



Diseñaremos el circuito **Controlador** utilizando diferentes métodos

- Método tradicional del diseño para los **MSS** de entradas múltiples
- Método de un flip-flop por estado
- Descripción en lenguaje **VHDL**

I. Diseño del circuito Controlador con el método tradicional.

1) Una vez que tenemos el **Diagrama ASM** diseñado, se requiere asignar el código de identificación para cada estado. El número de bits de código depende del número de estados. Para asignar a los siete estados códigos diferentes, se requieren tres bits.

Esto significa que el **Bloque de Memoria de Estados** debe tener tres flip-flops.

Para la asignación de códigos de estados del **Diagrama ASM** se deben aplicar las mismas reglas que en la asignación de códigos en el **Diagrama de Estados**.

Asignaremos los códigos de la siguiente manera:

		y2y1			
		00	01	11	10
y0	0	0	2	6	4
	1	1	3	7	5
		Ta	Td	Tf	Tg
		Tb	Tc	Te	Φ

125



2) Elaborar los mapas del **Decodificador de Estado Siguierte**. Para cada estado se deben analizar las condiciones de transición para cada bit.

Y2 :

		y2y1			
		00	01	11	10
y0	0	0	2	6	4
	1	1	3	7	5
		0	CR	1	0
		0	A=B	0	Φ

Y0 :

		y2y1			
		00	01	11	10
y0	0	0	2	6	4
	1	1	3	7	5
		MP	CR	0	0
		1	0	1	Φ

Y1 :

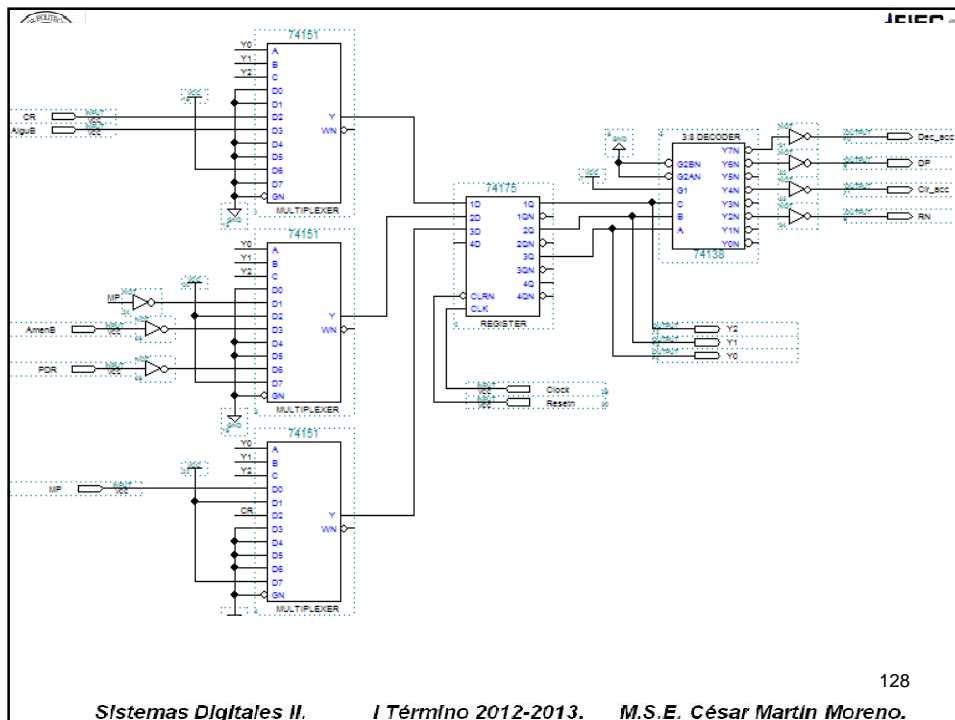
		y2y1			
		00	01	11	10
y0	0	0	2	6	4
	1	1	3	7	5
		0	1	PDR	0
		MP	A<B	1	Φ

