

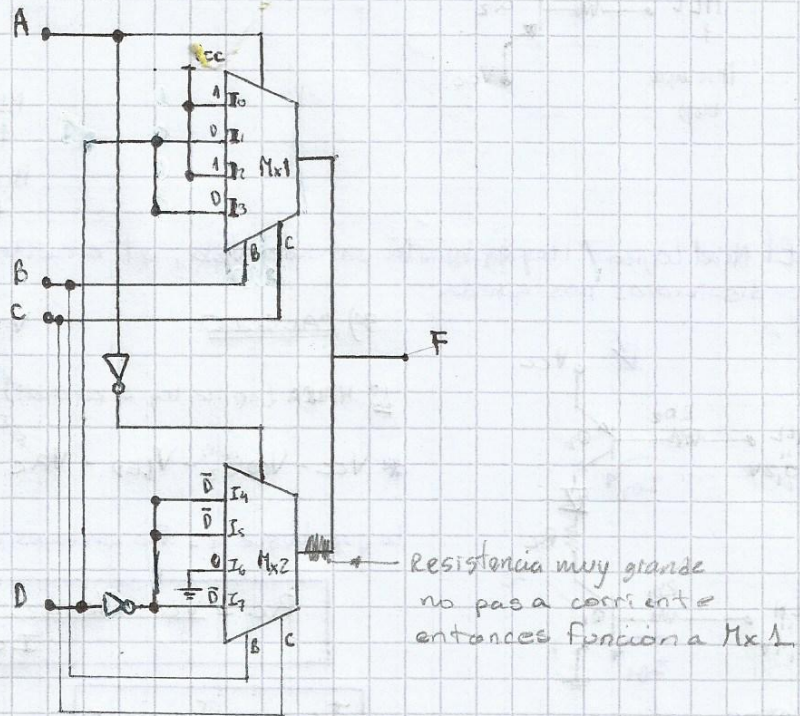
1) Implementar 2 multiplexores 4:1, si la salida de los mismos esta a alta impedancia

$$F_1 = \Sigma(0, 1, 3, 4, 5, 7, 8, 10, 14)$$

1º) Armo la tabla de verdad

2º) Como pide un multi 4:1 entonces divido cada 2.

	A	B	C	D	F <sub>1</sub>	
I <sub>0</sub>	0	0	0	0	1	A
	0	0	0	1	1	
I <sub>1</sub>	0	0	1	0	0	D
	0	0	1	1	1	
I <sub>2</sub>	0	1	0	0	1	A
	0	1	0	1	1	
I <sub>3</sub>	0	1	1	0	0	D
	0	1	1	1	1	
I <sub>4</sub>	1	0	0	0	1	$\bar{D}$
	1	0	0	1	0	
I <sub>5</sub>	1	0	1	0	1	$\bar{D}$
	1	0	1	1	0	
I <sub>6</sub>	1	1	0	0	0	0
	1	1	0	1	0	
I <sub>7</sub>	1	1	1	0	1	$\bar{D}$
	1	1	1	1	0	



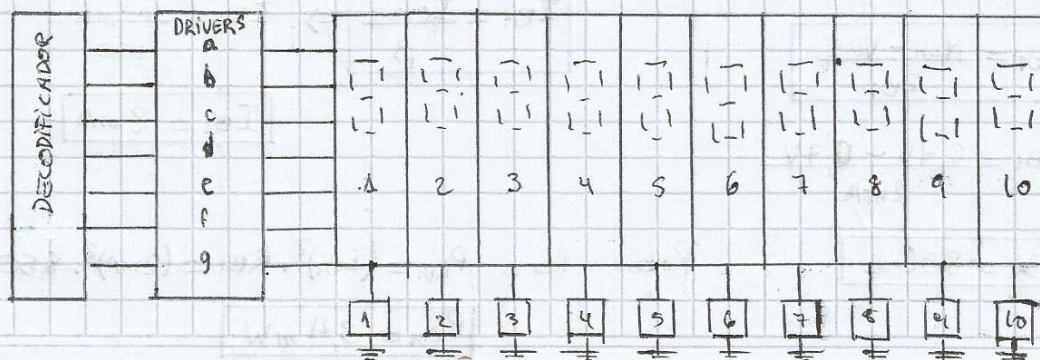
Cuando es alta impedancia, nos quiere decir q' hay una resistencia muy grande entonces la corriente va a optar ir por la menor resistencia

