

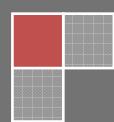
2015

Apunte de estudio para rendir el final

Técnicas Digitales II

Contiene una “Parte 1” referente a una introducción y desarrollo conciso teórico de los temas a tratar; y una “Parte 2” con la resolución de 25 exámenes finales.

Sebastián Azcurra



Prólogo

Bueno, antes que nada, espero que este apunte les sirva de guía para cuando preparen la materia para rendirla. Quiero aclarar que en realidad lo que voy a hacer es una transcripción en computadora, de un scaneo que hizo Yamil Taler, que se los pasó Alejandro Cáceres y que son de autoría de Ignacio Morales. Por razones de tiempo, no llegué a pasar tooooooodos los exámenes finales, pero con lo que hay, es bastante. Además, voy a sacar un poco de información de un libro que el Ing. Canali me pasó en pdf, que es "Analog-Digital Conversion" de Walt Kester, de Analog Devices (está disponible en la "wiki" de la cátedra para descargarlo capítulo por capítulo).

Este apunte lo pienso subir tanto en pdf como en doc, cosa que si quieren irlo complementando con la actualización de los exámenes o modificándolo, se pueda hacer sin problemas, y sea cada vez más útil. Por ejemplo, en la parte teórica falta todo lo referente a Assembler, y además que seguramente está LLENÍSIMO de errores, jaja.

Por otro lado, les recomiendo que antes de rendir esta materia, primero rindan el final de Electrónica Aplicada II, ya que casi todo lo que se estudia para el final de Técnicas Digitales II (exceptuando Assembler), es de Electrónica Aplicada II. El apunte que recomiendo y requeterecomiendo para sacar Electrónica Aplicada II es el de "LIANJOSHO" (pseudónimo del autor), ya que está completísimo y conciso, solo faltaría a ese apunte agregar un práctico de amplificadores clase "B", pero bueno, eso cuando vean esa materia.

Por último les comento que prácticamente esta materia, digitalizada, la preparé en 3 semanas y media, con un montón de contratiempos de por medio (pintar un futón, arreglar unas cuantas canillas que perdían, etc, etc, etc, del dpto., luego de volver de un mes en Ecuador tirado panza arriba, jaja..... Ah.... Sumado a que una noche dormí fuera del departamento porque me olvidé las llaves adentro.....plop!), o sea que con 2 semanas basta y resobra!

EXITOS A TODOS!

Sebastián Azcurra

Índice

Tópico	Página
Prólogo	1
Índice	2
Programa analítico de la materia	5
Parte 1: Introducción Teórica	7
Unidad 2: Conversión A/D (Analógico/Digital) y D/A (Digital/Analógico)	7
Teorema del Muestreo (Nyquist-Shannon)	7
Sistema de adquisición de señales	10
Acondicionamiento de señal	11
Muestreo	14
Contador	18
Registro de desplazamiento (Shift register)	19
Otros componentes	21
Otros circuitos	28
Conversores	30
Conversor A/D del microcontrolador ARM (LPC2114)	30
Conversores D/A, introducción	30
Conversor D/A de resistencias ponderadas	31
Conversor D/A en escalera R-2R	32
Conversores A/D, introducción	33
Conversor A/D Flash (paralelo)	34
Conversor A/D de aproximaciones sucesivas	36
Conversor A/D de rampa en escalera	37
Conversor A/D de pendiente simple	38
Conversor A/D de pendiente doble	41

Unidad 3: Microcontroladores	44
Arquitectura de los microprocesadores	44
Arquitectura Von Neumann	44
Arquitectura Harvard	45
Datos principales del µC LPC2114	46
Descripción de componentes y fundamentaciones teóricas	47
UART-RS232C	47
PC: Program Counter	49
PSR: Registro de Estado de Programa	49
Banderas de condición	50
VIC: Controlador Vectorizado de Interrupciones	51
Complemento a 2	51
Temporizador LM555	54
Tolerancias normalizadas de los resistores	56
Bancos de memoria del LPC2114	57
Parte 2: Resolución de exámenes finales	58
Examen final: 30 de agosto de 2007	59
Examen final: 20 de diciembre de 2007	63
Examen final: 28 de febrero de 2008	67
Examen final: 22 de mayo de 2008	71
Examen final: 24 de julio de 2008	75
Examen final: 09 de octubre de 2008	81
Examen final: 04 de diciembre de 2008	85
Examen final: 05 de febrero de 2009	89
Examen final: 12 de febrero de 2009	93
Examen final: 26 de febrero de 2009	101
Examen final: 03 de diciembre de 2009	106

Examen final: 04 de febrero de 2010	113
Examen final: 11 de febrero de 2010	118
Examen final: 25 de febrero de 2010	125
Examen final: 18 de noviembre de 2010	130
Examen final: 27 de julio de 2011	136
Examen final: 13 de octubre de 2011	141
Examen final: 24 de noviembre de 2011	147
Examen final: 15 de diciembre de 2011	154
Examen final: 22 de diciembre de 2011	161
Examen final: 24 de mayo de 2012	167
Examen final: 05 de julio de 2012	172
Examen final: 26 de julio de 2012	179
Examen final: 20 de diciembre de 2012	187

Programa analítico de la materia

El mismo es el actual que bajé de la página del departamento (23 de diciembre de 2014).



*Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento. Ingeniería Electrónica*

PROGRAMA ANALÍTICO DE: TÉCNICAS DIGITALES 2 (Plan 95/ adecuado 2006)

Nivel	Cuatrimestre	Código	Hs. semanales
4to	Anual		5

Correlatividades:

Para cursar:

Cursada: Informática 2 - Técnicas Digitales 1 - Electrónica Aplicada 1.

Aprobada: Química general - Física 2 -

Para rendir:

Aprobada: Informática 2 - Técnicas Digitales 1 - Electrónica Aplicada 1.

Estrategia Metodológica: Clases Teóricas, (Exposición del tema por parte del Docente). Clases Prácticas de aula, (El Docente expone la técnica a aplicar en ejercicios y problemas tipo y luego guía a los estudiantes en la resolución de los que se plantean a la clase).

Criterios de evaluación: Evaluación continua durante el curso mediante pruebas parciales. Evaluación final mediante examen integrador.

Contenidos:

UNIDAD 1: MUESTREO.

Concepto de muestreo. Teorema de Shannon. Circuitos de muestreo. Circuitos de retención. Tiempos de captura y retención. Criterio de selección de componentes. Ejemplos de diseño aplicados a problemas clásicos.

Duración: 2 Semanas

UNIDAD 2: CONVERSIÓN A/D Y D/A.

Convertidores d/a básicos. Discusión de los circuitos clásicos. Convertidores a/d. Rampa simple. Doble rampa y doble rampa modificado. Aproximaciones sucesivas, registro de aproximaciones sucesivas. Convertidor flash. Tiempos de conversión. Resolución. Discusión de las especificaciones (monotonidad, linealidad, etc.). Criterios de selección. Conexión de convertidores a/d y d/a en sistemas basados en microprocesadores. Procedimientos de adquisición de datos. Multiplexado analógico. Criterios de diseño de sistemas.

Duración: 8 Semanas

UNIDAD 3: MICROPROCESADORES.

UNIDAD 3.1

Introducción. Ciclos de búsqueda y de ejecución. Set de instrucciones. Modos de direccionamiento. Operandos. Descripción del microprocesador 8088. Arquitectura. Modos de operación. Bus de datos, direcciones y control.

Duración: 4 Semanas

Figura 1: Programa analítico de la materia, 2014 (1º parte) - (Cortesía Dpto. Electrónica, FRC, UTN – Descarga: 23/12/2014)

**UNIDAD 3.2.**

Operaciones lógicas, aritméticas, de movimiento de datos, de entrada / salida, etc. Estructura de programas. Instrucciones de salto incondicional y condicional. Salto a subrutinas. Uso del stack como estructura de datos. Interrupciones. Funcionamiento y operación de las máscaras.

Duración: 6 Semanas

UNIDAD 3.3.

Estructura de una máquina Von Neumann. Entrada / salida. Modos y tipos de implementación discreta. Periféricos: paralelo, timer, controlador de interrupciones, dispositivos para comunicación serial. Norma RS 232-C. Implementación. UART.

Duración: 4 Semanas

UNIDAD 3.4

Implementación de una máquina Von Neumann. Arquitectura de la PC XT. Descripción de los buses. Forma de operación. BIOS. Descripción del propósito y operación de sus rutinas. Criterio de diseño de rutinas de BIOS.

Duración: 2 Semanas

UNIDAD 3.5

Conexión de periféricas a la PC. Forma de operación de los buses. Acceso a memoria. Acceso a E/S. Operación de los controladores clásicos. Diseño de una placa típica.

Duración: 2 Semanas

UNIDAD 3.6

Introducción técnica al DOS. Estructura de disco, FAT, modos de operación . Distribución de la memoria. Carga de programas. Funciones de supervisión.

Duración: 2 Semanas

UNIDAD 4: MICROCONTROLADORES

Descripción y utilización.

Duración: 2 Semanas

Bibliografía:

Analog/digital conversion handbook - Analog Devices. (editado por Prentice Hall).

The iAPX 88 Handbook - Intel Corporation.

The PC XT Technical Description - IBM Corporation.

The DOS technical description - IBM Corporation

IBM ROM BIOS - Ray Duncan. Microsoft Press.

Peripheral products databook. - Intel Corporation.

Figura 2: Programa analítico de la materia, 2014 (2º parte) - (Cortesía Dpto. Electrónica, FRC, UTN – Descarga: 23/12/2014)

Parte 1: Introducción Teórica

Bueno, estos son conceptos básicos que luego se utilizarán para el desarrollo de los ejercicios de los finales.

Unidad 2: Conversión A/D (Analógico/Digital) y D/A (Digital /Analógico)

Teorema del Muestreo (Nyquist - Shannon)

La frecuencia de muestreo debe ser al menos **2 veces** la frecuencia a mostrar ($f_m \geq 2.f_{\max}$) para que sea posible recuperar la función original $A(t)$. “ f_m ” se denomina *Frecuencia de muestreo*.

A efectos prácticos se toma 5 veces la frecuencia de la señal ($f_m \approx 5.f_{\max}$) haciendo un sobremuestreo. De esta forma se logra una mejora en la calidad de la señal reconstruida .

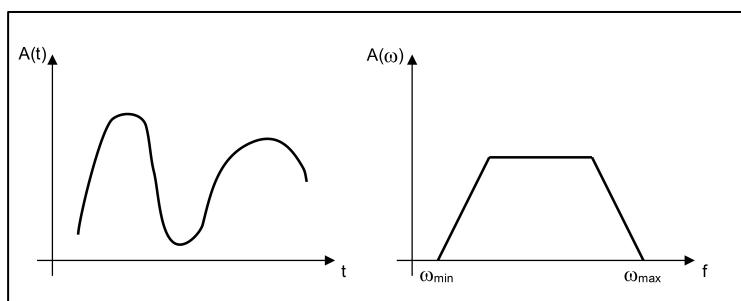


Figura 3: Señal en el dominio del tiempo y en el dominio de la frecuencia

Donde obviamente:

$$\omega_{\max} = 2\pi \cdot f_{\max} \quad (1)$$

Para evitar error en la toma de señal, se debe calcular el tiempo de adquisición o apertura (t_s). Dicho tiempo deberá ser tal que en el momento de mayor crecimiento de la señal, la misma no variará más de un cierto error (ε). Como conclusión, el tiempo de adquisición, muestreo, debe estar en el orden de la señal a muestrear.

El siguiente desarrollo es referente al error mencionado. Tomando como ejemplo la siguiente señal:

$$v = V \cdot \sin(\omega t) \quad (2)$$

Derivando respecto del tiempo:

$$\frac{dv}{dt} = V \cdot \omega \cdot \cos(\omega t) \quad (3)$$

Donde la máxima variación ocurre para $\cos(\omega t) = 1$, de esta forma se tiene que:

$$\frac{\Delta V}{t_s} = V \cdot \omega = V \cdot 2 \cdot \pi \cdot f \quad (4)$$

Que es la máxima pendiente de la señal.

Ahora si se define el **error** como el cociente entre la variación de tensión y el valor pico a pico, se tiene:

$$\epsilon = \frac{\Delta V}{2V} = \frac{\text{Fondo Escala}}{\text{Excursión Señal}} = \frac{V_{\max} \cdot 2 \cdot \pi \cdot f \cdot t_s}{2 \cdot V_{\max}} = \pi \cdot f \cdot t_s \quad (5)$$

Y este error debe ser menor o igual a medio bit, o sea:

$$\epsilon \leq \frac{1}{2} \cdot \text{bit} \quad (6)$$

Gráficamente se puede ver como:

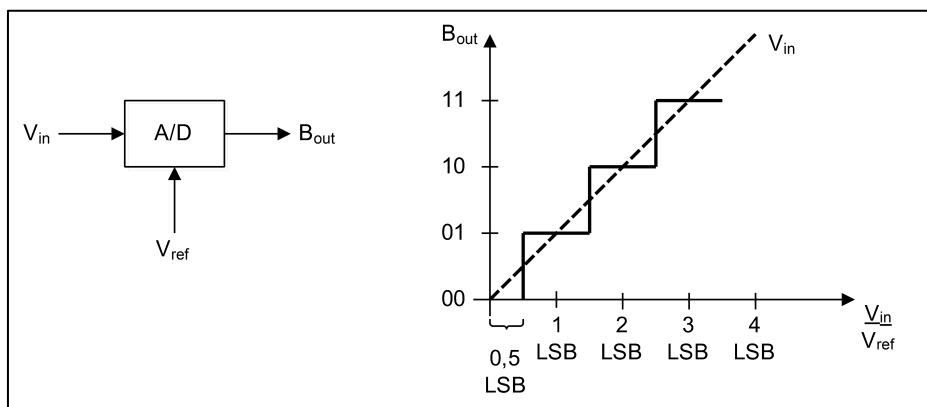


Figura 4: Esquema básico de conversión A/D y representación gráfica del error

En fórmulas:

$$\epsilon < \frac{1}{2^n \cdot 2} = \frac{1}{2^{n+1}} = \frac{1}{2 \cdot 2^n} = \frac{1}{2 \cdot \pi \cdot f \cdot 2^n} \Rightarrow t_s < \frac{1}{2 \cdot \pi \cdot f \cdot 2^n} \quad (7)$$

Por lo tanto se tendrá que:

$$V_{ref} \cdot (b_1 \cdot 2^{-1} + b_2 \cdot 2^{-2} + \dots + b_N \cdot 2^{-N}) = V_{in} \pm V_x \quad (8)$$

Donde:

$$-\frac{1}{2} V_{LSB} \leq V_x \leq \frac{1}{2} V_{LSB} \quad (9)$$

Finalmente, con un ejemplo donde $V_{ref} = 5[V]$ y un conversor de 4[bits]:

$$V_{ref} \cdot \left(\frac{b_3}{2^3} + \frac{b_2}{2^2} + \frac{b_1}{2^1} + \frac{b_0}{2^0} \right) = V_{in} \pm V_x \quad (10)$$

Y tomando los últimos 2 bits para calcular el error mayor, se tiene:

$$V_{ref} \cdot \left(\frac{1}{2^1} + \frac{0}{2^0} \right) = V_{in} \pm V_x \quad (11)$$

$$5[V] \cdot \left(\frac{1}{2} + 0 \right) = V_{in} \pm V_x \quad (12)$$

$$2,5[V] = V_{in} \pm V_x \quad (13)$$

Luego, para averiguar V_x .

$$V_x = \frac{V_{ref}}{2^n \cdot 2} = \frac{V_{ref}}{2^{n+1}} = \frac{5[V]}{2^{1+1}} = \frac{5[V]}{4} = 1,25[V] \quad (14)$$

Por lo que finalmente se tiene que:

$$2,5[V] = V_{in} \pm 1,25[V] \quad (15)$$

Sistema de adquisición de señales

Luego de algunos conceptos teóricos básicos, se comenzará a desarrollar la utilización práctica de los mismos.

Para comenzar, a continuación se muestra un diagrama en bloques de un sistema genérico de adquisición de señales:

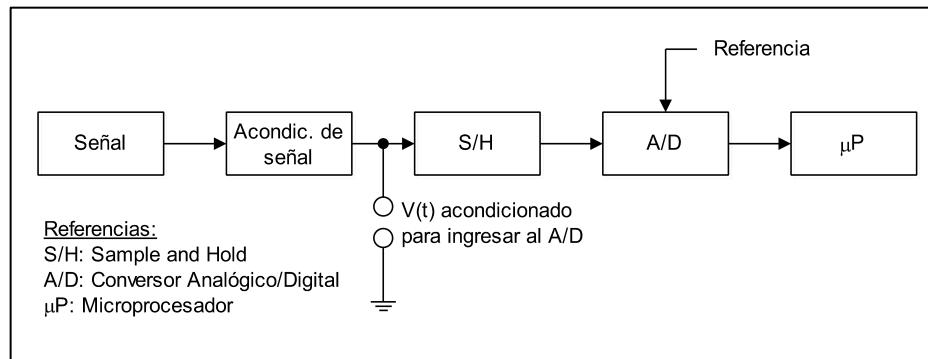


Figura 5: Diagrama en bloques genérico de un sistema de adquisición de señales

La señal que se pretende lograr, $V(t)$, es la del máximo rango del A/D, para aprovecharlo al máximo, logrando algo similar a la siguiente figura:

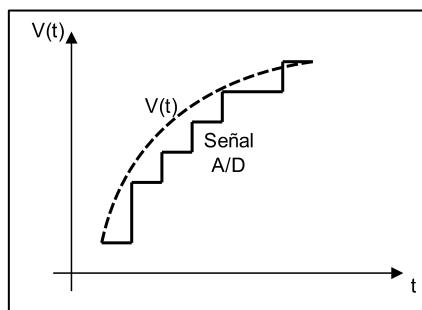


Figura 6: Ejemplo gráfico del porqué del acondicionamiento de la señal

El valor de cada escalón debe ser la aproximación inferior al valor posible. Esto se obtiene teniendo en cuenta la siguiente ecuación:

$$\text{Valor exacto escalón} = \frac{V_{ref}}{2^n} \quad (16)$$

Por otro lado, el error admitido en la cadena de medición es de 1[bit], o sea $\epsilon_{CM} = 1[\text{bit}] = 1\text{LSB}$, por lo tanto, descomponiendo este error en la cadena de medición resulta:

$$\epsilon_{CM} = 1\text{LSB} = \epsilon_{A/D} + \epsilon_{V_{ref}} + \epsilon_{Ganancia} \quad (17)$$

Y cada uno de ellos se describe a continuación:

$$\epsilon_{A/D} = \frac{1}{2}\text{LSB} \quad \text{ya que} \quad -\frac{1}{2}\text{LSB} < V_x < \frac{1}{2}\text{LSB} \quad (18)$$

$$\mathcal{E}_{V_{ref}} \leq \frac{1}{4} \cdot LSB = \frac{V_{ref}}{2^n \cdot 4} \quad (\text{debido al ripple}) \quad (19)$$

$\mathcal{E}_{Ganancia}$ se debe a la relación del error máximo admitido, sobre V_{ref} :

$$\mathcal{E}_{Ganancia} = \frac{\frac{1}{4} LSB}{V_{ref}} = \frac{\frac{1}{4} \cdot 2^n \cdot 4}{V_{ref}} = \frac{1}{2^n \cdot 4} \quad (20)$$

O en el mejor de los casos como la relación del error máximo admitido, sobre la V_{max} del amplificador:

$$\mathcal{E}_{Ganancia} = \frac{\frac{1}{4} LSB}{V_{amp}} = \frac{\frac{1}{4} LSB}{LSB \cdot n^o \text{cuentas}} = \frac{1}{4} \quad (21)$$

Como conclusión se puede mencionar que para reducir los errores anteriormente descriptos, se puede recurrir a:

- La utilización de baterías o reguladores estabilizados como el LM336, TL431, etc., para reducir el $\mathcal{E}_{V_{ref}}$.
- Utilizar fuentes de alimentación dedicadas y bien filtradas para los AOs de adaptación de señal y el A/D.

Acondicionamiento de señal

Ahora se desarrollan las etapas del sistema de adquisición de señales.

En esta etapa, básicamente se busca llevar a la señal a un valor óptimo para ser tomada por el A/D, logrando su mejor rendimiento aprovechándolo al máximo posible. Generalmente son circuitos donde se involucran AOs en diferentes configuraciones según sea el caso.

Los diferentes casos y soluciones que se pueden presentar son los siguientes:

Señal a digitalizar mayor al rango del A/D: en este caso se utiliza un divisor resistivo en conjunto con un AO en configuración *Seguidor de señal*, como muestra la siguiente figura:

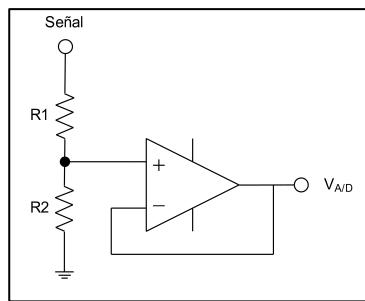


Figura 7: Acondicionamiento de señal cuando la señal a digitalizar es mayor al rango del A/D

Dicho circuito responde a la siguiente ecuación:

$$V_{A/D} = V_{señal} \cdot \frac{R2}{R1 + R2} \quad (22)$$

Necesidad de montar la señal sobre una continua: aquí se debe utilizar un AO en configuración *Sumador no inversor*, como se muestra a continuación:

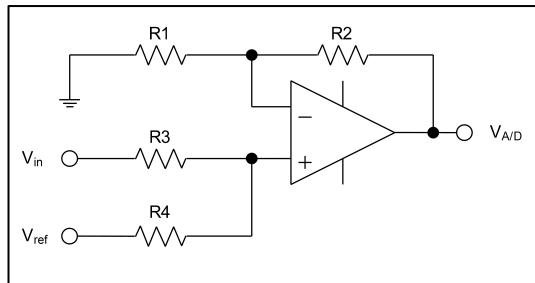


Figura 8: Acondicionamiento de señal cuando se necesita montar la señal sobre una continua

En este circuito es conveniente hacer las siguientes consideraciones:

$$\begin{aligned} \rightarrow & R1 // R2 = R3 // R4 \\ \rightarrow & R1 = R2 \quad \wedge \quad R3 = R4 = R \end{aligned} \quad (23)$$

De esta manera, la ecuación resultante del circuito es la siguiente:

$$V_{A/D} = \left(\frac{R2}{R3} \right) V_{in} + \left(\frac{R2}{R4} \right) V_{ref} = \left(\frac{R2}{R} \right) V_{in} + \left(\frac{R2}{R} \right) V_{ref} = \left(\frac{R2}{R} \right) (V_{in} + V_{ref}) \quad (24)$$

Digitalización de una señal diferencial de señal: para este último caso se utilizará un AO en configuración *Diferencial*, como muestra la siguiente figura:

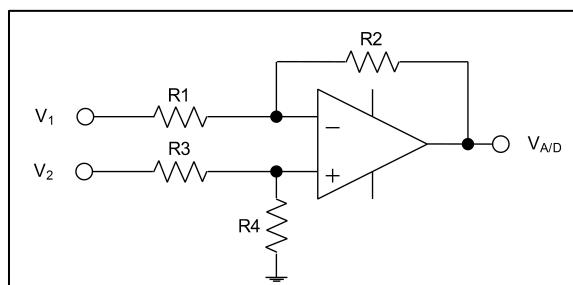


Figura 9: Acondicionamiento de señal para la digitalización de una señal diferencial de tensión

En donde se tienen las siguientes consideraciones:

$$\rightarrow \quad R1 = R2 \quad \wedge \quad R3 = R4 \quad (25)$$

Por lo tanto, la ecuación resultante del circuito es la siguiente:

$$V_{A/D} = \frac{R2}{R1} \cdot (V_2 - V_1) \quad (26)$$

Utilización de Amplificador de Instrumentación: el cual es otra posible solución, pero que se debe tener en cuenta la ganancia del mismo, por ejemplo en un INA101 varía de 1 a 1000[veces], y en un INA121 de 1 a 10000[veces], por lo tanto la correcta elección del AO de instrumentación adecuado para las condiciones específicas del circuito es fundamental. Se recomienda utilizar la ganancia en aproximadamente 500[veces], pero no es una regla estricta.

El circuito básico de un AO de instrumentación es el siguiente:

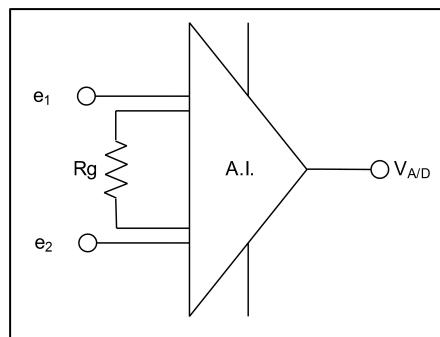


Figura 10: Circuito básico de un AO de instrumentación

Y la ecuación que gobierna este circuito básico es la siguiente:

$$\text{Ganancia} = 1 + \frac{50[k\Omega]}{Rg} \quad (27)$$

Consideraciones a tener en cuenta: en los circuitos vistos anteriormente son que:

- Las resistencias a utilizar deben ser de valores pequeños para reducir o evitar ruido térmico.
- Las tolerancias de las resistencias deben ser del 1[%] al 0,1[%].
- Los AO s a utilizar deben ser buenos, o sea, principalmente estables, como por ejemplo el OP07.
- La ganancia a utilizar en los AO s debe ser menor al 10[%] para evitar oscilaciones por corrimiento de fase. Los AO s de instrumentación son un caso particular.

- Los AO s a utilizar deben ser preferentemente multietapas.

Muestreo

Básicamente en esta etapa se pasa la señal de entrada, en el dominio del tiempo continuo, al dominio del tiempo discreto. Gráficamente se puede presentar de la siguiente manera:

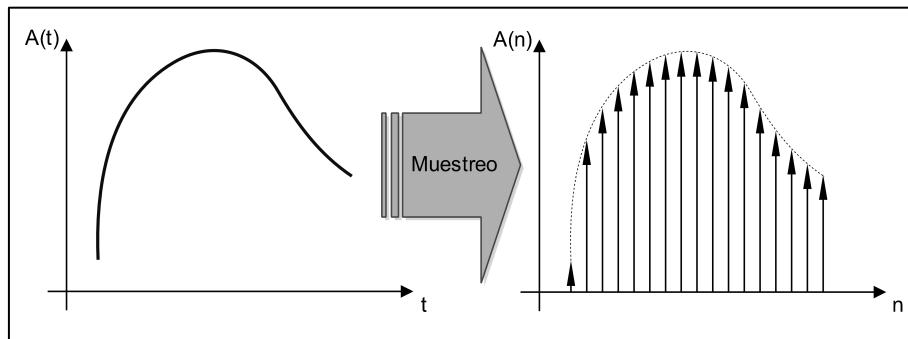


Figura 11: Ejemplo descriptivo del muestreo

Se debe tener en cuenta que cuando se trabaja con frecuencias elevadas, o sea por encima de las audiofrecuencias, se debe utilizar un S/H (Sample & Hold), ya que si la señal de entrada tiene una alta variación, el conversor no llegará a responder en su velocidad de conversión.

Para tener una introducción al tema, a continuación se muestra un modelo básico de muestreo:

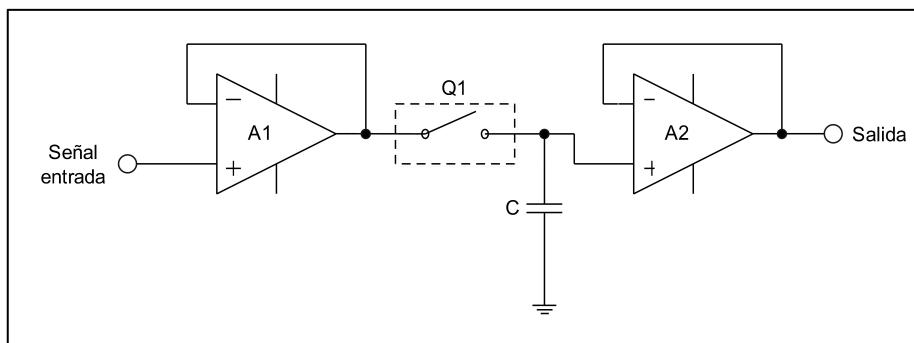


Figura 12: Modelo básico de muestreo

En el modelo anterior se utilizan 2 AO s en configuración Seguidor con las siguientes finalidades:

- “A2” evita que “C” se descargue por la salida.
- “A1” hace que la impedancia de la fuente de entrada sea cero, independizando la carga de “C”, y por ende la constante de tiempo dependerá exclusivamente de “ R_{ON} ” y “C”, de la siguiente manera:

$$\tau = R_{ON} \cdot C \quad (28)$$

El diagrama temporal del circuito anterior es el siguiente:

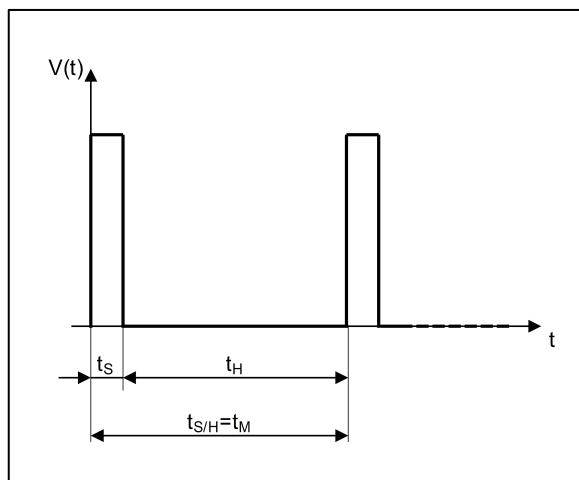


Figura 13: Diagrama temporal explicativo del modelo básico de muestreo

Por lo tanto:

$$t_{S/H} = t_M = t_S + t_H \quad (29)$$

Donde:

- $t_{S/H} = t_M$: es el tiempo de *sample & hold* o tiempo de *muestreo*.
- t_S : es el tiempo de *sample* o tiempo de *muestra*.
- t_H : es el tiempo de *hold* o tiempo de *retención*.

Y la relación entre t_S y t_H es la siguiente:

$$t_S \approx (0,1 a 1).t_H \quad (30)$$

Ahora, llevando lo anterior a un ejemplo práctico, se muestra el siguiente diagrama utilizando un transistor V-MOS como llave ideal:

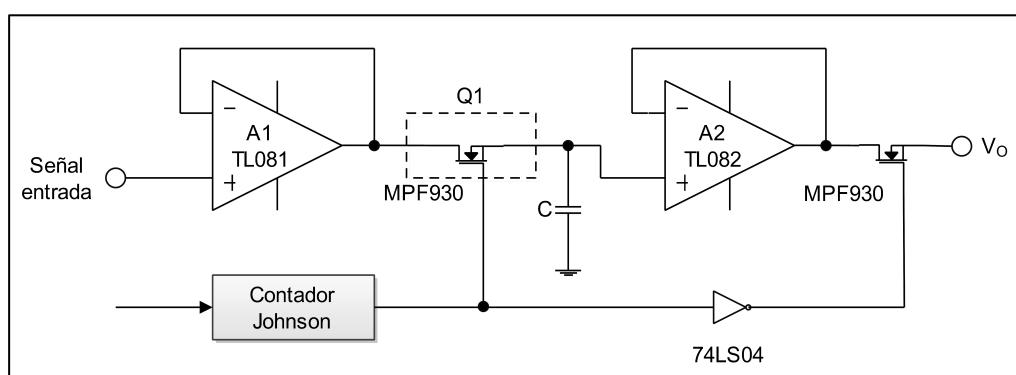


Figura 14: Circuito práctico de muestreo

Aquí el transistor tiene un tiempo de encendido $t_{ON} \ll 10[ps]$, y una $R_{ds-ON} \ll 0,1[\Omega]$, por lo que se aproxima bastante a una llave ideal.

Como señal de control de dicha llave se hace uso de un contador Johnson, a una frecuencia de $1[GHz]$, y una relación de pulsos 1:1000, por lo tanto:

$$t_s = 1[ns] \quad (31)$$

$$t_H = 1[\mu s] \quad (32)$$

Por lo tanto, para tener una precisión de $0,01[\%]$, se deberá tener un $t_s = 9.\tau$, de esta forma se obtiene que:

$$t_s = 9.\tau = 9.(R_{ds-ON} \cdot C) + t_{ON-MOSFET} \quad (33)$$

Muestreo de varias señales: para esta finalidad se utiliza un solo conversor A/D con un multiplexor analógico anteriormente. Para ver esto gráficamente, a continuación se presenta un esquema con un multiplexor analógico (4051) de 8 canales, asociado con un filtro pasa bajos (aunque es a modo teórico, ya que a modo práctico no se utiliza), un S/H y un A/D:

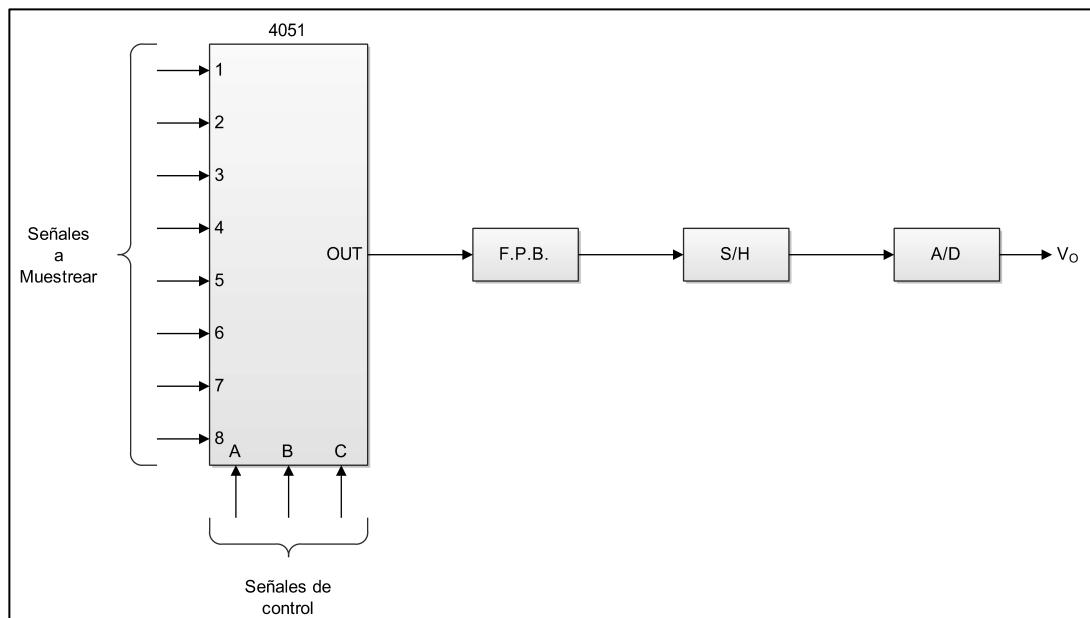


Figura 15: Ejemplo de muestreo de 8 señales

A continuación se muestran las configuraciones analógicas de diferentes multiplexores analógicos.

Multiplexor analógico de 2 canales

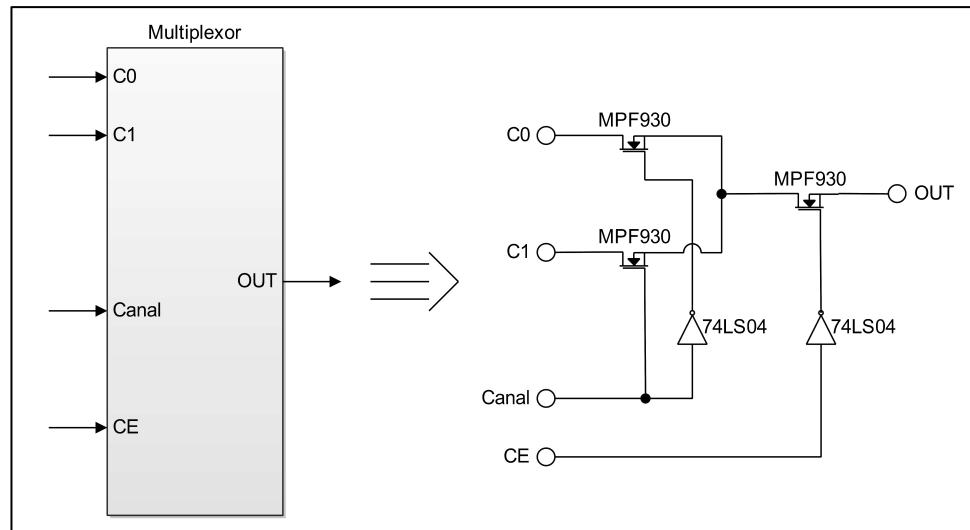


Figura 16: Configuración analógica de un multiplexor analógico de 2 canales

Su funcionamiento es el siguiente:

Canal	CE	
	1	0
0	0	C0
1	0	C1

Multiplexor analógico de 4 canales

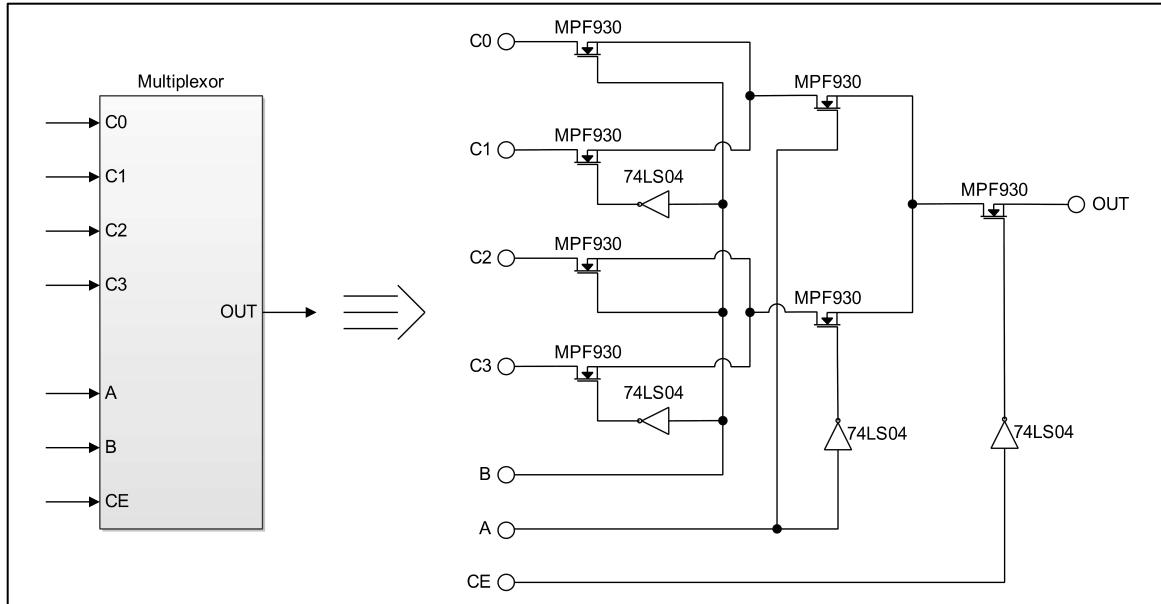


Figura 17: Configuración analógica de un multiplexor analógico de 4 canales

Su funcionamiento es el siguiente:

A	B	CE	
		0	1
0	0	C3	0
0	1	C2	0
1	0	C1	0
1	1	C0	0

Interfaz periférica programable, PPI 8255: ésta es una interfaz de propósitos generales. Su forma general es la siguiente:

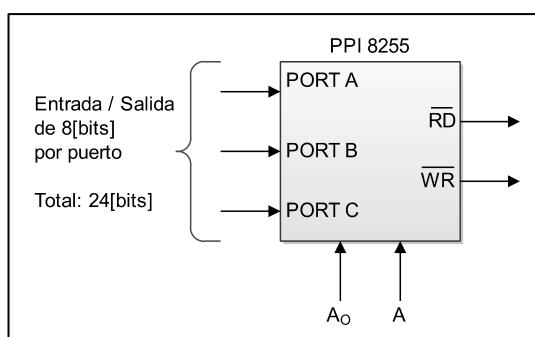


Figura 18: Representación de un PPI 8255

Posee:

- 2[bits] para Lectura/Escritura.
- 2[bits] para direccionamiento (PORT).

Contador

Es el circuito secuencial más sencillo de obtener, compuesto de elementos de memoria y lógica combinacional.

Existen de dos tipos:

- Asíncronos: los FFs (Flip-Flops) no están sincronizados por los pulsos del reloj.
- Síncronos: los pulsos de reloj se aplican simultáneamente a todos los FFs. Hace uso de las entradas síncronas (J, K, D, S, R) para predeterminar el estado futuro del contador.

Registro de desplazamiento (Shift register)

Es una conexión en cascada de una serie de FFs. Por cada pulso de reloj, la información se desplaza de un FF a otro.

Existen dos tipos: *Serie/Paralelo* y *Serie/Serie*, los cuales se describen a continuación.

Shift register Serie/Paralelo: a continuación se muestra un esquema genérico:

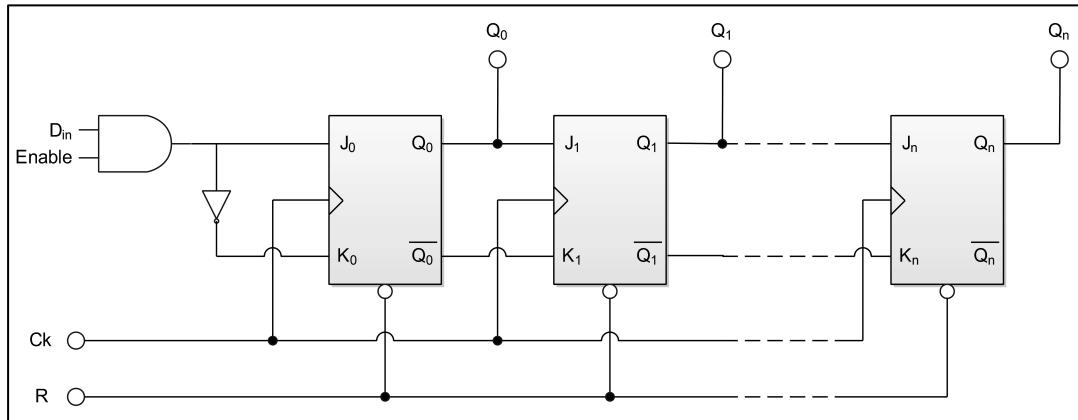


Figura 19: Shift register Serie/Paralelo

Shift register Serie/Serie: a continuación se muestra un esquema genérico:

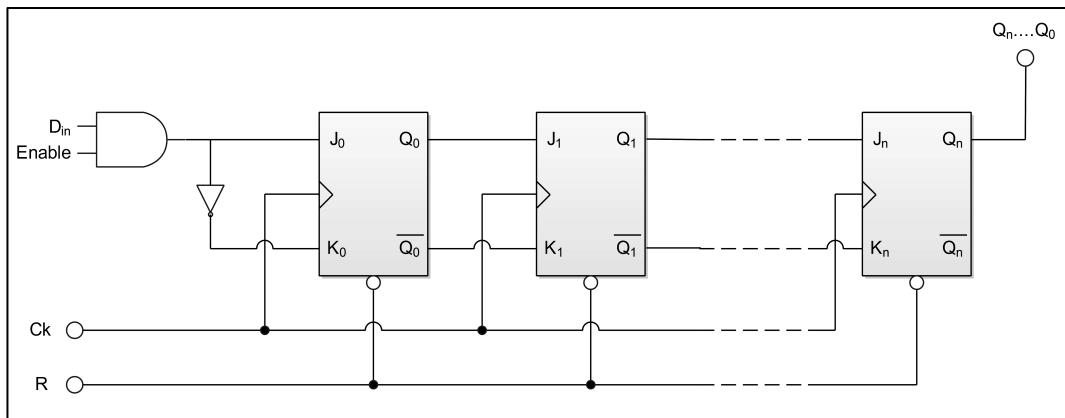


Figura 20: Shift register Serie/Serie

Contador Johnson: a continuación se muestra un esquema genérico:

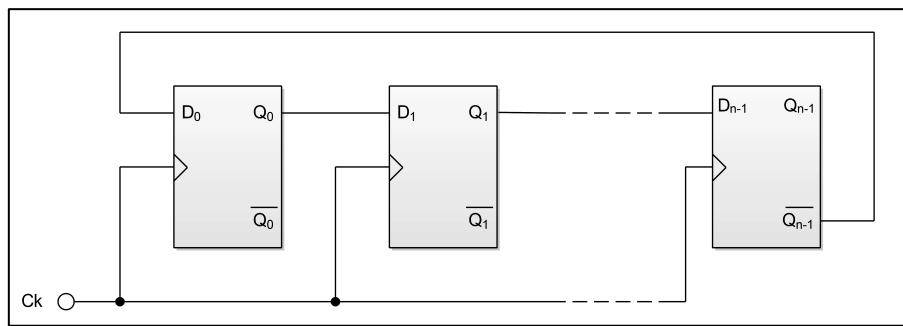


Figura 21: Contador Johnson genérico

Genéricamente se denomina que es de módulo $M = 2 \cdot n$, así por ejemplo para $n = 2 \Rightarrow M = 4$, o para $n = 3 \Rightarrow M = 6$.

A continuación se muestra un diagrama de tiempos de un contador Johnson de $n = 2$, o sea módulo $M = 4$:

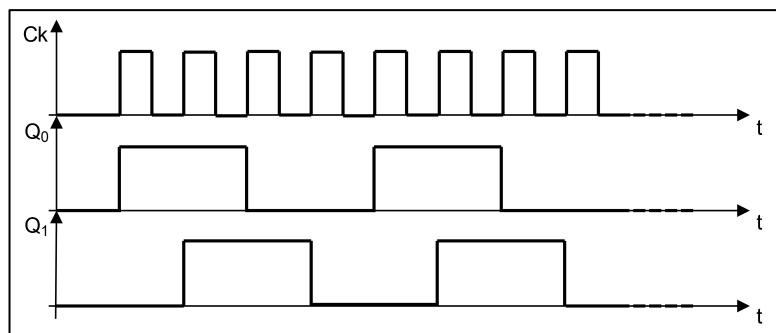


Figura 22: Diagrama de tiempos de un contador Johnson $M=4$

Otros componentes

Aquí se enumerarán algunos otros componentes y circuitos necesarios según sea el caso. Algunos ya se vieron en el presente apunte, y otros se verán más adelante. La información es provista por fabricantes y se mostrarán las características principales de los mismos.

Compuerta AND

- De 2 entradas: 74LS08 (Cortesía de Fairchild SemiconductorTM).

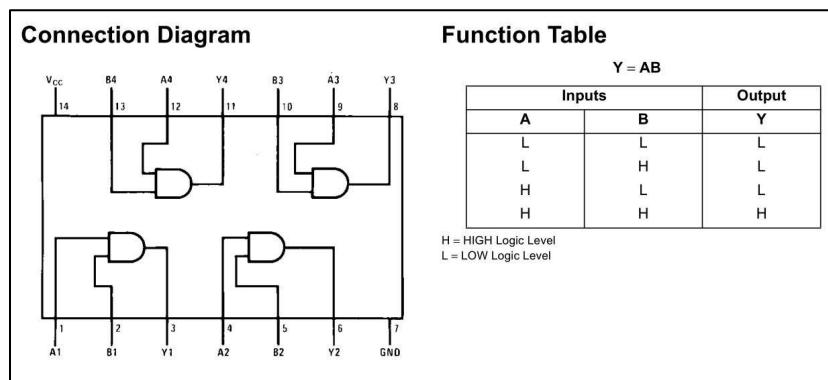


Figura 23: Diagrama de conexión y Tabla de función del 74LS08 (Cortesía de Fairchild SemiconductorTM)

- De 4 entradas: 74LS21 (Cortesía de Hitachi, Ltd., y Texas Instruments).

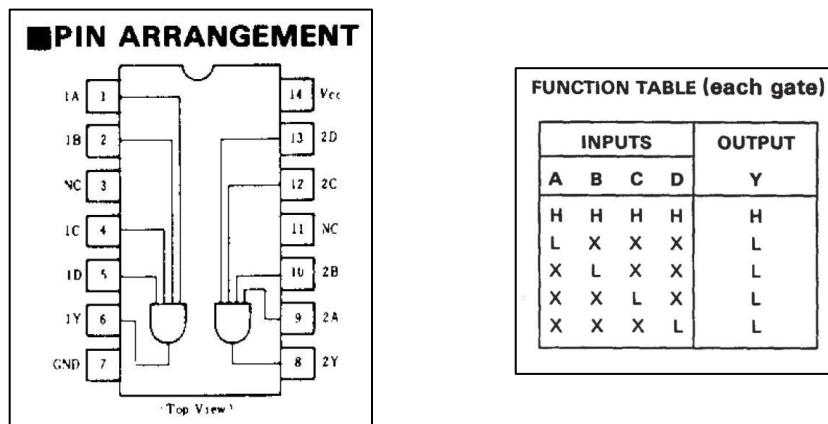


Figura 24: Diagrama de conexión (Cortesía de Hitachi, Ltd.) y Tabla de función (Cortesía de Texas Instruments) del 74LS21

Inversor

- Común: 74LS04 (Cortesía de On Semiconductor).

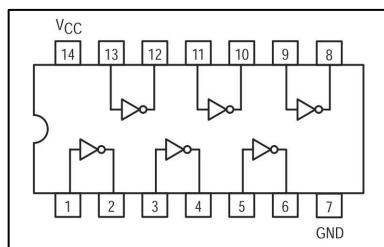


Figura 25: Diagrama de conexión del 74LS04 (Cortesía de On Semiconductor)

- Trigger Schmitt: 74LS14 (Cortesía de Freescale Semiconductor).

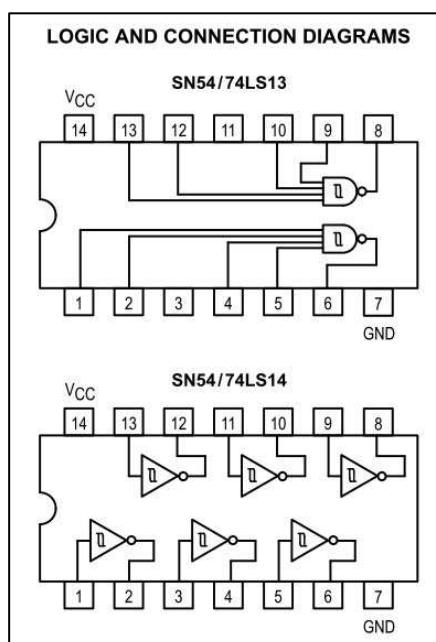


Figura 26: Diagrama de lógica y conexión del 74LS14 (Cortesía de Freescale Semiconductor)

Amplificador operacional (AO)

- Común: OP07 (Cortesía de Analog Device).

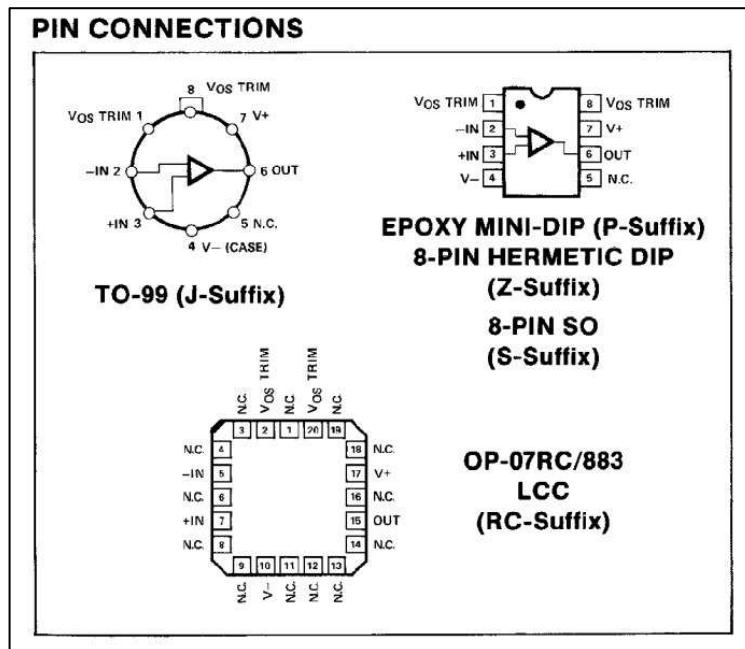


Figura 27: Diagrama de conexión del OP07 (Cortesía de Analog Device)

- Instrumentación: INA115 (Cortesía de Burr-Brown®).

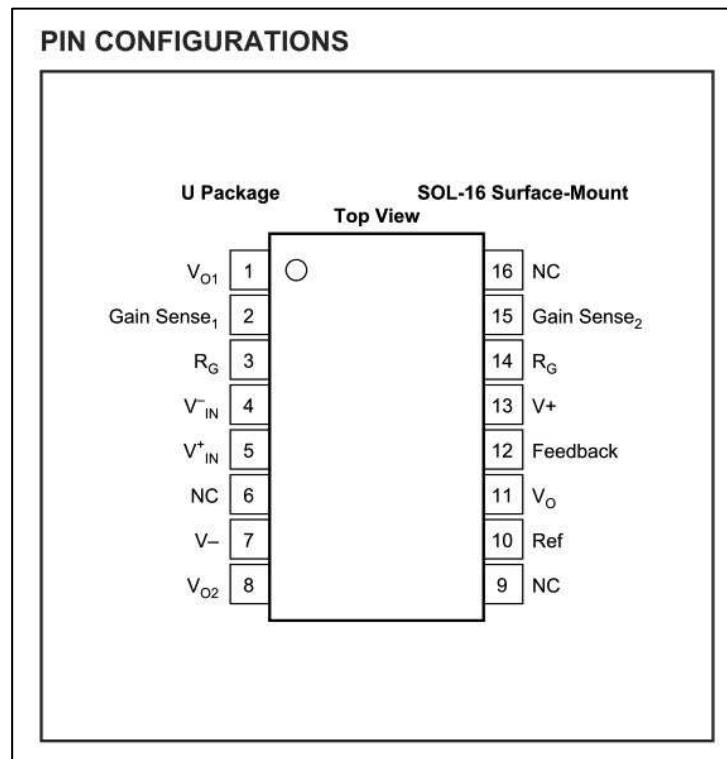


Figura 28: Diagrama de conexión del INA115 (Cortesía de Burr-Brown®)

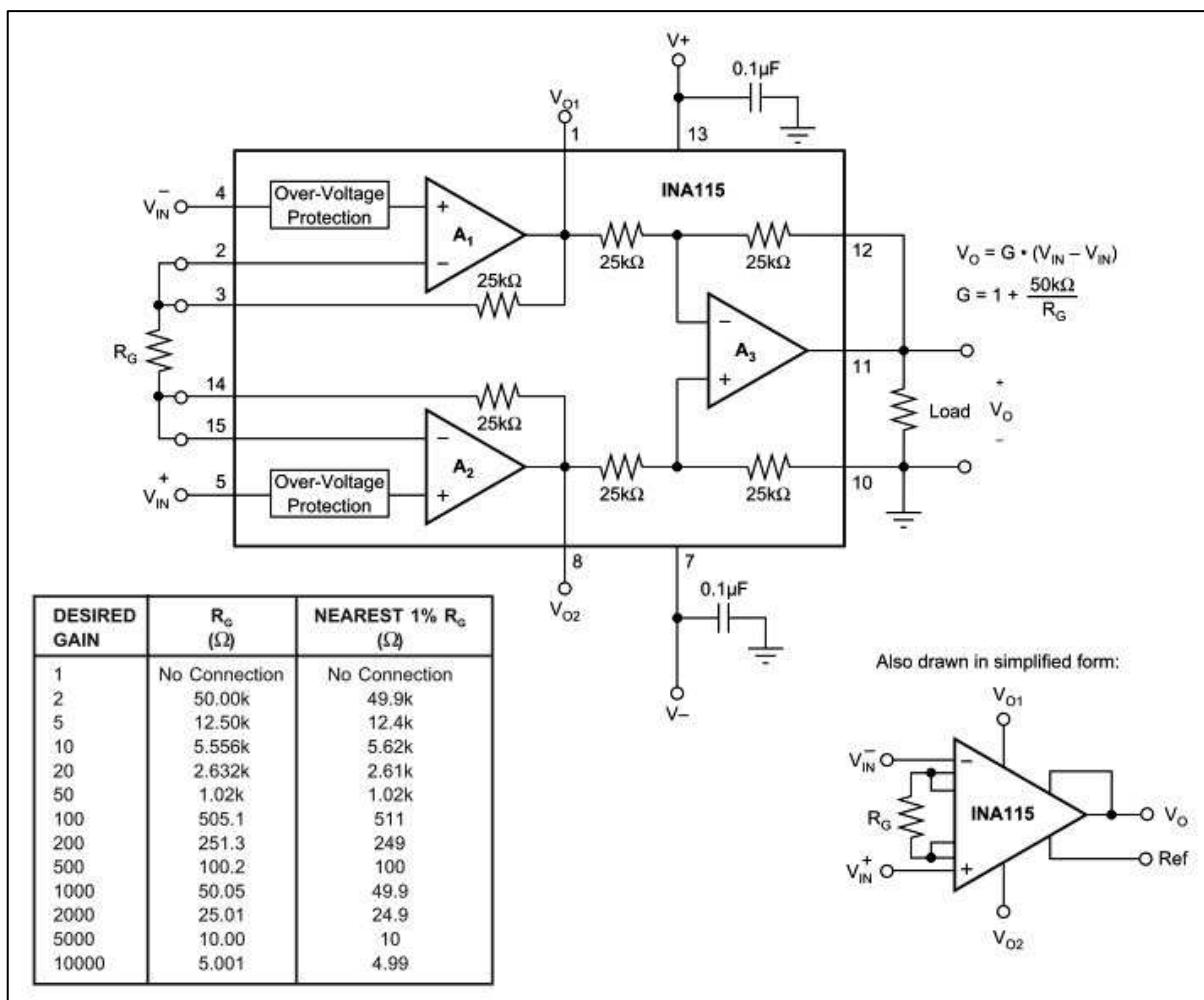


Figura 29: Esquema de conexión básico, Datos de configuración y Diagrama interno del INA115 (Cortesía de Burr-Brown®)

Nota al margen y opinión personal, quienes deban rendir o estén cursando Electrónica Aplicada II, les recomiendo que se tomen un tiempo y lean detenida y detalladamente la hoja de datos de este componente, es sumamente completa, académica y por sobre todo tiene muchos ejemplos de aplicación.