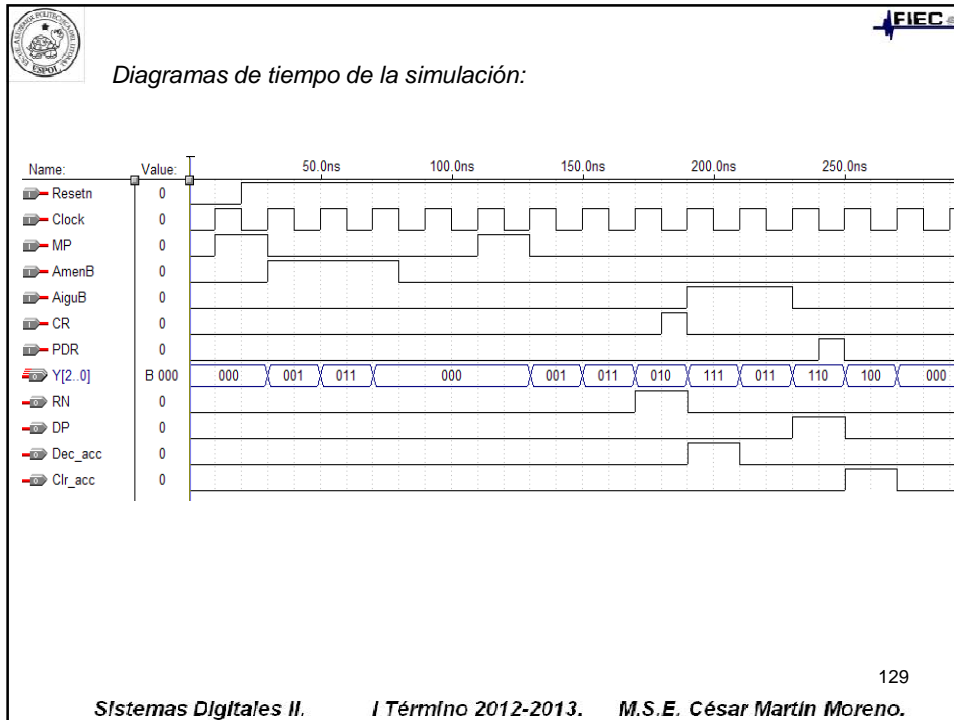
Se pueden simplificar los mapas e implementar las expresiones para **Y2**, **Y1** y **Y0** utilizando puertas lógicas.



Algo más práctico en este caso sería utilizar **Multiplexores 8 a 1** cuyas entradas deben ser conectadas a las señales respectivas de cada celda como indican los mapas.

126

Sistemas Digitales II. I Término 2012-2013. M.S.E. César Martín Moreno.





II. Diseño del circuito Controlador con el método de un flip-flop por estado (ONE HOT).

Debe haber tantos flip-flop como estados hay. No se requiere asignar códigos de estado.

Para definir las expresiones del **Decodificador de Estado Siguiende** se debe analizar el **Diagrama ASM** para detectar de que manera se llega a cada estado.

Si se llega con varias condiciones usamos el operador lógico OR (+) para unir las.

$$Y_a = T_a.\overline{MP} + T_c.A < B + T_g$$

$$Y_b = T_a.MP + T_b.MP$$

$$Y_c = T_b.\overline{MP} + T_e$$

$$Y_d = T_c.A < B.\overline{A} = B + T_d.\overline{CR}$$

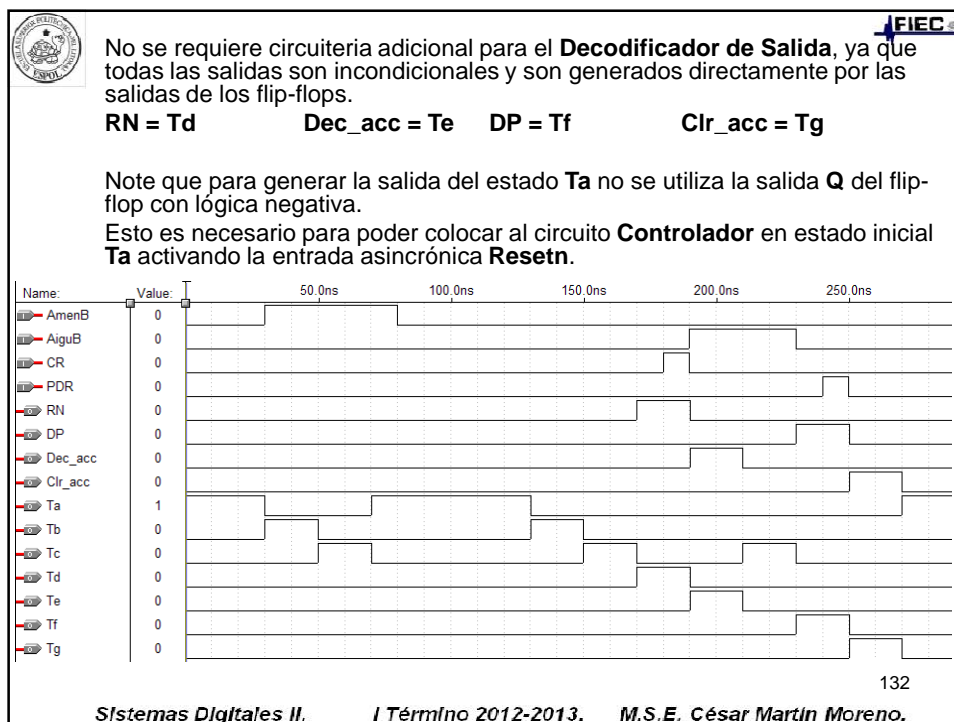
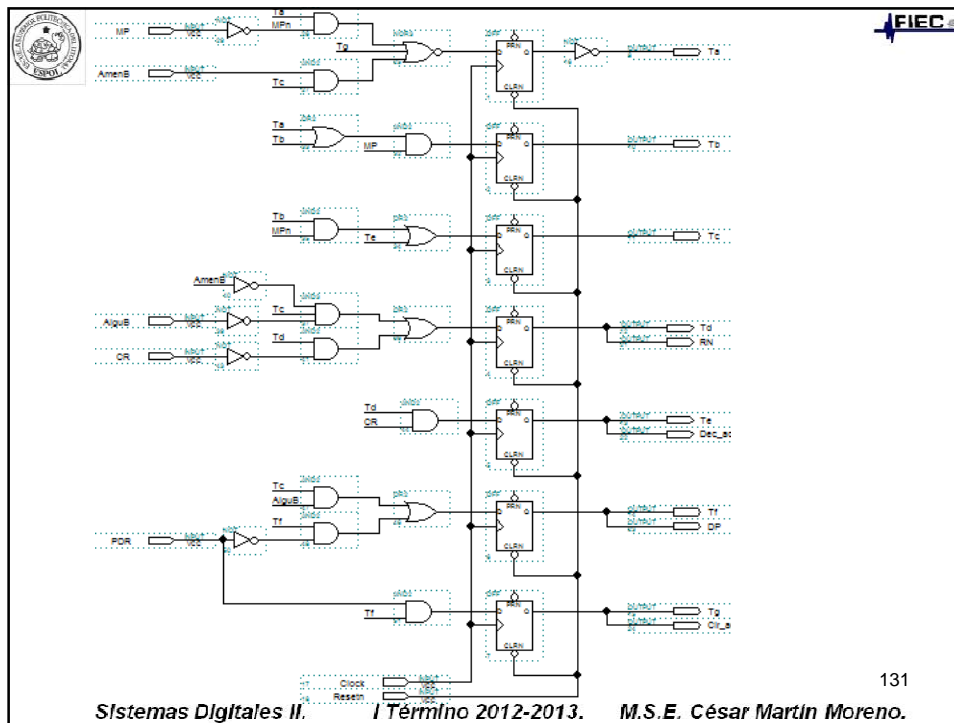
$$Y_e = T_d.CR$$

