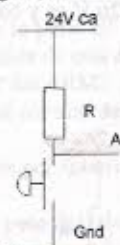


Finales – Técnicas Digitales II

Examen Final 2014-03-06

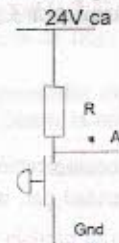
- 1.- Diseñe la interfase para el control de un contactor con bobina de 220 V ca y consumo de 300 mA. El diseño debe asumir que Ud. utiliza un pin de E/S del ARM para ello. Se debe:
 - a. Elegir si la señal de mando para el contactor opera por nivel o flanco. En cada caso elegirlo.
 - b. Dibujar el circuito de mando, con todos los componentes necesarios.
- 2.- Diseñe la interfase para el manejo de la señal de entrada de la figura, usando el punto (A) como acceso hacia el procesador.



- 3.- Usando un procesador basado en arquitectura ARM, con un ADC interno de 10 bits y tensión de referencia fija de 3,3 V. Utilizando un transductor cuya función de transferencia es $20 \mu\text{A}/^\circ\text{C}$, calcular:
 - Ganancia necesaria para poder medir una temperatura de hasta 80°C con una resolución de $0,1^\circ\text{C}$ por cuenta.
 - Se desea un error de 1%. Especificar la tolerancia de los resistores que establecen la ganancia del amplificador.
 - Especificar las características que debe tener el amplificador. La señal es de continua.
4. Se tiene un sistema basado en ARM-7 que tiene tres lazos de comunicación por RS 232-C. Se debe implementar el canal N° 3 de manera discreta. Las condiciones de trabajo son: Frame: 8 bits dato, paridad par, un bit stop. Bitrate: 900 bps. Usted debe implementar el canal de recepción, que implica:
 - Diseño del reloj para el receptor de datos. Esto incluye:
 - Especificación de la frecuencia de trabajo y la relación marca-espacio.
 - Especificación básica del circuito de reloj (p. ej. basado en 555 o similar; usando dos inversores y un cristal; etc.) NO ES NECESARIO el circuito real.
 - Diseño del circuito de sincronización de los datos de recepción, completo. Debe incluir la adaptación de nivel RS 232 a lógica usada.

Examen Final 2014-07-03

- 1.- Usted dispone de un pin de e/s de un procesador ARM. Con el debe operar un contactor con bobina de 220 V ca y consumo de 300 mA que DEBE ser accionado por el flanco decreciente de un pulso de $1 \mu\text{s}$ de ancho (mínimo). El diseño debe asumir que Ud. memoriza ese flanco en un componente externo para accionar el contactor y utiliza un otro pin de E/S del ARM para desactivarlo. Se debe:
 - a. Dibujar el circuito de mando, con todos los componentes necesarios.
 - b. Escribir una rutina en el Assembler del ARM para generar el flanco descendente, con la duración especificada.
- 2.- Diseñe la interfase para el manejo de la señal de entrada de la figura, usando el punto (A) como acceso hacia el procesador. Si $R = 47\text{k}\Omega$ dimensione los componentes para que la constante de tiempo sea menor que 5 ms.



- 3.- Usando un procesador basado en arquitectura ARM, con un ADC interno de 10 bits y tensión de referencia fija de 3,3 V. Se tiene un transductor cuya función de transferencia es $80 \mu\text{V}/^\circ\text{C}$ [$0\text{V} = 0^\circ\text{C}$], calcular:
 - Ganancia necesaria para poder medir una temperatura de hasta 50°C con una resolución de $0,05^\circ\text{C}$ por cuenta.
 - Se desea un error de 0,1%. Especificar la tolerancia de los resistores que establecen la ganancia del amplificador.

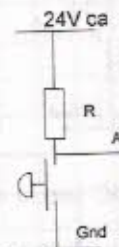
Duda en coef. amp.

Examen Final 2014-05-22

IDEM Examen final 2014-03-06

Examen Final 2014-02-20

- 1.- Diseñe la interfase para el control de un contactor con bobina de 220 V ca y consumo de 150 mA. Asimismo la del manejo de la señal de entrada de la figura, usando el punto (A) como acceso hacia el procesador.



2.- IDEM Ejercicio 3 Examen final 2014-03-06

3.- IDEM Ejercicio 4 Examen final 2014-03-06

Examen Final 2014-07-03

1.- IDEM Ejercicio 1 Examen Final 2014-07-03

2.- IDEM Ejercicio 2 Examen Final 2014-07-03

3.- IDEM Ejercicio 2 Examen Final 2014-07-03

4.- IDEM Ejercicio 4 Examen Final 2014-03-06

Examen Final 2013-12-12

1.- IDEM Ejercicio 2 Examen Final 2014-02-20

2.- IDEM Ejercicio 1 Examen Final 2014-02-20

- 3.- Una variable de tipo word denominada "puntero" posee la dirección de memoria de un vector de 16 elementos de tipo byte sin signo, escribir el código en assembler necesario para calcular el promedio de este vector.

- 4.- Dibuje el circuito de control de un display de diodos de siete segmentos de 4 dígitos y su interfase con el microcontrolador, para la misma no se permite usar mas de 4 líneas GPIO del microcontrolador en total.

Examen Final 2013-10-31

- 1.- Se tiene un sistema basado en ARM-7 que tiene tres lazos de comunicación por RS 232-C. Se debe implementar el canal N° 3 de manera discreta. Las condiciones de trabajo son: Frame: 8 bits dato, paridad par, un bit stop. Bitrate: 900 bps. Usted debe implementar el canal de recepción, que implica:

- Diseño del reloj para el receptor de datos. Esto incluye:
 - Especificación de la frecuencia de trabajo y la relación marca-espacio.
 - Especificación básica del circuito de reloj (p. ej. basado en 555 o similar; usando dos inversores y un cristal; etc.) NO NECESARIO el circuito real.
- Diseño del circuito de sincronización de los datos de recepción, completo. Debe incluir la adaptación de nivel RS 232 a lógica usada.
- Diseño del circuito de pasaje serie-paralelo, incluyendo el detector de paridad.
- Diagrama de bloques de la operación, con el uso de interrupciones.

- 2.- Se desea medir la cantidad de agua en un tanque y para ello se lo pesa. El peso del tanque vacío es de 1780 Kg y puede contener hasta 8 m³ de agua. Se pesa el conjunto con una galga extensiométrica y la salida de esta es 0,4 μ V por Kg. Se pide:
- Para una resolución de 1 litro. ¿Puede usarse el conversor del ARM? ¿para 10 litros? Justificar.
 - Se requiere un error de 1% y resolución de 10 L. Diseñe el circuito de adaptación para una referencia de 3 V. Especifique las tolerancias de los componentes pasivos y de la referencia.
 - Escriba la rutina en Assembler del ARM que convierta el peso del tanque en LITROS de agua. ATENCIÓN: Litros y no litros x 10.

Examen Final 2013-02-28

- 1.- Se desea medir la cantidad de agua en un tanque y para ello se lo pesa. El peso del tanque vacío es de 1780 Kg y puede contener hasta 8 m³ de agua. Se pesa el conjunto con una galga extensiométrica y la salida de esta es 0,4 μ V por Kg. Se pide:
- Para una resolución de 1 litro. ¿Puede usarse el conversor del ARM? ¿para 10 litros? Justificar.
 - Se requiere un error de 1% y resolución de 10 l. Diseñe el circuito de adaptación para una referencia de 3 V. Especifique las tolerancias de los componentes pasivos y de la referencia.
 - Diseñe lo necesario para que el sistema ARM usado opere por interrupción tomando una muestra cada 10 segundos. Describa como ha de configurarse el controlador de interrupciones.
 - Escriba la rutina en Assembler del ARM que convierta el peso del tanque en LITROS de agua. ATENCIÓN: Litros y no litros x 10.
 - Dibuje el circuito de control de un display de diodos de siete segmentos que indique el contenido de agua del tanque en litros, a partir del ARM y hardware externo. Este display NO estará multiplexado.

Examen Final 2013-05-23

- 1.- Utilizando un transductor cuya función de transferencia es 20 μ V/Kg, calcular:
- Ganancia necesaria para poder medir una carga de hasta 800 kg con una resolución exacta de 1 kg por cuenta.
 - Máxima resolución posible y ganancia necesaria para una carga ahora de 300 kg.
 - Calcular error de cuantificación para ambos casos.
- 2.- Especificar nombres de registro y valores a asignar para configurar el controlador de interrupciones de la siguiente forma:
- Asignar el timer0 al vector 0 (IRQ).
 - Asignar el ADC a la FIQ.

Nota: solo para el caso de la interrupción vectorizada incluir además los registros necesarios para especificar la función de atención.

3.- IDEM Ejercicio 3 Examen Final 2013-12-12

4.- IDEM Ejercicio 4 Examen Final 2013-12-12

Examen Final 2013-07-04

- 1.- Usando un procesador basado en arquitectura ARM, por favor elabore este problema: Un silo de granos se ha instrumentado con sensores de temperatura AD592C. Se usan 8 de estos sensores, con un multiplexor adecuado. La función de transferencia del mismo es: 1 μ A/K. Se requiere una resolución de 0,1 °C en el intervalo 10-60 °C. Ud. debe:
- diseñe el hardware necesario para que se muestree esa señal a una tasa de 1 Hz, operando por interrupción. Esto significa que se mide UN SENSOR cada s. Debe incluir el reloj, el multiplexor, el controlador de interrupción del ARM y debe detallar la configuración del mismo.
 - Diseñe el hardware para acoplar el transductor al conversor elegido. Recuerde que se mide en °C y el transductor opera en K.
 - Establezca las tolerancias de los componentes pasivos de su circuito para un error de 1%.
 - Cada vez que se complete el multiplexado de las señales (cada 8 s) se debe calcular el promedio de ellas y además la diferencia de cada una de las medidas con ese promedio. Escriba el diagrama de flujo para esto. Escriba en Assembler del ARM la rutina para calcular el promedio.
 - Si la diferencia de cualquiera de las mediciones con el promedio supera los 2 °C se debe actuar un ventilador, accionando un contactor externo cuya bobina opera en 220 Vca y consume 0,2 A. Diseñe la interfase adecuada.

