Lecture Assignment #8

1)
x31A1: LDI R1, label
a)9 bit PC offset = 0 1010 0100
1010 0010 1010 0100 = xA2A4

b)Register Transfer description:
DR <- Mem[Mem[(PC)+SEXT(IR[8:0])]]
The DR is given a value which is stored at “label”

c)

1) ADDR1MUX selects PC
2) ADDR2MUX selects SEXT(IR[8:0])
3) MARMUX selects output of address adder
4) Assert GateMARMUX
5) Assert Ld.MAR
6) MEM.EN/R
   b) Wait for R
7) MDRMUX selects Mem
8) Asset LD.MDR
9) Assert GateMDR
10) Assert LD.MDR
11) DRMUX selects IR[11:9]
12)Mem.EN/R
   b) Wait for R
13) MDRMUX selects Mem
14) Assert LD.MDR
15) Assert Gate.MDR
16) Assert LD.Reg

2)STR R0, R6, x0

a) 0111 000 110 00 0000 = x7180
b) Register Transfer description: MEM[(R6)+SEXT(IR[5:0] <-(R0)]
   Basically R0 is equal to the value inside of R6 added by the 6 bit 2's complement of x0.
c) 1) SR1MUX selects R6
   2)ADDR1MUX selects SR1 bus
   3) ADDR2MUX selects SEXT(IR[5:0])
   4)MARMUX selects output of address adder
   5) Assert GateMARMUX
   6) Assert LD.MAR
   7) ALUK selects "pass through"
   8) Assert Gate.ALU
   9) MDRMUX selects bus
   10) Assert Ld.MDR

3)
x35D0: BRzp label

a)Offset is 1 0001 1110 therefore 0000 011 1 0001 1110 = x071E
b) Register Transfer description: PC<-(PC)+SEXT(IR[8:0])
c) 1) ADDR1MUX selects PC
2)ADDR2MUX selects SEXT(IR[8:0])
3) PCMUX selects output of address adder
4) Assert Ld.PC

5.40) The signal labeled “A” is used to assert the ld.PC

5.41a) Y is the condition code where global bus is tested
b) There is an error where the signal of “X” is being generated

42) D, because MOVE is a simple instruction that can be implemented using “ADD” and “AND”. NAND is also a simple instruction where it can be implemented using “AND” and then “NOT” and SHFL, where contents are shifted left 2 bits and stored into Ri can be implemented using MUL.