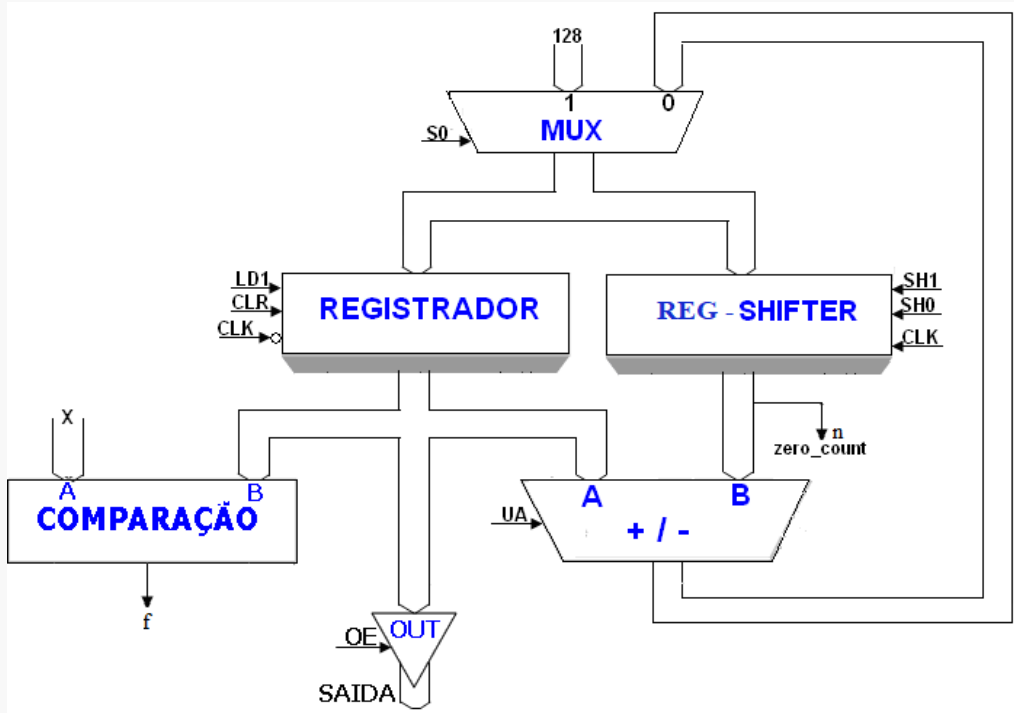
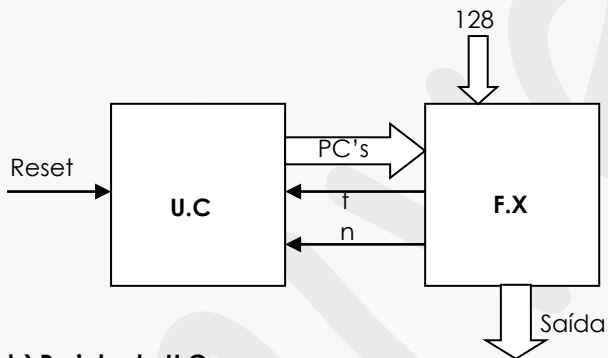


EXEMPLOS RTL

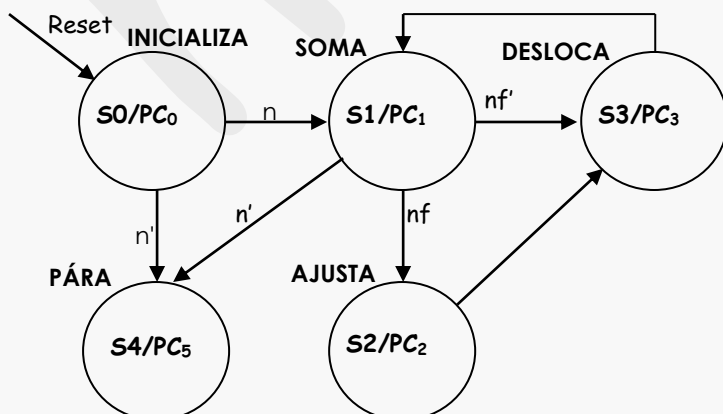
Exemplo: Determinar o valor de x , sendo $x \geq 0$, e x é um valor dado.



a) Representação esquemática do sistema digital



b) Projeto da U.C.



