

ESCUELA POLITÉCNICA NACIONAL

**FACULTAD DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

**DISEÑO E IMPLEMENTACIÓN DE UN OSCILOSCOPIO
CON UN dsPIC**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO
EN ELECTRÓNICA Y CONTROL**

**Dibujes Salgado Cristian Marcelo
Torres Morales Edwin Alexis**

DIRECTOR: Dr. Luis Corrales

Quito, Febrero 2008

DECLARACIÓN

Nosotros, Cristian Marcelo Dibujes Salgado y Edwin Alexis Torres Morales, declaramos que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

La Escuela Politécnica Nacional, puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Cristian M. Dibujes S.

Edwin A. Torres M.

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Cristian Marcelo Dibujes Salgado y Edwin Alexis Torres Morales, bajo mi supervisión.

Dr. Luis Corrales
DIRECTOR DEL PROYECTO

AGRADECIMIENTO

A **Dios** y a la **Virgen María** por darme,
la oportunidad de culminar mi carrera
y a mi madre **María Fabiola Salgado**
por darme la vida y ser el eje principal
de mi familia, quien siempre me dio
ánimo para luchar y nunca
darme por vencido.

Cristian M. Dibujes S.

AGRADECIMIENTO

Agradecimiento, a todos aquellos profesores de la Facultad de Ingeniería Eléctrica y Electrónica de la Escuela Politécnica Nacional, que supieron enseñarme a trazar mi ruta profesional y de los cuales siempre recibí el apoyo cognoscitivo necesario para cualquier duda que toda labor estudiantil presenta. Y el agradecimiento más grande y eterno para mis padres, el **Dr. Ernesto Torres** y la **Lcda. Hipatia Morales**, quienes, no solo durante la realización de este proyecto, sino siempre, fueron y son el pilar fundamental en mi vida, con cuyo apoyo, entrega, amor incondicional y sobre todo con su ejemplo, siempre sé que puedo salir adelante en todos los retos que se presenten, son lo más importante y lo más valioso que tengo, los quiero mucho....

Edwin A. Torres M.

DEDICATORIA

A todas aquellas personas que estuvieron junto a mi,
apoyándome y enseñándome la responsabilidad
que conlleva obtener el título de ingeniero,
les retribuyo, por darme su confianza y amistad
y que siempre estarán presentes, en cualquier
actividad que desempeñe.

Cristian M. Dibujes S.

DEDICATORIA

A todas aquellas personas que de muchas formas estuvieron en aquellos momentos en que parecía que el esfuerzo no rendía frutos, les ofrendo este esfuerzo, porque supieron ayudarme a no desmayar y a levantarme para continuar con más ánimo.

Edwin A. Torres M.

ÍNDICE

| | |
|---|-----|
| RESUMEN | xi |
| PRESENTACIÓN | xii |
| CAPÍTULO 1 ESTUDIO DE LAS CARACTERÍSTICAS DE UN OSCILOSCOPIO | 1 |
| 1.1 OBJETIVO PRINCIPAL | 2 |
| 1.2 OSCILOSCOPIOS TRADICIONALES | 2 |
| 1.2.1 OSCILOSCOPIOS DIGITALES VS. OSCILOSCOPIOS ANÁLOGOS | 3 |
| 1.2.2 VENTAJAS DEL OSCILOSCOPIO PROPUESTO FRENTE A LOS TRADICIONALES | 5 |
| 1.3 MECANISMOS DE DISPARO (TRIGGER) | 5 |
| 1.4 TIPOS DE MUESTREO | 6 |
| 1.4.1 TIEMPO REAL..... | 6 |
| 1.4.2 TIEMPO EQUIVALENTE | 8 |
| 1.4.2.1 Muestreo aleatorio repetitivo | 8 |
| 1.4.2.2 Muestreo secuencial..... | 9 |
| 1.4.2.3 Submuestreo | 10 |
| 1.5 MÉTODOS DE CONVERSIÓN Y TRANSFERENCIA | 12 |
| 1.5.1 DE MANERA CONTINUA..... | 12 |
| 1.5.2 POR RÁFAGAS..... | 13 |
| 1.6 ERRORES QUE SUCEDEN DURANTE EL MUESTREO | 13 |
| 1.6.1 ERROR DE COMPENSACIÓN | 14 |
| 1.6.2 ERROR DE GANANCIA | 14 |
| 1.6.3 ERROR DE APERTURA | 15 |
| 1.6.4 ERROR DE NO-LINEALIDAD DIFERENCIAL | 16 |
| 1.6.5 ERROR DE NO-LINEALIDAD INTEGRAL | 17 |
| 1.6.6 ERROR DE CUANTIZACIÓN | 18 |
| 1.6.7 ERROR ABSOLUTO..... | 19 |
| 1.7 MODO DE COMUNICACIÓN | 20 |
| 1.7.1 USB (UNIVERSAL SERIAL BUS)..... | 20 |
| 1.7.1.1 Tipos de transferencia | 21 |
| 1.7.1.1.1 Transferencia de datos isócrona | 22 |
| 1.7.1.2 Cables y conectores. | 23 |
| 1.7.1.3 Sustitución de los puertos serie y paralelo | 24 |
| 1.7.1.4 Ventajas de utilizar el interfaz USB..... | 24 |
| 1.7.1.5 Desventajas de utilizar el interfaz USB. | 25 |
| 1.8 PROPUESTA DE DISEÑO | 25 |
| CAPÍTULO 2 DISEÑO Y CONSTRUCCIÓN DEL HARDWARE DEL OSCILOSCOPIO | 28 |

| | | |
|---------|---|----|
| 2.1 | GENERALIDADES..... | 29 |
| 2.2 | DISEÑO DE LAS PROTECCIONES, ACONDICIONAMIENTO Y FILTRADO EN CADA CANAL | 30 |
| 2.2.1 | CIRCUITO DE ENTRADA DE UN CANAL | 30 |
| 2.2.2 | DISEÑO DEL CIRCUITO DE CONTROL DE RELÉS QUE ACCIONAN LOS ACONDICIONAMIENTOS DE VOLTAJE Y LOS ACOPLAMIENTOS..... | 32 |
| 2.2.2.1 | Decodificador binario 74HCT139 | 33 |
| 2.2.2.2 | Interruptores bilaterales 74HC4066..... | 33 |
| 2.2.3 | DISEÑO DE CIRCUITOS DE ACCIONAMIENTO DE ACONDICIONAMIENTOS DE VOLTAJE Y ACOPLAMIENTOS | 34 |
| 2.2.3.1 | Circuito de accionamiento de acoplamiento..... | 35 |
| 2.2.3.2 | Circuito de accionamiento de acondicionamientos de voltaje | 35 |
| 2.2.4 | ETAPA DE ACONDICIONAMIENTO DE LAS SEÑALES DE VOLTAJE..... | 37 |
| 2.2.4.1 | Diseño de la etapa de acondicionamiento de las señales de voltaje..... | 39 |
| 2.2.4.2 | Descripción del circuito diseñado | 39 |
| 2.2.5 | FILTRO ANTIALIASING A LA SALIDA DEL CIRCUITO DE ACONDICIONAMIENTO | 40 |
| 2.2.5.1 | Efecto aliasing | 40 |
| 2.2.5.2 | Filtro anti-aliasing..... | 41 |
| 2.2.5.3 | Parámetros de un filtro pasa bajos | 42 |
| 2.2.5.4 | Elección de la frecuencia de muestreo y de la frecuencia del filtro | 43 |
| 2.2.5.5 | Filtro RC pasabajos utilizado..... | 44 |
| 2.2.5.6 | Filtro de segundo orden pasabajos utilizado | 44 |
| 2.2.6 | DISEÑO DEL FILTRO ANTI-ALIASING IMPLEMENTADO..... | 45 |
| 2.3 | DISEÑO DE LA TARJETA DE CONTROL Y COMUNICACIÓN | 48 |
| 2.3.1 | ADQUISICIÓN Y ADMINISTRACIÓN DE LOS DATOS DEL AD..... | 48 |
| 2.3.1.1 | Descripción de las características más relevantes del dsPIC30F3011 | 49 |
| 2.3.1.2 | INTERRUPCIONES EN EL DSPIC30F3011..... | 50 |
| 2.3.1.3 | Módulo AD de 10 bits del dsPIC30F3011..... | 51 |
| 2.3.2 | MODULO DE COMUNICACIÓN USBMOD4..... | 52 |
| 2.3.2.1 | Descripción de características del USBMOD4..... | 53 |
| 2.3.2.2 | Distribución de pines y modo de uso del USBMOD4 | 53 |
| 2.3.2.3 | Instalación de controladores del USBMOD4..... | 55 |
| 2.3.3 | DESCRIPCIÓN DE FUNCIONAMIENTO DE LA TARJETA DE CONTROL Y COMUNICACIÓN..... | 55 |
| 2.3.3.1 | Requerimientos del control..... | 55 |
| 2.3.3.2 | Conversión AD | 56 |
| 2.3.3.3 | Tratamiento de los datos resultantes de la conversión AD | 56 |
| 2.3.3.4 | Control de la comunicación entre el dsPIC y el USBMOD4..... | 57 |
| 2.3.3.5 | Control de los circuitos de acondicionamiento y de los acoplamientos..... | 59 |

| | | |
|---|--|-----------|
| 2.4 | DIAGRAMAS CIRCUITALES DEFINITIVOS | 61 |
| 2.5 | CONSUMO DE POTENCIA | 64 |
| 2.5.1 | REQUERIMIENTOS DE ENERGÍA | 64 |
| 2.5.2 | CÁLCULO DEL CONSUMO DE POTENCIA..... | 64 |
| CAPÍTULO 3 DESARROLLO DEL PROGRAMA DEL DSPIC30F3011 (FIRMWARE) | | 66 |
| 3.1 | INTRODUCCIÓN AL DESARROLLO DEL FIRMWARE..... | 67 |
| 3.2 | HERRAMIENTA DE PROGRAMACIÓN MPLAB C30 | 67 |
| 3.3 | IDENTIFICACIÓN DE TODAS LAS ÓRDENES EMITIDAS DESDE LA PC, CON LA INTERRUPCIÓN EXTERNA INT0..... | 68 |
| 3.4 | CONEXIÓN Y ENCENDIDO DEL DISPOSITIVO..... | 79 |
| 3.5 | MODO EN TIEMPO REAL | 80 |
| 3.5.1 | ADQUISICIÓN DE DATOS PARA MODO DE TIEMPO REAL | 81 |
| 3.5.2 | LECTURA Y PREPARACIÓN DE DATOS PARA MODO DE TIEMPO REAL..... | 82 |
| 3.5.3 | ACONDICIONAMIENTO Y ENVÍO DE TRAMAS PARA MODO DE TIEMPO REAL .. | 83 |
| 3.6 | MODO DE ALMACENAMIENTO DE DATOS..... | 85 |
| 3.6.1 | ADQUISICIÓN DE DATOS PARA MODO DE ALMACENAMIENTO..... | 85 |
| 3.6.2 | LECTURA Y PREPARACIÓN DE DATOS PARA MODO DE ALMACENAMIENTO.... | 86 |
| 3.6.3 | ACONDICIONAMIENTO Y REGISTRO DE TRAMAS PARA MODO DE ALMACENAMIENTO | 87 |
| 3.6.4 | ENVÍO DE TRAMAS PARA EL MODO DE ALMACENAMIENTO | 88 |
| 3.6.5 | USO DE LA INTERRUPCIÓN EXTERNA INT2 DEL DSPIC PARA EL MODO DE ALMACENAMIENTO | 90 |
| 3.7 | ALGORITMO DEL TRIGGER..... | 90 |
| CAPÍTULO 4 DESARROLLO DE LA INTERFAZ DEL OSCILOSCOPIO EN LA PC | | 95 |
| 4.1 | REQUERIMIENTOS BÁSICOS PARA EL FUNCIONAMIENTO DEL SOFTWARE..... | 96 |
| 4.2 | PRESENTACIÓN DE LA INTERFAZ DE USUARIO | 99 |
| 4.3 | EVENTOS Y SUBROUTINAS MÁS IMPORTANTES PARA EL CONTROL DEL OSCILOSCOPIO | 101 |
| 4.3.1 | CONEXIÓN MANUAL DEL MÓDULO..... | 101 |
| 4.3.2 | CONEXIÓN AUTOMÁTICA DEL MÓDULO | 102 |
| 4.3.3 | DESCONEXIÓN DEL MÓDULO..... | 104 |
| 4.3.4 | ENCENDIDO Y APAGADO DEL OSCILOSCOPIO | 104 |
| 4.3.5 | ENCENDIDO Y APAGADO DE LOS CANALES | 106 |
| 4.3.6 | ESCALAS DE TIEMPO | 107 |
| 4.3.7 | ESCALAS DE VOLTAJE..... | 108 |
| 4.3.7.1 | Eventos de escalamiento de voltaje para el circuito de acondicionamiento de escala máxima | 109 |