Examen Final 2013-09-05

- 1.- Usando un procesador basado en arquitectura ARM, por favor elabore este problema: Una planta de tratamiento térmico implica un proceso de carga de un horno, calentamiento controlado del material cargado, un tiempo a temperatura constante y luego enfriamiento también controlado.
- Diseñe el hardware necesario para que se detecte la condición de carga completa. Esto implica: un sensor de proximidad con salida de 220 Vca (correcto = 220 V ca) que detecta que el carro con la carga está en posición y otro óptico con salida de luz uv que detecta que la puerta está cerrada. Este último debe tener una interface optoacoplada. Se pide el circuito fotorreceptor en este último caso y la interfase hacia la entrada del ARM en el primero.
 - Para calentar el horno debe leerse una señal en un potenciómetro cuyos extremos están conectados a 12 Vcc y a masa. Se lo debe leer en 8 bits. Diseñe la etapa de adaptación de señal necesaria para usar el A/D del ARM. Esta señal es la pendiente de la recta de calentamiento.

menor que 32 bits. ($RES \ll 2^{32}$)

- d. La temperatura del horno que varía entre 10 y 950 °C se mide con una termocupla que da un valor de 3 μ V por °C. Ignore la compensación de punta fría. Diseñe la interface para que esta lectura se pueda manejar por el A/D del ARM y especifique las tolerancias de componentes para error < 2%. Considere la resolución normal del A/D y que el fondo de escala corresponde a 1024 °C
- e. El mando de potencia es un contactor cuya bobina opera en 220 Vca y consume 0,2 A. Diseñe la interfase adecuada a partir de un pin de I/O del ARM. La corriente máxima de salida es de 8 mA

IDEM exam 13/7/4 1e

Examen Final 2013-07-25

IDEM Examen final 2014-09-05

Examen Final 2013-02-07

IDEM Examen final 2014-07-04 (Solo varia algunos valores, idéntico ejercicio)

Examen Final 2012-07-26

- 1.- Se tiene que recibir una señal RS 232-C operando a 600 bps. Ud debe pasar la información a un registro de desplazamiento. El frame tiene formato 8, P, 1. Ud. debe transferir la info recibida en serie a un registro de ocho bits. Este registro es leído por el micro cada vez que se completa un caracter y es la **única relación** del procesador con el enlace serie.
- Establezca el método de sincronización de la señal RS 232, con sus componentes necesarios. Especifique la tolerancia (bajo norma) de la frecuencia de reloj y defina cual ha de ser esta frecuencia para una adecuada sincronización. Considere que la señal de entrada ha de muestrearse lo mas cerca posible de $T_{bit}/2$.
 - Dibuje el circuito asociado al registro de desplazamiento. Dibuje asimismo un circuito detallado de los dos primeros bits del mismo registro. Establezca el conjunto de señales necesarias para que este registro opere adecuadamente (reloj; líneas de control de flujo de datos; enables; etc.)
 - Diseñe la interfase del registro y el micro. De algún modo este diagrama debe incluir el clock de comando del registro. Escriba el diagrama de flujo del programa que controla la operación. **Escriba un programa** para controlar la paridad. El conjunto debe operar por interrupción. Esta interrupción se genera cuando se detecta el final del carácter, y debe ejecutarse durante el bit de stop.
- 2.- Un transductor genera información con una función de transferencia de 2,5 μ A/bit. Diseñe la interfase del mismo hacia un conversor A/D de 12 bits, con referencia de 10.0 V. La señal es de cc.
- Establezca las tolerancias de los componentes pasivos de su circuito para un error de 0,1%.
 - Diseñe el hardware necesario para que se muestree esa señal a una tasa de 10Hz, operando por interrupción. Debe incluir el reloj, el controlador de interrupción que use y debe detallar la configuración del mismo.

Examen Final 2012-12-20

- 1.- Se debe medir la salida de un transductor integrado de temperatura. El rango de medición va de 0 a +50 °C (temperatura ambiente). Se pide una resolución de 0,1 °C. La corriente de salida es de 100 μ A por °C, con características similares a las de una fuente ideal de corriente. El error de la cadena completa debe ser igual o menor que 1%. Debe asimismo llevar cuenta de la hora, con resolución de 1 s.
- Diseñe el circuito de acondicionamiento de señal para usar el A/D del ARM. Referencia: 3.0 V. Para el valor de error especificado, indique las tolerancias de los componentes pasivos y explique la elección de los activos.
 - Diseñe el circuito de reloj a utilizar para generar la salida de 1 Hz. Explique como hace para usar esa salida de 1Hz para obtener la información de la hora.
 - La hora DEBE ESTAR en formato hh mm ss, con hh {0 ... 23}. La información **BCD** de este reloj debe ser guardada en un vector. Especifique que tipo de dato usará cada entrada de ese vector (byte, half word o word) para que contenga la información completa de la hora.
 - A partir de la salida de 1 Hz de (2.-), explique como hace para que el sistema opere por interrupción para hacer una conversión del A/D. especifique cual entrada de interrupción usa del ARM usado en los TT PP.
 - Diseñe el servicio de interrupción para que lea **ocho veces** el A/D en cada toma de datos. Escriba un programa en el Assembler del ARM para acumular las lecturas y hacer luego el promedio.

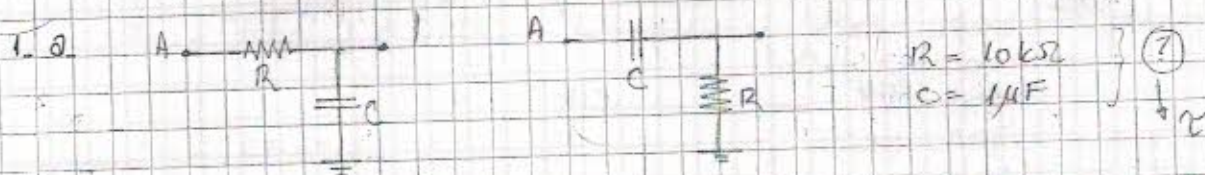
Examen Final 2012-05-24

(IDEM 2012/7/20)

- 1.- Se debe pasar información a un registro de desplazamiento. Mediante un puerto, Ud. debe transferir en serie una cadena de bits a un registro de ocho bits.
- Dibuje el circuito asociado al registro de desplazamiento. Dibuje asimismo un circuito detallado de los dos primeros bits del mismo registro. Establezca el conjunto de señales necesarias para que este registro opere adecuadamente (reloj [de 1 kHz]; líneas de control de flujo de datos; enables; etc.)
 - Diseñe la interfase del registro y el micro. De algún modo este diagrama debe incluir el clock de comando del registro. Este clock se ha establecido en 1 kHz. Escriba el diagrama de flujo del programa que controla la operación.
- IDEM Ejer 2 - 2012/7/20*
- 2.- Un transductor genera información con una función de transferencia de 2,5 μ A/bit. Diseñe la interfase del mismo hacia un conversor A/D de 12 bits, con referencia de 10.0 V. La señal es de cc.
- Establezca las tolerancias de los componentes pasivos de su circuito para un error de 0,1%.

1

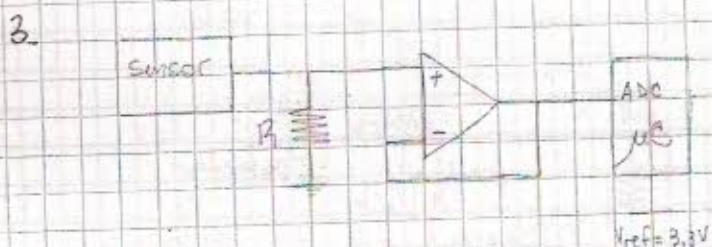
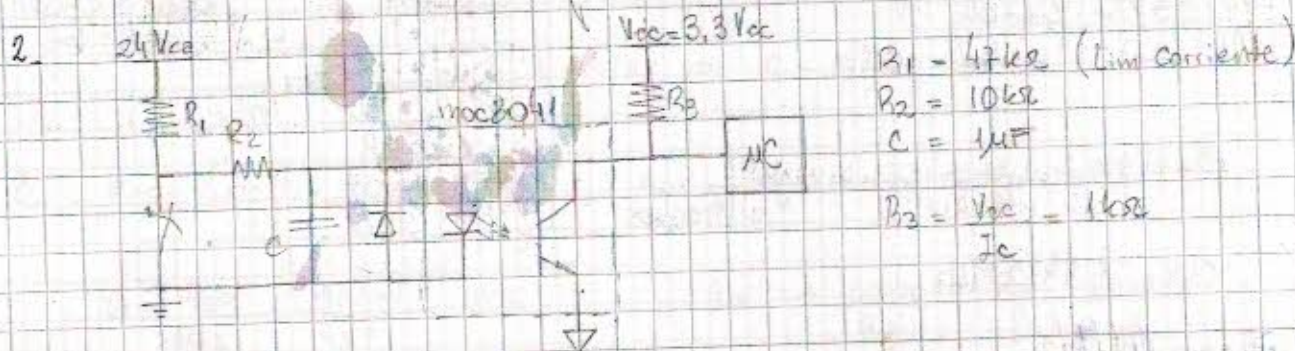
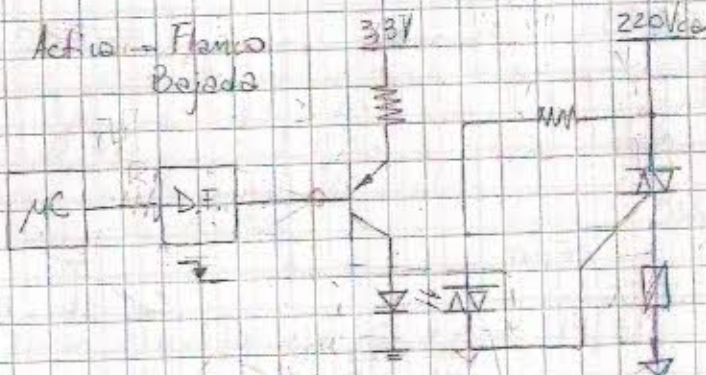
Exam 14/3/6