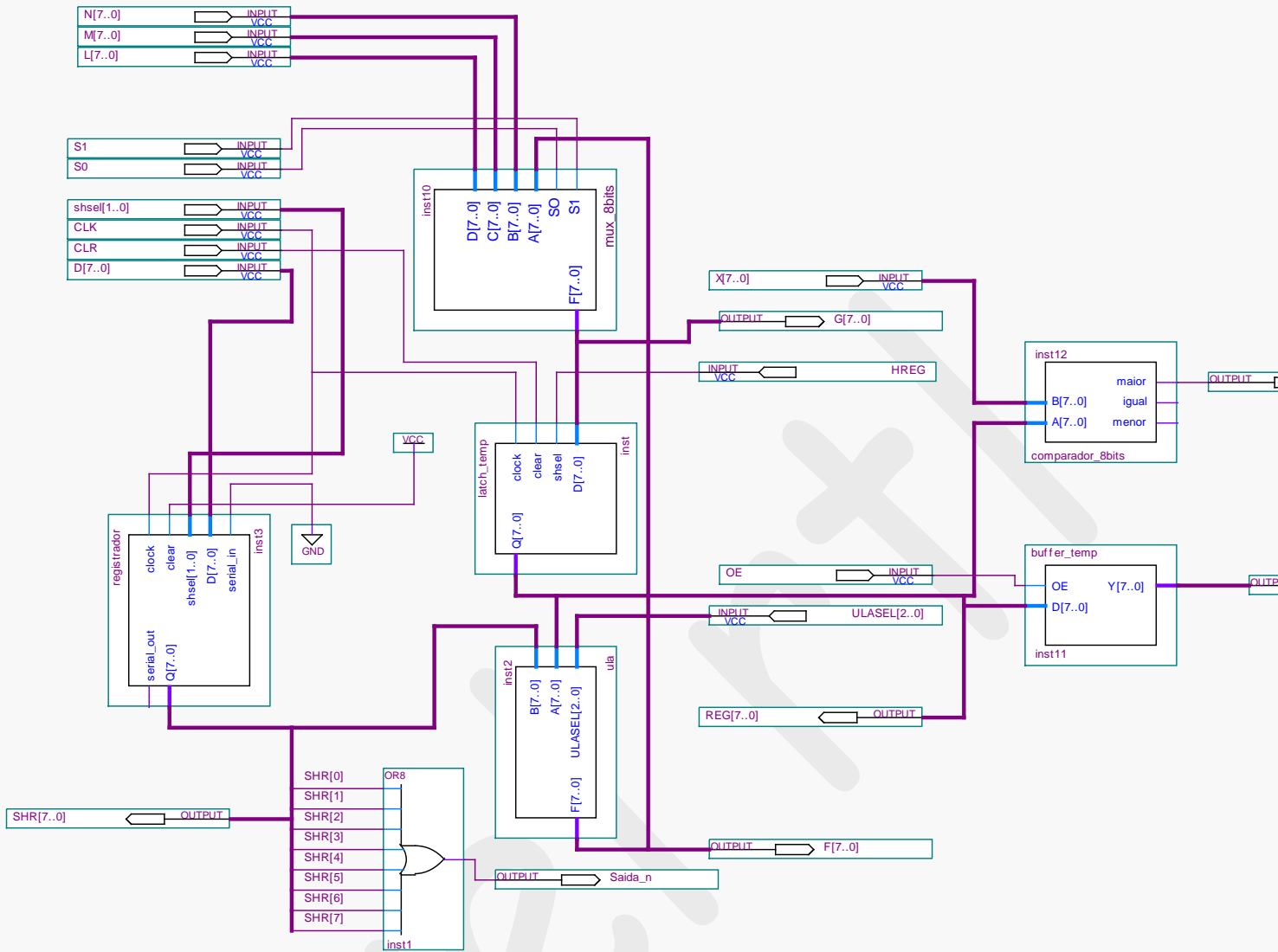
c) Quadro de instruções.