| | | |
|---|---|-----|
| 4.3.7.2 | Eventos de escalamiento de voltaje para el circuito de acondicionamiento de escala media | 110 |
| 4.3.7.3 | Eventos de escalamiento de voltaje para el circuito de acondicionamiento de escala mínima | 112 |
| 4.3.7.4 | Evento de final de base de tiempo de escalamiento de protección | 114 |
| 4.3.8 | EVENTOS DE ACOPLAMIENTOS (AC, DC, GND) PARA UN CANAL | 115 |
| 4.3.9 | AUTOESCALAMIENTO | 115 |
| 4.3.9.1 | Evento de autoescalamiento para un canal | 116 |
| 4.3.9.2 | Evento de finalización de tiempo de autoescalamiento | 118 |
| 4.3.9.3 | Aplicación del circuito de escala máxima para un canal | 120 |
| 4.3.9.4 | Revisión de sub-escalas de la escala máxima para un canal | 121 |
| 4.3.9.5 | Revisión de sub-escalas de la escala media para los 3 canales | 121 |
| 4.3.9.6 | Aplicación del circuito de escala media para un canal | 122 |
| 4.3.9.7 | Revisión de sub-escalas de la escala media para un canal | 123 |
| 4.3.9.8 | Revisión de sub-escalas de la escala mínima para los 3 canales | 124 |
| 4.3.9.9 | Aplicación del circuito de escala mínima para un canal | 125 |
| 4.3.9.10 | Evento de finalización de base de tiempo para autoescalar con la mínima escala para los 3 canales | 125 |
| 4.3.9.11 | Revisión de sub-escalas de la escala mínima para un canal | 126 |
| 4.3.10 | TRIGGER | 128 |
| 4.3.10.1 | Evento de encendido/apagado del trigger | 128 |
| 4.3.10.2 | Evento de aplicación del trigger respecto a un canal | 129 |
| 4.3.10.3 | Evento de aplicación del trigger con flanco de subida (positivo) | 129 |
| 4.3.10.4 | Evento de aplicación de nivel variable (manual) para el trigger | 130 |
| 4.3.10.5 | Evento de aplicación de nivel fijo (automático) para el trigger | 130 |
| 4.3.10.6 | Finalización de base de tiempo y cálculo del nivel para el trigger en modo fijo (automático) | 131 |
| 4.3.11 | RECEPCIÓN DE TRAMAS | 132 |
| 4.3.11.1 | Inicio del evento de recepción de tramas | 132 |
| 4.3.11.2 | Inicio de trazado de la(s) señal(es) | 133 |
| 4.3.11.3 | Trazado y definición de condiciones de escalamiento | 135 |
| 4.3.11.4 | Definición de condiciones de autoescalamiento | 137 |
| CAPÍTULO 5 PRUEBAS Y RESULTADOS DEL OSCILOSCOPIO | | 139 |
| 5.1 | ELEMENTOS IMPORTANTES A CONSIDERAR PREVIOS A LAS PRUEBAS | 140 |
| 5.1.1 | RESUMEN DE INCOVENIENTES Y SOLUCIONES DE LA ETAPA DE DISEÑO | 140 |
| 5.1.2 | CALIBRACIÓN | 141 |
| 5.2 | ESPECIFICACIONES DEL OSCILOSCOPIO | 142 |
| 5.3 | REALIZACIÓN DE LAS PRUEBAS | 143 |
| 5.3.1 | EQUIPO QUE SE UTILIZÓ EN LA REALIZACIÓN DE LAS PRUEBAS | 143 |

| | | |
|---|---|------------|
| 5.3.2 | INCONVENIENTES Y LIMITACIONES DEL PROCESO DE REALIZACIÓN DE LAS PRUEBAS | 143 |
| 5.4 | RESULTADOS DE LAS PRUEBAS REALIZADAS | 144 |
| 5.4.1 | PRUEBA CON UNA ONDA TRIANGULAR A 60HZ Y AMPLITUD 2.5V..... | 144 |
| 5.4.2 | PRUEBA CON UNA ONDA A 1KHZ Y AMPLITUD 2.5V..... | 145 |
| 5.4.3 | PRUEBA CON UNA ONDA A 5KHZ Y AMPLITUD 2.5V..... | 146 |
| 5.4.4 | PRUEBA CON UNA ONDA A 10KHZ Y AMPLITUD 25V | 147 |
| 5.4.5 | PRUEBA CON UNA ONDA A 20KHZ Y AMPLITUD 25V | 148 |
| 5.4.6 | PRUEBA CON UNA ONDA A 30KHZ Y AMPLITUD 25V | 149 |
| 5.4.7 | PRUEBA CON UNA ONDA A 40KHZ Y AMPLITUD 25V..... | 150 |
| 5.4.8 | PRUEBA CON DOS FORMAS DE ONDA, UNA SENOIDAL DE 1KHZ Y UNA CUADRADA DE 10KHZ, AMBAS CON AMPLITUD DE 25V | 151 |
| 5.4.9 | PRUEBA CON DOS FORMAS DE ONDA, UNA CUADRADA DE 60HZ Y UNA TRIANGULAR DE 5KHZ, AMBAS CON AMPLITUD DE 25V | 152 |
| 5.4.10 | PRUEBA CON DOS FORMAS DE ONDA, UNA CUADRADA DE 20KHZ Y UNA CUADRADA DE 40KHZ, AMBAS CON AMPLITUD DE 25V | 153 |
| 5.5 | ANÁLISIS DE RESULTADOS Y ERRORES..... | 154 |
| 5.5.1 | IDENTIFICACIÓN DE ERRORES EN EL MUESTREO | 154 |
| 5.5.2 | ANÁLISIS DE RESULTADOS Y ERRORES DE MEDICIÓN CON RESPECTO AL OSCILOSCOPIO COMERCIAL | 154 |
| 5.5.2.1 | Errores en la medición de frecuencia..... | 154 |
| 5.5.2.2 | Errores en la medición de voltaje..... | 157 |
| 5.6 | COSTOS DEL PROYECTO..... | 160 |
| 5.6.1 | COSTO DE LA LICENCIA DEL SOFTWARE VISUAL BASIC 6.0..... | 160 |
| 5.6.2 | COSTO DE DISEÑO, DESARROLLO Y CONSTRUCCIÓN | 160 |
| 5.6.3 | COSTOS DE LOS ELEMENTOS DEL HARDWARE..... | 160 |
| 5.6.3.1 | Costos de chasis y fuente | 161 |
| 5.6.3.2 | Costos de la tarjeta de comunicación y control..... | 161 |
| 5.6.3.3 | Costos de las tarjetas de los canales..... | 162 |
| 5.6.4 | ANÁLISIS DEL COSTO TOTAL DEL PROYECTO..... | 163 |
| 5.7 | COMPARACIÓN CON OTROS EQUIPOS SIMILARES DISPONIBLES EN EL MERCADO | 164 |
| CAPÍTULO 6 CONCLUSIONES Y RECOMENDACIONES..... | | 167 |
| 6.1 | CONCLUSIONES..... | 168 |
| 6.2 | RECOMENDACIONES | 170 |
| BIBLIOGRAFÍA | | 172 |
| ANEXOS | | 174 |

ÍNDICE DE FIGURAS

| | |
|---|----|
| Figura 1.1 (a) Osciloscopio analógico de 2 canales, 20 MHz. (b) Osciloscopio analógico/digital de 2 canales, 150 MHz..... | 2 |
| Figura 1.2 Esquematación de un Osciloscopio Analógico. | 3 |
| Figura 1.3 Método de muestreo en tiempo real. | 7 |
| Figura 1.4 Método de muestreo aleatorio repetitivo..... | 8 |
| Figura 1.5 Método de muestreo secuencial. | 10 |
| Figura 1.6 Espectro completo de una señal. | 11 |
| Figura 1.7 Efecto del aliasing. | 11 |
| Figura 1.8 Espectro de la señal filtrada. | 11 |
| Figura 1.9 La señal muestreada no es afectada..... | 12 |
| Figura 1.10 Error de compensación..... | 14 |
| Figura 1.11 Error de ganancia. | 15 |
| Figura 1.12 Error de apertura. | 16 |
| Figura 1.13 DNL Error. | 16 |
| Figura 1.14 INL Error..... | 17 |
| Figura 1.15 Error de cuantificación. | 18 |
| Figura 1.16 Error absoluto..... | 19 |
| Figura 1.17 Conectores de la serie A y la serie B..... | 23 |
| Figura 1.18 Diagrama esquemático del presente proyecto. | 26 |
| | |
| Figura 2.1 Circuito de Protecciones de entrada. | 31 |
| Figura 2.2 Circuito de Acoplamientos (AC, DC, Tierra). | 34 |
| Figura 2.3 Circuito de escalamientos 2.5 V, 25 V, 250 V. | 35 |
| Figura 2.4 Circuito de acondicionamiento de las señales de voltaje. | 39 |
| Figura 2.5 Demostrando aliasing, rojo es la verdadera onda, mientras azul es un alias. | 41 |
| Figura 2.6 Parámetros fundamentales de un Filtro Pasa Bajos..... | 42 |
| Figura 2.7 Relación entre frecuencia de muestreo y frecuencia del filtro. | 43 |
| Figura 2.8 Circuito pasa bajo..... | 44 |
| Figura 2.9 Respuesta de ganancia en el dominio de la frecuencia..... | 44 |
| Figura 2.10 Filtro pasa bajos de segundo orden. | 44 |
| Figura 2.11 Filtro anti-aliasing implementado. | 47 |
| Figura 2.12 Módulo de comunicación USBMOD4..... | 52 |
| Figura 2.13 Conexión de líneas de datos y comunicación entre el dsPIC30F3011 y el USBMOD4. | 58 |
| Figura 2.14 Circuito de la tarjeta de control y comunicación..... | 60 |
| | |
| Figura 3.1 Inicio de la interrupción INT0..... | 68 |
| Figura 3.2 Identificación de todas las órdenes..... | 70 |