Detector de Flanco
de Bajada

Detector de Flanco de Subida

5.



$$V_{iso} = \frac{V_{ref}}{2^N} = \frac{3,3V}{2^{10}} = 3,22 \text{ mV/cade}$$

$$R_{es} = 0.1^{\circ}\text{C} \cdot \frac{20 \text{ mA}}{^{\circ}\text{C}} = 2 \text{ mA/}^{\circ}\text{C}$$

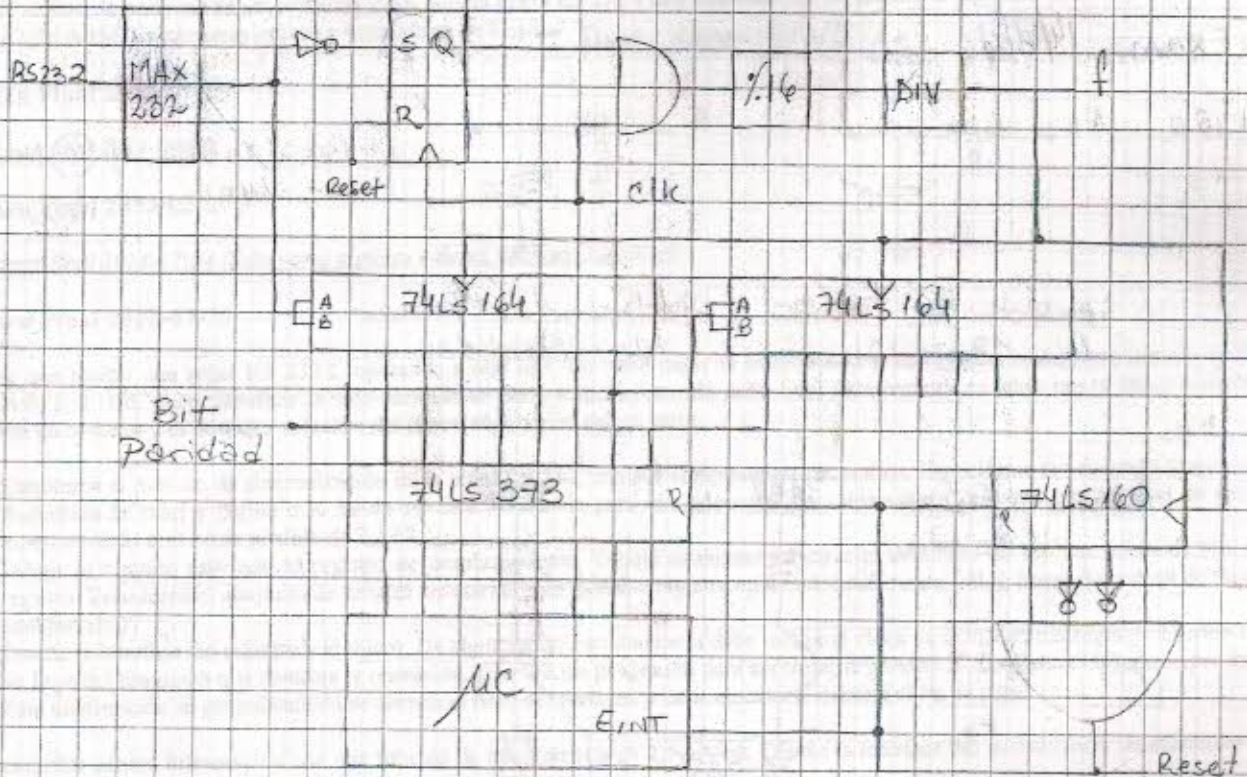
$$G = \frac{V_{USB}}{R_{ES}} = \frac{3,22 \text{ mV/cuenta}}{2 \mu\text{A/cuenta}} = 1,61 \text{ k}\Omega$$

- $E \leq 1\% \Rightarrow G = P_0 + \Delta R \rightarrow$ tolerância 0,1%

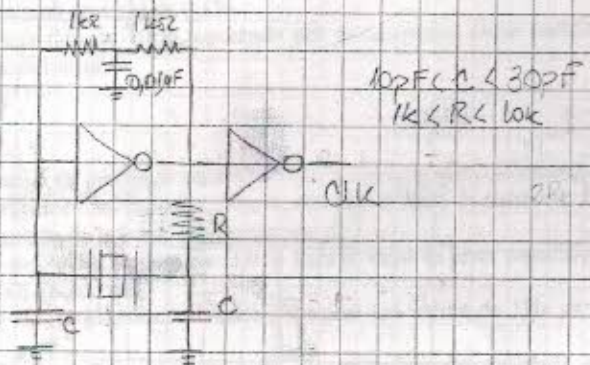
 $\Delta R \ll E$

el error es directamente
proporcional a la resistencia

4. Baudrate: 900 bps ; 8P1



MAX232 - Adaptador TTL
 74LS164 - Shift Register
 74LS373 - Latch
 74LS160 - Contador BCD

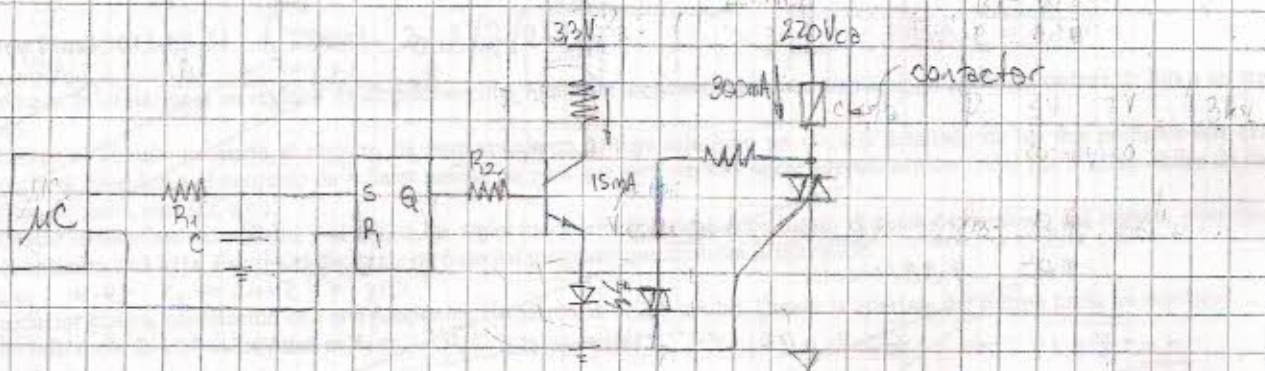


$$\text{BaudRate} = \frac{\text{CLK}}{16 \times \text{Div}} = \frac{1.8432 \text{ MHz}}{16 \times 128} = 900 \text{ bps}$$

$$\text{CLK} = 1.8432 \text{ MHz}$$

Exam 14/7/3

1.2.



b. Se supone $CLK = 14.7456 \text{ MHz}$

$$T_{cy} = \frac{T_{osc}}{4} = 3.3 \text{ MHz}$$

$$t_{cy} = 271 \text{ ns}$$

$$4 \text{ ciclos } nop \rightarrow 1.08 \mu\text{s}$$

Propongo 6 ciclos como solución por ser 1 μs mínimo.

```
reset: mov r3, #0x1
      nop
      nop
      nop
      nop
      nop
      nop
      mov r3, #0x0
```

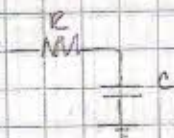
loop: b loop.

* En caso de necesitarse mas ciclos se puede usar una subrutina, pasando a un reg la cantidad de ciclos nop necesarios.

mov r3, #100 //ejemplo

```
demora: nop
        sub r3, r3, #1
        bne demora;
```

2. Idem circuito ejer 2 exam 14/3/6



$$\tau = R \cdot C = 5 \text{ ms } \text{ ó menor}$$

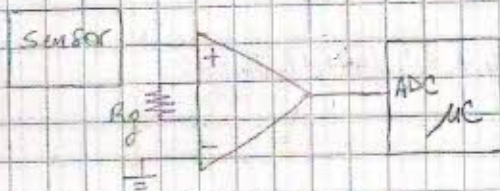
$$\text{si elijo } R = 47 \text{ k}\Omega \Rightarrow C = 0.1 \mu\text{F}$$

$$3. \quad R_{es} = 0.05^\circ\text{C} \times \frac{80 \mu\text{V}}{^\circ\text{C}} = 4 \mu\text{V}, \quad V_{isb} = \frac{V_{ref}}{2^N} = \frac{3.3\text{V}}{2^{10}} = 3.22 \text{ mV}$$

$$G = \frac{V_{isb}}{R_{es}} = \frac{3.22 \text{ mV}}{4 \mu\text{V}} = 805$$

$$G = 1 + \frac{50 \text{ k}\Omega}{R_g} \Rightarrow R_g = \frac{50 \text{ k}\Omega}{G-1}$$

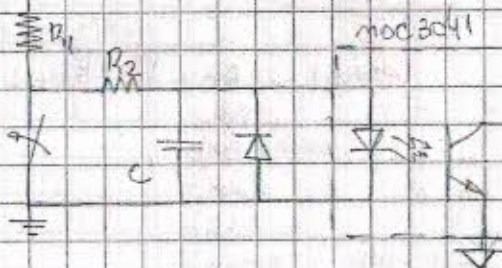
$$R_g = 62 \Omega$$



$$E \leq 0.1\% \rightarrow G = 1 + \frac{50 \text{ k}\Omega}{R_g + \Delta R} \therefore E \gg \Delta R \Rightarrow \text{tolerancia } 0.01\%$$