Transistor MOSFET

- MPF930 (Cortesía de Motorola). Se activa con un “1” en la compuerta.

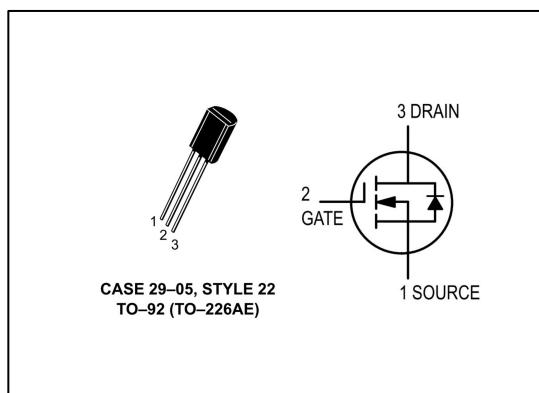


Figura 30: Muestra del encapsulado TO-92 y Diagrama de conexión del MPF930 (Cortesía de Motorola)

Shift Register

- 74LS164 (Cortesía de Fairchild SemiconductorTM).

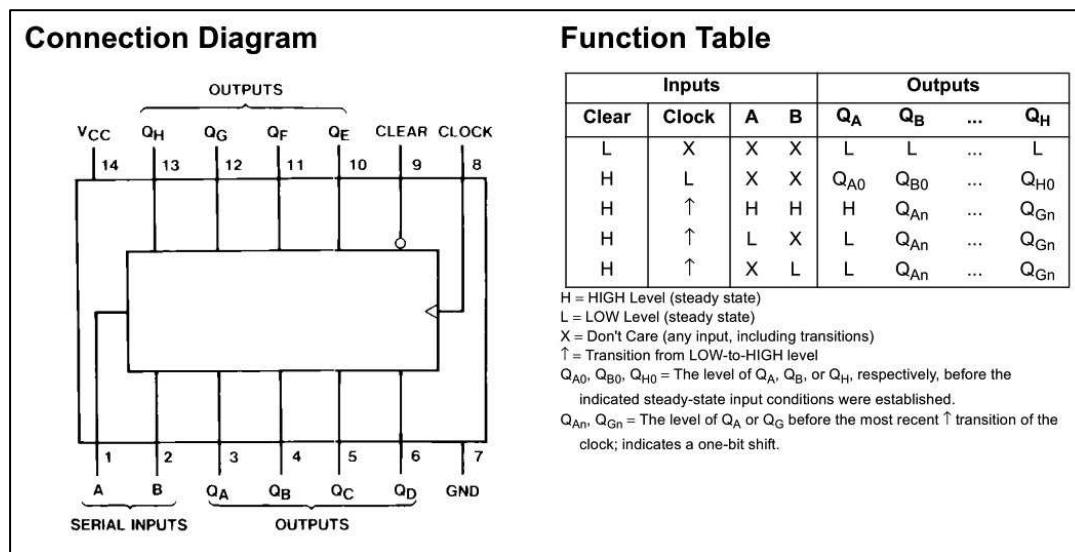


Figura 31: Diagrama de conexión y Tabla de funcionamiento del 74LS164 (Cortesía de Fairchild Semiconductor™)

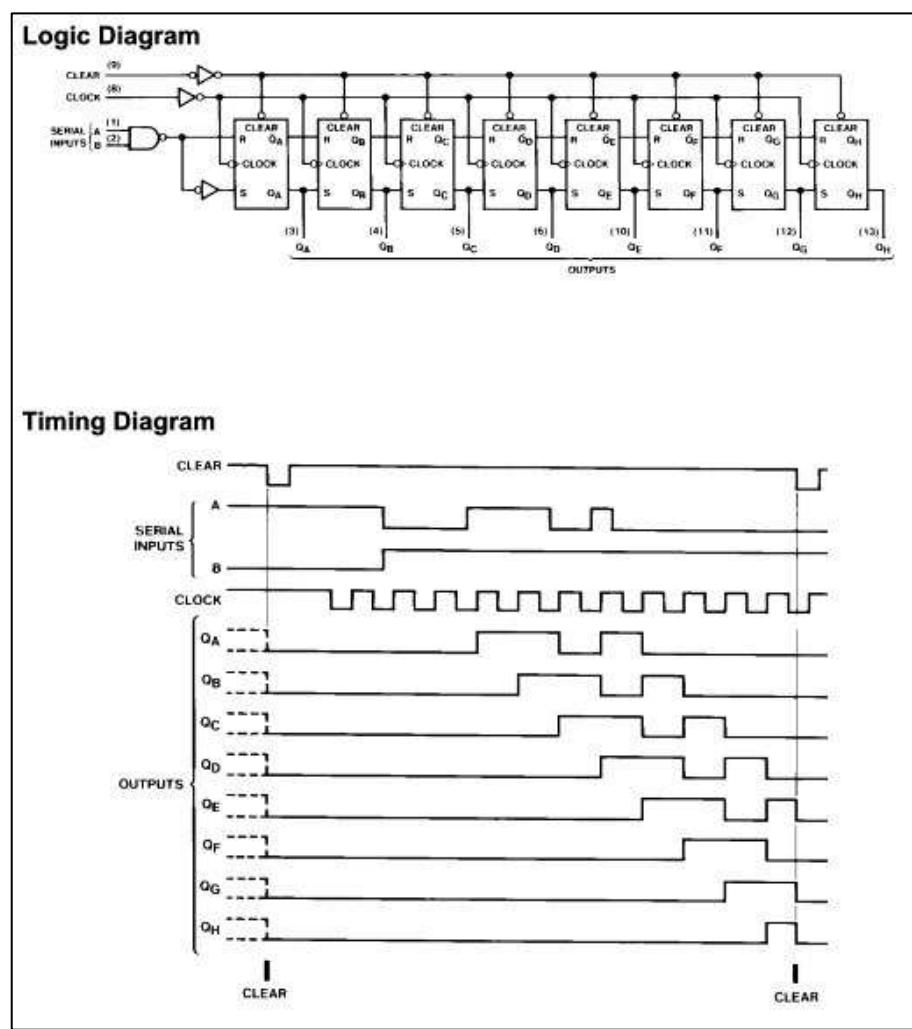


Figura 32: Configuración interna y lógica, y Diagrama de tiempos del 74LS164 (Cortesía de Fairchild Semiconductor™)

Interfaz periférica programable (PPI)

- 8255A (PPI) (Cortesía de AMD).

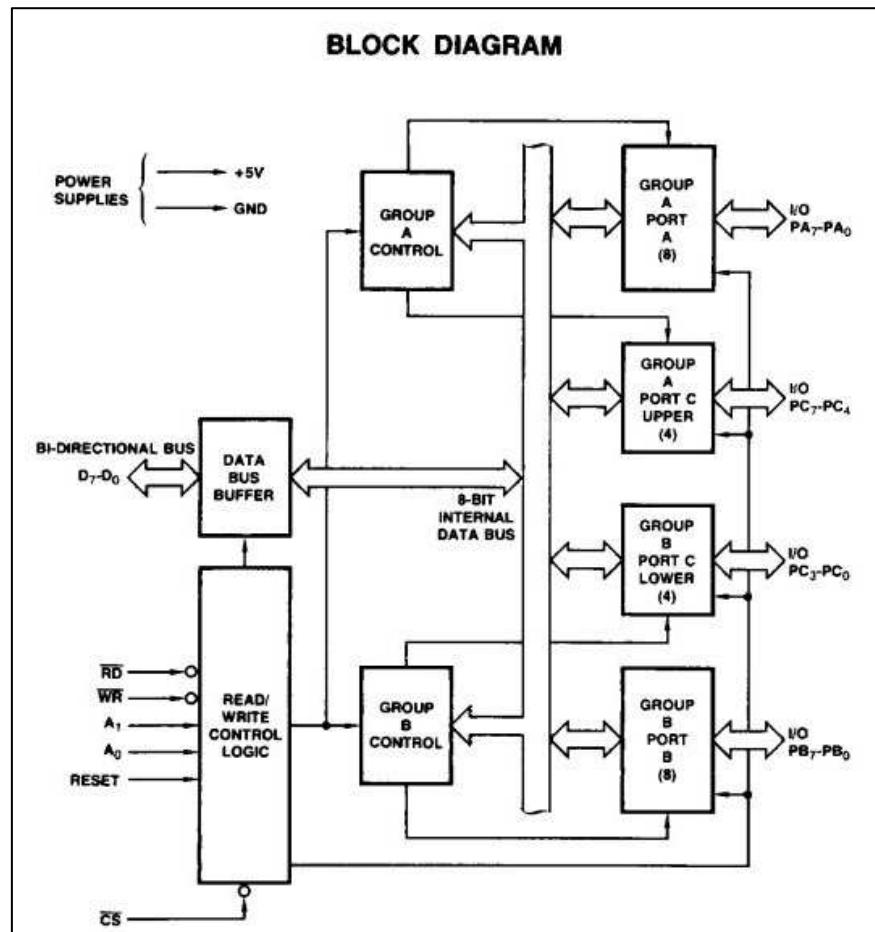


Figura 33: Diagrama en bloques del 8255A (Cortesía de AMD)

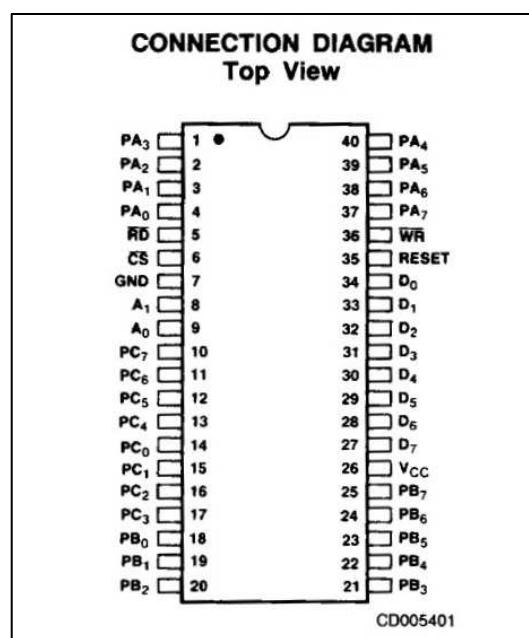


Figura 34: Diagrama de conexión del 8255A (Cortesía de AMD)

Optoacopladores

- Con transistor: CNY17 (Cortesía de Vishay Semiconductors).

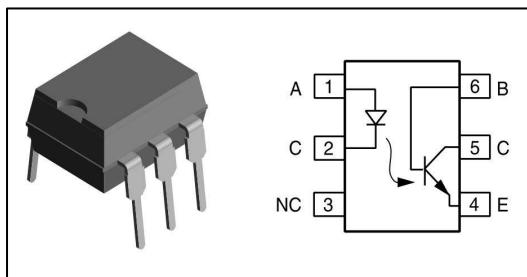


Figura 35: Presentación del encapsulado y Diagrama de conexión del CNY17 (Cortesía de Vishay Semiconductors)

- Con tiristor: MOC3011 (Cortesía de Texas Instruments).

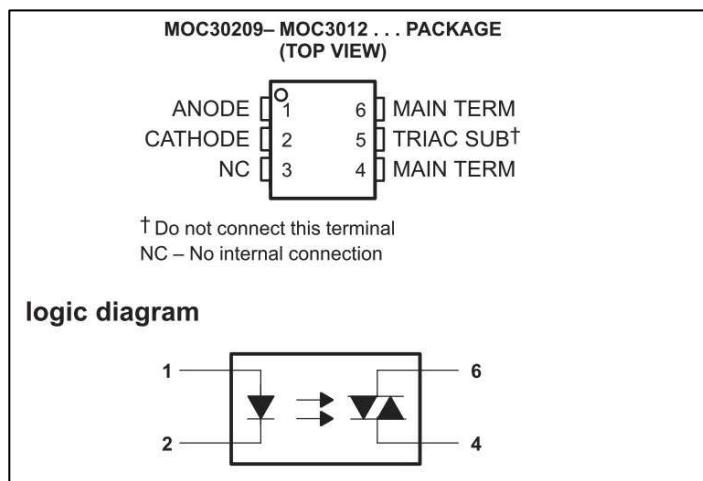


Figura 36: Diagrama de conexión y lógica del MOC3011 (Cortesía de Texas Instruments)

Tiristor

- BTA12 (Cortesía de ST).

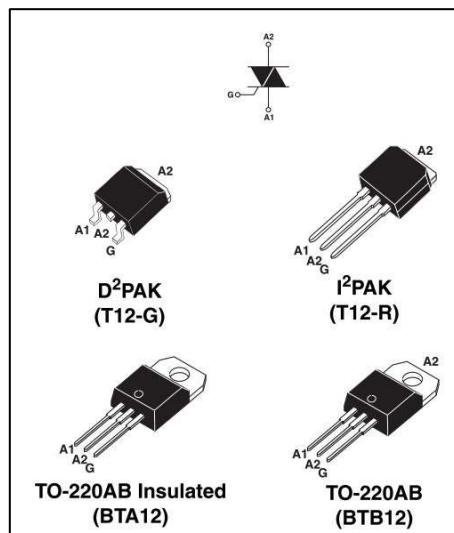


Figura 37: Diagrama de conexión y presentación de encapsulados del BTA12 (Cortesía de ST)

Timer

- LM555 (Cortesía de National Semiconductor).

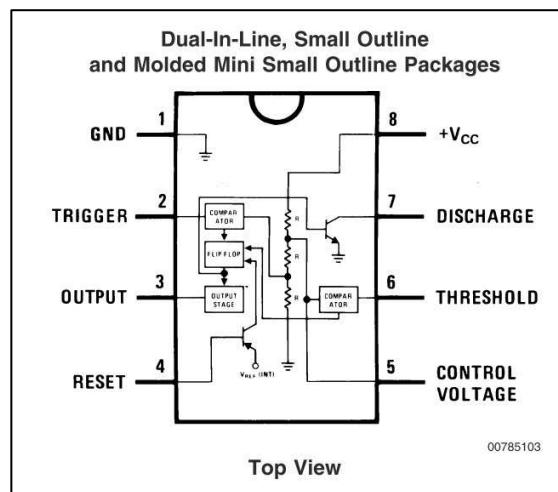


Figura 38: Diagrama de conexión y configuración interna del LM555 (Cortesía de National Semiconductor)

Otros circuitos

A continuación se verán algunas configuraciones o circuitos a utilizar:

Con salidas optoacopladas

- Con optoacoplador a transistor

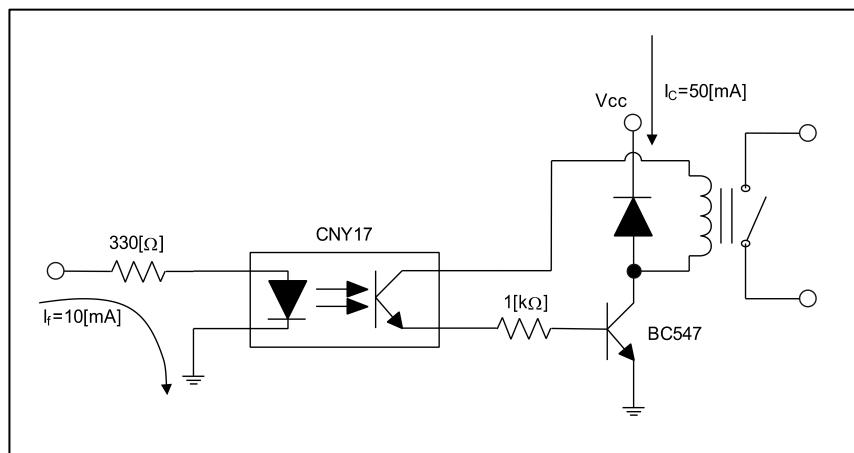


Figura 39: Aplicación de circuito con optoacoplador a transistor

- Con optoacoplador a tiristor

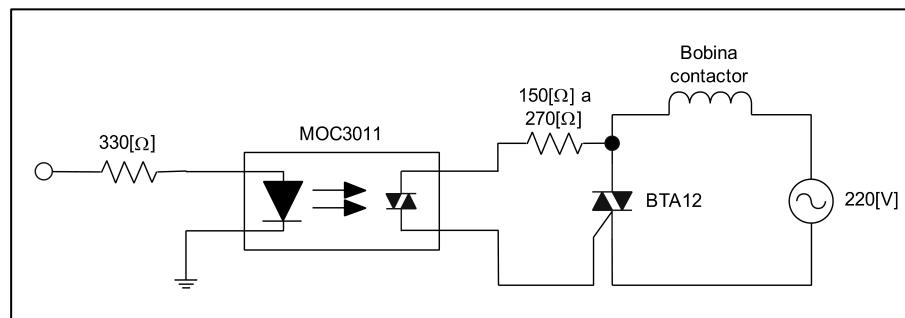


Figura 40: Aplicación de circuito con optoacoplador a tiristor

Arranque estrella-triángulo

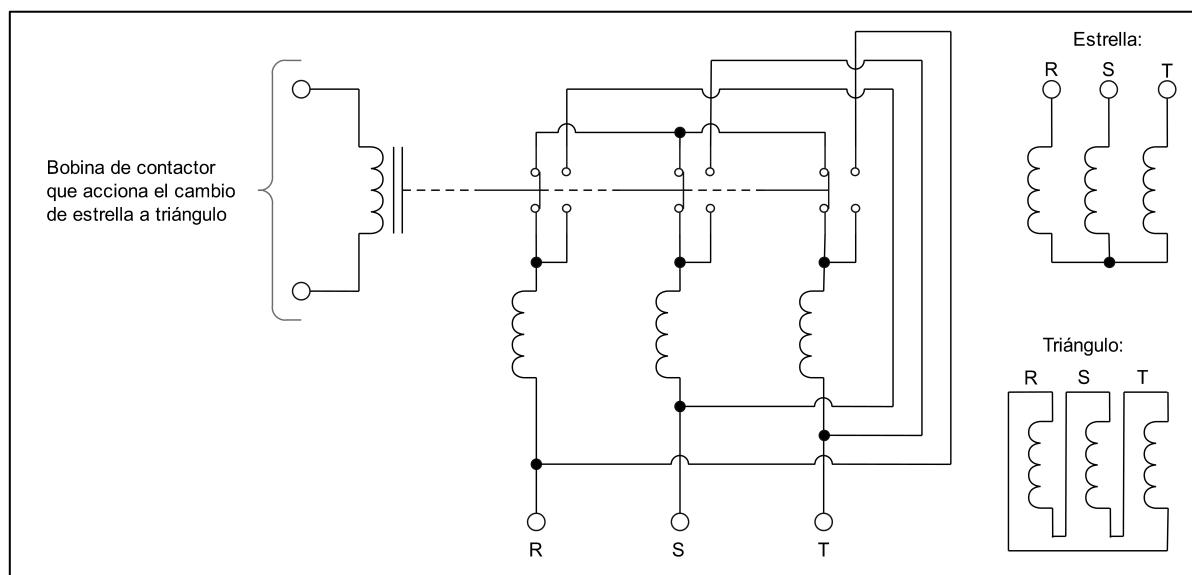


Figura 41: Circuito básico de arranque estrella-triángulo

Conversores

Luego de la introducción y teoría, en esta sección se desarrollarán los diversos tipos de conversores.

Conversor A/D del microcontrolador ARM (LPC2114)

Simplemente se enumerarán las características del mismo, el desarrollo propiamente dicho se encontrará más abajo cuando se desarrollen los distintos tipos de conversores A/D. Dichas características son:

- Resolución: 10[bits].
- Tipo: de Aproximaciones Sucesivas.
- Tiempo de conversión $\geq 2,44[\mu\text{s}]$ (11 ciclos de 4,5[MHz] máximo).
- Canales: multiplexación entre 4 entradas (AIN0 a AIN3).
- Tensión de referencia: 3[V].
- Valor del LSB: $1\text{LSB} = \frac{V_{ref}}{2^n} = \frac{3[V]}{2^{10}} = 2,9297[mV]$
- Otras características:

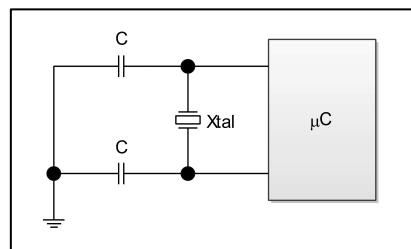


Figura 42: Conexión del cristal en el microcontrolador

- Cristal: 14,7456[MHz].
- PLL x 4 < 60[MHz] máximo.
- $V_O = 3,3[V]$, $I_{O_{max}} = 4[mA]$.

Conversores D/A, introducción

El principio básico de estos conversores es la suma ponderada de los niveles de tensión o corriente que corresponden a cada bit del código de entrada.

Conversor D/A de resistencias ponderadas

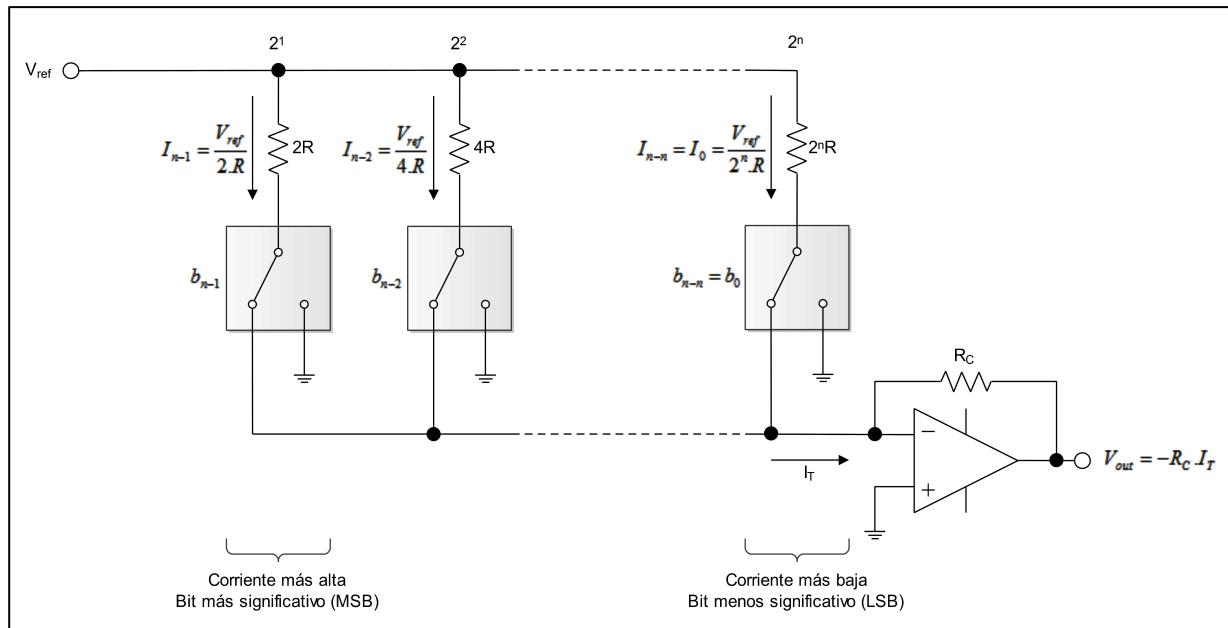


Figura 43: Esquema de un conversor D/A de resistencias ponderadas

El conversor consta de 3 partes: la fuente de referencia (V_{ref}), la red de resistencias ponderadas, y el convertidor corriente / tensión (I/V).

Cada uno de los bits de entrada (b_i) tiene asociado un interruptor y una resistencia. La red de resistencias representa los pasos binarios de los bits del código digital de entrada. La resistencia de menor valor ($2R$) corresponde al bit más significativo (MSB).

La corriente que circula por cada rama es proporcional al peso binario de su bit asociado, en fórmulas:

$$I_T = b_0 \cdot I_0 + b_1 \cdot I_1 + \dots + b_{n-1} \cdot I_{n-1} \quad (34)$$

$$= \frac{V_{ref}}{R} \left(\frac{b_0}{2^n} + \frac{b_1}{2^{n-1}} + \dots + \frac{b_{n-1}}{2^0} \right) \quad (35)$$

$$I_T = \frac{V_{ref}}{2^n \cdot R} \sum_{i=0}^{n-1} (b_i \cdot 2^i) \quad (36)$$

Por otro lado:

$$V_{out} = -R_C \cdot I_T \quad (37)$$

$$= -V_{ref} \frac{R_C}{2^n \cdot R} \sum_{i=0}^{n-1} (b_i \cdot 2^i) \quad (38)$$

Ahora si $R_C = R$:

$$V_{out} = -\frac{V_{ref}}{2^n} \sum_{i=0}^{n-1} (b_i \cdot 2^i) \quad (39)$$

El gran inconveniente que posee este tipo de conversores es que es necesario utilizar resistencias de valores muy distintos y las tolerancias de las mismas deben ser muy pequeñas, para tener una buena precisión, por lo que hace difícil su fabricación a gran escala para un número de bits elevado.

Conversor D/A en escalera R-2R

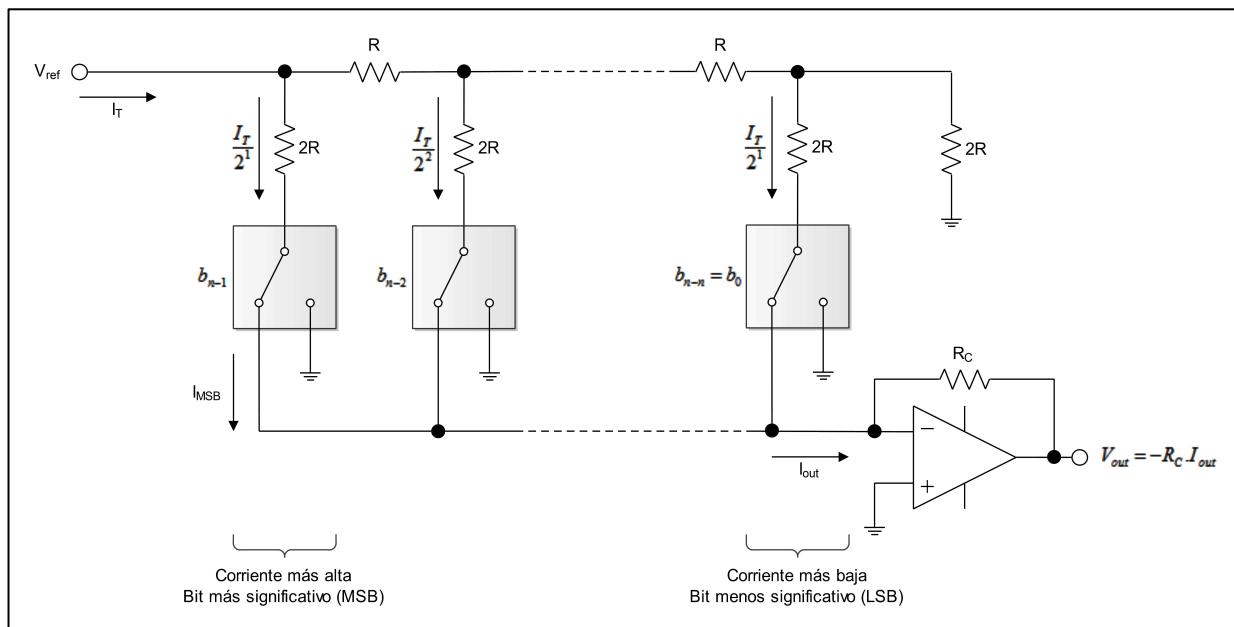


Figura 44: Esquema de un conversor D/A en escalera R-2R

En fórmulas, su funcionamiento es el siguiente:

$$I_T = \frac{V_{ref}}{R} \quad (40)$$

$$I_{MSB} = \frac{I_T}{2^1} \quad (41)$$

Desarrollando:

$$I_{out} = b_{n-1} \frac{I_T}{2^1} + b_{n-2} \frac{I_T}{2^2} + \dots + b_0 \frac{I_T}{2^n} \quad (42)$$

$$I_{out} = \frac{V_{ref}}{2^n \cdot R} (b_{n-1} \cdot 2^{n-1} + b_{n-2} \cdot 2^{n-2} + \dots + b_0 \cdot 2^0) \quad (43)$$

$$V_{out} = -I_{out} \cdot R_C \quad (44)$$

Reemplazando (43) en (44):

$$V_{out} = -V_{ref} \frac{R_C}{2^n \cdot R} (b_{n-1} \cdot 2^{n-1} + b_{n-2} \cdot 2^{n-2} + \dots + b_0 \cdot 2^0) \quad (45)$$

$$V_{out} = -V_{ref} \frac{R_C}{2^n \cdot R} \sum_{i=0}^{n-1} (b_i \cdot 2^i) \quad (46)$$

Nuevamente, si $R_C = R$:

$$V_{out} = -\frac{V_{ref}}{2^n} \sum_{i=0}^{n-1} (b_i \cdot 2^i) \quad (47)$$

Conversores A/D, introducción

Bueno, si bien un poco ya se vio este tema más arriba, se hará una teoría más profunda del tema.

La tarea de estos conversores es la de transformar una señal analógica, de tensión o de corriente, variable en el tiempo en información binaria, a partir del cociente entre la longitud de la señal de entrada y la de referencia. En fórmulas es:

$$N = \frac{V_i}{V_{ref}} = \sum_{i=1}^{\infty} \left(\frac{b_i}{2^i} \right) \quad (48)$$

Por supuesto que la ecuación (48) se refiere al resultado exacto, ideal de la conversión, la que se logra para un número infinito de bits.

A continuación se muestra en ejes coordinados, la característica de transferencia del conversor:

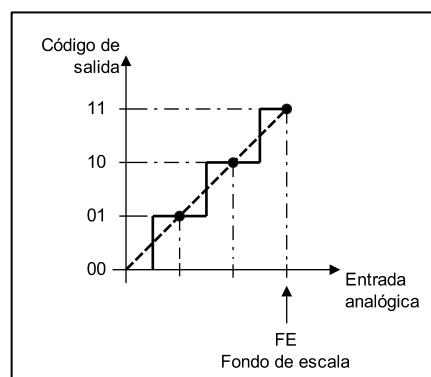


Figura 45: Característica de transferencia de la conversión A/D

Del gráfico se extrae el concepto de *resolución* del conversor, teniendo en cuenta que éste corresponde al bit menos significativo (LSB), y responde a la siguiente fórmula:

$$\text{resolución} = \frac{FE}{2^n - 1} \quad (49)$$

Donde el término “ $2^n - 1$ ” corresponde a la cantidad de escalones, intervalos o estados posibles.

Si el número de bits del conversor es muy grande, por ejemplo de 8[bits] o más, la resolución se puede aproximar a:

$$\text{resolución} = \frac{FE}{2^n} \quad (50)$$

Algunas características de estos conversores son las siguientes:

- *Error de offset*: es el desplazamiento horizontal de la característica de transferencia.
- *Ganancia*: es la pendiente de la recta que une los puntos medios de los peldaños primero y último de la función de transferencia del conversor. Idealmente es la mitad.
- *Monotonidad*: un conversor A/D es monótono si aumenta su salida a medida que la entrada también aumenta su valor.
- *Tiempo de conversión* (t_c): es el tiempo necesario para que el conversor realice una conversión. Ésta comienza cuando se activa la señal de entrada de inicio de conversión (SC: Start Conversion), y finaliza después de un cierto tiempo (t_c) cuando lo indica la señal de salida de fin de conversión (EOC: End Of Conversion).

Conversor A/D Flash (paralelo)

En este tipo de conversores, la V_{in} , en este caso tensión de entrada a convertir, se aplica a una cadena de comparadores, donde se compara con valores de tensión que cambian progresivamente de un comparador a otro mediante el uso de divisores resistivos.

Al aplicarse la tensión de entrada, para un grupo de comparadores, la tensión de entrada supera las tensiones de referencia generando un nivel alto; mientras que para otro grupo, la tensión de entrada es inferior a las de referencia y generan un nivel bajo. Las salidas de los comparadores se conectan a un codificador que convierte sus entradas en código binario. Cuando la señal “SC” se activa por medio de un flanco de subida activa las salidas de los comparadores y el codificador. El código binario se almacena en un registro de salida hasta el momento que la señal de control “SC” tome el valor “0” indicando el fin de la conversión (EOC).

A continuación se muestra el circuito esquemático de este tipo de conversores:

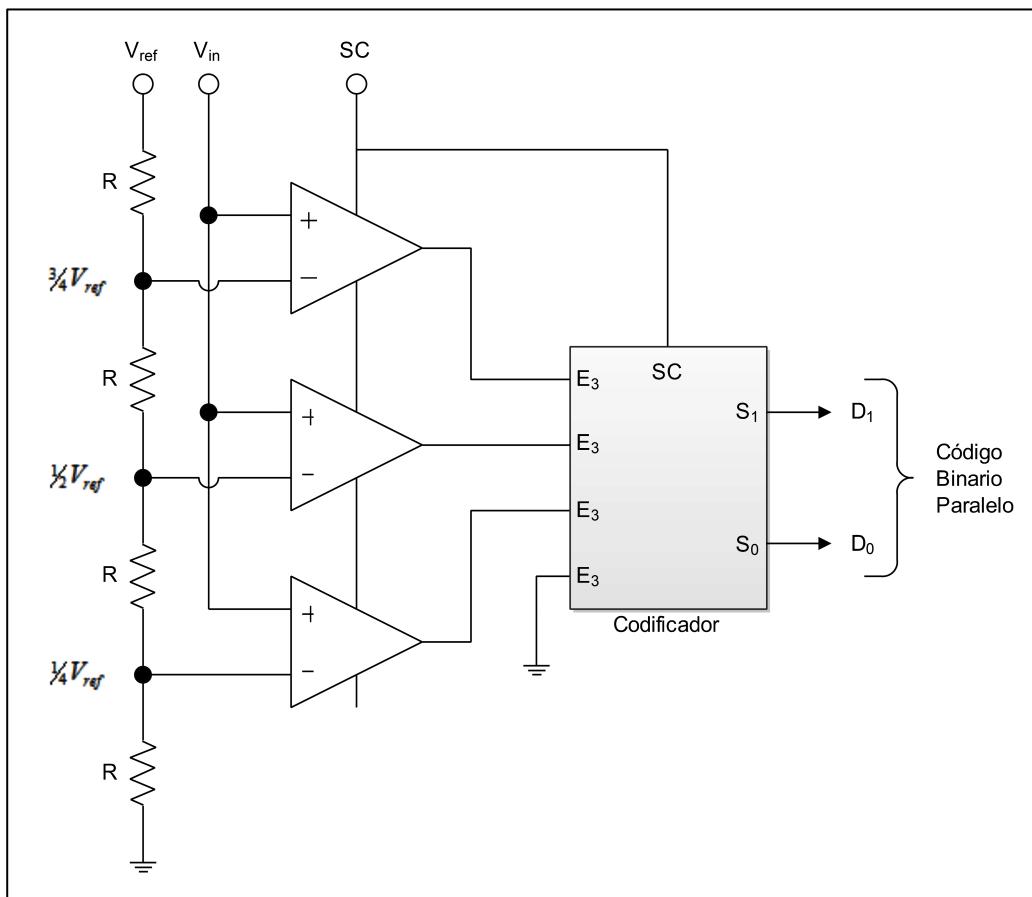


Figura 46: Circuito esquemático de un conversor A/D tipo flash de 2[bits]

Y su tabla de funcionamiento es la siguiente:

E_3	E_2	E_1	E_0	S_1	S_0
1	1	1	0	1	1
0	1	1	0	1	0
0	0	1	0	0	1
0	0	0	0	0	0

Para

$$V_{in} > \frac{3}{4} V_{ref}$$

$$V_{in} > \frac{1}{2} V_{ref}$$

$$V_{in} > \frac{1}{4} V_{ref}$$

$$V_{in} < \frac{1}{4} V_{ref}$$

Tabla 1: Tabla de funcionamiento de un conversor A/D tipo flash de 2[bits]

Algunas características de este tipo de conversores, son las siguientes:

- Se necesitan “ $2^n - 1$ ” comparadores para “n” bits de resolución (8[bits] máximo).
- La conversión es muy rápida, entre 10[ns] a 100[ns].
- Una gran capacidad de entrada limita el trabajo a altas frecuencias.
- Los comparadores deben ser muy rápidos y por lo tanto consumen mucha energía.

Para lograr aumentar la resolución, sin aumentar la potencia, pero sacrificando un poco el tiempo de conversión, es por medio del *conversor A/D*

semiparalelo (o semiflash) con el cual se pueden obtener resoluciones de 12[bits]. El circuito esquemático de este tipo de conversores es el siguiente:

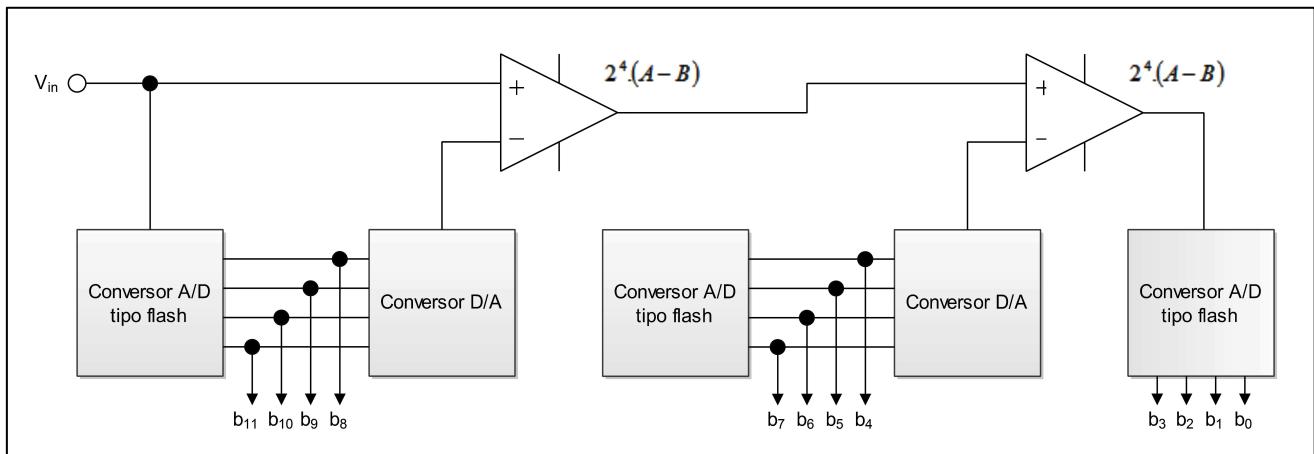


Figura 47: Circuito esquemático de un conversor semiparalelo de 12[bits]

Su funcionamiento es el siguiente: el primer conversor obtiene los 4 bits de mayor peso. Estos son convertidos nuevamente a una señal analógica por medio del conversor D/A y restados de la tensión de entrada. Cada resta es multiplicada por “ 2^n ”, donde “n” es el número de bits de los conversores flash. Así se vuelve a repetir el proceso hasta terminar la conversión. Finalmente cabe remarcar que todos los conversores A/D tienen la misma escala de tensiones.

Conversor A/D de aproximaciones sucesivas

Se compone de un registro de aproximaciones sucesivas (SAR: Successive Approximations Register), un conversor D/A y un comparador. Al iniciar la conversión, el SAR envía al conversor D/A el código correspondiente a la mitad de escala (1000....0000). el conversor D/A genera la tensión que corresponde a éste código y la compara con V_{in} . La salida del comparador va al SAR, que decide cuál debe ser el MSB y luego envía al conversor D/A un nuevo código que mantiene el resultado anterior y pone a “1” el siguiente bit. El proceso se repite tantas veces como bits tenga el código de salida, o sea, hasta fijar el valor del LSB. El tiempo de conversión resulta del producto entre el número de bits más “1”, y el período del reloj. Esto se debe a que la palabra digital final no pasa al registro de salida hasta el siguiente flanco de reloj, en el que también se informa el fin de la conversión (EOC). En fórmula:

$$t_{conv} = (n + 1) \cdot t_{clk} \quad (51)$$

A continuación se muestra un esquema de un conversor A/D de aproximaciones sucesivas; junto a un árbol de decisión para un conversor, por ejemplo, de 2[bits]:

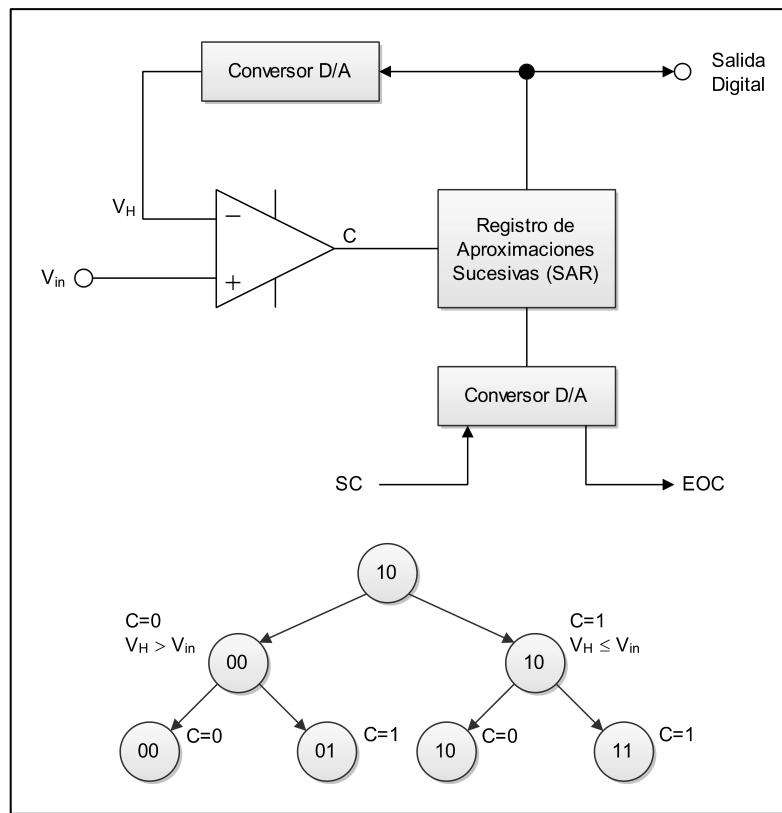


Figura 48: Esquema de un conversor A/D de aproximaciones sucesivas y árbol de decisión para un ejemplo de 2[bits]

Algunas características de este tipo de conversores son las siguientes:

- Es el más utilizado.
- Tienen un tiempo de conversión constante y solo mayor a los de tipo *flash* o *semiflash*.
- Poseen una excelente resolución.
- Es una buena relación costo / velocidad.

Conversor A/D de rampa en escalera

Este tipo de conversores están compuestos por un contador, un conversor D/A, una compuerta AND y un comparador (AO).

La conversión inicia resemando el contador y aplicando la tensión V_{in} . Luego, el conversor D/A comenzará a generar una tensión de salida equivalente al número binario almacenado en el contador, generando una rampa. Dicha rampa de tensión continuará aumentando hasta que la salida del comparador cambie al estado bajo, o sea la compuerta AND sale con "0", y el contador se detiene. A continuación un circuito de control almacenó el código de salida y lo entrega.

El circuito esquemático y diagrama de tiempos del presente conversor se muestra a continuación:

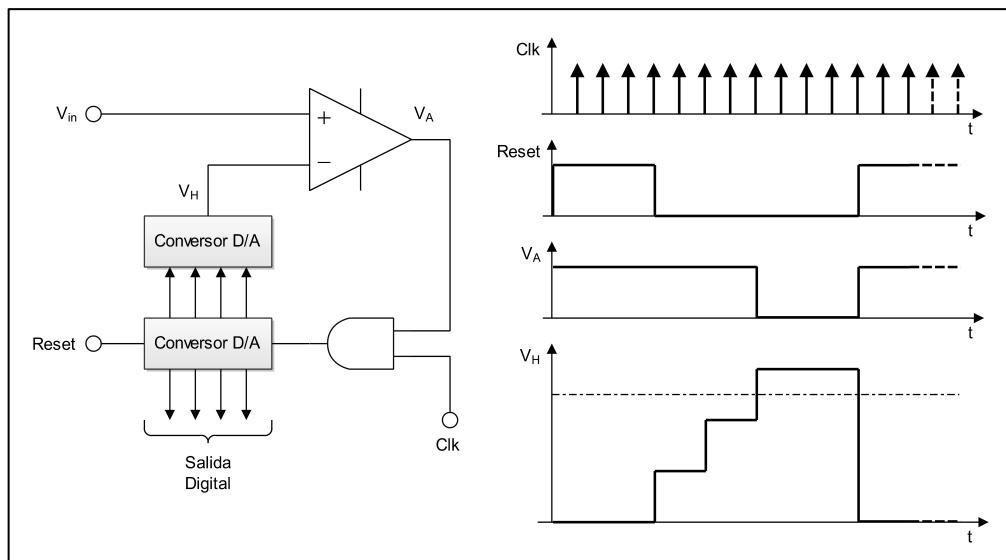


Figura 49: Circuito esquemático y diagrama de tiempos de ejemplo de un conversor A/D de rampa en escalera

Hay que recalcar que este tipo de convertidores es lento, ya que el tiempo de conversión se encuentra en función de la tensión analógica de entrada. En el peor de los casos, el tiempo de conversión es el siguiente:

$$t_{conv} = (2^n + 1).t_{clk} \quad (52)$$

Por lo tanto, el mismo crece exponencialmente con la cantidad de bits.

Conversor A/D de pendiente simple

Este tipo de conversores funcionan convirtiendo tensión en tiempo. Se emplea un generador de rampa lineal basado en un integrador para generar una tensión de referencia de pendiente constante. La conversión inicia con la activación de “SC” poniendo en cero el contador y un capacitor “C” inicia su descarga. Luego se habilita el contador y se genera la rampa. Cuando dicha rampa alcanza el valor de V_{in} finaliza la conversión. El capacitor se carga mediante una fuente de corriente constante.

Esquemáticamente, el circuito de este tipo de conversores es el siguiente:

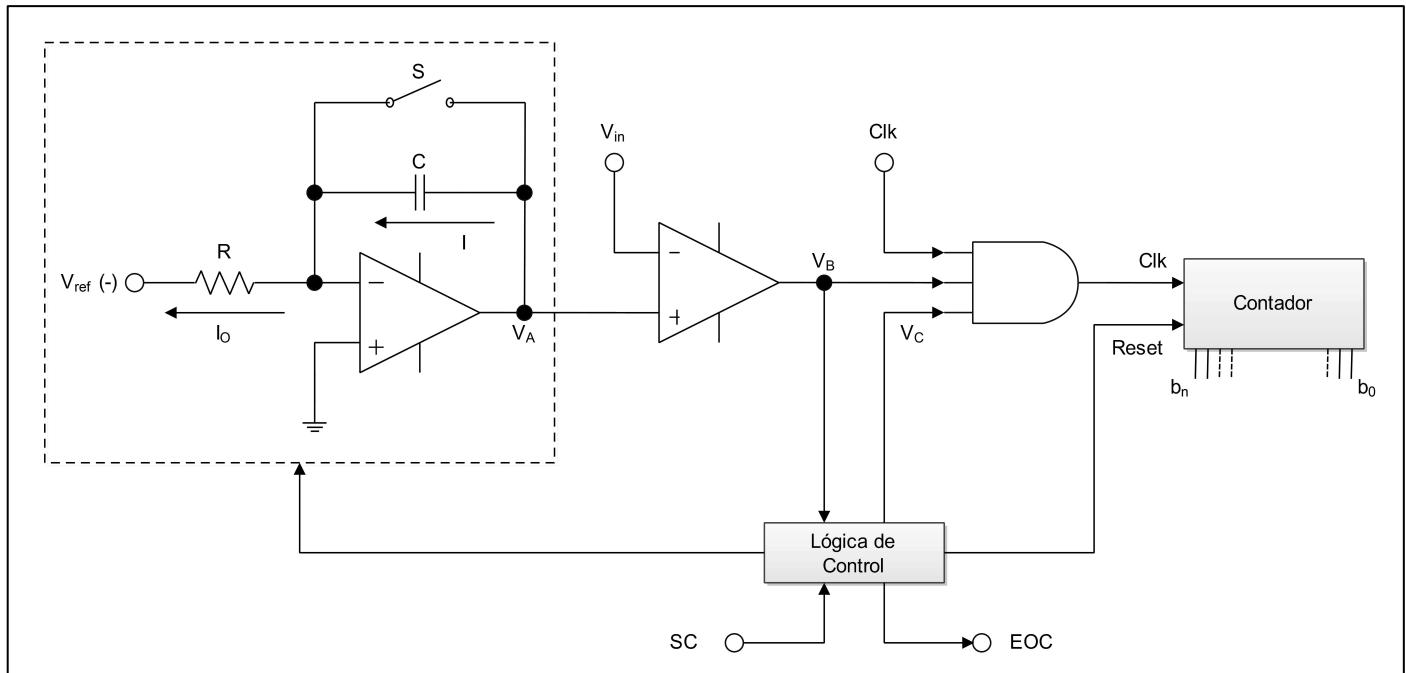


Figura 50: Circuito esquemático de un conversor A/D de pendiente simple

En fórmulas:

$$I_o = -\frac{V_{ref}}{R} \quad (53)$$

Por otro lado:

$$I = C \cdot \frac{dV_c}{dt} \quad (54)$$

Igualando (53) y (54):

$$C \cdot \frac{dV_c}{dt} = -\frac{V_{ref}}{R} \quad (55)$$

Resolviendo:

$$V_c = -V_{ref} \cdot \frac{t}{R \cdot C} \quad (56)$$

Ahora, se obtendrá el valor de “\$V_A\$”:

$$V_A = -\frac{1}{R \cdot C} \int v_{ref}(t) \cdot dt = -\frac{1}{R \cdot C} \cdot V_{ref} \cdot t_C \quad (57)$$

Como se mencionó anteriormente, la tensión en el capacitor crecerá hasta igualar la tensión en la entrada; teniendo en cuenta esto, y a su vez como “\$V_{ref}\$” es negativa, la “\$V_c\$” finalmente queda:

$$V_C = V_{ref} \cdot \frac{t_C}{R.C} \quad (58)$$

Y por lo tanto, finalmente reemplazando “ V_{in} ” por “ V_{ref} ” en el punto que ambas se igualan, se puede despejar el tiempo real de conversión “ t_C ”:

$$t_C = R.C \cdot \frac{V_{in}}{V_{ref}} \quad (59)$$

El diagrama de tiempos de este tipo de conversores es el siguiente:

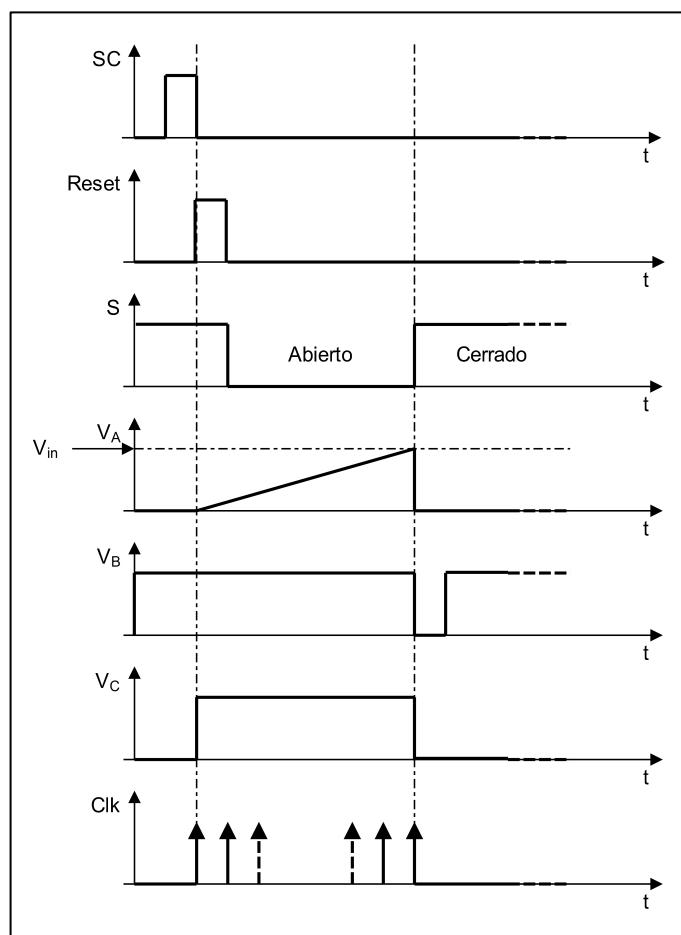


Figura 51: Diagrama de tiempos de un conversor A/D de pendiente simple

Algunas características de este tipo de conversores son las siguientes:

- La polaridad de “ V_{ref} ” debe ser opuesta a la de la señal de entrada “ V_{in} ”.
- El generador de rampa debe ser extremadamente lineal, por lo que “ R ”, “ C ” y “ f_{clk} ” deben ser sumamente constantes.
- Debido a “ C ”, posee problemas respecto de la polarización dieléctrica.
- La salida del contador al final de cada ciclo de conversión será:

$$n = f \cdot t_C \quad (60)$$

$$n = f \cdot R \cdot C \cdot \frac{V_{in}}{V_{ref}} \quad (61)$$

- El tiempo de conversión depende de la magnitud de la señal de entrada, tal como se ve en la ecuación (59), en donde su valor máximo ocurre para $V_{in} = V_{ref}$, en donde este valor será:

$$t_{C-maz} = R \cdot C \quad (62)$$

Conversor A/D de pendiente doble

Este tipo de conversores realizan una integración doble para eliminar los errores de los conversores A/D de pendiente simple derivados del capacitor "C" y de " f_{clk} ", sumado a que logran una mayor inmunidad al ruido.

Su funcionamiento consiste en realizar una primer integración durante un tiempo constante de la tensión de entrada " V_{in} ", luego se integra la " V_{ref} " hasta alcanzar el valor inicial de cero a la salida del integrador.

