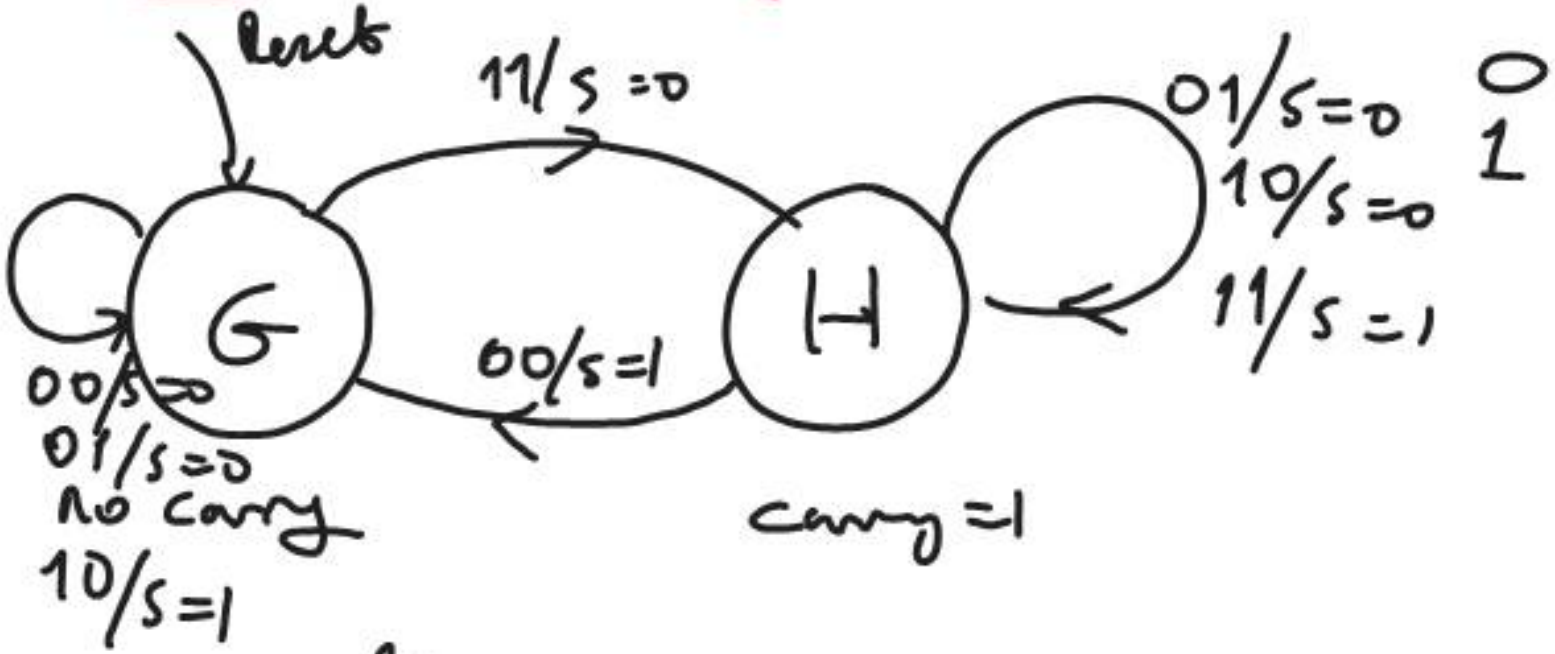
