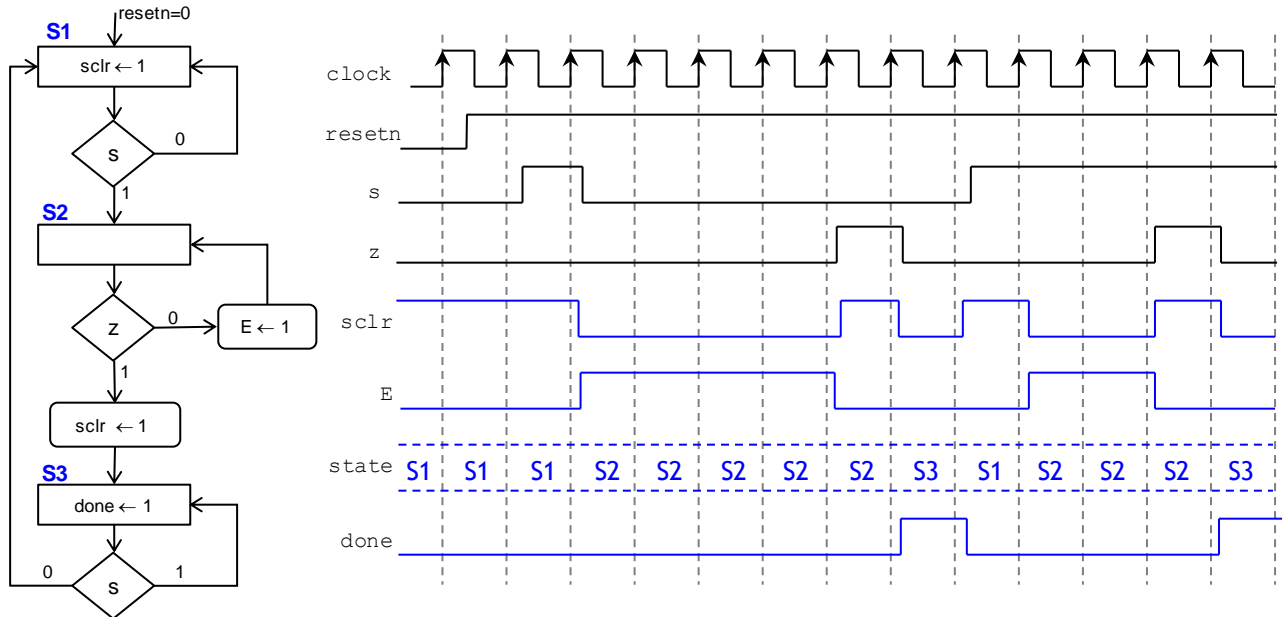


# Solutions - Quiz 4 (Online Section)

(Due Date: December 6th by 11:00 am)

## PROBLEM 1 (30 PTS)

- Complete the timing diagram of the following FSM (represented in ASM form):



## PROBLEM 2 (20 PTS)

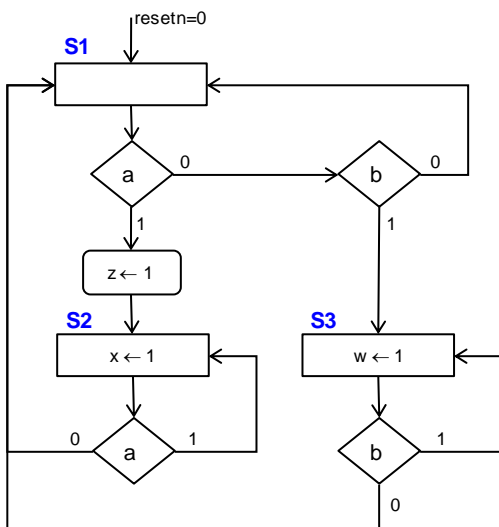
- Provide the state diagram (in ASM form) of the FSM whose VHDL description is listed below:

```
library ieee;
use ieee.std_logic_1164.all;

entity circ is
port ( clk, rstn: in std_logic;
      a, b: in std_logic;
      x,w,z: out std_logic);
end circ;
```

```
architecture behavioral of circ is
type state is (S1, S2, S3);
signal y: state;
begin
Transitions: process (rstn, clk, a, b)
begin
if rstn = '0' then y <= S1;
elsif (clk'event and clk = '1') then
case y is
when S1 =>
if a = '1' then y <= S2;
else
if b = '1' then y <= S3;
else y <= S1; end if;
end if;
when S2 =>
if a = '1' then y <= S2;
else y <= S1; end if;
when S3 =>
if b = '1' then y <= S3;
else y <= S1; end if;
end case;
end if;
end process;

Outputs: process (y,a)
begin
x <= '0'; w <= '0'; z <= '0';
case y is
when S1 => if a = '1' then z <= '1'; end if;
when S2 => x <= '1';
when S3 => w <= '1';
end case;
end process;
end behavioral;
```



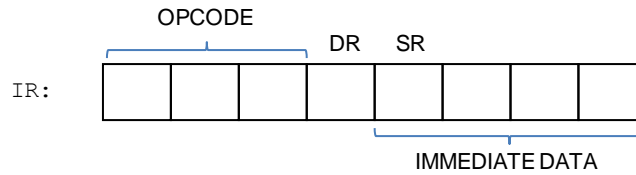
**PROBLEM 3 (30 PTS)**

▪ **Basic Processor:**

*Available Registers:* R0 (register 0, 4 bits), R1 (register 1, 4 bits), PC (program counter, 4 bits),  
OUT (output register, 4 bits)  
IR (instruction register, 8 bits)

*Instruction Memory:* Stores up to 16 8-bit instructions.

*Instruction Set:* Instructions are specified on the Instruction Register (IR):



DR=0 ⇒ R0 is the destination register, DR=1 ⇒ R1 is the destination register.

SR=0 ⇒ R0 is the source register, SR=1 ⇒ R1 is the source register.

OPCODE (IR[7..5])	Instruction	Operation
000	MOV DR, SR	DR ← SR
001	LOADI DR, DATA	DR ← DATA, DATA = IR[3..0]
010	ADD DR, SR	DR ← DR + SR
011	ADDI DR, DATA	DR ← DR + DATA, DATA = IR[3..0]
100	SR0 DR, SR	DR ← 0&SR[3..1]
101	IN DR	DR ← IN
110	OUT DR	OUT ← DR
111	JNZ DR, ADDRESS	PC ← PC + 1 if DR=0 PC ← IR[3..0] if DR≠0 * ADDRESS = IR[3..0]

- Write an assembly program for a counter from 1 to 12: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 1, 2, 3, .... The count must be shown on the output register (OUT). Note that you can only have up to 16 instructions.

```

start: loadi R0,1
      loadi R1,4

loop:  out R0 → OUT: shows the count
      addi R0,1
      addi R1,1
      jnz R1, loop
      loadi R0,1
      jnz R0, start
    
```

**PROBLEM 4 (20 PTS)**

- Sequence detector (with overlap): Draw the state diagram (in ASM form) of a circuit that detects the following sequence: 0110. The detector must assert an output z= '1' when the sequence is detected.