A continuación se muestra el circuito esquemático de este tipo de conversores:

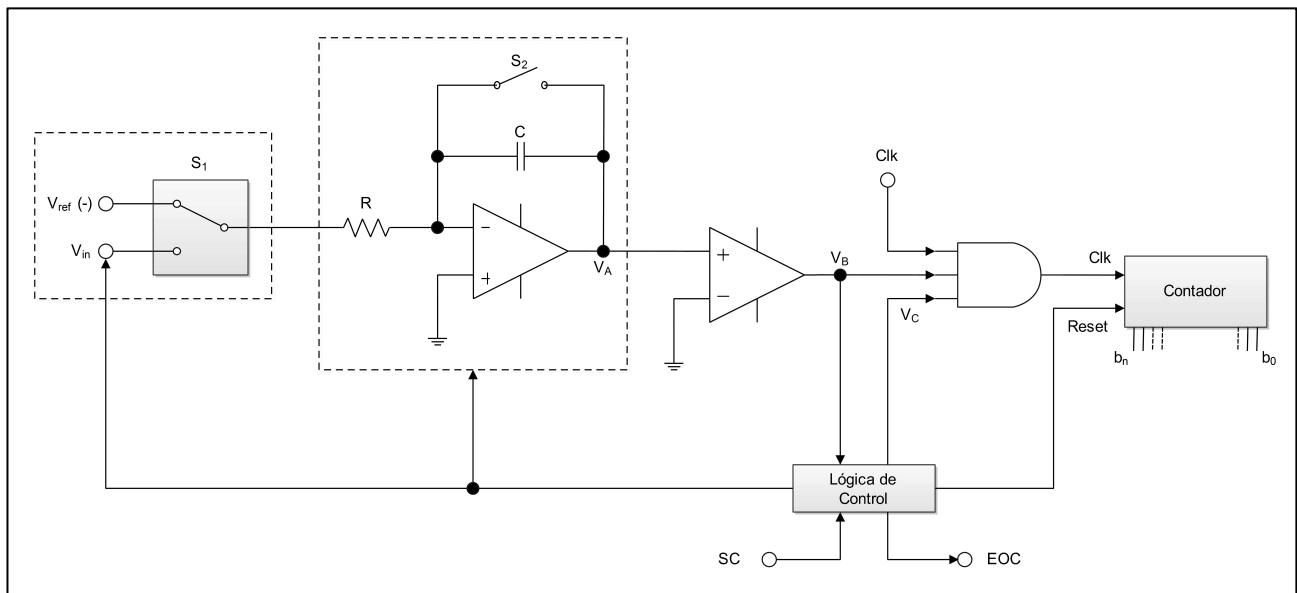


Figura 52: Circuito esquemático de un conversor A/D de pendiente doble

Por lo tanto, durante un tiempo:

$$t_1 = N_1 \cdot t_{CLK} \quad (63)$$

Se produce la rampa de pendiente:

$$-\frac{V_{in}}{R.C} \quad (64)$$

. Al cabo de “ t_1 ”, la salida del integrador alcanzará el valor:

$$V_H = -t_1 \cdot \frac{V_{in}}{R.C} \quad (65)$$

A continuación se resetea el contador a cero, se conmuta “ S_1 ” y se inicia la segunda integración, con una rampa de pendiente:

$$\frac{V_{ref}}{R.C} \quad (66)$$

Durante el tiempo:

$$t_2 = N_2 \cdot t_{CLK} \quad (67)$$

Tiempo que transcurre hasta alcanzar nuevamente el valor de cero.

Y nuevamente la salida alcanzará un valor:

$$V_H = -t_2 \cdot \frac{V_{ref}}{R.C} \quad (68)$$

Por lo tanto, igualando los valores de “ V_H ”, o sea las ecuaciones (65) y (68), y operando matemáticamente, se obtiene lo siguiente:

$$-t_1 \cdot \frac{V_{in}}{R.C} = -t_2 \cdot \frac{V_{ref}}{R.C} \quad (69)$$

$$t_1 \cdot V_{in} = t_2 \cdot V_{ref} \quad (70)$$

Reemplazando (63) y (67) en (70):

$$N_1 \cdot t_{CLK} \cdot V_{in} = N_2 \cdot t_{CLK} \cdot V_{ref} \quad (71)$$

Donde finalmente:

$$N_2 = \frac{V_{in}}{V_{ref}} \cdot N_1 \quad (72)$$

Donde en (72) se termina por demostrar la independencia respecto de “ f_{CLK} ” y “ $R.C$ ”.

A continuación se muestra el diagrama de tiempos de este tipo de conversores:

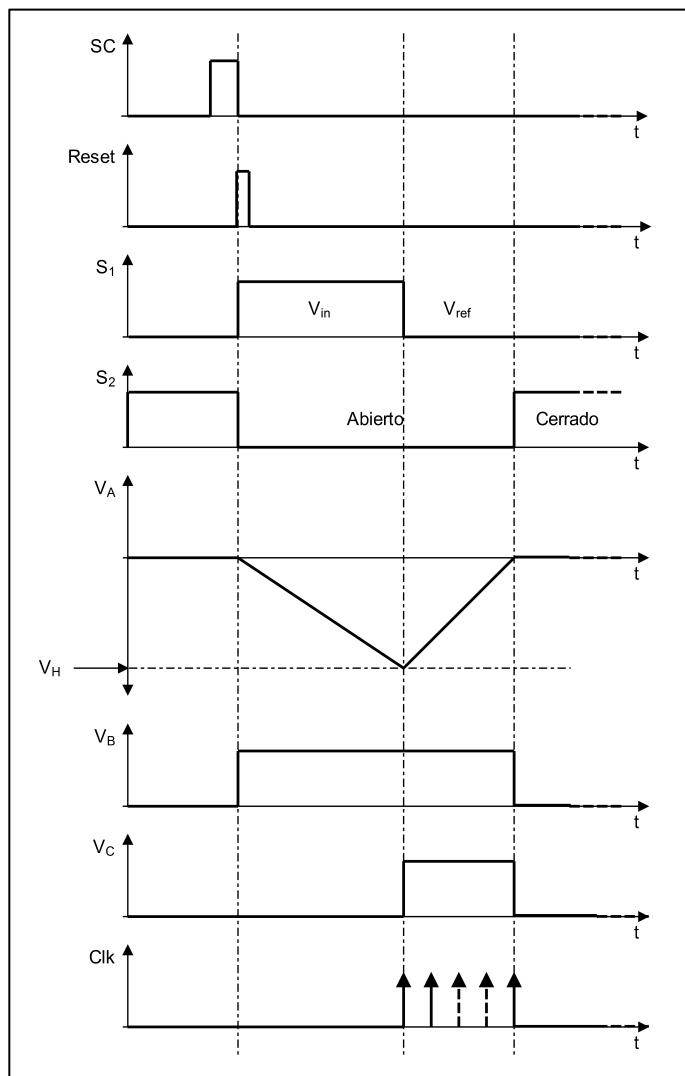


Figura 53: Diagrama de tiempos de un conversor A/D de pendiente doble

Por lo tanto, el tiempo de conversión es:

$$t_C = t_1 + t_2 \quad (73)$$

Y el tiempo de conversión máximo posible será:

$$t_{C-\max} = 2 \cdot (2^n \cdot t_{CLK}) = 2^{n+1} t_{CLK} \quad (74)$$

Algunas características de este tipo de conversores, son las siguientes:

- Son lentos, entre 30 a 40 conversiones por segundo.
- Permiten conseguir una altísima resolución, desde 24[bits] y más.

Unidad 3: Microprocesadores

Arquitectura de los microprocesadores

Existen 2 arquitecturas diferentes, las cuales relacionan el uso y distribución de la memoria.

Estas arquitecturas son:

- Von Neumann.
- Harvard.

Las cuales se desarrollan a continuación.

Arquitectura Von Neumann

Los sistemas con μPs (microprocesadores) se basan en éste tipo de arquitectura, en la cual la CPU (Central Processing Unit) está conectada a una memoria principal única, donde se guardan las instrucciones del programa y los datos. A dicha memoria se accede a través de un sistema de buses único (control, direcciones y datos).

A continuación se muestra un diagrama en bloques de esta arquitectura:

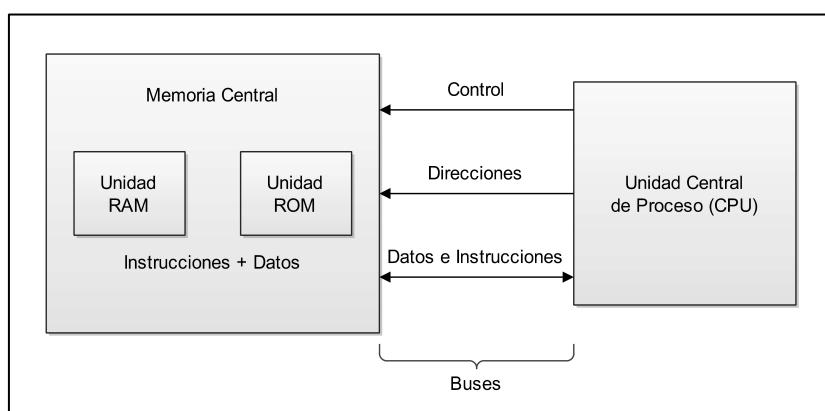


Figura 54: Diagrama en bloques de una máquina Von Neumann

Algunas características de este tipo de arquitectura, son las siguientes:

- Al tener un único bus hace que el μP sea más lento en la respuesta, ya que no puede buscar en memoria una nueva instrucción hasta que no finalice la transferencia de datos de la instrucción anterior.
- La limitación de la longitud de las instrucciones por el bus de datos hace que el μP tenga que realizar varios accesos a memoria para buscar instrucciones complejas.

Arquitectura Harvard

Este tipo de arquitecturas es la que utilizan los μ Cs (microcontroladores). Tiene la CPU conectada a dos memorias, una de instrucciones y otra de datos, por medio de 2 buses diferentes. Ambos buses son totalmente independientes, por lo que permite que puedan tener contenidos diferentes y de distinta longitud.

Además, el tiempo de acceso a las instrucciones puede superponerse con el de los datos, logrando una velocidad mayor en cada operación.

A continuación se muestra un diagrama en bloques de esta arquitectura:

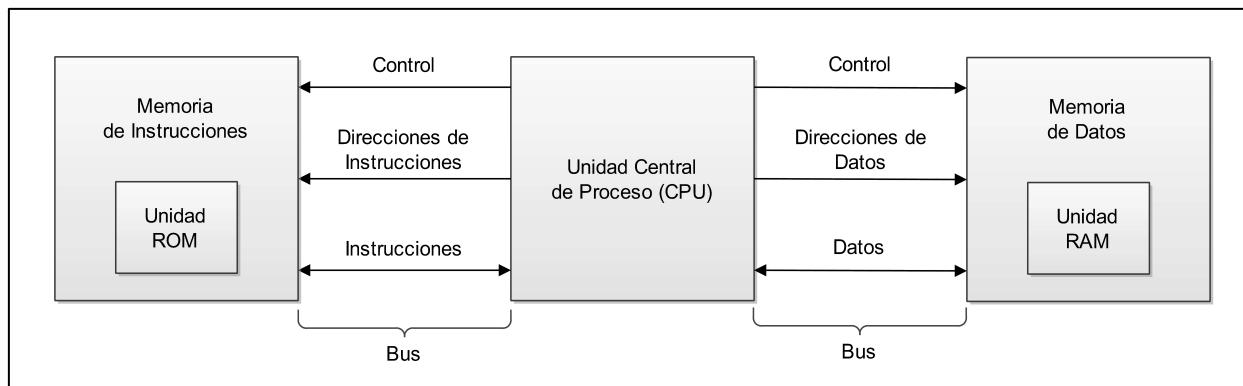


Figura 55: Diagrama en bloques de una máquina Harvard

Este tipo de arquitectura posee tres tipos de buses principales y uno de selección:

1. **Bus de Direcciones**: todo dispositivo de memoria o E/S posee entradas de dirección, que permiten seleccionar una localidad de memoria dentro del dispositivo. La cantidad de posiciones dentro del dispositivo determina la cantidad de líneas de dirección. Por ejemplo, una memoria de “1K” tiene 10 líneas de dirección, ya que la cantidad posible de localidades a las que se puede acceder es: $2^{10} = 1024$.
2. **Bus de Datos**: es el conjunto de terminales para salida o entrada de datos que poseen los dispositivos de memoria o E/S. Por ejemplo, una memoria “1K x 8” indica “1024” bytes o localidades, cada una de 8[bits].
3. **Bus de Control**: todo dispositivo usado en un sistema de buses, cuenta con líneas de control. El caso más simple es la memoria ROM, con 1 señal denominada “OE” (Output Enable), la cual permite dar salida al bus de datos de la localidad seleccionada por el Bus de Direcciones.
4. **Bus de Selección**: cualquier dispositivo que pueda ser colocado en un bus, deberá tener algún terminal con la función de selección a ese dispositivo. Normalmente esta señal puede ser “CS” (Chip Select), o “CE” (Chip Enable).

Si en un bus del μ P se coloca, por ejemplo, una memoria RAM de variables y una memoria ROM para el programa, el μ P deberá distinguir entre ellos y a su vez permitir acceder a sus direcciones internas. Para esto se utiliza un circuito decodificador que permite proveer a las memorias y los dispositivos E/S de una zona exclusiva del mapa de memoria del μ P. Así se puede distinguir entre 2 tipos de líneas de direccionamiento:

- Externa: diferencian al dispositivo del resto.
- Interna: la que se conecta directamente al dispositivo.

A continuación se muestra un diagrama esquemático de ejemplo de lo anterior:

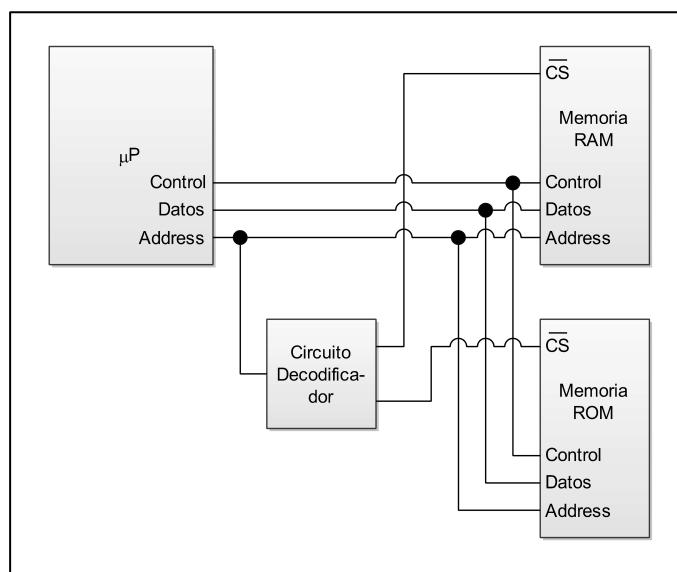


Figura 56: Diagrama esquemático sobre la utilización de los buses

Datos principales del μ C LPC2114

A continuación se darán las características del “pin-out” del μ C utilizado por la cátedra:

- Total de pines: 64[pines].
- 2 puertos: P0 y P1. Comprendidos desde P#.0 a P#.31; son bidireccionales de E/S de 32[bits].
- 4 Entradas A/D: Ubicadas en los pines P0.27, P0.28, P0.29 y P0.30.

Descripción de componentes y fundamentaciones teóricas

A continuación se describen varios componentes y funcionamiento del µC.

UART – RS232C

Se trata de un módulo receptor / transmisor asíncrono universal (UART - Universal Asynchronous Receiver / Transmitter) utilizado para realizar comunicaciones serie, donde los datos son enviados sin una señal de reloj externa. Los datos son divididos en palabras de una longitud fija, generalmente octetos, a las cuales se le agregan al comienzo códigos de sincronismo y al final se agregan otros.

Uno de los estándares más comunes es el RS232. Es una norma para el intercambio de datos entre 2 dispositivos: “DTE” (Data Terminal Equipment, o Equipo Terminal de Datos) y “DCE” (Data Communication Equipment, o Equipo de Comunicación de Datos). A pesar de que éste estándar soporta comunicación síncrona, su uso más habitual es en comunicaciones asíncronas.

Por ejemplo, en una comunicación habitual, la PC sería el DTE, y el modem el DCE. En el caso de 2 PCs, o sea DTEs entre sí, la conexión se denomina “NULL MODEM”.

En la tabla siguiente se encuentran todas las señales que manejan la interface RS232, los pines correspondientes a los diferentes tipos de conectores, ya sean DB9 o DB25, y si es propia del DTE o del DCE:

Señal	Descripción	Origen		Conector	
		DTE	DCE	DB25	DB9
GND Common Ground	--	X	X	7	5
TxD Transmitted Data	Datos enviados del DTE al DCE.	X		2	3
RxD Received Data	Datos enviados del DCE al DTE.		X	3	2
RTS Request To Send	Solicitud del DTE para preparar al DCE a recibir un dato.	X		4	7
CTS Clear To Send	Respuesta del RTS, para que el DTE transmita.		X	5	8
DTR Data Terminal Ready	Solicitud del DTE de que está listo para conectarse.	X		20	4
DSR Data Set Ready	Solicitud del DCE indicando que está listo para recibir del DTE.		X	6	6
DCD Data Carrier Detect	Señal del DCE cuando la conexión ha sido establecida con el equipamiento remoto.		X	8	1
RI Ring Indicator	Señal del DCE cuando se detectó una señal de “ring” en la línea telefónica.		X	22	9

Tabla 2: Tabla descriptiva de las distintas señales de una interface RS232

A continuación se muestra el “Control de flujo” de la comunicación de la norma RS232, lo que se denomina también “Handshaking”:

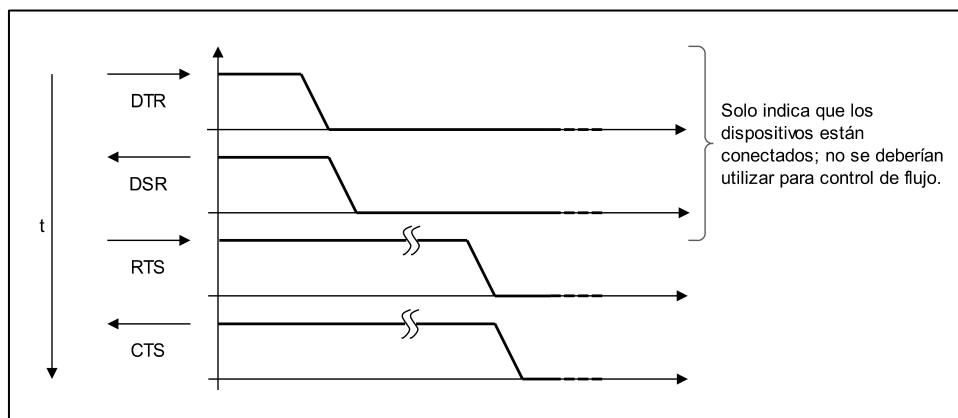


Figura 57: Control de flujo de una comunicación RS232

Algunas características son las siguientes:

- El LPC2114 implementa 2 UART. La UART 1 implementa todos los pines de entrada / salida; mientras que la UART 0 solo posee los necesarios para una comunicación de 3 líneas (Rx: pin 22, Tx: pin 19).
- *DLR (Divisor Latch Register)*: está compuesto por 2 registros de 8[bits] cada uno (*DLL*: Parte baja, y *DLM*: Parte alta). El valor total tiene una longitud de 16[bits] y es divisor del VPB (Velocidad Por Bit), que se deberá configurar para determinar el “*Baud Rate*” de la transmisión / recepción. En fórmulas, por la norma RS232:

$$\text{baud rate} = \frac{\text{VPB}}{16.\text{DLR}} \quad (75)$$

La frecuencia base será dividida por 16 veces el valor almacenado en DLR para conseguir las velocidades estándar. Según la norma RS232C, las velocidades estandarizadas en baudios son: 75, 110, 150, 300, 600, 1200, 2400, 4800, 9600 y 19200.

Un ejemplo de UART es la 8250, que es un circuito integrado diseñado para implementar interfaces para comunicaciones seriales. Se utiliza en PCs, impresoras, módems, etc. Incluye un generador de “bit rate” programable, permitiéndole trabajar en distintas frecuencias.

Baudios y bits por segundo

El “*Baudio*” es una unidad de medida que representa el número de símbolos transmitidos por segundo. Cada símbolo puede transportar uno o más bits (estados posibles, depende de la modulación).

El “*Bit Rate*” es la velocidad en [bits/segundos].

Por ejemplo, para una comunicación “8N1” a “9600[baudios]”, lo que significa:

1 bit de stop + 8 bits de datos + 1 bit de stop = 10[bits].

Por lo tanto se tiene:

$$\left[\frac{\text{caracteres}}{\text{segundo}} \right] = \frac{9600}{10} = 960 \left[\frac{\text{caracteres}}{\text{segundo}} \right] \quad (76)$$

$$\text{bits de datos} = 960 \left[\frac{\text{caracteres}}{\text{segundo}} \right] \cdot 8 = 7680 [\text{bps de datos}] \quad (77)$$

PC: Program Counter

El PC se ubica en el registro 15. Es un puntero a la instrucción que está 2 instrucciones delante de la que está en ejecución.

Una instrucción en la etapa de ejecución se completará aunque haya aparecido una interrupción.

Gráficamente se comprende de la siguiente manera:

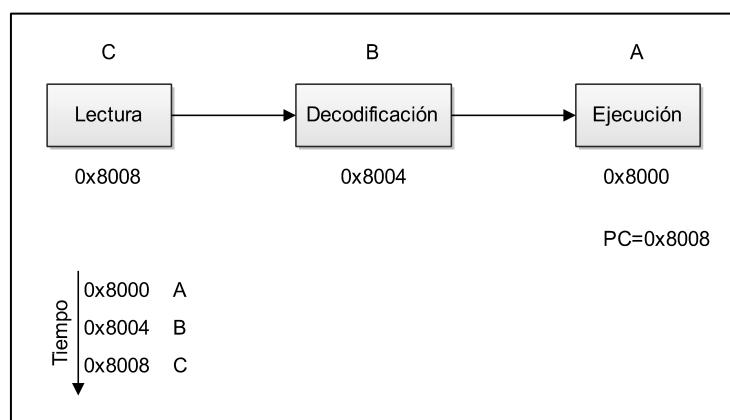


Figura 58: Funcionamiento del PC (Program Counter)

El ARM posee 7 modos de operación. En todos los modos se encuentran disponibles 16 registros generales y 1 o 2 registros de estado. Cada registro tiene 32[bits]. En los registros generales se encuentra el PC (R15). En los registros de estado, el CPSR y en algunos modos el SPSR.

PSR: Registro de Estado de Programa

Este registro sirve para guardar información de la operación más reciente de la ALU, controla la habilitación de las interrupciones, y establecen el modo de operación del procesador (7 modos de operación).

A continuación se muestra el formato del PSR:

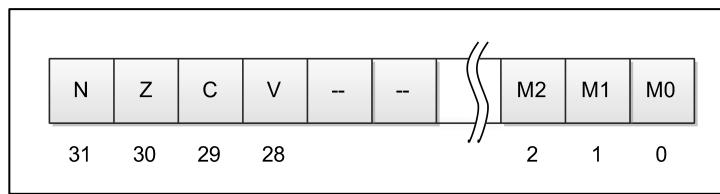


Figura 59: Formato del PSR

Donde “N”, “Z”, “C” y “V” son “banderas de condición”, las cuales se describen en el siguiente título.

El “CPSR” contiene la información del estado actual del μC (Current).

El “SPSR” es una copia del CPSR en el modo anterior de operación (Saved). Sirve para poder regresar en forma transparente de los modos de excepción.

Banderas de condición

Son testeadas desde las instrucciones que poseen condición y de esta forma determina si son ejecutadas o no. Además son modificados por las instrucciones de comparación (CMN, CMP, TEQ, TST, etc.), o por cualquier instrucción aritmética, lógica o de movimiento con el agregado de la opción “s”.

Las condiciones posibles pueden ser:

- *EQ*: Igual.
- *NE*: Distinto.
- *LT*: Menor.
- *GT*: Mayor.
- *LE*: Menor o igual.
- *GE*: Mayor o igual.

A continuación se muestran las diferentes banderas de condición y su significado:

- *N* (*Negative*, o *Negativo*): toma el valor “1” si el resultado es negativo (complemento a 2 -El cual se describe más adelante-), o “0” si es positivo.
- *Z* (*Zero*, o *Cero*): toma el valor “1” si el resultado es cero, y “0” para cualquier otro valor.
- *C* (*Carry*, o *Acarreo*): en una suma toma el valor “1” si se produce un desborde. En una resta toma el valor “0” si se produce préstamo. También se modifica en operaciones que involucran corrimientos.

- *V* (*oOverflow, o Desborde*): en una suma o resta, toma el valor “1” si se produce un desborde con signo.

VIC: Controlador Vectorizado de Interrupciones

Posee 32 entradas de interrupciones en 3 categorías:

- *FIQ (Fast Interrupt reQuest)*: de prioridad mayor.
- *IRQ vectorizada (Interrupt ReQuest)*: de prioridad media.
- *IRQ no vectorizada (Interrupt ReQuest)*: de prioridad menor.

Dentro de las 32 entradas de interrupciones, las más utilizadas son:

- *Eint0*: VIC channel 14.
- *Timer0*: VIC channel 14.
- *ADC*: VIC channel 18.

Complemento a 2

Aquí se citarán diferentes condiciones a las cuales se utiliza el “*Complemento a 2*”.

En la representación de números con signo, el bit de mayor peso (MSB) indica el signo: “0” para positivos y “1” para negativos, por lo tanto, el rango de representación queda acotado entre:

$$-2^{(n-1)} \leq \text{rango} \leq 2^{(n-1)} - 1 \quad (78)$$

Por lo tanto, para un ejemplo de un número de 3[bits] se tendrá:

		Complemento a 1	Complemento a 2	Valor final
3	→ 0 1 1	→ 1 0 0	→ 1 0 1	→ -3
2	→ 0 1 0	→ 1 0 1	→ 1 1 0	→ -2
1	→ 0 0 1	→ 1 1 0	→ 1 1 1	→ -1
0	→ 0 0 0			
-1	→ 1 1 1			
-2	→ 1 1 0			
-3	→ 1 0 1			
-4	→ 1 0 0			

Negativos

Figura 60: Ejemplo de “Complemento a 2” para un número de 3[bits]

La forma de pasar un número positivo a su “Complemento a 2”, consta de 2 pasos:

1. Se toma el número binario positivo y se invierten los “1” por “0”, y los “0” por “1”, manteniendo la posición de peso. Este paso consiste en transformar el número binario positivo en “Complemento a 1”.
2. Al número binario resultante del paso 1, se le suma “1” en el LSB, y finalmente éste es el número binario positivo transformado en “Complemento a 2”.

Para una mejor comprensión, a continuación se muestra paso a paso cómo pasar el número “+25” a su “Complemento a 2”:

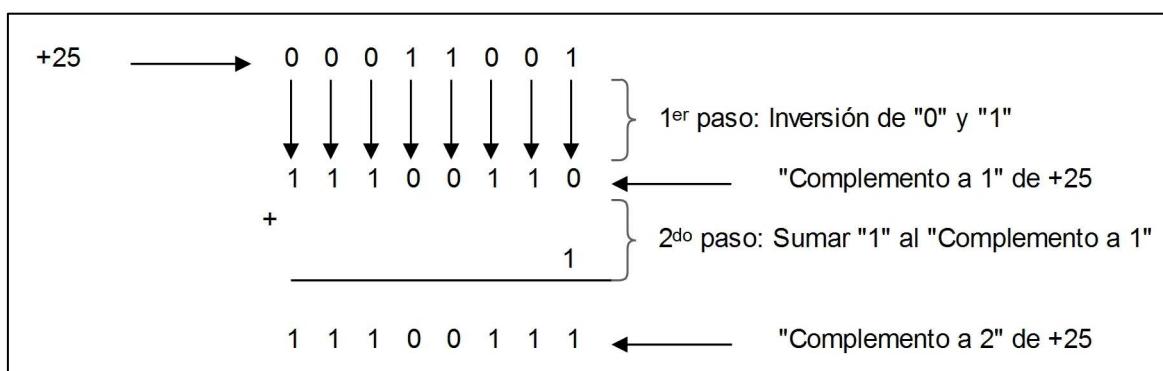


Figura 61: Ejemplo para convertir un número positivo (+25) en su "Complemento a 2"

Suma de 2 números positivos

En la suma, si hay acarreo final, el mismo se desprecia.

Por ejemplo, se sumarán los números positivos “25” y “40”, y sus equivalentes en binario:

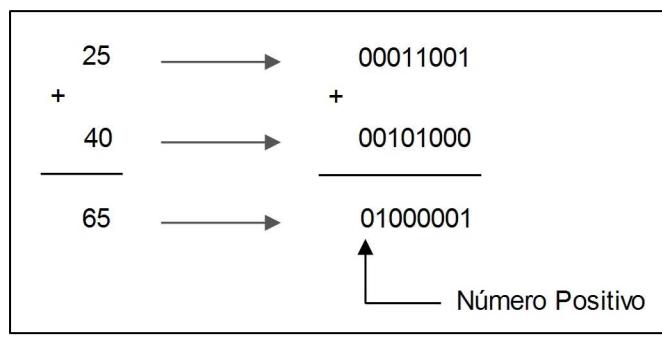


Figura 62: Ejemplo de Complemento a 2 de la suma de 2 números positivos

Cuando el número de bits requerido para representar la suma excede al número de bits de los 2 números, se produce un “desborde” (Overflow), que se indica mediante un “1” de signo incorrecto. Por lo tanto, un “desborde” se puede producir cuando ambos números son positivos o negativos.

A continuación se muestra un ejemplo de “desborde”, con la suma de los números “107” y “68”:

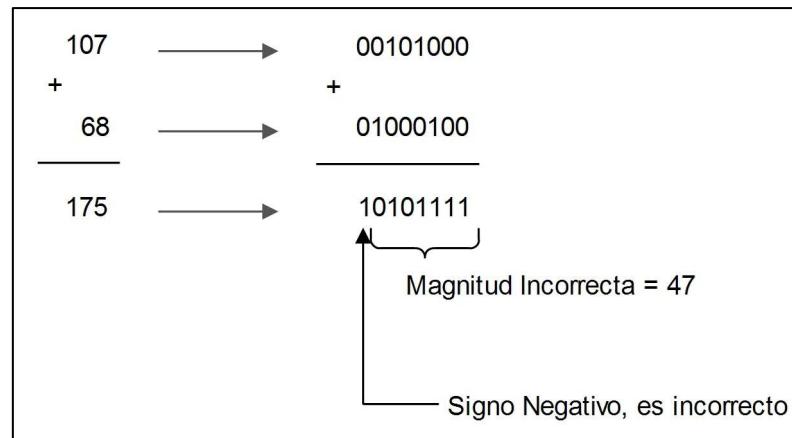


Figura 63: Ejemplo de "desborde" en una suma de 2 números positivos

Suma de 1 número positivo y 1 número negativo ($|positivo| > |negativo|$)

A continuación se muestra un ejemplo con los números “40” y “-25”:

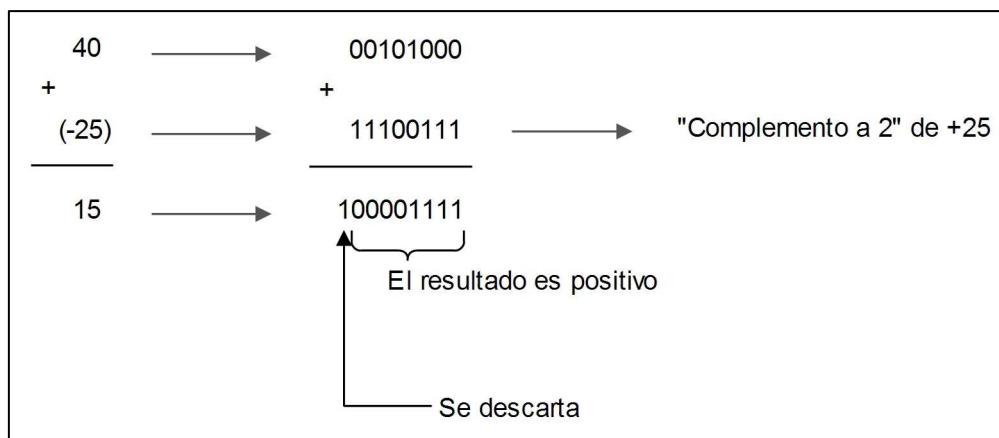


Figura 64: Ejemplo de suma de 1 número positivo y 1 número negativo ($|positivo| > |negativo|$)

Suma de 1 número positivo y 1 número negativo ($|positivo| < |negativo|$)

A continuación se muestra un ejemplo con los números “25” y “-40”, donde el resultado es negativo y por lo tanto está en “Complemento a 2”:

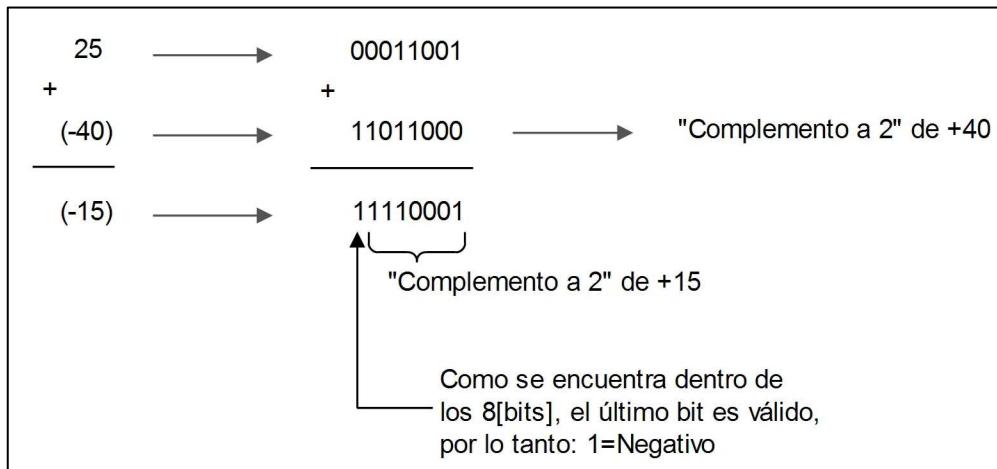


Figura 65: Ejemplo de suma de 1 número positivo y 1 número negativo ($|positivo| < |negativo|$)

Suma de 2 números negativos

A continuación se muestra el ejemplo de la suma entre los números “-25” y “-40”:

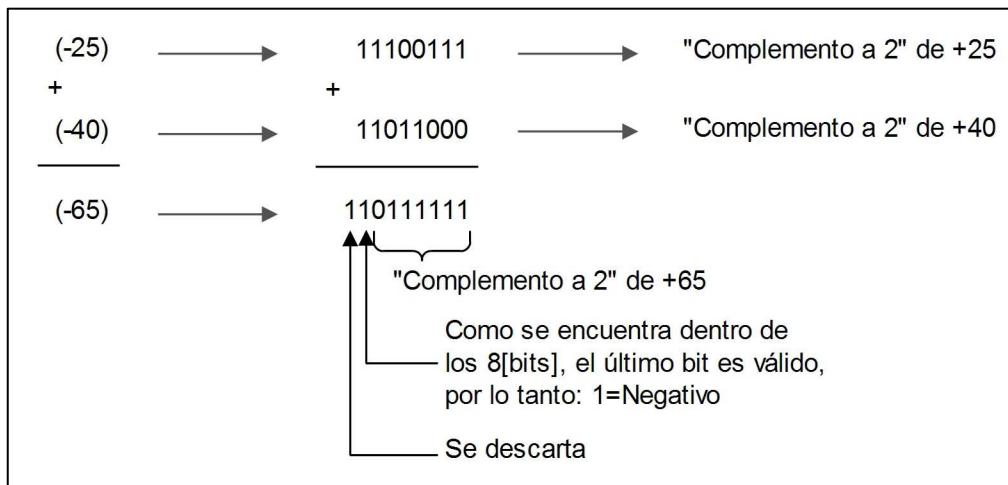


Figura 66: Ejemplo de suma de 2 números negativos

Temporizador LM555

Es de uso frecuente, y aquí se verán 2 configuraciones simples para su utilización con el mínimo de componentes.

Como punto a tener en cuenta es la amplitud en el rango de alimentación del mismo, ya que esta puede ser desde 5[V] a 15[V].

Circuito “Astable” (Oscilador)

A continuación se presenta el circuito de la configuración “Astable” y su diagrama de tiempos del LM555:

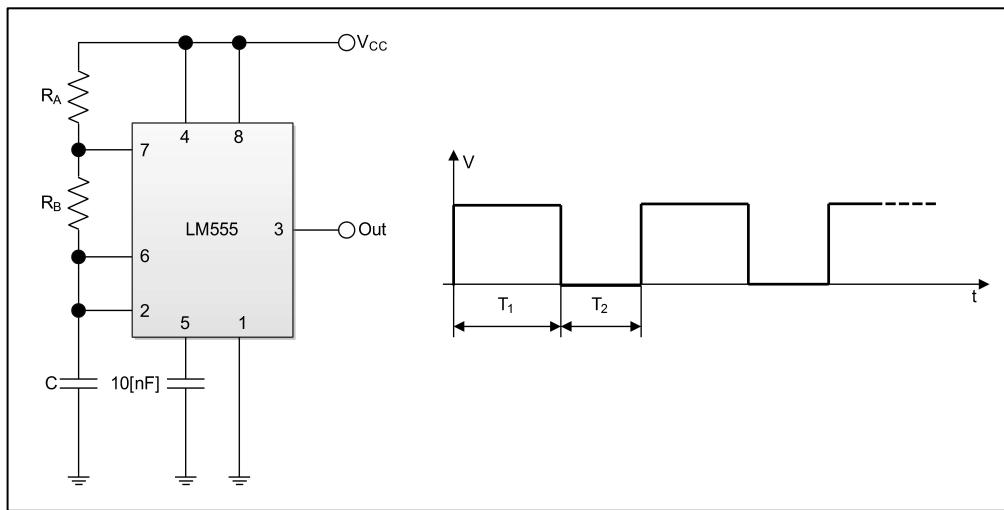


Figura 67: Configuración "Astable" y Diagrama de tiempos del LM555

En dicha configuración hay que tener en cuenta los rangos de “ R_A ”, “ R_B ” y “ C ”:

$$1[k\Omega] \leq R_A, R_B \leq 1[M\Omega] \quad (79)$$

$$1[nF] \leq C \leq 100[\mu F] \quad (80)$$

Ahora, para el cálculo de la frecuencia de trabajo, se hace de la siguiente manera:

$$T_1 = 0,693 \cdot (R_B) \cdot C \quad (81)$$

$$T_2 = 0,693 \cdot (R_A + R_B) \cdot C \quad (82)$$

Por lo que finalmente, teniendo en cuenta la (81) y la (82), la frecuencia de trabajo “ f ” queda:

$$f = \frac{1}{T_1 + T_2} = \frac{1,44}{(R_A + 2 \cdot R_B) \cdot C} \quad (83)$$

Donde aplicando la (83), y teniendo en cuenta (79) y (80), el rango de “ f ” es:

$$0,1[Hz] \leq f \leq 100[kHz] \quad (84)$$

Circuito “Monoestable”

A continuación se presenta el circuito de la configuración “Monoestable” y su diagrama de tiempos del LM555:

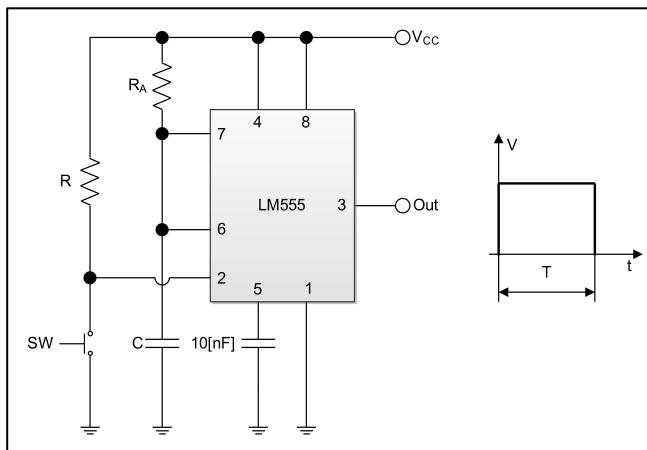


Figura 68: Configuración "Monoestable" y Diagrama de tiempos del LM555

Su funcionamiento consiste en apretar el pulsador “SW” para que en la salida “Out” aparezca un “1” un cierto tiempo “T”. Este tiempo se calcula de la siguiente forma:

$$T = 1,1 \cdot R_A \cdot C \quad (85)$$

Donde el rango de “T” es:

$$1[\mu s] \leq T \leq 10[s] \quad (86)$$

Tolerancias normalizadas de los resistores

Si bien este tema se ve en profundidad en la cátedra “Tecnología Electrónica”, simplemente se mostrarán los valores de las tolerancias de los resistores tanto en valores “porcentuales [%]” como en “partes por millón [ppm]”, y la relación entre ambos, a través de la siguiente tabla:

Porcentual	[%]	0,5	0,2	0,1	0,05	0,025	0,02	0,01
Partes por millón	[ppm]	5000	2000	1000	500	250	200	100

Tabla 3: Tolerancias normalizadas de los resistores en valores porcentuales [%] y en partes por millón [ppm], y la equivalencia entre ambos

Bancos de memoria del LPC2114

Tiene un tamaño de 16[Mb], y se distribuye de la siguiente manera:

Banco	Direcciones	
	Inicio	Final
0	80000000	80FFFFFF
1	81000000	81FFFFFF
2	82000000	82FFFFFF
3	83000000	83FFFFFF

Tabla 4: Distribución de los bancos de memoria del LPC2114

Parte 2: Resolución de exámenes finales

Aquí he intentado ser lo más conciso y directo en las resoluciones, sin divagar en las mismas y tratando de hacerlo paso por paso.

Una vez que se comprende la mecánica de resolución de los ejercicios, ya prácticamente uno sabe qué hacer.

Casi todos los ejercicios son similares en la forma de encararlos, por más que las situaciones del problema sean cambiantes. Hay que saber interpretar bien y correctamente la situación presentada, sin dejar de lado ningún detalle.

La presentación del enunciado de cada examen, como así su formato, lo transcribí literalmente de las copias originales de cada uno, para evitar las típicas copias de copias de copias, en las cuales es bastante difícil de leer y deducir; me pareció lo más apropiado.

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 30 de agosto de 2007.

Se tiene una señal proveniente de un puente de galgas extensiometrías, destinado a obtener información de peso en una balanza analítica. La carga máxima es de 50[g] y se requiere resolución de 1[mg].

Esta información existe en forma de una señal de CC, de $2[\mu\text{V}]$, que tiene superpuesto ruido de distribución blanca. Para filtrar el ruido se usa un algoritmo de promediación de 16[muestras], equivalente a un filtro PB de ocho polos. Es necesario preservar la integridad de la señal, por lo que se usa un sample-hold. Las especificaciones de esta etapa son:

Ancho de banda de la entrada: $10[\text{MHz}]$

Tiempo de apertura: $20[\text{ns}]$

Tiempo de retención: $8[\mu\text{s}]$

Como primer punto de este examen debe Ud. diseñar el circuito de sample-hold, con su base de tiempos y su lógica de comando (sample).

Se necesita que el conversor entregue un incremento de salida por cada [mg]. Para estas condiciones debe Ud. determinar:

- La ganancia de la etapa de acondicionamiento de señal y un bosquejo de su diseño, especificando componentes.
- El valor de la fuente de referencia del conversor a/d (del tipo multiplicador).
- El tiempo de conversión máximo que debe especificarse para el a/d.

Presente un esquema del circuito que, a partir de un flanco ascendente de una variable [pesar], genere 16 ciclos de muestreo de la señal y su lectura a intervalos iguales de $500[\mu\text{s}]$, pidiendo interrupción al procesador cada vez.

Confeccione el mapa y el esquema básico de su circuito total.

Resolución

Datos:

- Carga máxima: 50[g].
- Resolución: 1[mg].
- Paso de conversión: 2[μ V/mg].
- $\Delta B=10[\text{MHz}]$.
- $t_s=20[\text{ns}]$.
- $T_m=8[\mu\text{s}]$.
- $V_{ref}=10[\text{V}]$.

Cálculos:

$$\text{cuentas} = \frac{\text{car.máx.}}{\text{resolución.}} = \frac{50[\text{g}]}{1[\text{mg}]} \Rightarrow \text{cuentas} = 50000[\text{cuentas}] \quad (1)$$

$$\begin{aligned} n_conv &= \frac{\ln(\text{cuentas})}{\ln(2)} = \frac{\ln(5000)}{\ln(2)} \Rightarrow n_conv = 15,61[\text{bits}] \Rightarrow \\ &\Rightarrow n_conv \geq 16[\text{bits}] \end{aligned} \quad (2)$$

Por lo tanto, se deberá implementar un **Conversor de 16[bits]**.

Y la ganancia será:

$$G = \frac{V_{LSB}}{V_{mín_conv}} = \frac{\frac{V_{ref}}{2^{n_conv}}}{\frac{V_{mín_conv}}{2[\mu\text{V}]}} = \frac{\frac{10[\text{V}]}{2^{16}}}{2[\mu\text{V}]} \Rightarrow G = 76,294[\text{veces}] \quad (3)$$

El circuito del puente y la adaptación de la ganancia son los siguientes:

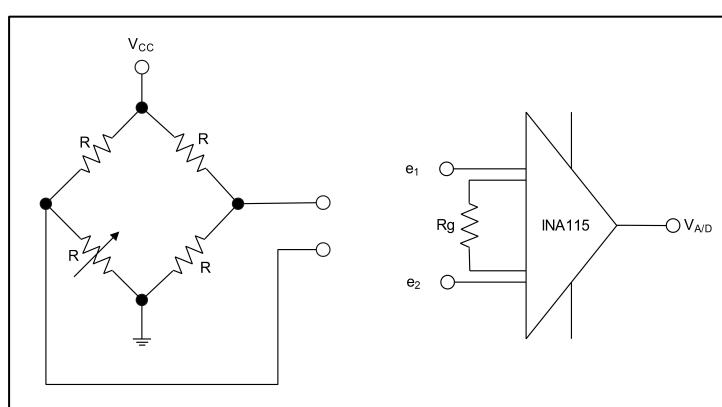


Figura 69: -Final 30.08.07- Circuito del puente y adaptación de ganancia

Donde en el INA115 “ R_g ” se calcula:

$$G = 1 + \frac{50[k\Omega]}{R_g} \Rightarrow$$

$$\Rightarrow R_g = \frac{50[k\Omega]}{G - 1} = \frac{50[k\Omega]}{76,294[\text{veces}] - 1} \Rightarrow R_g = 664,1[\Omega]$$
(4)

El circuito del sample-hold es:

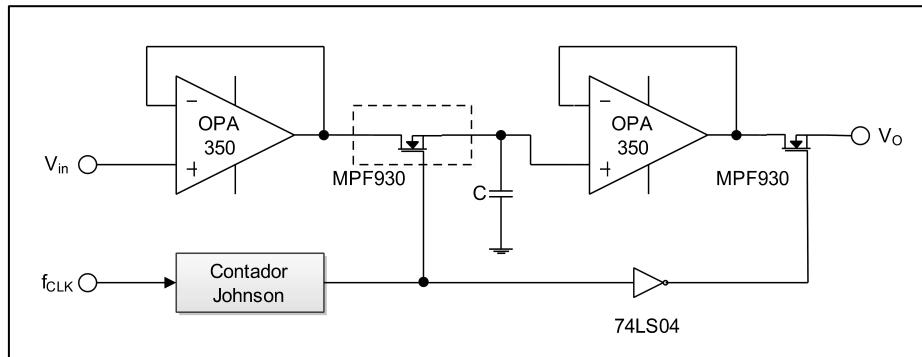


Figura 70: -Final 30.08.07- Circuito del sample-hold

Y los cálculos son:

$$\Delta B|_{OPA350} = 35[MHz] \quad (5)$$

$$t_m = t_s + t_h \Rightarrow t_h = t_m - t_s = 8[\mu s] - 20[ns] \Rightarrow t_h = 7,98[\mu s] \quad (6)$$

Luego:

$$Duty Cycle = D = \frac{t_s}{t_m} = \frac{20[ns]}{8[\mu s]} \Rightarrow D \cong \frac{1}{400} \quad (7)$$

Por lo tanto, de (7), se tiene que el módulo del contador es **N_{contador}=400**.

Luego:

$$f_{CLK} = \frac{1}{t_s} = \frac{1}{20[ns]} \Rightarrow f_{CLK} = 50[MHz] \quad (8)$$

Luego, del teórico se tiene que:

$$t_s = 9 \cdot \tau = 9 \cdot (R_{ds-ON} \cdot C) + t_{ON-MPF930} \Rightarrow$$

$$\Rightarrow C = \frac{t_s - t_{ON-MPF930}}{9 \cdot R_{ds-ON}} \quad (9)$$

Donde:

$$R_{ds-ON} \cong 1[\Omega] \quad \wedge \quad t_{ON-MPF930} = 7[ns] \quad (10)$$

Por lo tanto:

$$C = \frac{20[nS] - 7[nS]}{9.1[\Omega]} \Rightarrow C = 1,444[nF] \quad (11)$$

Además:

$$f_m = \frac{1}{t_m} = \frac{1}{8[\mu s]} \Rightarrow f_m = 125[kHz] \quad (12)$$

También se saca como conclusión:

$$t_{CONV}|_{máx} = 8[\mu s] \quad (13)$$

A continuación se realiza el último punto solicitado:

$$\begin{aligned} 16 \text{ muestras} &= \text{Cant.muestras}.t_{CONV}|_{máx} = 16.8[\mu s] \Rightarrow \\ &\Rightarrow 16 \text{ muestras} = 128[\mu s] \end{aligned} \quad (14)$$

A continuación se muestran los diagramas de flujo correspondientes:

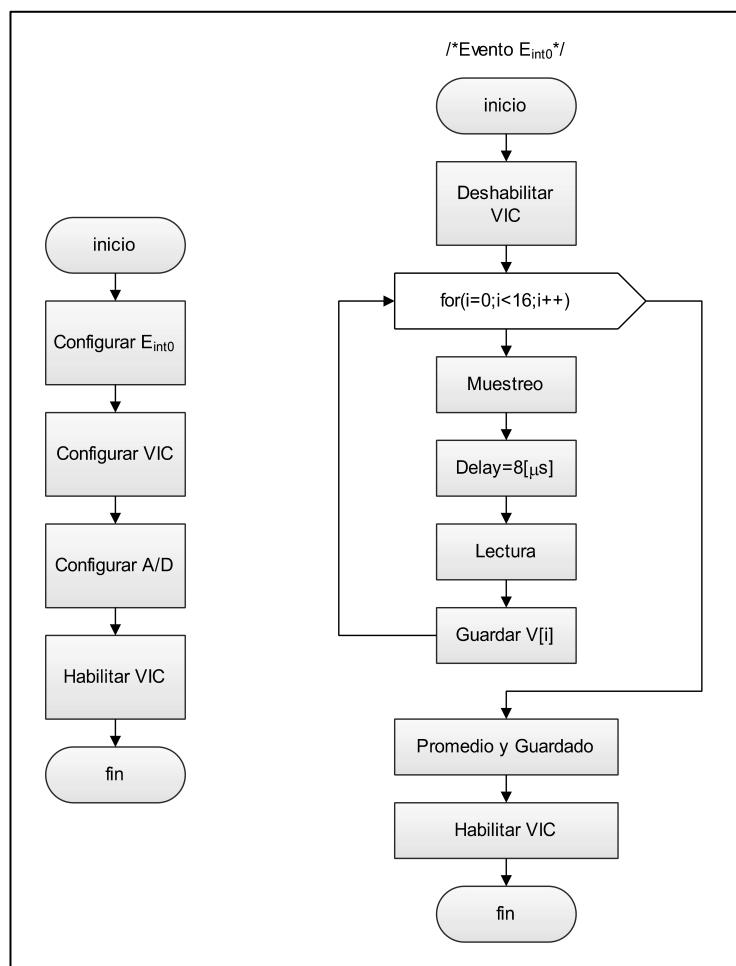


Figura 71: -Final 30.08.07- Diagramas de flujo de las rutinas (fuente de interrupción 14)

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 20 de diciembre de 2007.

1. A partir de un conversor D/A de 16[bits], con referencia de 5,0[V]. Ud. deberá implementar un conversor A/D de 12[bits]. Todos los circuitos deberán estar mapeados a partir de 300h. Se pide:
 - Circuito a usar, con componentes detallados.
 - Diagrama de flujo detallado del funcionamiento.
 - ¿Cuál es el valor máximo de la tensión de entrada a medir?
 - ¿Cómo se obtiene el tiempo de conversión?
2. Para el caso anterior, diseñe el acondicionador de señal adecuado para procesar la información de un puente de termistores cuya señal de salida es $15[\mu\text{V}/^\circ\text{C}]$, para un rango de $0\text{-}40[^\circ\text{C}]$ y una discriminación de $0,01[^\circ\text{C}]$. no olvide la posibilidad de corregir el cero. Especifique componentes, incluyendo tolerancias de resistores.

NOTA: Es similar al examen del 5 de febrero de 2009.

Resolución

Datos:

- $\Delta T: 40[^\circ\text{C}]$.
- Resolución: $0,01[^\circ\text{C}]$.
- Paso de conversión: $15[\mu\text{V}/^\circ\text{C}]$.
- $N_{A/D}=12[\text{bits}]$.
- $V_{ref}=5[\text{V}]$.

Cálculos:

$$t_{CONV} = t_{CLK} \cdot (N + 1) \Rightarrow t_{CONV} = 13 \cdot t_{CLK} \quad (1)$$

$$\begin{aligned} \Delta T = 40[^\circ\text{C}] \Rightarrow \text{cuentas} &= \frac{\Delta T}{\text{resolución}} = \frac{40[^\circ\text{C}]}{0,01[^\circ\text{C}]} \Rightarrow \\ \Rightarrow \text{cuentas} &= 4000[\text{cuentas}] \end{aligned} \quad (2)$$

$$n = \frac{\ln(\text{cuentas})}{\ln(2)} = \frac{\ln(4000)}{\ln(2)} \Rightarrow n = 11,96[\text{bits}] \Rightarrow n \cong 12[\text{bits}] \quad (3)$$

$$\begin{aligned} V_{paso} &= \text{Paso de conversión.resolución} = 15 \left[\frac{\mu\text{V}}{^\circ\text{C}} \right] \cdot 0,01[^\circ\text{C}] \Rightarrow \\ \Rightarrow V_{paso} &= 150[n\text{V}] \end{aligned} \quad (4)$$

$$V_{LSB} = \frac{V_{ref}}{2^n} = \frac{5[V]}{2^{12}} \Rightarrow V_{LSB} = 1,221[m\text{V}] \quad (5)$$

Por lo tanto, la ganancia que deberá tener el AO, en este caso un AI, será:

$$G = \frac{V_{LSB}}{V_{paso}} = \frac{1,221[m\text{V}]}{150[n\text{V}]} \Rightarrow G = 8140[\text{veces}] \quad (6)$$

El AI que se utilizará, es el siguiente:

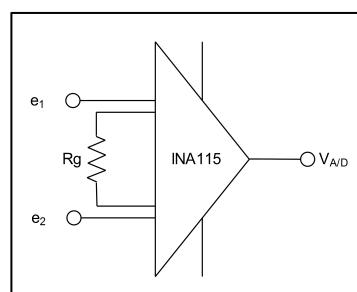


Figura 1: -Final 20.12.07- AI utilizado

El cual responde a la siguiente ecuación:

$$\begin{aligned} G &= 1 + \frac{50[k\Omega]}{R_g} \Rightarrow \\ \Rightarrow R_g &= \frac{50[k\Omega]}{G - 1} = \frac{50[k\Omega]}{8140[veces] - 1} \Rightarrow R_g = 6,143[\Omega] \end{aligned} \quad (7)$$

Ahora se calcularán las tolerancias a utilizar en los componentes, de acuerdo al *Error de Ganancia* (ε_G), de la siguiente manera:

$$\varepsilon_T = \varepsilon_{A/D} + \varepsilon_{Vref} + \varepsilon_G \quad (8)$$

Donde cada uno de ellos es:

$$\varepsilon_{A/D} = \frac{1}{2} V_{LSB} = \frac{1}{2} \cdot 1,221[mV] \Rightarrow \varepsilon_{A/D} = 610,5[\mu V] \quad (9)$$

$$\varepsilon_{Vref} = \varepsilon_G = \frac{1}{2} \varepsilon_{A/D} = \frac{1}{2} \cdot 610,5[\mu V] \Rightarrow \varepsilon_{Vref} = \varepsilon_G = 305,25[\mu V] \quad (10)$$

Por lo tanto:

$$\begin{aligned} \text{tolerancia} &\leq \frac{\varepsilon_G}{V_{ref}} = \frac{305,25[\mu V]}{5[V]} \Rightarrow \text{tolerancia} = 61,05[ppm] \\ \Rightarrow \text{tolerancia} &\Big|_{\text{normalizada}} = 50[ppm] \end{aligned} \quad (11)$$

Para la corrección del cero del puente se puede utilizar un AO en configuración diferencial:

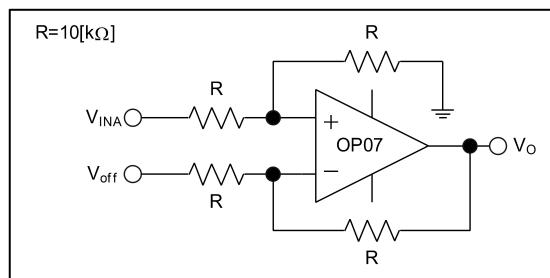


Figura 2: -Final 20.12.07- Circuito de corrección del cero del puente

El cual responde a la siguiente fórmula:

$$V_o = V_{INA} + V_{off} \quad (12)$$

Para la regulación de la tensión de offset (V_{off}), se utiliza el siguiente circuito:

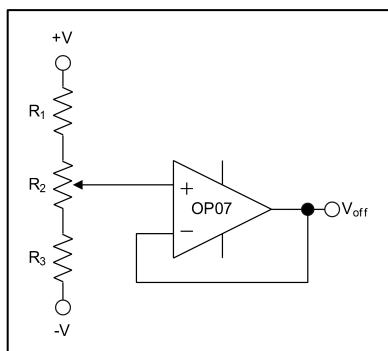


Figura 3: -Final 20.12.07- Circuito para la regulación de la tensión de offset (V_{off})

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 28 de febrero de 2008.