Exam 14/02/20

1. 24Vca



$$R_1 = 47 \text{ k}\Omega \text{ (lim corriente)}$$

$$R_2 = 10 \text{ k}\Omega$$

$$C = 1 \mu\text{F}$$

$$R_3 = 1 \text{ k}\Omega$$

3.3V

R3

MC

R4

23V

R5

15mA

15mA

Vcc = 27.0Vca

$$\beta = 100 ; V_{cesat} = 0$$

$$R_4 = \frac{3.3\text{V} - 0.9\text{V}}{15\text{mA}/100} = 17.3 \text{ k}\Omega$$

$$R_5 = \frac{V_{cc} - V_{cesat} - V_{opte}}{I_{ua}} = 126 \Omega$$

Exam 13/12/12

3. Reset: mov r3, Puntero // Vector
mov r4, #0 // contador
mov r5, #0 // acumulador

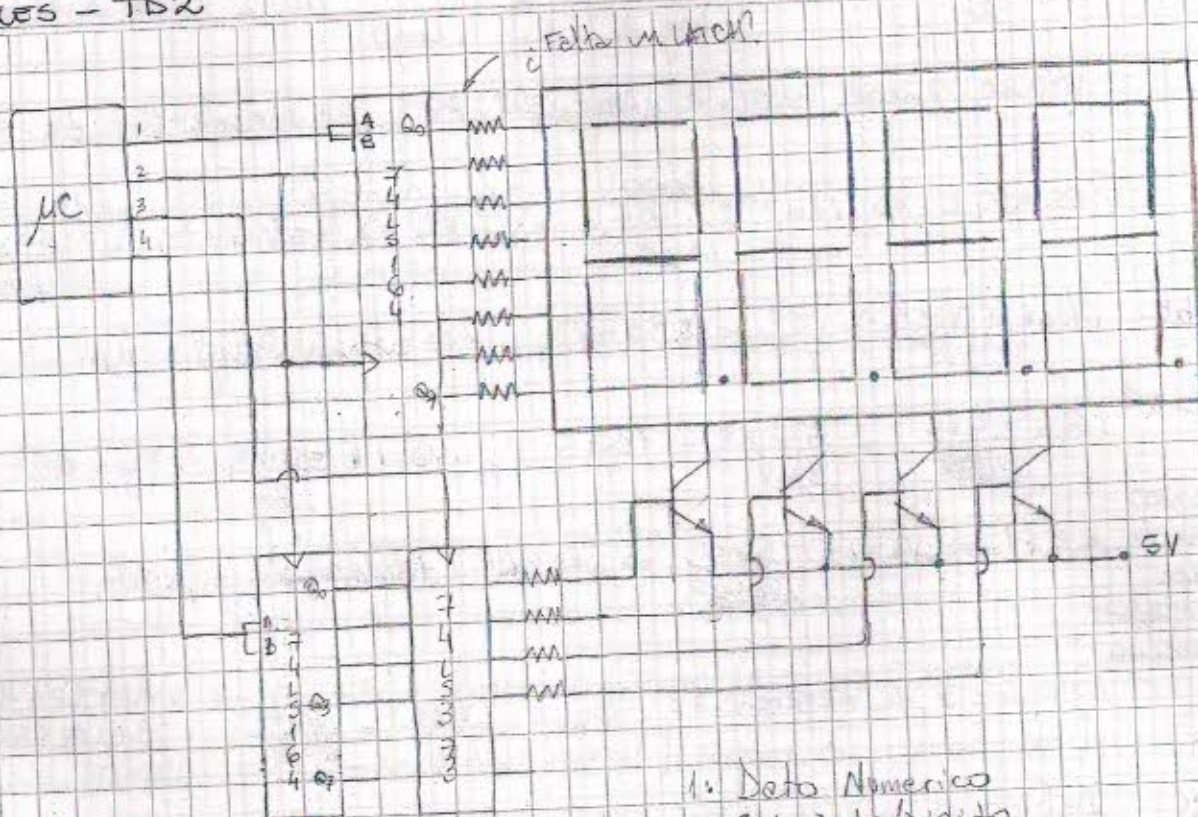
prog: mov r3, Puntero
lwr b r6, [r3], #1
add r5, r6
add r4, r4, #1
cmp r4, #16
bne prog

mov r7, r6, asr #4
str r7, prom

loop: b loop

Puntero: .byte 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
prom: .byte 0
end

4

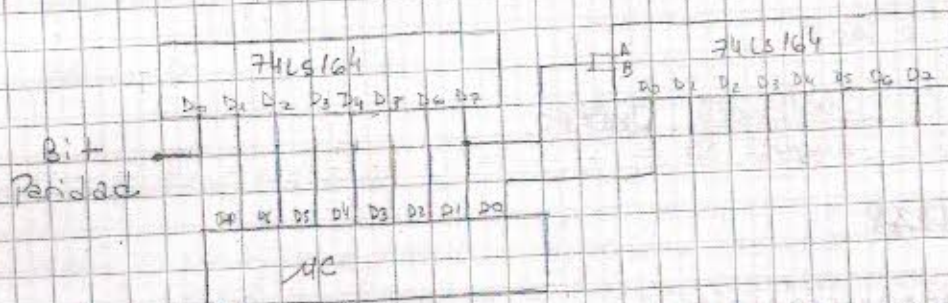


74LS164: Shift register
74LS373: Latch

1: Dato Numerico
2: CLK Dato/Digito
3: CLK Visualización
4: Dato Digito

Exam 13/10/31

1. IDEM circuito ejer 4, exam 14/3/6. Agrega detector Paridad.



Bit Paridad
D7-D0
XOR
error

Al realizar una XOR de toda la palabra de datos en conjunto con el bit de paridad se detectan errores de transmisión.

2.2 $2^N = \frac{\Delta V}{Res} \therefore N = \frac{\log \Delta V / Res}{\log 2} = \frac{\log 8000 L / 1L}{\log 2} = 12,97 \Rightarrow N = 13 \text{ bits}$

No se puede usar el ADC del ARM por ser de 10 bits

$Res = 10L \Rightarrow N = \frac{\log \frac{8000L}{10L}}{\log 2} = 9,64 \Rightarrow N = 10 \text{ bits}$ Si se puede utilizar.

b. $V_{usb} = \frac{V_{ref}}{2^N} = \frac{3V}{2^{10}} = 2,93 \text{ mV}$; $Res = 10L \times \frac{0,4 \mu V}{L} = 4 \mu V$

* $G = \frac{V_{usb}}{Res} = \frac{2,93 \text{ mV}}{4 \mu V} = 732,5$; $G = 1 + \frac{50k\Omega}{R_g} \therefore R_g = 68,35 \Omega$

con $A \geq 500$

se deberían $e \leq 1\% \Rightarrow G = 1 + \frac{50k\Omega}{R_g + \Delta R} \Rightarrow e \ll \Delta R \therefore \text{tolerancia} = 0,1\%$

usar dos

amplificadores

en cascada

Tolerancia de Referencia: $E_{V_{ref}} = \frac{V_{usb}}{4} = \frac{V_{ref}}{2^{n+4}}$ Ampl Op < 10 Vicos
Ampl Op < 500 Vicos

c. Debido a la lectura de a 10L, se debe multiplicar por 10 para obtener la medida en litros.

reset:

```
ldrb r3, LECT
mov r4, #10
mul r5, r3, r4
str r5, PESO
```

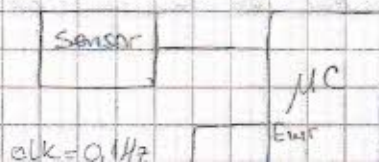
loop: b loop

```
LECT: .halfword 0x0FF0
PESO: .word 0
```

Exam 13/02/28

1. a. IDEM resolución ejer 2a, exam 13/10/21
b. " 2 b. "
d. " 2 c. "

c.



Circuito utilizando interrupciones externas (clock externo), se puede resolver también usando el TIMER0


```
#include "lpc2114.h" // PAG 240 DataSheet

#define TIMER_MATCH0 (1<<0)
#define INTERRUPT_ON_MATCH (1<<0)
#define RESET_ON_MATCH (1<<1)

Void irq_timer0(void) attribute ((interrupt ("IRQ")));

int cont = 0;

int main()
{
    // Config TIMER 0
    TOPR = 0x3F8; // Prescaler = 1000
    TOTCR = 0x02; // reset timer counter y prescaler.

    TOMCR &= 0xFF8; // MATCH0 = TOMCR(0:2) y TOMPR0
    TOMCR |= ((INTERRUPT_ON_MATCH | RESET_ON_MATCH) << 10);
    TOMPR0 = 0x24000; // cuenta hasta llegar (10ms)
    TOTCR = 0x01; // Habilita TIMER y Prescaler

    VICVectCnt10 = 4; // fuente de int. Timer0 (4)
    VICVectCnt10 |= 0x20; // habilita fuente de int.
    VICVectAddr0 = (unsigned long) irq_timer0; // asigna puntero a la func.
    VICIntEnable |= (1<<4);

    return 0;
}

Void irq_timer0(void)
{
    if (T0IR & TIMER_MATCH0) // verifica llamado IRQ
    {
        LECT(); // realiza la función requerida cada 10Seg.
        T0IR = TIMER_MATCH0;
    }
}
```

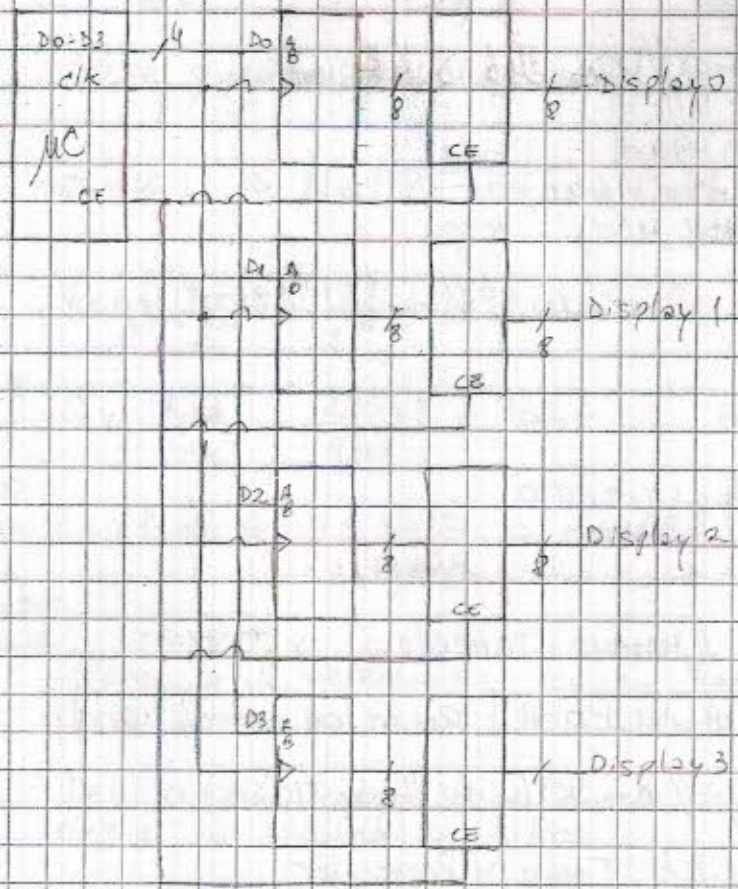
* Este prog genera una interrupción cada 10[S];

$$delay = \frac{1}{f_{clk}} \times TOPR \times TOMPR0 = \frac{1}{14.7456MHz} \times 1000 \times 147456 = 10[S]$$

e.