Item	Instrução	S ₀	LD1	SH1	SH0	UA	OE	PC'S	Valor
1	INICIAL.	1	0	0	0	0	0	0	04
3	SOMA	0	1	0	0	0	0	1	10
4	AJUSTA	0	1	0	0	1	0	2	12
5	DESLOCA	0	0	0	1	0	0	3	04
6	PÁRA	0	0	0	0	0	1	4	01

d) Implementação da U.C.

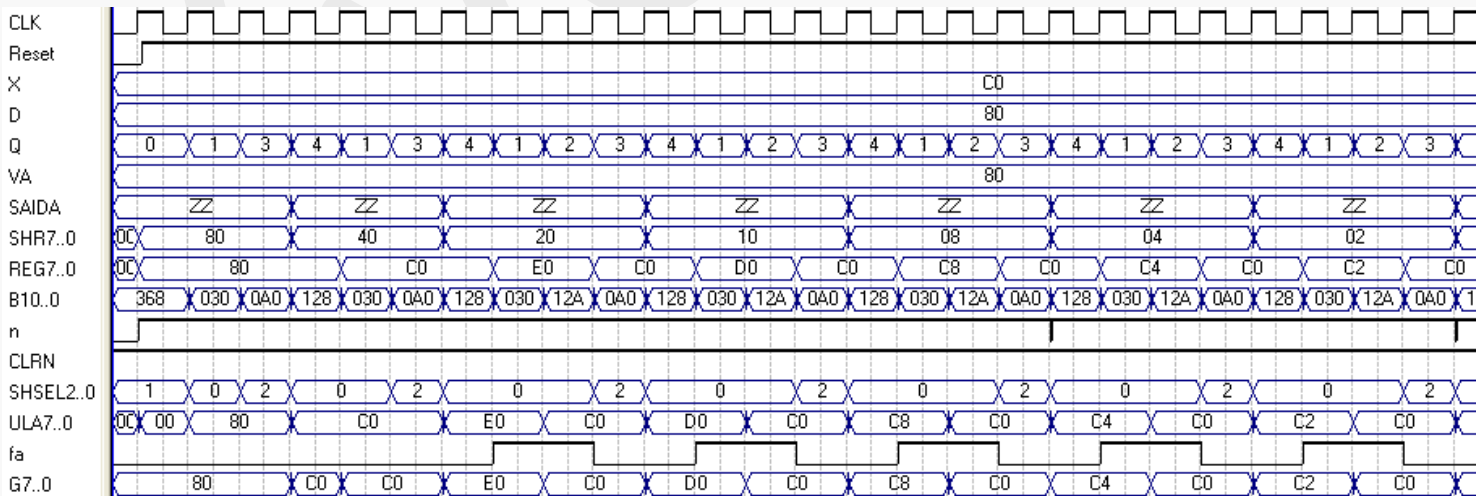
			n'f'	n'f	nf'	nf	-
Q ₂	Q ₁	Q ₀	00	01	10	11	Saída
0	0	0	100	100	001	001	PC0
0	0	1	100	100	011	010	PC1
0	1	0	011	011	011	011	PC2
0	1	1	010	010	010	010	PC3
1	0	0	100	100	100	100	PC4
1	0	1	-	-	-	-	-
1	1	0	-	-	-	-	-
1	1	1	-	-	-	-	-