1. Se debe construir una balanza con capacidad de pesar hasta 5[kg] con resolución de 0,1[g]. ¿Qué conversor debe usarse para ello? (nº de bits).
2. Para el caso anterior la señal de peso se obtiene con un puente de galgas extensiométricas que tiene una sensibilidad de 0,012[μ V/g]. ¿Qué ganancia debe tener el amplificador de modo que con una referencia de 5,0[V] un escalón de salida del conversor represente 0,1[g]?
3. Para la pregunta anterior, ¿Cuál es el error admisible de ganancia?
4. ¿Qué tolerancia deben tener los resistores usados en el amplificador para asegurar el error calculado en la pregunta 3?
5. Para una señal $v(t) = 6,4 \cdot \text{sen}(0,72E6 \cdot t)$, ¿Cuál será el tiempo de apertura (=tiempo de adquisición) de un sample/hold destinado a capturar y retener la información? Justificar en no más de tres renglones.
6. Para la señal de 5, ¿Cuál es el tiempo de conversión mínimo del A/D a utilizar?
7. Para la señal de 5, se requiere una resolución de 12[bits], ¿Qué conversor se debe usar? (especifique tipo, referencia a usar y modo de operación).
8. ¿Qué es una máscara de interrupción, cómo se la controla y para qué se usa?
9. Con la hoja de datos adjunta de la PPI i8255, que está mapeada a partir de 300h, escriba un programa en el Assembler del 8088 que programe los puertos A y B como salidas y el C como entradas en el modo cero.
10. ¿Cómo se usa un salto condicional en modo FAR, es decir, fuera del segmento corriente?
11. ¿Qué diferencia existe entre los saltos condicionales lógicos y los aritméticos?, dar dos ejemplos de cada uno.
12. Describa sucintamente los modelos usuales en operación en punto flotante.

Por lo menos seis de las preguntas deben estar correctas para aprobar. Esas seis deben incluir al menos dos del conjunto {2, 5, 9}.

NOTA: Resuelto hasta el punto 7.

Resolución

Cálculos:

1. Capacidad=5[kg].

Resolución=0,1[g].

$$cuentas = \frac{capacidad}{resolución} = \frac{5[kg]}{0,1[g]} \Rightarrow cuentas = 50000[cuentas] \quad (1)$$

$$n = \frac{\ln(cuentas)}{\ln(2)} = \frac{\ln(50000)}{\ln(2)} \Rightarrow n = 15,61[bits] \quad (2)$$

Por lo tanto se necesita un **Conversor de 16[bits]**.

2. $V_{ref}=5,0[V]$.

Sensibilidad=0,012[$\mu V/g$].

Escalón de salida=0,1[g].

$$V_{LSB} = \frac{V_{ref}}{2^n} = \frac{5,0[V]}{2^{16}} \Rightarrow V_{LSB} = 76,294[\mu V] \quad (3)$$

$$V_{paso} = \text{escalón_de_salida}.\text{sensibilidad} = 0,1[g].0,012\left[\frac{\mu V}{g}\right] \Rightarrow \\ \Rightarrow V_{paso} = 1,2[nV] \quad (4)$$

$$G = \frac{V_{LSB}}{V_{paso}} = \frac{76,294[\mu V]}{1,2[nV]} \Rightarrow G = 61578[veces] \quad (5)$$

La ganancia es muy elevada para una sola etapa, por lo tanto se implementarán 2 AI similares. Para calcular la ganancia para cada uno, se realiza el siguiente cálculo:

$$G = G_1 \cdot G_2 \quad (6)$$

Ahora, si se considera:

$$G_1 = G_2 \quad (7)$$

Tomando (6), se puede hacer:

$$G = G_1^2 \Rightarrow G_1 = \sqrt{G} = \sqrt{61578[veces]} \Rightarrow G_1 = 252,147[veces] \quad (8)$$

Que es el valor de ganancia para cada uno de los 2 AI.

Los 2 AI que se utilizarán son similares, como el de la siguiente figura:

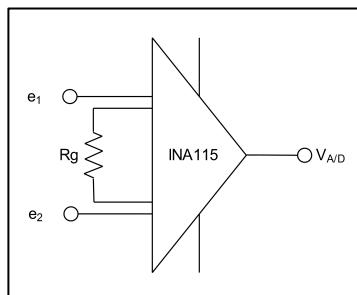


Figura 1: -Final 28.02.08- Modelo de AI utilizado

El cual responde a la siguiente fórmula:

$$\begin{aligned} G_1 = 1 + \frac{50[k\Omega]}{R_g} &\Rightarrow R_g = \frac{50[k\Omega]}{G_1 - 1} = \frac{50[k\Omega]}{252,147[veces] - 1} \Rightarrow \\ &\Rightarrow R_g = 199,087[\Omega] \end{aligned} \quad (9)$$

$$3. \quad \mathcal{E}_G = \frac{1}{4}V_{LSB} = \frac{1}{4}79,294[\mu V] \Rightarrow \mathcal{E}_G = 19,074[\mu V] \quad (10)$$

$$4. \quad \text{tolerancia} = \frac{\mathcal{E}_G}{V_{ref}} = \frac{19,074[\mu V]}{5[V]} \Rightarrow \text{tolerancia} = 3,815[ppm] \quad (11)$$

5. El análisis parte de la fórmula de la señal a estudiar, y luego se desarrolla, como se muestra a continuación:

$$v(t) = 6,4 \cdot \sin(0,72 \times 10^6 \cdot t) \quad (12)$$

Por lo tanto, en su forma genérica, la (12) es:

$$v(t) = V \cdot \sin(\omega \cdot t) \quad (13)$$

Y lo que se debe averiguar es la máxima variación de (13), para lo cual a ésta se la deriva y luego se iguala a "1", obteniendo la máxima variación, de la siguiente manera:

$$\frac{dv(t)}{dt} = \omega \cdot V \cdot \cos(\omega \cdot t) \quad (14)$$

Por lo tanto, la máxima variación se dará para:

$$\cos(\omega \cdot t) = 1 \quad (15)$$

Teniendo en cuenta (15), la (14) queda:

$$\frac{\Delta V}{t_s} = \omega \cdot V \Rightarrow \Delta V = \omega \cdot V \cdot t_s \quad (16)$$

Ahora, teniendo en cuenta el *Error Total de Conversión* (ε):

$$\varepsilon < \frac{\Delta V}{2V} \quad (17)$$

Reemplazando el valor de “ ΔV ” de (16) en (17) queda:

$$\varepsilon < \frac{\omega V t_s}{2V} = \frac{\omega t_s}{2} = \frac{1}{2} \cdot \frac{1}{2^n} \quad (18)$$

Por lo tanto, tomando los 2 términos de (18) y haciendo pasos algebraicos:

$$t_s < \frac{\frac{1}{2} \cdot \frac{1}{2^n} \cdot 2}{\omega} \Rightarrow t_s < \frac{1}{2\pi f \cdot 2^n} \quad (19)$$

Para el caso particular del punto requerido:

$$t_s < \frac{V_{LSB}}{\omega} = \frac{76,294[\mu V]}{0,72 \times 10^6 \left[\frac{rad}{s} \right]} \Rightarrow t_s < 105,964[ps] \cong 106[ps] \quad (20)$$

$$6. \quad f_m = 5 \cdot f_s \quad (21)$$

Donde:

$$f_s = \frac{\omega}{2\pi} = \frac{0,72 \times 10^6 \left[\frac{rad}{s} \right]}{2\pi} \Rightarrow f_s = 114,592[kHz] \quad (22)$$

Reemplazando (22) en (21):

$$f_m = 5 \cdot f_s = 5 \cdot 114,592[kHz] \Rightarrow f_m = 572,960[kHz] \quad (23)$$

Por lo tanto:

$$t_c = \frac{1}{f_m} = \frac{1}{572,960[kHz]} \Rightarrow t_c = 1,745[\mu s] \quad (24)$$

7. Se utilizará un conversor de aproximaciones sucesivas, de 12[bits] y 5[V]:

$$t_{CLK} = \frac{t_c}{n+1} = \frac{1,745[\mu s]}{12[bits]+1} \Rightarrow t_{CLK} = 134,231[ns] \quad (25)$$

$$f_{CLK} = \frac{1}{t_{CLK}} = \frac{1}{134,231[ns]} \Rightarrow f_{CLK} = 7,45[MHz] \quad (26)$$

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 22 de mayo de 2008.

1. Se dispone de un ADC de 16[bits] con el cual se desea medir la señal producida por un sensor de presión con un rango de 0 a 100[kPa] y 40[mV] a fondo de escala.
 - (1 pto) Si se posee un amplificador de ganancia 25 ¿qué tensión de referencia necesito en el ADC para medir con una resolución de 10[Pa] por cuenta?
 - (1 pto) ¿Cuál es el menor paso de resolución (en entero) que puedo lograr? –Se refiere al sensor-.
 - (1 pto) Sabiendo que el sensor posee una precisión de 0,25[%] a fondo de escala, ¿cuál sería la cantidad de bits óptima del ADC?
2. (3 pto) Para el ADC hipotético mostrado en la figura 1 (Que nunca apareció), diseñar una interfaz con un microprocesador Intel 8088 que permita al microprocesador leer una conversión y lanzar una nueva y por parte del ADC, notificar al mircroprocesador a través de una interrupción que la conversión en curso ha finalizado.
Diseñar el mapeo de forma tal que quede el ADC ubicado en la dirección 300h del I/O.
3. (1 pto) Se dispone de dos variables de tipo word (var1 y var2) y una tercera (var3), se debe sumar var1 a var3, y luego var2 a var3, realizar el programa en assembler que efectúe esta operación, primero considerando que las tres variables son sin signo y luego repetir el programa pero ahora para variables con signo.
4. (1 pto) Enumere las instrucciones de cadena junto con los registros y banderas que usa.
5. (1 pto) Escriba un programa que cople (¿cople?) 200h word ubicado en la posición 5000h del segmento de datos a la dirección 6000h del mismo segmento utilizando instrucciones de cadena.
6. (1 pto) Realice un mapa donde detalle las posiciones relativas donde se guardaron AX, VAR1 y el IP con respecto al SP valuado en el punto A.

```
push AX
push var1
call suma
...
suma PROC
    ←a
    ...
ret
Suma ENDP
```

NOTA: Resueltos puntos 1. Y 3.

Resolución

Cálculos:

1. ADC=16[bits].

Rango medición sensor=0 a 100[kPa].

Tensión a fondo de escala=40[mV].

$G_{AO}=25[\text{veces}]$.

Resolución por cuenta=10[Pa].

Precisión sensor=25[%] a fondo de escala.

$$1.1. \quad \left. \text{cuentas} \right|_{\text{posibles}} = 2^n = 2^{16} \Rightarrow \left. \text{cuentas} \right|_{\text{posibles}} = 65536[\text{cuentas}] \quad (1)$$

$$\begin{aligned} \left. \text{cuentas} \right|_{\text{usadas}} &= \frac{\text{rango sensor}}{\text{resolución por cuenta}} = \frac{100[\text{kPa}]}{10[\text{Pa}]} \Rightarrow \\ &\Rightarrow \left. \text{cuentas} \right|_{\text{usadas}} = 10000[\text{cuentas}] \end{aligned} \quad (2)$$

Por lo tanto, aplicando regla de 3 simple:

$$\begin{aligned} 100[\text{kPa}] &\quad 40[\text{mV}] \\ 10[\text{Pa}] &\quad V_{\text{paso}} \Rightarrow \\ \Rightarrow V_{\text{paso}} &= \frac{10[\text{Pa}].40[\text{mV}]}{100[\text{kPa}]} \Rightarrow V_{\text{paso}} = 4[\mu\text{V}] \end{aligned} \quad (3)$$

Aplicando a este valor la Ganancia del AO (G_{AO}):

$$V_{LSB} = V_{\text{paso}} \cdot G_{AO} = 4[\mu\text{V}].25[\text{veces}] \Rightarrow V_{LSB} = 100[\mu\text{V}] \quad (4)$$

Finalmente, el valor de la tensión de referencia (V_{ref}) para estas condiciones es:

$$V_{LSB} = \frac{V_{ref}}{2^n} \Rightarrow V_{ref} = V_{LSB} \cdot 2^n = 100[\mu\text{V}] \cdot 2^{16} \Rightarrow V_{ref} = 6,554[\text{V}] \quad (5)$$

$$\begin{aligned} 1.2. \quad \text{resolución}_{\text{calculada}} &= \frac{\text{rango sensor}}{2^n} = \frac{100[\text{kPa}]}{2^{16}} \Rightarrow \\ &\Rightarrow \text{resolución}_{\text{calculada}} = 1,526[\text{Pa}] \Rightarrow \text{resolución}_{\text{entera}} = 2[\text{Pa}] \end{aligned} \quad (6)$$

Nuevamente aplicando regla de 3 simple:

$$\begin{aligned}
 & 100[kPa] \text{ ____ } 40[mV] \\
 & 2[Pa] \text{ ____ } V_{paso} \Rightarrow \\
 & \Rightarrow V_{paso} = \frac{2[Pa].40[mV]}{100[kPa]} \Rightarrow V_{paso} = 800[nV]
 \end{aligned} \tag{7}$$

Por lo tanto, el nuevo “ V_{LSB} ” será:

$$V_{LSB} = V_{paso}.G_{AO} = 800[nV].25[veces] \Rightarrow V_{LSB} = 20[\mu V] \tag{8}$$

Finalmente, el valor de la tensión de referencia (V_{ref}) para estas condiciones es:

$$V_{LSB} = \frac{V_{ref}}{2^n} \Rightarrow V_{ref} = V_{LSB}.2^n = 20[\mu V].2^{16} \Rightarrow V_{ref} = 1,311[V] \tag{9}$$

1.3. Teniendo en cuenta la ecuación (2) –Mi resolución difiere de la del apunte-:

$$\begin{aligned}
 & 100[\%] \text{ _____ } 10000[cuentas] \\
 & 25[\%] \text{ _____ } resolución_{real} \Rightarrow \\
 & resolución_{real} = \frac{0,25.10000[cuentas]}{1} \Rightarrow resolución_{real} = 2500[cuentas]
 \end{aligned} \tag{10}$$

$$n = \frac{\ln(resolución_{real})}{\ln(2)} = \frac{\ln(2500[cuentas])}{\ln(2)} \Rightarrow n = 11,288[bits] \tag{11}$$

Por lo tanto, para estas condiciones se utilizará un **Conversor A/D de 12[bits]**.

$$precisión_{A/D} = \frac{1}{2^n} = \frac{1}{2^{12}} \Rightarrow precisión_{A/D} = 0,024414[\%] \tag{12}$$

3. Sin signo:

reset:

```
mov r0,#VAR1
mov r1,#VAR2
mov r2,#VAR3
```

suma:

```
ldr r3,[r0]
ldr r4,[r1]
add r5,r3,r4
str r5,[r2]
```

loop: b loop

Con signo:

reset:

```
    mov r0,#VAR1
    mov r1,#VAR2
    mov r2,#VAR3
```

suma:

```
    ldrsh r3,[r0]
    ldrsh r4,[r1]
    add  r5,r3,r4
    str   r5,[r2]
```

loop: b loop

*/ Constantes

VAR1: .half

VAR2: .half

VAR3: .half

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 24 de julio de 2008.

Se tiene una balanza destinada a pesar personas. Ésta tiene cuatro puntos de apoyo y en cada uno de ellos hay un puente de “strain gages” para medir el peso. El peso resultante es, entonces, la suma de los pesos individuales de cada apoyo. La resolución de la balanza es de 50[g] sobre un máximo admisible de 150[kg]. Ud. deberá implementar los circuitos adecuados para funcionamiento, que deberán estar mapeados a partir de 300h. la salida de cada puente es de 50[μ V/kg].

1. ¿Qué ganancia debe tener el amplificador en cada puente para que, usando referencia estándar (5,0[V]) del conversor, cada incremento del conversor represente 0,05[kg]?
2. ¿Cuál es el error de ganancia admisible para el conversor a usar?
3. ¿Qué tolerancia deben tener los resistores para garantizar ese error?
4. Diseñe este amplificador. Especifique componentes, incluyendo tolerancias de resistores.
5. Diseñe el multiplexor usado, con detalle de sus componentes.
6. Diseñe un circuito que, a partir de usar un reloj e interrupciones, permita tomar 16 muestras sucesivas de cada canal de “strain gages”.
7. Escriba un programa para tomar esas 16 muestras a partir del circuito de 6.
8. Escriba un programa para realizar las operaciones aritméticas adecuadas (debe especificar cuáles son) para dar el peso final. –Cálculo del promedio-

Resolución

Datos:

Cant. de “strain gage”=4[un].

Resolución balanza=50[g].

Peso máximo balanza=150[kg].

Salida de cada “strain gage”=50[μ V/kg].

$V_{ref}=5,0[V]$.

Paso=0,05[kg].

Cálculos:

$$1. \quad Peso\ máx|_{galga} = \frac{Peso\ máx|_{balanza}}{Cant.\ strain\ gage} = \frac{150[kg]}{4[un]} \Rightarrow \\ \Rightarrow Peso\ máx|_{galga} = 37,5[kg] \quad (1)$$

$$Peso\ mín|_{galga} = \frac{resolución|_{balanza}}{Cant.\ strain\ gage} = \frac{50[g]}{4[un]} \Rightarrow \\ \Rightarrow Peso\ mín|_{galga} = 12,5[g] \quad (2)$$

$$cuentas = \frac{Peso\ máx|_{galga}}{Peso\ mín|_{galga}} = \frac{37,5[kg]}{12,5[g]} \Rightarrow cuentas = 3000[cuentas] \quad (3)$$

$$n_{A/D} = \frac{\ln(cuentas)}{\ln(2)} = \frac{\ln(3000[cuentas])}{\ln(2)} \Rightarrow n_{A/D} = 11,551[bits] \Rightarrow \\ \Rightarrow n_{A/D} = 12[bits] \quad (4)$$

$$V_{LSB} = \frac{V_{ref}}{2^{n_{A/D}}} = \frac{5,0[V]}{2^{12}} \Rightarrow V_{LSB} = 1,221[mV] \quad (5)$$

$$V_{paso_galga} = salida_{galga} \cdot Peso\ mín|_{galga} = 50[\mu V / kg] \cdot 12,5[g] \Rightarrow \\ \Rightarrow V_{paso_galga} = 625[nV] \quad (6)$$

$$G = \frac{V_{LSB}}{V_{paso_galga}} = \frac{1,221[mV]}{625[nV]} \Rightarrow G = 1954[veces] \quad (7)$$

2. $\varepsilon_G = \frac{1}{2} \cdot V_{LSB} = \frac{1}{2} \cdot 1,221[mV] \Rightarrow \varepsilon_G = 610,5[\mu V]$ (8)

3. $tolerancias = \frac{\varepsilon_G}{V_{ref}} = \frac{610,5[\mu V]}{5,0[V]} \Rightarrow tolerancias = 122,1[ppm] \Rightarrow$
 $\Rightarrow tolerancias = 100[ppm]$ (9)

4. El amplificador a utilizar es un INA115, que es un Amplificador de Instrumentación (AI), el cual tiene la siguiente configuración:

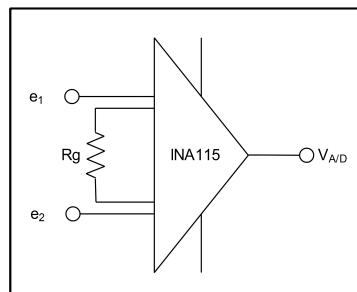


Figura 1: -Final 24.07.08- Modelo de AI utilizado

El cual responde a la siguiente función de transferencia, de donde se hallará “R_g”:

$$G = 1 + \frac{50[k\Omega]}{R_g} \Rightarrow R_g = \frac{50[k\Omega]}{G-1} = \frac{50[k\Omega]}{1954[veces]-1} \Rightarrow R_g = 25,602[\Omega]$$
 (10)

5. El multiplexado a utilizar es el siguiente:

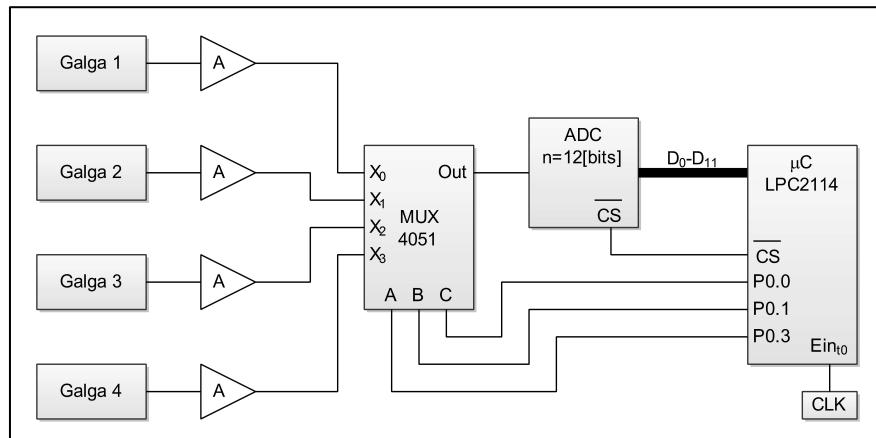


Figura 2: -Final 24.07.08- Multiplexado utilizado

6. Se toma el tiempo entre lecturas (t_{el}) de 25[μs], por lo tanto, las 16 muestras por canal (t_{mu}) es:

$$t_{mu} = cant_{muestras} \cdot t_{el} = 16[muestras] \cdot 25[\mu s] \Rightarrow t_{mu} = 400[\mu s]$$
 (11)

Por lo tanto, el tiempo total del ciclo (t_{ciclo}) es:

$$t_{ciclo} = cant_{canales}.t_{mu} = 4[\text{canales}].400[\mu\text{s}] \Rightarrow t_{ciclo} = 1,6[\text{ms}] \quad (12)$$

Por lo que se tomará:

$$t_{ciclo} = 2[\text{ms}] \quad (13)$$

Por lo tanto, la frecuencia de reloj (f_{CLK}) será:

$$f_{CLK} = \frac{1}{t_{ciclo}} = \frac{1}{2[\text{ms}]} \Rightarrow f_{CLK} = 500[\text{Hz}] \quad (14)$$

El generador de clock que se utilizará es un LM555 en configuración “Astable” (oscilador), como muestra la siguiente figura:

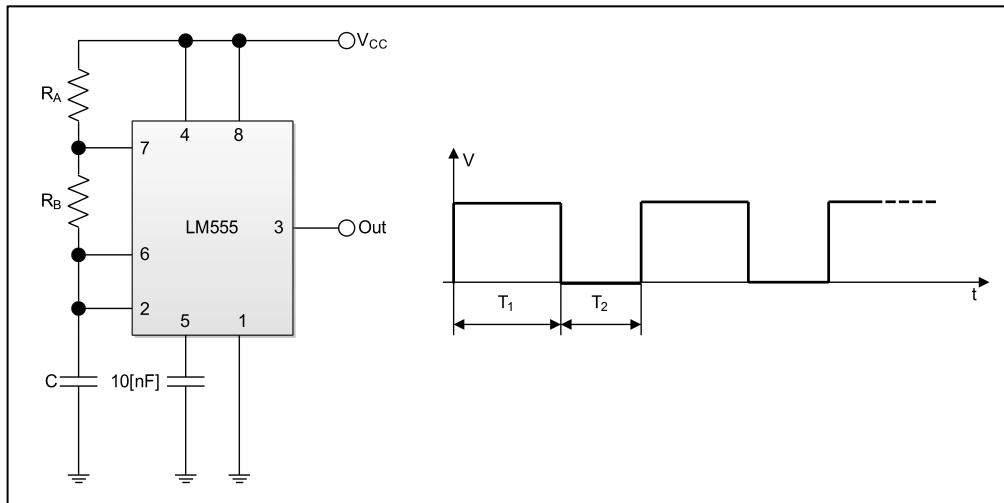


Figura 3: -Final 24.07.08- Configuración astable (oscilador) del LM555 como generador de clock

El cual responde a la siguiente función de transferencia:

$$f = \frac{1,44}{(R_A + 2.R_B).C} = 500[\text{Hz}] \Rightarrow 2,88[\text{ms}] = (R_A + 2.R_B).C \quad (15)$$

Por lo tanto, seleccionando el siguiente juego de valores de los componentes:

- $C=100[\text{nF}]$.
- $R_A+2.R_B=28,8[\text{k}\Omega]$.
- $R_B=12[\text{k}\Omega]$.
- $R_A=4,8[\text{k}\Omega] \Rightarrow R_A=4,7[\text{k}\Omega]$.

Ahora, reconfirmando el valor de (14) aplicando en (15) estos valores:

$$f = \frac{1,44}{(R_A + 2.R_B).C} = \frac{1,44}{(4,7[k\Omega] + 2.12[k\Omega]).100[nF]} \Rightarrow \\ \Rightarrow f = 501,742[Hz]$$
(16)

Finalmente reconfirmando valores entre (12) y (13):

$$t_{ciclo} = \frac{1}{f} = \frac{1}{501.742[Hz]} \Rightarrow t_{ciclo} = 1,993[ms] \quad (17)$$

Valor que se encuentra entre los valores de (12) y (13), validando la configuración.

7. A continuación se muestra el diagrama de flujo del programa a utilizar:

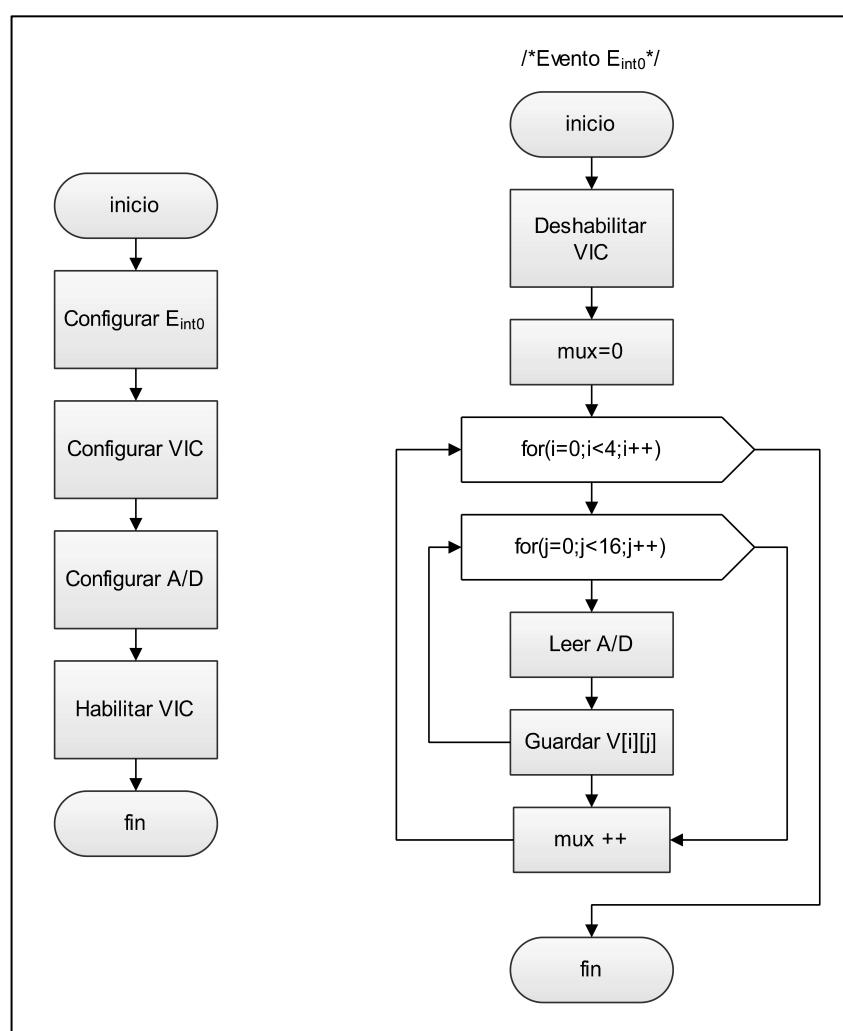


Figura 4: -Final 24.07.08- Diagrama de flujo del programa utilizado

8. A continuación se muestra el diagrama de flujo del programa utilizado para la medición de peso final:

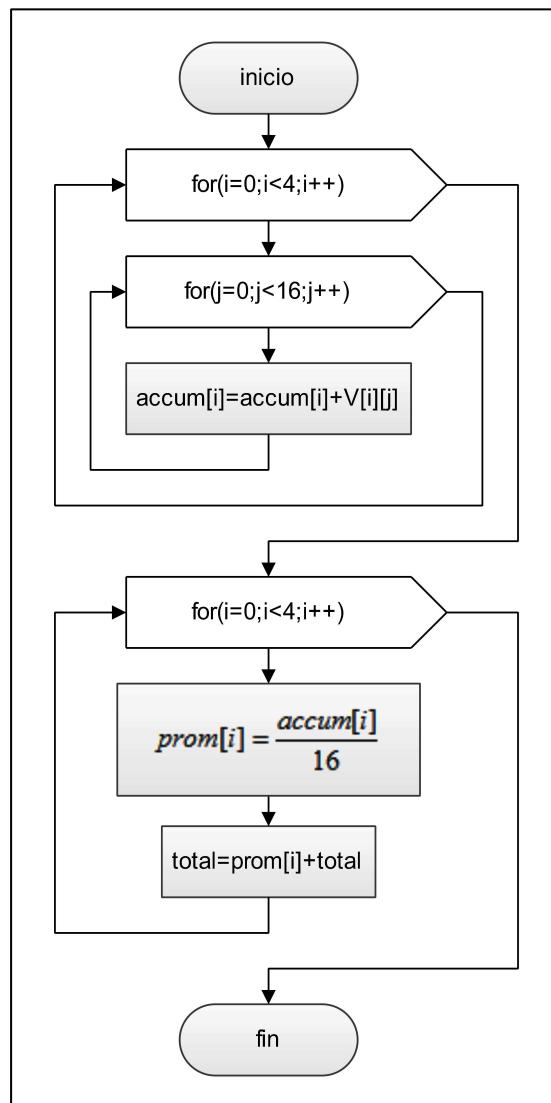


Figura 5: -Final 24.07.08- Diagrama de flujo del programa utilizado para la medición de peso final

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 09 de octubre de 2008.

Para medir la humedad relativa del aire se usa un instrumento llamado “psicrómetro”, que emplea dos termómetros iguales, uno de ellos embebido en agua destilada. La evaporación del agua absorbe calor del bulbo de este último y entonces la diferencia de temperaturas entre los dos termómetros es la base para calcular el valor de humedad.

Para implementar un dispositivo de estos se usan dos termómetros de estado sólido que dan una señal de $12[mV/^{\circ}C]$. El intervalo de medición está comprendido entre $-15[^{\circ}C]$ y $50[^{\circ}C]$, con una resolución de $0,1[^{\circ}C]$.

Se pide:

1. Definir el conversor a usar. La referencia del mismo es de $10,0[V]$. Especificar con qué error debe operar ésta.
2. Diseñar el circuito de adaptación de señal. Especificar componentes para un error de $0,1[%]$.
3. La humedad es una función no lineal, cuya solución se obtiene por tramos y se tabula. Las entradas de la tabla son: Temperatura del termómetro seco y $(Ts-Th)$, ambas expresadas a $0,1[^{\circ}C]$. En cada cruce de la tabla hay un valor entero de humedad, entre $0[%]$ y $100[%]$. Se pide diseñar la tabla y especificar sobre qué memoria no volátil (tamaño) debe grabarse.
4. Escriba el programa en Assembler que lea un valor de la tabla a partir de una variable Word Tseco (temperatura Term. Seco) y otra Deltat (Ts-Th). Ts-Th para zonas continentales no es mayor que $15[^{\circ}C]$.

NOTA: Resueltos puntos 1. y 2..

Resolución

Datos:

Cant. de termómetros=2[un].

Señal_{termóm.}=12[mV/°C], –Individual-.

t_{mín}=-15[°C] , –Individual-.

t_{máx}=50[°C] , –Individual-.

Resolución=0,1[°C].

V_{ref}=10,0[V].

Cálculos:

$$1. \quad pasos = \frac{\Delta t}{resolución} = \frac{t_{máx} - t_{mín}}{resolución} = \frac{50[°C] - (-15[°C])}{0,1[°C]} \Rightarrow \\ \Rightarrow pasos = 650[pasos]$$
(1)

$$2. \quad n_{A/D} = \frac{\ln(pasos)}{\ln(2)} = \frac{\ln(650)}{\ln(2)} \Rightarrow n_{A/D} = 9,3442[bits] \Rightarrow n_{A/D} = 10[bits]$$
(2)

Por lo tanto se utilizará un **Conversor A/D de 10[bits]**.

$$2. \quad V_{paso} = señal_{termóm.} \cdot resolución = 12 \left[\frac{mV}{°C} \right] \cdot 0,1[°C] \Rightarrow V_{paso} = 1,2[mV]$$
(3)

$$V_{LSB} = \frac{V_{ref}}{2^{n_{A/D}}} = \frac{10,0[V]}{2^{10}} \Rightarrow V_{LSB} = 9,76563[mV]$$
(4)

$$G = \frac{V_{LSB}}{V_{paso}} = \frac{9,76563[mV]}{1,2[mV]} \Rightarrow G = 8,13803[veces]$$
(5)

El circuito a utilizar para la adaptación de la señal al conversor A/D es el siguiente:

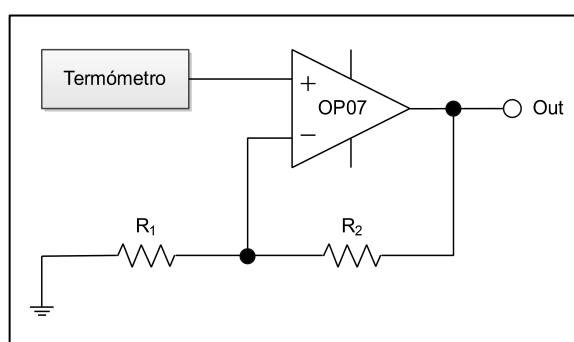


Figura 1: -Final 09.10.08- Circuito utilizado para la adaptación de la señal al conversor A/D

El cual posee la siguiente función de transferencia:

$$\begin{aligned} G = 1 + \frac{R_2}{R_1} &\Rightarrow \frac{R_2}{R_1} = G - 1 = 8,13803[\text{veces}] - 1 \Rightarrow \frac{R_2}{R_1} = 7,13803 \Rightarrow \\ &\Rightarrow R_2 = 7,13803 \cdot R_1 \end{aligned} \quad (6)$$

Por lo tanto, si se selecciona:

- $R_1 = 10[\text{k}\Omega]$.

Indefectiblemente:

- $R_2 = 71,4[\text{k}\Omega]$.

Para corroborar valores, se aplicará la ecuación (6) y se contrastará su resultado con la (5):

$$G = 1 + \frac{R_2}{R_1} = 1 + \frac{71,4[\text{k}\Omega]}{10[\text{k}\Omega]} \Rightarrow G = 8,14[\text{veces}] \quad (7)$$

Por lo tanto se comprueba la validez de los valores de resistencias propuestos.

Para un error de 0,1[%] en los componentes, se tiene:

$$\varepsilon < 0,1[\%] = 0,001 \cdot V_{ref} = 0,001 \cdot 10[V] \Rightarrow \varepsilon < 10[mV] \quad (8)$$

Por otro lado, para despejar “ ε_G ” se parte de:

$$\varepsilon = \varepsilon_{A/D} + \varepsilon_{V_{ref}} + \varepsilon_G \quad (9)$$

A su vez:

$$\varepsilon_{A/D} = \frac{1}{2} \cdot V_{LSB} = \frac{1}{2} \cdot 9,76563[mV] \Rightarrow \varepsilon_{A/D} = 4,88282[mV] \quad (10)$$

Y como:

$$\varepsilon_{V_{ref}} = \varepsilon_G \quad (11)$$

De (9) se despeja “ ε_G ”:

$$\begin{aligned} \varepsilon &= \varepsilon_{A/D} + \varepsilon_{V_{ref}} + \varepsilon_G \Rightarrow \\ &\Rightarrow \varepsilon_{V_{ref}} = \varepsilon_G = \frac{\varepsilon - \varepsilon_{A/D}}{2} = \frac{10[mV] - 4,88282[mV]}{2} \Rightarrow \\ &\Rightarrow \varepsilon_G = 2,55859[mV] \end{aligned} \quad (12)$$

Por lo tanto, la tolerancia para los componentes será

$$\begin{aligned} \text{tolerancia} &= \frac{\mathcal{E}_G}{V_{ref}} = \frac{2,55859[mV]}{10,0[V]} \Rightarrow \text{tolerancia} = 255,859[ppm] \Rightarrow \\ &\Rightarrow \text{tolerancia} = 250[ppm] \end{aligned} \tag{13}$$

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 04 de diciembre de 2008.

1. Diseñe el circuito de sample and hold para una señal:

$s(t) = 6,4 \cdot \text{sen}(1,23E6 \cdot t)[V]$ que se sobremuestrea a 4,5[veces] y se procesa con un A/D de 10[bits] y 10,0[V] de referencia.

Especifique claramente:

- a) Frecuencia de muestreo final.
- b) Tiempo de apertura.
- c) Tiempo de retención.
- d) Diagrama circuital usado, con especificación detallada de los componentes.

2. Diseñe el circuito de adaptación para una señal de cc que fluctúa entre -25,43[mV] y +34,56[mV]. Se usará un conversor A/D de un solo cuadrante con referencia de 10,0[V] y 14[bits] de resolución. Especifique detalladamente los componentes usados.
3. Escriba un programa en Assembler del 8088 para inicializar un periférico 8255 en modo cero, con el puerto “A” configurado como entrada y los puertos “B” y “C” como salidas. El puerto está mapeado a partir de 0300h.
4. Diseñe el circuito de hardware para recibir los datos en un enlace RS-232-C a 9600[bits/s]. Use solamente componentes MSI convencionales. El diseño DEBE incluir registro de desplazamiento y la sincronización del reloj del receptor. Use el servicio de interrupciones del procesador.

NOTA: Resueltos puntos 1. y 2.. En el cálculo del punto 2., el valor obtenido de “tolerancia” de los componentes difiere en 10 veces menos que el cálculo original, por favor, verificar dicho cálculo.

Resolución

Cálculos:

1. Datos:

$$\text{Señal: } s(t) = 6,4 \cdot \sin(1,23E6 \cdot t)[V].$$

Sobremuestreo=4,5[veces].

$n_{A/D}=10[\text{bits}]$.

$V_{ref}=10,0[V]$.

a) En su forma genérica, la “Señal” tiene la siguiente forma:

$$S(t) = S \cdot \sin(\omega \cdot t)[V] \quad (1)$$

Por lo tanto, la frecuencia “f” de la señal es:

$$\omega = 2\pi \cdot f_{señal} \Rightarrow f_{señal} = \frac{\omega}{2\pi} = \frac{1,23 \times 10^6}{2\pi} \Rightarrow f_{señal} = 195,76058[kHz] \quad (2)$$

Por lo tanto, la frecuencia “f” de la señal es:

$$\begin{aligned} f_{muestreo} &= f_{señal} \cdot \text{sobremuestreo} = 195,76058[kHz] \cdot 4,5[\text{veces}] \Rightarrow \\ &\Rightarrow f_{muestreo} = 880,92261[kHz] \end{aligned} \quad (3)$$

$$b) \quad t_m = \frac{1}{f_{muestreo}} = \frac{1}{880,92261[kHz]} \Rightarrow t_m = 1,13517[\mu s] \quad (4)$$

$$t_s = 0,1 \cdot t_m = 0,1 \cdot 1,13517[\mu s] \Rightarrow t_s = 113,517[ns] \quad (5)$$

$$\begin{aligned} c) \quad t_m &= t_s + t_h \Rightarrow t_h = t_m - t_s = 1,13517[\mu s] - 113,517[ns] \Rightarrow \\ &\Rightarrow t_m = 1,02165[\mu s] \end{aligned} \quad (6)$$

d) El esquema circuitual para el control del sample and hold es:

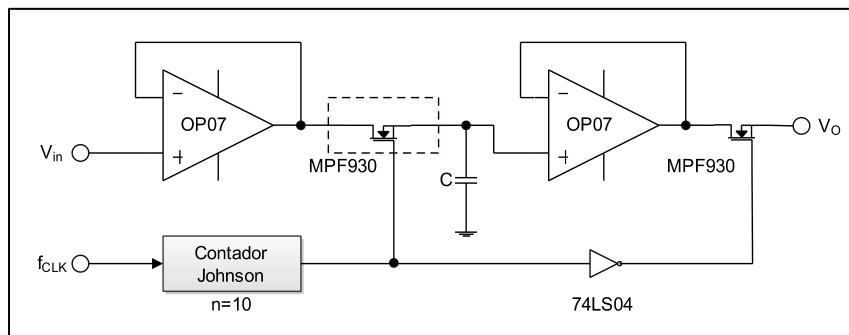


Figura 1: -Final 04.12.08- Circuito utilizado para sample and hold

Por lo tanto, para calcular "C", hay que tener en cuenta que:

- $R_{DS_on}=1[\Omega]$.
- $t_{D_on}=7[ns]$.

Entonces:

$$\begin{aligned} t_s &= 9.R_{DS_on}.C + t_{D_on} \Rightarrow \\ \Rightarrow C &= \frac{t_s - t_{D_on}}{9.R_{DS_on}} = \frac{113,517[ns] - 7[ns]}{9.1[\Omega]} \Rightarrow C = 11,83522[nF] \end{aligned} \quad (7)$$

2. Datos:

$$V_{CC_mín} = -25,43[mV].$$

$$V_{CC_máx} = 34,56[mV].$$

$$V_{ref} = 10,0[V].$$

$$n_{A/D} = 14[\text{bits}].$$

$$\begin{aligned} \Delta V_{CC} &= V_{CC_máx} - V_{CC_mín} = 34,56[mV] - (-25,43[mV]) \Rightarrow \\ \Rightarrow \Delta V_{CC} &= 59,99[mV] \end{aligned} \quad (8)$$

$$G = \frac{V_{ref}}{\Delta V_{CC}} = \frac{10[V]}{59,99[mV]} \Rightarrow G = 166,7[\text{veces}] \quad (9)$$

$$V_{LSB} = \frac{V_{ref}}{2^{n_{A/D}}} = \frac{10,0[V]}{2^{14}} \Rightarrow V_{LSB} = 610,35156[mV] \quad (10)$$

$$\begin{aligned} tolerancia &= \frac{\frac{1}{4}.V_{LSB}}{V_{ref}} = \frac{\frac{1}{4}.610,35156[mV]}{10[V]} \Rightarrow \epsilon_G = 15,25879[ppm] \Rightarrow \\ \Rightarrow tolerancia &= 10[ppm] \end{aligned} \quad (11)$$

Por lo tanto, el circuito a utilizar es el siguiente:

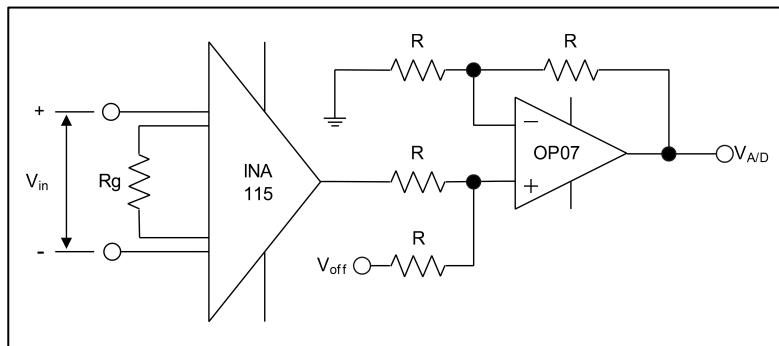


Figura 2: -Final 04.12.08- Circuito utilizado para la adaptación de señal

Por lo tanto, el cálculo de los valores de los componentes es el siguiente:

$$G = 1 + \frac{50[k\Omega]}{R_g} \Rightarrow R_g = \frac{50[k\Omega]}{G-1} = \frac{50[k\Omega]}{166,7[veces]-1} \Rightarrow \\ \Rightarrow R_g = 301,76[\Omega]$$
(12)

Para el cálculo de la tensión de offset (V_{off}), se utiliza un AO en configuración “sumador”, al cual, si se selecciona un valor fijo de “R”, como por ejemplo: $R=10[k\Omega]$, se obtiene la siguiente función de transferencia:

$$V_{off} = G \cdot |V_{CC_mín}| = 166,7[veces] \cdot 25,43[mV] \Rightarrow V_{off} = 4,23904[V]$$
(13)

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 05 de febrero de 2009.

Se dispone de un conversor D/A de 16[bits], mapeado en la dirección 310h del mapa de entrada/salida. Este conversor tiene una fuente de referencia de 5,0[V] nominales. Además se dispone de un 8255 mapeado en la dirección 320h y un controlador de interrupciones ubicado en 328h.

Se pide:

- Simular con estos componentes y los que resultaren necesarios de añadir un conversor A/D de 12[bits].
- Especificar la variación máxima de la fuente de referencia para los 12 y 16[bits] de error, de modo que éste resulte menor que $\frac{1}{4}$ del bit menos significativo.
- Escribir en forma de diagrama de flujo el esquema de funcionamiento detallado del conversor A/D propuesto.
- Especificar, con estos datos, cuál es el valor del incremento mínimo de 12[bits] en [mV].

Escribir la rutina de Assembler para el manejo del registro de aproximaciones sucesivas del A/D a implementar.

Considerar como obligatorio el uso de un 8088 en modo común.

Resolución

Datos:

$$n_{D/A} = 16[\text{bits}]$$

$$\text{Mapeo}_{D/A} = 310h$$

$$V_{ref} = 5,0[V]$$

$$\text{Mapeo}_{8255} = 320h$$

$$\text{Mapeo}_{\text{ctrlador_IRQ}} = 328h$$

$$n_{A/D} = 12[\text{bits}]$$

Cálculos:

La correlación entre los bits del conversor D/A de 16[bits] y el conversor A/D de 12[bits] es la siguiente:

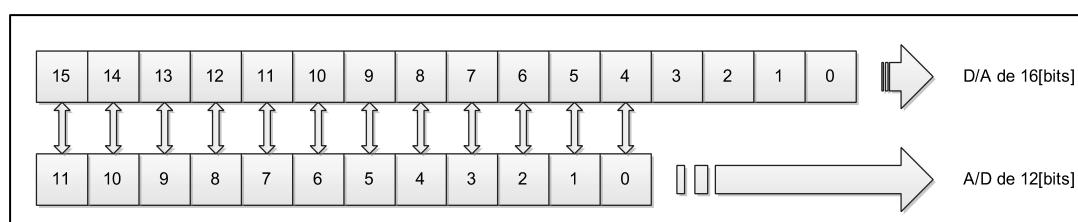


Figura 1: -Final 05.02.09- Correlación entre el conversor D/A de 16[bits] y el conversor A/D de 12[bits]

A continuación se muestra el circuito utilizado para la combinación de ambos conversores trabajando en conjunto:

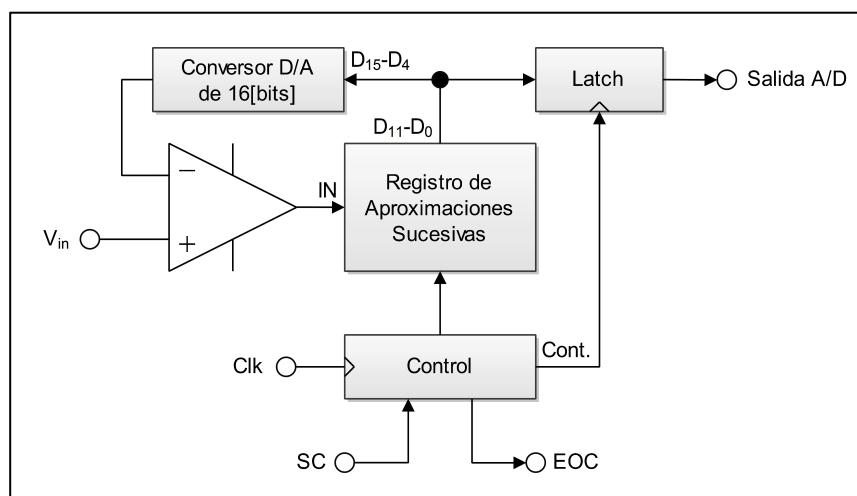


Figura 2: -Final 05.02.09- Combinación de ambos conversores

Los errores máximos para la fuente de referencia, para los casos de 12[bits] y 16[bits], son los siguientes:

$$\epsilon_{V_{ref}} < \frac{1}{4} \cdot V_{LSB} \quad (1)$$

Y como:

$$V_{LSB} = \frac{V_{ref}}{2^n} \quad (2)$$

Aplicando (1) y (2) para cada caso particular, queda:

$$\epsilon_{V_{ref_12[bits]}} = \frac{V_{ref}}{4^2} = \frac{V_{ref}}{4 \cdot 2^n} = \frac{5,0[V]}{4 \cdot 2^{12}} \Rightarrow \epsilon_{V_{ref_12[bits]}} = 305,17578[\mu V] \quad (3)$$

$$\epsilon_{V_{ref_16[bits]}} = \frac{V_{ref}}{4^2} = \frac{V_{ref}}{4 \cdot 2^n} = \frac{5,0[V]}{4 \cdot 2^{16}} \Rightarrow \epsilon_{V_{ref_16[bits]}} = 19,07349[\mu V] \quad (4)$$

El diagrama de flujo del conversor propuesto es el siguiente:

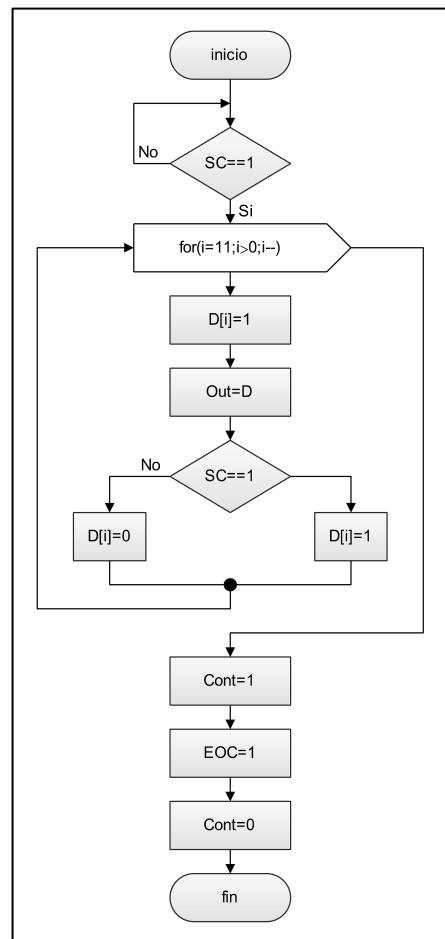


Figura 3: -Final 05.02.09- Diagrama de flujo del conversor propuesto

Luego, se aplica (2) para calcular el “ V_{LSB} ” con los datos anteriores para 12[bits], de la siguiente manera:

$$V_{LSB} = \frac{V_{ref}}{2^n} = \frac{5,0[V]}{2^{12}} \Rightarrow V_{LSB} = 1,2207[mV] \quad (5)$$

Finalmente, la rutina para el manejo del registro de aproximaciones sucesivas, es el siguiente:

reset:

```
mov r0,#2048
mov r1,#11
mov r2,#IN
mov r3,#OUT
mov r6,#0
```

sar:

```
str r0,[r3]
ldr r4,[r2]
mov r5,r4,lsl r1
orr r6,r6,r5
mov r0,r6
subs r1,r1,#1
bne sar
```

loop: b loop

*/Constantes

```
IN:      .word
OUT:     .word
```

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 12 de febrero de 2009.

Se debe medir potencia en alterna en 3 fases ($3 \times 380[V]$). Para ello es necesario medir tensión, corriente y la fase entre ellas.

Se necesita una precisión del 1[%] (clase 1). Se pregunta:

1. ¿Qué resolución han de tener el (los) conversores a usar? Elija en este punto cuántos conversores usará. Justifique. Debe tener en cuenta el multiplexor, si usa sólo uno.
2. Diseñe el circuito de medición de corriente. Tenga en cuenta la precisión requerida. Especifique detalladamente todos los componentes.
3. Diseñe el circuito de adaptación de señal que se usará para medir tensión. Tenga en cuenta la precisión requerida. Especifique detalladamente todos los componentes.
4. Proponga un método para calcular la potencia instantánea que garantice el error deseado. Describalo en forma de diagrama de flujo.
5. Proponga un método para medir fase.

Resolución

Datos:

Medición de: V , I , ϕ .

Fases: $3 \times 380[V_{CA}]$.

Precisión = $1[\%]$ = $0,01$.

A continuación se muestra un pequeño esquema de cómo es la medición:

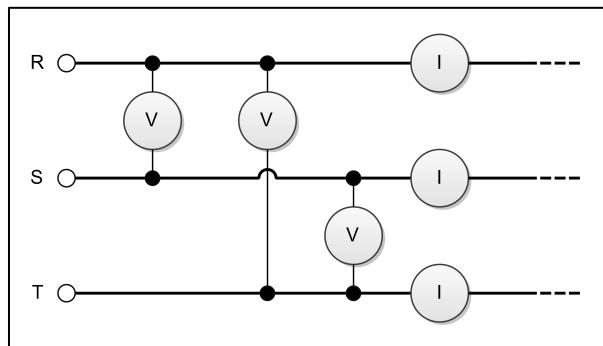


Figura 1: -Final 12.02.09- Esquema de la medición a realizar

Cálculos:

1. Por lo tanto, si se desea una precisión del $1[\%]$, en total se tendrá una cantidad de:

$$cuentas = 100[cuentas] \quad (1)$$

Luego:

$$\begin{aligned} n_{A/D} &= \frac{\ln(cuentas)}{\ln(2)} = \frac{\ln(100[cuentas])}{\ln(2)} \Rightarrow n_{A/D} = 6,64386[bits] \Rightarrow \\ &\Rightarrow n_{A/D} = 8[bits] \end{aligned} \quad (2)$$

Por lo tanto, se concluye de utilizar un **Conversor A/D de 8[bits]**, junto a un **Multiplexor de 16 canales**, de los cuales 3 canales serán para la tensión, 3 para la corriente y 3 para la fase.

