# Department of Electrical & Computer Engineering
# Dalhousie University

## ECED 4260
## IC Design and Fabrication
## Final Examination

| | |
|---|---|
| Instructor: | Y. Ma |
| Exam date: | Dec. 8, 2017 |
| Exam duration: | 180 minutes |
| Aids permitted: | This is a closed book exam. No books or notes are to be consulted. Two double-sided 8.5″x11″ formula sheets are allowed. |

Instructions:
1. Provide your name printed, signature and I.D. number on this page.
2. Verify that this booklet contains 7 pages (including the cover page).
3. Neatly enter your answers in the spaces provided.
4. Use the reverse sides of the pages for rough work.

Student name: _____

Signature: _____

Student ID: _____

Reference Solution

| Question | Worth | Mark | Subject |
|---|---|---|---|
| 1 | 5 | | Multiple-Choice Questions |
| 2 | 6 | | Design of Fast Adder |
| 3 | 6 | | Booth Multiplication |
| 4 | 7 | | Numbers and stuck at model |
| 5 | 11 | | RTL design |
| | | | |
| Total | 35 | | |

**Question 1.**

Multiple-choice questions. Circle **ONLY ONE** that apply. [0.5 pt each]

(1) RTL used in this course stands for

C (a) Resistor Transistor Logic (b) Register Transfer Language
(c) Register Transfer Level (d) Register Transfer Logic

(2) For a random access memory (RAM) device, the time it takes to **read** or **write** a bit of

A memory is independent of the bit location in the RAM, this is
(a) True (b) False

(3) In IC design, sometimes in order to reduce power dissipation, we need to disable

B portions of the circuitry so that the flip-flops in them do not have to switch. What technique can be used to do that?
(a) Two Phase Clocking (b) Gating the Clock (c) Clock Skew (d) Synchronization

C (4) Variable Entered Maps are used to:
(a) Assign pins (b) Replace K-map (c) Reduce the size of K-map (d) Assign outputs

(5) Convert the 2's complement number "11011" into decimal:

D (a) 27 (b) -27 (c) -4 (d) -5

(6) The Booth Re-coded bits of 2's complement number "11011" are:

B (a) 1 0 -1 +1 0 -1 (b) -1 +1 0 -1 (c) +1 -1 0 0 1 (d) -1 +1 0

(7) The Fast Booth equivalent of 2's complement number M "11011" is:

B (a) -2M 0 (b) –M -M (c) –M 0 (d) 2M 0

(8) If we need to define a port in VHDL that can be both read and updated only within

C the module, what port mode should we use?
(a) In (b) Out (c) Buffer (d) Inout

(9) Which of the following CAN **NOT** be detected using "stuck-at fault" model and a test

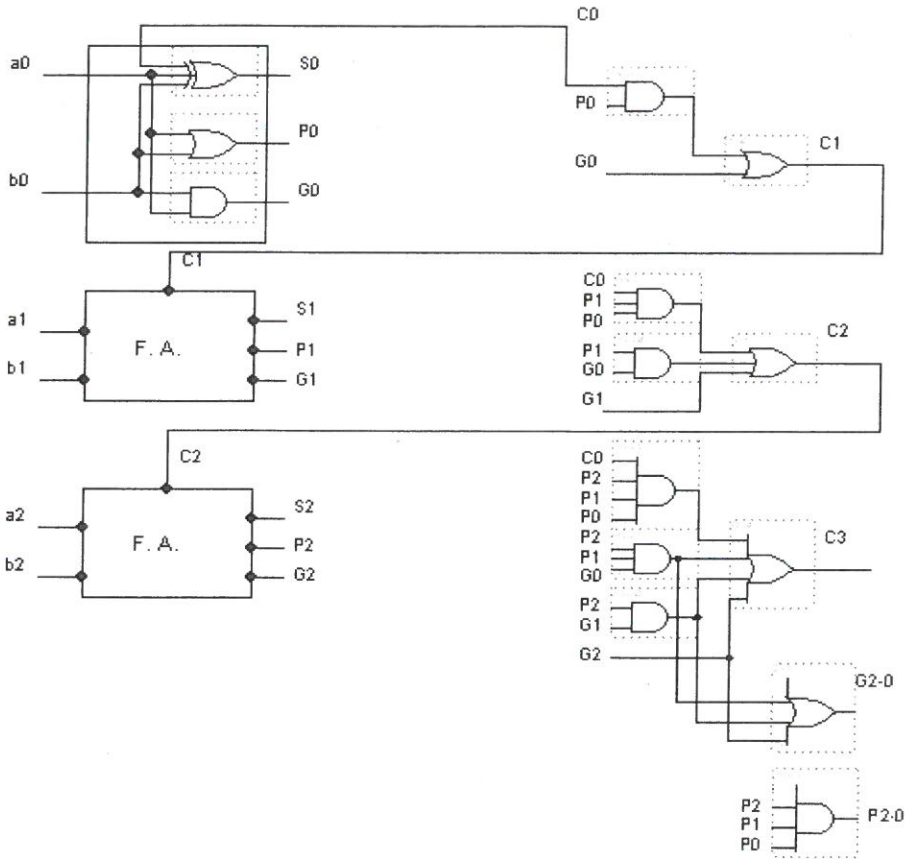
C pattern generator?
(a) Lines tied to ground (b) Manufacture defect (c) Design errors (d) Lines tied to Vcc

