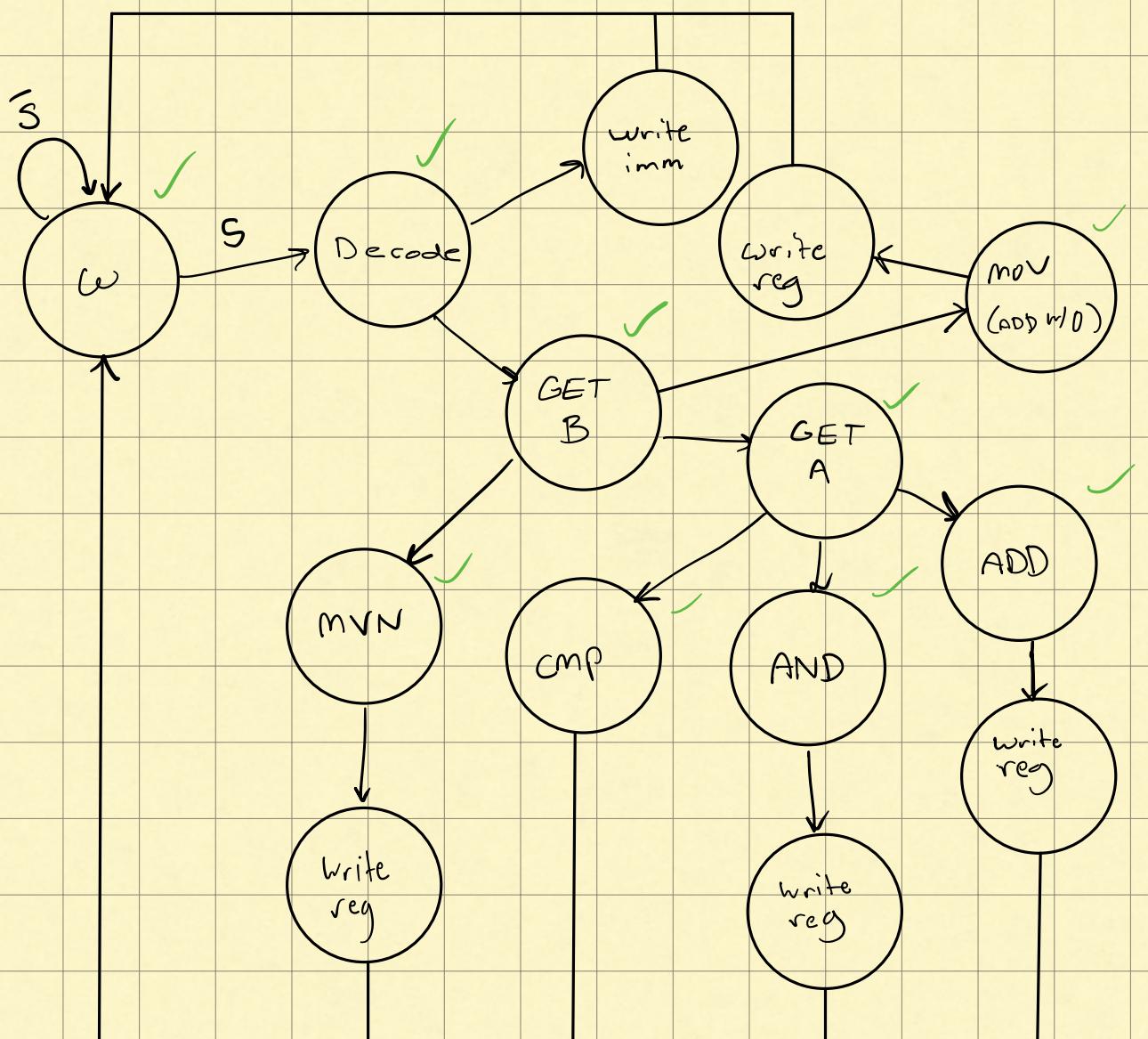


15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Op code				op				Rn				im8/Rd			
110	-	mov	10	-	MOV_im			Rn		im8		im8		im8	
					00	-	MOV	000		Rd		sh		im8	
101	-	ALU	00	-	ADD			Rn		Rd				Rm	
					01	-	CMP	Rn		000					
					10	-	AND	Rn		Rd					
					11	-	MVN	000		Rd					