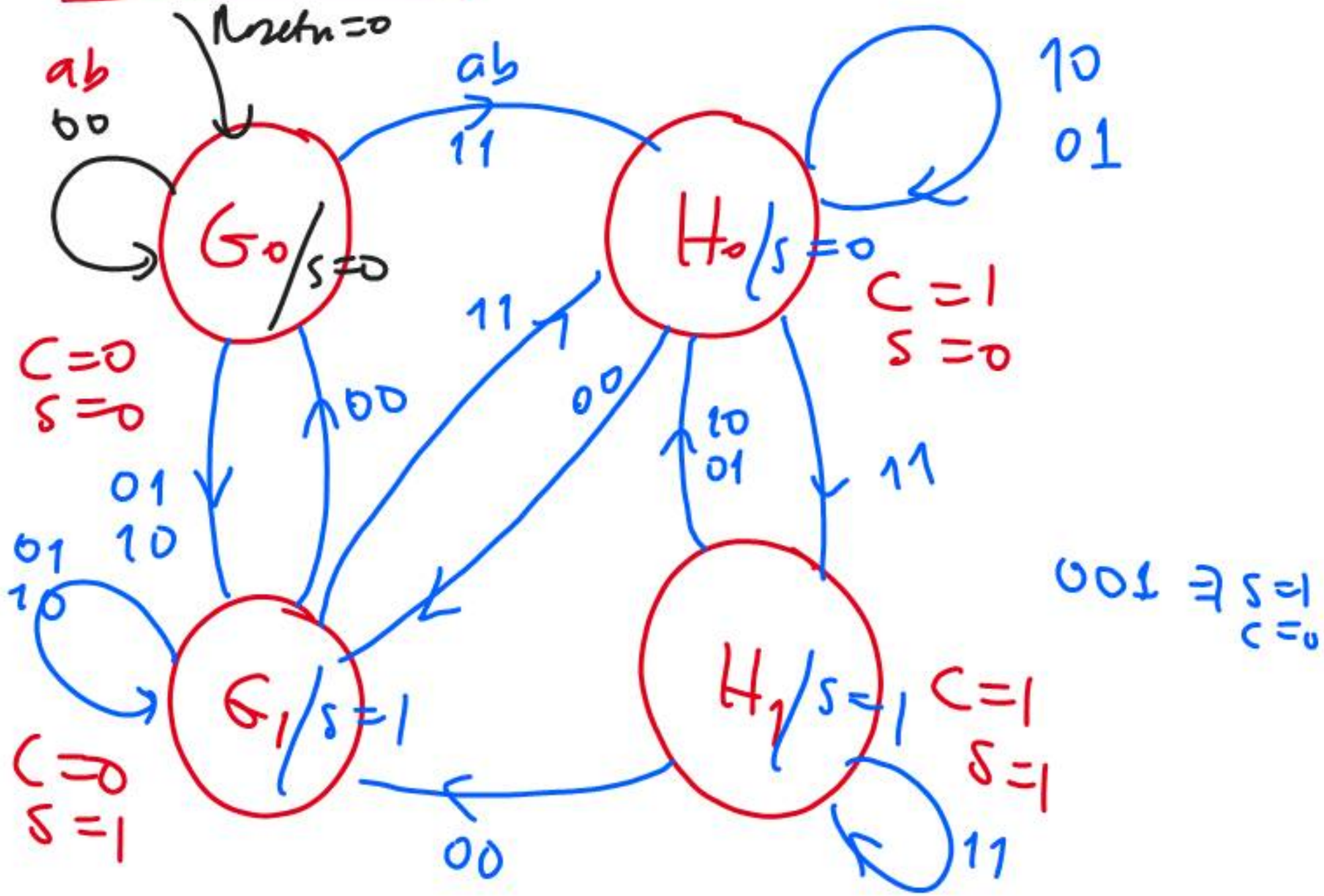
Full Adder (1-bit)

