

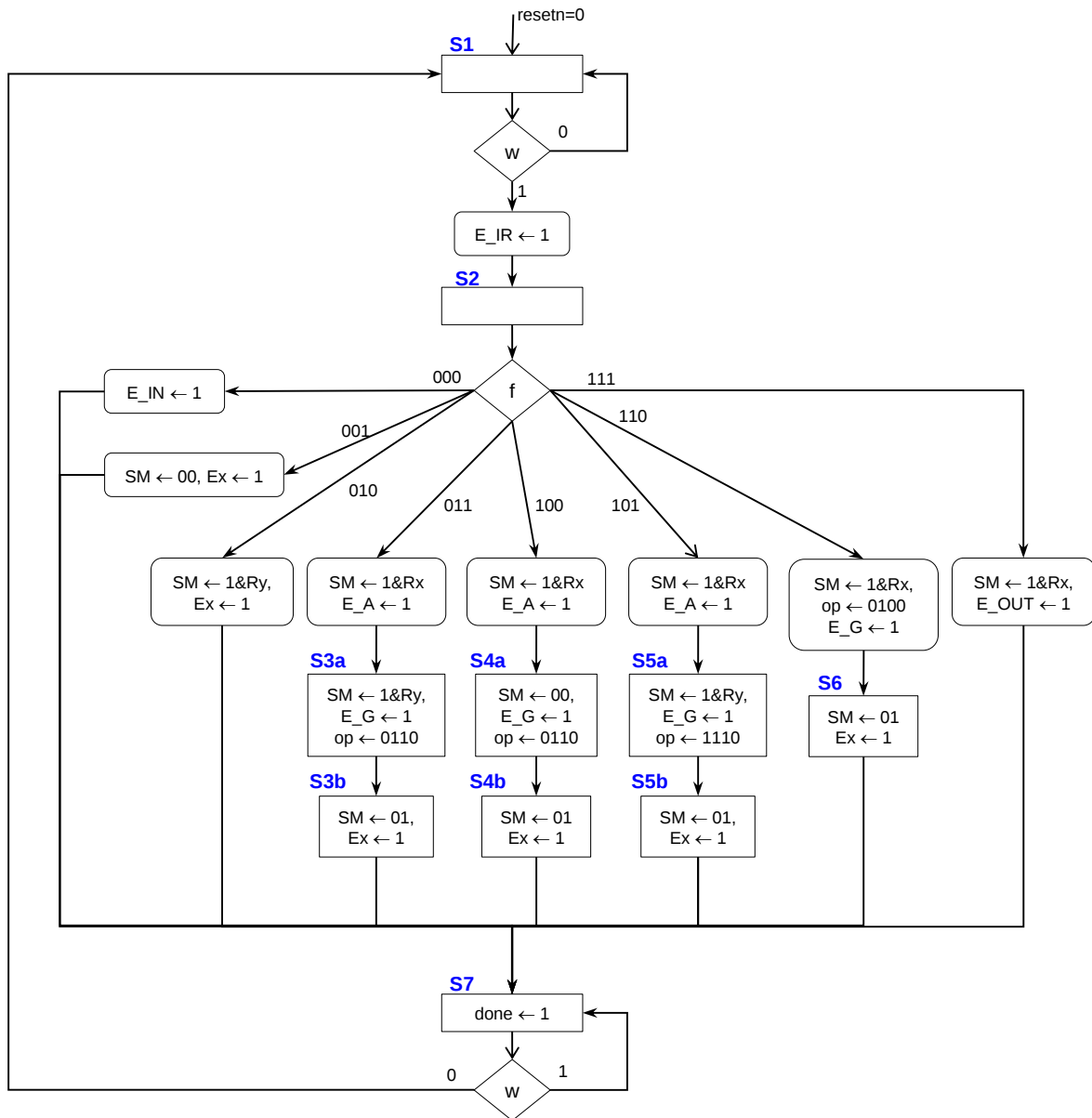
(Due date : **002**: March 15th, **003**: March 16th

- ✓ Implement a Digital System: Control Unit and Datapath Unit.
- ✓ Describe Algorithmic State Machine (ASM) charts in VHDL.
- ✓ Learn the basics of Microprocessor Design.

