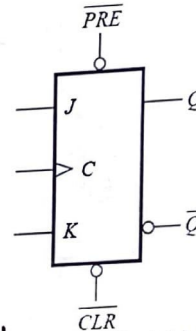
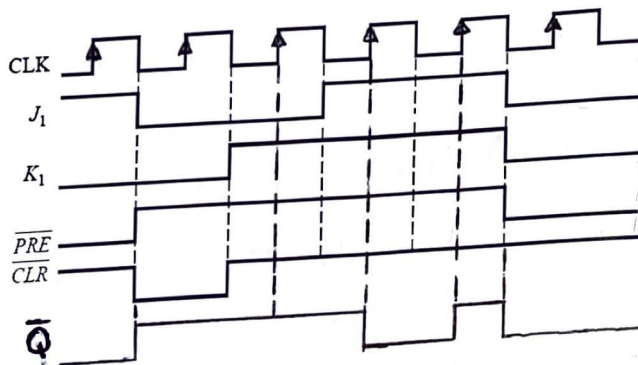


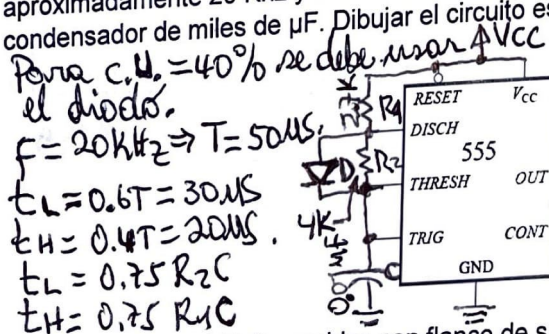
- NOTAS: 1) Obligatorio presentar el parcial con lapicero y en la hoja entregada.
 2) No se permite el uso de calculadora programable ni teléfono celular.
 3) La duración del parcial es 2 horas.

1. Diseñar un convertidor de código Gray de 4 bits a código binario. Designar las entradas con las variables G_D, G_C, G_B, G_A , y las salidas D, C, B, A (LSB).
2. a) Mediante calculo representar en binario complemento a 2 el decimal 64, usar el mínimo número de bits.
 b) Obtener por calculo el mínimo número de bits necesarios para representar en el sistema complemento a 2 el rango de números decimales del 52 al -33.
 c) Realizar la siguiente operación en binario complemento a 2: de -29 restar 35. Usar el mínimo número de bits.
 d) Realizar la siguiente operación con números binarios: 110.01×101.011
3. El Flip Flop de la figura tiene entradas asíncronas que operan de manera independiente a las entradas síncronas (J y K) y a la entrada de reloj, son activas en bajo y significan: \overline{PRE} =PreSet, \overline{CLR} =Clear=Reset. Las señales mostradas se aplican a las entradas del Flip Flop, dibujar la señal en la salida Q' .



No se asume inicialmente el nivel de \overline{Q} porque $PR=0$ lo coloca en cero.

4. Usar el C.I. 555 para diseñar un oscilador de reloj con una frecuencia de aproximadamente 20 KHz y ciclo útil del 40%. No usar resistencias del orden de $M\Omega$ ni condensador de miles de μF . Dibujar el circuito escribiendo los valores obtenidos.



Se asume $C = 0.01 \mu F \Rightarrow$

$R_1 = t_H / 0.75C = 20 \times 10^{-6} / 0.75 \times 0.01 \times 10^{-6}$

$R_1 \approx 2.7 \text{ k}\Omega$

$R_2 = t_L / 0.75C = 30 \times 10^{-6} / 0.75 \times 0.01 \times 10^{-6}$

$R_2 \approx 4 \text{ k}\Omega$

Para 555 $R_1 \geq 1 \text{ k}\Omega$.

5. Usar Flip Flops J-K disparables con flanco de subida para implementar un contador descendente con $M=20$. Presentar el diseño de la lógica adicional para obtener dicho modulo.

2.a) Obtener el complemento a 2 de 64. Como el número es positivo simplemente se convierte el decimal a binario y se le adiciona a la izquierda el bit de signo 0.

$64 \div 16 = 4 \text{ Res. } 0$

$4 \div 16 = 0 \text{ Res. } 4$

$64_{10} = 40_{16} = 01000000_2$

b) Para este rango el número de bits lo determina el número que tenga la mayor magnitud y en este caso es +52, también se puede resolver convirtiendo 52 a binario y agregar el bit de signo 0.

$$52 \div 16 = 3 \text{ Res. } 4 \rightarrow 52_{10} = 34_{16} = 0110000 \text{ ó sea 7 bits.}$$

$$3 \div 16 = 0 \text{ Res. } 3$$

c) $(-29) \rightarrow (-29) \quad | -29| = 29 = 0011101 \quad ; \quad |-35| = 35 = 0100011$
 $- (+35) \quad + (-35)$
 $\begin{array}{r} 11111 \\ 1100011 \\ + 1011101 \\ \hline 101000000 \end{array}$
 $\begin{array}{r} 1100010 \\ 1100010 \\ + 1 \\ \hline 1100011 \end{array}$
 $\begin{array}{r} 1011100 \\ 1011100 \\ + 1 \\ \hline 1011101 \end{array}$

bit de desborde y no es parte de la respuesta.

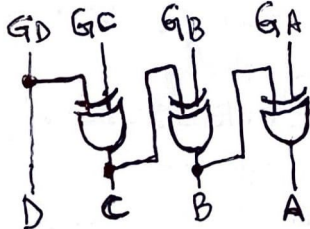
d) $\begin{array}{r} 110011 \\ 110011 \\ \times 11011 \\ \hline 110011 \\ 110011 \\ 110011 \\ 110011 \\ 110011 \\ \hline 11001111000011 \end{array}$

1. Diseño del convertidor de código Gray a binario de 4 bits.

G ₄ G ₃ G ₂ G ₁	D C B A	G ₄ G ₃ G ₂ G ₁	D C B A
0 0 0 0	0 0 0 0	1 1 0 0	1 0 0 0
0 0 0 1	0 0 0 1	1 1 0 1	1 0 0 1
0 0 1 0	0 0 1 0	1 1 1 0	1 0 1 0
0 0 1 1	0 0 1 1	1 1 1 1	1 0 1 1
0 1 0 0	0 1 0 0	1 0 0 0	1 1 0 0
0 1 0 1	0 1 0 1	1 0 0 1	1 1 0 1
0 1 1 0	0 1 1 0	1 0 1 0	1 1 1 0
0 1 1 1	0 1 1 1	1 0 1 1	1 1 1 1

De la tabla se obtiene:

1. D = G₄.
2. C se obtiene comparando G₄ con G₃ ó sea C = G₄ ⊕ G₃.
3. B se obtiene comparando G₃ con G₂ ó sea B = G₃ ⊕ G₂.
4. A se obtiene comparando G₂ con G₁ ó sea A = G₂ ⊕ G₁.



5. Para implementar este contador son necesarios 5 Flip Flops.

CK	E	D	C	B	A	PR
0	1	1	1	1	1	1
1	1	1	1	1	0	1
2	1	1	1	0	1	1

Se usa entrada asincrónica PR por ser descendente, PR = E + C, se usan solo las variables que son 0 en el E.T.