| | |
|---|-----|
| Figura 3.3 Conexión, desconexión, encendido y apagado del osciloscopio. | 79 |
| Figura 3.4 Algoritmo de adquisición de datos para modo de TIEMPO REAL. | 81 |
| Figura 3.5 Algoritmo de lectura y preparación de datos para modo de TIEMPO REAL. | 82 |
| Figura 3.6 Algoritmo de acondicionamiento y envío de tramas para modo de TIEMPO REAL. | 83 |
| Figura 3.7 Algoritmo de adquisición de datos para modo de ALMACENAMIENTO. | 85 |
| Figura 3.8 Algoritmo de lectura y preparación de datos para el modo de ALMACENAMIENTO. | 87 |
| Figura 3.9 Algoritmo de acondicionamiento y registro de tramas para modo de ALMACENAMIENTO. | 87 |
| Figura 3.10 Algoritmo de envío de tramas para el modo de ALMACENAMIENTO. | 88 |
| Figura 3.11 Confirmación de envío de cada trama en el modo de ALMACENAMIENTO. | 90 |
| Figura 3.12 Algoritmo de comprobación previa a la aplicación del trigger. | 91 |
| Figura 3.13 Aplicación del trigger. | 93 |
| | |
| Figura 4.1 Ventana de Administrador de dispositivos en Windows XP. | 97 |
| Figura 4.2 Ventana de Propiedades del <i>Virtual Com Port</i> | 98 |
| Figura 4.3 Ventana <i>Advanced Settings</i> | 98 |
| Figura 4.4 Interfaz gráfica del osciloscopio. | 99 |
| Figura 4.5 Ventana de “INFORME RÁPIDO”. | 100 |
| Figura 4.6 Ventana de “MEDICIÓN CON CURSORES”. | 100 |
| Figura 4.7 Ventana de conexión manual. | 101 |
| Figura 4.8 Algoritmos de conexión manual del módulo. | 102 |
| Figura 4.9 Conexión automática del módulo. | 103 |
| Figura 4.10 Desconexión del módulo. | 104 |
| Figura 4.11 Algoritmo de encendido y apagado del osciloscopio. | 105 |
| Figura 4.12 Algoritmo de encendido y apagado del canal 1. | 106 |
| Figura 4.13 Evento de escalamiento en tiempo para la escala de 5 ms/DIV. | 107 |
| Figura 4.14 Evento de escalamiento de voltaje en 10 V/DIV para el canal 1. | 109 |
| Figura 4.15 Evento de escalamiento de voltaje en 1 V/DIV para el canal 1. | 111 |
| Figura 4.16 Evento de escalamiento de voltaje en 10 mV/DIV para el canal 1. | 113 |
| Figura 4.17 Algoritmo de comprobación para aplicar o no escalamiento de protección. | 113 |
| Figura 4.18 Evento de finalización de base de tiempo de escalamiento de protección. | 114 |
| Figura 4.19 Evento de finalización de base de tiempo de escalamiento de protección (continuación). | 115 |
| Figura 4.20 Eventos de acoplamiento para el canal 1. | 115 |
| Figura 4.21 Evento de autoescalamiento para el canal 1. | 117 |
| Figura 4.22 Evento de finalización de tiempo de autoescalamiento. | 118 |
| Figura 4.23 Comprobación final de evento de autoescalamiento. | 119 |
| Figura 4.24 Aplicación del circuito de escala máxima para el canal 1. | 120 |
| Figura 4.25 Algoritmo de revisión de subescalas de la escala máxima para el canal 1. | 120 |
| Figura 4.26 Inicio de revisión de subescalas para escala media. | 121 |
| Figura 4.27 Revisión de subescalas para escala media (continuación). | 122 |
| Figura 4.28 Aplicación del circuito acondicionador de escala media para el canal 1. | 123 |

| | |
|--|-----|
| Figura 4.29 Revisión de subescalas de la escala media para el canal 1..... | 123 |
| Figura 4.30 Inicio de revisión de subescalas de la escala mínima..... | 124 |
| Figura 4.31 Aplicación del circuito acondicionador de escala mínima para el canal 1..... | 125 |
| Figura 4.32 Evento de finalización de base de tiempo para autoescalar con la mínima escala..... | 126 |
| Figura 4.33 Revisión de subescalas del circuito acondicionador de escala mínima para el canal 1..... | 126 |
| Figura 4.34 Revisión de subescalas del circuito acondicionador de escala mínima para el canal 1 (continuación)..... | 127 |
| Figura 4.35 Evento de encendido y apagado del trigger..... | 128 |
| Figura 4.36 Evento de aplicación del trigger respecto al canal 1..... | 129 |
| Figura 4.37 Evento de aplicación del trigger con flanco de subida..... | 129 |
| Figura 4.38 Evento de aplicación de nivel variable (manual) para el trigger..... | 130 |
| Figura 4.39 Evento de aplicación de nivel fijo para el trigger..... | 130 |
| Figura 4.40 Cálculo del nivel del trigger en modo fijo (automático)..... | 131 |
| Figura 4.41 Inicio de evento de recepción de tramas..... | 132 |
| Figura 4.42 Limpieza de pantalla para empezar a graficar otra..... | 133 |
| Figura 4.43 Inicio de graficación de la(s) señal(es)..... | 134 |
| Figura 4.44 Graficación de señales para el canal 1 y definición de condiciones de escalamiento..... | 135 |
| Figura 4.45 Graficación de señales para el canal 2 y definición de condiciones de escalamiento..... | 136 |
| Figura 4.46 Graficación de señales para el canal 3 y definición de condiciones de escalamiento..... | 137 |
| Figura 4.47 Definición de condiciones de autoescalamiento..... | 138 |
| | |
| Figura 5.1 Onda triangular a 60 Hz y 2.5V de amplitud, tomada con el osciloscopio TDS210..... | 144 |
| Figura 5.2 Onda triangular a 60 Hz y 2.5V de amplitud, tomada con el osciloscopio para PC..... | 144 |
| Figura 5.3 Onda triangular a 1 kHz y 2.5V de amplitud, tomada con el osciloscopio TDS210..... | 145 |
| Figura 5.4 Onda triangular a 1 kHz y 2.5V de amplitud tomada con el osciloscopio para PC..... | 145 |
| Figura 5.5 Onda triangular a 5 kHz y 2.5V de amplitud tomada con el osciloscopio TDS210..... | 146 |
| Figura 5.6 Onda triangular a 5 kHz y 2.5V de amplitud tomada con el osciloscopio para PC..... | 146 |
| Figura 5.7 Onda senoidal a 10 kHz y 25V de amplitud tomada con el osciloscopio TDS210..... | 147 |
| Figura 5.8 Onda senoidal a 10 kHz y 25V de amplitud tomada con el osciloscopio para PC..... | 147 |
| Figura 5.9 Onda senoidal a 20 kHz y 25V de amplitud tomada con el osciloscopio TDS210..... | 148 |
| Figura 5.10 Onda senoidal a 20 kHz y 25V de amplitud tomada con el osciloscopio para PC..... | 148 |
| Figura 5.11 Onda senoidal a 30 kHz y 25V de amplitud tomada con el osciloscopio TDS210..... | 149 |
| Figura 5.12 Onda senoidal a 30 kHz y 25V de amplitud tomada con el osciloscopio para PC..... | 149 |
| Figura 5.13 Onda senoidal a 40 kHz y 25V de amplitud tomada con el osciloscopio TDS210..... | 150 |
| Figura 5.14 Onda senoidal a 40 kHz y 25V de amplitud tomada con el osciloscopio para PC..... | 150 |
| Figura 5.15 Onda senoidal a 1 kHz y una cuadrada a 10 kHz, las dos con 25V de amplitud tomadas con el osciloscopio TDS210..... | 151 |
| Figura 5.16 Onda senoidal a 1 kHz y una cuadrada a 10 kHz, las dos con 25V de amplitud tomadas con el osciloscopio para PC..... | 151 |

| | |
|---|-----|
| Figura 5.17 Onda cuadrada a 60 Hz y una triangular a 5 kHz, las dos con 25V de amplitud tomadas con el osciloscopio TDS210. | 152 |
| Figura 5.18 Onda cuadrada a 60 Hz y una triangular a 5 kHz, las dos con 25V de amplitud tomadas con el osciloscopio para PC. | 152 |
| Figura 5.19 Onda cuadrada a 20 KHz y una cuadrada a 40 kHz, las dos con 25V de amplitud tomadas con el osciloscopio TDS210..... | 153 |
| Figura 5.20 Onda cuadrada a 20 KHz y una cuadrada a 40 kHz, las dos con 25V de amplitud tomadas con el osciloscopio para PC. | 153 |
| Figura 5.21 Osciloscopio PHILIPS PM3323..... | 165 |
| Figura 5.22 Osciloscopio GDS-810S. | 165 |
| Figura 5.23 Osciloscopio para PC, ETC M621/E..... | 165 |
| Figura 5.24 Osciloscopio para PC, ETC M522. | 166 |

ÍNDICE DE TABLAS

| | |
|---|-----|
| Tabla 1.1 Índices máximos y fases de los cuatro tipos de transferencia..... | 22 |
| Tabla 2.1 Consumo máximo para la tarjeta de un canal. | 31 |
| Tabla 2.2 Coeficientes del filtro Butterworth pasa-bajos de segundo orden. | 46 |
| Tabla 2.3 Distribución de pines del módulo USBMOD4 (pines más importantes)..... | 54 |
| Tabla 2.4 Función de las salidas del dsPIC para el control de los circuitos de acondicionamiento y acoplamiento. | 59 |
| Tabla 2.5 Consumo de potencia de componentes activos. | 64 |
| Tabla 3.1 Estructura de TRAMAS y las 6 posibilidades de la cabecera. | 83 |
| Tabla 4.1 Equivalencias de valores límite en 10 bits de resolución vertical, para las subescalas de la escala máxima (-250V a 250V)..... | 121 |
| Tabla 4.2 Equivalencias de valores máximos en 10 bits de resolución vertical, para las subescalas de la escala media (-25V a 25V) en base a la escala máxima. | 124 |
| Tabla 4.3 Equivalencias de valores máximos en 10 bits de resolución vertical, para las subescalas de la escala mínima (-2.5V a 2.5V) en base a la misma escala mínima. | 127 |
| Tabla 5.1 Especificaciones del osciloscopio. | 142 |
| Tabla 5.2 Errores en la medición de frecuencia..... | 156 |
| Tabla 5.3 Errores en la medición de voltaje. | 159 |
| Tabla 5.4 Detalle de los costos de los elementos para chasis y fuente. | 161 |
| Tabla 5.5 Detalle de los costos referentes a la tarjeta de comunicación y control..... | 161 |
| Tabla 5.6 Detalle de los costos referentes a las tarjetas de los 3 canales. | 162 |

| | |
|--|-----|
| Tabla 5.7 Costo total del primer prototipo..... | 163 |
| Tabla 5.8 Costo por módulo para 3 máquinas. | 163 |
| Tabla 5.9 Costo mínimo por módulo..... | 164 |

RESUMEN

En este proyecto se diseña y construye un osciloscopio que emplea una PC para desplegar las señales a ser analizadas. El objetivo fue ofrecer una alternativa a un osciloscopio comercial. El osciloscopio debía ser capaz de trabajar con las señales que típicamente se encuentran en la industria (60 o 120 Hz) pero debe ser de bajo costo.

El diseño del osciloscopio se basó en un dsPIC, específicamente el **dsPIC30F3011** para aprovechar su velocidad de procesamiento.

El Firmware del dsPIC fue desarrollado en lenguaje C, utilizando la herramienta de programación **MPLAB C30**. La interfaz gráfica (programa **VISUALSCOPE 1.0.1**) fue desarrollada en el lenguaje **Visual Basic 6.0**.

La comunicación debe gestionarse utilizando la interfaz USB que posee cualquier computador en la actualidad.

Para transferir las señales desde el sistema microprocesado hasta la PC se empleó el pórtico USB. El módulo comercial **USBMOD4**, sirvió para posibilitar la conexión entre el dsPIC y el módulo.

El equipo diseñado consta de 3 canales, tiene autoescalamiento de voltaje para poder medir formas de onda de hasta 200 Vpico.

El módulo fue probado y funcionó como se esperaba, pero presentó algunas limitantes en cuanto a respuesta de frecuencia. Alrededor de los 15kHz, el sistema pierde precisión para dibujar correctamente estas señales. Hasta 40kHz el sistema trabaja con un porcentaje de error del 2.24%.

PRESENTACIÓN

El uso de un computador de escritorio o portátil, es muy extendido a todo nivel, por tanto, que mejor que utilizarlo como una herramienta que posibilite algún ahorro de dinero. Todos aquellos que se dedican de una u otra forma a trabajos en el campo de la electrónica, saben de la importancia de tener al alcance un osciloscopio.