NOTE: Don't forget to sign extend im8

Vsel mux encoding

0001 → sximm8

0010 → C/datapath_out

0100 → {8'b0, PC}

1000 → mdata

nSel mux encoding

001 → Rn

010 → Rd

100 → Rm

6 5 4 3 2 1 0

0 = 1 0 0 0 0 0 0

1 1 = 1 1 1 1 0 0 1

2 = 0 1 0 0 1 0 0

3 = 0 1 1 0 0 0 0

4 = 0 0 1 1 0 0 1

5 = 0 0 1 0 0 1 0

5 6 = 0 0 0 0 0 1 0

4 7 = 1 1 1 1 0 0 0

8 = 0 0 0 0 0 0 0

9 = 0 0 1 1 0 0 0

A = 0 0 0 1 0 0 0

b = 0 0 0 0 0 1 1

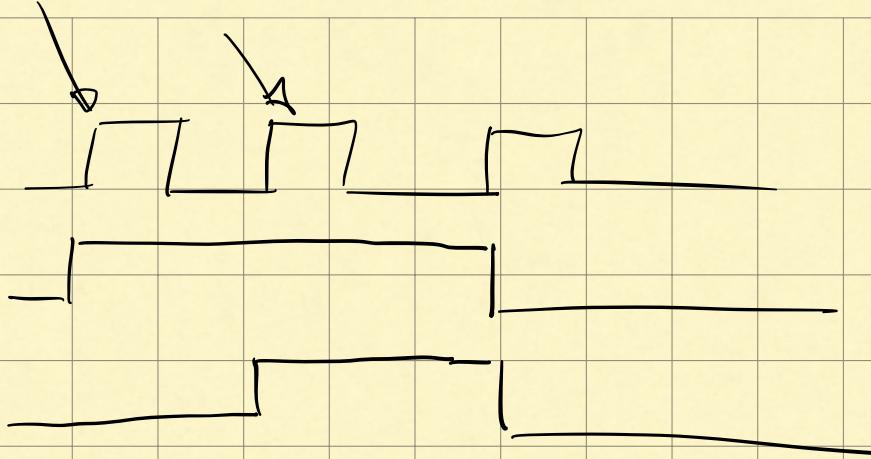
c = 1 0 0 0 1 1 0

d = 0 1 0 0 0 0 1

E = 0 0 0 0 1 1 0

F = 0 0 0 1 1 1 0

clock



load

(a)

ADD R2, R2, R0

1 1 0 1 0 0 0 0
 | |

1 1 0 1 0 0 0 1
 | 0 1

1 0 1 0 0 0 0 1
0 0 1 0 0 0 0 0

$$\begin{array}{r} 1 1 1 \\ 1 1 1 \\ \hline 0 0 0 \end{array}$$

Nov R0, #127

11010000 111111

Test for Errors in Shift : (0) LSL (LSB = 0)
(1) LSR (MSB = 0)
(11) LSR (MSB = B[5]). ~

Test for Errors in Status Bits :

- show (-) ans .
- show overflow "
- show ans = 0.

Test for Errors in AND

- AND w/ LSL ✓
- AND w/ LSR (MSB = 0)
- AND w/ 2SR

Test for Errors in MVN

- MVN w/ LSL ✓
- MVN w/ LSR (MSB = 0)
- MVN w/ 2SR

Mov R0, #5 → 5

MUN RI, RD → -6 (FFFA)

ADD R2, R0, R1, LSR #1 → 5 + $\frac{-6}{2} = 5 - 3 = 2$

$$\text{AND } R_3, R_0, R_2 \rightarrow 5 \& 2 = 0$$

ADD R3, R3, R0

CMP R0, R3

May 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 1

1 0 1 1 0 0 0 0 0 1 0 0 0 0 0

101000001011001

1 0 1 1 0 0 0 0 0 1 1 0 0 0 0 1 0

1 0 1 0 0 0 1 1 1 0 1 1 0 0 0 0 0 0

1 0 1 0 1 0 0 0 0 0 0 0 0 0 1 1