Formas de ondas da U.C. com saídas PC's



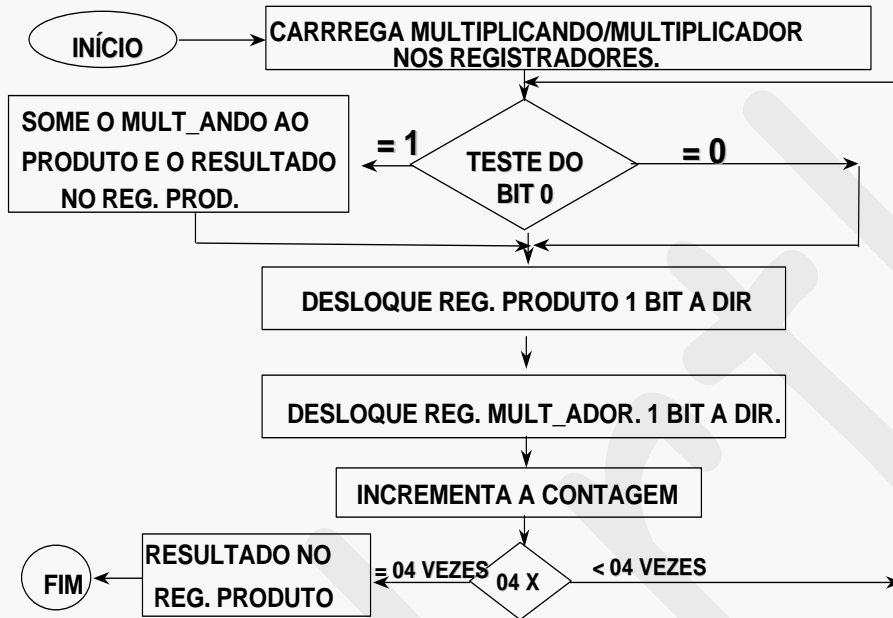
Simulação das formas de ondas do fluxo de dados.

X = C0



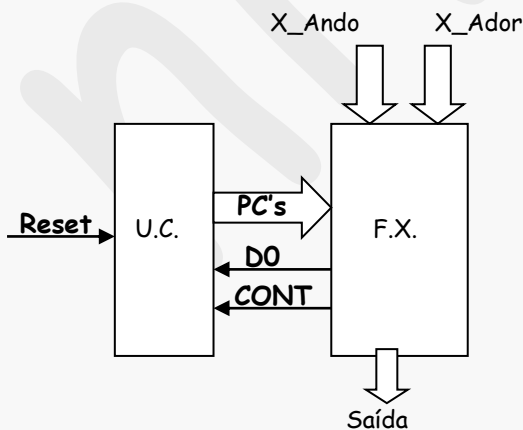
VI - EXERCÍCIOS PROPOSTOS

EXERCÍCIO1 - Dado o algoritmo abaixo, realizar a simulação dos números Multiplicando = (+ 6) e Multiplicador (+ 5). Apresentar toda a sequência do algoritmo.

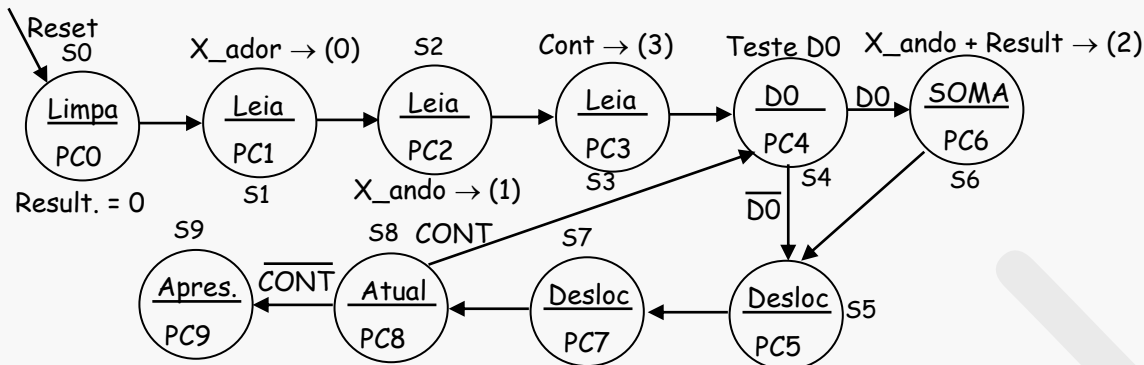


Multiplicando	Multiplicador	Produto	Bit(0)	Contagem
0111	0101	0000 0000	1	4
0111	0010	0011 1000	0	3
0111	0001	0001 1100	1	2
0111	0000	0100 0110	0	1
0111	0000	0010 0011	0	0

Representação esquemática do multiplicador.



a) Projeto da U.C. - Modelo de descrição do sistema por diagrama de estados.