La necesidad de un osciloscopio es más notoria en los laboratorios estudiantiles como es el caso del laboratorio de instrumentación de la Facultad de Ingeniería Eléctrica y Electrónica. Es por esto que se pensó en construir un osciloscopio basado en un dsPIC, que utilice el monitor de una PC para abaratar su costo.

Con este propósito, el presente trabajo se desarrolla de la manera siguiente:

En el Capítulo 1 se presenta una descripción de las principales características que definen a un osciloscopio en general, para tener una base para comprender el funcionamiento del mismo.

En el Capítulo 2 se detalla el diseño y construcción del equipo, debiendo mencionar como sus partes constitutivas más importantes, la tarjeta de control y comunicación donde está el cerebro de todo el funcionamiento que es el dsPIC, y también el módulo USBMOD4 que gestiona la comunicación vía USB.

En el Capítulo 3 se detalla el desarrollo del programa del dsPIC (Firmware), el cual gestiona todo el control y funcionamiento del osciloscopio para PC.

En el Capítulo 4 se presentan el desarrollo de los algoritmos y subrutinas más importantes del programa que constituye la interfaz gráfica del osciloscopio (VISUALSCOPE 1.0.1) y su interacción con el Firmware del dsPIC.

En el Capítulo 5 se realizan las pruebas de funcionamiento de todo el sistema para comprobar su correcto funcionamiento.

Finalmente, en el Capítulo 6, se exponen las conclusiones y recomendaciones que se extraen del desarrollo y resultados del presente proyecto.

CAPÍTULO 1

ESTUDIO DE LAS CARACTERÍSTICAS DE UN OSCILOSCOPIO

ESTUDIO DE LAS CARACTERÍSTICAS DE UN OSCILOSCOPIO

1.1 OBJETIVO PRINCIPAL

El Objetivo Principal del presente proyecto es diseñar y construir un osciloscopio para aplicaciones eléctricas y de control de bajo costo, y así evitar el gasto en los muy costosos osciloscopios que existen en el mercado. Para esto se analizan a continuación algunas de las características que poseen algunos de estos.

El presente proyecto está basado en un microcontrolador dsPIC como el cerebro de todo el funcionamiento. El HMI y la interfaz gráfica se presentan en la pantalla de la PC.

1.2 OSCILOSCOPIOS TRADICIONALES

Los osciloscopios tradicionalmente se componen de un hardware cuya interfaz gráfica se basa en un CRT (tubo de rayos catódicos) o los más modernos con pantalla LCD, diseñados para mostrar variaciones de voltaje (periódicas o no periódicas), esto los hace voluminosos y muy caros. En la Figura 1.1 se muestran dos tipos de osciloscopios que existen en el mercado.



Figura 1.1 (a) Osciloscopio analógico de 2 canales, 20 MHz. (b) Osciloscopio analógico/digital de 2 canales, 150 MHz.

En el presente proyecto, el Hardware se basa en un microcontrolador dsPIC para el control de los canales y acondicionamiento de las señales, mientras que

su interfaz gráfica o HMI se presenta mediante software en la pantalla de cualquier PC.

Etimológicamente la palabra *OSCILOSCOPIO* se forma del latín “*oscillare*” que significa girar al revés y al derecho, y del griego clásico “*Skopein*” que significa observar, apuntar o examinar.

El corazón del tradicional osciloscopio CRT es la propia pantalla del CRT (tubo de rayos catódicos). El CRT es una bombilla de vidrio al vacío. En el frente se tiene una pantalla de vidrio, que está cubierta por dentro con un material de fósforo. Este fósforo brillará produciendo luz, por el choque de los electrones. La parte trasera está formada por el cañón de electrones. Un pequeño elemento calentador es contenido dentro de un cilindro de metal llamado cátodo. Cuando el calentador es activado por la aplicación de un voltaje a través de él, la temperatura del cátodo se eleva y entonces emite electrones.

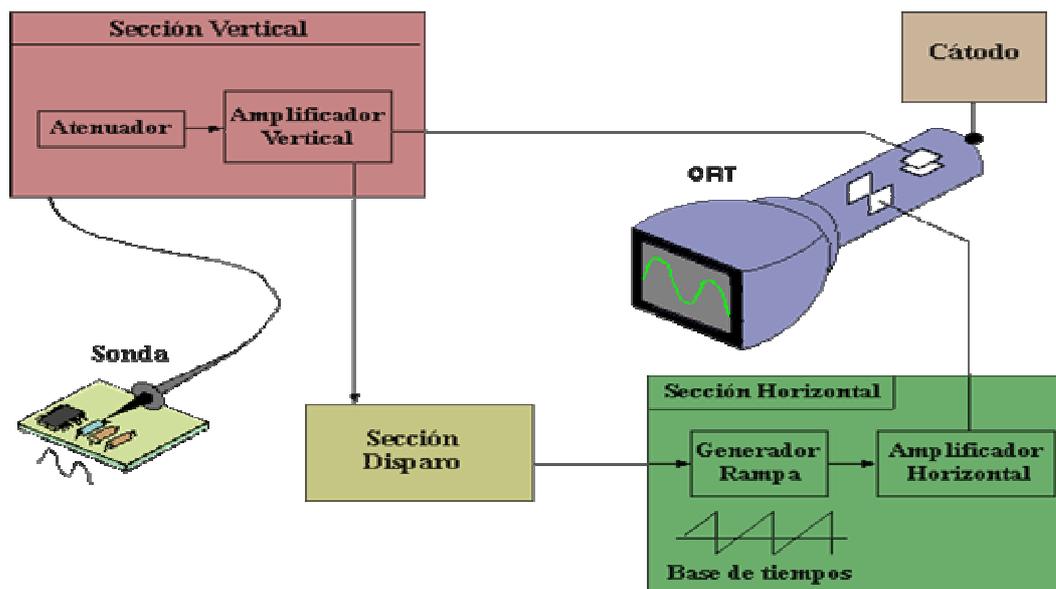


Figura 1.2 Esquematación de un Osciloscopio Analógico¹.

1.2.1 OSCILOSCOPIOS DIGITALES VS. OSCILOSCOPIOS ANÁLOGOS

Los osciloscopios digitales tienen 2 ventajas principales sobre los osciloscopios análogos:

¹ Tomado de: <http://www.ucm.es/info/electron/laboratorio/instrumentos/osc/osc.html>

- a) La habilidad de mostrar señales de baja y muy baja frecuencia. Las señales en el rango de 10 a 100 Hz son muy difíciles de apreciar en un osciloscopio análogo tradicional debido al parpadeo del trazo y la corta persistencia del haz sobre la pantalla. Las señales de menos de 10 Hz son imposibles de verse en un osciloscopio análogo. Tan pronto como el haz va formando la forma de onda, la imagen se marchita y desaparece antes de un trazo completo.

- b) La habilidad de mantener o retener una señal en memoria por largos períodos de tiempo.

Los osciloscopios analógicos tienen también ventajas sobre los osciloscopios digitales:

- a) No están limitados por la velocidad de un conversor AD como en los digitales.

- b) Son preferibles cuando es prioritario visualizar variaciones rápidas de la señal de entrada en tiempo real.

Pero los osciloscopios digitales en general tienen 2 grandes desventajas:

- a) Son caros.

- b) Al igual que los osciloscopios análogos, aunque no resulta una desventaja determinante, poseen una pantalla pequeña para apreciar las formas de onda (generalmente de 5 pulgadas). Aunque algunos poseen software para PC, y pueden conectarse a la PC, y así usan a la PC como interfaz (pero continúan con la primera desventaja, son extremadamente caros).

1.2.2 VENTAJAS DEL OSCILOSCOPIO PROPUESTO FRENTE A LOS TRADICIONALES

- a) Uso de la interfaz gráfica de una PC (monitor), con lo cual se puede apreciar de mejor manera las formas de onda.
- b) Bajo costo comparado con el altísimo costo de los osciloscopios tradicionales.
- c) Uso directo de resultados en un documento, como por ejemplo copiar y pegar un gráfico.

1.3 MECANISMOS DE DISPARO (TRIGGER)

En un osciloscopio, el mecanismo de disparo es el que permite que cualquier señal de entrada, se represente o grafique en pantalla, de forma estable o sincronizada. En palabras sencillas, es el sistema que hace que la graficación de cualquier señal en la pantalla, se inicie en su extremo izquierdo en un determinado momento para un determinado valor de la señal, inclusive cuando se cumpla con un determinado flanco de la señal. En los osciloscopios existen 3 tipos de mecanismos de disparo según la finalidad con la que se utilice. Estos son:

➤ **Mecanismo de disparo básico**

En este, el osciloscopio utiliza la misma señal que se desea visualizar como onda de referencia. Es el caso más común. Este es el trigger que usan los osciloscopios analógicos y digitales para poder mantener fija la imagen que se muestra en la pantalla.

➤ **Mecanismo de disparo de línea (o de la red eléctrica)**

En este el osciloscopio utiliza una fracción de la tensión de red eléctrica, con la que el mismo se alimenta, como onda de referencia. Este modo de disparo es

muy útil cuando las ondas a visualizar proceden de un circuito cuyo ritmo de funcionamiento viene marcado por la propia red eléctrica. Es el caso de los rectificadores monofásicos y trifásicos. Una vez seleccionada y ajustada esta fuente de sincronismo, todas las ondas que se visualicen en la pantalla (como tienen una referencia común que es la red eléctrica) estarán de por sí sincronizadas.

➤ **Disparo externo**

Los osciloscopios poseen un conector especial llamado típicamente "*trig in*". Está concebido para introducir por él una señal específica de disparo, que será utilizada como onda de referencia. La aplicación es para "capturar" ondas esporádicas o no repetitivas. Esta técnica de medida requiere de circuitos auxiliares (externos al osciloscopio) que fabriquen la señal de sincronismo externo cuando se produzca el evento a visualizar.

1.4 TIPOS DE MUESTREO

1.4.1 TIEMPO REAL

Este método de muestreo es el más simple de todos y es el único que permite digitalizar completamente señales no periódicas y transitorias.

Cada muestra y el tiempo en que fue tomada tienen una correspondencia directa con su equivalente en tiempo real.

A mayor tasa de muestreo en comparación al ancho de banda de la señal, se obtiene una mayor definición en el resultado.

Nyquist desarrolló un teorema que dice que para reconstruir una señal de frecuencia F_s , se debe muestrear a una tasa mayor a $2F_s$. Sin embargo, esto no se aplica tan directamente como parece. Esta teoría se aplica solamente a señales de ancho de banda limitado que no contienen ningún componente sobre la frecuencia F_s , pero los bordes rápidos de las señales digitales de alta

velocidad pueden contener armónicos significativos sobre sus frecuencias fundamentales. Mas aún, no importa cuan rápido se muestree, nunca se podrá recomponer esta señal a la perfección.

Sin embargo, en la práctica se ha encontrado que muestreando a una velocidad cuatro veces mayor que la mayor componente de frecuencia de la señal, se puede obtener un resultado muy confiable y con bajo error.

Si se muestrea a una frecuencia menor a $4F_s$ pero mayor a $2F_s$, se obtiene un resultado más lejano al ideal, y cuanto más se acerque al limite de $2F_s$, más errores contendrá la reconstrucción. Así mismo, si se pasa por debajo del umbral de $2F_s$ para la frecuencia de muestreo, el fenómeno conocido como *aliasing* es inevitable¹.

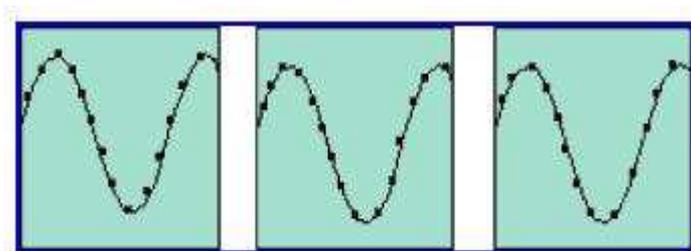


Figura 1.3 Método de muestreo en tiempo real.

- Ventajas:
 - La única opción para una medida correcta de señales no periódicas.
- Desventajas:
 - Ancho de banda relacionado directamente con la tasa de muestreo.
 - Susceptible al *aliasing* a velocidades de muestreo lentas.