16.05.2011



Mealy-type machine

Full Adder (1 bit) moore machine



State Table

Present State	Next State $q_1 q_0$				Output
	00	01	10	11	S
G0	G0	G1	G1	H0	0
G1	G0	G1	G1	H0	1
H0	G1	H0	H0	H1	0
H1	G1	H0	H0	H1	1

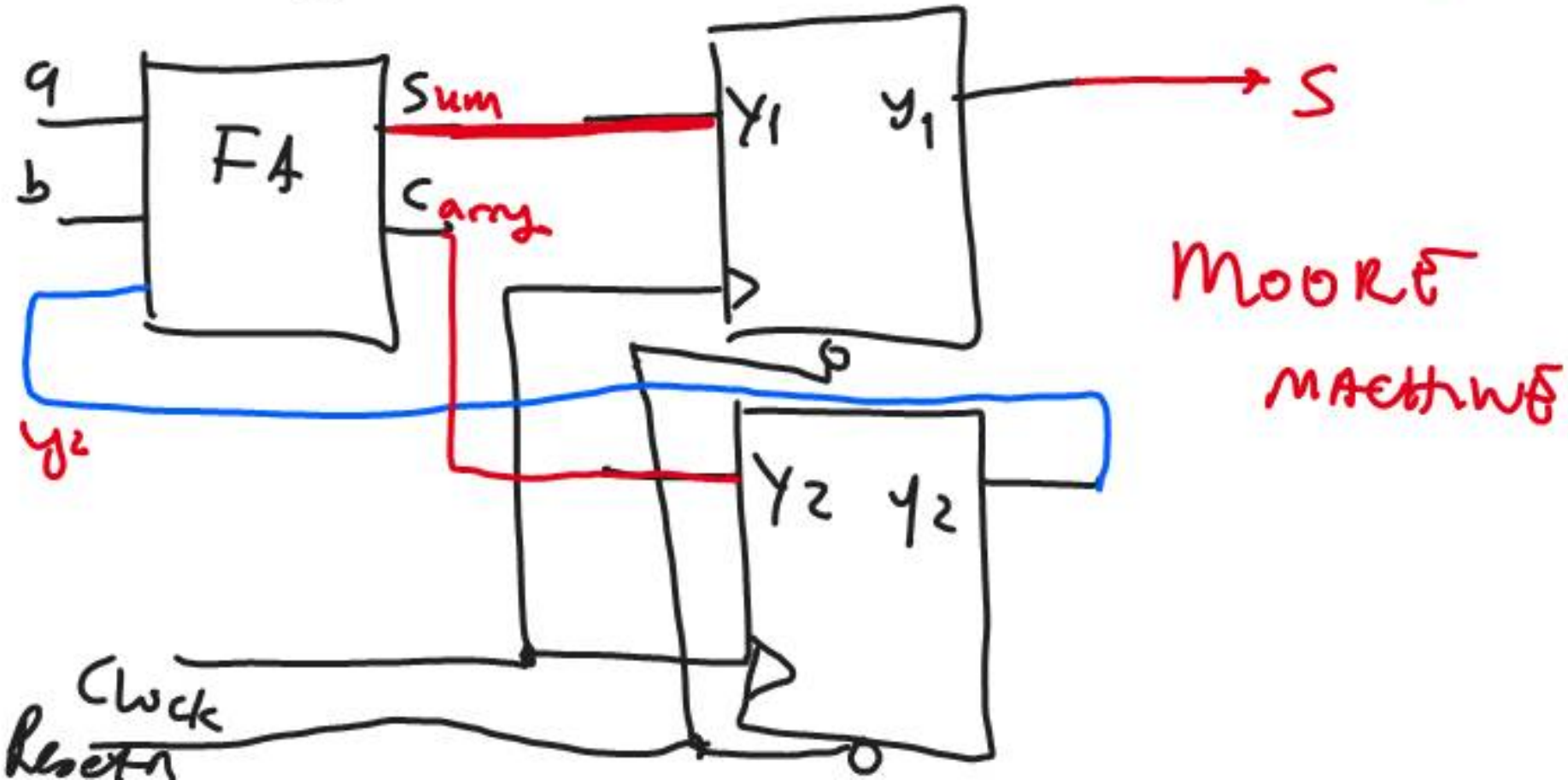
State Assigned Table

$y_2 y_1$	Next State $q_1 q_0$				Output
	00	01	10	11	S
	$Y_2 Y_1$				
00	00	01	01	10	0
01	00	01	01	10	1
10	01	10	10	11	0
11	01	10	10	11	1

