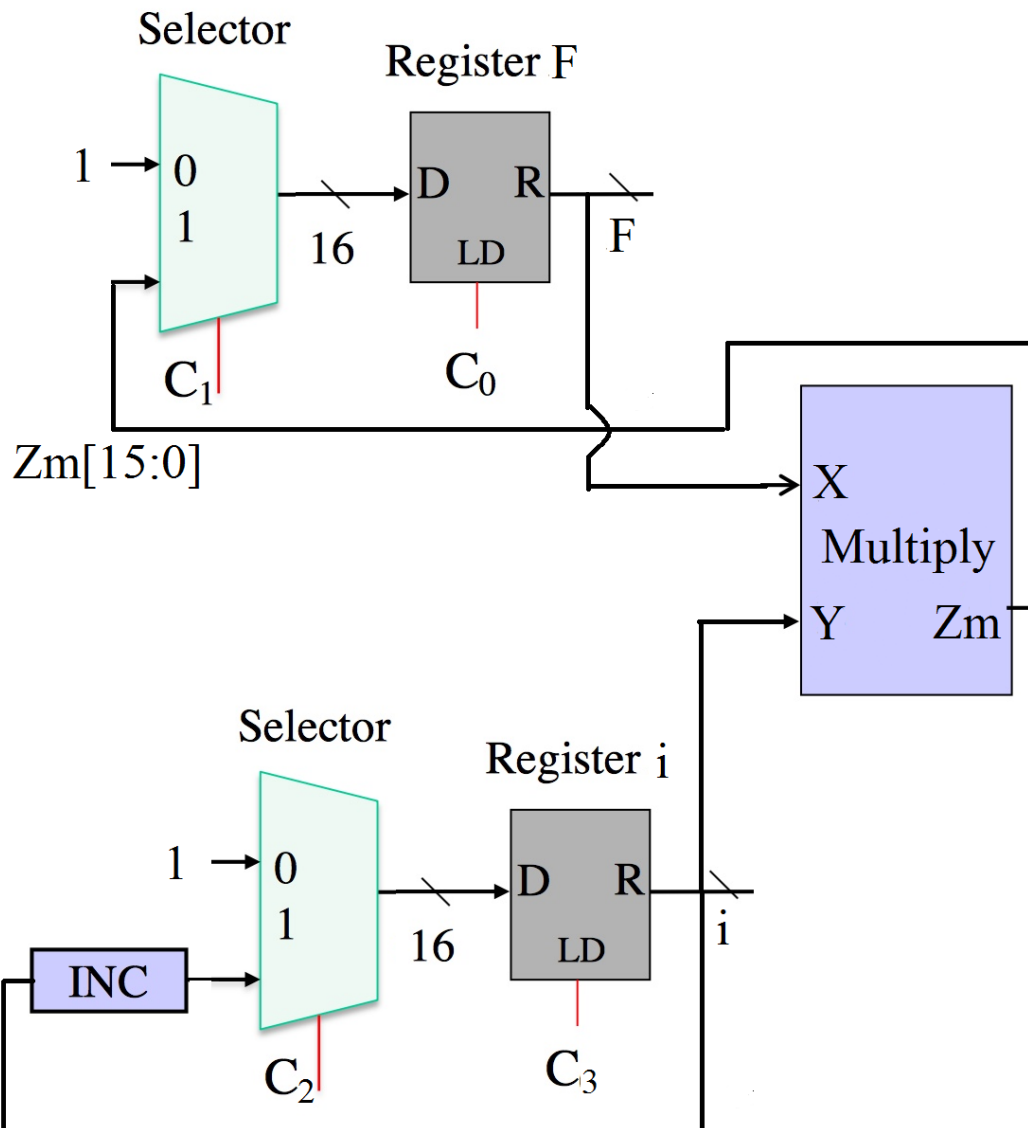
Subtract 0.5 points, if inputs, local objects and outputs are missing or incorrect.

Use the assumptions mentioned by students to evaluate the above conditions.

0 points for no attempt.

4.2. Describe the data subsystem with a schematic diagram. You can treat module *Multiply*(X, Y, Z, start, done) as a black box.

Solution:



Rubrics:

5 points, if the schematic diagram for the data subsystem is correct.