2. Se decide utilizar una resistencia “shunt”. Suponiendo $I_{RMS}|_{máx} = 10[A]$, y con $R = 0,1[\Omega]$, se obtiene:

$$V_{RMS}|_{máx} = I_{RMS}|_{máx} \cdot R = 10[A] \cdot 0,1[\Omega] \Rightarrow V_{RMS}|_{máx} = 1[V] \quad (3)$$

Por lo tanto, se puede plantear el siguiente esquema para la medición de corriente:

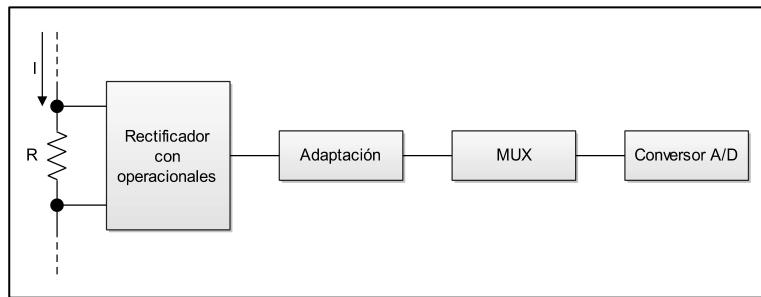


Figura 2: -Final 12.02.09- Esquema de la medición de corriente

Por lo tanto, el bloque de “Rectificador con operacionales” consiste en un “Convertidor de CA a CC”, el cual también es un “Rectificador de precisión”, y su circuito es el siguiente:

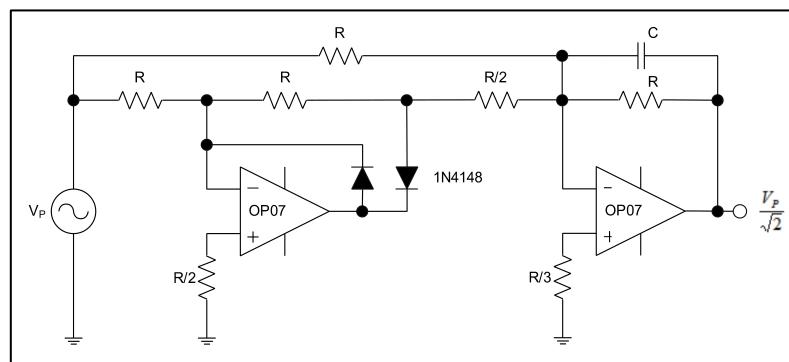


Figura 3: -Final 12.02.09- Circuito convertidor CA a CC (Rectificador de precisión)

Ahora, el bloque de “Adaptación” se resuelve de la siguiente forma, tomando convenientemente el valor de la tensión de referencia $V_{ref}=5[V]$:

$$V_{LSB} = \frac{V_{ref}}{2^{n_{A/D}}} = \frac{5[V]}{2^8} \Rightarrow V_{LSB} = 19,53125[mV] \quad (4)$$

$$V_{paso} = \frac{V_{RMS}|_{máx}}{cuentas} = \frac{1[V]}{100} \Rightarrow V_{paso} = 10[mV] \quad (5)$$

$$G = \frac{V_{LSB}}{V_{paso}} = \frac{19,53125[mV]}{10[mV]} \Rightarrow G = 1,95313[veces] \quad (6)$$

Por lo tanto se utilizará un AO de la siguiente forma:

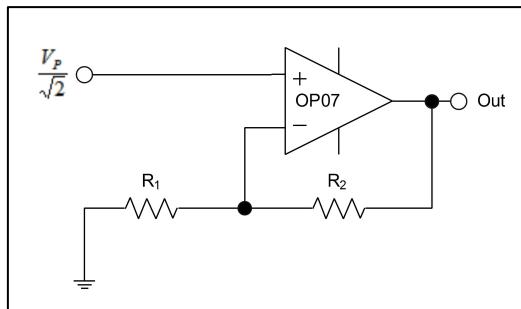


Figura 4: -Final 12.02.09- Circuito utilizado para la adaptación de la señal

Dicho circuito responde a la siguiente función de transferencia:

$$\begin{aligned} G &= 1 + \frac{R_2}{R_1} = 1,95313[\text{veces}] \Rightarrow 1,95313[\text{veces}] - 1 = \frac{R_2}{R_1} \Rightarrow \\ &\Rightarrow R_2 = 0,95313[\text{veces}].R_1 \end{aligned} \quad (7)$$

Donde el valor de “G” viene de la ecuación (6). Por lo tanto, si se selecciona:

$$R_2 = 10[k\Omega] \quad (8)$$

De (7) se despeja el valor de “R₁”:

$$R_1 = \frac{R_2}{0,95313[\text{veces}]} = \frac{10[k\Omega]}{0,95313[\text{veces}]} \Rightarrow R_1 = 10,49[k\Omega] \quad (9)$$

3. Se utilizará el mismo circuito que el de la *Figura 3*, pero antecedido por un divisor resistivo como el de la figura siguiente (tomando solo dos fases: “R” y “S”, para los otros dos pares de juegos entre fases, el sistema utilizado es idéntico):

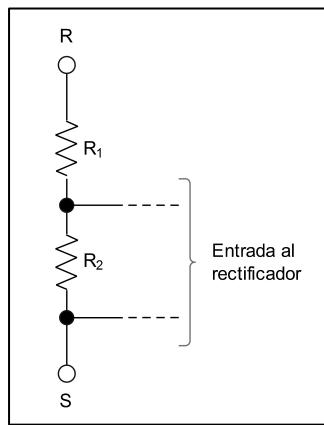


Figura 5: -Final 12.02.09- Circuito utilizado para la adaptación de la señal de tensión, antes del rectificador

Teniendo en cuenta lo siguiente:

$$\begin{aligned} V_{R_S} &= 380[V] = V_{RMS} \Rightarrow \\ &\Rightarrow V_p = \sqrt{2}.V_{RMS} = \sqrt{2}.380[V] \Rightarrow V_p = 537,4[V_p] \end{aligned} \quad (10)$$

Ahora sí, del divisor se tiene que:

$$\frac{R_1}{R_2} = 250 \Rightarrow R_1 = 250.R_2 \quad (11)$$

Por lo tanto, si se selecciona:

$$R_2 = 100[\Omega] \quad (12)$$

Por medio de (11) se obtiene el valor de “R₁”:

$$R_1 = 250.R_2 = 250.100[\Omega] \Rightarrow R_1 = 25[k\Omega] \quad (13)$$

Ahora, en la adaptación de la señal, el valor de “V_{LSB}” es el mismo que el obtenido en la ecuación (4), por lo tanto trayendo dicha ecuación:

$$V_{LSB} = \frac{V_{ref}}{2^{n_{A/D}}} = \frac{5[V]}{2^8} \Rightarrow V_{LSB} = 19,53125[mV] \quad (14)$$

Pero “V_{paso}” es diferente, tal como:

$$V_{paso} = \frac{V_{R-S}}{\text{cuentas.} \frac{R_1}{R_2}} = \frac{380[V]}{100[\text{cuentas}].250} \Rightarrow V_{paso} = 15,2[mV] \quad (15)$$

Finalmente:

$$G = \frac{V_{LSB}}{V_{paso}} = \frac{19,53125[mV]}{15,2[mV]} \Rightarrow G = 1,285[\text{veces}] \quad (16)$$

El circuito que se utilizará es el mismo de la *Figura 4*, y por lo tanto son válidas las ecuaciones (7), (8) y (9), que se traen nuevamente a continuación:

$$\begin{aligned} G &= 1 + \frac{R_2}{R_1} = 1,285[\text{veces}] \Rightarrow 1,285[\text{veces}] - 1 = \frac{R_2}{R_1} \Rightarrow \\ &\Rightarrow R_2 = 0,285[\text{veces}].R_1 \end{aligned} \quad (17)$$

Donde el valor de “G” viene de la ecuación (16). Por lo tanto, si se selecciona:

$$R_2 = 10[k\Omega] \quad (18)$$

De (17) se despeja el valor de “R₁”:

$$R_1 = \frac{R_2}{0,285[\text{veces}]} = \frac{10[k\Omega]}{0,285[\text{veces}]} \Rightarrow R_1 = 35,1[k\Omega] \quad (19)$$

4. Para este punto, hay que partir de la siguiente fórmula:

$$P_{inst} = V \cdot I \cdot \cos(\varphi) \quad (20)$$

Por lo tanto, para no introducir mayores errores, el tiempo de multiplexado debe ser lo suficientemente corto. Para lo anterior se verificará de la siguiente manera:

$$f_{línea} = 50[\text{Hz}] \quad (21)$$

Por lo tanto:

$$f_m = 100 \cdot f_{línea} = 100 \cdot 50[\text{Hz}] \Rightarrow f_m = 5[\text{kHz}] \quad (22)$$

Ya que se está trabajando con valores muy bajos de frecuencia, se puede hacer un sobremuestreo elevado, como se hizo en (22) en 100[veces]. Por lo tanto, el tiempo de muestreo “ t_m ” será:

$$t_m = \frac{1}{f_m} = \frac{1}{5[\text{kHz}]} \Rightarrow t_m = 200[\mu\text{s}] \quad (23)$$

Por lo tanto, el multiplexor que se utilizará, posee la siguiente configuración:

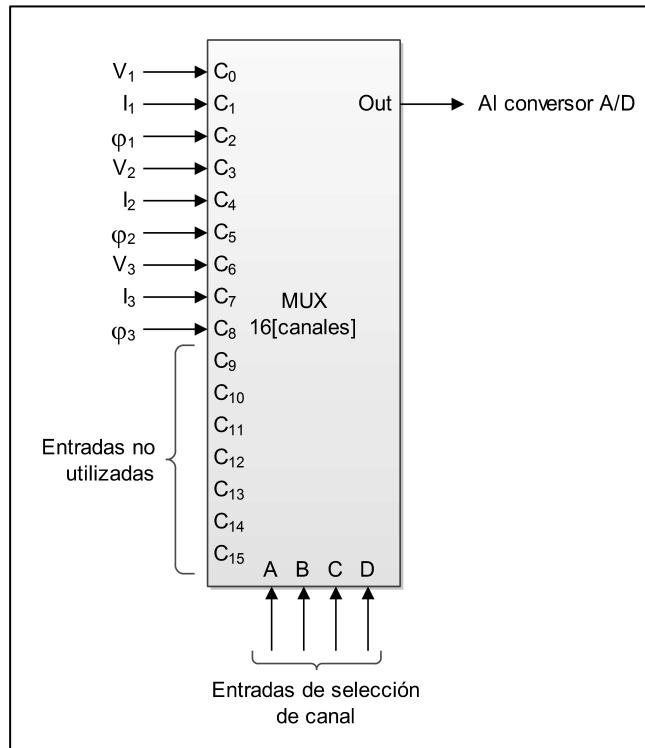


Figura 6: -Final 12.02.09- Multiplexor utilizado

Todo el conjunto responde al siguiente diagrama de flujo:

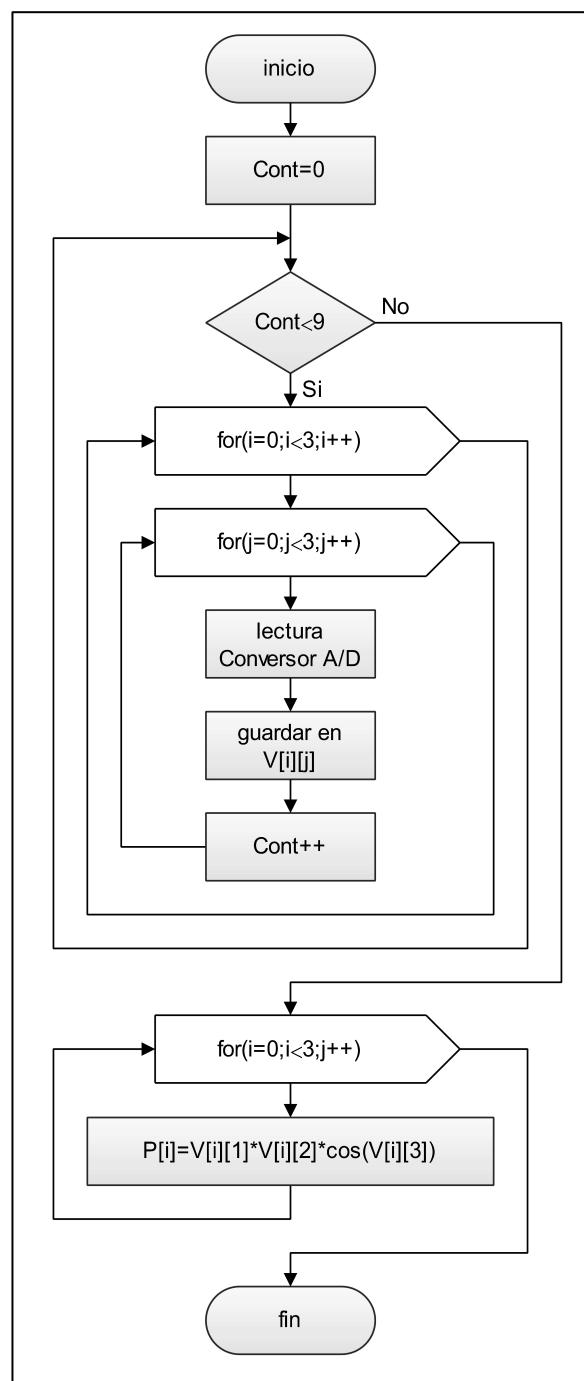


Figura 7: -Final 12.02.09- Diagrama de flujo de la implementación para medir las tres variables solicitadas

5. A continuación se propone un método para llevar a cabo la medición de la fase:

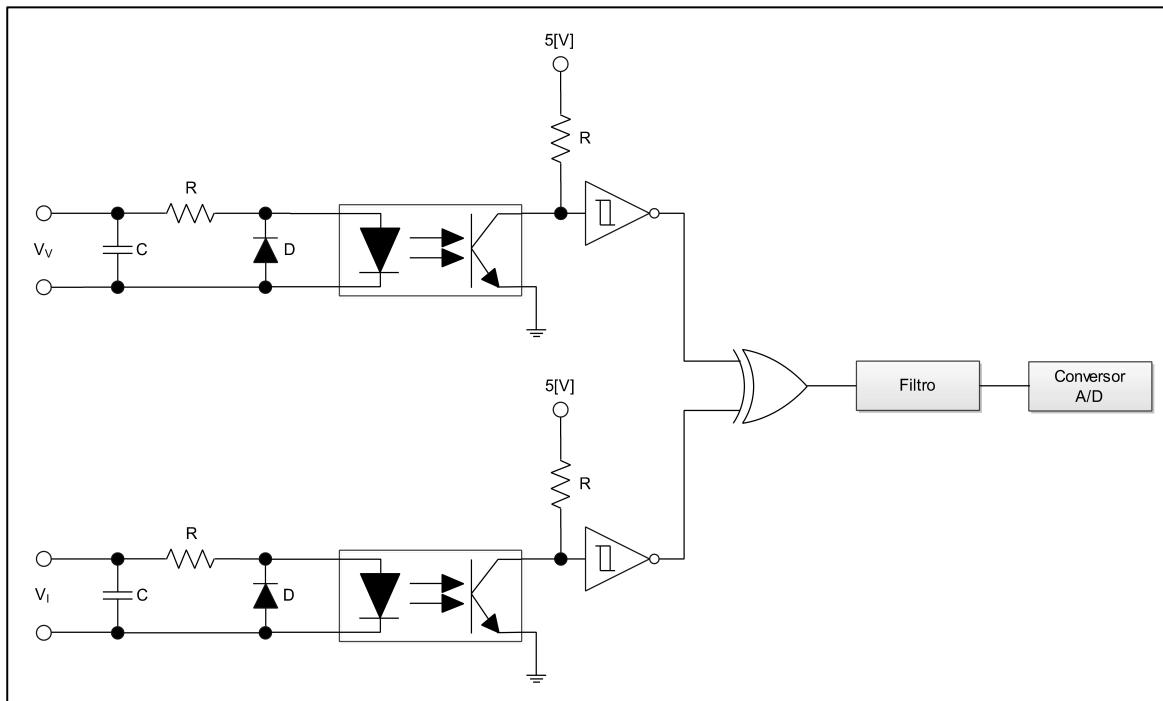


Figura 8: -Final 12.02.09- Circuito utilizado para la medición de fase

La base del presente circuito consiste en que la salida de la compuerta será una señal, cuyo ciclo útil es proporcional al ángulo “ ϕ ”, siendo los valores extremos, luego del filtro:

- $\varphi_{máx} = 180^\circ$, para $V_{salida_filtro} = 5[V]$.
 - $\varphi_{mín} = 0^\circ$, para $V_{salida_filtro} = 0[V]$.

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 26 de febrero de 2009.

Sea una señal que admite una expresión: $s(t) = 2,2 \cdot \text{sen}(4,18E5 \cdot t + 0,986)[V]$.

1. ¿Cuál es la frecuencia de muestreo según Nyquist? ¿Cuál es la recomendable? ¿Cuál es el tiempo de conversión mínimo para este caso?
2. En la señal de 1. se requiere discriminar la amplitud a 4,4[mV]. Especificar el número de bits del conversor a usar.
3. Para usar un conversor de un solo cuadrante y referencia de 10,0[V], diseñar el circuito de adaptación de señal.
4. Se requiere un error total igual a $\frac{1}{4}$ lsb. Especifique los componentes de error admisibles en la etapa anterior.
5. Diseñar el circuito de sample and hold necesario. Calcule el tiempo de apertura y el de retención. Especificar todos sus componentes.
6. Se implementa el equipo a partir de un NEC V20 a 20[MHz]. ¿Puede operarse por interrupción? Justificar respuesta.
7. Diseñe con componentes discretos (transmisores y compuertas) un multiplexor analógico de dos canales. El canal activo se comanda con la variable “canal” {0,1} y la salida con la variable “sal” {0,1}. Existirá salida cuando sal=0.

Para aprobar el examen se requieren cuatro puntos totalmente correctos, los que deben incluir tres del conjunto de {3, 4, 5, 6, 7}.

Resolución

Cálculos:

Datos:

$$\text{Señal: } s(t) = 2,2 \cdot \sin(4,18E5 \cdot t + 0,986)[V].$$

- En su forma genérica, la “Señal” tiene la siguiente forma:

$$S(t) = S \cdot \sin(\omega \cdot t + \phi)[V] \quad (1)$$

Por lo tanto, la frecuencia “f” de la señal es:

$$\omega = 2\pi \cdot f_{señal} \Rightarrow f_{señal} = \frac{\omega}{2\pi} = \frac{4,18 \times 10^5}{2\pi} \Rightarrow f_{señal} = 66,52677[kHz] \quad (2)$$

Por lo tanto, la frecuencia “f_{m_Nyquist}” de la señal es:

$$\begin{aligned} f_{m_Nyquist} &= f_{señal} \cdot 2[\text{veces}] = 66,52677[kHz] \cdot 2[\text{veces}] \Rightarrow \\ &\Rightarrow f_{m_Nyquist} = 133,05353[kHz] \end{aligned} \quad (3)$$

Ahora, la frecuencia mínima recomendable de muestreo, o sea la “f_m”, en la práctica es 5[veces] la frecuencia de la señal “f_{señal}”, por lo tanto:

$$\begin{aligned} f_m &= f_{señal} \cdot 5[\text{veces}] = 66,52677[kHz] \cdot 5[\text{veces}] \Rightarrow \\ &\Rightarrow f_m = 332,63385[kHz] \end{aligned} \quad (4)$$

Y el tiempo de conversión mínimo de “f_m” es:

$$t_m = \frac{1}{f_m} = \frac{1}{332,63385[kHz]} \Rightarrow t_m = 3,00631[\mu s] \quad (5)$$

- El número de bits del conversor A/D “n_{A/D}” para discriminar la amplitud de 4,4[mV], o sea “V_{paso}”, se define como:

$$cuentas = \frac{S_{p-p}}{V_{paso}} = \frac{4,4[V]}{4,4[mV]} \Rightarrow cuentas = 1000[cuentas] \quad (6)$$

$$n_{A/D} = \frac{\ln(cuentas)}{\ln(2)} = \frac{\ln(1000[cuentas])}{\ln(2)} \Rightarrow n_{A/D} = 9,96578[bits] \quad (7)$$

Por lo tanto se utilizará un **Conversor A/D de 10[bits]**.

3. Primero se calculará la ganancia “G” para adaptar la señal. Recordar que la tensión de referencia del conversor A/D es $V_{ref}=10[V]$, entonces:

$$G = \frac{V_{LSB}}{V_{paso}} = \frac{\frac{V^{ref}}{2^{n_{A/D}}}}{\frac{10[V]}{4,4[mV]}} = \frac{10[V]}{2^{10}} \Rightarrow G = 2,21946[\text{veces}] \quad (8)$$

Por lo tanto se utilizarán 2 AOs, uno para adaptar la ganancia, y el otro para el nivel de offset “ V_{off} ”, de acuerdo al siguiente circuito:

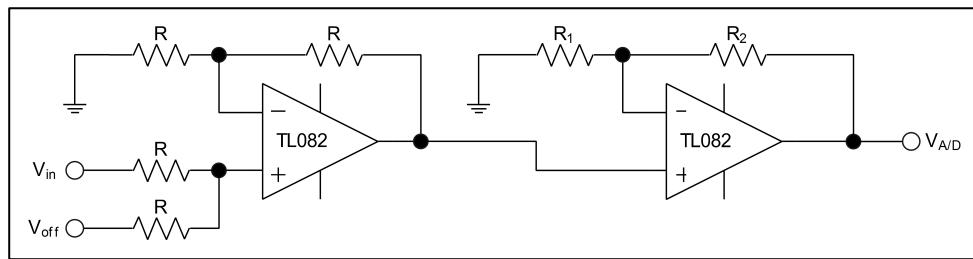


Figura 1: -Final 26.02.09- Circuito utilizado para la adaptación de señal

Donde se seleccionan los siguientes valores:

$$R = 10[k\Omega] \quad (9)$$

Luego, el AO utilizado para la ganancia responde a la siguiente función de transferencia:

$$\begin{aligned} G &= 1 + \frac{R_2}{R_1} \Rightarrow R_2 = (G - 1).R_1 = (2,21946[\text{veces}] - 1).R_1 \Rightarrow \\ &\Rightarrow R_2 = 1,21946[\text{veces}].R_1 \end{aligned} \quad (10)$$

Por lo tanto, si se selecciona:

$$R_1 = 10[k\Omega] \quad (11)$$

Aplicando la ecuación (10), “ R_2 ” queda:

$$R_2 = 1,21946[\text{veces}].R_1 = 1,21946[\text{veces}].10[k\Omega] \Rightarrow R_2 = 12,19[k\Omega] \quad (12)$$

4. Se plantea que el error total “ ε ” debe ser igual a $\frac{1}{4}$ de lsb, por lo tanto:

$$\varepsilon = \frac{1}{4}.V_{LSB} = \frac{1}{4} \cdot \frac{V^{ref}}{2^{n_{A/D}}} = \frac{1}{4} \cdot \frac{10[V]}{2^{10}} \Rightarrow \varepsilon = 2,44141[mV] \quad (13)$$

Pero recordando que el error total es la suma de los errores de ganancia y de la tensión de referencia, en partes iguales, se deduce lo siguiente:

$$\begin{aligned} \varepsilon &= \varepsilon_G + \varepsilon_{V_{ref}} \Rightarrow \varepsilon_{V_{ref}} = \varepsilon_G = \frac{\varepsilon}{2} = \frac{2,44141[mV]}{2} \Rightarrow \\ &\Rightarrow \varepsilon_G = 1,22071[mV] \end{aligned} \quad (14)$$

Por lo tanto, la tolerancia de los componentes se calcula de la siguiente manera:

$$\text{tolerancia} = \frac{\varepsilon_G}{V_{ref}} = \frac{1,22071[mV]}{10[V]} \Rightarrow \text{tolerancia} = 122,07050[ppm] \quad (15)$$

Por lo tanto se toma una **tolerancia=100[ppm]**.

5. A continuación se muestra el circuito utilizado para el sample and hold:

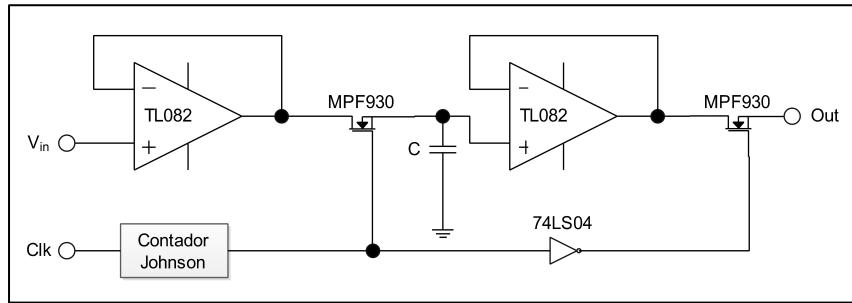


Figura 2: -Final 26.02.09- Circuito utilizado para el sample and hold

El tiempo de muestreo ya se calculó en la ecuación (5), con el siguiente valor:

$$t_m = \frac{1}{f_m} = \frac{1}{332,63385[kHz]} \Rightarrow t_m = 3,00631[\mu s] \quad (16)$$

Ahora, en la práctica se toma que el tiempo de apertura (sample) "t_s" sea 0,1[veces] el de muestra, por lo tanto:

$$t_s = 0,1 \cdot t_m = 0,1 \cdot 3,00631[\mu s] \Rightarrow t_s = 300,63086[ns] \quad (17)$$

Por lo tanto, utilizando la (17) en la siguiente ecuación, se encuentra el tiempo de retención (hold) "t_h", de la siguiente manera:

$$\begin{aligned} t_m &= t_s + t_h \Rightarrow t_h = t_m - t_s = 3,00631[\mu s] - 300,63086[ns] \Rightarrow \\ &\Rightarrow t_h = 2,70568[\mu s] \end{aligned} \quad (18)$$

Por lo tanto, lo único que falta por averiguar es el valor del capacitor "C", el cual se halla teniendo en cuenta los siguientes valores de los MPF930:

- R_{DS_on}=1[Ω].
- t_{D_on}=7[ns].

Y por medio de la siguiente ecuación, se calcula el valor de "C":

$$\begin{aligned} t_s &= 9 \cdot R_{DS_on} \cdot C + t_{D_on} \Rightarrow C = \frac{t_s - t_{D_on}}{9 \cdot R_{DS_on}} = \frac{300,63086[ns] - 7[ns]}{9 \cdot 1[\Omega]} \Rightarrow \\ &\Rightarrow C = 32,62565[nF] \end{aligned} \quad (19)$$

6. Sabiendo que para este caso que $t_m=3,00631[\mu s]$, y que el NEC V20 es compatible pin a pin con el μP 8086, se tiene:

- 1 instrucción \Rightarrow 2 ciclos de máquina
- 1 ciclo de máquina \Rightarrow 2 ciclos de clock

Por lo tanto:

$$\begin{aligned} 1 \text{ instrucción} &= 4 \cdot \text{ciclos de clock} = 4 \cdot \frac{1}{f_{NECV20}} = 4 \cdot \frac{1}{20[MHz]} \Rightarrow \\ &\Rightarrow 1 \text{ instrucción} = 200[ns] \end{aligned} \quad (20)$$

O sea que se cumple que:

$$\begin{aligned} t_{NECV20} &<< t_m \\ 200[ns] &<< 3,00631[\mu s] \end{aligned} \quad (21)$$

Por lo tanto, a la frecuencia que trabaja el procesador, las interrupciones no causan problemas a la hora de muestrear.

7. A continuación se muestra el circuito del multiplexor solicitado:

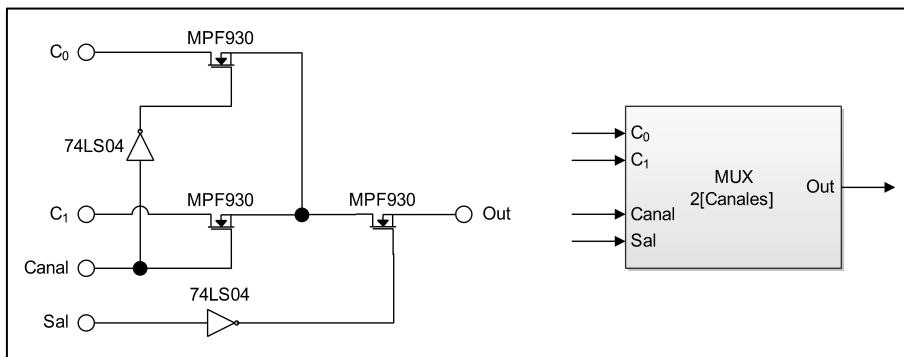


Figura 3: -Final 26.02.09- Circuito del multiplexor de 2[Canales] solicitado

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 03 de diciembre de 2009.

Sea una señal que admite una expresión: $s(t) = 1,18 \cdot \text{sen}(1,2E3 \cdot t)[V]$.

1. ¿Cuál es la frecuencia de muestreo según Nyquist? ¿Cuál es la recomendable? ¿Cuál es el tiempo de conversión mínimo para este caso?
2. En la señal de 1. se requiere discriminar la amplitud a 1,0[mV]. Especificar el número de bits del conversor a usar.
3. Para usar un conversor de un solo cuadrante y referencia de 10,0[V], diseñar el circuito de adaptación de señal.
4. Se requiere un error total igual a 1 lsb. Especifique los componentes de error admisibles en la etapa anterior.
5. Se implementa el equipo a partir de un procesador de ciclo de máquina (ciclo de base de operación. La instrucción más simple tarda dos ciclos de máquina) igual a dos ciclos de clock con un reloj de 20[MHz]. ¿Puede operarse por interrupción? Justificar respuesta.
6. Diseñe con componentes discretos (transistores y compuertas) un multiplexor analógico de dos canales. El canal activo se comanda con la variable “canal” {0,1} y la salida con la variable “sal” {0,1}. Existirá salida cuando sal=0.
7. Describa el funcionamiento de las instrucciones pop y push, cuál es su función, qué registros modifica y en qué posición de memoria leen o escriben un dato, además si antes de llamar a una subrutina guardó un valor en el stack, de qué manera lo puedo leer dentro de la subrutina sin alterar el correcto funcionamiento del programa.
8. Una memoria de 2[kb] se encuentra mapeada a partir de la dirección 0x4E800, para el direccionamiento externo se utilizaron solo las líneas A19, A18 y A15, generando por lo tanto imágenes de la misma en el mapa. Si el microprocesador realiza lecturas en las siguientes direcciones absolutas, en qué caso se direcciona la memoria y cuál es la posición leída dentro de ella.
0x7FFF
0x46800
0x6A9FF

Para aprobar el examen se requieren cinco puntos totalmente correctos, los que deben incluir tres del conjunto de {3, 4, 5, 6, 7}.

Resolución

Cálculos:

Datos:

$$\text{Señal: } s(t) = 1,18 \cdot \sin(1,2E3 \cdot t)[V].$$

- En su forma genérica, la “Señal” tiene la siguiente forma:

$$S(t) = S \cdot \sin(\omega \cdot t)[V] \quad (1)$$

Por lo tanto, la frecuencia “f” de la señal es:

$$\omega = 2\pi \cdot f_{señal} \Rightarrow f_{señal} = \frac{\omega}{2\pi} = \frac{1,2 \times 10^3}{2\pi} \Rightarrow f_{señal} = 190,98593[Hz] \quad (2)$$

Por lo tanto, la frecuencia “f_{m_Nyquist}” de la señal es:

$$\begin{aligned} f_{m_Nyquist} &= f_{señal} \cdot 2[\text{veces}] = 190,98593[Hz] \cdot 2[\text{veces}] \Rightarrow \\ &\Rightarrow f_{m_Nyquist} = 381,97186[Hz] \end{aligned} \quad (3)$$

Ahora, la frecuencia mínima recomendable de muestreo, o sea la “f_m”, en la práctica es 5[veces] la frecuencia de la señal “f_{señal}”, por lo tanto:

$$\begin{aligned} f_m &= f_{señal} \cdot 5[\text{veces}] = 190,98593[Hz] \cdot 5[\text{veces}] \Rightarrow \\ &\Rightarrow f_m = 954,92965[Hz] \end{aligned} \quad (4)$$

Y el tiempo de conversión mínimo de “f_m” es:

$$t_m = \frac{1}{f_m} = \frac{1}{954,92965[Hz]} \Rightarrow t_m = 1,0472[ms] \quad (5)$$

- Si se desea discriminar pasos de 1,0[mV], se tendrá:

$$cuentas = \frac{S_{p-p}}{V_{paso}} = \frac{2,36[V]}{1,0[mV]} \Rightarrow cuentas = 2360[cuentas] \quad (6)$$

Por lo tanto, el número de bits del conversor A/D “n_{A/D}” será:

$$n_{A/D} = \frac{\ln(cuentas)}{\ln(2)} = \frac{\ln(2360[cuentas])}{\ln(2)} \Rightarrow n_{A/D} = 11,20457[bits] \quad (7)$$

Por lo tanto se utilizará un **Conversor A/D de 12[bits]**.

3. El circuito de adaptación de señal se compone de dos partes, una para llevar el nivel de la señal de entrada a niveles adecuados por medio de una tensión de offset “ V_{off} ” que se llevará a cabo por medio de un AO; y la otra parte es la ganancia necesaria para ingresar al conversor A/D también con otro AO. En primer lugar se harán algunos cálculos y luego se implementará el circuito:

$$G = \frac{V_{ref}}{V_{LSB}} = \frac{2^{n_{A/D}}}{V_{paso}} = \frac{10,0[V]}{1,0[mV]} \Rightarrow G = 2,44144[\text{veces}] \quad (8)$$

$$V_{off} = |\hat{S}| \Rightarrow V_{off} = 1,18[V] \quad (9)$$

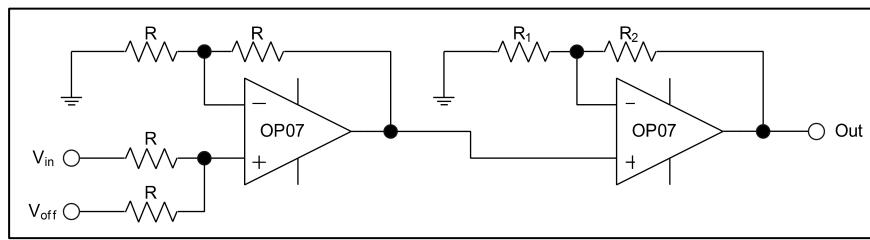


Figura 1: -Final 03.12.09- Circuito utilizado para la adaptación de señal

Donde se selecciona:

$$R = 10[k\Omega] \quad (10)$$

Luego, para el 2^{do} AO, para la ganancia, responde a la siguiente función de transferencia:

$$\begin{aligned} G &= 1 + \frac{R_2}{R_1} \Rightarrow R_2 = (G - 1) \cdot R_1 = (2,44144[\text{veces}] - 1) \cdot R_1 \Rightarrow \\ &\Rightarrow R_2 = 1,44144[\text{veces}] \cdot R_1 \end{aligned} \quad (11)$$

De lo cual, si se selecciona:

$$R_1 = 10[k\Omega] \quad (12)$$

De (11) se obtiene “ R_2 ” de la siguiente forma:

$$R_2 = 1,44144[\text{veces}] \cdot R_1 = 1,44144[\text{veces}] \cdot 10[k\Omega] \Rightarrow R_2 = 14,4141[k\Omega] \quad (13)$$

4. Si se desea un error total “ ε ” de 1 lsb, entonces se tiene:

$$\varepsilon = 1 \cdot V_{LSB} = 1 \cdot \frac{V_{ref}}{2^{n_{A/D}}} = 1 \cdot \frac{10,0[V]}{2^{12}} \Rightarrow \varepsilon = 2,44141[mV] \quad (14)$$

Pero “ ε ” está formado por tres partes, por lo tanto:

$$\begin{aligned}\mathcal{E} &= \mathcal{E}_{V_{ref}} + \mathcal{E}_G + \mathcal{E}_{A/D} \Rightarrow \mathcal{E}_G + \mathcal{E}_{V_{ref}} = \mathcal{E} - \mathcal{E}_{A/D} = \mathcal{E} - \frac{1}{2} \cdot V_{LSB} = \\ &= 2,44141[mV] - \frac{1}{2} \cdot 2,44141[mV] \Rightarrow \mathcal{E}_G + \mathcal{E}_{V_{ref}} = 1,22071[mV]\end{aligned}\tag{15}$$

Luego, para sacar el valor de “ $\mathcal{E}_{V_{ref}}$ ”:

$$\mathcal{E}_{V_{ref}} = \mathcal{E}_G = \frac{\mathcal{E} - \mathcal{E}_{A/D}}{2} = \frac{1,22071[mV]}{2} \Rightarrow \mathcal{E}_{V_{ref}} = 610,3525[\mu V]\tag{16}$$

Finalmente, la tolerancia de los componentes se calcula como:

$$\text{tolerancia} = \frac{\mathcal{E}_{V_{ref}}}{V_{ref}} = \frac{610,3525[\mu V]}{10,0[V]} \Rightarrow \text{tolerancia} = 61,03525[ppm]\tag{17}$$

Por lo tanto, se toma una **tolerancia=50[ppm]**.

5. Aquí se debe tener en cuenta lo siguiente:

- 1 instrucción \implies 2 ciclos de máquina
- 1 ciclo de máquina \Rightarrow 2 ciclos de clock

Por lo tanto, se tiene lo siguiente:

$$\text{instrucción} = 2 \cdot \text{ciclos_de_máquina} = 2 \cdot (2 \cdot \text{ciclos_de_clock})\tag{18}$$

Y cada ciclo de clock “ T_{Clk} ” es:

$$T_{Clk} = \frac{1}{f_{Clk}} = \frac{1}{20[MHz]} \Rightarrow T_{Clk} = 50[ns]\tag{19}$$

Reemplazando (18) en (17), y resolviendo:

$$\begin{aligned}\text{instrucción} &= 2 \cdot (2 \cdot \text{ciclos_de_clock}) = 4 \cdot 50[ns] \Rightarrow \\ &\Rightarrow \text{instrucción} = 200[ns]\end{aligned}\tag{20}$$

Por lo tanto, para corroborar si se puede utilizar, se debe cumplir que:

$$\begin{aligned}\text{instrucción} &<< t_m \\ 200[ns] &<< 1,0472[ms]\end{aligned}\tag{21}$$

Con lo que se concluye que el procesador **si** puede trabajar con interrupciones, sin alterar el proceso de muestreo.

6. A continuación se muestra el circuito del multiplexor solicitado:

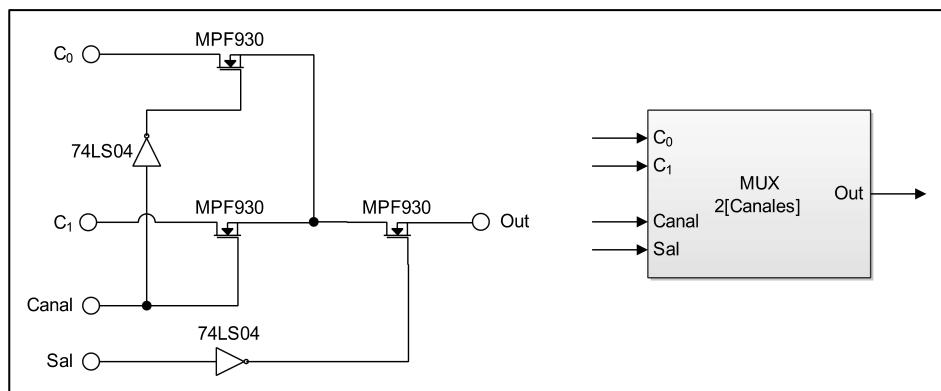


Figura 2: -Final 03.12.09- Circuito utilizado para el multiplexor solicitado

7. **Pop:** Carga múltiples registros desde memoria.

Push: Guarda múltiples registros en la memoria.

Ambas instrucciones son casos especiales de las instrucciones “stm” y “ldm”.

Al manejar el “Stack”, se trata de formar una pila donde cada vez que se guardan los datos en la posición de memoria, ésta es actualizada según donde apunta el “Stack”.

Aquí se divide en dos estados:

- Full Stack: El puntero indica la dirección del último dato.
- Empty Stack: El puntero indica la primera dirección libre.

Según la forma en que se guardan los datos, se puede clasificar en:

- Descending Stack: Los datos se guardan en posiciones decrementales de a 4[bytes] con respecto a la anterior y se leen incrementando 4[bytes] de la anterior.
- Ascending Stack: Similar al anterior, pero inversamente.

Así se puede definir 4 tipos de “Stack”: FD, ED, FA y EA.

El modo “FD” (Full Descending) es el más común. Se utiliza en el registro “r13” como “SP” (Stack Pointer). En las instrucciones “pop” y “push”, es el modo por defecto.

A continuación se muestra gráficamente cómo funcionan ambas instrucciones.

La siguiente figura es un ejemplo de “push”, el cual se describe a continuación:

`push {r1,r2}` \longrightarrow `stmfd r13!,{r1,r2}`

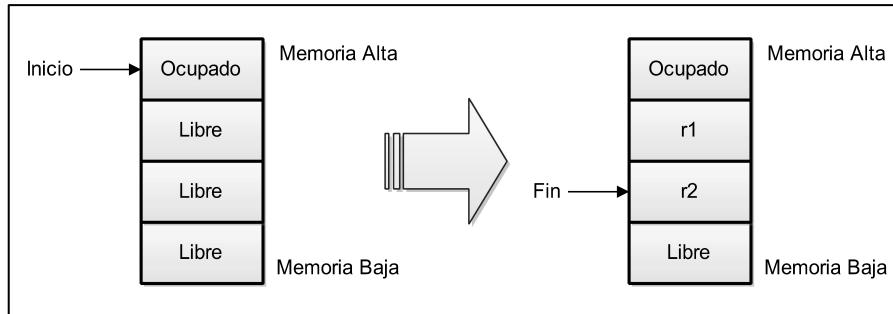


Figura 3: -Final 03.12.09- Ejemplo de la instrucción “push”

Ahora se muestra un ejemplo de “pop”, el cual se describe a continuación:

`pop {r1,r2}` \longrightarrow `ldmfd r13!,{r1,r2}`

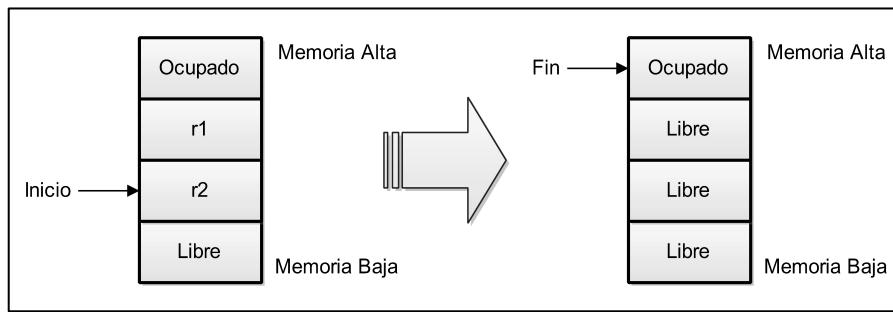


Figura 4: -Final 03.12.09- Ejemplo de la instrucción “pop”

Las instrucciones “pop” y “push” pueden guardar una lista (o todas) de los registros de propósitos generales (de “r0” a “r7”). Los registros a ser guardados se separan por comas y entre llaves, como por ejemplo: `push {r0,r1}`.

8. Primero hay que hacer algunas consideraciones:

Datos:

- Tamaño memoria RAM=2[Kb].
- Dirección inicio=0x4E800.
- Líneas de direccionamiento interno= A19, A18 y A15.
- Lecturas en: 0x7FFF, 0x46800 y 0x6A9FF.

Desarrollo

$$RAM_de_2[kb] \Rightarrow$$

$$\Rightarrow n_{RAM} = \frac{\ln(RAM_de_2[kb])}{\ln(2)} = \frac{\ln(2048)}{\ln(2)} \Rightarrow n_{RAM} = 11[bits] \quad (22)$$

Luego, respecto de las líneas de direccionamiento, se tiene 5 cifras hexadecimal, por lo tanto:

$$n_{address} = (cant_cifras_{hexa}.4) - 1 = (5.4) - 1 \Rightarrow n_{address} = 19[bits] \quad (23)$$

Con estos datos se arma el mapa de direccionamiento, como se muestra a continuación:

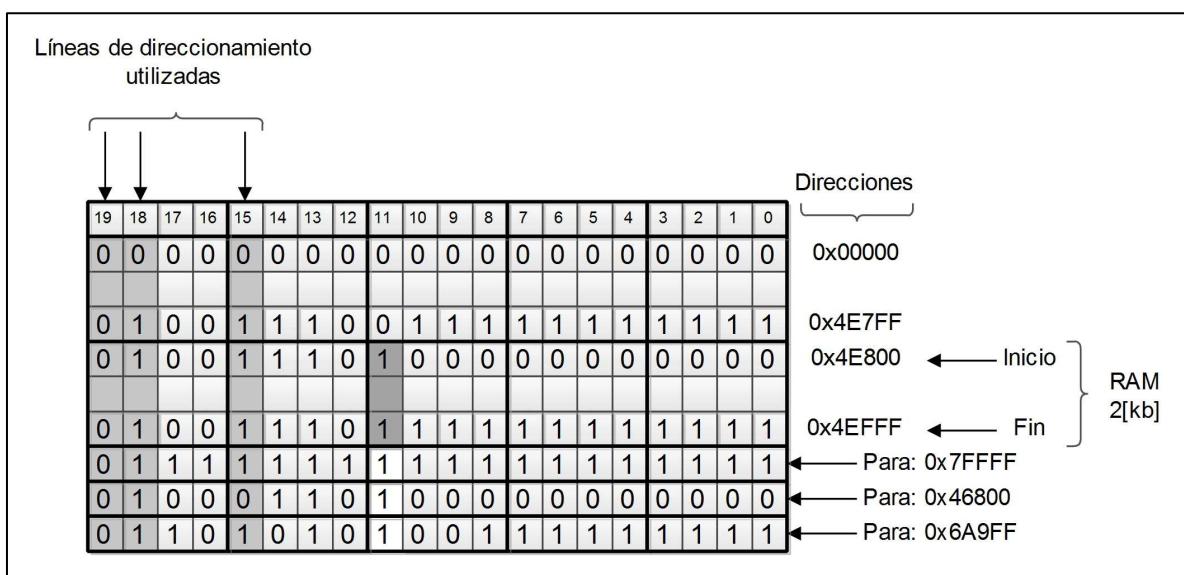


Figura 5: -Final 03.12.09- Mapeo de memoria RAM solicitada y direcciones a analizar

Por lo tanto, se ve claramente que para los casos se tiene:

- Para 0x7FFF **se lee en 0x4EFFF**.
- Para 0x46800 **NO se lee**.
- Para 0x6A9FF **se lee en 0x4A9FF**.

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 04 de febrero de 2010.

1. Se desea procesar una señal de un puente de galgas extensiométricas. La salida del puente es de $1,086[\mu\text{V}]$ por cada kg. de peso. Se debe poder pesar hasta $120[\text{kg}]$. la resolución demanda es de $100[\text{g}]$.
2. En la señal de 1., se requiere especificar el número de bits del conversor a usar.
3. Para usar un conversor con referencia de $10,0[\text{V}]$, diseñar el circuito de adaptación de señal.
4. Se requiere un error total igual a $0,1[\%]$. Especifique los componentes de error admisibles en la etapa anterior.
5. Diseñe con componentes discretos (transistores y compuertas) un multiplexor analógico de cuatro canales. El canal activo se comanda con la variable “canal” $\{00,01,10,11\}$ y la salida con la variable “sal” $\{0,1\}$. Existirá salida cuando sal=0.
6. Escriba el programa en Assembler que lea 16 veces la salida del conversor, sume cada una de las lecturas en una memoria de dimensión adecuada, y luego calcule el promedio de las lecturas.
7. Mapee una memoria de $32[\text{kb}]$ en un mapa de $1[\text{Mb}]$, de modo que no existan imágenes sino a intervalos de $64[\text{kb}]$.

Para aprobar el examen se requieren cinco puntos totalmente correctos, los que deben incluir tres del conjunto de $\{3, 4, 5, 6, 7\}$.

Resolución

Cálculos:

1. Datos:

$$\text{Paso} = 1,086[\mu\text{V}/\text{kg}]$$

$$\text{Peso máximo} = 120[\text{kg}]$$

$$\text{Resolución} = 100[\text{g}]$$

$$\text{cuentas} = \frac{\text{peso_máximo}}{\text{resolución}} = \frac{120[\text{kg}]}{100[\text{g}]} \Rightarrow \text{cuentas} = 1200[\text{cuentas}] \quad (1)$$

$$\begin{aligned} V_{\max} &= \text{peso_máximo} \cdot \text{paso} = 120[\text{kg}] \cdot 1,086[\mu\text{V}/\text{kg}] \Rightarrow \\ &\Rightarrow V_{\max} = 130,32[\mu\text{V}] \end{aligned} \quad (2)$$

$$2. \quad n_{A/D} = \frac{\ln(\text{cuentas})}{\ln(2)} = \frac{\ln(1200[\text{cuentas}])}{\ln(2)} \Rightarrow n_{A/D} = 10,22882[\text{bits}] \quad (3)$$

Por lo tanto se utilizará un **conversor A/D de 12[bits]**, ya que se consigue en el mercado en forma fácil y económica.

3. El cálculo de la ganancia “G” a utilizar es el siguiente:

$$\begin{aligned} G &= \frac{V_{ref}}{V_{LSB}} = \frac{\frac{V_{ref}}{2^{n_{A/D}}}}{\text{paso.resolución}} = \frac{\frac{10[\text{V}]}{2^{12}}}{1,086[\mu\text{V}/\text{kg}] \cdot 100[\text{g}]} \Rightarrow \\ &\Rightarrow G = 22480,7[\text{veces}] \end{aligned} \quad (4)$$

Por lo tanto se utilizarán 2 AI en cascada, donde la ganancia de cada uno es:

$$G_{AI} = \sqrt{G} = \sqrt{22480,7[\text{veces}]} \Rightarrow G_{AI} = 149,93585[\text{veces}] \quad (5)$$

A continuación se muestra el AI a utilizar:

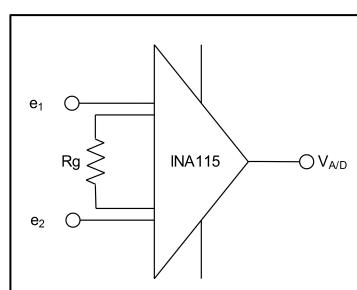


Figura 1: -Final 04.02.10- AI a utilizar en cascada. Los dos son idénticos

El mismo tiene la siguiente función de transferencia:

$$G_{AI} = 1 + \frac{50[k\Omega]}{R_g} \Rightarrow R_g = \frac{50[k\Omega]}{(G_{AI} - 1)} = \frac{50[k\Omega]}{(149,93585[\text{veces}] - 1)} \Rightarrow \\ \Rightarrow R_g = 335,71501[\Omega]$$
(6)

4. Si el error total “ ε ” es igual a 0,1[%], se tiene:

$$\varepsilon = 0,001 \cdot V_{ref} = 0,001 \cdot 10,0[V] \Rightarrow \varepsilon = 10[mV]$$
(7)

Ahora, “ ε ” se descompone en:

$$\varepsilon = \varepsilon_{A/D} + \varepsilon_G + \varepsilon_{V_{ref}} \Rightarrow \\ \Rightarrow \varepsilon_G = \varepsilon_{V_{ref}} = \frac{\varepsilon - \varepsilon_{A/D}}{2} = \frac{\varepsilon - \frac{V_{LSB}}{2^{n_{A/D}}}}{2} = \frac{\varepsilon - \frac{2^{n_{A/D}}}{2}}{2} = \frac{10[mV] - \frac{10[V]}{2^{12}}}{2} \Rightarrow \\ \Rightarrow \varepsilon_G = 4,38965[mV]$$
(8)

Por lo tanto la tolerancia es:

$$\text{tolerancia} = \frac{\varepsilon_G}{V_{ref}} = \frac{4,38965[mV]}{10,0[V]} \Rightarrow \text{tolerancia} = 438,96484[ppm]$$
(9)

O sea que se definirá una **tolerancia=250[ppm]**.

5. A continuación se muestra el multiplexor solicitado:

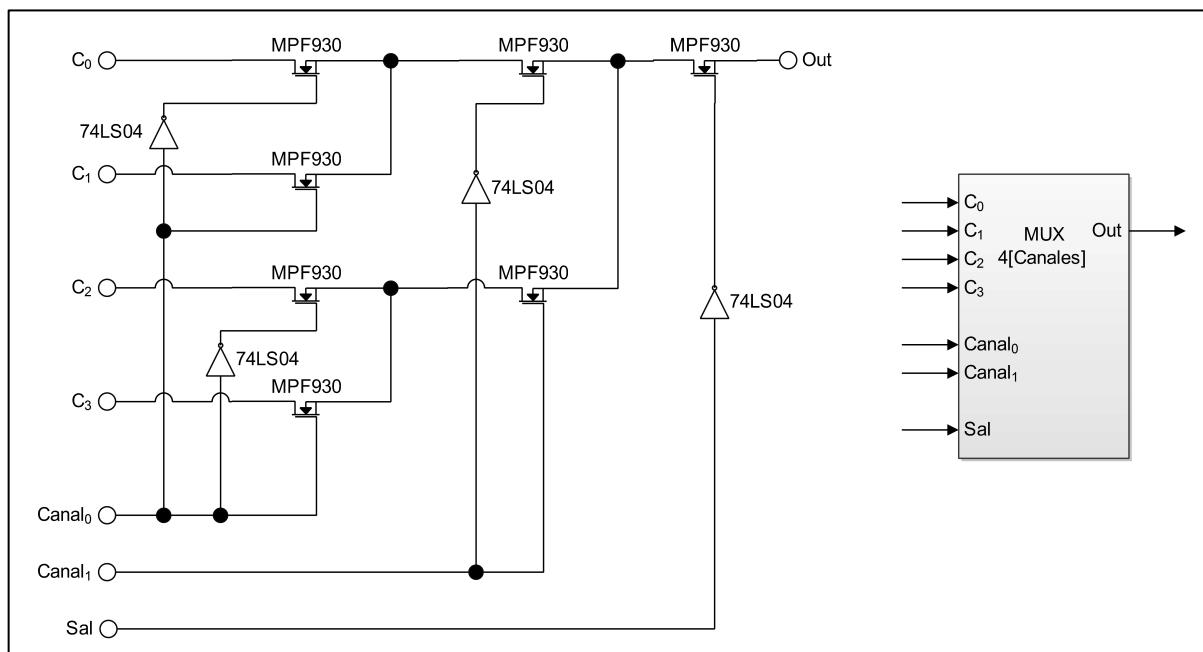


Figura 2: -Final 04.02.10- Circuito multiplexor solicitado

6. A continuación se muestra el desarrollo del programa solicitado:

reset:

```
mov r0,#16  
mov r1,#ADC  
mov r3,#0
```

lectura:

```
ldr r2,[r1],#4
add r3,r3,r2          //acumulador
subs r0,r0,#1
bne lectura           //salta si es ≠0
mov r2,r2,lsr#4       //divide por 16
mov r4,#PROM
str r2,[r4]
```

*/CONSTANTES

PROM: word

7 Datos:

- Memoria RAM=32[kb].
 - Mapa=1[Mb].
 - Intervalos entre imágenes=64[kb].

Antes de realizar el mapeo, se deben hallar algunos valores:

$$n_{RAM} = \frac{\ln(RAM_de_32[kb])}{\ln(2)} = \frac{\ln(32768)}{\ln(2)} \Rightarrow n_{RAM} = 15[bits] \quad (10)$$

$$n_{intervalos_imágenes} = \frac{\ln(intervalos_imágenes)}{\ln(2)} = \frac{\ln(65536)}{\ln(2)} \Rightarrow n_{RAM} = 16[bits] \quad (11)$$

$$n_{mapa} = \frac{\ln(mapa)}{\ln(2)} = \frac{\ln(1[Mb])}{\ln(2)} \Rightarrow n_{RAM} = 20[bits] \quad (12)$$

Ahora si se confecciona el mapeo de la memoria RAM, según los requisitos solicitados, mostrándose a continuación:

19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1

} RAM de 32[kb]

Figura 3: -Final 04.02.10- Mapeo solicitado de la memoria RAM

Finalmente, a continuación se muestra el circuito resultante del mapeo arriba presentado:

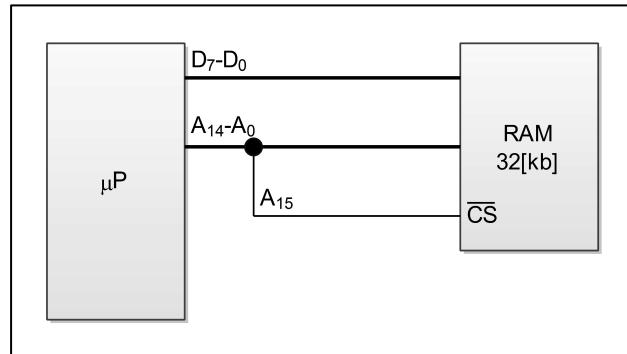


Figura 4: -Final 04.02.10- Circuito resultante del mapeo

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 11 de febrero de 2010.

1. Se debe realizar un módulo de captura que digitalice una señal de 2[kHz] cada una, las mismas poseen un rango de $\pm 0,5[V]$.
Las señales no son sinusoidales y se debe analizar hasta la 10^{ma} armónica.
La captura se realiza con un ADC de 12[bits] de 5,0[V] de referencia.
Se pide:
 - Resolución en [mV] de la señal.
 - Frecuencia de sampleo.
 - Diseño de la Etapa de Adaptación de señal (ganancia, valores de resistencias).
2. Diseñe un programa en assembler, que barra un vector de 1000 palabras de 16 bits con signo, encuentre el valor mínimo, y reste ese valor a cada elemento de ese mismo vector.
3. Diseñe el circuito selector que mapee dos memorias, una de 32[kb] x 8, y otra de 16[kb] x 8, en la parte más baja de un mapa de 1[Mb] x 8 y en forma correlativas, solo se pide el circuito selector, el cual tendrá como entrada las líneas de addres necesarias y como salida las dos señales de CS, las memorias se pueden mapear con imágenes.
4. Se debe diseñar un canal de transmisión serial, el mismo debe soportar un flujo de datos de: 1 trama de 24[bytes] cada 20[ms], se debe configurar el canal en modo 8N1, definir mínima velocidad necesaria (normalizada), calcular DLL y DLM para un clock de 1,8432[MHz].
5. Diseñe con componentes discretos (transistores y compuertas) un multiplexor analógico de cuatro canales. El canal activo se comanda con la variable “canal” {00,01,10,11} y la salida con la variable “sal” {0,1}. Existirá salida cuando sal=0.