$$Y_f = T_c.A = B + T_f.\overline{PDR}$$

$$Y_g = T_f.PDR$$

130

Sistemas Digitales II. I Término 2012-2013. M.S.E. César Martín Moreno.





III. Descripción del circuito Controlador en lenguaje VHDL.

```

library ieee;
use ieee.std_logic_1164.all;

entity mvc_cod_vhdl is
port ( Clock, Resetn      : in std_logic;
      MP, AmenB, AiguB, CR, PDR : in std_logic;
      RN, Dec_acc, DP, Clr_acc  : out std_logic);
end mvc_cod_vhdl;

architecture comportamiento of mvc_cod_vhdl is
  type estado is (Ta, Tb, Tc, Td, Te, Tf, Tg);
  signal y: estado;

begin
  MSS_transiciones: process(Resetn, Clock)
  begin
    if Resetn = '0' then y <= Ta;
    elsif (Clock'event and Clock = '1') then
      case y is
        when Ta=>
          if MP = '0' then y <= Ta; else y <= Tb; end if;
        when Tb=>
          if MP = '1' then y <= Tb; else y <= Tc; end if;
        when Tc=>
          if AmenB = '1' then y <= Ta;
          elsif AiguB = '1' then y <= Tf; else y <= Td; end if;
        when Td=>
          if CR = '1' then y <= Te; else y <= Td; end if;
        when Te=> y <= Tc;
      end case;
    end if;
  end process MSS_transiciones;

  RN <= '1' when y=Td else '0';
  Dec_acc <= '1' when y=Te else '0';
  DP <= '1' when y=Tf else '0';
  Clr_acc <= '1' when y=Tg else '0';
end comportamiento;

```

133

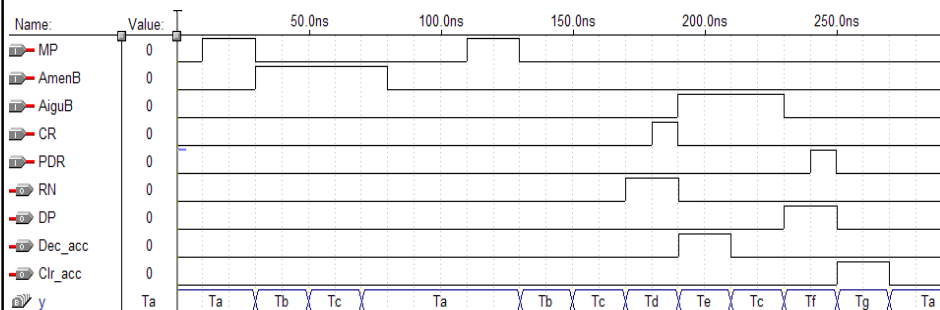
Sistemas Digitales II. I Término 2012-2013. M.S.E. César Martín Moreno.



```

    when Tf=>
      if PDR = '1' then y <= Tg; else y <= Tf; end if;
    when Tg=> y <= Ta;
  end case;
end if;
end process MSS_transiciones;
RN <= '1' when y=Td else '0';
Dec_acc <= '1' when y=Te else '0';
DP <= '1' when y=Tf else '0';
Clr_acc <= '1' when y=Tg else '0';
end comportamiento;

```



134

Sistemas Digitales II. I Término 2012-2013. M.S.E. César Martín Moreno.