CS 2110 — Quiz 3, Group A
February 2018

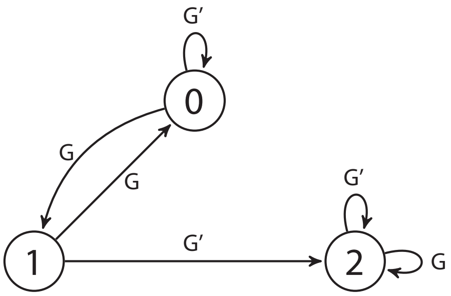
Answer Key
This quiz is worth a total of 100 points .

You are allowed to use one sheet of scrap paper. Feel free to request scrap paper from your Teaching Assistants. Please make sure that all of your answers are contained within the answer boxes **or the fill-in lines.** Do not write your work in the answer boxes, keep all of your work on your scrap paper . You will NOT be given credit for showing work. Having anything except the answer inside the boxes or above the fill-in lines reduces autograder performance and might cause incorrect results. Make sure to write your name, username, and answers legibly. You will not receive credit for illegible answers.

State Machines

1. Fill in the truth table for a binary encoded (reduced) state machine according to the state diagram. 16

   S1, S0 are the two bits representing the current state's number and N1, N0 are the ones representing the next state, where the bits denoted with 0 represent the least-significant bit.



S1	S0	G	N1	N0
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	0
1	0	0	1	0
1	0	1	1	0
1	1	0	x	x
1	1	1	x	x

2. Simplify the following K-map and write the simplified boolean expressions in the box. Only the expressions will be graded, and not the groupings drawn on the map. Write your answer in sum/product notation, for example AB + BC . Only the most efficient (the most simplified) expression will get full credit. 14

AB \ CD	11	10	00	01
11	1	0	1	x
10	0	1	0	0
00	x	1	0	x
01	x	0	1	1

Write your simplified expression in this box:

Solution: BD + BC ′ + B ′CD ′

LC-3 Control Signals

3. The control signals activated for each cycle of a particular LC3 instruction are listed below. Answer the questions based on these signals.

The Fetch Phase **Cycle 1:** GatePC, LD.MAR, PCMUX=PC+1, LD.PC **Cycle 2:** MEM.EN, MDR.SRC.MUX=MEM, LD.MDR **Cycle 3:** GateMDR, LD.IR	**The Decode Phase** You do not need to worry about the decode phase.	**The Execute Phase** **Cycle 1:** ADDR1MUX=BaseR, ADDR2MUX=offset6, MARMUX=ADDER, GateMARMUX, LD.MAR **Cycle 2:** ALUK=PASSA, GateALU, MDR.SRC.MUX=BUS, LD.MDR **Cycle 3:** MEM.EN, MEM.WE

(a) For each of the cycles, tick the boxes for the registers whose values are changed, as well as the RAM box if memory is written to. Tick GPR if the values of any of the General Purpose Registers (R0 through R7) are changed. 30

Fetch Cycle 1: ○ GPR √ **MAR** ○ MDR ○ CC √ **PC** ○ RAM ○ IR
Fetch Cycle 2: ○ GPR ○ MAR √ **MDR** ○ CC ○ PC ○ RAM ○ IR
Fetch Cycle 3: ○ GPR ○ MAR ○ MDR ○ CC ○ PC ○ RAM √ **IR**
Execute Cycle 1: ○ GPR √ **MAR** ○ MDR ○ CC ○ PC ○ RAM ○ IR
Execute Cycle 2: ○ GPR ○ MAR √ **MDR** ○ CC ○ PC ○ RAM ○ IR
Execute Cycle 3: ○ GPR ○ MAR ○ MDR ○ CC ○ PC √ **RAM** ○ IR

(b) What is the instruction being executed here? 10
○ LD ○ LDI ○ LDR ○ LEA ○ ST √ **STR** ○ STI

4. Fill in the control signals for each of the clock cycles of the **execute phase** of the provided instructions. You should **NOT** write the fetch / decode phases (like PC = PC+1). **You do not need to list values for SR / DR / etc.**

Boolean signals:
LD.MAR, LD.MDR, LD.REG, LD.CC, LD.PC, GatePC, GateMDR, GateALU, GateMARMUX, MEM.EN, MEM.WE

Multiplexer signals and possible values:
PCMUX ∈ {PC+1, BUS, ADDER}, ADDR1MUX ∈ {PC, BaseR},
ADDR2MUX ∈ {ZERO, offset6, PCoffset9, PCoffset11}, MARMUX ∈ {ZEXT, ADDER},
MDR.SRC.MUX ∈ {BUS, MEM}, ALUK ∈ {ADD, AND, NOT, PASSA}, SR2MUX ∈ {SR2, SEXT}

(a) LDR 20
Cycle 1: **ADDR1MUX=BaseR, ADDR2MUX=offset6, MARMUX=ADDER, GateMARMUX, LD.M**
Cycle 2: **MEM.EN, MDR.SRC.MUX=MEM, LD.MDR**
Cycle 3: **GateMDR, LD.REG, LD.CC**

(b) BRnzp 10
Cycle 1: **ADDR1MUX=PC, ADDR2MUX=PCoffset9, PCMUX=ADDER, LD.PC**