Result → Saída Cont - 1 → (3) Desl. X_ador → (0) Desl. Result → (2)

S0 = 0000, S1 = 0001, S2 = 0010, S3 = 0011, S4 = 0100, S5 = 0101, S6 = 0110, S7 = 0111, S8 = 1000 e S9 = 1001.

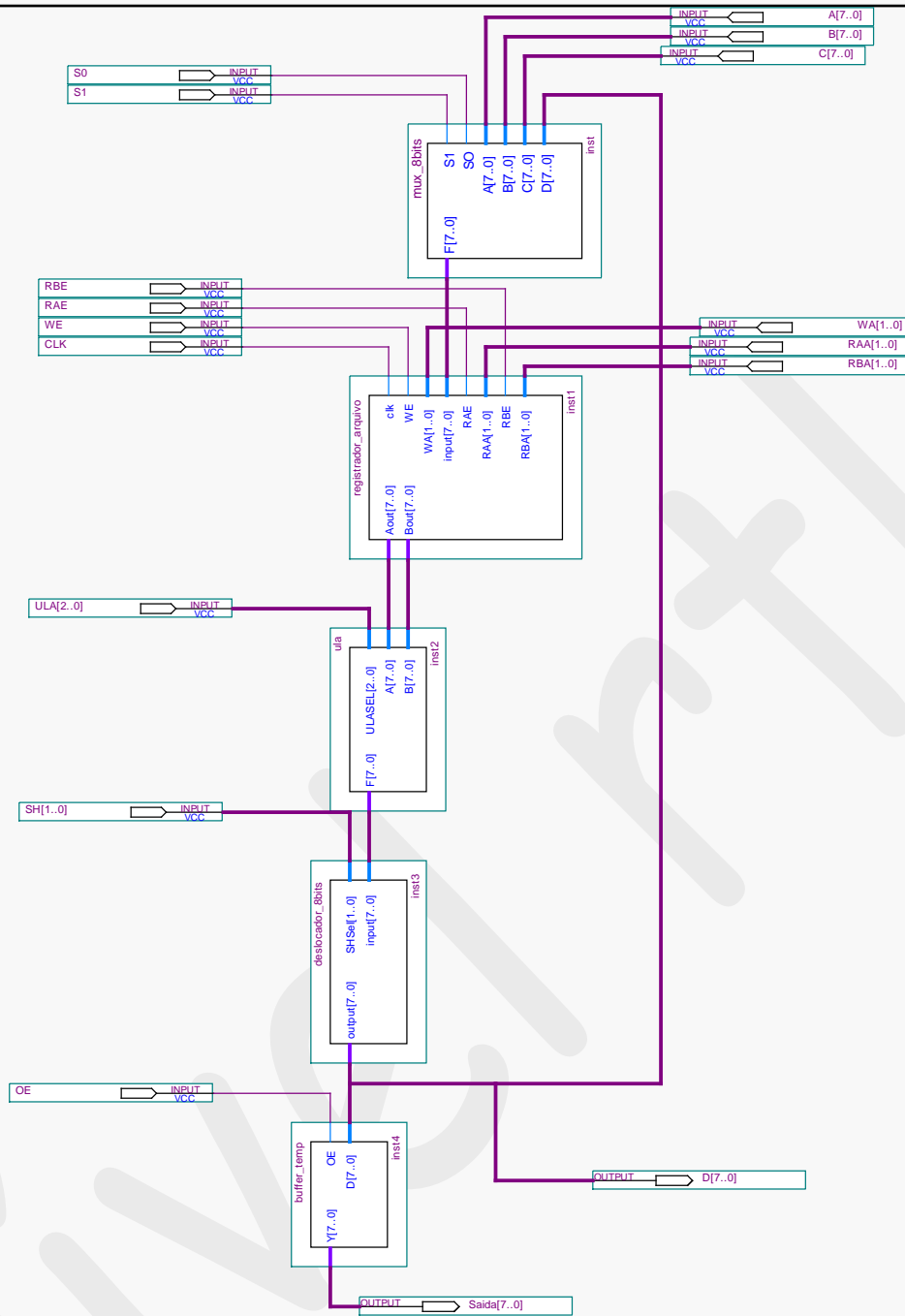
b) Tabela de estados e saída.