¹ Tomado de:

<http://tritium.fis.unb.br/Fis3Exp/www.tmo.hp.com/tmo/pia/BasicInstrument/TUTnBRIEF/English/BI-T-Sampling-010.html>

1.4.2 TIEMPO EQUIVALENTE

El muestreo en tiempo equivalente cuenta con las siguientes características:

- $F_m > 2F_s$
- Sirve solo para señales periódicas
- la señal se va construyendo en barridos sucesivos

Además abarca los siguientes tipos de muestreo: aleatorio repetitivo, secuencial y sub muestreo.

1.4.2.1 Muestreo aleatorio repetitivo

Este tipo de muestreo se utiliza para aumentar la frecuencia máxima de medición de la señal de entrada.

El muestreo aleatorio repetitivo captura datos sobre la forma de onda adquiriendo puntos en más de una ocurrencia del trigger. Esto significa que la forma de onda en sí misma debe ser repetitiva (periódica) y no un acontecimiento transitorio, puesto que la tasa de muestreo en estos casos es típicamente demasiado baja para señales transitorias de alta frecuencia.

En cada ocurrencia del disparador se adquieren algunos puntos de referencia, luego todos los puntos muestreados se juntan en un cuadro compuesto de la forma de onda. Cada punto es puesto en su lugar apropiado midiendo el tiempo transcurrido entre el trigger y la propia muestra.

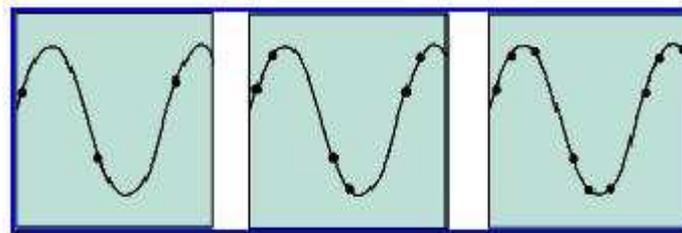


Figura 1.4 Método de muestreo aleatorio repetitivo.

A medida que se muestrea más ciclos, más muestras se obtiene, y luego son ubicadas en un mismo ciclo pero en la posición correspondiente medida a partir del trigger.

- Ventajas:
 - Ofrece mayor ancho de banda que el muestreo en tiempo real.
 - No es susceptible al *aliasing* en señales repetitivas.
- Desventajas:
 - No es apto para mediciones de señales no periódicas y de alta velocidad.

1.4.2.2 Muestreo secuencial

Los capturadores digitales de gran ancho de banda tienden a utilizar el muestreo secuencial.

Este método captura una muestra por ciclo (o cada x cantidad de ciclos) pero con un determinado tiempo muy exacto entre el disparador y el punto de captura.

Para este tipo de muestreo es necesario, al igual que en el caso anterior, una señal periódica. Cada muestra es tomada pasado cierto intervalo de tiempo luego de disparado el *trigger*. Para la siguiente captura, el intervalo de espera es incrementado, y por lo tanto dicha muestra es tomada un instante de tiempo después en el ciclo que la muestra anterior.

Los puntos son tomados siempre en puntos diferentes del ciclo de la señal de entrada, sin importar en qué ciclo fue tomada la muestra. De este modo, al finalizar la captura de todas las muestras, cada una de ellas es posicionada en un único ciclo, pero en la posición que le corresponde a partir del tiempo transcurrido desde el *trigger*.

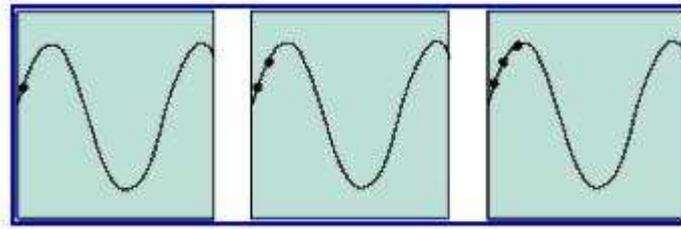


Figura 1.5 Método de muestreo secuencial.

- Ventajas:
 - Ofrece el mayor ancho de banda disponible (hasta 50Ghz).
 - Muy bajo ruido.
 - puede utilizar un A/D lento y de alta precisión.

- Desventajas:
 - No puede capturar eventos previos al *trigger* e incluso pasado un mínimo tiempo tras el.
 - Susceptible al *aliasing* a velocidades de muestreo lentas.
 - No puede medir transitorios.

1.4.2.3 Submuestreo

Se conoce que el teorema de Nyquist dice que la frecuencia de muestreo tiene que ser mayor al doble de la frecuencia máxima de la señal a muestrear. Sin embargo muchos confunden *ancho de banda* con *frecuencia máxima*. El error aquí es que una señal contenida en los 100KHz, puede tener un ancho de banda de 1KHz. De este modo, según el teorema de Nyquist bien aplicado, se podría muestrear dicha señal a una frecuencia mayor a 2KHz (la que es mucho menor a 200KHz). En la Figura 1.6 se muestra gráficamente cómo se superponen los espectros de señales de gran ancho de banda muestreadas a una frecuencia menor a su ancho de banda. Este método de explicación es conocido como el método *fanfold*.

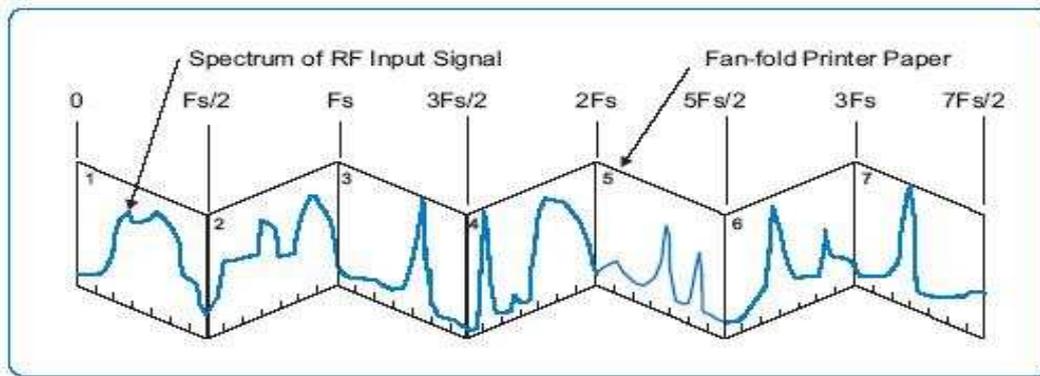


Figura 1.6 Espectro completo de una señal¹.

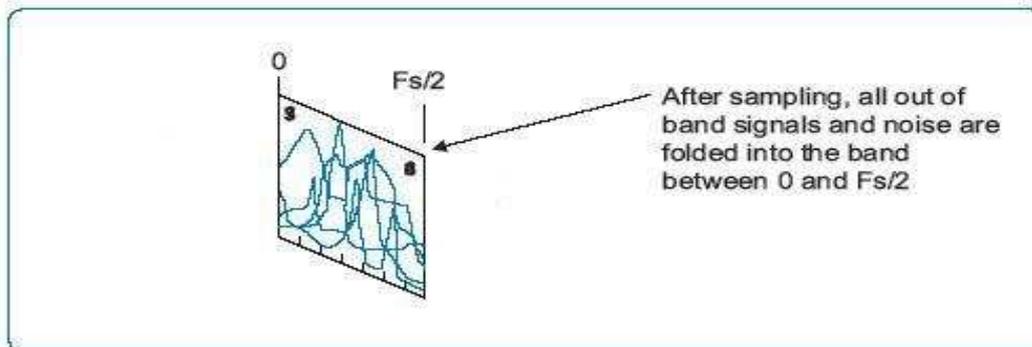


Figura 1.7 Efecto del aliasing².

Ahora se toma por ejemplo el caso de una señal de ancho de banda acotado, y centrado en una frecuencia alta. A su vez se hace que pase por filtros para estar seguros de que se evita el *aliasing*, entonces se tiene un resultado fiable de la señal muestreada.

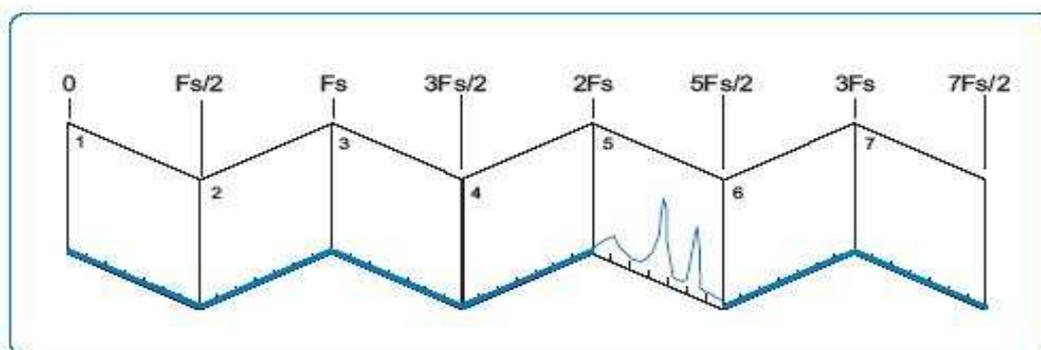


Figura 1.8 Espectro de la señal filtrada³.

¹ Tomado de: www.pentek.com/deliver/TechDoc.cfm/PutUndersamp.pdf

² Tomado de: www.pentek.com/deliver/TechDoc.cfm/PutUndersamp.pdf

³ Tomado de: www.pentek.com/deliver/TechDoc.cfm/PutUndersamp.pdf

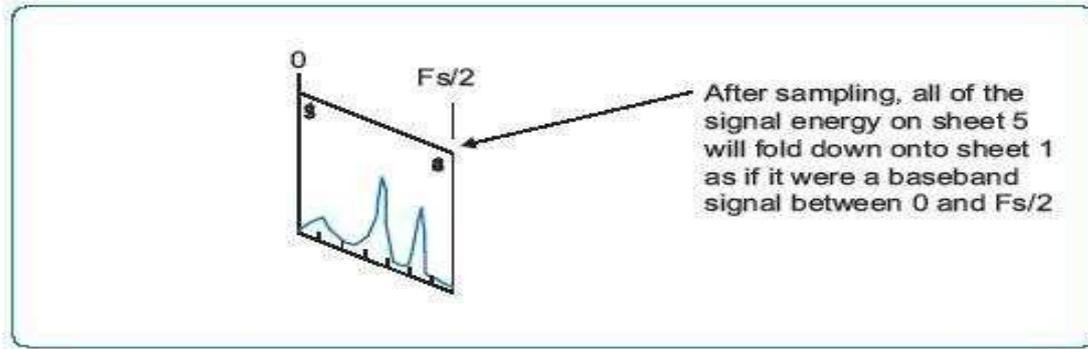


Figura 1.9 La señal muestreada no es afectada¹.

- **Ventajas:**
 - Permite medir señales de alta frecuencia con conversores de baja frecuencia.
- **Desventajas:**
 - La señal debe ser de ancho de banda acotado.
 - Necesidad de filtrado *anti-aliasing*.
 - No apto para medición de grandes anchos de banda.
 - Carece de sentido en señales de banda base.

1.5 MÉTODOS DE CONVERSIÓN Y TRANSFERENCIA

Los métodos de conversión y transferencia disponibles son los siguientes:

1.5.1 DE MANERA CONTINUA

La señal se va muestreando y procesando de manera continua y sin interrupciones contiene las siguientes características.

- **Cuello de botella:** tasa de muestreo (*sample rate*) del ADC, y velocidad de transferencia hacia el ordenador.
- **Cuando y porque se usa:**

¹ Tomado de: www.pentek.com/deliver/TechDoc.cfm/PutUndersamp.pdf

- para medir transitorios (es la única forma de poder medirlos).
- para medir señales lentas (periódicas y no periódicas).