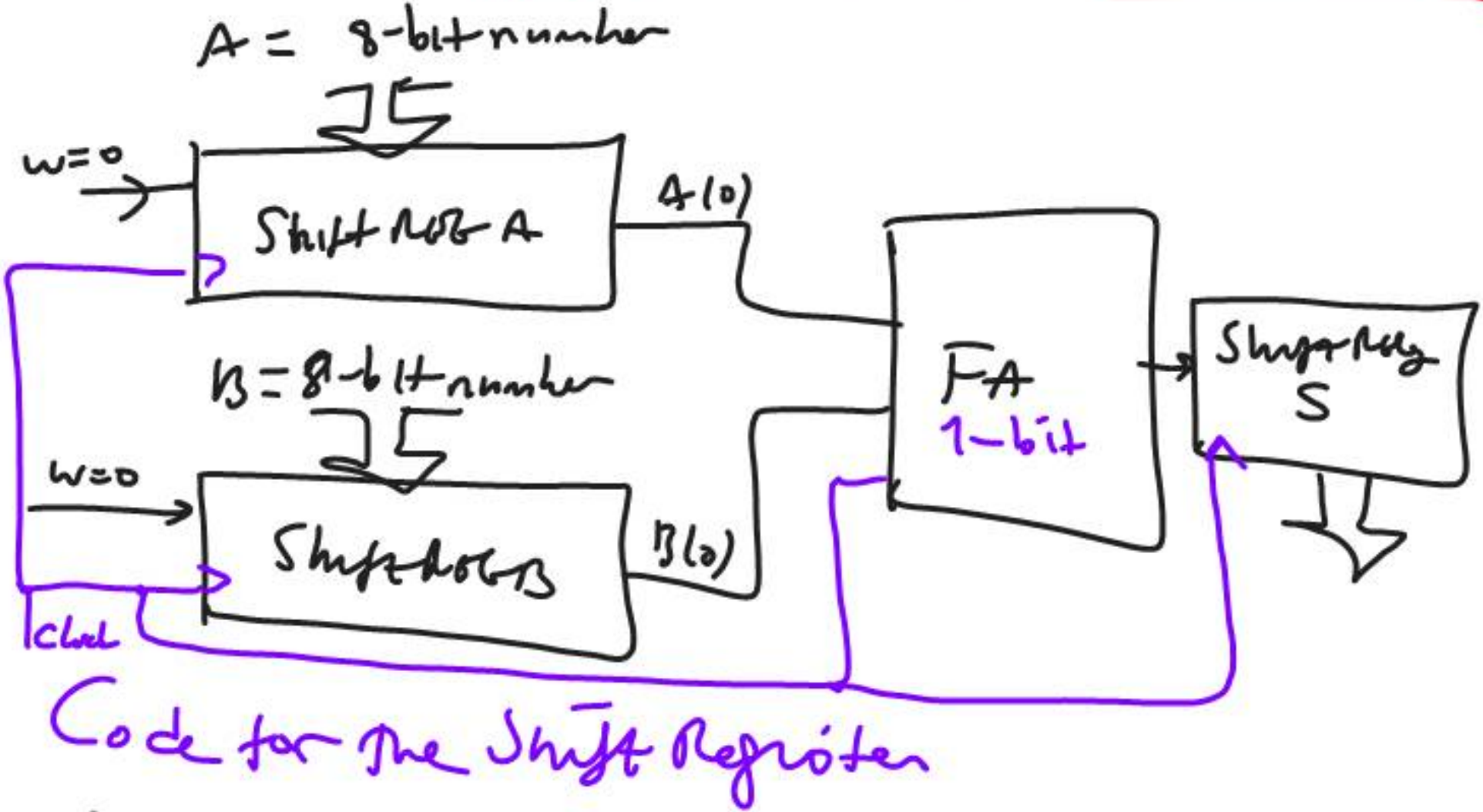
$Y_1 = a \oplus b \oplus y_2$ Sum

$Y2 = ab + ay_2 + by_2$ Majority function (Carry)

$S = y_1$



VHDL code for a 1-bit FA MEALY TYPE



Code for the Shift Register

```
ENTITY ShiftReg IS
GENERIC ( N : INTEGER := 8 );
PORT ( R : IN STD_LOGIC_VECTOR( N-1 DOWNTO 0 );
```

```
L, E, W : IN STD_LOGIC;
Clock   : IN STD_LOGIC;
Q       : BUFFER STD_LOGIC_VECTOR(N-1 DOWNTO 0));
END Shiftreg;

ARCHITECTURE Behavior OF Shiftreg IS
BEGIN
 PROCESS
 BEGIN
 WAIT UNTIL Clock'EVENT AND Clock = '1';
 IF E = '1' THEN
   IF L = '1' THEN
     Q <= R --> parallel load of 8-bit
   ELSE
```

Genbits : For i IN 0 TO N-2 LOOP

Q(i) ⇐ Q(i+1)

END LOOP;