Para aprobar el examen se requieren tres puntos totalmente correctos.

Resolución

Cálculos:

1. Datos:

$$f_s = 2[\text{kHz}].$$

$$V_s = \pm 0,5[V].$$

Señales NO sinusoidales.

Análisis hasta la 10^{ma} armónica.

$$n_{A/D} = 12[\text{bits}].$$

$$V_{\text{ref}} = 5,0[V].$$

$$\text{resolución} = \frac{V_{Sp-p}}{2^{n_{A/D}}} = \frac{1,0[V]}{2^{12}} \Rightarrow \text{resolución} = 0,2441[mV] \quad (1)$$

$$f_s|_{10^\circ_arm} = f_s \cdot 10 = 2[\text{kHz}] \cdot 10 \Rightarrow f_s|_{10^\circ_arm} = 20[\text{kHz}] \quad (2)$$

Por lo tanto, la frecuencia mínima de muestreo, de Nyquist es:

$$f_{m_Nyquist} = 2[\text{veces}] \cdot f_s|_{10^\circ_arm} = 2.20[\text{kHz}] \Rightarrow f_{m_Nyquist} = 40[\text{kHz}] \quad (3)$$

Pero se tomará la frecuencia de muestreo recomendada, la cual es:

$$f_m = 5[\text{veces}] \cdot f_s|_{10^\circ_arm} = 5.20[\text{kHz}] \Rightarrow f_m = 100[\text{kHz}] \quad (4)$$

Finalmente, el tiempo de conversión “t_m” será:

$$t_m = \frac{1}{f_m} = \frac{1}{100[\text{kHz}]} \Rightarrow t_m = 10[\mu\text{s}] \quad (5)$$

Por supuesto que al tratarse de una señal con parte positiva y negativa, la etapa de “adaptación” consiste en una parte de desplazamiento de nivel y otra de ganancia apta para entrar al conversor A/D. Por lo tanto, se realizan los siguientes cálculos:

$$G = \frac{V_{ref}}{V_{Sp-p}} = \frac{5,0[V]}{1,0[V]} \Rightarrow G = 5[\text{veces}] \quad (6)$$

$$V_{off} = 0,5[V] \quad (7)$$

A continuación se muestra el circuito completo de la adaptación de señal:

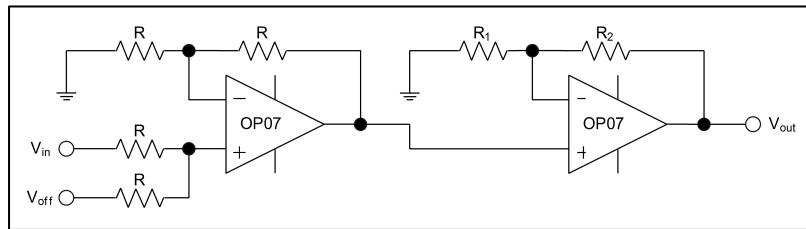


Figura 1: -Final 11.02.10- Circuito adaptador de señal

Donde se calculan los siguientes valores:

$$R = 10[k\Omega] \quad (8)$$

$$\begin{aligned} G = 1 + \frac{R_2}{R_1} &\Rightarrow R_2 = (G - 1).R_1 = (5[veces] - 1).R_1 \Rightarrow \\ R_2 &= 4[veces].R_1 \end{aligned} \quad (9)$$

O sea que si se selecciona:

$$R_1 = 10[k\Omega] \quad (10)$$

Aplicando (9), resulta:

$$R_2 = 4[veces].R_1 = 4.10[k\Omega] \Rightarrow R_2 = 40[k\Omega] \quad (11)$$

La tolerancia de los componentes es:

$$\varepsilon_G = \frac{1}{4}.V_{LSB} = \frac{1}{4} \cdot \frac{V_{ref}}{2^{n_{A/D}}} = \frac{1}{4} \cdot \frac{5,0[V]}{2^{12}} \Rightarrow \varepsilon_G = 305,176[\mu V] \quad (12)$$

Finalmente:

$$\text{tolerancia} = \frac{\varepsilon_G}{V_{ref}} = \frac{305,176[\mu V]}{5,0[V]} \Rightarrow \text{tolerancia} = 61,035[ppm] \quad (13)$$

Por lo tanto se tomara una **tolerancia=50[ppm]**.

2. El diagrama de flujo correspondiente a lo solicitado, es el siguiente:

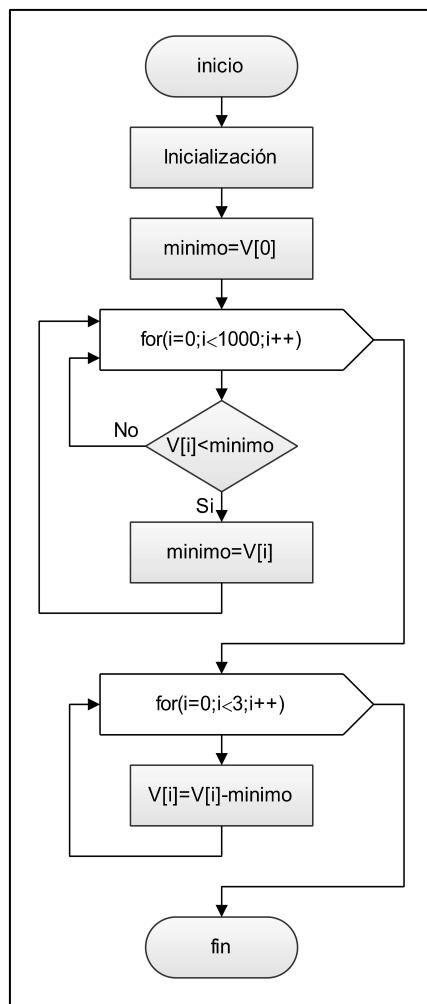


Figura 2: -Final 11.02.10- Diagrama de flujo correspondiente al programa solicitado

Y por lo tanto, la codificación en assembler es la siguiente:

reset:

```

mov r0,#VECT //Lectura del vector
mov r1,#1000 //Contador
mov r2,#0 //Mínimo
mov r3,#0 //Registro para hacer la resta
ldrsh r2,[r0],#2 //Inicialización del primer dato como mínimo
sub r1,r1,#1

```

lectura:

```

ldrsh r4,[r0],#2 //Carga el vector y luego incrementa "r0"
cmp r2,r4 //Compara cual de los dos registros es menor
movlt r3,r4 //Guarda en "r2" el nuevo mínimo (N=V)
subs r1,r1,#1
bne lectura //Salta mientras sea distinto de cero (Z=0)
mov r0,#VECT

```

```
mov      r1,#1000
```

cambio:

```
ldrsh    r4,[r0],#2
sub      r3,r4,r2      //Resta cada valor con el mínimo, alojado en "r2"
strh    r3,[r0]         //Guarda "r3" en memoria
subs    r1,r1,#1
bne     cambio
```

loop: b loop

*/Constantes

VECT: .half

3. Antes de realizar el mapeo y circuito correspondientes, se deben hacer algunos cálculos:

$$n_{RAM_32[kb]} = \frac{\ln(capac_{RAM_32[kb]})}{\ln(2)} = \frac{\ln(32[kb])}{\ln(2)} \Rightarrow n_{RAM_32[kb]} = 15[bits] \quad (14)$$

$$n_{RAM_16[kb]} = \frac{\ln(capac_{RAM_16[kb]})}{\ln(2)} = \frac{\ln(16[kb])}{\ln(2)} \Rightarrow n_{RAM_16[kb]} = 14[bits] \quad (15)$$

$$n_{mapa_1[Mb]} = \frac{\ln(capac_{mapa_1[Mb]})}{\ln(2)} = \frac{\ln(1[Mb])}{\ln(2)} \Rightarrow n_{mapa_1[Mb]} = 20[bits] \quad (16)$$

Por lo tanto, el mapa es el siguiente:

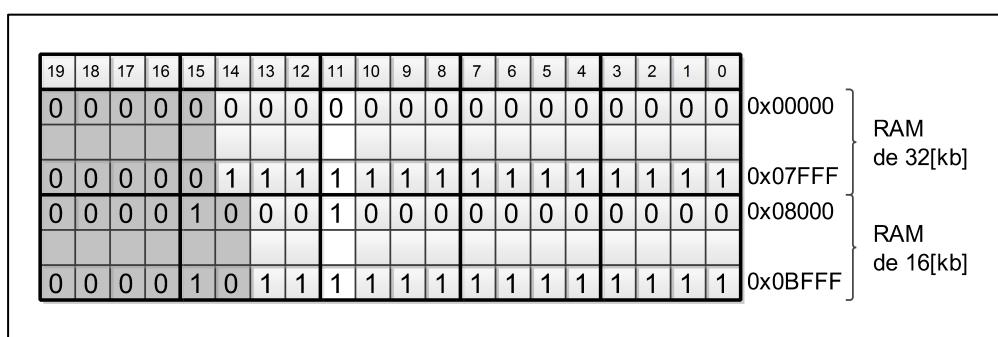


Figura 3: -Final 11.02.10- Mapa de memoria solicitado

El circuito selector de dicho mapeo es el siguiente:

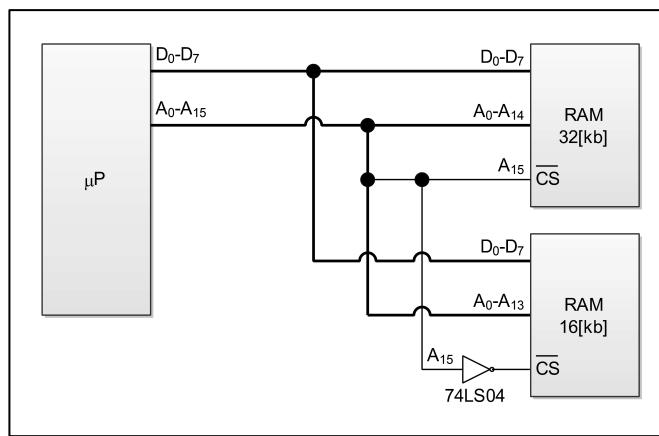


Figura 4: -Final 11.02.10- Circuito selector del mapeo anterior

4. Datos:

Modo del canal: 8N1.

Trama: 24[bytes] cada 20[ms].

f_{Clik}=1,8432[MHz].

Cálculos:

8N1 quiere decir: 8[bits] de datos, 1[bit] de stop, sin bit de paridad. A ello hay que sumarle 1[bit] de start. Por lo tanto, la trama en total tiene: **10[bits]**.

Luego:

$$\begin{aligned} 20[\text{ms}] &= 24[\text{bytes}] \\ 1[\text{s}] &= x[\text{bytes}] \Rightarrow \\ \Rightarrow x[\text{bytes}] &= \frac{24[\text{bytes}].1[\text{s}]}{20[\text{ms}]} \Rightarrow x[\text{bytes}] = 1200 \left[\frac{\text{bytes}}{\text{s}} \right] \end{aligned} \tag{17}$$

Por lo tanto, como cada byte está formado por 8[bits] de datos, se tiene una velocidad final de:

$$\begin{aligned} \text{velocidad} &= x[\text{bytes}].n_{\text{datos}} = 1200 \left[\frac{\text{bytes}}{\text{s}} \right].8[\text{bits}] \Rightarrow \\ \Rightarrow \text{velocidad} &= 9600 \left[\frac{\text{bits}}{\text{s}} \right] = 9600[\text{baudios}] \end{aligned} \tag{18}$$

A continuación:

$$\begin{aligned}
 baud_rate &= \frac{VPB}{16.DLR} \Rightarrow DLR = \frac{VPB}{16.baud_rate} = \frac{f_{clk}}{16.baud_rate} = \\
 &= \frac{1,8432[MHz]}{16.9600[baudios]} \Rightarrow DLR = 12[bits]
 \end{aligned} \tag{19}$$

Por lo tanto:

$$DLL = 12[bits] \tag{20}$$

$$DLM = 0[bits] \tag{21}$$

Por lo tanto, en hexadecimal, se tiene que:

$$DLR = 0x000C \tag{22}$$

Finalmente, recordando que la trama en total tiene 10[bits] por palabra, de acuerdo al baud rate, se tiene:

$$\begin{aligned}
 cant_caracteres_{seg} &= \frac{baud_rate}{n_{palabra}} = \frac{9600[baudios]}{10[bits]} \Rightarrow \\
 \Rightarrow cant_caracteres_{seg} &= 960 \left[\frac{caracteres}{s} \right]
 \end{aligned} \tag{23}$$

5. El circuito del multiplexor solicitado es el siguiente:

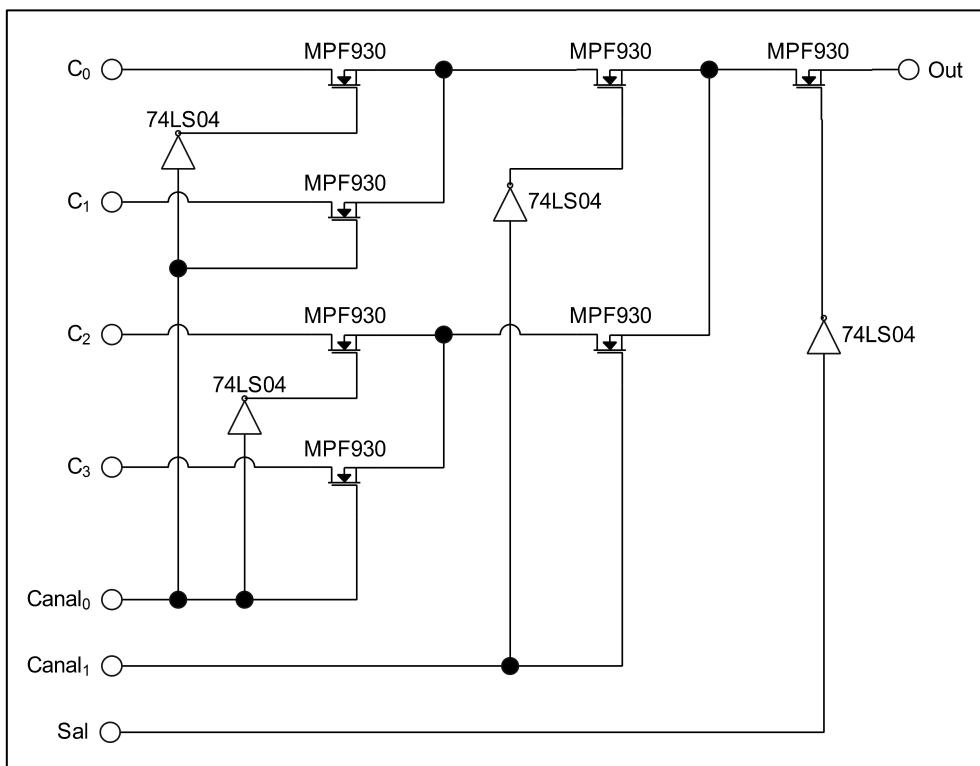


Figura 5: -Final 11.02.10- Circuito del multiplexor solicitado

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 25 de febrero de 2010.

Sea una señal que admite una expresión: $s(t) = 2,5 \cdot \text{sen}(4,18E5 \cdot t)[V]$.

1. ¿Cuál es la frecuencia de muestreo según Nyquist? ¿Cuál es la recomendable? ¿Cuál es el tiempo de conversión mínimo para este último caso?
2. En la señal de 1. se requiere discriminar la amplitud a 0,5[mV]. Especificar el número de bits del convertidor a usar.
3. Para usar un convertor de un solo cuadrante y referencia de 10,0[V], diseñar el circuito de adaptación de señal.
 - 3.1. Circuito electrónico usado.
 - 3.2. Componentes para que el error introducido por el circuito sea menor a $\frac{1}{4}$ lsb.
 - 3.3. Especificación de ripple de la(s) fuente(s) usada(s) para la misma especificación de error.
4. Se requiere el diseño del circuito que implemente la etapa de recepción de un canal RS 232-C a 4,8[kbps]. Usar solamente componentes discretos o lógica TTL MSI. Diseñar la interface al micro.
5. Diseñe con componentes discretos (transistores y compuertas) un multiplexor analógico de dos canales. El canal activo se comanda con la variable “canal” {0,1} y la salida con la variable “sal” {0,1}. Existirá salida cuando sal=0.

Para aprobar el examen se tener correctos por lo menos tres puntos, que deben incluir dos del conjunto {3,4,5}.

Resolución

Cálculos:

1. Antes que nada, lo primero es presentar la señal en su forma genérica:

$$s(t) = \hat{S} \cdot \sin(\omega t)[V] \quad (1)$$

Por lo tanto, lo primero que se encontrará es la frecuencia en [Hz] de la misma:

$$f_s = \frac{\omega}{2\pi} = \frac{4,18E5[\text{rad/s}]}{2\pi} \Rightarrow f_s = 66,527[\text{kHz}] \quad (2)$$

Ahora, la frecuencia de muestreo de Nyquist “ $f_{m_Nyquist}$ ” es:

$$\begin{aligned} f_{m_Nyquist} &= f_s \cdot 2[\text{veces}] = 66,527[\text{kHz}] \cdot 2[\text{veces}] \Rightarrow \\ &\Rightarrow f_{m_Nyquist} = 133,054[\text{kHz}] \end{aligned} \quad (3)$$

Pero, la frecuencia de muestreo recomendada “ f_m ” es:

$$\begin{aligned} f_m &= f_s \cdot 5[\text{veces}] = 66,527[\text{kHz}] \cdot 5[\text{veces}] \Rightarrow \\ &\Rightarrow f_m = 332,635[\text{kHz}] \end{aligned} \quad (4)$$

Por último, el tiempo de muestreo mínimo “ t_m ” es:

$$t_m = \frac{1}{f_m} = \frac{1}{332,635[\text{kHz}]} \Rightarrow t_m = 3,006[\mu\text{s}] \quad (5)$$

2. Lo primero que se calculará es la cantidad de cuentas necesarias para la conversión:

$$cuentas = \frac{S_{p-p}}{resolución} = \frac{5[V]}{0,5[mV]} \Rightarrow cuentas = 10000[cuentas] \quad (6)$$

Por lo tanto, para calcular la cantidad de bits del conversor A/D “ $n_{A/D}$ ” a utilizar, se hace de la siguiente manera:

$$n_{A/D} = \frac{\ln(cuentas)}{\ln(2)} = \frac{\ln(10000[cuentas])}{\ln(2)} \Rightarrow n_{A/D} = 13,288[bits] \quad (7)$$

Por lo tanto se utilizará un **conversor A/D de 14[bits]**.

3. Lo primero a calcular es la ganancia “ G ” necesaria para llevar los niveles adecuados, para ello:

$$G = \frac{V_{LSB}}{V_{paso}} \quad (8)$$

De donde:

$$V_{LSB} = \frac{V_{ref}}{2^{n_{A/D}}} = \frac{10,0[V]}{2^{14}} \Rightarrow V_{LSB} = 610,352[\mu V] \quad (9)$$

Reemplazando (9) en (8):

$$G = \frac{V_{LSB}}{V_{paso}} = \frac{610,352[\mu V]}{0,5[mV]} \Rightarrow G = 1,221[veces] \quad (10)$$

Por otro lado, como se trata de la conversión de una señal con partes positiva y negativa, y el conversor A/D es de un solo cuadrante, se deberá incorporar una tensión de offset “ V_{off} ” que lleve la señal a convertir a un solo cuadrante. El circuito de adaptación de señal es el siguiente:

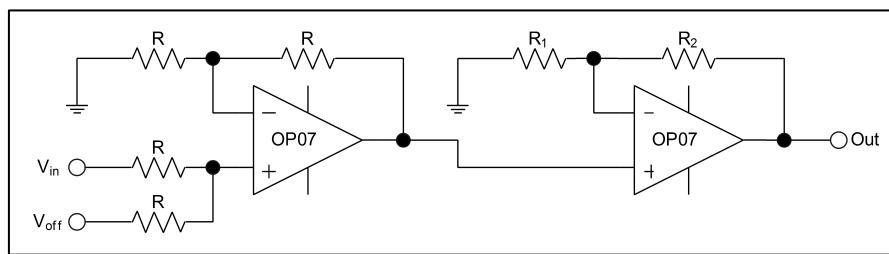


Figura 1: -Final 25.02.10- Circuito de adaptación

Donde los valores de los componentes para el mismo son los siguientes:

$$V_{off} = |\hat{S}| \Rightarrow V_{off} = 2,5[V] \quad (11)$$

$$R = 10[k\Omega] \quad (12)$$

$$G = 1 + \frac{R_2}{R_1} \Rightarrow R_2 = (G - 1).R_1 \quad (13)$$

Por lo tanto, seleccionando:

$$R_1 = 10[k\Omega] \quad (14)$$

Reemplazando (14) en (13):

$$R_2 = (G - 1).R_1 = (1,221[veces] - 1).10[k\Omega] \Rightarrow R_2 = 2,21[k\Omega] \quad (15)$$

Para averiguar la tolerancia de los componentes, se tiene como parámetro que el error introducido por el circuito, o sea, el circuito adaptador, es menor a $\frac{1}{4}$ lsb, por lo tanto se toma como parámetro el *error de ganancia*, “ ε_G ”, y se calcula:

$$\varepsilon_G = \frac{V_{LSB}}{4} = \frac{610,352[\mu V]}{4} \Rightarrow \varepsilon_G = 152,588[\mu V] \quad (16)$$

$$\text{tolerancia} = \frac{\varepsilon_G}{V_{ref}} = \frac{152,588[\mu\text{V}]}{10,0[\text{V}]} \Rightarrow \text{tolerancia} = 15,259[\text{ppm}] \quad (17)$$

Por lo tanto, para los componentes se tomará una **tolerancia=10[ppm]**.

Por último, el error de ripple “ $\varepsilon_{\text{ripple}}$ ” de las fuentes se calcula como:

$$\varepsilon_{\text{ripple}} = \frac{V_{\text{LSB}}}{4} = \frac{610,352[\mu\text{V}]}{4} \Rightarrow \varepsilon_{\text{ripple}} = 152,588[\mu\text{V}] \quad (18)$$

4. El circuito solicitado es el siguiente:

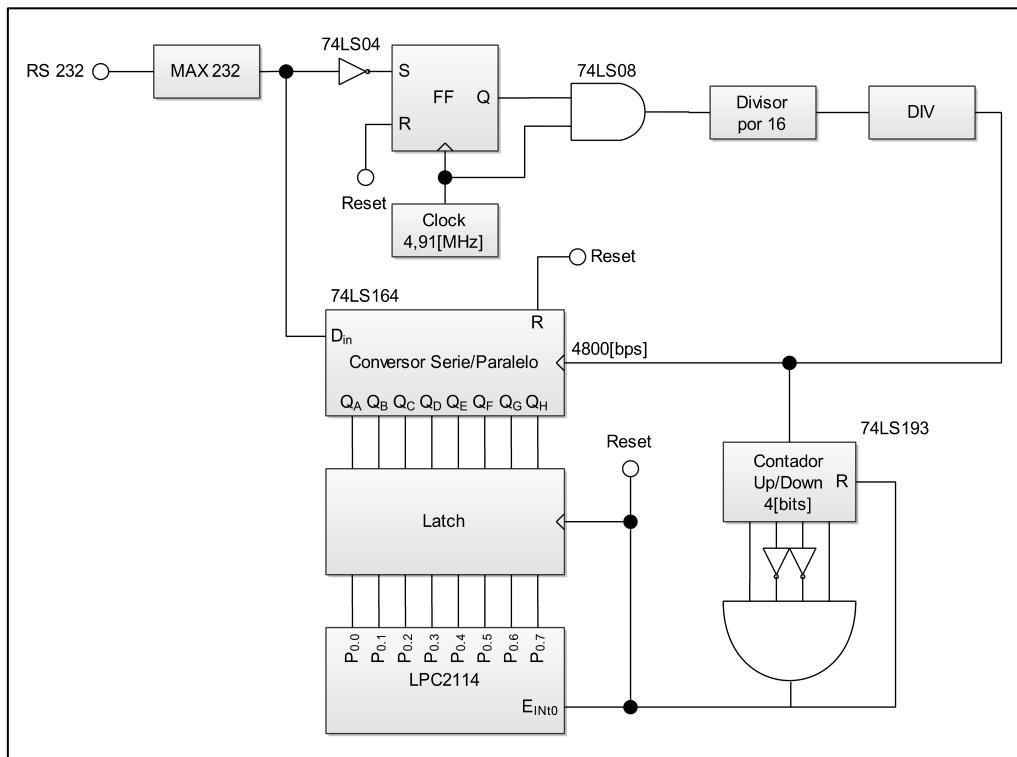


Figura 2: -Final 25.02.10- Circuito de interface al μC , de la recepción de un canal RS 232-C

Por último, el divisor “DIV” se calcula de la siguiente forma:

$$\begin{aligned} DIV &= \frac{f_{\text{Clk}}}{16.\text{baud_rate}} = \frac{4,91[\text{MHz}]}{16.4800[\text{bps}]} \Rightarrow \\ &\Rightarrow DIV = 63,932[\text{veces}] \Rightarrow DIV = 64[\text{veces}] \end{aligned} \quad (19)$$

5. El circuito del multiplexor solicitado es el siguiente:

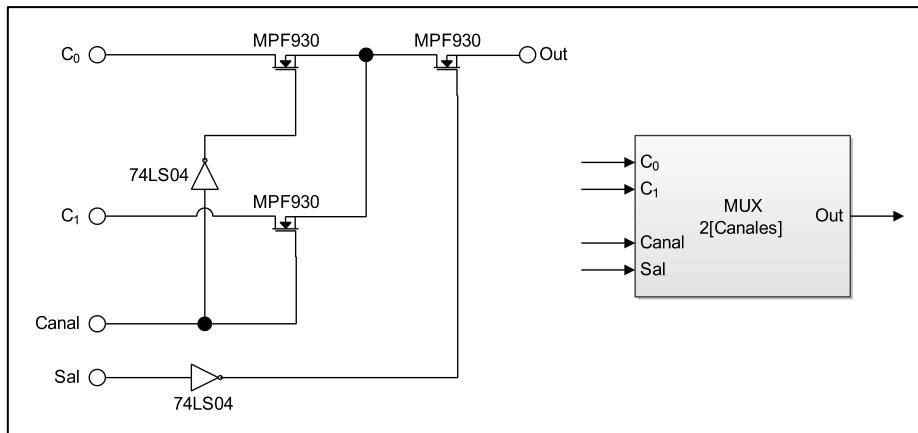


Figura 3: -Final 25.02.10- Circuito y representación del multiplexor solicitado

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 18 de noviembre de 2010.

1. Se debe controlar el arranque de un motor en modo estrella-triángulo.

1.a. Detalle el circuito a usar desde la salida de un puerto paralelo del microprocesador que Ud. estudió. Las condiciones que se exigen son:

- Debe estar optoacoplado (fuente para mando en cc: 12[V]).
- Debe accionar dos contactores tamaño 3, con mando de 220[Vca] y consumo de 350[mA] cada uno.
- La secuencia es: cierre contacto estrella / delay de 4[s] / apertura estrella / cierre triángulo.
- Dibuje el circuito completo, incluyendo el temporizador que use, que debe actuar por interrupción.

1.b. Dibuje el circuito de mando del motor, con las conexiones desde los contactores.

2. Dibuje el circuito de adaptación de señal de una termocupla.

- Rango de escala: 22 – 1600[μ V], equivalente a 0 – 400[°C].
- Resolución deseada: 0,1[°C].
- El conversor usa una referencia de 10,0[V].
- Se desea un error $e < 0,1[\%]$. Especificar componentes para este caso.

3. Diseñe con componentes discretos (transistores y compuertas) un multiplexor analógico de dos canales. El canal activo se comanda con la variable “canal” {0,1} y la salida con la variable “sal” {0,1}. Existirá salida cuando sal=0.

Para aprobar se necesita tener correctos por lo menos dos puntos.

Resolución

Cálculos:

1. Datos:

Optoacoplador con fuente de 12[Vcc].

Contactores tamaño 3: 2 (220[Vca], 350[mA]).

Secuencia: contacto estrella → delay 4[s] → apertura estrella → cierre triángulo.

Circuito temporizador: comandado por interrupción.

Cálculos:

A continuación se muestra el circuito general de comando:

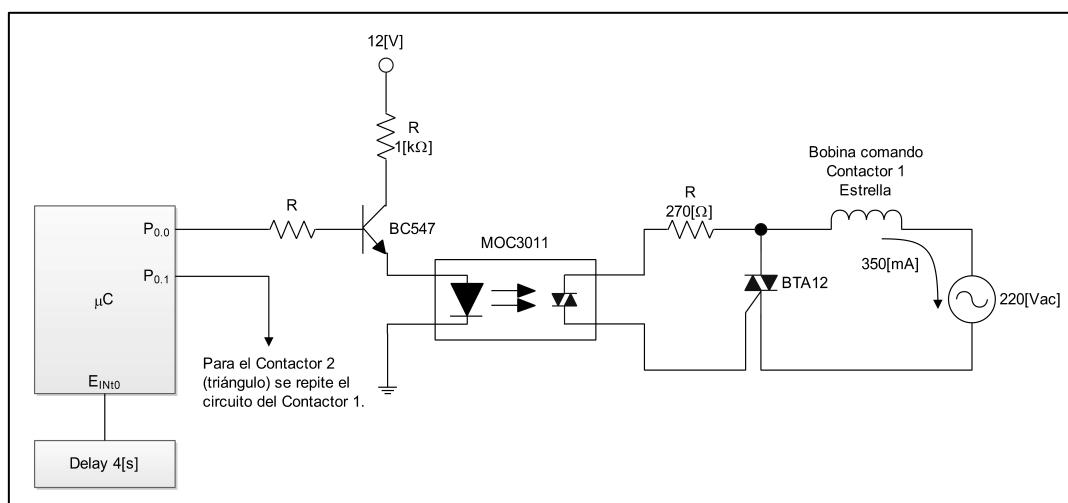


Figura 1: -Final 18.11.10- Circuito general de comando

Luego, para el delay de 4[s] se utiliza un LM555 configurado como monoestable, y su configuración se encuentra a continuación:

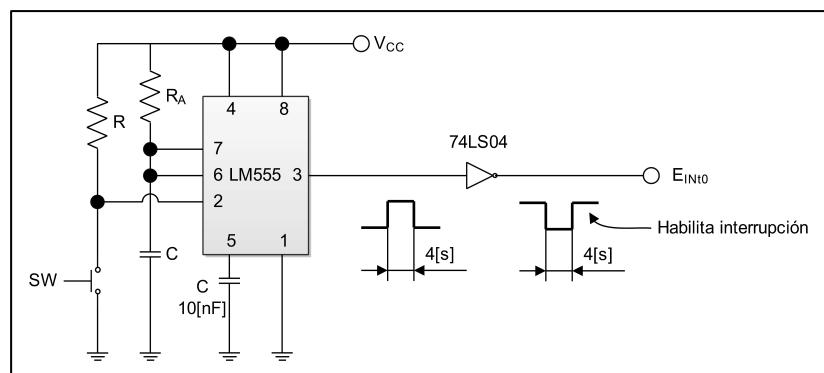


Figura 2: -Final 18.11.10- Circuito utilizado para generar el delay de 4[s]

La configuración anterior del LM555 como monoestable, responde a la siguiente función de transferencia:

$$T = 1,1 \cdot R_A \cdot C \Rightarrow R_A \cdot C = \frac{T}{1,1} \quad (1)$$

Como:

$$T = 4[s] \quad (2)$$

Reemplazando (2) en (1):

$$R_A \cdot C = \frac{T}{1,1} = \frac{4[s]}{1,1} \Rightarrow R_A \cdot C = 3,636[s] \quad (3)$$

Por lo tanto, si se selecciona uno de los dos valores incógnitas ("R_A", "C"), por ejemplo:

$$R_A = 330[k\Omega] \quad (4)$$

Reemplazando (4) en (3):

$$\begin{aligned} R_A \cdot C = 3,636[s] &\Rightarrow C = \frac{3,636[s]}{R_A} = \frac{3,636[s]}{330[k\Omega]} \Rightarrow \\ &\Rightarrow C = 11,019[\mu F] \Rightarrow C \cong 11[\mu F] \end{aligned} \quad (5)$$

El circuito de potencia es el siguiente:

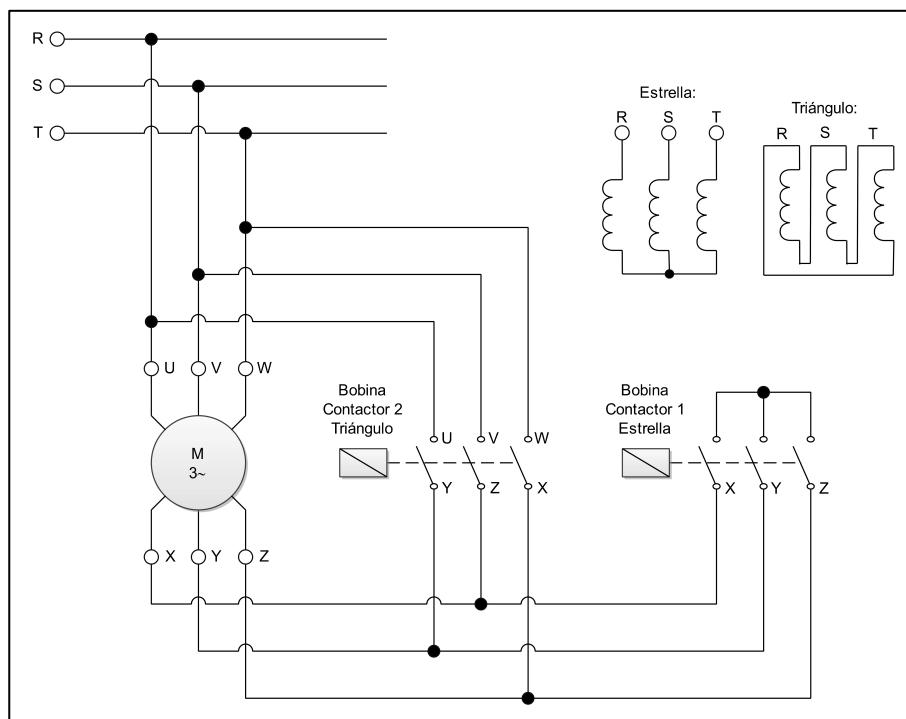


Figura 3: -Final 18.11.10- Circuito de potencia para un arranque estrella-triángulo

2. Datos:

Rango de escala: 22 - 1600[μ V], equivalentes a 0 – 400[°C].

Resolución deseada: 0,1[°C].

Referencia: 10,0[V].

Error total: $\varepsilon < 0,1[\%]$.

Cálculos:

$$cuentas = \frac{rango_{\circ C}}{resolución} = \frac{400[\circ C]}{0,1[\circ C]} \Rightarrow cuentas = 4000[cuentas] \quad (6)$$

$$n_{A/D} = \frac{\ln(cuentas)}{\ln(2)} = \frac{\ln(4000[cuentas])}{\ln(2)} \Rightarrow n_{A/D} = 11,966[bits] \quad (7)$$

Por lo tanto se utilizará un **conversor A/D de 12[bits]**.

$$G = \frac{V_{LSB}}{V_{paso}} \quad (8)$$

$$V_{LSB} = \frac{V_{ref}}{2^{n_{A/D}}} = \frac{10[V]}{2^{12}} \Rightarrow V_{LSB} = 2,441[mV] \quad (9)$$

$$V_{paso} = \frac{rango_{V_termocupla}}{cuentas} = \frac{1600[\mu V] - 22[\mu V]}{4000[cuentas]} \Rightarrow V_{paso} = 394,5[nV] \quad (10)$$

Reemplazando (9) y (10) en (8):

$$G = \frac{V_{LSB}}{V_{paso}} = \frac{2,441[mV]}{394,5[nV]} \Rightarrow G = 6188[veces] \quad (11)$$

El circuito de adaptación es el siguiente:

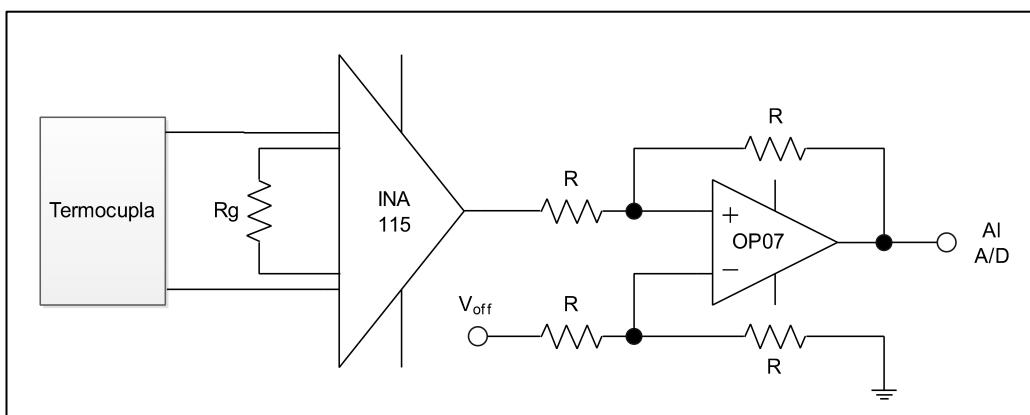


Figura 4: -Final 18.11.10- Circuito de adaptación solicitado

En el circuito propuesto se puede apreciar que la adaptación de señal se compondrá de dos etapas, por un lado la correspondiente a la ganancia propiamente dicha por medio de un AI, y la otra un desplazamiento del nivel inicial, para aprovechar todo el rango de la conversión.

Luego, los componentes a calcular del circuito son:

$$\begin{aligned} G = 1 + \frac{50[k\Omega]}{R_g} &\Rightarrow R_g = \frac{50[k\Omega]}{(G-1)} = \frac{50[k\Omega]}{(6188[veces]-1)} \Rightarrow \\ &\Rightarrow R_g = 8,08[\Omega] \end{aligned} \quad (12)$$

El desplazamiento se realiza por medio de una tensión de offset “ V_{off} ” que se introduce en un AO “OP07” en la entrada inversora, junto con la señal de entrada amplificada por el AI en la entrada no inversora. Por lo tanto se tendrá una V_{off} de valor negativo, igual a:

$$V_{off} = V_{termocupla_mín}.G = 22[\mu V].6188[veces] \Rightarrow V_{off} = 136,136[mV] \quad (13)$$

Finalmente, la tolerancia de los componentes se calcula como:

$$\begin{aligned} \varepsilon < 0,1[\%] &\Rightarrow \varepsilon = 0,1[\%].V_{ref} = 0,001.10,0[V] \Rightarrow \\ &\Rightarrow \varepsilon = 10[mV] = \varepsilon_{A/D} + \varepsilon_G + \varepsilon_{V_{ref}} \end{aligned} \quad (14)$$

Luego:

$$\varepsilon_{A/D} = \frac{1}{2}.V_{LSB} = \frac{1}{2}.2,441[mV] \Rightarrow \varepsilon_{A/D} = 1,221[mV] \quad (15)$$

Reemplazando (15) en (14) y operando:

$$\begin{aligned} \varepsilon = \varepsilon_{A/D} + \varepsilon_G + \varepsilon_{V_{ref}} &\Rightarrow \varepsilon_G = \varepsilon_{V_{ref}} = \frac{\varepsilon - \varepsilon_{A/D}}{2} = \frac{10[mV] - 1,221[mV]}{2} \Rightarrow \\ &\Rightarrow \varepsilon_G = 4,39[mV] \end{aligned} \quad (16)$$

Por lo tanto:

$$\text{tolerancia} = \frac{\varepsilon_G}{V_{ref}} = \frac{4,39[mV]}{10,0[V]} \Rightarrow \text{tolerancia} = 439[ppm] \quad (17)$$

O sea que se tomará una **tolerancia=250[ppm]**.

3. El circuito del multiplexor solicitado es el siguiente:

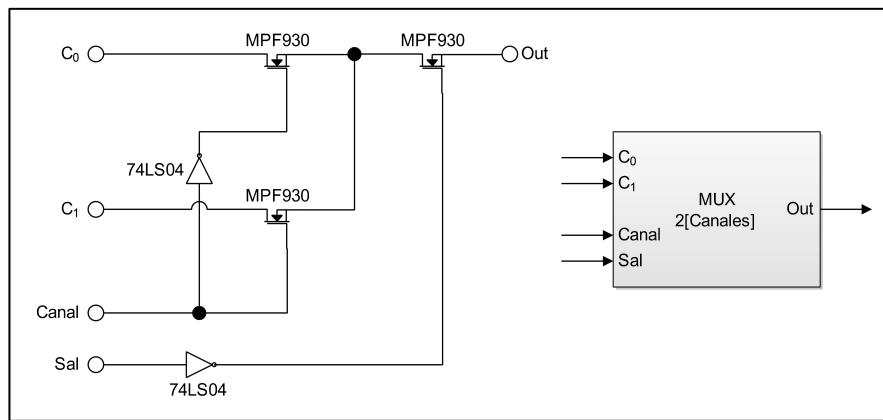


Figura 5: -Final 18.11.10- Circuito del multiplexor solicitado

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 27 de julio de 2011.

Usando un procesador basado en arquitectura ARM, resuelva los siguientes problemas:

1. Un transductor genera información con una función de transferencia de $2,5[\mu\text{A}/\text{bit}]$.
 - 1.a. Diseñe la interface del mismo hacia un conversor A/D de 12[bits], con referencia de $5,0[\text{V}]$. La señal es de cc.
 - 1.b. Establezca las tolerancias de los componentes pasivos de su circuito para un error total de $0,1[\%]$.
 - 1.c. Diseñe el hardware necesario para que se muestre esa señal a una tasa de $10[\text{Hz}]$, operando por interrupción. Debe incluir el reloj externo y su conexión con el controlador de interrupción del ARM, detallando la configuración del mismo.
2. Un display formado por dígitos LEDs de siete segmentos conectado a un puerto del microcontrolador, mostrará el valor leído, el mismo es de cátodo común y se opera de forma multiplexada (un dígito a la vez) a $50[\text{Hz}]$.
 - 2.a. ¿Cuántos dígitos deben usarse?
 - 2.b. Dibuje el circuito asociado al display (un solo dígito). Establezca el conjunto de señales necesarias para que este display opere adecuadamente (reloj, líneas de control de flujo de datos, enables, etc.).
 - 2.c. Escriba el diagrama de flujo del programa que controla la operación de su micro en relación a este display. De algún modo este diagrama debe incluir el clock de comando del sistema. Este clock se establece en $50[\text{Hz}]$.

Para aprobar, cuatro de los puntos de {1i, 2i} deben estar totalmente correctos.

Resolución

Cálculos:

1. Datos:

Función de transferencia=2,5[$\mu\text{A}/\text{bit}$].

$n_{\text{A/D}}=12[\text{bits}]$.

$V_{\text{ref}}=5,0[\text{V}]$.

$\varepsilon=0,1[\%]$.

Cálculos:

$$1.\text{a. } V_{\text{LSB}} = \frac{V_{\text{ref}}}{2^{n_{\text{A/D}}}} = \frac{5,0[\text{V}]}{2^{12}} \Rightarrow V_{\text{LSB}} = 1,221[\text{mV}] \quad (1)$$

Por lo tanto el circuito de adaptación es el siguiente:

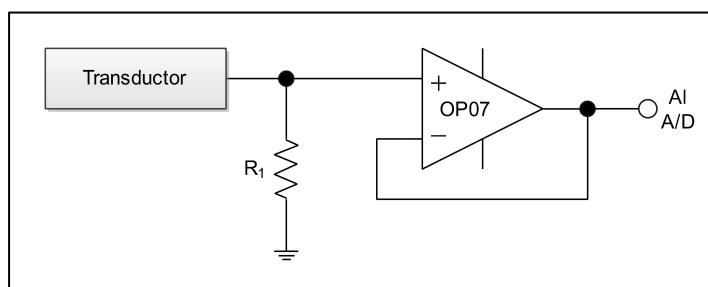


Figura 1: -Final 27.07.11- Circuito adaptador

Donde:

$$R_1 = \frac{V_{\text{LSB}}}{ft} = \frac{1,221[\text{mV}]}{2,5[\mu\text{A}/\text{bit}]} \Rightarrow R_1 = 488,4[\Omega] \quad (2)$$

$$1.\text{b. } \begin{aligned} \varepsilon &= 0,1[\%] = 0,001 \cdot V_{\text{ref}} = 0,001 \cdot 5,0[\text{V}] \Rightarrow \\ &\Rightarrow \varepsilon = 5[\text{mV}] = \varepsilon_{\text{A/D}} + \varepsilon_G + \varepsilon_{V_{\text{ref}}} \end{aligned} \quad (3)$$

$$\varepsilon_{\text{A/D}} = \frac{1}{2} \cdot V_{\text{LSB}} = \frac{1}{2} \cdot 1,221[\text{mV}] \Rightarrow \varepsilon_{\text{A/D}} = 305,25[\mu\text{V}] \quad (4)$$

Reemplazando (4) en (3) y despejando:

$$\begin{aligned} \varepsilon &= \varepsilon_{\text{A/D}} + \varepsilon_G + \varepsilon_{V_{\text{ref}}} \Rightarrow \varepsilon_G = \varepsilon_{V_{\text{ref}}} = \frac{\varepsilon - \varepsilon_{\text{A/D}}}{2} = \\ &= \frac{5[\text{mV}] - 305,25[\mu\text{V}]}{2} \Rightarrow \varepsilon_G = 2,347[\text{mV}] \end{aligned} \quad (5)$$

Por lo tanto, la tolerancia de los componentes será:

$$\text{tolerancia} = \frac{\mathcal{E}_G}{V_{ref}} = \frac{2,347[mV]}{5,0[V]} \Rightarrow \text{tolerancia} = 469,4[\text{ppm}] \quad (6)$$

O sea que se utilizará una **tolerancia=250[ppm]**.

1.c. El circuito solicitado es el siguiente:

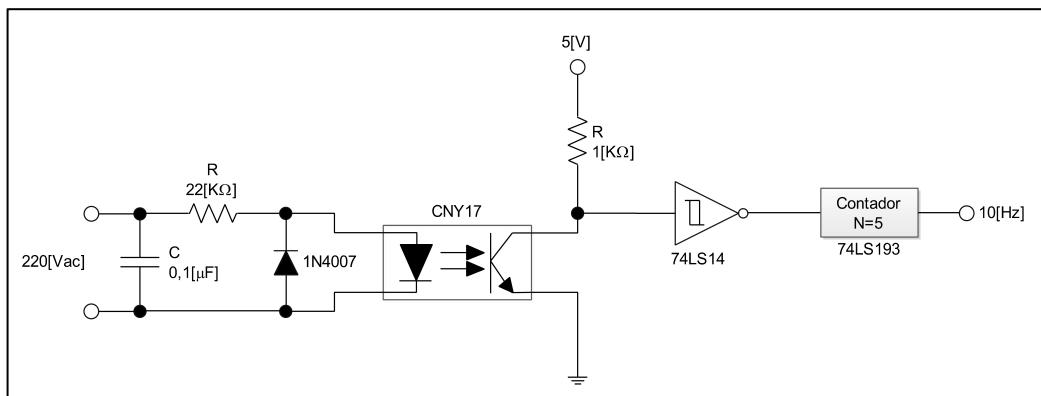


Figura 2: -Final 27.07.11- Circuito de muestreo solicitado

2. Datos:

Display de dígitos LEDs.

Display cátodo común.

Conexión multiplexada.

$f_{\text{display}} = 50[\text{Hz}]$.

2.a. Como $n_{A/D}=12[\text{bits}]$, se tendrá:

$$\text{cuentas} = 2^{n_{A/D}} = 2^{12} \Rightarrow \text{cuentas} = 4096[\text{cuentas}] \quad (7)$$

Por lo tanto se necesitarán **4 dígitos**.

2.b. Primero, un pequeño bosquejo de un dígito, de cómo es su funcionamiento:

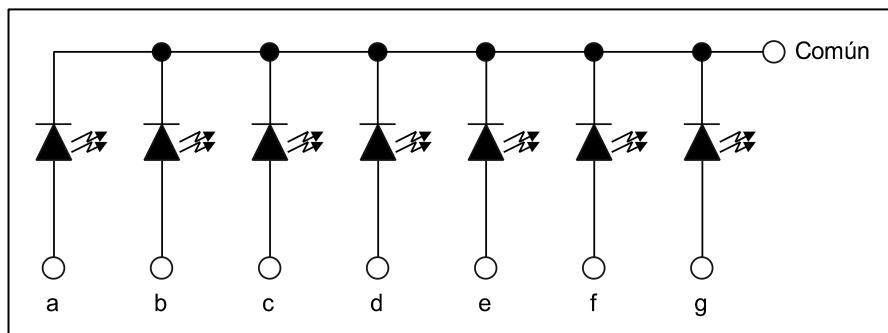


Figura 3: -Final 27.07.11- Esquema de un display LED de 7 segmentos

En el cual, hay que tener en cuenta los siguientes valores del microcontrolador ARM:

- $V_O=3,3[V]$.
- $I_O=3[mA]$.

Ahora sí, teniendo en cuenta los valores citados anteriormente, el circuito para 1 display es el siguiente:

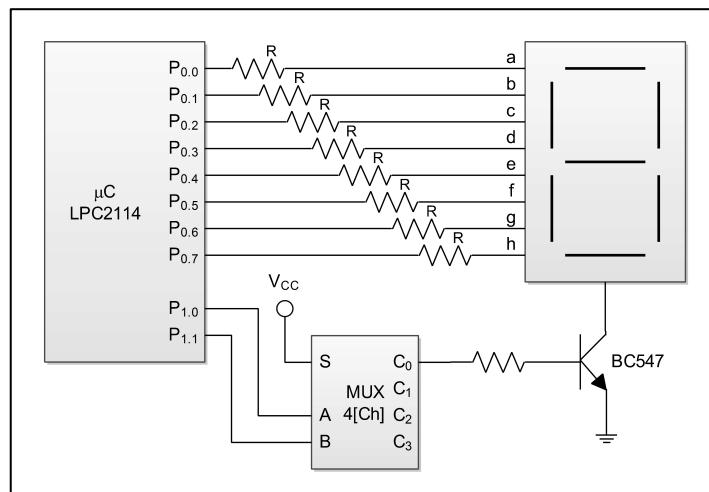


Figura 4: -Final 27.07.11- Circuito para el manejo de un display LED de 7 segmentos

Hay que tener en cuenta que “P_{1.0}” y “P_{1.1}” forman un contador de 2[bits] con una frecuencia de 50[Hz].

Para calcular las resistencias “R”, se procede de la siguiente manera:

$$R = \frac{V_{CC} - V_F - V_{CE_SAT}}{I_F} = \frac{3,3[V] - 1,5[V] - 0,2[V]}{3[mA]} \Rightarrow \\ \Rightarrow R = 533,33[\Omega]$$
(8)

Entonces se utilizará una **R=560[Ω]**.

2.c. El diagrama de flujo correspondiente, es el siguiente:

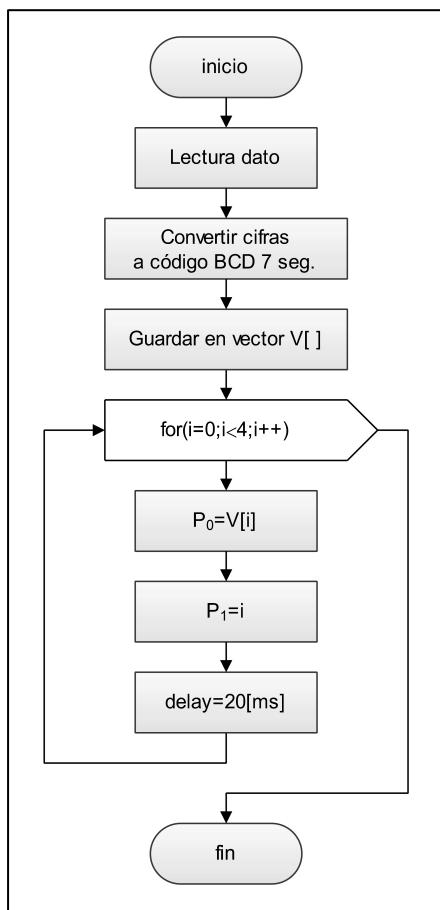


Figura 5: -Final 27.07.11- Diagrama de flujo utilizado para el muestreo en los displays

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 13 de octubre de 2011.

Nota: todo lo relacionado al microprocesador está expresado en primer término a INTEL y entre paréntesis al ARM. Cada uno debe seleccionar uno de ellos para su respuesta.

1. Escriba una secuencia de instrucciones que recorra un bloque de 250[bytes] de memoria. El bloque comienza en la dirección contenida en la 200h. este programa debe contar todos los números pares y los números negativos. La dirección PAR de tamaño byte en la memoria debe contener la cuenta de los números pares y la dirección NEGATIVO debe contener la cuenta de los números negativos. (2P).
2. ¿Cuál es la diferencia entre las instrucciones NEG (NEG) y NOT (MVN)? (1P).
3. Indicar el significado de las instrucciones JG y JL (BGT y BLT) a diferencia de JA y JB (BHI y BLO). (1P).
4. ¿cuántos tipos diferentes de interrupciones (excepciones) están disponibles en el microprocesador? Ilustre el contenido de un vector de interrupción y explique la finalidad del mismo. (1,5 P).
5. Describir e indicar los nombres de las líneas que constituyen la RS-232-C completa. (1P).
6. Desarrolle una secuencia corta de instrucciones que borre los tres bits más a la izquierda en DH (R1), sin cambiar DH (R1) y almacene el resultado en BH (R2). (1P).
7. Si el 8250 debe generar una señal serial a una frecuencia de 2400[baudios] y el divisor de la tasa de baudios está programado para 16 ¿Cuál es la frecuencia de reloj conectado al 8250? (0,75 P).
8. Si se desea convertir una señal sinusoidal de entrada de 1[MHz] con un conversor de 8[bits] de aproximaciones sucesivas, ¿qué tiempo máximo de conversión se necesita para el A/D? ¿Cuál será la frecuencia mínima de reloj interno del A/D? (0,75 P).
9. En un conversor A/D tipo flash, ¿qué cantidad de comparadores se necesitan? (0,5 P).
10. ¿Cuál grupo de terminales es utilizado durante la operación bidireccional del 8255? (0,5 P).

NOTA: Punto 10 no resuelto.

Resolución

Cálculos:

1. Datos:

Bloque de memoria=250[bytes].

inicio_{bloque}=200h.

Cálculos:

El diagrama en bloques del programa es el siguiente:

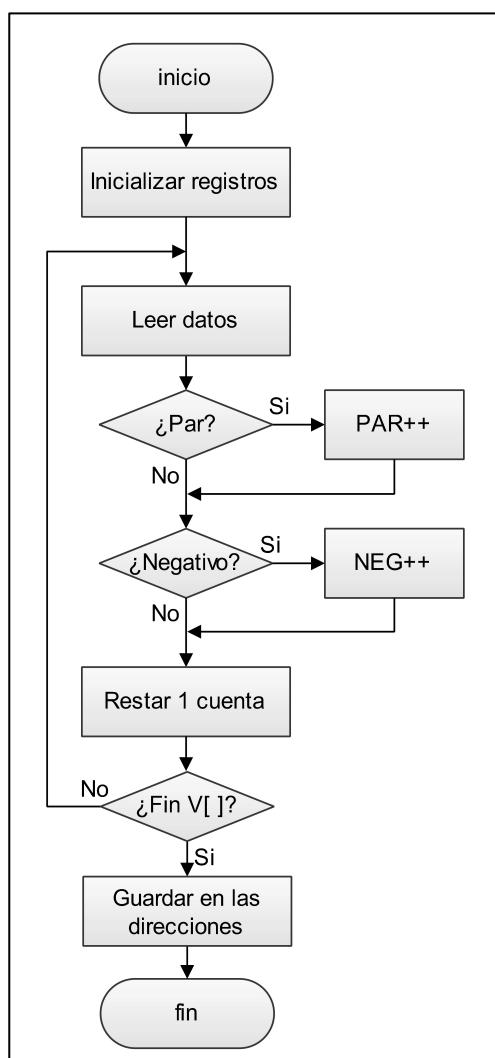


Figura 1: -Final 13.10.11- Diagrama de flujo del programa solicitado

A continuación se muestra el programa codificado en assembler:

```

        mov    r0,#0x200 /*Posición de memoria
        mov    r1,#250
        mov    r2,#0      /*Contador “par”
        mov    r3,#0      /*Contador “negativo”

contabilidad:
        ldr    r5,[r0],#1 /*Cargar valor y luego avanzar 1[byte]
        mvn    r4,#0
        mvn    r4,r4,lsl #7
        and    r4,r5,r4
        cmp    r4,#1
        addeq  r2,r2,#1
        cmp    r5,#0
        addlt  r3,#0      /*”lt” suma si es menor
        subs   r1,r1,#1
        bne    contabilidad

        mov    r6,#PAR
        str    r2,[r6]
        mov    r7,#NEG
        str    r3,[r7]

loop:   b loop

/*Constantes
#PAR      byte
#NEG      byte

```

- NEG: Negar, negativo del valor que tiene el registro. Complemento a 2. Modifica todas las banderas.