(10) Represent -8.25 using IEEE 754 Single Precision Floating Point number:

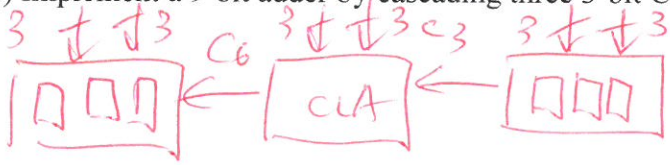
C

| | | | |
|---|---|---|---|
| (a) | 1 | 10000010 | 10000100000000000000000 |
| (b) | 1 | 10000011 | 00010000000000000000000 |
| (c) | 1 | 10000010 | 00001000000000000000000 |
| (d) | 1 | 10000001 | 00000100000000000000000 |

**Question 2.** (6 marks) The whole issue to make a faster adder is to resolve the carry across the entire adder structure more quickly. The schematic below shows a 3-bit carry look-ahead (CLA) adder.




2.1 (3 marks) Implement a 9-bit adder by cascading three 3-bit CLAs.

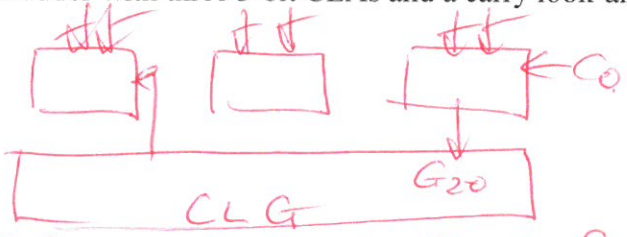



(1)

The critical path is $C_0 \rightarrow C_3 \rightarrow C_6 \rightarrow C_8 \rightarrow S_8$ (1)

The worst case delay = $3 + 2 + 2 + 3 = 10\ \tau_g$. (1)

2.2 (3 marks) Implement a 9-bit adder with three 3-bit CLAs and a carry look-ahead generator (CLG).




(1)

The critical path is $C_0 \rightarrow G_{2\text{-}0} \rightarrow C_6 \rightarrow C_8 \rightarrow S_8$ (1)

The worst case delay = $3 + 2 + 2 + 3 = 10\ \tau_g$ (1)

**Question 3.** (6 marks)

3. Multiply the following pairs of two's complement notation numbers using **Booth** multiplication for the **first** problem and **fast Booth** multiplication for the **second** problem. Show all partial products in the spaces provided. Convert the final result to a Hexadecimal representation.

Booth Multiplication:

| | 1 | 0 | 1 | 0 | 1 |
|---|---|---|---|---|---|
| | | 0 | 1 | 1 | 0 |
| | | +1 | 0 | -1 | 0 |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| 1 0 | 1 | 1 | 1 | 1 | 1 | 0 |

-66

BE

-42 H

Hexadecimal representation:

---

Fast Booth Multiplication:

| | | | | 0 | 1 | 1 | 0 | 1 |
|---|---|---|---|---|---|---|---|---|
| | | | | 1 | 0 | 1 | 1 | 0 |
| | | | | -m | | +2m | | -2m |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |

17E H

-82 H

Hexadecimal representation:

$m^* = 10011$

$2m = 011010$

$2m^* = 100110$

8x16+2

= -130

**Question 4.** (7 marks)

4.1 (3 marks) For the pair of IEEE 754 single precision floating point numbers below, express the given numbers in decimal, and find their **sum** and **product**. Represent the results in IEEE 754 notation.