2) Diseñar drivers necesarios para activar 10 display de 7 segmentos, Arado común, sistema multiplexado. La I<sub>prom</sub> (corriente promedio) de los segmentos es igual a 20mA. Realizar el diagrama en bloque del sistema.

Dibujar cada uno de los drivers necesarios. Realizar la selección de los componentes y el calculo de potencia disipada en los mismos.

- a)  $\beta = 100$  b)  $V_{CC} = 5V$  c)  $V_{CE(saturación)} = 0,2V$  d) Nivel logico alto (NLH) = 2,4V  
e) Nivel logico bajo = 0,2V f) Umbral de los segmentos = 1,4V → quiere decir del led.

1º) DIAGRAMA EN BLOQUE.

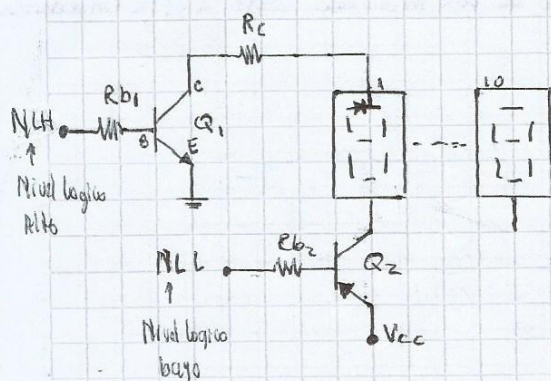


NOTA

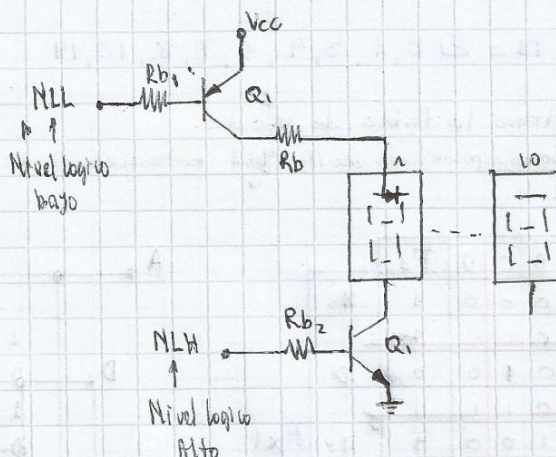


## 2) CIRCUITO

## Anodo Común



## Catodo Común



El Nivel Logico Alto y bajo sería la entrada  $V_{bb}$ , el circuito se repetiría 10 veces, si reediseñamos nos queda:

## 3) CALCULO

$$V = I \cdot R =$$

1ª MALLA (aplico ley de Kirchhoff)

$$* V_{CC} - V_{CEQ2}^{(sat)} - V_{LED} - V_{RC} - V_{CEQ2}^{(sat)} = 0$$

Lo que busco es  $R_C$  entonces despejo

$$R_C = \frac{V_{CC} - V_{CEQ2}^{(sat)} - V_{LED} - V_{CEQ2}^{(sat)}}{I_{CLED}}$$