Sintaxis: NEG {Cond} {s} Rd, Op 2

Ejemplo: neg r2,r1 //Multiplica “r1” por (-1) y lo guarda en “r2” (r2=0-r1)

MVN: Mover negado, complementa el valor. Se usa para formar máscaras. Modifica sólo las banderas “N” y “Z”.

Sintaxis: MVN {Cond} {s} Rd, Op 2

Ejemplo: mvn r3,r4 //Genera el complemento a 1 de r4 y lo guarda en //r3 (NOT r4)

3. BGT: Salto con condición de mayor ($Z=0$, $N=V$).

BLT: Salto con condición de menor ($N=V$).

BHI: Salto con condición de sobre ($C=1$, $Z=0$).

BLO: Salto con condición de debajo ($C=0$).

La diferencia entre ambos pares de instrucciones es que los primeros (BGT y BLT) tienen en cuenta las banderas “N” y “V” para poder operar con números con signo; los segundos (BHI y BLO) sólo consideran la bandera “C” en la comparación.

4. Una “excepción” es cualquier condición que necesita alterar el funcionamiento normal del programa. Los microcontroladores ARM poseen 5 modos de excepción, a los que se puede ingresar desde 7 tipos de excepciones. A continuación se muestra gráficamente lo dicho anteriormente:

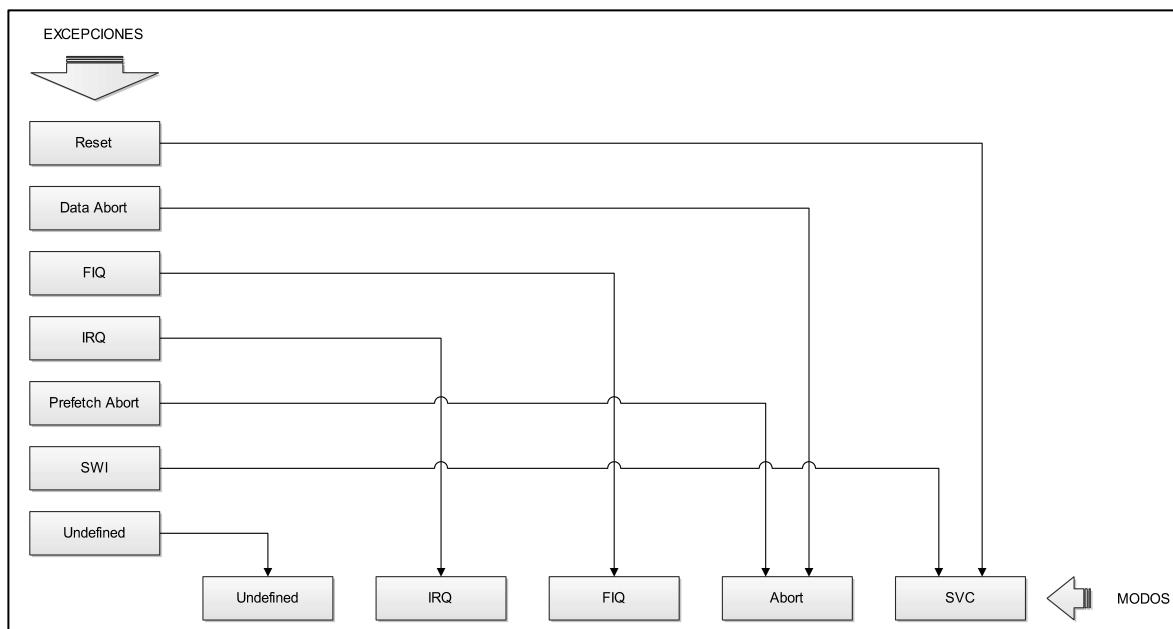


Figura 2: -Final 13.10.11- Diagrama de correspondencia entre excepciones y modos

El controlador de interrupciones “VIC” posee 32 entradas de interrupción en 3 categorías: “FIQ”, “IRQ vectorizado” y “IRQ no vectorizado”. Cada uno se describe a continuación:

FIQ (Fast Interrupt reQuest): Tienen la prioridad más alta.

IRQ vectorizada: Tienen prioridad media y es un vector de 16 elementos, donde el “0” tiene la mayor prioridad.

IRQ no vectorizada: Tienen la prioridad más baja.

Las fuentes de interrupción son 32, algunas de las cuales son:

TIMER 0: VIC channel 4.

EINT 0: VIC channel 14.

A/D: VIC channel 18.

5. GND (Common Ground): Debe estar conectado al chasis y a tierra.

TxD (Transmitted Data): Datos enviados del DTE al DCE.

RxD (Received Data): Datos enviados desde el DCE al DTE.

RTS (Request To Send): Solicitud del DTE para preparar al DCE a recibir un dato.

CTS (Clear To Send): Respuesta al RTS, para que el DTE transmita.

DTR (Data Terminal Ready): Solicitud del DTE que está listo para conectarse.

DSR (Data Set Ready): Solicitud del DCE que está listo para recibir del DTE.

DCD (Data Carrier Detect): Señal del DCE cuando la conexión ha sido establecida con el equipamiento remoto.

RI (Ring Indicator): Señal del DCE que indica que ha detectado una señal de llamada.

DTE: Equipo terminal de datos (PC).

DCE: Equipo de comunicación de datos (Módem).

6. La secuencia solicitada en assembler es la siguiente:

```
mvn    r3,#0      //r3=111....111
      mov    r3,r3,lsl #3 //r3=00011....1
      and    r2,r1,r3    //r2=r1 AND r3
```

7. Resolución para el LPC2114:

Datos:

Baud rate=2400[baudios].

DLR=16[bits].

Cálculos:

$$\begin{aligned} baud_rate &= \frac{VPB}{16.DLR} \Rightarrow \\ \Rightarrow VPB &= baud_rate \cdot 16.DLR = 2400[\text{baudios}] \cdot 16.16[\text{bits}] \Rightarrow \\ \Rightarrow VPB &= 614,4[\text{kHz}] \end{aligned} \quad (1)$$

8. Resolución para el LPC2114:

Datos:

$$f_{in}=1[\text{MHz}].$$

$$n_{A/D}=8[\text{bits}]$$

A/D de aproximaciones sucesivas.

Cálculos:

$$f_{m_Nyquist} = 2[\text{veces}] \cdot f_{in} = 2[\text{veces}] \cdot 1[\text{MHz}] \Rightarrow f_{m_Nyquist} = 2[\text{MHz}] \quad (2)$$

$$f_{m_recomendada} = 5[\text{veces}] \cdot f_{in} = 5[\text{veces}] \cdot 1[\text{MHz}] \Rightarrow f_{m_recomendada} = 5[\text{MHz}] \quad (3)$$

$$t_m = \frac{1}{f_{m_recomendada}} = \frac{1}{5[\text{MHz}]} \Rightarrow t_m = 200[\text{ns}] \quad (4)$$

Ahora, para un conversor A/D de aproximaciones sucesivas:

$$t_m = (n_{A/D} + 1) \cdot t_{Clk} \Rightarrow t_{Clk} = \frac{t_m}{n_{A/D} + 1} = \frac{200[\text{ns}]}{8[\text{bits}] + 1} \Rightarrow t_{Clk} = 22,22[\text{ns}] \quad (5)$$

$$f_{mín_A/D} = \frac{1}{t_{Clk}} = \frac{1}{22,22[\text{ns}]} \Rightarrow f_{mín_A/D} = 45[\text{MHz}] \quad (6)$$

9. La cantidad de comparadores en un conversor A/D tipo flash es la siguiente:

$$n_{comparadores} = 2^{n_{A/D}} - 1 = 2^8 - 1 \Rightarrow n_{comparadores} = 255[\text{comparadores}] \quad (7)$$

La cantidad de bits se tomo como ejemplo del caso anterior, o sea $n_{A/D}=8[\text{bits}]$.

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 24 de noviembre de 2011.

Se debe presentar en un display numérico, formado por lámparas halógenas del modo que muestra en la fig.1, el valor de la temperatura ambiente en grados Fahrenheit. La regla de conversión es:

$$T_F = T_C * 9/5 + 32$$

La información a presentar varía entre 14 y 122[°F] (-10 a 50[°C]), con resolución de 0,1[°F]. El transductor a usar es un AD592J, con salida en corriente, cuya FT es:

$$I[K] = I[\mu A]$$

Esta señal se supone sin error. Usted debe:

1. Seleccionar el conversor A/D adecuado. Si usa el del ARM, dibuje el esquema. Si no lo hace, especifique la referencia.
2. Diseñar, por software o hardware el método de transformación de [K] a [°F]. Dibuje el circuito o escriba el pseudocódigo del programa.
3. Dibuje detalladamente la interface del ARM con el display. NO se debe multiplexar, pues las lámparas son incandescentes. USE LATCHES. Haga el esquema de cada uno de los cuatro dígitos y su control y dibuje detalladamente el control de un dígito y la interface de una barra.
4. Escriba en Assembler del ARM una rutina para multiplicar por 9/5, usando aritmética entera de 16[bits] de datos. Suponga la existencia de una subrutina “div5” que pasando por los primeros cuatro bytes del Stack el valor del dividendo, devuelve del mismo modo el valor del resultado.

220[Vca]-0-0-0-0-0-0-0-0-0-0-gnd

fig.1: 12 lámparas halógenas conectadas en serie. Corriente: 1[A].

Resolución

Datos:

$$T_{Fmín}=14[^{\circ}F] \rightarrow T_{Cmín}=-10[^{\circ}C].$$

$$T_{Fmáx}=122[^{\circ}F] \rightarrow T_{Cmáx}=50[^{\circ}C].$$

$$\text{resolución}=0,1[^{\circ}F].$$

$$\text{Transductor: AD592} \rightarrow FT=1[\mu\text{A/K}].$$

Cálculos:

$$T_F = T_C \cdot \frac{9}{5} + 32[^{\circ}C] = 1,8 \cdot T_C + 32[^{\circ}C] \quad (1)$$

Pero:

$$T_C = T_K - 273[^{\circ}C] \quad (2)$$

Reemplazando (2) en (1), y operando:

$$\begin{aligned} T_F &= 1,8 \cdot T_C + 32[^{\circ}C] = 1,8 \cdot (T_K - 273[^{\circ}C]) + 32[^{\circ}C] = \\ &= 1,8 \cdot T_K - 491,4[^{\circ}C] + 32[^{\circ}C] \Rightarrow \\ &\Rightarrow T_F = 1,8 \cdot T_K - 459,4[^{\circ}C] \end{aligned} \quad (3)$$

Además, de (3) también se puede despejar:

$$\begin{aligned} T_F &= 1,8 \cdot T_K - 459,4[^{\circ}C] \Rightarrow T_K = \frac{T_F + 459,4[^{\circ}C]}{1,8} \Rightarrow \\ &\Rightarrow T_K = \frac{T_F}{1,8} + 255,22[^{\circ}C] \end{aligned} \quad (4)$$

Ahora, aplicando la fórmula final de (4) para “ $T_{Fmín}$ ” y “ $T_{Fmáx}$ ”, se obtiene:

$$T_{Kmín} = \frac{T_{Fmín}}{1,8} + 255,22[^{\circ}C] = \frac{14[^{\circ}F]}{1,8} + 255,22[^{\circ}C] \Rightarrow T_{Kmín} = 263[K] \quad (5)$$

$$T_{Kmáx} = \frac{T_{Fmáx}}{1,8} + 255,22[^{\circ}C] = \frac{122[^{\circ}F]}{1,8} + 255,22[^{\circ}C] \Rightarrow T_{Kmáx} = 323[K] \quad (6)$$

1. Se harán los siguientes cálculos para seleccionar el conversor A/D a utilizar:

$$cuentas = \frac{T_{Fmáx} - T_{Fmín}}{resolución} = \frac{122[^{\circ}F] - 14[^{\circ}F]}{0,1[^{\circ}F]} \Rightarrow cuentas = 1080[cuentas] \quad (7)$$

$$n_{A/D} = \frac{\ln(cuentas)}{\ln(2)} = \frac{\ln(1080[cuentas])}{\ln(2)} \Rightarrow n_{A/D} = 10,077[bits] \quad (8)$$

Por lo tanto se utilizará un **conversor A/D de 12[bits]**, y debido a que el conversor A/D del microcontrolador es de 10[bits], no se podrá utilizar, por lo tanto se seleccionará un conversor A/D externo. La tensión de referencia será:

$$V_{ref} = 10[V] \quad (9)$$

Por lo tanto:

$$V_{LSB} = \frac{V_{ref}}{2^{n_{A/D}}} = \frac{10[V]}{2^{12}} \Rightarrow V_{LSB} = 2,441[mV] \quad (10)$$

2. El circuito típico del AD592 es el siguiente:

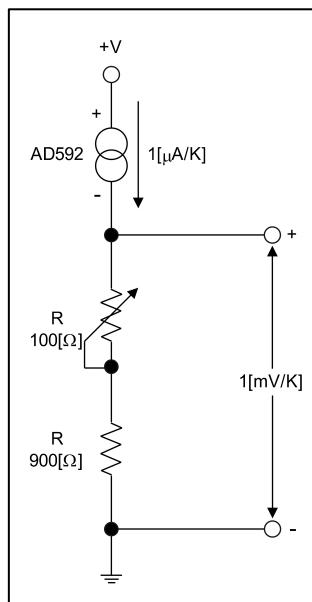


Figura 1: -Final 24.11.11- Configuración típica del transductor AD592

A continuación aplicando (4) se transforma la resolución de $[^{\circ}F]$ a $[K]$:

$$resolución_{[K]} = \frac{resolución_{[^{\circ}F]}}{1,8} = \frac{0,1[^{\circ}F]}{1,8} \Rightarrow resolución_{[K]} = 55,55[mK] \quad (11)$$

En este caso no se incluye la constante “+255,22 $[^{\circ}C]$ ” de (4), ya que la misma es simplemente un factor de corrimiento de escala para la medición de temperatura.

Por lo tanto, la tensión por cada paso de medición del AD592 “ V_{paso} ” es:

$$1[K] \text{ _____ } 1[mV] \\ 55,55[mK] \text{ _____ } x \Rightarrow \frac{55,55[mK].1[mV]}{1[k]} \Rightarrow x = V_{paso} = 55,556[\mu V] \quad (12)$$

Ahora sí es posible calcular la ganancia “G” a utilizar:

$$G = \frac{V_{LSB}}{V_{paso}} = \frac{2,442[mV]}{55,556[\mu V]} \Rightarrow G = 43,938[\text{veces}] \quad (13)$$

Por lo tanto, se utilizará un AI para dicha ganancia. Por otro lado, ya que la escala se inicia en $T_{Fmín}=14[^{\circ}\text{F}]$, se hará también una etapa de desplazamiento de nivel para llevar este valor como inicio de la escala del conversor A/D y aprovechar todo su rango posible por medio de una tensión de offset “ V_{off} ”. El circuito de lo descripto anteriormente, es el siguiente:

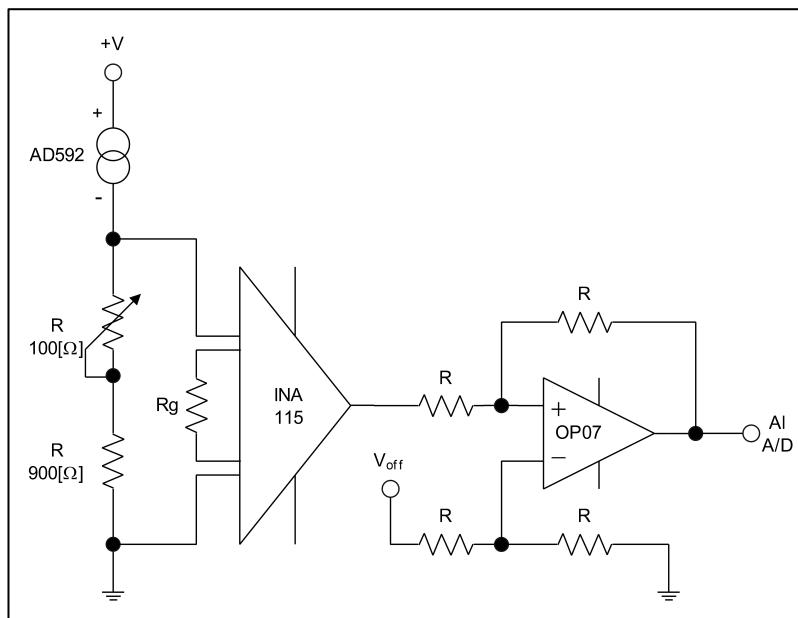


Figura 2: -Final 24.11.11- Circuito de adaptación completa

Los valores a averiguar, se realizan de la siguiente manera:

$$R = 10[k\Omega] \quad (14)$$

$$G = 1 + \frac{50[k\Omega]}{R_g} \Rightarrow R_g = \frac{50[k\Omega]}{(G-1)} = \frac{50[k\Omega]}{(43,938[\text{veces}]-1)} \Rightarrow \\ \Rightarrow R_g = 1,164[k\Omega]$$

$$V_{off} = (T_{Kmín}.resolución_{[K]})G = (263[K].1[mV]).43,938[\text{veces}] \Rightarrow \\ \Rightarrow V_{off} = 11,556[V]$$

Cabe recordar que “ V_{off} ”, al ingresar a la entrada inversa del OP07, se resta a la señal de entrada amplificada, llevando el nivel de $T_{Fmín}=14[^{\circ}F]$ al inicio de la escala de conversión del conversor A/D.

3. El circuito detallado y solicitado, es el siguiente:

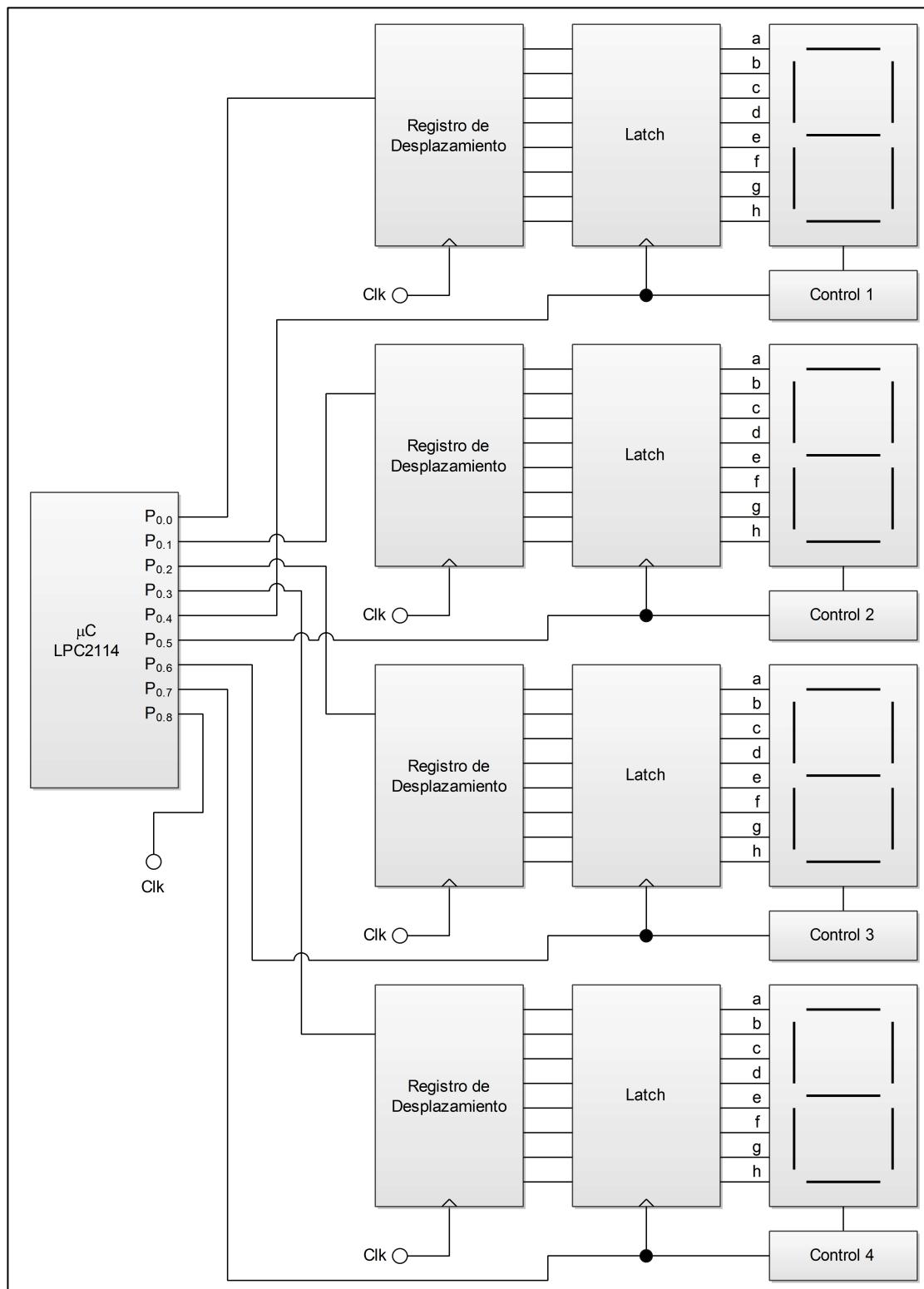


Figura 3: -Final 24.11.11- Circuito detallado del ejercicio

A continuación se encuentra un circuito de los 4 circuitos de control (Control 1, Control 2, Control 3 y Control 4), ya que los 4 son similares. Junto con éste, también se detalla el funcionamiento interno de un display, el cuál es similar para los 4:

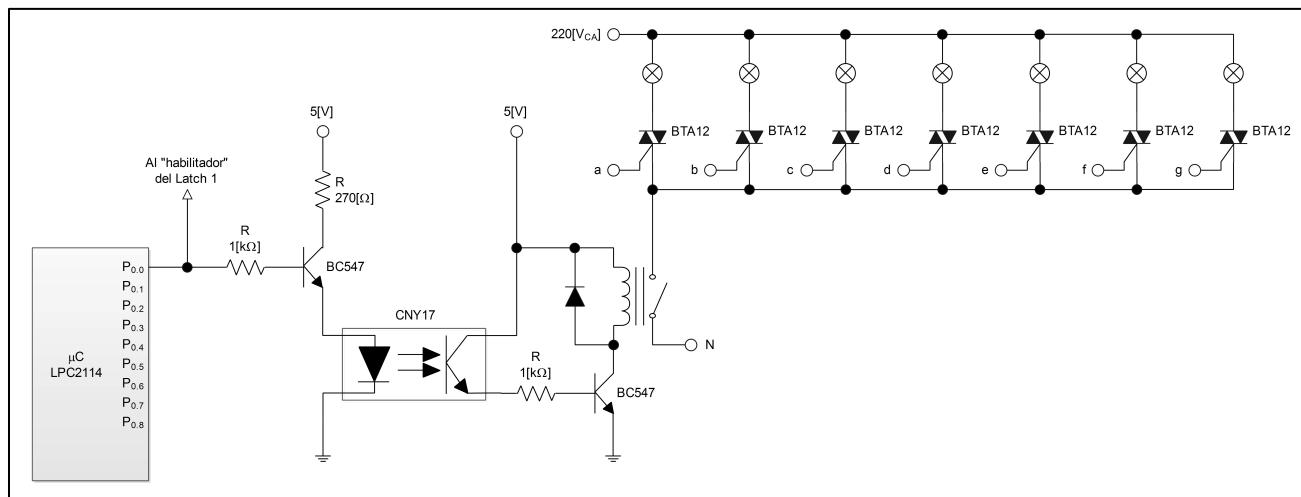


Figura 4: -Final 24.11.11- Circuito detallado de control del ejercicio

El “Clock” de los registros es una salida del microcontrolador.

4. El programa solicitado es el siguiente:

reset:

```
pop {r2}
```

mult:

```
mov r1,r2,lsl #3 //r1=r2.23=8.r2
add r1,r2,r1 //r1=r2+8.r2=9.r2
```

```
mov r3,r1 //“r1” dividendo
mov r4,#5 //“r4” divisor
```

div:

```
cmp r3,r4 //r3-r4
bls div2 //si es menor o igual
sub r3,r3,r4
add r5,r5,#1
b div
```

div2:

```
mov r6,#RESULTADO
mov r7,#RESTO
push {r6}
```

loop: b loop

*/Constantes

DATO: .half

RESULTADO: .half

RESTO: .half

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 15 de diciembre de 2011.

Usando un procesador basado en arquitectura ARM, elabore este problema:

Un silo de granos se ha instrumentado con sensores AD592C. Se usan 16 de estos sensores, con un multiplexor adecuado. La función de transferencia del mismo es de $1[\mu\text{A}/\text{K}]$, se requiere una resolución de $0,1[^\circ\text{C}]$ en el intervalo $10 - 60[^\circ\text{C}]$. Ud. debe:

1. Diseñe el hardware necesario para que se muestree esa señal a una tasa de $0,5[\text{Hz}]$, operado por interrupción. Esto significa que se mide un sensor cada $\frac{1}{2}[\text{s}]$. Debe incluir reloj, multiplexor, el controlador de interrupción del ARM y debe detallar la configuración del mismo.
2. Diseñe el hardware para acoplar el transductor al conversor elegido. Recuerde que se mide en $[^\circ\text{C}]$ y el transductor en $[\text{K}]$.
3. Establezca las tolerancias de los componentes pasivos de su circuito para un error de $0,1[\%]$.
4. Cada vez que complete el multiplexado de las señales (cada $8[\text{s}]$), 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 de eso. Escriba en Assembler del ARM la rutina para calcular el promedio.
5. Si la diferencia de cualquiera de las medidas con el promedio supera los $2[^\circ\text{C}]$ se debe activar un ventilador accionando un contactor externo cuya bobina opera con $220[\text{V}_{\text{CA}}]$ y corriente $0,2[\text{A}]$. diseñe la interface adecuada.

Resolución

Datos:

$$T_{Cmín}=10[°C].$$

$$T_{Cmáx}=60[°C].$$

$$\text{resolución}=0,1[°C].$$

$$\text{Transductor: AD592} \rightarrow FT=1[\mu\text{A/K}].$$

$$\text{Cantidad de trasductores: } 16[\text{un}].$$

1. Datos:

$$f_m=0,5[\text{Hz}].$$

Controlado por interrupción.

$$T_{entre_sens}=0,5[\text{s}].$$

Cálculos:

$$f_m = \frac{1}{T_{entre_sens}} = \frac{1}{0,5[\text{s}]} \Rightarrow f_m = 2[\text{Hz}] \quad (1)$$

Por lo tanto, el tiempo total “ t_T ” utilizado para la medición total será:

$$t_T = \text{cant_transductores}.T_{entre_sens} = 16[\text{un}].0,5[\text{s}] \Rightarrow t_T = 8[\text{s}] \quad (2)$$

El circuito principal del problema, es el siguiente:

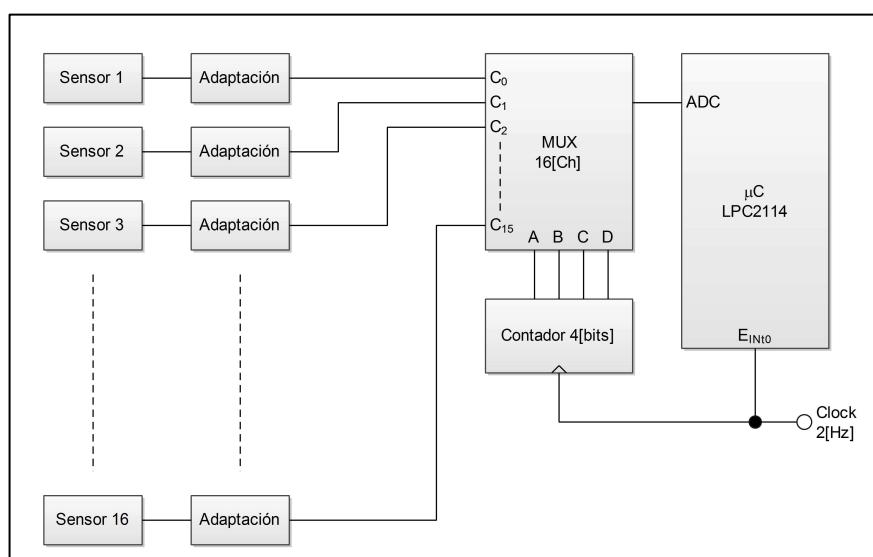


Figura 1: -Final 15.12.11- Circuito general del problema planteado

Y la temporización se llevará a cabo por medio de un LM555 configurado como “astable”, tal como lo muestra la siguiente figura:

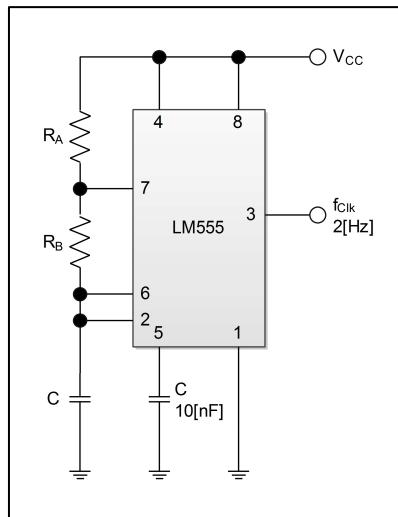


Figura 2: -Final 15.12.11- LM555 configurado “astable” para generar la frecuencia de clock necesaria

Los valores de los resistores y el capacitor, se calculan de la siguiente forma:

$$f_{Clk} = \frac{1,44}{(R_A + 2.R_B).C} = 2[Hz] \Rightarrow \frac{1,44}{2[Hz].C} = R_A + 2.R_B \quad (3)$$

Si se selecciona:

$$C = 1[\mu F] \quad (4)$$

Reemplazando este valor en (3):

$$R_A + 2.R_B = \frac{1,44}{2[Hz].C} = \frac{1,44}{2[Hz].1[\mu F]} \Rightarrow R_A + 2.R_B = 720[k\Omega] \quad (5)$$

Por lo tanto, se puede seleccionar el siguiente juego de valores para los resistores:

$$\begin{aligned} R_A + 2.R_B &= 720[k\Omega] \Rightarrow R_A = 60[k\Omega] \\ &\Rightarrow R_B = 330[k\Omega] \end{aligned} \quad (6)$$

VIC: El controlador vectorizado de interrupciones posee 32 entradas de interrupción en 3 categorías: FIQ, IRQ vectorizadas e IRQ no vectorizadas. La fuente de interrupción “E_{INT0}” es la 14. Cuando se activa, el VIC le pasa al microcontrolador la dirección donde tiene que ir para responder a la interrupción.

2. El circuito de conexión del sensor AD592C es el siguiente:

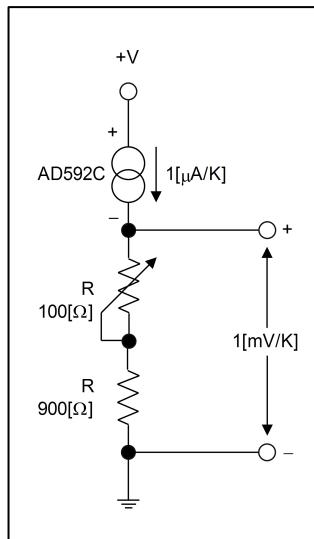


Figura 3: -Final 15.12.11- Circuito típico de conexión del sensor AD592C

Respecto del cálculo del conversor A/D, se muestran los siguientes cálculos:

$$\text{cuentas} = \frac{T_{\text{Cmáx}} - T_{\text{Cmín}}}{\text{resolución}} = \frac{60[\text{°C}] - 10[\text{°C}]}{0,1[\text{°C}]} \Rightarrow \text{cuentas} = 500[\text{cuentas}] \quad (7)$$

$$n_{\text{A/D}} = \frac{\ln(\text{cuentas})}{\ln(2)} = \frac{\ln(500[\text{cuentas}])}{\ln(2)} \Rightarrow n_{\text{A/D}} = 8,966[\text{bits}] \quad (8)$$

Por lo tanto se utilizará un **conversor A/D de 10[bits]**, y como el microcontrolador posee un conversor A/D de estas características, se utilizará éste. Su tensión de referencia es **V_{ref}=3[V]**.

$$V_{\text{LSB}} = \frac{V_{\text{ref}}}{2^{n_{\text{A/D}}}} = \frac{3[\text{V}]}{2^{10}} \Rightarrow V_{\text{LSB}} = 2,93[\text{mV}] \quad (9)$$

$$V_{\text{paso}} = \text{resolución}.FT_{\text{AD592C_V}} = 0,1[\text{°C}].1\left[\frac{\text{mV}}{\text{K}}\right] \Rightarrow V_{\text{paso}} = 100[\mu\text{V}] \quad (10)$$

$$G = \frac{V_{\text{LSB}}}{V_{\text{paso}}} = \frac{2,93[\text{mV}]}{100[\mu\text{V}]} \Rightarrow G = 29,3[\text{veces}] \quad (11)$$

Por lo tanto, el circuito adaptador se compone de la siguiente forma:

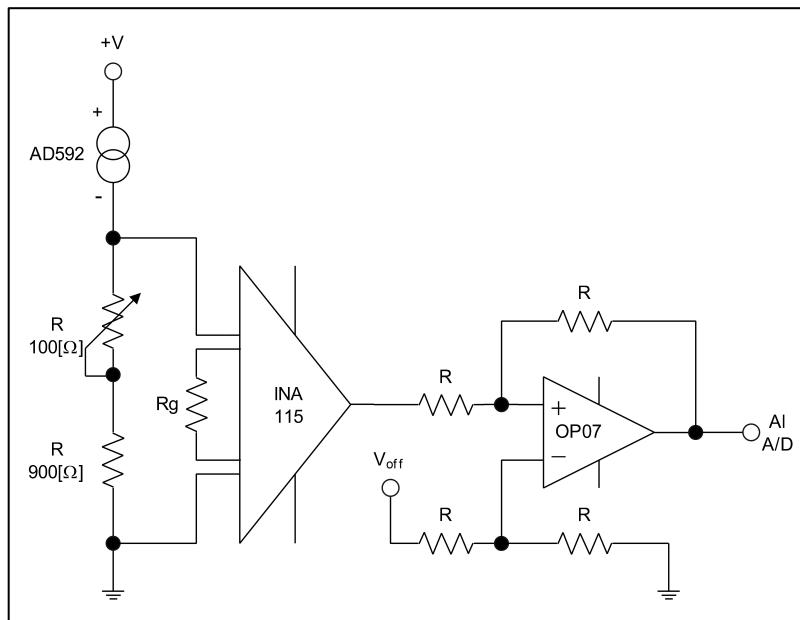


Figura 4: -Final 15.12.11- Circuito de adaptación

Donde, los valores a calcular son los siguientes:

$$R = 10[k\Omega] \quad (12)$$

$$G = 1 + \frac{50[k\Omega]}{R_g} \Rightarrow R_g = \frac{50[k\Omega]}{(G-1)} = \frac{50[k\Omega]}{(29,3[veces]-1)} \Rightarrow R_g = 1,767[k\Omega] \quad (13)$$

$$\begin{aligned} T_{[^\circ C]} &= T_{[K]} - 273,4[^\circ C] \Rightarrow T_{[K]} = 273,4[^\circ C] + T_{[^\circ C]} \Rightarrow \\ &\Rightarrow T_{Kmín} = 273,4[^\circ C] + T_{Cmín} = 273,4[^\circ C] + 10[^\circ C] \Rightarrow T_{Kmín} = 283,4[K] \end{aligned} \quad (14)$$

$$V_{off} = FT_{AD592C} \cdot T_{Kmín} \cdot G = 1 \left[\frac{mV}{K} \right] \cdot 283,4[^\circ C] \cdot 29,3[veces] \Rightarrow V_{off} = 8,304[V] \quad (15)$$

3. Teniendo en cuenta un error total “ ε ” de 0,1[%]:

$$\varepsilon = \varepsilon_{A/D} + \varepsilon_G + \varepsilon_{V_{ref}} = 0,1[%].V_{ref} = 0,001.3[V] \Rightarrow \varepsilon = 3[mV] \quad (16)$$

$$\varepsilon_{A/D} = \frac{V_{LSB}}{2} = \frac{2,93[mV]}{2} \Rightarrow \varepsilon_{A/D} = 1,465[mV] \quad (17)$$

Reemplazando (17) en (16) y operando:

$$\varepsilon_G = \varepsilon_{V_{ref}} = \frac{\varepsilon - \varepsilon_{A/D}}{2} = \frac{3[mV] - 1,465[mV]}{2} \Rightarrow \varepsilon_G = 767,5[\mu V] \quad (18)$$

$$\text{tolerancia} = \frac{\varepsilon_G}{V_{ref}} = \frac{767,5[\mu V]}{3[V]} \Rightarrow \text{tolerancia} = 255,833[ppm] \quad (19)$$

Por lo tanto se tomará una **tolerancia=250[ppm]**.

4. El diagrama de flujo inicial es el siguiente:

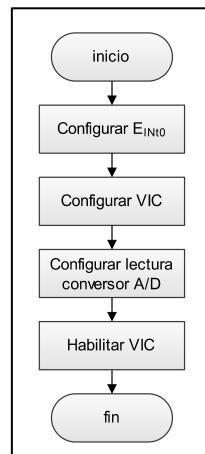


Figura 5: -Final 15.12.11- Diagrama de flujo de la configuración inicial

A continuación se muestra el diagrama de flujo de las rutinas:

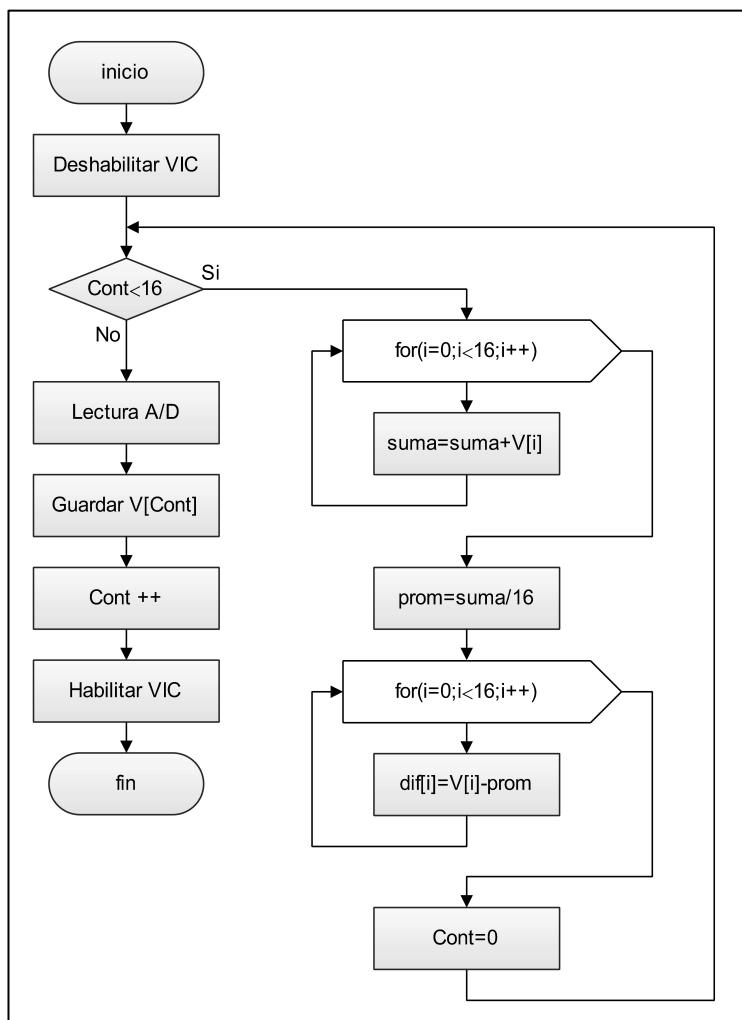


Figura 6: -Final 15.12.11- Diagrama de flujo de los cálculos solicitados

El programa en Assembler es el siguiente:

reset:

```
mov r0,#VECT
mov r1,#0          //suma
mov r2,#0          //promedio
mov r3,#16         //contador
mov r6,#PROM
```

prom:

```
ldr r4,[r0],#4
add r1,r1,r4
subs r3,r3,#1
bne prom        //mientras sea ≠0
mov r2,r1,lsr #4 //se divide por  $2^4=16$ 
str r2,[r5]      //se guarda “promedio” en “r5”
```

loop: b loop

*/Constantes

VECT: .word

PROM: .word

5. La interface solicitada es la siguiente:

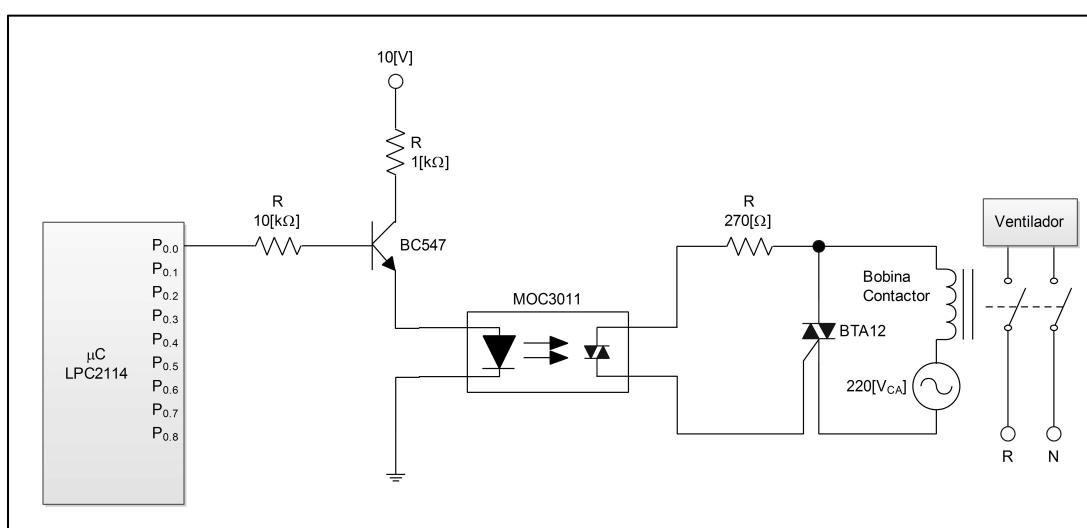


Figura 7: -Final 15.12.11- Circuito de la interface solicitada

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 22 de diciembre de 2011.

Se debe diseñar una balanza para pesar ganado con error total de 0,1[%]. Esto implica el uso de cuatro celdas de carga de 1500[kg] de capacidad cada una, conectada cada una en la esquina de una jaula cuadrada. Se necesita una resolución de 1[kg] en cada celda. La sensibilidad de las celdas es de 5[μ V/kg]. el valor máximo a pesar es el de cuatro animales juntos, estimado en 4000[kg] a los que hay que sumar la jaula que los contiene y que pesa 600[kg]. Se pide:

1. Asuma que la referencia de su conversor es de 10,0[V]. diseñe el circuito de adaptación de señal para usar esa referencia, estableciendo como condición que CADA ESCALÓN de la salida del conversor DEBE ser equivalente a 1[kg].
2. Se debe diseñar un mecanismo de ajuste para compensar el peso de la jaula. La indicación de cero se da por medio de una tecla (cero). Usted puede usar hardware para compensar el peso de la jaula INCLUYÉNDOLO EN EL DISEÑO DEL PUNTO 1., o bien, por algún mecanismo de software, generar un offset por programa para hacer lo propio. La señal de la tecla (cero) se obtiene de un circuito como el de la fig. 1. Diseñe la interface adecuada desde "A" hasta el micro y el diagrama de programa.

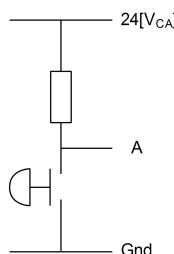


Figura 1. Circuito de los pulsadores

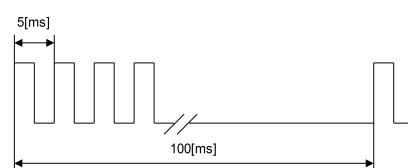


Figura 2. Diagrama de tiempos

3. Con una tecla igual a la de la fig. 1 marcada (pesar) se inicia un ciclo de pesada. Ud. debe diseñar el multiplexor para las señales de las celdas de carga, con el reloj adecuado para ello. Cada 100[ms] se debe hacer un ciclo de lecturas de las cuatro celdas, que deben leerse a intervalos de 5[ms] entre sí (fig. 2). Debe hacerse un total de ocho (8) ciclos de lectura de las cuatro celdas para cada pesada. Ud. debe diseñar el circuito que hace la temporización y hacer el diagrama del programa que hace la lectura.
4. Asuma que los valores leídos de cada celda se han guardado en cuatro bloques de memoria en forma consecutiva. Escriba en Assembler la rutina que tome los valores leídos de cada una de las cuatro celdas, y los sume para encontrar el peso de los animales.

Resolución

Datos:

Cant. de celdas: $Cant_{celdas} = 4[\text{un}]$.

Error total: $\varepsilon = 0,1\%[$ %].

Capac. Máxima por celda: $Capac_{máx_celda} = 1500[\text{kg}]$

$resolución_{celda} = 1[\text{kg}]$.

$sensibilidad_{celda} = 5[\mu\text{V}/\text{kg}]$.

$peso_{máx_animal} = 4000[\text{kg}]$.

$peso_{jaula} = 600[\text{kg}]$.

1. Datos:

$V_{ref} = 10,0[\text{V}]$.

$paso = 1[\text{kg}]$.

Cálculos:

Como cada escalón debe medir $1[\text{kg}]$, se puede utilizar directamente la $sensibilidad_{celda}$ para los cálculos. Los cálculos son:

$$\begin{aligned} cuentas &= \frac{peso_{máx_animal} + peso_{jaula}}{cant_{celdas} \cdot resolución_{celda}} = \frac{4000[\text{kg}] + 600[\text{kg}]}{4[\text{un}] \cdot 1[\text{kg}]} \Rightarrow \\ &\Rightarrow cuentas = 1150[cuentas] \end{aligned} \quad (1)$$

$$n_{A/D} = \frac{\ln(cuentas)}{\ln(2)} = \frac{\ln(1150[cuentas])}{\ln(2)} \Rightarrow n_{A/D} = 10,167[bits] \quad (2)$$

Por lo tanto se utilizará un **conversor A/D de 12[bits]**.

$$V_{LSB} = \frac{V_{ref}}{2^{n_{A/D}}} = \frac{10,0[\text{V}]}{2^{12}} \Rightarrow V_{LSB} = 2,441[mV] \quad (3)$$

$$V_{paso} = paso \cdot sensibilidad_{celda} = 1[\text{kg}] \cdot 5 \left[\frac{\mu\text{V}}{\text{kg}} \right] \Rightarrow V_{paso} = 5[\mu\text{V}] \quad (4)$$

$$G = \frac{V_{LSB}}{V_{paso}} = \frac{2,441[mV]}{5[\mu\text{V}]} \Rightarrow G = 488,2[veces] \quad (5)$$

Por lo tanto se utilizará un AI como el de la siguiente figura:

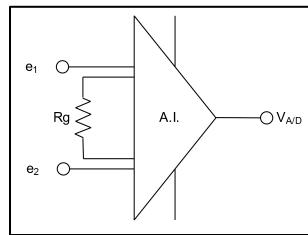


Figura 1: -Final 22.12.11- AI (INA115) utilizado para la ganancia

Donde:

$$G = 1 + \frac{50[k\Omega]}{R_g} \Rightarrow R_g = \frac{50[k\Omega]}{(G-1)} = \frac{50[k\Omega]}{(488,2[veces]-1)} \Rightarrow R_g = 102,627[\Omega] \quad (6)$$

El circuito completo con todos los componentes, es el siguiente:

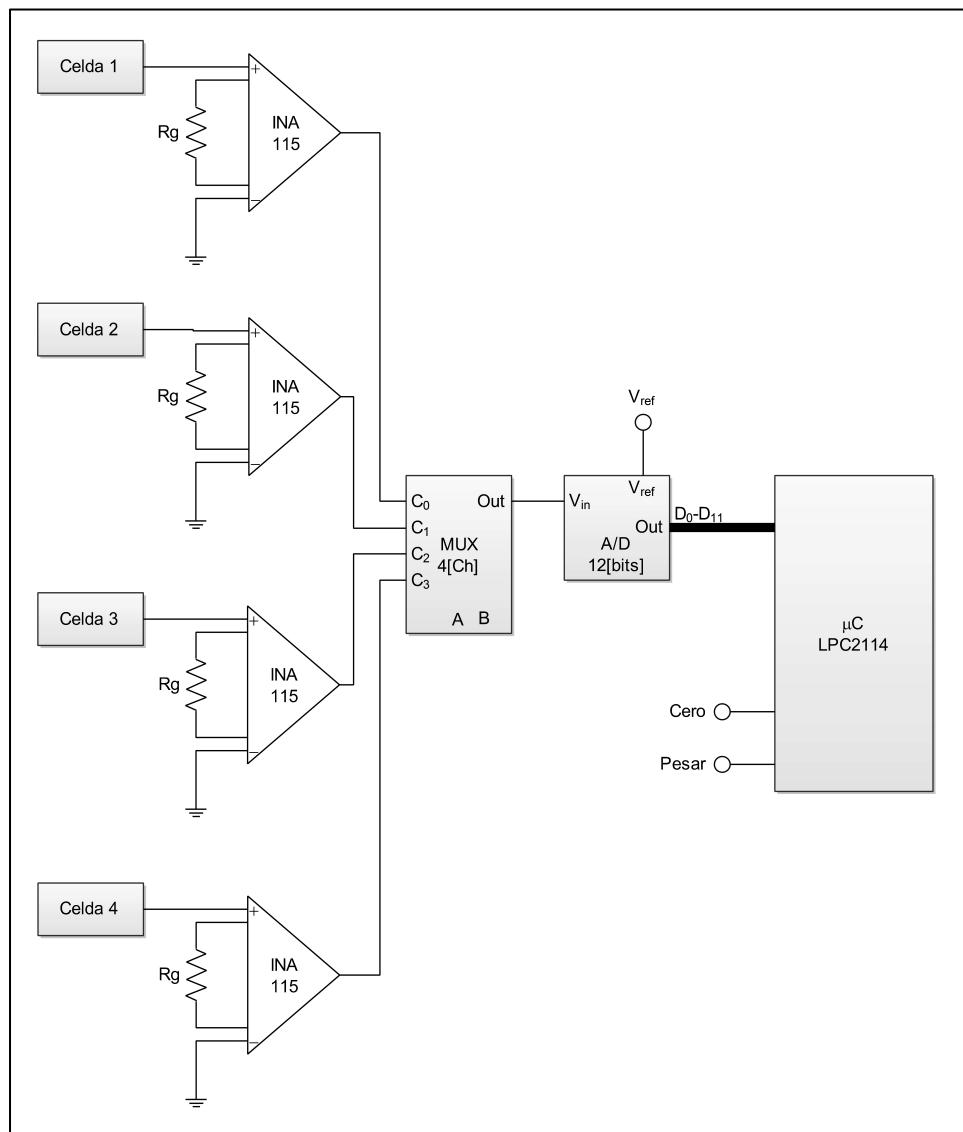


Figura 2: -Final 22.12.11- Circuito completo a utilizar

2. A continuación se muestra el circuito para comandar el pulsador “cero”:

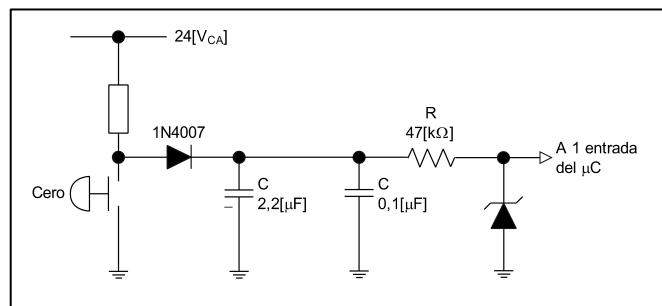


Figura 3: -Final 22.12.11- Circuito utilizado para el pulsador “cero”

Como la resolución es de 1[kg], por lo tanto se puede hacer que el “cero” compensado de la jaula sea la resta de 600[cuentas], por lo tanto, el diagrama de flujo para el “cero” es el siguiente:

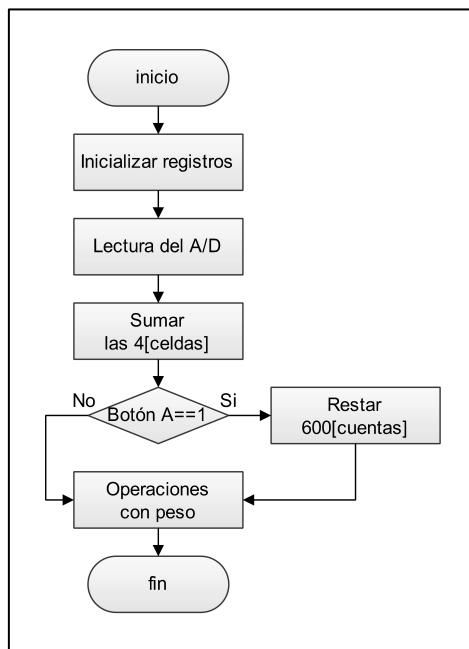


Figura 4: -Final 22.12.11- Diagrama de flujo utilizado para el pulsador “cero”

3. El diagrama de flujo solicitado es el siguiente:

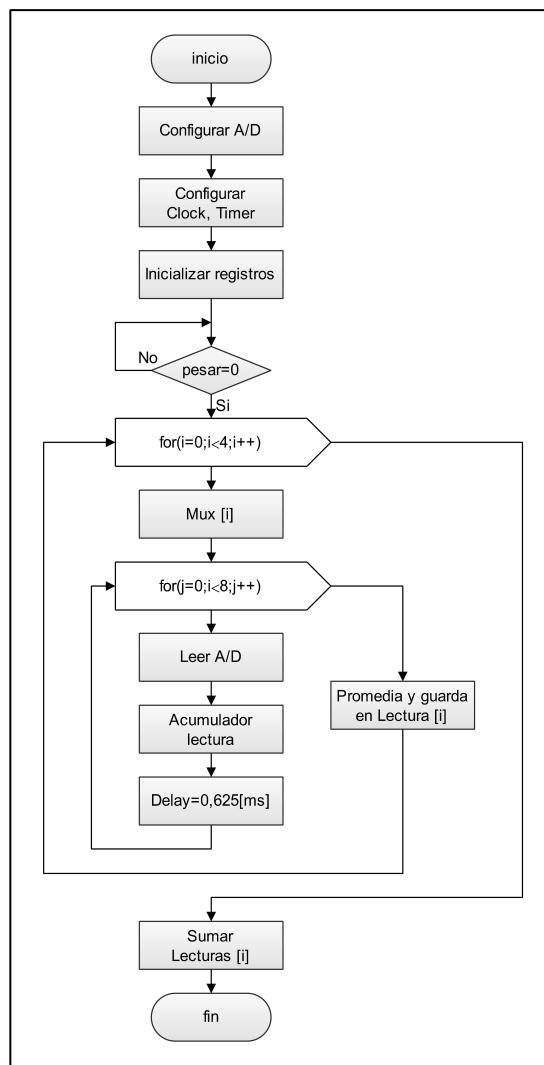


Figura 5: -Final 22.12.11- Diagrama de flujo utilizado para el pulsador "pesar"

4. El programa en Assembler utilizado para leer el peso final de los animales, es el siguiente:

reset:

```

mov r0,#lectura
mov r1,#4
mov r3,#0
  
```

lectura:

```

ldr r2,[r0],#4          //Lee el valor y luego lo incrementa
add r3,r2,r3
subs r1,r1,#1
bne lectura             //Mientras sea ≠0
mov r4,#peso
str r3,[r4]              //Guarda en "peso"
  
```

loop: b loop

*/Constantes

lectura: .word

.word

.word

.word

peso: .word

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 24 de mayo de 2012.