| | | |
|---|---|---|
| 1 | 10000010 | 01000000000000000000000 |
| 0 | 10000100 | 10010000000000000000000 |

$-1.01 \times 2^3 = -1010_b = -10_d$

$+1.1001 \times 2^5 = 110010_b = 32_h$
$= 50_d$

① Sum: $-0.0101 \times 2^5 + 1.1001 \times 2^5 = 1.01 \times 2^5$

$-0.0101$

$1.01$

| 0 | 10000100 | 01100 - - - - - |
|---|---|---|

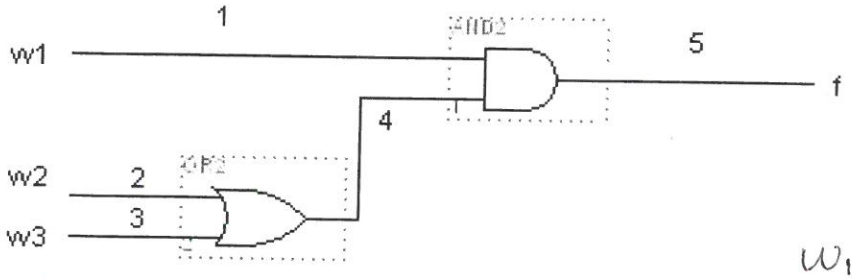
$40_d$

② $1.1001 \times 1.01 = 1.111101$

$-1.111101 \times 2^8$ $\quad -500_d$

| 1 | 10000111 | 1111010 - - - - - |
|---|---|---|

4.2 (4 marks) Using the single-stuck-at model for the following circuit, find out the minimum set of test vectors for every stuck at fault on a labeled wire. Provide the input (w1,w2,w3), correct output (f) and faulty output (f*).




| | $w_1w_2w_3$ | f | f* |
|---|---|---|---|
| 1 s.a.0 = 4 s.a.0 = 5 s.a.0 | (11φ)(1,φ1) | 1 | 0 |
| 2 s.a.1 = 3 s.a.1 = 4 s.a.1 | (100) | 0 | 1 |
| 1 s.a.1 | (01φ)(0φ1) | 0 | 1 |
| 2 s.a.0 | (110) | 1 | 0 |
| 3 s.a.0 | (101) | 1 | 0 |
| 5 s.a.1 | (0φφ)(100) | 0 | 1 |