Estado	Q ₃	Q ₂	Q ₁	Q ₀	00	01	10	11	PC
S0	0	0	0	0	0001	0001	0001	0001	0
S1	0	0	0	1	0010	0010	0010	0010	1
S2	0	0	1	0	0011	0011	0011	0011	2
S3	0	0	1	1	0100	0100	0100	0100	3
S4	0	1	0	0	0101	0101	0110	0110	4
S5	0	1	0	1	0111	0111	0111	0111	5
S6	0	1	1	0	0101	0101	0101	0101	6
S7	0	1	1	1	1000	1000	1000	1000	7
S8	1	0	0	0	1001	0100	1001	0100	8
S9	1	0	0	1	1001	1001	1001	1001	9

Quadro de instrução

Item	Instrução	I _{E1,0}	W _E	W _{A1,0}	R _{AE}	RAA _{1,0}	R _{BE}	RAB _{1,0}	ULA _{2,0}	SH _{1,0}	OE	PC	Hex
1	Limpa Result.	11	1	10	1	10	1	10	101	00	0	0	1EDA8
2	Leia X_Ador	00	1	00	0	xx	0	xx	xxx	xx	0	1	04000
3	Leia X_Ando	01	1	01	0	xx	0	xx	xxx	xx	0	2	0D000
4	Leia Cont	10	1	11	0	xx	0	xx	xxx	xx	0	3	17000
5	Teste D0	xx	0	xx	1	00	0	xx	000	00	0	4	00800
6	Desloc.Result.	11	1	10	1	10	0	xx	000	10	0	5	1EC04
7	Soma	11	1	10	1	10	1	01	100	00	0	6	1ED60
8	Desloc.X_Ador	11	1	00	1	00	0	xx	000	10	0	7	1C804
9	Atualiza	11	1	11	1	11	0	xx	111	00	0	8	1FE38
10	Apresenta	xx	0	xx	1	10	0	xx	000	00	1	9	00C01

Arquitetura fluxo de dados.



Unidade de Controle.