$$I_{CLED} = I_{prom} \cdot n$$

$n = n^\circ$  de display

$$I_{CLED} = 20mA \cdot 10$$

$$I_{CLED} = 200mA$$

$$R_C = \frac{5V - 0,2V - 1,4V - 0,2V}{200mA}$$

$$R_C = 16\Omega$$

$$Potencia_{R_C} = I_C^2 \cdot R = (200mA)^2 \cdot 16\Omega$$

$$P_{R_C} = 0,64W$$

2ª MALLA p/ calcular  $R_{B1}$

$$* V_{bb}(NLH) - V_{RA1} - V_{BE} = 0$$

$$R_{B1} = \frac{NLH - V_{BE}}{I_{B1}}$$

$$R_{B1} = \frac{2,4V - 0,7V}{2mA}$$

$$R_{B1} = 850\Omega$$

Si

$$I_{B1} = \frac{I_{CLED}}{\beta} \Rightarrow I_{B1} = \frac{200mA}{100}$$

$$I_{B1} = 2mA$$

$$Potencia_{R_{B1}} \Rightarrow P_{R_{B1}} = (I_{B1})^2 \cdot R_{B1} = (2mA)^2 \cdot 850\Omega$$

$$P_{R_{B1}} = 3,4mW$$



3° MALLA p/ cálculo de  $R_{B2}$

$$* V_{CC} - V_{BE} - V_{R_{B2}} - N_{LL} = 0$$

$$R_{B2} = \frac{V_{CC} - V_{BE} - N_{LL}}{I_{B2}}$$

$$I_{B2} = \frac{I_{max}}{\beta}$$

$$I_{max} = I_{CLED} \cdot n'$$

$n' = n^\circ$  de segmento  
que es = 7

$$I_{B2} = \frac{1,4A}{100}$$

$$I_{max} = 200mA \cdot 7 \text{ (segmento)}$$

$$I_{max} = 1,4A$$

$$I_{B2} = 14mA$$

$$R_{B2} = \frac{V_{CC} - V_{BE} - N_{LL}}{I_{B2}} = \frac{5V - 0,7V - 0,2V}{14mA}$$

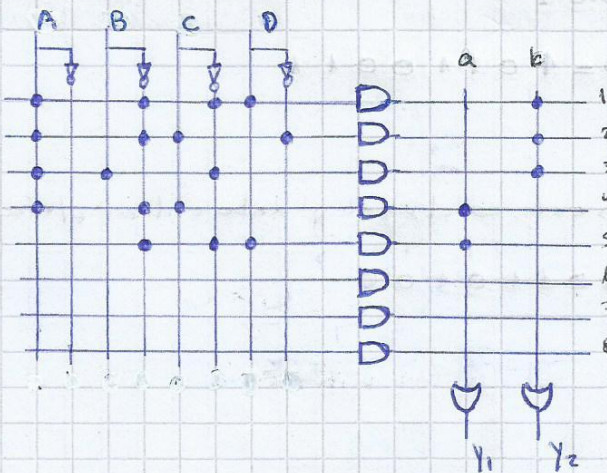
$$R_{B2} = 292,9\Omega$$

$$\text{Potencia } R_{B2} : P_{RB2} = (I_{B2})^2 \cdot R_{B2} = 14mA^2 \cdot 292,9\Omega$$

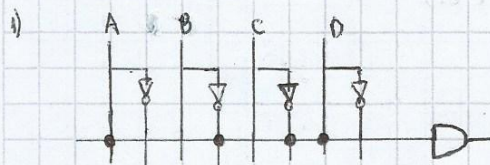
$$P_{RB2} = 57,4mW$$

3° PAL (10), describir  $Y_1$  e  $Y_2$

← solo vale 10 puntos



1) Para empezar a resolver, tenemos dos matrices una de AND y otra de OR. Los puntos q' estan marcados nos indica las lineas a utilizar. Si vemos la primera matriz seria