Shift Register LATCH

Shift register = 74LS164
LATCH = 74LS273



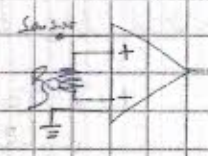
DO-D3: Datos a Display
clk: clk
CE: habilita latch
(mantiene durante carga de datos)

Exam 13/05/23

1.2. $N = \frac{\log 80000/1kg}{\log 2} = 9,64 \Rightarrow N = 10 \text{ bit} \Rightarrow \text{utilizo el del ADS}$

$V_{LSB} = \frac{V_{ref}}{2^N} = \frac{3,3V}{2^{10}} = 3,22 \text{ mV}$; $R_{ES} = 1kg \times \frac{20\mu V}{kg} = 20\mu V$

$G = \frac{V_{LSB}}{R_{ES}} = 161$ $G = 1 + \frac{50kg}{R_g} \Rightarrow R_g = 312,5 \Omega$



b. $2^N = \frac{\text{Rango}}{R_{ES}}$ $R_{ES} = \frac{\text{Rango}}{2^N} = \frac{300kg \times 20\mu V/kg}{2^{10}} = 5,86\mu V$

$V_{LSB} = 3,22 \text{ mV}$; $G = \frac{V_{LSB}}{R_{ES}} = \frac{3,22 \text{ mV}}{5,86\mu V} = 550$

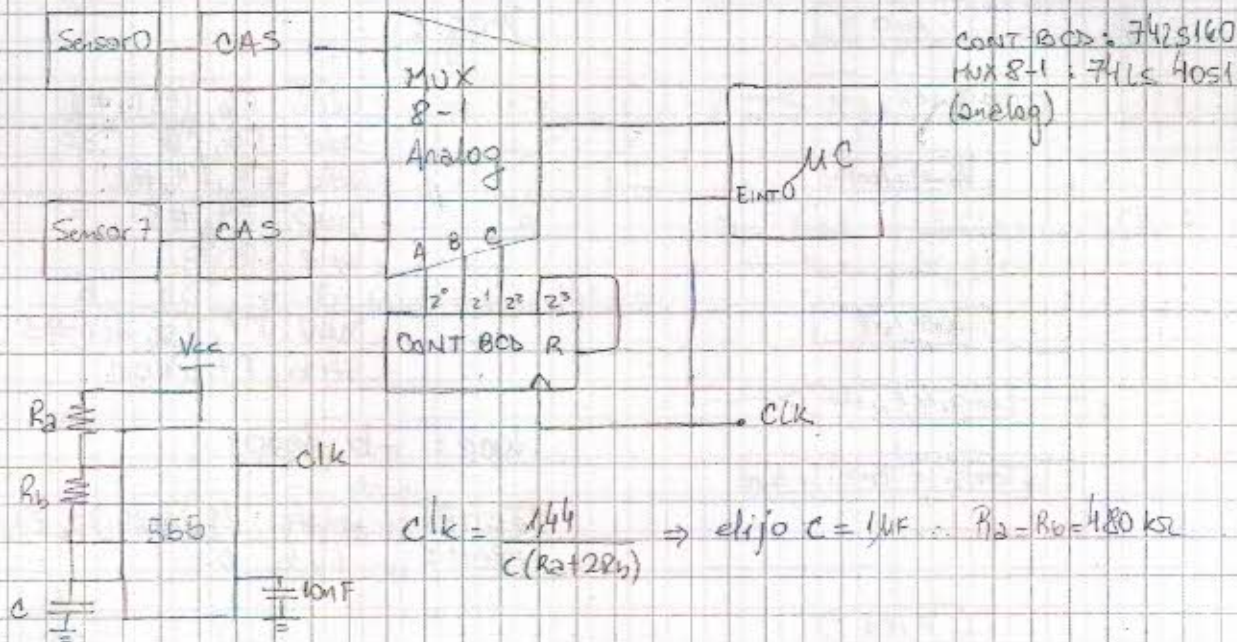
c. Para ambos casos es igual por tratarse del mismo ADC

a. $V_{LSB} = 16\mu V$ $0,05\%$ - error

- 2 a. VICVectCntl0 = 4; // fuente Timer0
 VICVectCntl0 = 0x20; // habilitar fuente interrupción
 VICVectAddr0 = (unsigned long) 1000000; // punto a función
 VICVectEnable = (1<<4); // habilita interrup. del Timer0.
- b. VICIntSelect = (1<<18); // habilita int de FIQ
 VICIntEnable = (1<<18); // habilita la int del ADC (18)

Exam 13/07/04

2



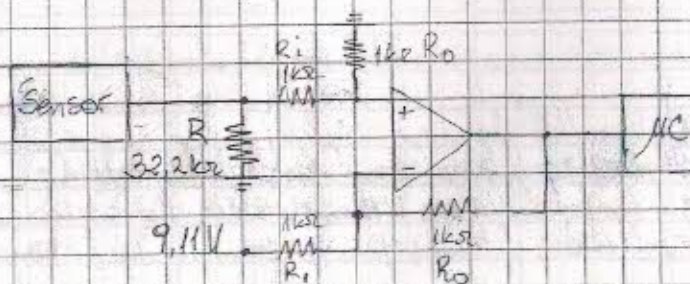
- VICIntSelect = (1<<14); // habilita int de FIQ
 VICIntEnable = (1<<14); // habilita la int de int Externa EINT0 (14)

b.

$$N = \frac{\log(60-10)}{\log 2} \cdot 0,1^\circ C = 8,97 \Rightarrow N = 10 \text{ bits} \rightarrow \text{uso ARM}$$

$$V_{LSB} = \frac{V_{ref}}{2^{10}} = \frac{3,3V}{1024} = 3,22 mV; \quad Res = 0,1^\circ C \frac{\mu A}{^\circ C} = 0,1 \mu A$$

$$G = \frac{V_{LSB}}{Res} = \frac{3,22 mV}{0,1 \mu A} = 32,2 k\Omega$$



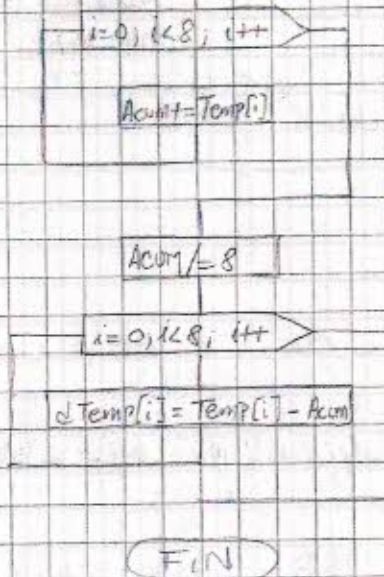
c. $\epsilon \leq 1\%$; $G = \frac{R_0 + \Delta R_0}{R_i + \Delta R_i} \Rightarrow \Delta R \ll R \dots \text{tolerancia} = 0,1\%$

d. INICIO
 Acum = 0
 dTemp = {}
 i = 0

reset:

mov r3, TEMP // vector
 mov r4, #0 // contador
 mov r5, #0 // acum

prog:



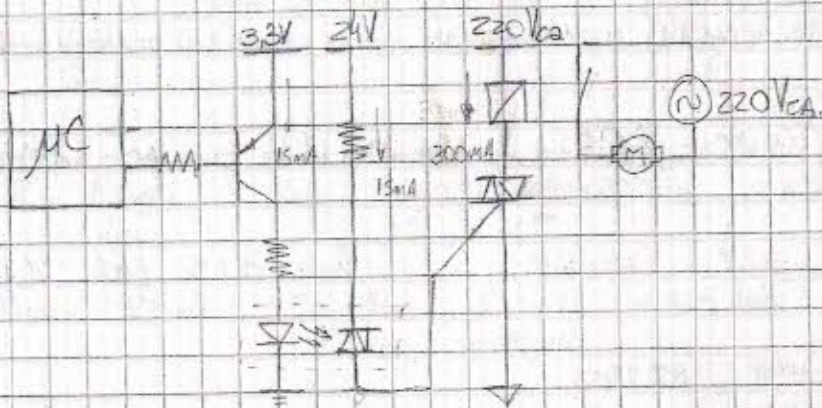
lrb r6, [r3], #1
 add r5, r6
 add r4, r4, #1
 cmp r4, #8
 bne prog

mov r7, r5, asr #3
 strb r7, PROI

loop: b loop

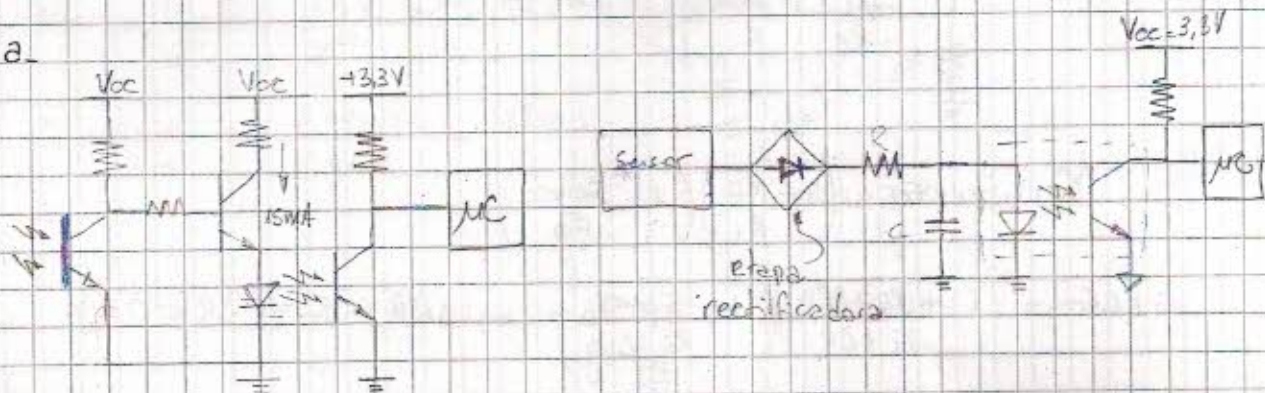
TEMP: .byte 25, 30, 25, 30, 27, 28, 26, 27
 PROI: .byte 0

e.