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_unsigned.all;
```

ENTITY uc_multiplica IS

PORT(

- clkuc : IN std_logic;
- reset : IN std_logic;
- d0,cont : IN std_logic;
- B : OUT std_logic_vector(16 downto 0);
- q : OUT std_logic_vector(3 downto 0);

END uc_multiplica;

ARCHITECTURE Behavioral OF uc_multiplica IS

TYPE maq_estado IS (limpa, Xador, Xando, Contagem, teste, desloca_result, soma, desloca_Xador, atualiza, apresenta);

SIGNAL state: maq_estado;

BEGIN

PROCESS (clkuc)

BEGIN

IF reset = '0' THEN

state <= limpa;

ELSIF clkuc'EVENT AND clkuc = '1' THEN

CASE state IS

WHEN limpa =>

state <= Xador;

WHEN Xador =>

state <= Xando;

WHEN Xando =>

state <= Contagem;

WHEN Contagem =>

state <= teste;

WHEN teste =>

IF DO = '1' THEN

state <= soma;

ELSE

state <= desloca_result;

END IF;

WHEN soma =>

state <= desloca_result;

WHEN desloca_result =>

state <= desloca_Xador;

WHEN desloca_Xador =>

state <= atualiza;

WHEN atualiza =>

IF (cont = '1') THEN

state <= teste;

ELSE

state <= apresenta;

END IF;

WHEN apresenta =>

state <= apresenta;

END CASE;

END IF;

CASE state IS

WHEN limpa =>

B(16)<='1';B(15)<='1';B(14)<='1';B(13)<='1';B(12)<='0';B(11)<='1';B(10)<='1';B(9)<='0';B(8)<='1';

B(7)<='1';B(6)<='0';B(5)<='1';B(4)<='0';B(3)<='1';B(2)<='0';B(1)<='0'; B(0)<='0';

WHEN Xador =>

B(16)<='0';B(15)<='0';B(14)<='1';B(13)<='0';B(12)<='0';B(11)<='0';B(10)<='0';B(9)<='0';B(8)<='0';B(7)<='0';B(6)<='0';B(5)<='0';B(4)<='0';B(3)<='0';B(2)<='0';B(1)<='0'; B(0)<='0';

WHEN Xando =>

B(16)<='0';B(15)<='1';B(14)<='1';B(13)<='0';B(12)<='1';B(11)<='0';B(10)<='0';B(9)<='0';B(8)<='0';B(7)<='0';B(6)<='0';B(5)<='0';B(4)<='0';B(3)<='0';B(2)<='0';B(1)<='0'; B(0)<='0';

WHEN Contagem =>

B(16)<='1';B(15)<='0';B(14)<='1';B(13)<='1';B(12)<='1';B(11)<='0';B(10)<='0';B(9)<='0';B(8)<='0';B(7)<='0';B(6)<='0';B(5)<='0';B(4)<='0';B(3)<='0';B(2)<='0';B(1)<='0';B(0)<='0';

WHEN teste=>

B(16)<='0';B(15)<='0';B(14)<='0';B(13)<='0';B(12)<='0';B(11)<='1';B(10)<='0';B(9)<='0';B(8)<='0';B(7)<='0';B(6)<='0';B(5)<='0';B(4)<='0';B(3)<='0';B(2)<='0';B(1)<='0';B(0)<='0';

WHEN desloca_result =>

B(16)<='1';B(15)<='1';B(14)<='1';B(13)<='1';B(12)<='0';B(11)<='1';B(10)<='1';B(9)<='0';B(8)<='0';B(7)<='0';B(6)<='0';B(5)<='0';B(4)<='0';B(3)<='0';B(2)<='1';B(1)<='0';B(0)<='0';

WHEN soma =>

B(16)<='1';B(15)<='1';B(14)<='1';B(13)<='1';B(12)<='0';B(11)<='1';B(10)<='1';B(9)<='0';B(8)<='1';B(7)<='0';B(6)<='1';B(5)<='1';B(4)<='0';B(3)<='0';B(2)<='0';B(1)<='0';B(0)<='0';

WHEN desloca_Xador =>

B(16)<='1';B(15)<='1';B(14)<='1';B(13)<='0';B(12)<='0';B(11)<='1';B(10)<='0';B(9)<='0';B(8)<='0';B(7)<='0';B(6)<='0';B(5)<='0';B(4)<='0';B(3)<='0';B(2)<='1';B(1)<='0';B(0)<='0';

WHEN atualiza =>

B(16)<='1';B(15)<='1';B(14)<='1';B(13)<='1';B(12)<='1';B(11)<='1';B(10)<='1';B(9)<='1';B(8)<='0';B(7)<='0';B(6)<='0';B(5)<='1';B(4)<='1';B(3)<='1';B(2)<='0';B(1)<='0';B(0)<='0';

WHEN apresenta =>

```
B(16)<='0';B(15)<='0';B(14)<='0';B(13)<='0';B(12)<='0';B(11)<='1';B(10)<='1';B(9)<='0';B(8)<='0';B(7)<='0';B(6)<='0';B(5)<='0';B(4)<='0';B(3)<='0';B(2)<='0';B(1)<='0';B(0)<='1';
```

END CASE;

END PROCESS;

WITH state SELECT

q <="0000" WHEN limpa,

"0001" WHEN Xador,

"0010" WHEN Xando,

"0011" WHEN Contagem,

"0100" WHEN teste,

"0101" WHEN desloca_result,

"0110" WHEN soma,

"0111" WHEN desloca_Xador,

"1000" WHEN atualiza,

"1001" WHEN apresenta;

END Behavioral;

e) Formas de ondas.

7. Parte final:
