EENG284: Introduction to Digital Design

Spring 2021

Homework Assignment #10

Due: 11:59pm, Monday, April 26th, 2021.

1. A 8kx32 RAM is full of integer data. Design a circuit that determines the sum of the integers *between* addresses A and B. The values of A and B are to be read in using a two-line handshake where the circuit is to act as a passive consumer. The sum is to be placed in a 32-bit register S. Turn in; an algorithm the datapath and control unit, the control word table, the memory input equations, and output equations. The control unit is to be implemented using a ones hot encoding.

2. A 256x8 RAM is full of data. Design a circuit that jumps around in memory. The current address should be stored in a register called PC. If the MSB of the fetched word is 1, then the remaining seven bits represent a 7-bit 2's complement number; add these seven bits to the PC. If the MSB of the fetched word is 0 then just increment the PC. Repeat this process forever. Turn in; an algorithm the datapath and control unit, the control word table, the memory input equations, and output equations. The control unit is to be implemented using a ones hot encoding.
   The desired behavior of the circuit is illustrated in Figure 1. In this figure if PC=0 then the word at that address (3F) has a MSB of 0 so the PC is incrementde to 1. The word at address 1 is fetched (BC) and has an MSB of 1 so the least significant seven bits of BC are added to the PC, making its new value 3D. repeating this process sees the PC goto address 21, 22, 21, 22 into a never ending cycle. Make sure the solution identifies how to add the least significant seven bits to an 8-bit PC.

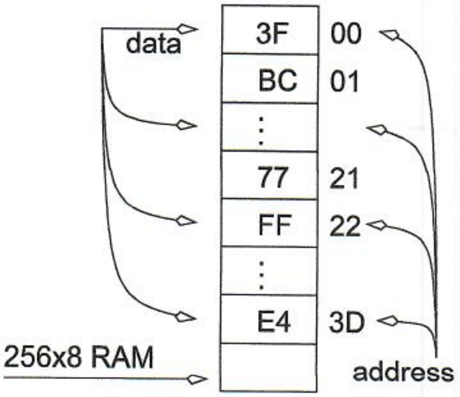


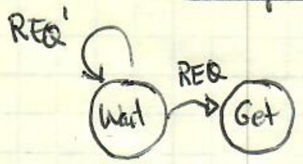
Figure 1: A 256x8 RAM loaded with some data.

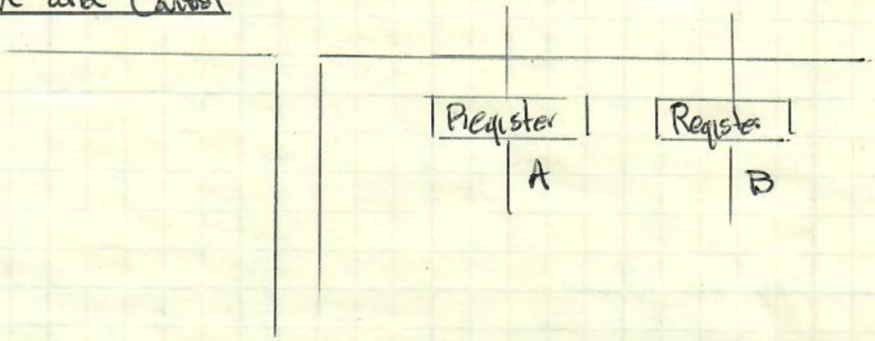
3. Design a circuit that reads successive words from a 1kx12 RAM and updates a 12-bit register called **ACC** based on the upper two bits of the memory word. The address of the current memory word should be contained in a register called PC (Program Counter). Since the words read from the RAM will tell us what operation to perform on the ACC, the memory word will be stored in a register called IR (Instruction Register). If the upper two bits of IR are:
   (a) 00 then add the lower 10 bits of the IR to ACC. Pad the upper two bits of the IR with 0's before adding to the ACC.
   (b) 01 then store the ACC to the address specified by the lower 10 bits of the IR.
   (c) 10 then load the ACC from from the address specified by the lower 10 bits of the IR.
   (d) 11 then clear the value of ACC to 0.

#1 Algorithm for(;;) {

```
Get both A & B @ same time
    with 1 handshake
sum = 0;
loop
    sum += RAM[i];
```

Datapath and Control





Control Word Table

	Register A	Register B	ACK
	0 hold 1 load	0 hold 1 load	
Wait			
Get			

#2 How hardware process RAM contents in example

PC	RAM[PC] MBR	MBR binary	MBR_7	Next PC	
0	3F	0011 1111	0	PC+1 0 + 1 = 1	
1	BC	1011 1100	1	$PC+MBR_{6-0}$ 0000 0001 + 0011 1100 0011 1101 = 3D	Sign extend
3D	E4	1110 0100	1	$PC+MBR_{6-0}$ 0011 1101 + 1110 0100 0010 0001 = 21	sign extend
21	77	0111 0111	0	PC + 1 = 22	
22	FF	1111 1111	1	$PC+MBR_{6-0}$ 0010 0010 + 1111 1111 0010 0001 = 21	sign extend