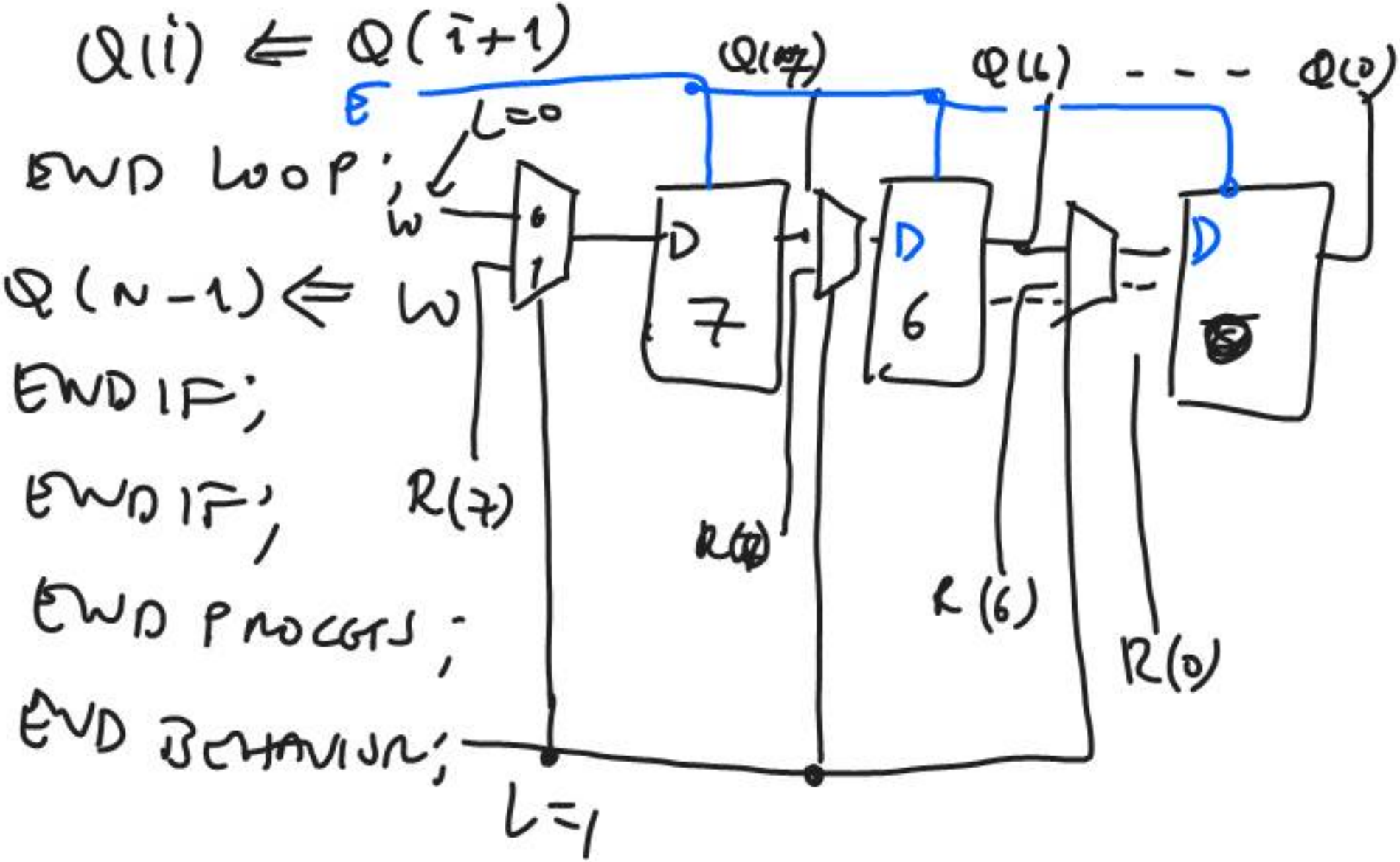
Q(N-1) ⇐ W

END IF;

END IF;

END PROCESS;

END BEHAVIOR;



ADDER (WITH REGISTER) Code:

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
ENTITY SERIAD IS
GENERIC(length: INTEGER:= 8);
PORT( Clock : IN STD_LOGIC;
      Reset : IN STD_LOGIC;
      A, B : IN STD_LOGIC_VECTOR (length-1 DOWNTO 0);
      Sum : BUFFER STD_LOGIC_VECTOR( length-1
                                     DOWNTO 0));
END SERIAD;
```

```
ARCHITECTURE Behavior OF SOMAR IS

COMPONENT ShiftReg
GENERIC( N: INTEGER:=8);
PORT( R:IN STD_LOGIC_VECTOR (N-1 DOWNTO 0);
      L,E,W:IN STD_LOGIC;
      Clock: IN STD_LOGIC;
      Q : BUFFER STD_LOGIC_VECTOR(N-1 DOWNTO 0));
END component;
```