1.5.2 POR RÁFAGAS

El procesamiento de conversión y transferencia por ráfagas sigue los siguientes pasos:

- a. Se toman muestras de la señal rápidamente hasta llenar un buffer.
- b. Se detiene el proceso de muestreo mientras se procesan (lentamente) los datos.
- c. Se vuelve a repetir el proceso.

Este método posee las siguientes limitantes:

- Cuello de botella: velocidad del microprocesador, velocidad transmisión por el bus serie/USB.
- Cuando y porque se usa:
 - se usa cuando el cuello de botella NO está en el ADC sino en otra etapa posterior (velocidad del microprocesador, transmisión placa-PC, etc.).
 - se usa para medir señales periódicas rápidas.
 - se puede usar también para medir señales muy lentas (tanto periódicas como no periódicas).

1.6 ERRORES QUE SUCEDEN DURANTE EL MUESTREO

Estos errores son parte de la etapa de conversión de la señal analógica en una representación digital de la misma. A continuación se presentan los diferentes tipos de errores que se encuentran en este proceso.

1.6.1 ERROR DE COMPENSACIÓN

El error de compensación está definido como la diferencia entre la compensación nominal y la que realmente se tiene. Para el caso de un conversor analógico-digital, el punto de compensación nominal es el valor que se tiene en la entrada para obtener una salida igual a cero.

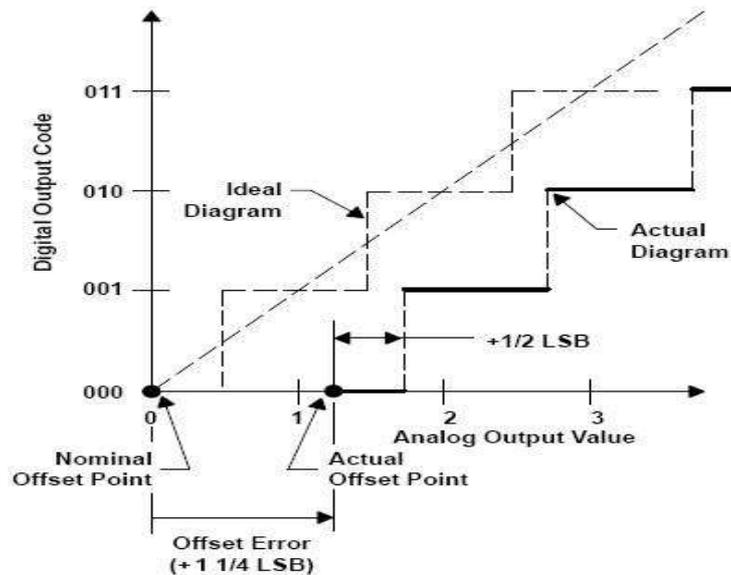


Figura 1.10 Error de compensación¹.

Cuando en la entrada se coloca este supuesto valor y en su salida se obtiene un valor diferente de cero, es justamente cuando existe un error de compensación. Esto bien puede corregirse ajustando los valores de compensación, y si no es posible ajustarlos, entonces se puede realizar las etapas de análisis de los datos obtenidos, haciendo los ajustes necesarios en estos valores.

1.6.2 ERROR DE GANANCIA

El error de ganancia se define como la diferencia entre el valor de ganancia nominal (ganancia esperada, calculada) y la ganancia real. Este valor se calcula cuando el error por compensación es nulo, caso contrario se obtendrá un error en un cálculo generado por otro error.

¹ Tomado de: <http://focus.ti.com/lit/an/slaa013/slaa013.pdf>

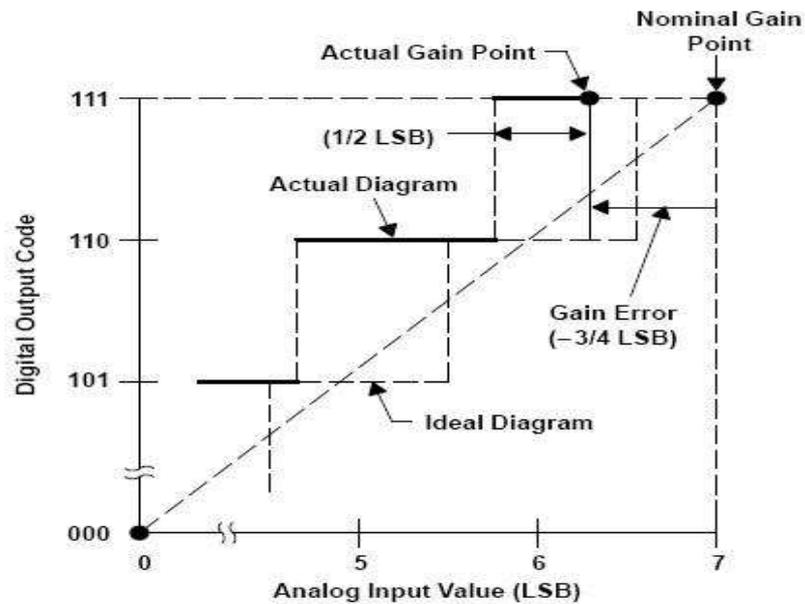


Figura 1.11 Error de ganancia¹.

La Figura 1.11 muestra de forma simple el significado de este error. Cuando se tiene un valor conocido a la entrada del convertidor, se espera un valor de salida determinado. Cuando la diferencia entre el valor esperado y el obtenido es constante e igual sin importar cual sea el valor en su entrada, entonces se trata de un error de ganancia. Este error es lineal y constante en todo el rango posible de entrada. Este error también es posible solucionarlo, aunque también se puede corregir en etapas posteriores.

1.6.3 ERROR DE APERTURA

El error de apertura es un tipo de error generado en los propios componentes y no es corregible. Se refiere a la incertidumbre en la señal de entrada en el momento en que se muestrea y retiene dicha señal (*sample&hold*), está dado desde el instante en que se la muestrea hasta que se la retiene, antes de pasar al proceso de conversión. El error de apertura puede ser generado por ruidos o también por variaciones en la señal de reloj. Este error también influye en las características finales del sistema.

¹ Tomado de: <http://focus.ti.com/lit/an/slaa013/slaa013.pdf>

Este error puede ser reducido si el tiempo de muestreo y retención disminuye. Si este valor es lo más pequeño posible, entonces se tiene mayor certeza de que la conversión realizada es correcta.

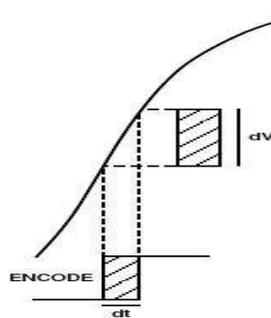


Figura 1.12 Error de apertura¹.

1.6.4 ERROR DE NO-LINEALIDAD DIFERENCIAL

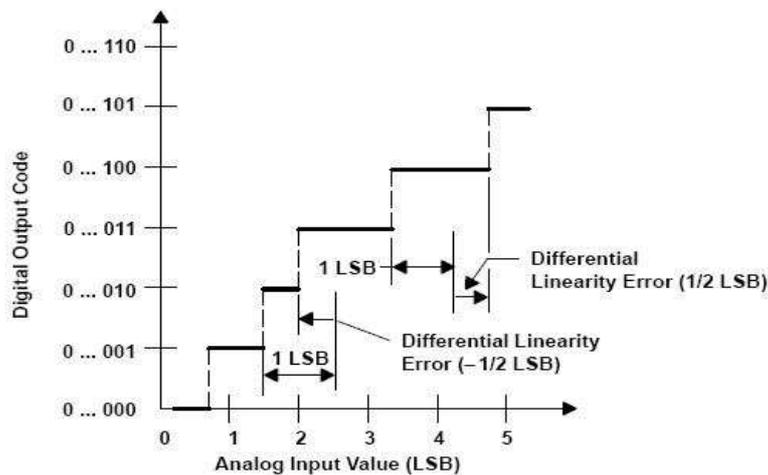


Figura 1.13 DNL Error².

El error de no-linealidad diferencial (*Differential Nonlinearity | DNL Error*) es la diferencia entre la variación de tensión nominal que genera un cambio en un bit a la salida del conversor (1 LSB) y la variación real que debe ocurrir. Es decir, que si el cambio de tensión que debe ocurrir a la entrada para que haya un cambio en la salida de un bit es exactamente el esperado, entonces el error DNL es cero. En cambio, si para que haya un cambio en la salida, el diferencial de tensión a la entrada debe ser mayor (o menor) al que se especifica, entonces existe este tipo de error. Incluso este error puede generar valores

¹ Tomado de: <http://focus.ti.com/lit/an/slaa013/slaa013.pdf>

² Tomado de: <http://focus.ti.com/lit/an/slaa013/slaa013.pdf>

binarios de salida que nunca se lleguen a dar, debido a que el rango de entrada tiene como respuesta una menor cantidad de valores debido al error citado (cuando el diferencial de tensión de entrada es mayor que el especificado para 1 LSB).

1.6.5 ERROR DE NO-LINEALIDAD INTEGRAL

El error de no-linealidad integral (Integral Nonlinearity | INL Error) está definido como la desviación de los valores de la función de transferencia real sobre una línea recta. Esta línea recta puede ser definida como la mejor aproximación que minimice estas desviaciones, o bien entre los puntos extremos de la función de transferencia una vez que se hayan anulado los errores de ganancia y compensación. Este segundo método es conocido como “linealidad de punto final”, y es el más usado ya que su verificación es más simple.

En un convertor analógico-digital esta desviación se mide en la transición de un paso al siguiente (es decir, un aumento en 1 LSB), y el nombre de error *integral* proviene de que se trata de la suma de estas desviaciones desde el nivel más bajo hacia un valor determinado. Con esta definición se puede conocer entonces el error causado por la no-linealidad integral en cada valor.

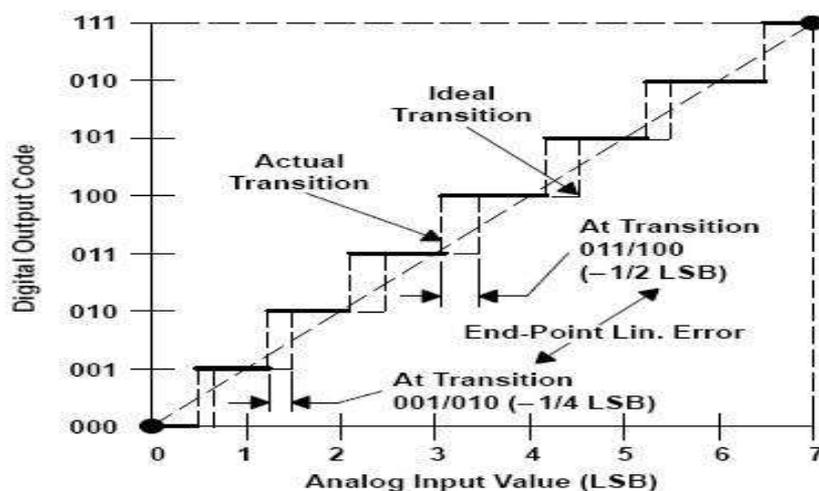


Figura 1.14 INL Error¹.

¹ Tomado de: <http://focus.ti.com/lit/an/slaa013/slaa013.pdf>

1.6.6 ERROR DE CUANTIZACIÓN

El error de cuantización esta basado en la naturaleza de las señales analógicas y las digitales. Una señal analógica es continua, y puede tener *infinitos* valores (por mas que la señal este acotada), mientras que una señal digital es discreta, tiene una cantidad *finita* de valores posibles.