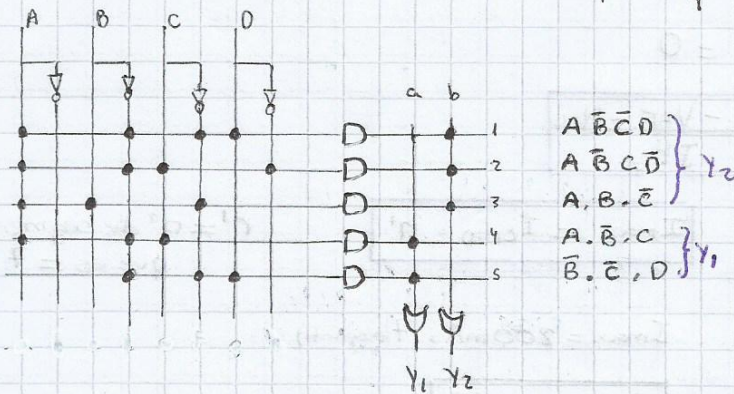


\* nos queda  $\Rightarrow A \cdot \bar{B} \cdot \bar{C} \cdot D$  xq' solo escribo los q' tienen punto marcado.

\* Repetiría lo mismo para las otras AND ejemplo la 2) nos queda  $A \cdot \bar{B} \cdot C \cdot \bar{D}$  y haci para todas las que tienen punto marcado excepto la 6, 7, 8 no estan con puntos.



20) Lo anterior es solo para la matriz de AND cuando pase a la matriz OR. Vemos que solo tiene 5 puntos marcados, eso quiere decir que solo utilizaremos 5 compuertas de AND para obtener la salida  $Y_1$  e  $Y_2$  que es lo que buscamos, como es OR sumamos el resultado de cada punto que esta marcado.



$$Y_1 = A \cdot \bar{B} \cdot C + \bar{B} \cdot \bar{C} \cdot D$$

$$Y_2 = A \cdot \bar{B} \cdot \bar{C} \cdot D + A \cdot \bar{B} \cdot C \cdot \bar{D} + A \cdot B \cdot \bar{C}$$

40) Operaciones con Aritmetica binaria si existe overflow

Hexadecimal  $\rightarrow$  a) 25 - 33 b) 40 - 61 c) 60 + 30

a) 25 - 33

1) convierte 33 a binario, luego invierte y lo sumo 1

$$\text{Hex } 33 = 0011,0011$$

$$+ \begin{array}{r} 11001100 \\ 1 \end{array}$$

$$11001101 = (-33) \rightarrow \text{este es el valor q' debo utilizar p/ efectuar la operacion}$$

$$\text{Hex } 25 = 00100101 \quad (-33) = 11001101$$

$$\begin{array}{r} + 00100101 = 25 \\ + 11001101 = -33 \\ \hline 11110010 = F2 \\ \text{Overflow} \end{array}$$

b) 40 - 61

$$61 = 01100001 \Rightarrow \begin{array}{r} + 10011110 \\ 1 \end{array}$$

$$10011111 = (-61)$$

$$\begin{array}{r} 01000000 (40) \\ + 10011111 (-61) \\ \hline 11011111 D F \\ \text{Overflow} \end{array}$$

c) 60 + 30

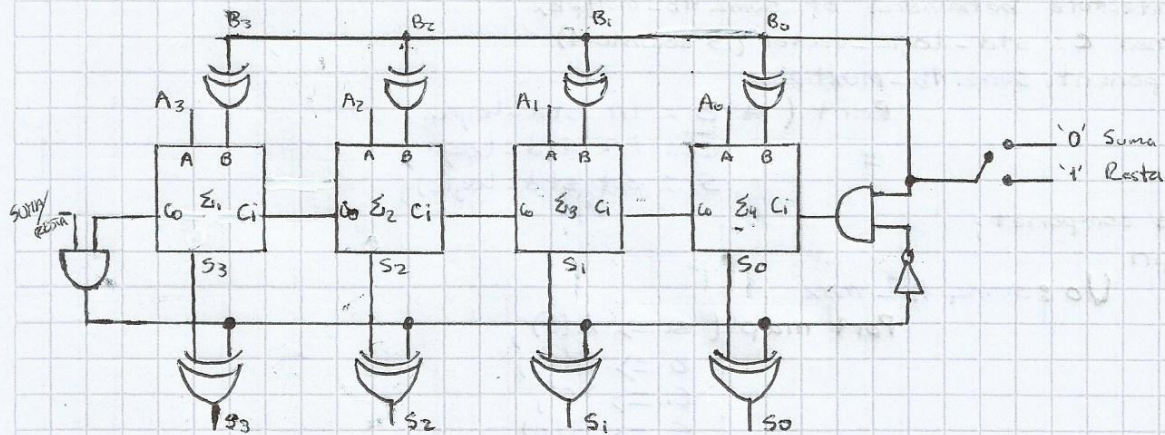
$$\begin{array}{r} 01100000 + 60 \\ + 00110000 + 30 \\ \hline 10010000 90 \end{array}$$