Subtract 1 point for every missing/unnecessary register or selector or control signal or operator like INC or wire.

Subtract 1 point if there is a missing label on selectors, registers or Multiply.

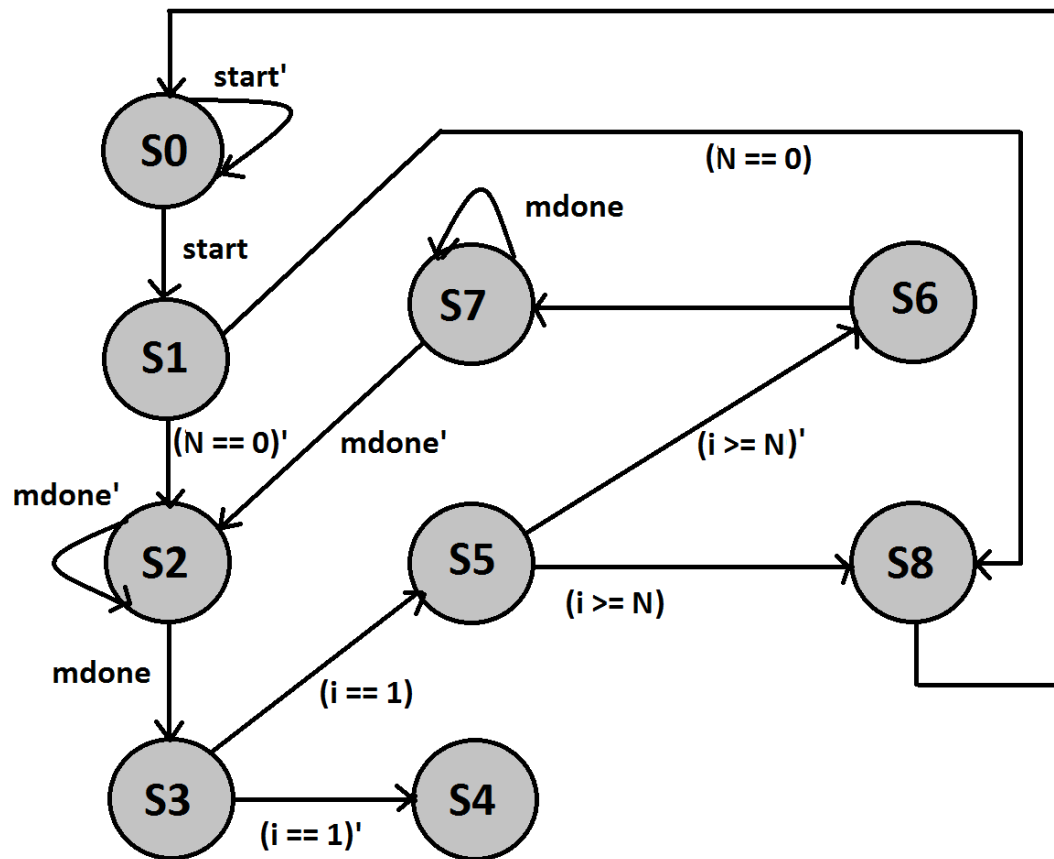
Subtract 2 points if missing the Multiply module.

Please refer to the program of the student when checking above.

4.3. Draw the state diagram of the control subsystem and describe the control signals using a truth table.

Solution:

State Diagram:



Control Signals:

<i>S</i>	<i>C</i> ₀	<i>C</i> ₁	<i>C</i> ₂	<i>C</i> ₃	<i>done</i>	<i>mstart</i>
S0	0	X	X	0	1	X
S1	1	0	0	1	0	X
S2	0	X	X	0	0	0
S3	0	X	X	0	0	0
S4	1	1	X	0	0	0
S5	0	X	X	0	0	0
S6	0	X	1	1	0	0
S7	0	X	X	0	0	1
S8	0	X	X	0	1	0

Rubrics:

9 points, if both the state diagram and the control signals table is correct.
Subtract 1 point for every wrong state in the control signals table.
Subtract 1 point for every missing condition or missing state or incorrect transition in the state diagram.
Subtract 4 points for missing state diagram.
Subtract 5 points for missing control signals table.
Please refer to the program and data subsystem diagram of the student when checking above.