Instruction Set:

F	Operation	Function
000	load IN	$IN \leftarrow \text{Switches}$
001	load Rx, IN	$Rx \leftarrow IN$
010	copy Rx, Ry	$Rx \leftarrow Ry$
011	add Rx, Ry	$Rx \leftarrow Rx + Ry$
100	add Rx, IN	$Rx \leftarrow Rx + IN$
101	xor Rx, Ry	$Rx \leftarrow Rx \text{ XOR } Ry$
110	inc Rx	$Rx \leftarrow Rx + 1$
111	load OUT, Rx	$OUT \leftarrow Rx$

Control Circuit: FSM



Arithmetic Logic Unit (ALU):

sel	Operation	Function	Unit
0000	y <= A	Transfer 'A'	Arithmetic
0001	y <= A + 1	Increment 'A'	
0010	y <= A - 1	Decrement 'A'	
0011	y <= B	Transfer 'B'	
0100	y <= B + 1	Increment 'B'	
0101	y <= B - 1	Decrement 'B'	
0110	y <= A + B	Add 'A' and 'B'	
0111	y <= A - B	Subtract 'B' from 'A'	
1000	y <= not A	Complement 'A'	Logic
1001	y <= not B	Complement 'B'	
1010	y <= A AND B	AND	
1011	y <= A OR B	OR	
1100	y <= A NAND B	NAND	
1101	y <= A NOR B	NOR	
1110	y <= A XOR B	XOR	
1111	y <= A XNOR B	XNOR	

VIVADO DESIGN FLOW FOR FPGAs:

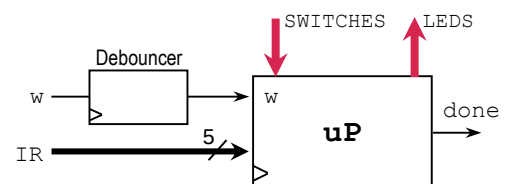
- ✓ Create a new Vivado Project. Select the **XC7A100T-1CSG324 Artix-7 FPGA** device.
- ✓ Write the VHDL code for the given circuit.
- ✓ With a 100 MHz input clock, write the VHDL testbench to test the following Assembly program:

```
load IN;      IN ← 0101 (SWs = 0101)
load R1, IN;  R1 ← 0101
copy R0, R1;  R0 ← 0101, R1 ← 0101
inc R1;       R1 ← 0110
xor R0, R1;   R0 ← 0101 xor 0110 = 0011
add R0, R1;   R0 ← 0011 + 0110 = 1001
load OUT, R0; OUT ← 1001
```

- ✓ Perform Functional Simulation and Timing Simulation of your design. **Demonstrate this to your TA.**
- ✓ I/O Assignment: Create the XDC file.
Nexys-4 DDR Board: Use SW0 to SW4 for the *IR* input, SW5 to SW8 for the input to the IN register, BTNC for the *w* input, CLK100MHZ for the input *clock*, CPU_RESET push-button for *resetsn*, a LED for *done*, and four LEDs for the output of register OUT.

SECOND ACTIVITY: TESTING (30/100)

- In order to properly test the microprocessor, we need to avoid mechanical bouncing on the pushbutton for input *w*. Connect the debouncer circuit (use the given files: *mydebouncer.vhd*, *my_genpulse_sclr.vhd*) on the input *w*.
- Note that you do not need to simulate the circuit that includes the debouncer.



- ✓ Generate and download the bitstream on the FPGA and test the Assembly Program. **Demonstrate this to your TA.**
- Submit (as a .zip file) all the generated files: VHDL code files, VHDL testbench (for the uP block), and XDC file to Moodle (an assignment will be created). DO NOT submit the whole Vivado Project.
- **You can work in teams of up to two (2) students. Each member of the team must demonstrate Functional Simulation, Timing Simulation and Testing using a different Assembly Program (at least 6 different instructions). Also, every team member must make a Moodle submission with a different testbench.**

TA signature: _____

Date: _____