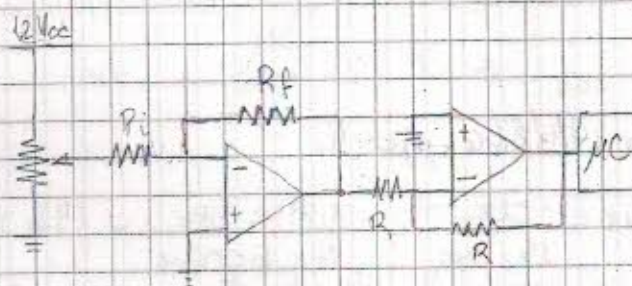


Exam 13/9/5

1.a.



b.



$$R_{ES} = \frac{12V}{2^8} = 46,88mV$$

$$V_{LSB} = \frac{3,3V}{2^{10}} = 3,22mV$$

$$G = \frac{V_{LSB}}{R_{ES}} = \frac{3,22mV}{46,88mV} = 0,069$$

$$G = -\frac{R_f}{R_i} \Rightarrow \text{si } R_i = 10k\Omega \Rightarrow R_f = 145k\Omega$$

c. reset:

ldr b 1B, POT
ldr r4, CON
mul r5, r4, r3
str r5, PROD

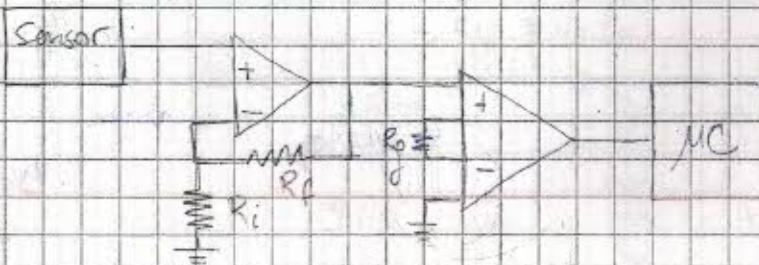
loop: b loop

POT: .byte 0x0F
CON: .word 0x0000FFFF
PROD: .word 0x00000000

d. $\Delta T = 950^{\circ}C - 10^{\circ}C$; $H = \frac{3\mu V}{^{\circ}C}$; $N = 10$; $e \leq 1/2$; $\text{Landa escala} = 1024^{\circ}C$

$$V_{LSB} = \frac{V_{ref}}{2^N} = \frac{3,3V}{1024} = 3,22mV ; R_{ES} = \frac{3\mu V}{^{\circ}C} \times \frac{1024^{\circ}C}{2^{10}} = \frac{3\mu V}{\text{cuenta}}$$

$$A = \frac{V_{LSB}}{R_{ES}} = \frac{3,22mV}{3\mu V} = 1073 ; G_1 = 5 \wedge G_2 = 2^{15}$$



$$G_T = G_1 \times G_2 = \left(1 + \frac{R_f}{R_i}\right) \left(1 + \frac{R_{of}}{R_{of}}\right)$$

$$G_T + \Delta G_T = \left(1 + \frac{R_f + \Delta R_f}{R_i + \Delta R_i}\right) \left(1 + \frac{R_{of} + \Delta R_{of}}{R_{of} + \Delta R_{of}}\right) \Rightarrow \Delta R \ll R \Rightarrow \Delta R = 0.2\%$$

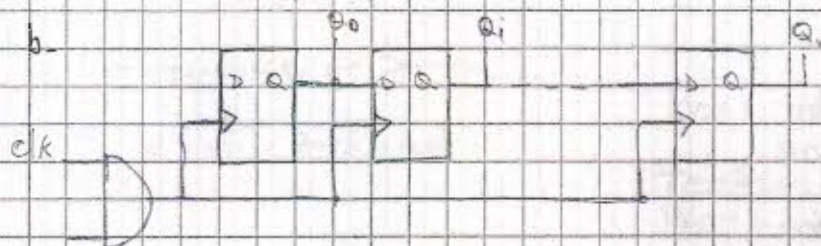
Tolerancia de los resistores $0.2\% = 2000 \text{ ppm}$

Exam 2012/7/6

1. a. Esquema IDEM a Exam 14/3/6 ejer 4.

$$\text{clk} = 1,8432 \text{ MHz}; \quad \text{Div} = \frac{\text{clk}}{16 \times \text{bps}} = \frac{1,8432 \text{ MHz}}{16 \times 600 \text{ bps}} = 192$$

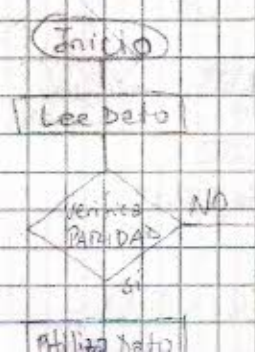
tolerancia reloj (bajo normal) = ?



enable

c. Ver esquema punto a de este examen.

Diag. de Flujo interrupción:
Se realiza cuando la interrupción es llamada.



unsigned char dato, paradas, cont = 0; aux, tag;
int i;

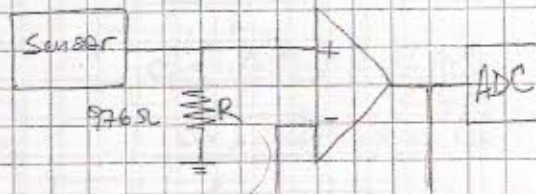
```
for (i = 0; i < 8; i++)
    aux = (dato >> i) & 1;
    if (aux)
        cont++;
```

```
cont += paradas;
if (cont % 2 == 0)
    tag = 1; // OK!
```


2.

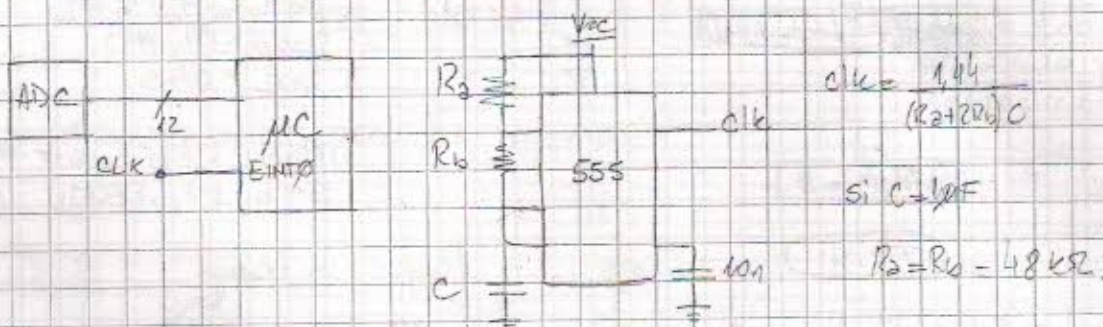
$$V_{LSB} = \frac{V_{ref}}{2^n} = \frac{10V}{2^{12}} = 2,44mV; \quad RES = \frac{2,5\mu A}{bit}$$

$$G = \frac{V_{LSB}}{RES} = \frac{2,44mV}{2,5\mu A} = 976\Omega$$



a. $G = R \Rightarrow \Delta G + G = R + \Delta R \Rightarrow \Delta R \ll E \dots \Delta R = 0,01\%$

b.



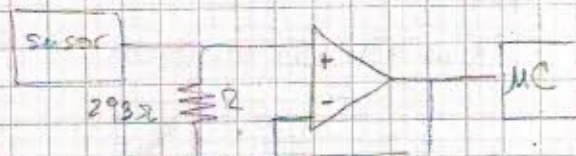
`VICIntSelect |= (1<<14); // habilita FIC`
`VICIntEnable |= (1<<14); // habilita int. ext EINT0 (14)`

Exam 2012/12/20

1. a.

$$V_{LSB} = \frac{V_{ref}}{2^n} = \frac{3V}{2^{10}} = 2,93mV; \quad RES = 0,1^\circ C \times 100\mu A / ^\circ C = 10\mu A$$

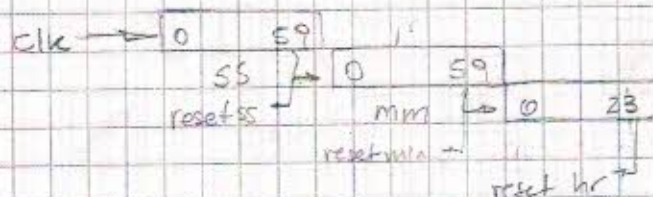
$$G = \frac{V_{LSB}}{RES} = \frac{2,93mV}{10\mu A} = 293\Omega;$$



$G = R \Rightarrow \Delta G + G = R + \Delta R \Rightarrow \Delta R \ll E \Rightarrow \Delta R = 0,1\%$

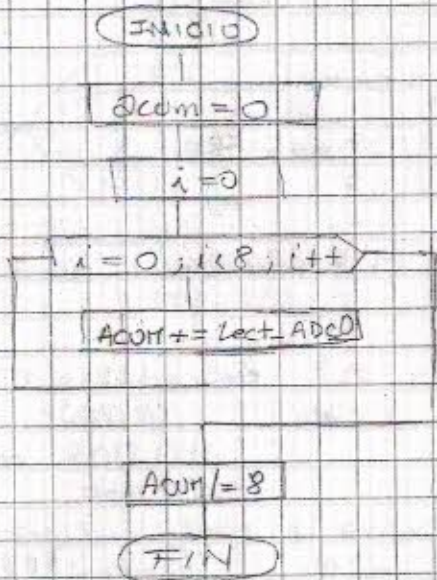
El buffer protege y separa los etapas.

b. Clock 1Hz \rightarrow 555; $clk = \frac{1,44}{(Ra + 2Rb) * C} \rightarrow C = 1\mu F \dots Ra = Rb = 480k\Omega$



d. IDEM a punto 2b, exam 2012/7/6

e. Diagrama de flujo del Servicio a la Interrupción



reset:

```

mov r3, #0 // acum
mov r4, #0 // 1 (limite)
mov r5, LECT // vector
  
```

otro:

```

ldrb r6, [r5], #1
add r3, r6
add r4, r4, #1
cmp r4, #8
bne otro

mov r7, r3, asr #3
strb r7, PROM.
  