40) a) Circuito Sumador y Restador.

b) ¿Que es margen de ruido? Tiempo de crecimiento y de retardo.

a) Circuito Sumador y Restador



b) MARGEN DE RUIDO

Es la inmunidad mínima que tiene un sistema de un ruido que se mete a lo largo. Ej. si tengo un cable, ¿cuanto ruido puedo inducir sin que cambie a la salida?

\* [Cuanto ruido pueda inducir en la línea sin que el otro vea el ruido]

\* Analisis de ruido en Alto (o tiempo de crecimiento).

EJEMPLO

$$* MRH = VOH_{1(MIN)} - VIH_{2(MAX)} = 3,5V - 2,5V = 1,3V$$

$VOH_{1(MIN)}$  = Tensión de salida alta 1  
 $VIH_{2(MAX)}$  = Tensión de entrada alta 2

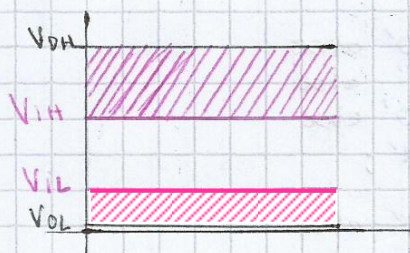
\* Analisis de ruido en Bajo (o tiempo de retardo).

$$* MRL = VOL_{1(MAX)} - VIL_{2(MIN)} = 0,2V - 0,3V = -0,1V$$

No puedo asegurar q' el sistema funcione

El margen de ruido q' siempre voy a tener es el mas pequeño

\* GRAFICA DE MARGEN DE RUIDO





## 5º) VHDL - Sumador 4bits con instanciación

Entity suma4b-multiplex is:

```

Port ( A,B : in std_logic_vector (3 downto 0)
      S : out std_logic_vector (3 downto 0)
      Cout : out std_logic );

```

end suma4b-multiplex;

Architecture behavioral of suma4b-multiplex

Signal C: std\_logic\_vector (3 downto 1)

Component suma1b-mux

```

Port ( A,B : in std_logic;
      Ci : in std_logic;
      S : out std_logic );

```

end component;

begin

U0: suma1b-mux

```

Port map ( a => A(0),
          b => B(0),
          Ci => '0',
          S => S(0),
          Co => C(1));

```

U1: suma1b-mux

```

Port map ( a => A(1),
          b => B(1),
          Ci => C(1),
          S => S(1),
          Co => C(2));

```

U2: suma1b-mux

```

Port map ( a => A(2),
          b => B(2),
          Ci => C(2),
          S => S(2),
          Co => C(3));

```

end behavioral;

Entity suma1b-mux is

```

Port ( a,b : in std_logic;
      Ci : in std_logic;
      S : out std_logic;
      Co : out std_logic );

```

Seria como una función

end suma1b-mux;

Architecture behavioral of suma1b-mux:

begin

```

S <= b when Ci = '0' end a = '0' else
    not b when Ci = '0' end a = '1' else
    not b when Ci = '1' end a = '0' else
    not b when Ci = '1' end a = '1';

```

```

Co <= '0' when Ci = '0' end a = '1' else
    b when Ci = '0' end a = '0' else
    b when Ci = '1' end a = '0' else
    '1' when Ci = '1' end a = '1';

```

end behavioral;