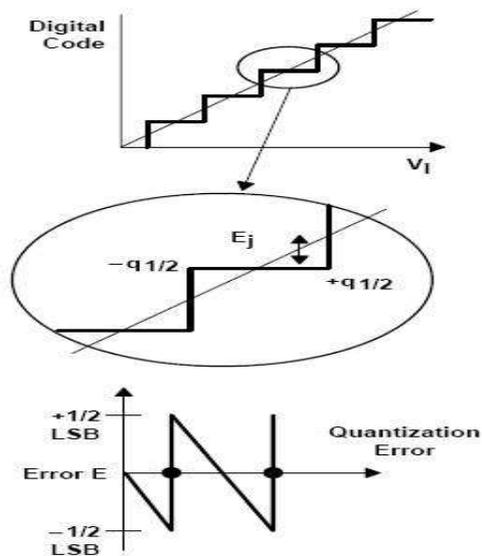


Figura 1.15 Error de cuantificación¹.

Aquí entonces radica este error. Al intentar traducir una señal que puede tener *infinitos* valores en otra que solo puede tener valores *finitos*, esta claro que se pierde información. La cantidad de valores posibles (o *estados* posibles) que puede tener la señal digital esta relacionada con la cantidad de bits con la cual ésta se representa. Sin embargo la cantidad de bits es una cantidad *finita*.

La única forma de hacer que este error sea nulo, es haciendo que la cantidad de bits con la que se representa el valor digital sea infinito. Esto implica que cierto valor digital representa a *muchos* (de hecho, infinitos) valores analógicos posibles. De esta forma, un valor analógico produce una salida digital, y esa salida digital si se vuelve a convertir a un valor analógico, puede que no corresponda con el valor original. Cierta información se ha perdido en este proceso, y esto se conoce como error de cuantización.

¹ Tomado de: <http://focus.ti.com/lit/an/slaa013/slaa013.pdf>

En el caso de un conversor ideal, donde la función de salida puede creerse como una *escalera perfecta*, el error entre la señal real de entrada y su correspondencia con la salida digital, tiene una función de densidad de probabilidad uniforme en el caso de que se asuma que la entrada es totalmente aleatoria.

Este error varía en el rango de $\pm 1/2$ LSB, o bien, $\pm q/2$, donde q es el ancho de un escalón, tal como se muestra en la Figura 1.15 siguiente.

1.6.7 ERROR ABSOLUTO

El error absoluto en la exactitud del proceso de conversión es la diferencia máxima que se encuentra entre el valor analógico de la señal de entrada con el valor medio que se define para ese mismo código binario de salida.

Este error es absoluto, y es por ello que incluye a todos los errores citados previamente (compensación, ganancia, no linealidad, e incluso cuantización).

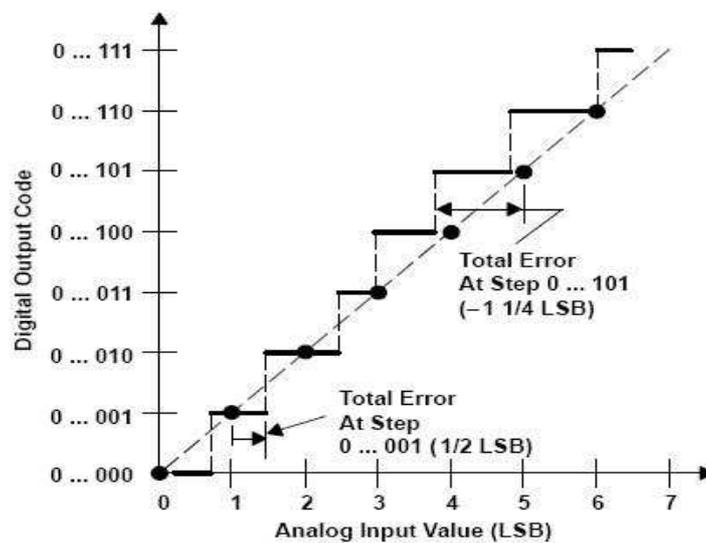


Figura 1.16 Error absoluto¹.

Más adelante en el capítulo de pruebas se verificará la presencia o no de estos errores, principalmente considerando que el conversor del dsPIC utiliza el método de conversión de "Aproximaciones sucesivas".

¹ Tomado de: <http://focus.ti.com/lit/an/slaa013/slaa013.pdf>

1.7 MODO DE COMUNICACIÓN

Hoy día resulta muy interesante observar como los avances tecnológicos sorprenden por la evolución tan rápida que presentan y algo que gusta es que cada vez son más fáciles de usar para cualquier persona; es decir, se están volviendo muy amigables y no se necesita ser un experto para poder comprender su funcionamiento, usarlos o instalarlos. Este es el caso de Universal Serial Bus, mejor conocido como USB.

El USB se dio a conocer debido sobre todo a la lista de tecnologías contenidas en Windows 98, es un nuevo bus que permite conectar una serie de dispositivos con una velocidad de datos que varía de baja a media-rápida.

Lo mejor de todo es la variedad de dispositivos que se pueden conectar a estos. Esto incluye discos duros, DVD-ROMs y CD-ROMs de alta velocidad, impresoras, escáneres... y la novedad: cámaras de fotos digitales, videocámaras DV, televisiones... teclados, mouses y un sin fin de dispositivos. De hecho, ya hay disponibles muchos elementos.

1.7.1 USB (Universal Serial Bus)

En un principio se tenía la interfaz serial y paralela, pero era necesario unificar todos los conectores creando uno más sencillo y de mayores prestaciones. Así nació el USB (Universal Serial Bus) con una velocidad de 12Mb/s. y su evolución, USB 2.0, apodado USB de alta velocidad, con velocidades en este momento de hasta 480Mb/s; es decir, 40 veces más rápido que las conexiones mediante cables USB 1.1.

Un puerto USB puede llegar a transmitir a velocidades entre 1,5 Mbps y 12 Mbps; un puerto paralelo entre 600 Kb/s a 1,5 Mb/s y un puerto serial puede llegar hasta 112 Kb/s. USB es una nueva arquitectura de bus o un nuevo tipo de bus desarrollado por un grupo de siete empresas (Compaq, Digital Equipment Corp, IBM PC Co., Intel, Microsoft, NEC y Northern Telecom) que forma parte de los avances plug-and-play y permite instalar periféricos sin tener

que abrir la máquina para instalar el hardware. Trabaja como interfaz para transmisión de datos y distribución de energía, que ha sido introducida en el mercado de PC's y periféricos para mejorar las lentas interfaces serie (RS-232) y paralelo. Esta interfaz de 4 hilos, distribuye 5V para alimentación.

USB se basa en el paso de un testigo, semejante a otros buses como los de las redes locales en anillo con paso de testigo, el controlador USB distribuye testigos por el bus. El dispositivo cuya dirección coincide con la que porta el testigo responde aceptando o enviando datos al controlador. Este también gestiona la distribución de energía a los periféricos que lo requieran. Emplea una topología de estrellas apiladas que permite el funcionamiento simultáneo de 127 dispositivos a la vez. En la raíz o vértice de las capas está el controlador anfitrión o host que controla todo el tráfico que circula por el bus. Esta topología permite a muchos dispositivos conectarse a un único bus lógico sin que los dispositivos que se encuentran más abajo en la pirámide sufran retardo. A diferencia de otras arquitecturas, USB no es un bus de almacenamiento y envío, de forma que no se produce retardo en el envío de un paquete de datos hacia capas inferiores, ya que cada puerto utiliza una única solicitud de interrupción (IRQ) independientemente de los periféricos que tenga conectados (sea 1 ó 127) por lo tanto no hay riesgo de conflictos entre una cantidad de dispositivos que de otra forma no podrían ser conectados por falta de recursos; de la misma manera tampoco utilizan DMA (asignación de memoria).

1.7.1.1 Tipos de transferencia

Cuando un driver del dispositivo en el *host* desea comunicarse con un dispositivo, inicia una transferencia. Una transferencia puede ser tan corta como un byte de datos o tan larga como el contenido de un archivo.

Las tipos de transferencia de datos que maneja USB son:

- Transferencias de control
- Transferencias Bulk

- Transferencias de Interrupción
- Transferencias Isócronas

| Tipos de Transferencia | Máxima tasa de transferencia de datos por <i>endpoint</i> (kilobytes/seg. de carga útil transferida) | | | Etapa (cero o más transacciones) | Fases |
|------------------------|--|-------------------|-------------------|----------------------------------|------------------|
| | <i>Low speed</i> | <i>Full speed</i> | <i>High speed</i> | | |
| Control | 24 | 832 | 15872 | Setup | <i>Token</i> |
| | | | | | Datos |
| | | | | | <i>Handshake</i> |
| | | | | Datos (IN o OUT) (opcional) | <i>Token</i> |
| | | | | | Datos |
| | | | | | <i>Handshake</i> |
| | | | | Estado (IN o OUT) | <i>Token</i> |
| | | | | | Datos |
| | | | | | <i>Handshake</i> |
| Bulk | 0.8 | 64 | 24576 | Datos (IN oOUT) | <i>Token</i> |
| | | | | | Datos |
| | | | | | <i>Handshake</i> |
| Interrupción | No permitido | 1216 | 53248 | Datos (IN o OUT) | <i>Token</i> |
| | | | | | Datos |
| | | | | | <i>Handshake</i> |
| Isócrona | No permitido | 1023 | 24576 | Datos (IN o OUT) | <i>Token</i> |
| | | | | | Datos |

Tabla 1.1 Índices máximos y fases de los cuatro tipos de transferencia.

Para este proyecto se aplica la transferencia de datos isócrona.

1.7.1.1.1 Transferencia de datos isócrona

Son transferencias que se realizan en tiempo real utilizadas cuando los datos tienen que llegar a una tasa constante o en un determinado tiempo, y puede tolerar errores ocasionales. Sus aplicaciones más comunes son voz y música codificadas y reproducidas en tiempo real. Aunque puede enviar un número fijo

de bytes por trama, los datos no se transfieren en un número constante de bits por segundo. Aunque en este caso no resulta conveniente para la apreciación gráfica de las señales utilizar este método, se vio la necesidad de aplicarlo, debido a las limitantes en velocidad que tiene el sistema principalmente en cuanto a la conversión de la señales.

1.7.1.2 Cables y conectores.

USB 1.1 transfiere señales y energía a los periféricos utilizando un cable de 4 hilos, apantallado para transmisiones a 12 Mbps y no apantallado para transmisiones a 1.5 Mbps a una longitud máxima de 5 metros.

En lo que respecta a los conectores son de dos tipos: serie A y serie B. Los primeros presentan las cuatro patillas correspondientes a los cuatro conductores alineadas en un plano. Los receptáculos se presentan en cuatro variantes: vertical, en ángulo recto, panel y apilado en ángulo recto así como para montaje pasamuros. Se emplean en aquellos dispositivos en los que el cable externo, está permanentemente unido a los mismos, tales como teclados, ratones, y hubs o concentradores.

Los conectores de la serie B presentan los contactos distribuidos en dos planos paralelos, dos en cada plano, y se emplean en los dispositivos que deban tener un receptáculo al que poder conectar un cable USB. Por ejemplo impresoras, scanner, y módems.

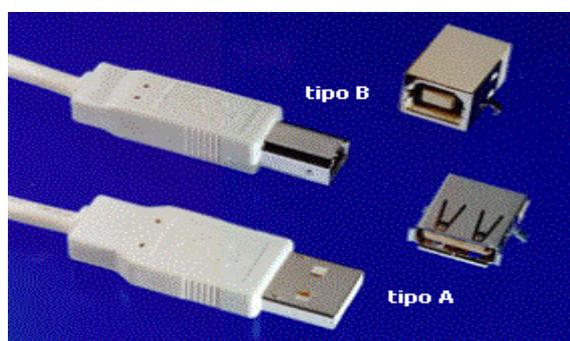


Figura 1.17 Conectores de la serie A y la serie B.

1.7.1.3 Sustitución de los puertos serie y paralelo

Para aplicaciones exigentes los 12 Mbps no fueron suficientes, por lo que a mediados del 2001 se presentó la nueva versión USB 2.0, con una velocidad de transferencia de 480 Mbps. Esta nueva versión es compatible con la versión anterior. Si se tienen dispositivos USB 1.1, no hay problema en conectarlos al puerto USB 2.0.