```

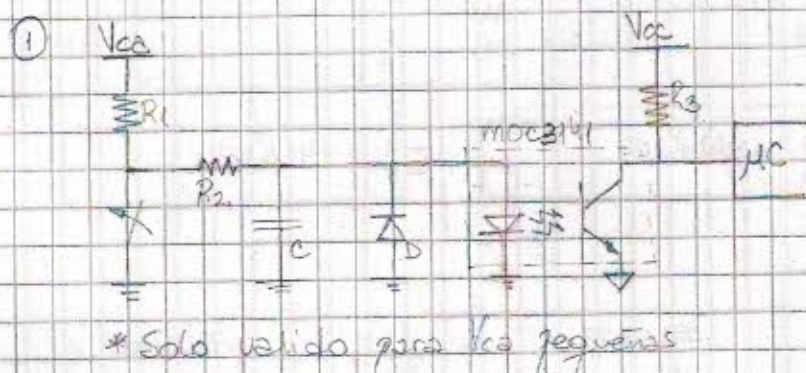
loop: b loop

```

LECT: .byte 0, 2, 4, 6, 8, 10, 12, 14
PROM: .byte 0x00
      .end
  
```


- ACONDICIONADORES DE SEÑAL (I/O)

ENTRADA:

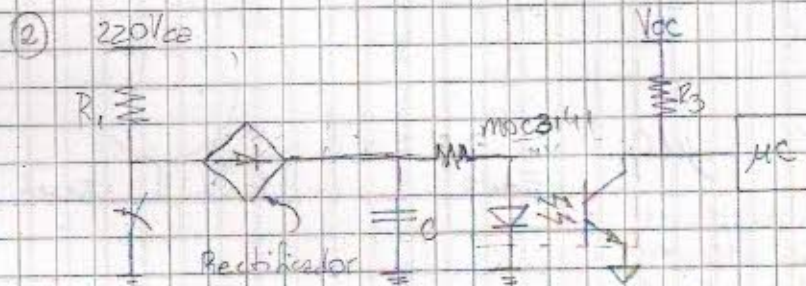


D → Proteje el pñlo de la tensión inversa de V_{ca} .

R_2 y C → Determina τ

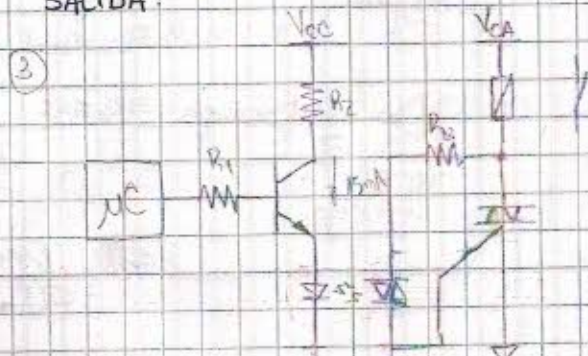
R_1 → Limita corriente de entrada y cortocircuito

R_3 → Pull up!



El rectificador soluciona el problema de la tensión inversa de V_{ca} .

SALIDA:



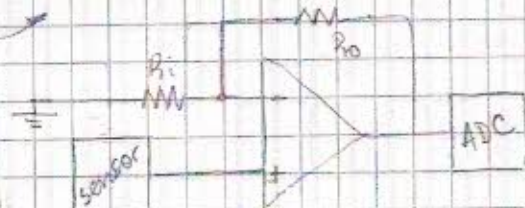
③ Adaptación con $V(t)$ en [A]



- ACONDICIONADOR DE SEÑAL (ADC)

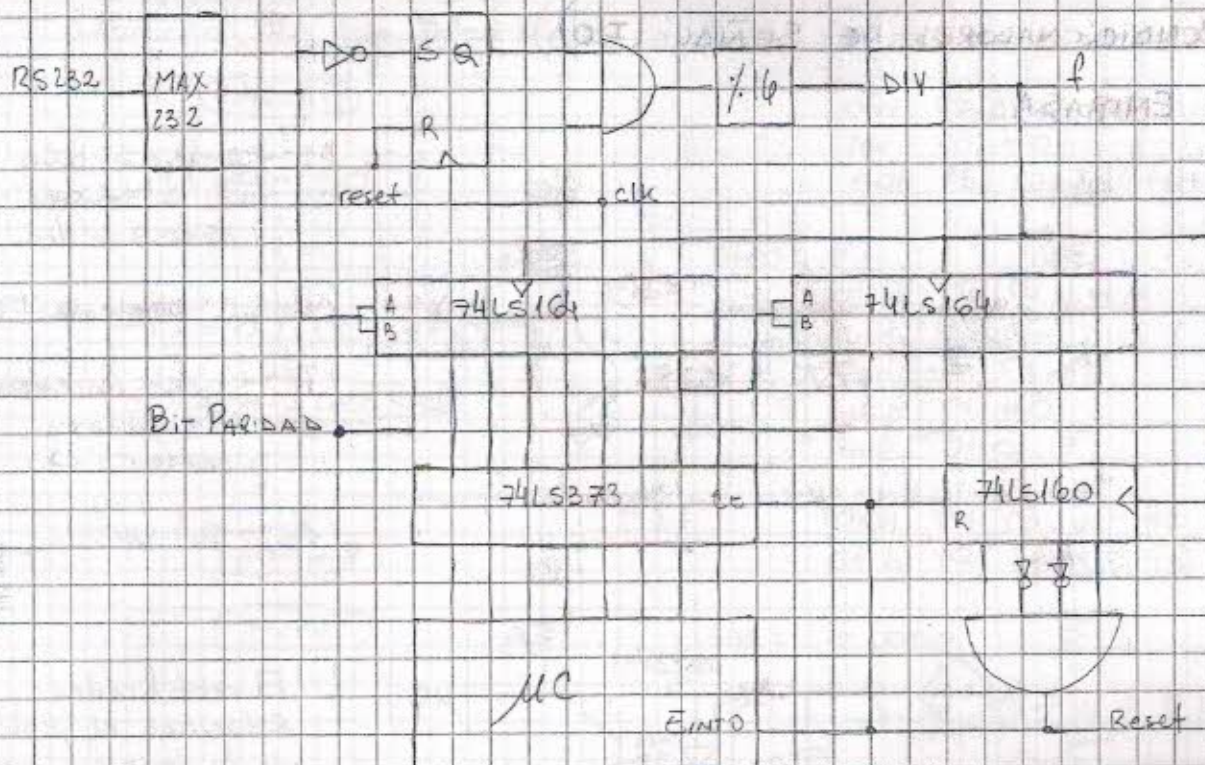
① Operacional: $G < 10$

② Amp Instrumentación: $G < 500$



por los
analog
en
operacionales

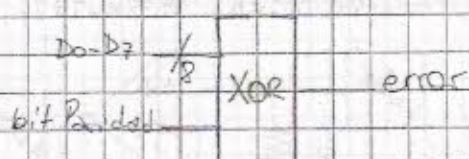
- UART DISCRETA (8PL)