1. Enviar información a un registro de desplazamiento mediante un puerto, debe transmitir en serie una cadena de 8[bits], se pide:
 - a. Diseñar circuito asociado al desplazamiento, mostrar los 2 primeros bits de transferencia, establezca todas las señales correspondientes (enable, clock, etc).
 - b. Diseñar la interfaz del registro con el microcontrolador, usando un clock de 1[kHz]. Diseñar el diagrama de flujo del programa que controla la operación.
2. Un transductor genera información con una función de transferencia de $2,5[\mu\text{A}/\text{bit}]$. Realice la interface del mismo hacia un conversor A/D de 12[bits], con referencia 10[V].
 - a. Establezca las tolerancias de los componentes pasivos de su circuito para un error de 0,1[%].
 - b. Diseñe el hardware para que se muestre esa señal a una tasa de 10[Hz] operado por interrupción. Debe incluir el reloj, el controlador de interrupciones que use y debe detallar la configuración del mismo.

Resolución

Cálculos:

- El circuito es el siguiente:

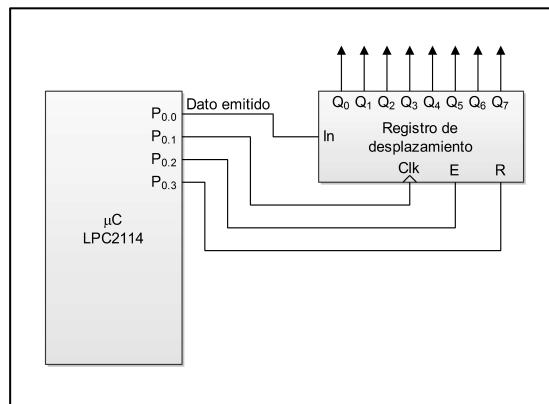


Figura 1: -Final 24.05.12- Circuito solicitado

A continuación se muestra la interfaz:

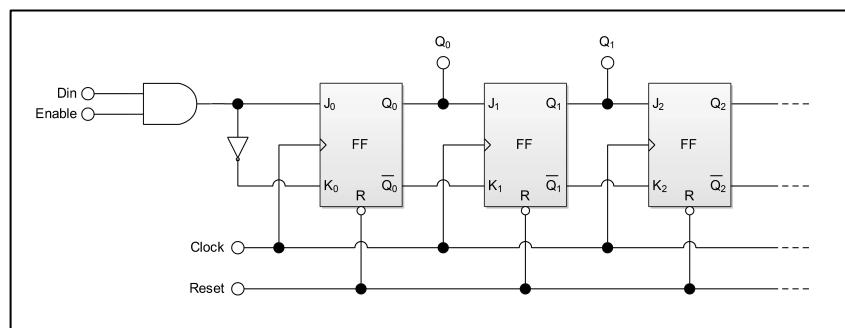


Figura 2: -Final 24.05.12- Circuito asociado de un registro de desplazamiento, sólo para los 2 primeros bits de transferencia

A continuación se muestra su diagrama de tiempos:

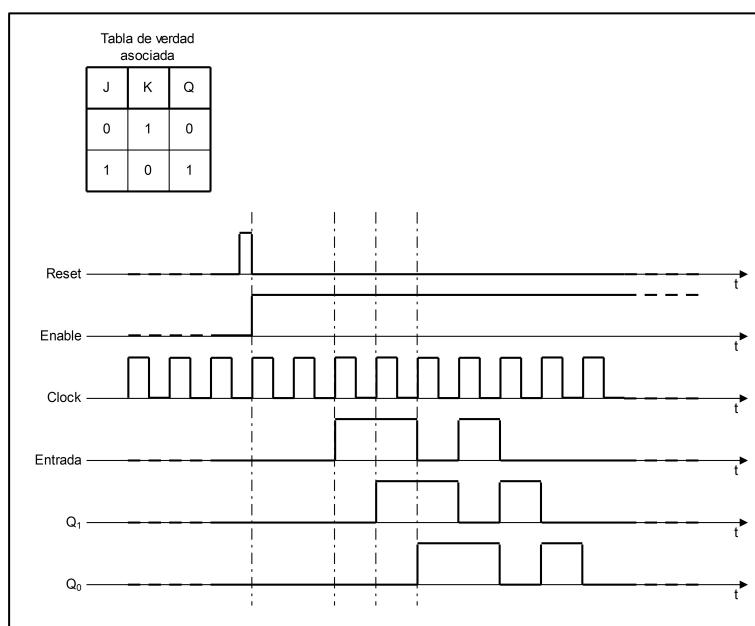


Figura 3: -Final 24.05.12- Tabla de verdad y diagrama de tiempos del registro de desplazamiento, para los dos primeros bits

Y por último, el diagrama de flujo de control de la operación a realizar, para 1 solo dato:

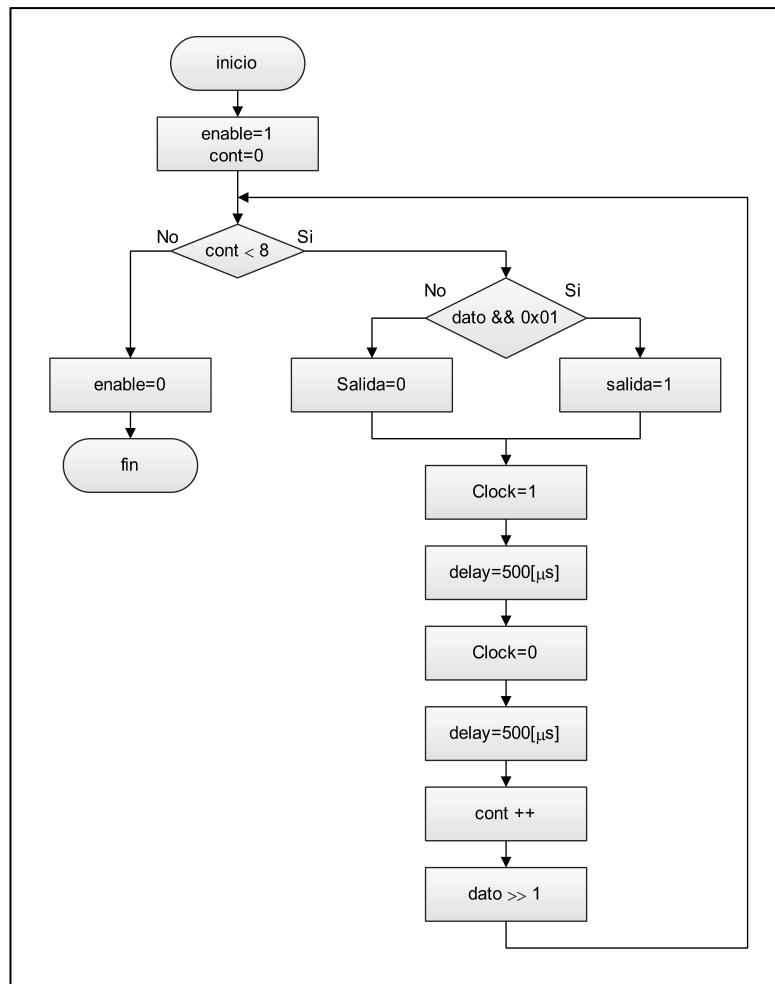


Figura 4: -Final 24.05.12- Diagrama de flujo correspondiente a 1 solo dato

2. Datos:

$$FT=2,5[\mu\text{A}/\text{bit}]$$

$$n_{A/D}=12[\text{bits}]$$

$$V_{ref}=10,0[\text{V}]$$

Cálculos:

A continuación se muestra el circuito utilizado para la interface del mismo al conversor A/D:

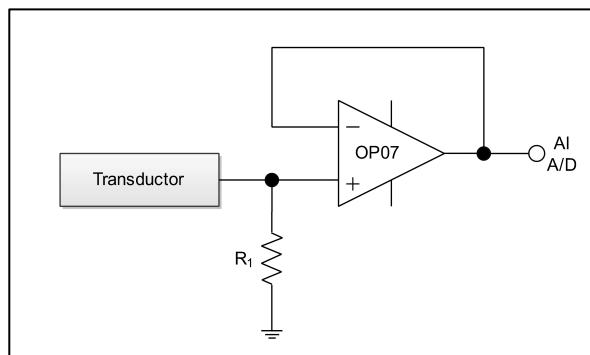


Figura 5: -Final 24.05.12- Circuito de adaptación del transductor utilizado

Para dicho circuito, se deben hacer los siguientes cálculos:

$$V_{LSB} = \frac{V_{ref}}{2^{n_{A/D}}} = \frac{10,0[\text{V}]}{2^{12}} \Rightarrow V_{LSB} = 2,441[\text{mV}] \quad (1)$$

$$R_1 = \frac{V_{LSB}}{I_{FT}} = \frac{2,441[\text{mV}]}{2,5[\mu\text{A}]} \Rightarrow R_1 = 976,563[\Omega] \quad (2)$$

a. El error total es 0,1[%], por lo tanto:

$$\begin{aligned} \varepsilon &= \varepsilon_{A/D} + \varepsilon_G + \varepsilon_{V_{ref}} = 0,1[\%] = 0,001 \cdot V_{ref} = 0,001 \cdot 10,0[\text{V}] \Rightarrow \\ &\Rightarrow \varepsilon = 10[\text{mV}] \end{aligned} \quad (3)$$

$$\varepsilon_{A/D} = \frac{V_{LSB}}{2} = \frac{2,441[\text{mV}]}{2} \Rightarrow \varepsilon_{A/D} = 1,221[\text{mV}] \quad (4)$$

Reemplazando (4) en (3) y operando:

$$\varepsilon_G = \varepsilon_{V_{ref}} = \frac{\varepsilon - \varepsilon_{A/D}}{2} = \frac{10[\text{mV}] - 1,221[\text{mV}]}{2} \Rightarrow \varepsilon_G = 4,39[\text{mV}] \quad (5)$$

$$\text{tolerancia} = \frac{\varepsilon_G}{V_{ref}} = \frac{4,39[\text{mV}]}{10,0[\text{V}]} \Rightarrow \text{tolerancia} = 438,95[\text{ppm}] \quad (6)$$

Por lo tanto se tomará una **tolerancia=250[ppm]**.

b. El circuito solicitado se muestra a continuación:

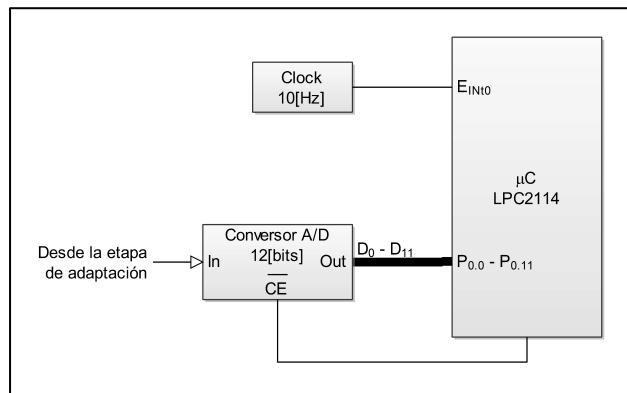


Figura 6: -Final 24.05.12- Diagrama general de la conexión del conversor A/D con el μ C

Y el circuito generador de la frecuencia solicitada es:

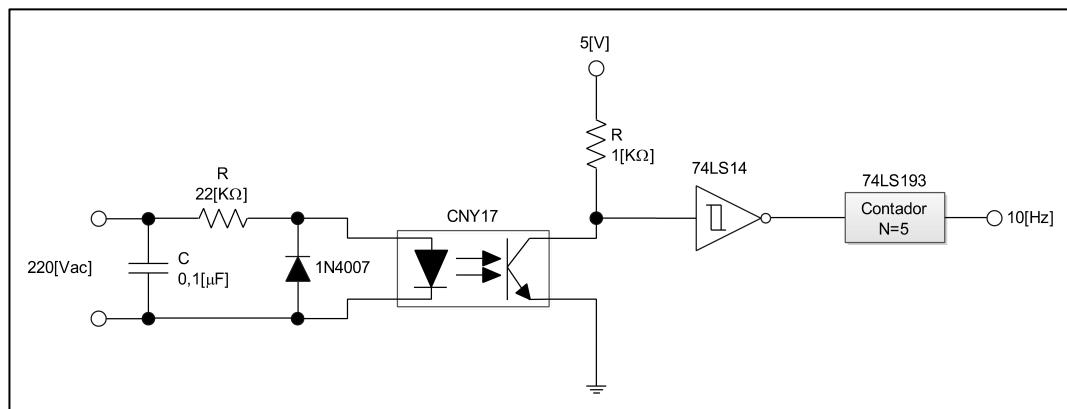


Figura 7: -Final 24.05.12- Circuito generador de la frecuencia de 10[Hz]

Y el diagrama de flujo correspondiente es el siguiente:

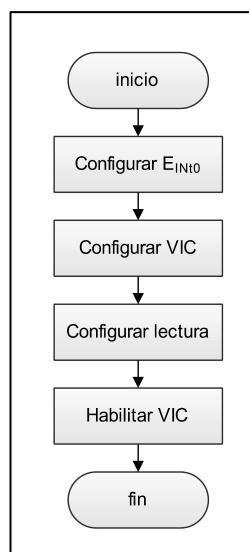


Figura 8: -Final 24.05.12- Diagrama de flujo para la configuración de las interrupciones

La fuente de interrupción "E_{INT0}" es la 14.

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 05 de julio de 2012.

1. Describa una secuencia de instrucciones que recorra un bloque de 100[bytes] de memoria. Este programa debe contar todos los números pares y los números negativos. La dirección “PAR” de tamaño byte debe almacenar el número de pares, y la dirección “NEG” la cantidad de números negativos.
2. ¿Cuál es la diferencia entre NEG y MVN?
3. Realizar una secuencia que ponga a cero los 3[bits] más a la izquierda en “R1”, sin cambiar “R1” y los guarde en “R2”.
4. Describir e indicar los nombres que conforma la RS-232-C completa.
5. Describa los siguientes términos: Simplex, Half Duplex, Duplex.
6. Si el 8250 tiene conectado un reloj que permite una velocidad máxima de 76800[baudios], ¿cuál es la frecuencia del reloj? ¿Qué valor debe programarse en el registro divisor para que trasmite a 9600[baudios]?
7. Sistema de adquisición de 8 canales de señales que varían entre 0,2[V] y 4[V], con un rango de frecuencia comprendido entre 1 y 10[MHz], con un A/D de 8[bits] de aproximaciones sucesivas ¿cuál es el tiempo máximo de conversión del A/D? ¿Cuál es la mínima frecuencia interna del A/D?
8. En un conversor A/D flash ¿cuántos comparadores se necesitan?
9. ¿Cuál es el tiempo de conversión del A/D doble rampa? Expresarlo en función de la frecuencia del reloj que utiliza.
10. Se requiere ante la presencia de un “1” en el bit “0” del puerto 305H que se saque por el bit “7” del puerto 304H un tren de pulsos que esté 0,2[s] en alto y 0,4[s] en bajo. Se dispone de una rutina llamada “delay” que produce un retardo de 5[ms]. El procedimiento que hay que describir mediante un diagrama de flujo debe monitoriar la entrada para cumplir con lo demandado y permanecer en esta tarea mientras el “1” esté presente en esa entrada. Cuando esta situación finalice provocara el fin del procedimiento.

Resolución

Cálculos:

1. Datos:

Bloque=100[bytes].

Cálculos:

El diagrama de flujo del ejercicio, es el siguiente:

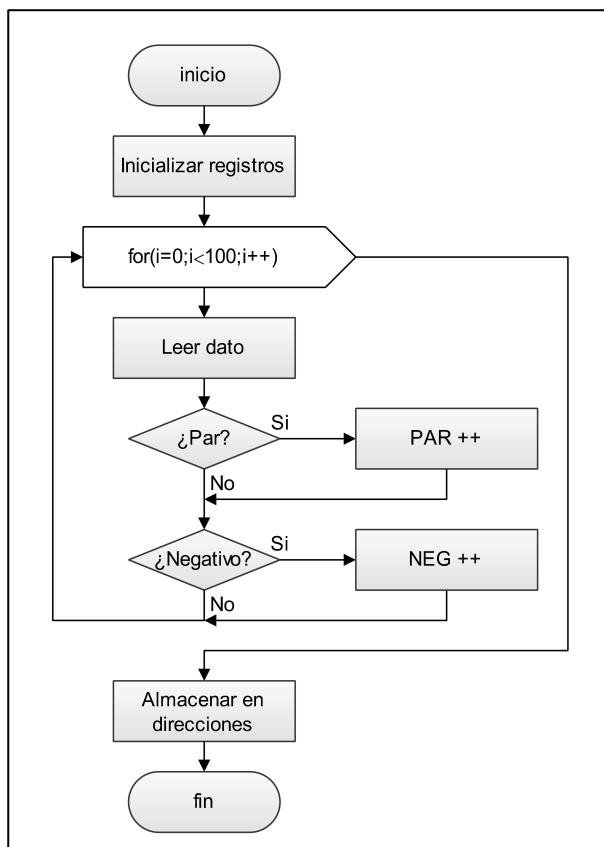


Figura 1: -Final 05.07.12- Diagrama de flujo para el programa solicitado

La codificación en Assembler es la siguiente:

reset:

```

mov  r1,#0      //contador par
mov  r2,#0      //contador negativos
mov  r3,#100    //longitud del vector a recorrer
mov  r4,#0x200  //guarda posición
  
```

lectura:

```

ldrb r5,[r4],#1
and  r6,r5,#1   //“and” lógica con “1”
cmp  r6,#0      //“r6=0”
  
```

```

addeq r1,r1,#1      //”PAR ++” (Z=1)
cmp   r5,#0
addmi r2,r2,#1      //”NEG ++” (N=1)
add   r4,r4,#1      //avanza 1 posición de memoria
subs  r3,r3,#1      //decrementa el contador en 1
bne   lectura        // (Z=0)
mov   r5,#PAR         //carga la 1ra posición del vector “PAR”
strb  r1,[r5]          //carga “r1” donde está apuntando “r5”
mov   r6,#NEG         //carga la 1ra posición del vector “NEG”
strb  r2,[r6]          //carga “r2” donde está apuntando “r6”

```

loop: b loop

*/Constantes

PAR: .byte
NEG: .byte

2. NEG: negar, negativo del valor que tiene el registro. Complemento a 2.

Sintaxis: NEG {Cond.} {s} Rd, Op2 → modifica todas las banderas.

Ejemplo: neg r2,r1 //multiplica “r1” por (-1) y lo guarda en r2 (0-r1).

MVN: mover negado, complementa el valor. Se usa para formar máscaras.

Sintaxis: MVN {Cond.} {s} Rd, Op2 → modifica solo “N” y “Z”.

Ejemplo: mvn r3,r4 //genera el complemento a 1 de “r4” y lo guarda en “r3”
//(NOT r4).

3. mvn r3,#0 //r3=11111.....111
mov r3,r3,lsl #3 //r3=00011....111
and r2,r1,r3 //r2=r1 AND r3

4. GND (Common Ground): debe estar conectada a chasis y tierra.

TxD (Transmitted Data): datos enviados del DTE al DCE.

RxD (Received Data): datos enviados del DCE al DTE.

RTS (Request To Send): solicitud del DTE al DCE para que se prepare para recibir datos.

CTS (Clear To Send): Respuesta al RTS, para que el DTE inicie la transmisión.

DTR (Data Terminal Ready): Solicitud del DTE que está listo para conectarse.

DSR (Data Set Ready): Solicitud del DCE que está listo para recibir datos del DTE.

DCD (Data Carrier Detect): Señal del DCE cuando la conexión ha sido establecida con el equipamiento remoto.

RI (Ring Indicator): Señal del DCE que indica que ha recibido una llamada.

DTE (Data Terminal Equipment): por ejemplo, la PC.

DCE (Data Connection Equipment): por ejemplo, el módem.

5. Los términos son los modos de transmisión.

Simplex: transmisión unidireccional. Por ejemplo radiodifusión de TV y audio.

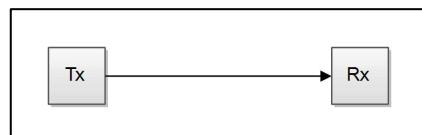


Figura 2: -Final 05.07.12- Modo de transmisión "Simplex"

Half duplex: permite transmitir en ambas direcciones, aunque la trasnmisión solo puede ocurrir en una dirección simultáneamente. Por ejemplo la banda de radio civil.

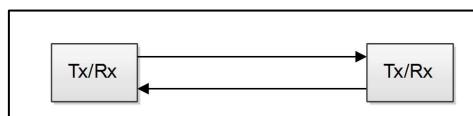


Figura 3: -Final 05.07.12- Modo de transmisión "Half duplex"

Full duplex: permite transmitir en ambas direcciones y simultáneamente por el mismo canal. Por ejemplo el teléfono.

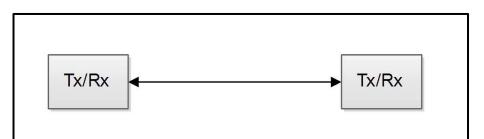


Figura 4: -Final 05.07.12- Modo de transmisión "Full duplex"

6. Resolviendo para el LPC2114:

$$\text{baud_rate} = \frac{\text{VPB}}{16.\text{DLR}} \quad (1)$$

Por lo tanto, para un "baud rate" mínimo:

$$\text{VPB}_{\min} = \text{baud_rate}.16 = 76800[\text{baudios}].16 \Rightarrow \text{VPB}_{\min} = 1,229[\text{MHz}] \quad (2)$$

Luego, de (1) se despeja el valor:

$$\begin{aligned} baud_rate &= \frac{VPB}{16.DLR} \Rightarrow DLR = \frac{VPB}{16.baud_rate} = \frac{1,229[MHz]}{16.9600[baudios]} \Rightarrow \\ &\Rightarrow DLR = 8[bits] \end{aligned} \quad (3)$$

7. Datos:

Cantidad de canales: $n_{canales}=8[\text{canales}]$.

$V_{S\min}=0,2[V]$.

$V_{S\max}=4,0[V]$.

$f_{S\min}=1[MHz]$.

$f_{S\max}=10[MHz]$.

$n_{A/D}=8[bits]$.

Tipo: aproximaciones sucesivas.

Cálculos:

$$\begin{aligned} f_m &= 5[\text{veces}].f_{S\max}.n_{canales} = 5[\text{veces}].10[MHz].8[\text{canales}] \Rightarrow \\ &\Rightarrow f_m = 400[MHz] \end{aligned} \quad (4)$$

$$t_{m_máx} = \frac{1}{f_m} = \frac{1}{400[MHz]} \Rightarrow t_{m_máx} = 2,5[nF] \quad (5)$$

$$\begin{aligned} t_{m_máx} &= (n_{A/D} + 1).t_{A/D} \Rightarrow t_{A/D} = \frac{t_{m_máx}}{(n_{A/D} + 1)} = \frac{2,5[nF]}{(8[bits] + 1)} \Rightarrow \\ &\Rightarrow t_{A/D} = 277,778[ps] \end{aligned} \quad (6)$$

$$f_{A/D_mín} = \frac{1}{t_{A/D}} = \frac{1}{277,778[ps]} \Rightarrow f_{A/D_mín} = 3,6[GHz] \quad (7)$$

8. Por ejemplo, para el caso del punto anterior, donde $n_{A/D}=8[bits]$, se necesitan:

$$\begin{aligned} comparadores &= 2^{n_{A/D}} - 1 = 2^8 - 1 \Rightarrow \\ &\Rightarrow comparadores = 255[comparadores] \end{aligned} \quad (8)$$

9. En un conversor A/D doble rampa, se tiene que:

$$t_m = T_1 + T_2 \quad (9)$$

Pero en el peor de los casos:

$$T_1 = T_2 = 2^{n_{A/D}} t_{Clk} \quad (10)$$

Por lo tanto, reemplazando (10) en (9) y operando:

$$t_{máx} = 2^{(n_{A/D}+1)} \cdot t_{Clk} \Rightarrow t_{máx} = \frac{2^{(n_{A/D}+1)}}{f_{Clk}} \quad (11)$$

10. Algunos cálculos preliminares:

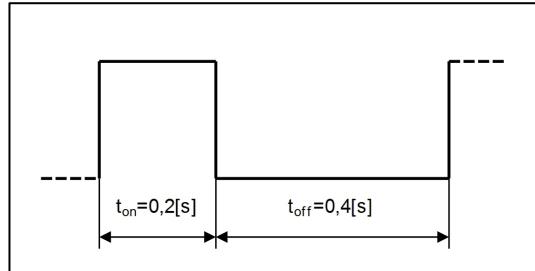


Figura 5: -Final 05.07.12- Representación gráfica de los tiempos en alto y en bajo

Datos:

$$t_{delay}=5[\text{ms}]$$

$$ciclos_{on} = \frac{t_{on}}{t_{delay}} = \frac{0,2[\text{s}]}{5[\text{ms}]} \Rightarrow ciclos_{on} = 40[\text{ciclos}] \quad (12)$$

$$ciclos_{off} = \frac{t_{off}}{t_{delay}} = \frac{0,4[\text{s}]}{5[\text{ms}]} \Rightarrow ciclos_{off} = 80[\text{ciclos}] \quad (13)$$

El diagrama de flujo correspondiente es el siguiente:

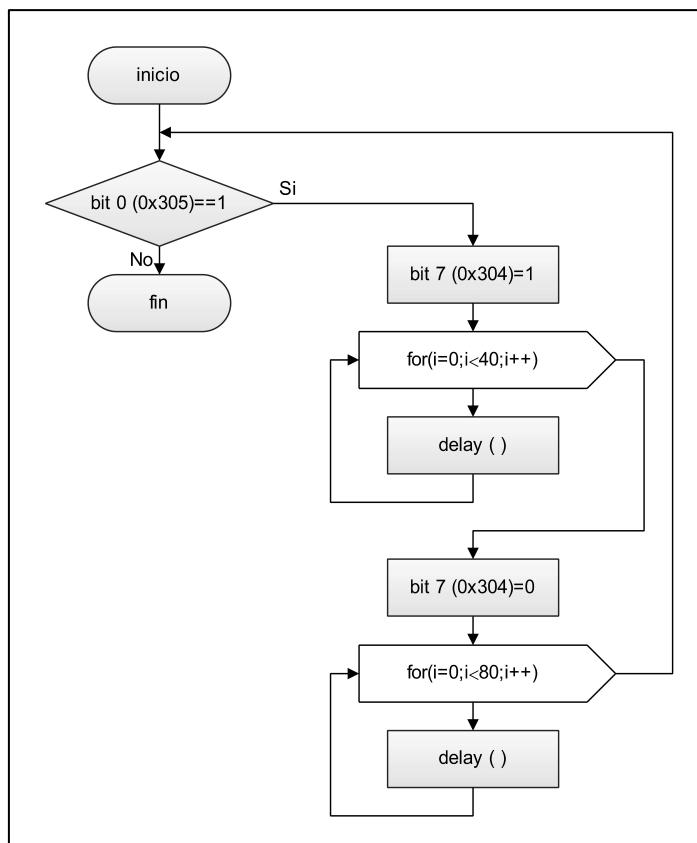


Figura 6: -Final 05.07.12- Diagrama de flujo solicitado

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 26 de julio de 2012.

1. Se tiene que recibir una señal RS-232-C operando a 600[bps]. Ud. debe pasar la información a un registro de desplazamiento. La trama tiene el formato 8N1. Ud. debe transferir la información recibida en serie a un registro de 8[bits]. Este registro es leído por el microcontrolador cada vez que se completa un carácter y es la única relación del procesador con el enlace serie.
 - a. Establezca el método de sincronización de la señal RS-232-C con sus componentes necesarios. Especifique la tolerancia (bajo norma) de la frecuencia de reloj y defina cuál ha de ser esta frecuencia para una adecuada sincronización. Considere que la señal de entrada ha de muestrearse lo más cerca posible de $T() / 2$.
 - b. Dibuje el circuito asociado al registro de desplazamiento. Dibuje asimismo un circuito detallado de los 2 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).
 - c. Diseñe la interface del registro y el microcontrolador. 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 fin del carácter y debe ejecutarse durante el bit de stop.
2. Un transductor genera información con una función de transferencia de $2,5[\mu\text{A}/\text{bit}]$. Realice la interface del mismo hacia un conversor A/D de 12[bits], con referencia $10,0[\text{V}]$.
 - a. Establezca las tolerancias de los componentes pasivos de su circuito para un error de $0,1[\%]$.
 - b. Diseñe el hardware para que se muestre esa señal a una tasa de $10[\text{Hz}]$ operando por interrupción. Debe incluir el reloj, el controlador de interrupción que usee y debe detallar la configuración del mismo.

Resolución

1. Datos:

Señal: RS-232-C.

Velocidad=600[bps].

Formato de trama: 8N1.

Registro: 8[bits].

a. Cálculos:

La trama 8N1 se forma de la siguiente forma:

- Sin paridad.
- 1[bit] de “Start”.
- 8[bits] de “Datos”.
- 1[bit] de “Stop”.

Gráficamente:

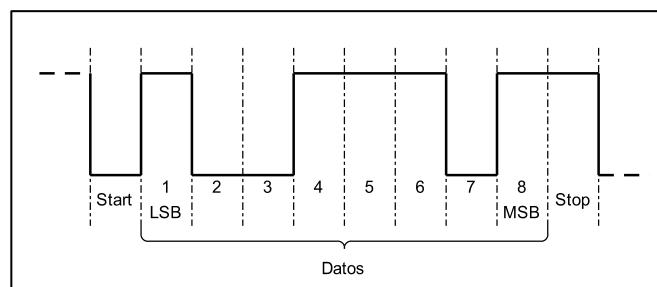


Figura 1: -Final 26.07.12- Formato de la trama 8N1

El circuito solicitado es el siguiente:

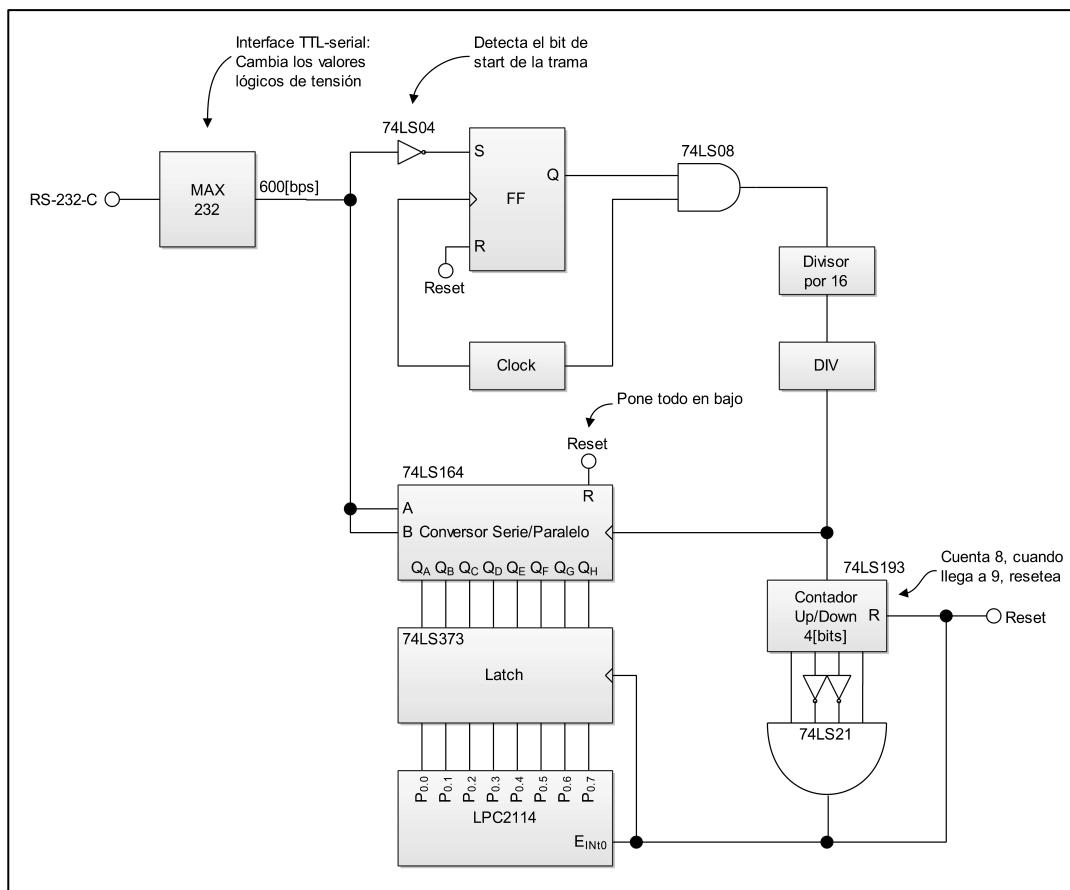


Figura 2: -Final 26.07.12- Circuito solicitado

Finalmente, el circuito generador del clock es el siguiente:

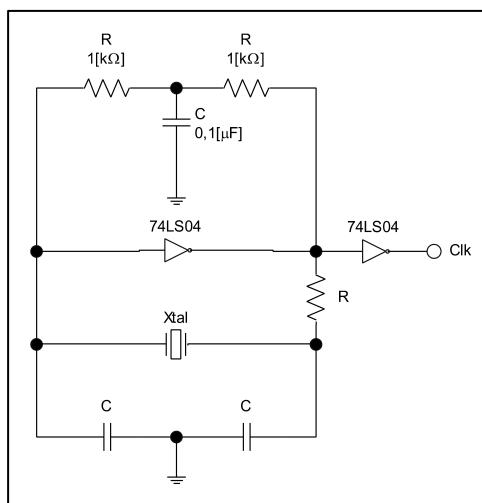


Figura 3: -Final 26.07.12- Circuito utilizado para la generación del clock

Se utiliza un clock de valor $16 \times 600[\text{bps}]$ para compensar el bloque del divisor.

- b. El circuito utilizado es el siguiente:

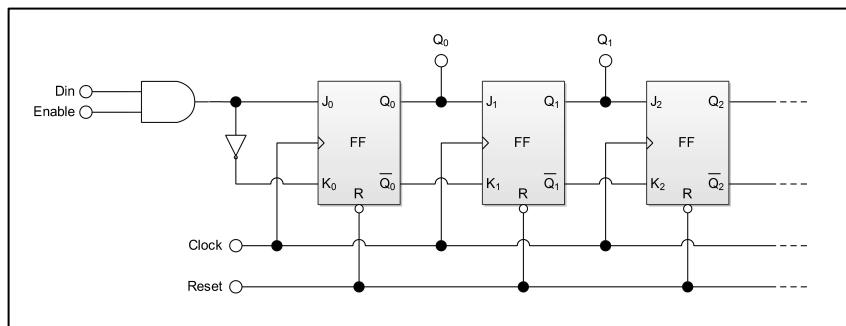


Figura 4: -Final 26.07.12- Circuito asociado de un registro de desplazamiento, sólo para los 2 primeros bits de transferencia

- c. Los diagramas de flujo son:

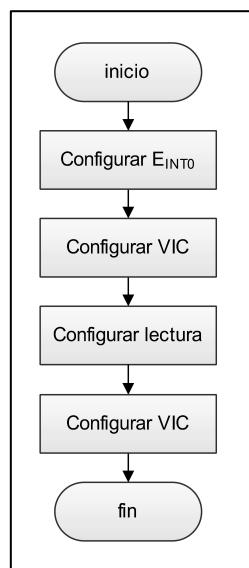


Figura 5: -Final 26.07.12- Diagrama de flujo correspondiente a la configuración general de las interrupciones

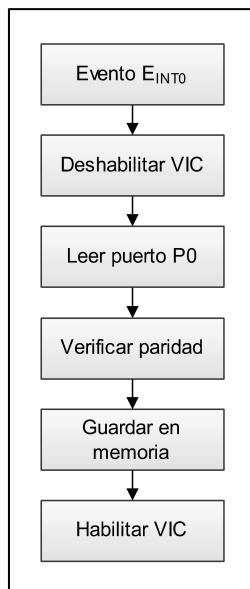


Figura 6: -Final 26.07.12- Diagrama de flujo correspondiente a la configuración del vector de interrupciones

Respecto del diagrama de flujo anterior, la fuente de interrupción “E_{INT0}”, la cual es la interrupción 14, y se ubica en el pin P0.1.

A continuación se muestra el diagrama de flujo correspondiente a la verificación de paridad “par”:

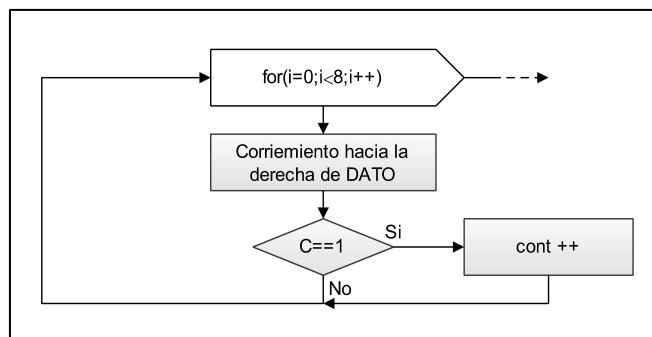


Figura 6: -Final 26.07.12- Diagrama de flujo correspondiente a la rutina para verificar la paridad (par)

La codificación en Assembler es la siguiente:

reset:

```

mov r0,#DATO      //se asigna a "r0" la etiqueta "DATO"
mov r1,#8          //se carga "r1" con el valor de 8, de 8[bites]
ldrb r3,[r0]        //se carga en "r3" el contenido de "r0"
  
```

paridad:

```

mov r3,r3,lsr #1
addcs r2,r2,#1    //C=1
subs r1,r1,#1
bne paridad       //salta si es ≠0
mov r2,r2,lsr #1
bcs control        //¿"r2" impar?
  
```

Control:

.....

*/Constantes

DATO: .byte

2. Datos:

$$FT_A = 2,5[\mu A/bit].$$

$$n_{A/D} = 12[bits].$$

$$V_{ref} = 10,0[V].$$

Cálculos:

La interface solicitada, tiene el siguiente circuito:

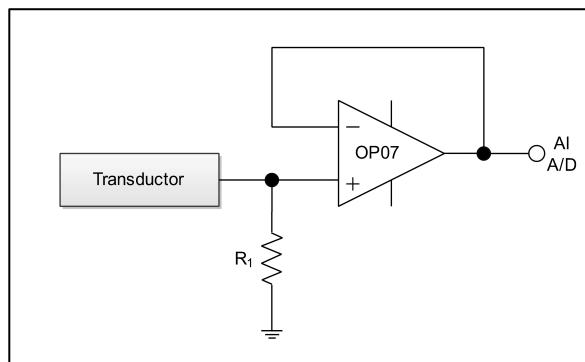


Figura 6: -Final 26.07.12- Circuito de la interface solicitada

De donde, para calcular el valor de “R₁”:

$$V_{LSB} = \frac{V_{ref}}{2^{n_{A/D}}} = \frac{10,0[V]}{2^{12}} \Rightarrow V_{LSB} = 2,441[mV] \quad (1)$$

$$R_1 = \frac{V_{LSB}}{I_{FT_A}} = \frac{2,441[mV]}{2,5[\mu A]} \Rightarrow R_1 = 976,4[\Omega] \quad (2)$$

a. Datos:

Error total: $\epsilon = 0,1[\%]$.

Cálculos:

$$\epsilon = \epsilon_{A/D} + \epsilon_G + \epsilon_{V_{ref}} = 0,1[\%]. V_{ref} = 0,001 \cdot 10,0[V] \Rightarrow \epsilon = 10[mV] \quad (3)$$

$$\epsilon_{A/D} = \frac{V_{LSB}}{2} = \frac{2,441[mV]}{2} \Rightarrow \epsilon_{A/D} = 1,221[mV] \quad (4)$$

$$\mathcal{E}_G = \mathcal{E}_{V_{ref}} = \frac{\mathcal{E} - \mathcal{E}_{A/D}}{2} = \frac{10[mV] - 1,221[mV]}{2} \Rightarrow \mathcal{E}_G = 4,39[mV] \quad (5)$$

$$\text{tolerancia} = \frac{\mathcal{E}_G}{V_{ref}} = \frac{4,39[mV]}{10,0[V]} \Rightarrow \text{tolerancia} = 439[\text{ppm}] \quad (6)$$

Por lo tanto se tomará una **tolerancia=250[ppm]**.

b. El circuito solicitado es el siguiente:

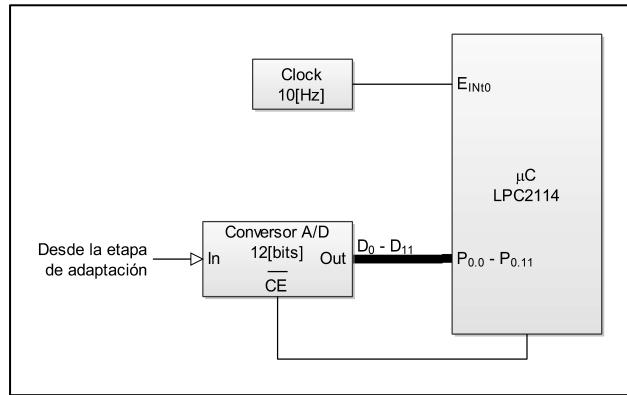


Figura 7: -Final 26.07.12- Diagrama general de la conexión del conversor A/D con el μC

El circuito generador de clock es el siguiente:

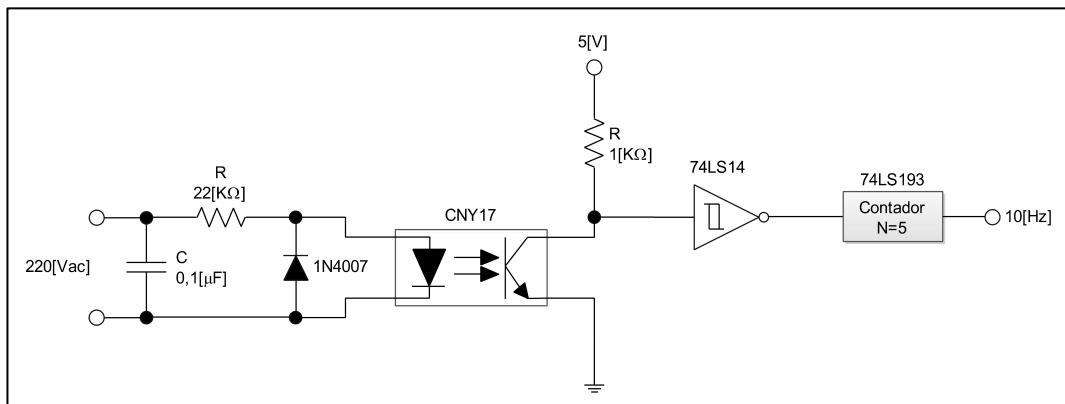


Figura 8: -Final 26.07.12- Circuito generador de la frecuencia de 10[Hz]

El diagrama de flujo correspondiente es:

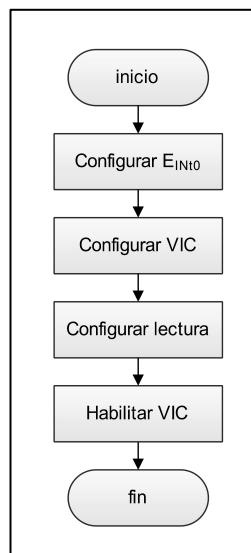


Figura 9: -Final 26.07.12- Diagrama de flujo para la configuración de las interrupciones

La fuente de interrupción “E_{INT0}” es la 14.

**Universidad Tecnológica Nacional
Facultad Regional Córdoba
Departamento de Ingeniería Electrónica**

Cátedra de Técnicas Digitales II.

Examen final: 20 de diciembre de 2012.

Se debe medir la salida de un transductor integrado de temperatura. El rango de medición es de 0 a 50[°C] (temperatura ambiente). Se pide una resolución de 0,1[°C]. La corriente de salida es de 100[μ A/°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].

1. 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.
2. Diseñe el circuito de reloj a utilizar para generar la salida de 1[Hz]. Explique cómo hace para usar esa salida de 1[Hz] para obtener la información de la hora.
3. 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 qué tipo de dato usará cada entrada de ese vector (byte, half word, word) para que contenga la información completa de la hora.
4. A partir de la salida de 1[Hz] de 2., explique cómo hace para que el sistema opere por interrupción para hacer una conexión del A/D. Especifique cuál entrada de interrupción usa del ARM usado en los TP.
5. Diseñe el servicio de interrupción para que lea 8 veces el A/D en cada toma de datos. Escriba un programa en Assembler del ARM para acumular las lecturas y hacer luego el promedio.

Resolución

Datos:

$$T_{\min} = 0[^{\circ}\text{C}]$$

$$T_{\max} = 50[^{\circ}\text{C}]$$

$$\text{resolución} = 0,1[^{\circ}\text{C}]$$

$$FT = 100[\mu\text{A}/^{\circ}\text{C}]$$

$$\text{Error: } \varepsilon \leq 0,1[\%]$$

$$\text{resolución}_{\text{hora}} = 1[\text{s}]$$

1. Datos:

Conversor A/D: propio del ARM $\rightarrow n_{\text{A/D}} = 10[\text{bits}]$.

$$V_{\text{ref}} = 3,0[\text{V}]$$

Cálculos:

El diagrama general para la conversión, es el siguiente:

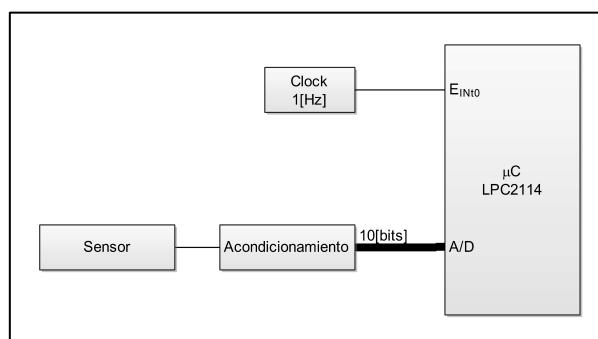


Figura 1: -Final 20.12.12- Diagrama general para la conversión

El circuito de acondicionamiento de señal, es el siguiente:

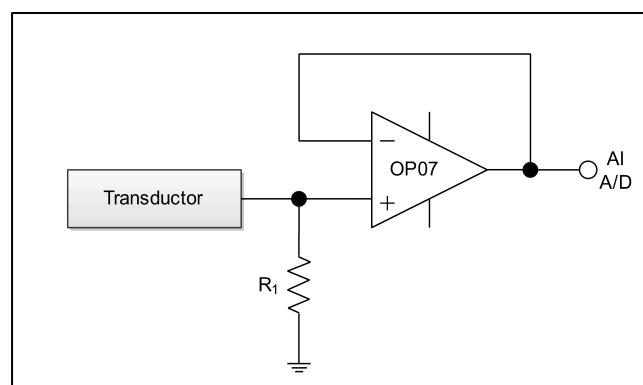


Figura 2: -Final 20.12.12- Circuito utilizado para el acondicionamiento

Del cual se hará el cálculo de “ R_1 ”:

$$R_1 = \frac{V_{LSB}}{I_{paso}} = \frac{V_{LSB}}{\text{resolución.FT}} \quad (1)$$

$$V_{LSB} = \frac{V_{ref}}{2^{n_{A/D}}} = \frac{3,0[V]}{2^{10}} \Rightarrow V_{LSB} = 2,93[mV] \quad (2)$$

Reemplazando (2) en (1):

$$R_1 = \frac{V_{LSB}}{I_{paso}} = \frac{2,93[mV]}{0,1[^{\circ}\text{C}].100[\mu\text{A}/^{\circ}\text{C}]} \Rightarrow R_1 = 292,969[\Omega] \quad (3)$$

Luego:

$$\epsilon = \epsilon_{A/D} + \epsilon_G + \epsilon_{V_{ref}} = 1[\%].V_{ref} = 0,01.3,0[V] \Rightarrow \epsilon = 30[mV] \quad (4)$$

$$\epsilon_{A/D} = \frac{V_{LSB}}{2} = \frac{2,93[mV]}{2} \Rightarrow \epsilon_{A/D} = 1,465[mV] \quad (5)$$

$$\epsilon_G = \epsilon_{V_{ref}} = \frac{\epsilon - \epsilon_{A/D}}{2} = \frac{30[mV] - 1,465[mV]}{2} \Rightarrow \epsilon_G = 14,268[mV] \quad (6)$$

$$\text{tolerancia} = \frac{\epsilon_G}{V_{ref}} = \frac{14,817[mV]}{3,0[V]} \Rightarrow \text{tolerancia} = 4756[ppm] \quad (7)$$

Por lo tanto se tomará una **tolerancia=2000[ppm]**.

2. Datos:

$$f_{Clk}=1[\text{Hz}]$$

Cálculos:

Para generar dicha frecuencia, se utiliza el siguiente circuito:

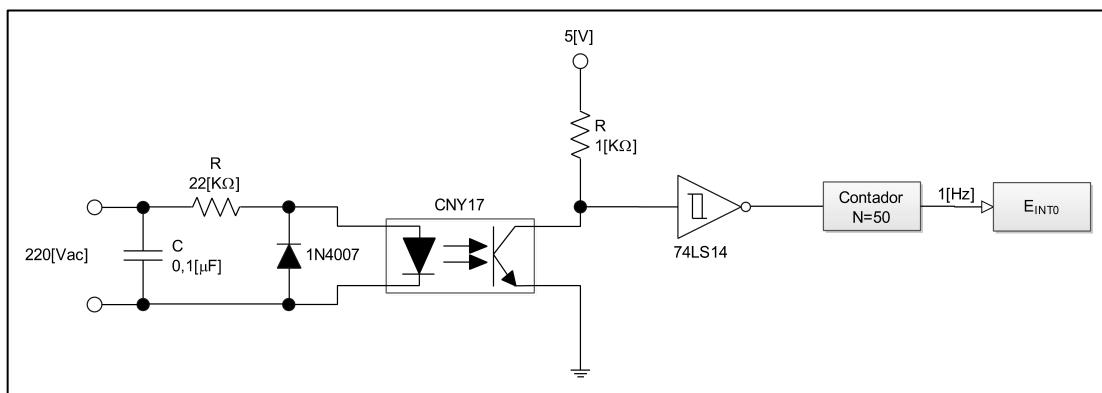


Figura 3: -Final 20.12.12- Circuito utilizado para la generación de clock (10[Hz])

3. En BCD, cada cifra se representa por 4[bits], por lo tanto, para guardar la información del reloj se tiene:

$$hh, mm, ss, = 6[cifras].4[bits] \Rightarrow hh, mm, ss, = 24[bits] \quad (8)$$

Por lo tanto se utilizará el **tipo de datos “word”**.

El esquema general para el conteo de la hora, será el siguiente:

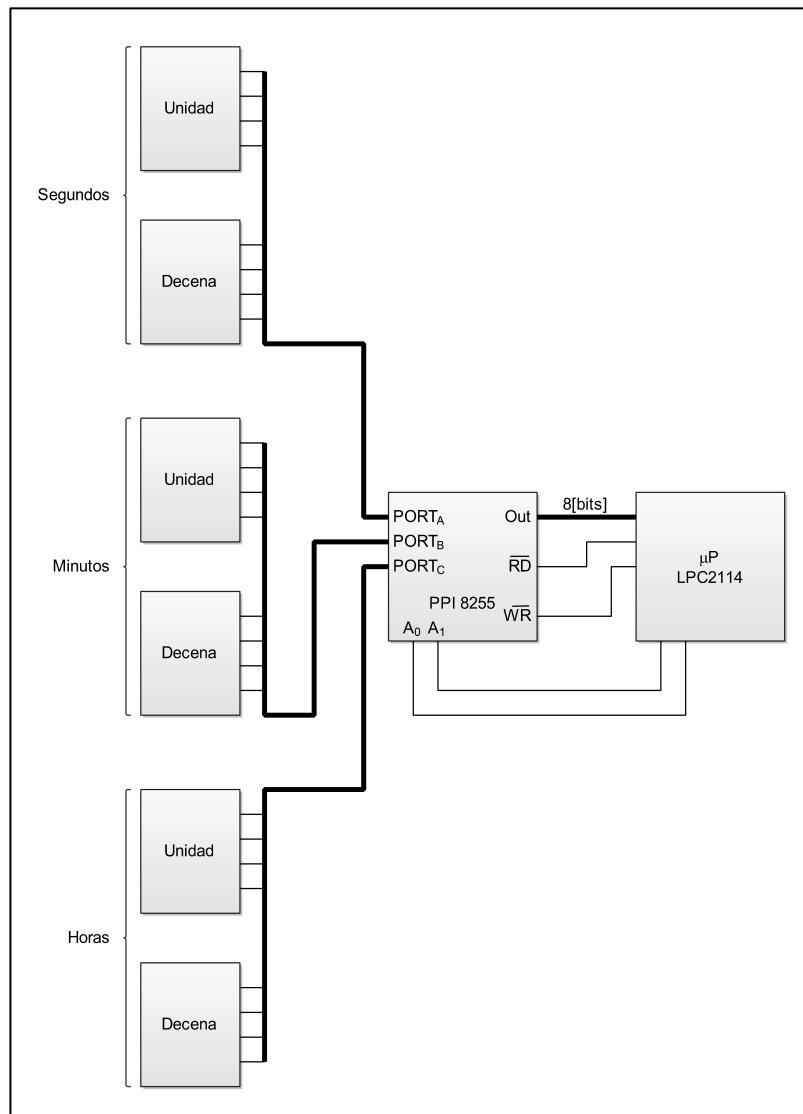


Figura 4: -Final 20.12.12- Esquema de conexión del circuito de cuenta de la hora

El PPI (Interfaz Periférica Programable) 8255A de Intel es un dispositivo programable de entrada/salida de propósito general. Tiene 40 pines, de los cuales 24 son de entrada/salida (3 puertos de 8[bits] cada uno). Tiene 3 modos de operación, en el modo “0”, cada uno de los 3 puertos tiene operaciones de entrada y salida, sin un protocolo para la transmisión.

Mediante “A₀” y “A₁” se selecciona el puerto (“A”, “B”, o “C”).

“RD” es para leer datos del 8255, “WR” es para enviar datos al mismo desde el μ P.

Con este arreglo se utilizarán 12[bits] para controlar los 24[bits] necesarios.

4. El diagrama de flujo correspondiente a la inicialización de las interrupciones, es el siguiente:

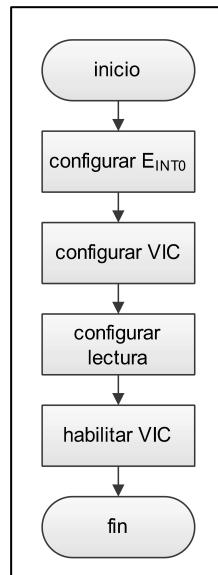


Figura 5: -Final 20.12.12- Diagrama de flujo de la inicialización de las interrupciones

A continuación se realiza el diagrama de flujo para “E_{INT0}”:

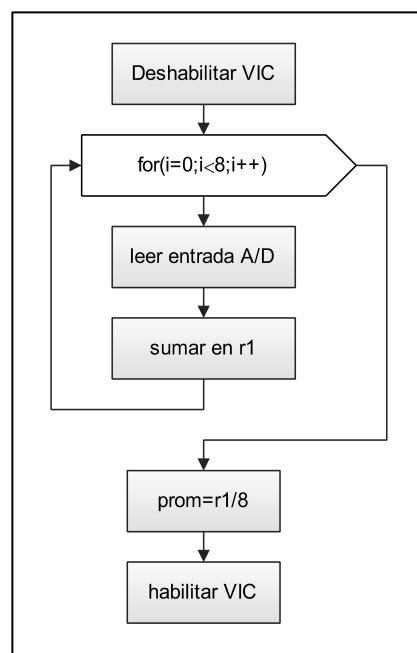


Figura 6: -Final 20.12.12- Diagrama de flujo para la rutina del punto 5

5. El programa solicitado es el siguiente:

reset:

```
    mov      r0,#ADC  
    mov      r1,#8  
    mov      r3,#0  
    mov      r4,#PROM
```

promedio:

```
    ldr      r2,[r0],#4      //el contenido de “r0+4[bytes]” lo guarda en “r2”  
    add      r3,r3,r2  
    subs     r1,r1,#1      //“r1” se decrementa en “1”  
    bne     promedio      //Z=0  
    mov      r3,r3,lsl #3  //divide por  $2^3=8$   
    str      r3,[r4]
```

loop: b loop

*/Constantes:

ADC: .word
 .word
 .word
 .word
 .word
 .word
 .word
 .word
 .word
PROM: .word