minimum set {100, 110, 101, 01φ} OR {0φ1}

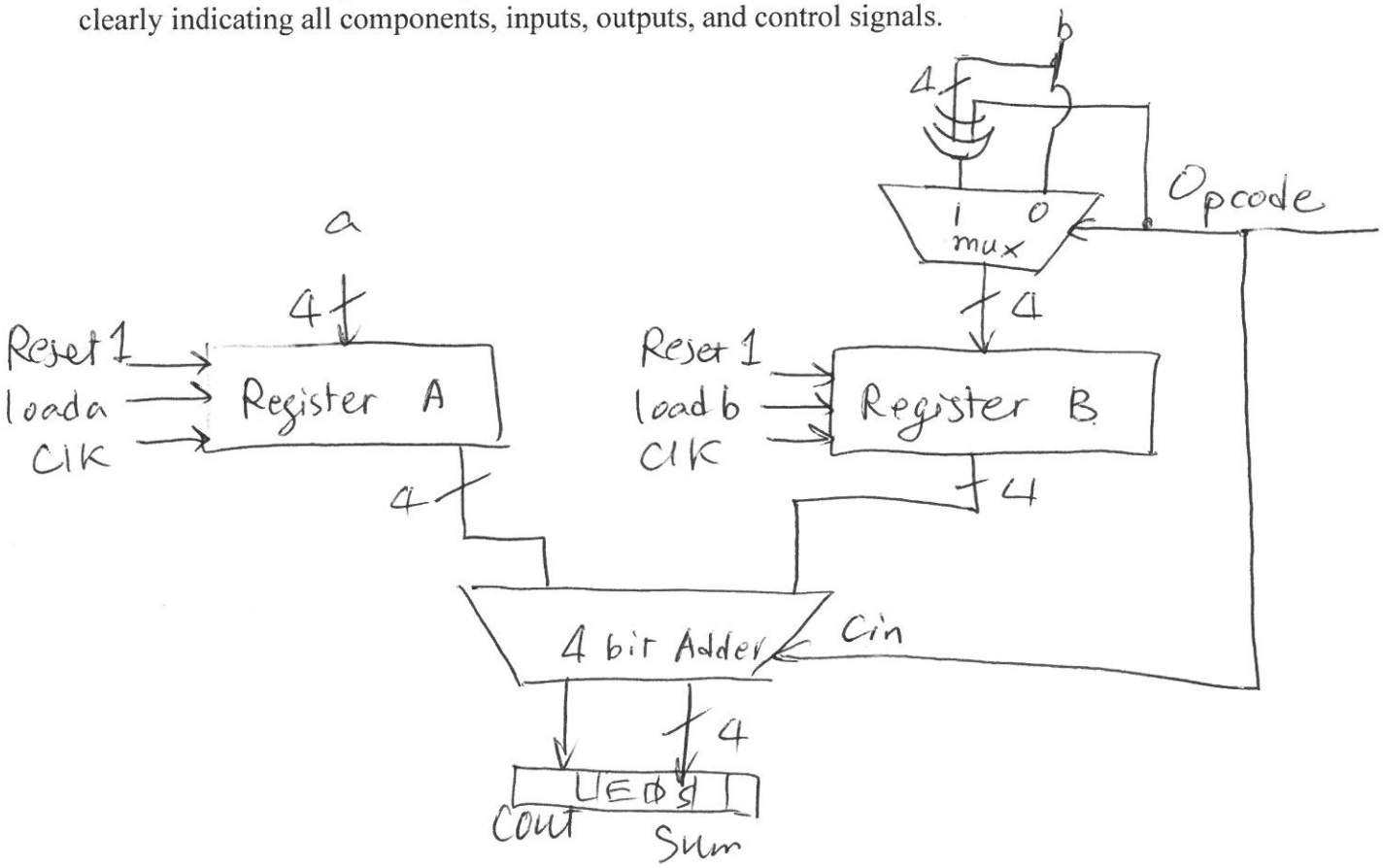
4 vectors

5 { 010, 011, 001 } either one is fine

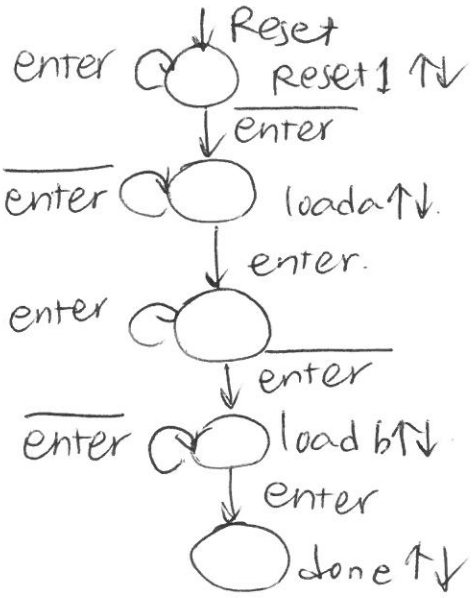
**Question 5.** (11 marks) RTL level design:
Design an Adder/Subtractor. Your design should take two 4-bit unsigned binary numbers, **a** and **b**, and compute their sum or difference based on an operation control input, **Opcode**. When the Opcode is '**0**', the operation required is: **a + b**; when the Opcode is '**1**', the operation required is **a – b**.

When the system is first powered up, the user shall press the **RESET** button to force the system into an initial state. Then the user uses a single switch to set the operation code **opcode**, and a set of four toggle switches to input two numbers **a** and **b**. The system accept number **a** when an **ENTER** button is pressed, then the user uses the same set of four toggle switches to set the second number. The system accept number **b** when the **ENTER** button is pressed again. Results will be displayed by LEDs. An LED labeled **DONE** indicates that the result has been computed.

(5.1) Datapath (5 marks). Datapath should be designed structurally. Available building blocks include: multiplexer, registers, n-bit adder, and XOR gates. Draw your datapath, clearly indicating all components, inputs, outputs, and control signals.



(5.2) Controller (5 marks). Your control path should be designed to be independent of the clock frequency. Draw a detailed state transition diagram indicating all states and outputs. Is your controller a Moore machine or a Mealy machine?




Moore FSM

Example

5.3 (1 mark) Draw a top level symbolic diagram showing the interfaces of your Datapath and Controller; label all signals, inputs, and outputs.