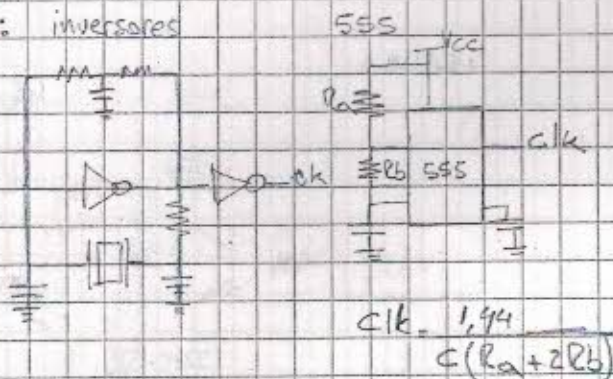
$$CLK = 1,8432 MHz$$

$$BaudRate = \frac{CLK}{16 \times DIV}$$

* Detección PARIDAD

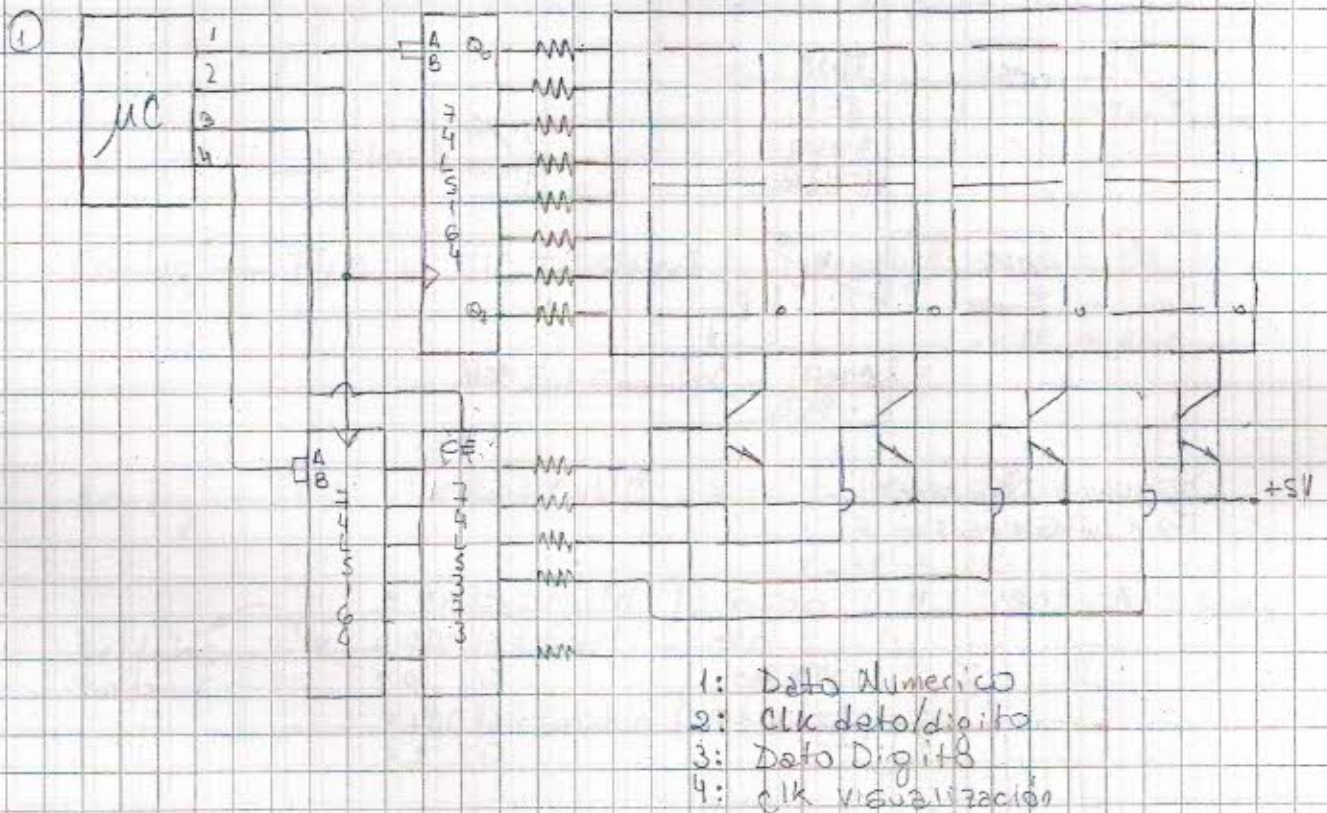


clock: inversores

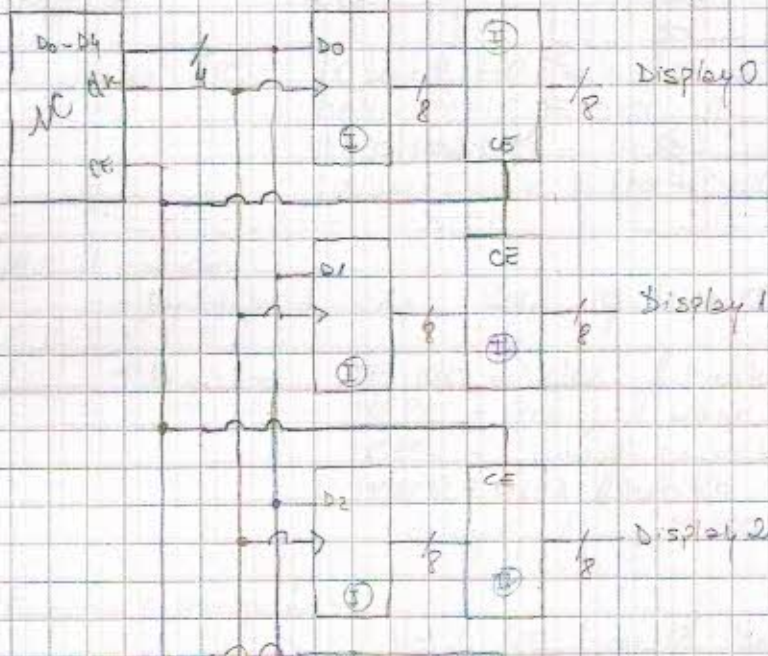


$$CLK = \frac{1,94}{C(R_A + 2R_B)}$$

- MANEJO DE DISPLAY 7 SEGMENTOS.



② 4 display sin MUX



③ Shift register:
74LS164

④ Latch:
74LS373

- ADQUISICIÓN DE VARIOS SENSORES.

1

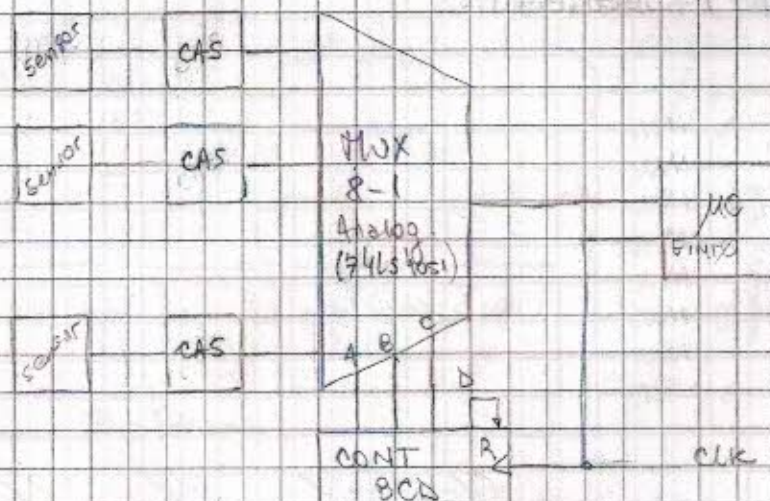
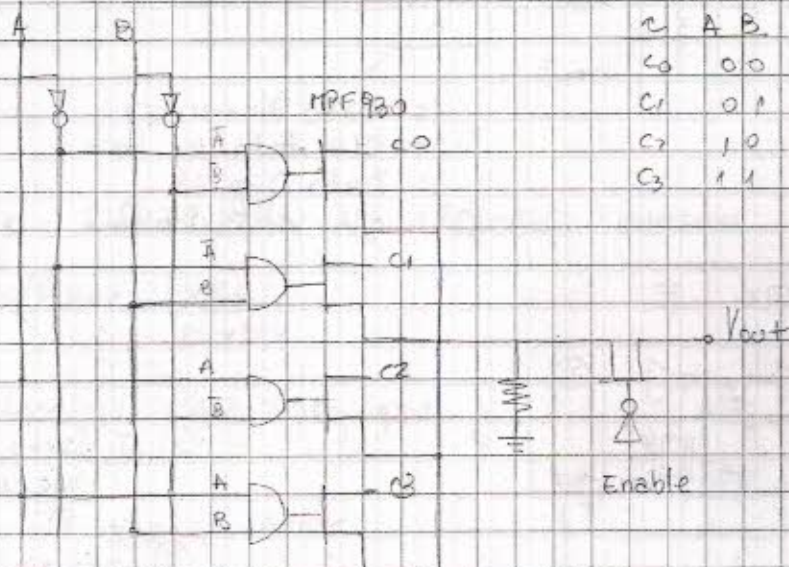


Diagrama Discreto
MUX Analógico.

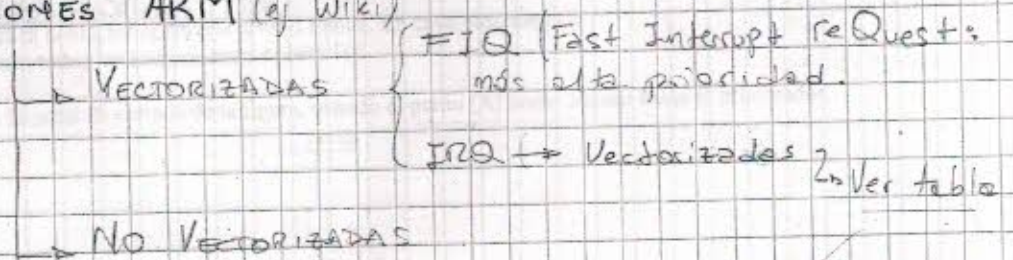


A	B
00	00
01	01
10	10
11	11

RESUMEN - TD2

C

- INTERRUPTIONES ARM (aj Wiki)



Config → FIQ: $VICIntSelect |= (1 \ll X);$ // $X=14 \rightarrow$ EXTI0
 $X=4 \rightarrow$ TIMER0
 $X=18 \rightarrow$ ADC

$VICIntEnable |= (1 \ll X);$

Config → IRQ: $VICVectCntl0 = X;$ // $X=14 \rightarrow$ EXTI0
 $X=4 \rightarrow$ TIMER0
 $X=18 \rightarrow$ ADC

$VICVectCntl0 |= 0x20;$ // Habilita Async Interrupt

* $VICVectAddr0 = IRQ_func();$

$VICIntEnable |= (1 \ll X);$

Se debe definir

`Void IRQ_func() attribute ((interrupt("IRQ"));`

• TIMER0

Configura VIC: $VICVectCntl0 = 4;$ // Fuente de interrupción
 $VICVectCntl0 = 0x20;$ // habilita fuente.
 $VICVectAddr0 = (long) irq_timer0;$ // apunta a la función
 $VICVectEnable |= (1 \ll 4);$ // habilita interrupción.

Inicialización:

- Prototipo función: `Void irq_timer0(void) attribute ((interrupt("IRQ"));`

- Registros:

- $TMR0 = \text{valor};$ // límite de cuenta.
- $T0PR = \text{pre};$ // valor prescaler.
- $T0PCR |= (\text{interrupt on match} | \text{reset on match});$
- $T0PCR \neq 0x01$ // habilita cuenta
- $\text{delay} = \frac{1}{f_{clk}} \times TMR0 \times T0PR$

Función Interrupción:

`Void irq_timer0(void)`
`{ if (TMR0 & T0PCR == 0x01)`