- Use MBR_7 in algorithm
- "Pluck" off MSB of MBR and run straight into control unit
- Sign Extend MBR_{6-0}

#3

IR contents		Name	Operation
00	immediate	Add immediate	ACC += immediate
01	address	Store ACC	RAM [address] = ACC
10	address	load ACC	ACC = RAM [address]
11		Clear ACC	ACC = 0

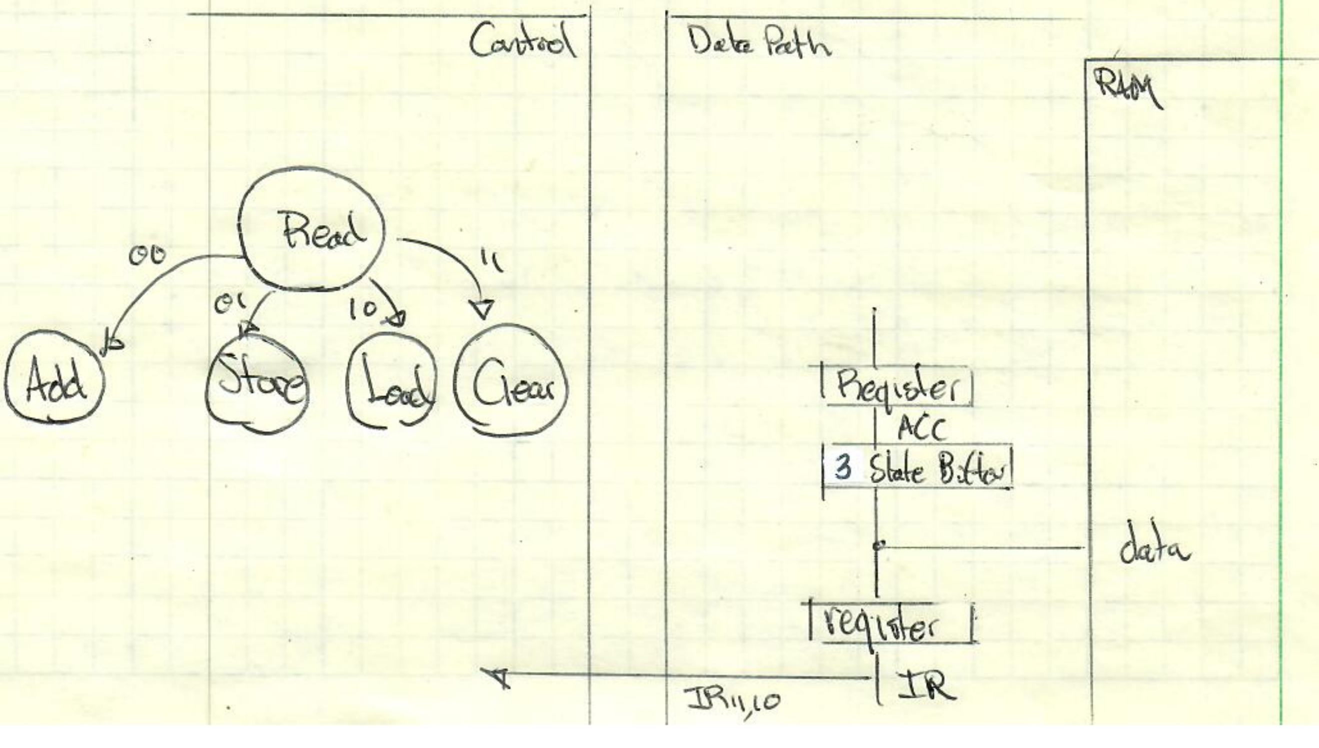
|← 12 bits →|

Algorithm

```
for (;;) {
    IR = RAM[PC]
    if ( $IR_{11,10}$ = = 00 )
                RAM[$IR_{9-0}$] = ACC
    else if ( $IR_{11,10}$ = 01 )
}
```

Datapath and Control

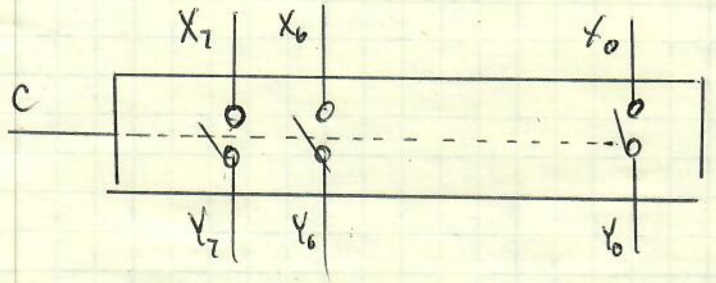
* RAM is an left & right hand side of assignment statement



EENG 28A | HW#10 Review

Since the RAM data lines are driven by 2 different sources we need a way to arbitrate which is asserting data.

Solution: 3-state buffer (page 87, 88)



X	C	Y
X	0	Z
X	1	X

when C=0 Y is not connected to any driven source this allows the bus connected to Y to be driven by another source (RAM) in read

When C=1 Y is driven by X In this configuration the RAM will be written