1.7.1.4 Ventajas de utilizar el interfaz USB

- a) Las velocidades del BUS que soporta la interfaz USB son: *High speed* (Alta velocidad) de 480 Megabits por segundo. *Full speed* (velocidad completa) en 12 Megabits por segundo y *Low speed* (Baja velocidad) en 1.5 Megabits por segundo.
- b) La interfaz USB maneja la señal de Vcc y tierra, proveniente de la fuente del computador o de un concentrador. Un periférico que requiere hasta 500 mA, puede obtener su energía del puerto USB.
- c) Un dispositivo con una interfaz USB es probable que tenga igual o menor costo que su equivalente con una interfaz serial RS232.
- d) Las especificaciones de Hardware para los controladores USB, receptores y los cables eliminan la mayor parte del ruido que podría producirse permitiendo la detección de los errores en los datos y se proceda a la notificación al transmisor para que se proceda con la retransmisión.
- e) Un sistema operativo que soporta USB debe hacer tres cosas:
 - 1. Detectar cuando un dispositivo se conecta o se retira del sistema.

2. Proporcionar un mecanismo que permita a los controladores del software comunicarse con el hardware del USB del computador, y las aplicaciones a las cuales deben tener acceso los periféricos USB.
3. Si el sistema operativo no incluye el driver de un periférico específico, el vendedor está en la obligación de proporcionar uno.

1.7.1.5 Desventajas de utilizar el interfaz USB.

- USB 2.0 es competitivo con la interfaz IEEE-1394 (firewire) de 400 Mbps, pero IEEE-1394b es una interfaz más rápida, puede operar a velocidades de 3.2 Gbps.
- El cable puede tener una distancia máxima de 5 metros. Otras interfaces, tales como RS-232, RS-485, y Ethernet, permiten cables mucho más largos. Se puede aumentar la longitud de un enlace USB hasta 30 metros, utilizando cables que enlacen 5 concentradores; es decir 6 segmentos de cable de 5 metros por cada uno.
- Para programar un periférico USB hay que conocer acerca de los protocolos USB.

A pesar de ser más rápida, no se seleccionó la interfaz IEEE-1394, debido a que las PC's no poseen dicha interfaz, y en contraposición, la interfaz USB 2.0 se puede encontrar actualmente en cualquier PC.

1.8 PROPUESTA DE DISEÑO

Este proyecto intenta lograr la misma funcionalidad de un osciloscopio tradicional, usando un microcontrolador dsPIC30F3011 para la adquisición de datos (incluyendo la circuitería analógica apropiada) y transferir los datos al PC (vía USB). Una aplicación desarrollada en lenguaje Visual Basic bajo el ambiente de Windows desplegará las formas de onda como se verían en un osciloscopio tradicional. Este osciloscopio podrá mostrar formas de onda de

baja frecuencia en tiempo real, pero para frecuencias más altas, será necesario leer un número finito de muestras y almacenarlas en la memoria RAM de dsPIC. Se llena la RAM, se descarga todo a la PC, inmediatamente se vuelve a llenar la RAM, se descarga, y así cíclicamente. La desventaja evidente es la de no poder monitorear de forma continua la señal de entrada, pero como la aplicación básica del osciloscopio es mostrar formas de onda periódicas, esto resulta totalmente irrelevante. Pero de tratarse de formas de onda **no periódicas**, esto sí resultaría un problema.

Una de las tareas realizadas en la etapa de análisis fue la de definir las áreas que se tendrían que abordar, donde la más importante (y la que se tuvo que definir primero) fue la del hardware digital de **“control y comunicación”** puesto que allí se define la arquitectura del sistema y también el programa del microcontrolador (firmware) que administrará todo el funcionamiento del osciloscopio. De ésta depende todo el resto del diseño.

Otra etapa no menos importante es la del **“acondicionamiento de las señales de voltaje de entrada”**. Esta etapa consistió del diseño y especificación de componentes a usar en la etapa de entrada para adaptación, protección y acondicionamiento.

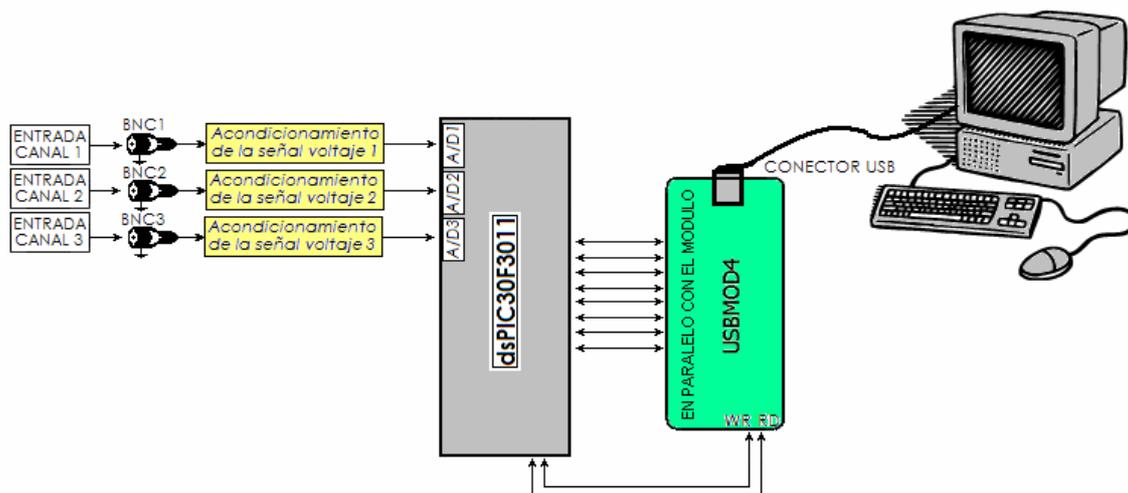


Figura 1.18 Diagrama esquemático del presente proyecto.

Además de todo lo anterior, se debe tener un HMI en la PC, para lo cual se requiere el diseño de una interfaz gráfica, desde la cual se controle todos los parámetros del osciloscopio.

Se debe también especificar el protocolo de comunicación a usar entre el firmware y el software, para cumplir con un diseño eficiente de osciloscopio.

En la Figura 1.18 se presenta un diagrama de bloques inicial del osciloscopio. Las partes que conforman el osciloscopio con sus respectivas funciones, se presentan a continuación:

- Fuente de poder para alimentación de toda la circuitería del proyecto con las debidas protecciones.
- Circuitería análoga para acondicionamiento de las señales de entrada de cada canal, incluyendo las debidas protecciones.
- Hardware para adquisición de datos, control y comunicación. La adquisición de datos a través del conversor análogo-digital, la administración y control se manejará con el firmware dsPIC30F3011 y se utilizará el puerto USB, con la ayuda del módulo integrado de transferencia de datos USBMOD4.
- Protocolo de comunicación para el manejo de los datos desde el dsPIC hacia la PC.

Lo descrito anteriormente es la idea básica de donde se partirá para el diseño del hardware y software del osciloscopio propuesto. Teniendo ya esta guía, los requerimientos, diseño e implementación se describen detalladamente en los siguientes capítulos.

CAPÍTULO 2

DISEÑO Y CONSTRUCCIÓN DEL HARDWARE DEL OSCILOSCOPIO

DISEÑO Y CONSTRUCCIÓN DEL HARDWARE DEL OSCILOSCOPIO

2.1 GENERALIDADES

Este capítulo describe los requerimientos previos, diseño y construcción del osciloscopio para PC, que incluye las protecciones, el acondicionamiento de las entradas de cada canal, la etapa de filtrado, selección de escalas de hardware, selección de acoplamiento, el control y la comunicación, y, finalmente, la fuente de alimentación.

Las protecciones se diseñarán en base a los valores de voltaje y corriente máximos que puedan soportar los elementos constitutivos de toda la etapa de acondicionamiento y filtrado de cada canal.

En el osciloscopio se podrán medir señales de voltaje de -200 V pico negativo como mínimo, 200 V pico positivo como máximo, y frecuencias de hasta 40 kHz.

La comunicación entre el módulo y la PC será solamente vía puerto USB 2.0, por ser el puerto actualmente más usado como interfaz para interconectar cualquier tipo de dispositivo electrónico.

La fuente es externa. No se podrá aprovechar la alimentación que provee el puerto USB de la PC, debido a las limitantes en cuanto a potencia que este presenta, que está por debajo de lo que necesita el módulo del osciloscopio.

Después de esta corta referencia de las características que va a tener el hardware del osciloscopio, a continuación se analiza todo lo referente al diseño del mismo.

2.2 DISEÑO DE LAS PROTECCIONES, ACONDICIONAMIENTO Y FILTRADO EN CADA CANAL

A continuación se detallan los requerimientos, diseño e implementación de todo el circuito de entrada, acondicionamiento y filtro de un canal. Todo este diseño es igual para los tres canales.

2.2.1 CIRCUITO DE ENTRADA DE UN CANAL

Los requerimientos a la entrada de cada canal, son básicamente de **protección**. La etapa de entrada de cada canal consiste en un fusible de 0.75 amperios, para proteger contra sobrecorriente, y un varistor de 220V para proteger contra sobrevoltaje, evitando así algún daño en los dispositivos integrados que conforman la etapa de acondicionamiento y filtro *anti-aliasing*.

Como se puede apreciar en la Figura 2.1, cada canal físicamente está constituido por un conector BNC al que se han conectado dos resistencias de 10K Ω / 5Wattios de potencia, en serie, y un potenciómetro de 100K Ω en paralelo, los cuales conforman un divisor de voltaje y corriente para limitar la señal de entrada hasta 176 Vrms de onda senoidal, especificada en el alcance de este proyecto.

Si se llama V_{in} al voltaje de entrada del sistema, y si el potenciómetro (100 [k Ω]) de la entrada se lo calibra a la mitad, es decir, en 50 [k Ω], se tiene

que:
$$V_{RVar1} = \frac{V_{in}}{2}.$$

De donde la protección contra sobrevoltaje será:

$$V_{pico} = 120 [V] \times \sqrt{2} = 169.705 [V];$$

$$V_{pico_DIM} = (1.25) \times V_{pico} = (1.25) \times 169.705 [V]; \quad \boxed{V_{pico_DIM} = 212.1 [V]}$$

Entonces se escogió $\boxed{\text{un Varistor de } 220 [V]}$

De la Tabla 2.1, donde se detalla el consumo máximo de corriente (dado por los fabricantes) de los elementos más significativos para la tarjeta de un canal, se tiene que el consumo máximo de corriente permitido para la etapa siguiente al circuito de protección de entrada es de $840 [mA]$.

| ELEMENTO | CONSUMO DE CORRIENTE POR UNIDAD | | CANTIDAD | CONSUMO DE CORRIENTE TOTAL | |
|--------------|---------------------------------|--------|----------|----------------------------|--------|
| | mA | | | mA | |
| | TÍPICA | MÁXIMA | | TÍPICA | MÁXIMA |
| MAX477 | 100 | 120 | 5 | 500 | 600 |
| 74HC4066 | 20 | 20 | 2 | 40 | 40 |
| Relés de 5V | 100 | 100 | 2 | 200 | 200 |
| TOTAL | | | | 740 | 840 |

Tabla 2.1 Consumo máximo para la tarjeta de un canal.

Calculando el valor de la corriente en la mitad inferior del potenciómetro de $100 [k\Omega]$, Para el caso de máxima caída de voltaje pico ($110 [V]$), se tiene que es bastante despreciable, apenas $2.2 [mA]$, entonces se puede asumir que la corriente en la mitad superior de dicho potenciómetro es de $840 [mA]$.

La corriente (I_w) en la rama de las resistencias de $10 [k\Omega]/5 [W]$ máximo debe llegar a:

$$P_w = I_w^2 \cdot R_w ; \quad I_w = \sqrt{\frac{P_w}{R_w}} = \sqrt{\frac{5 [W]}{20000 [\Omega]}} ; \quad \underline{I_w = 15.8 [mA]}$$

Este también es un valor despreciable.

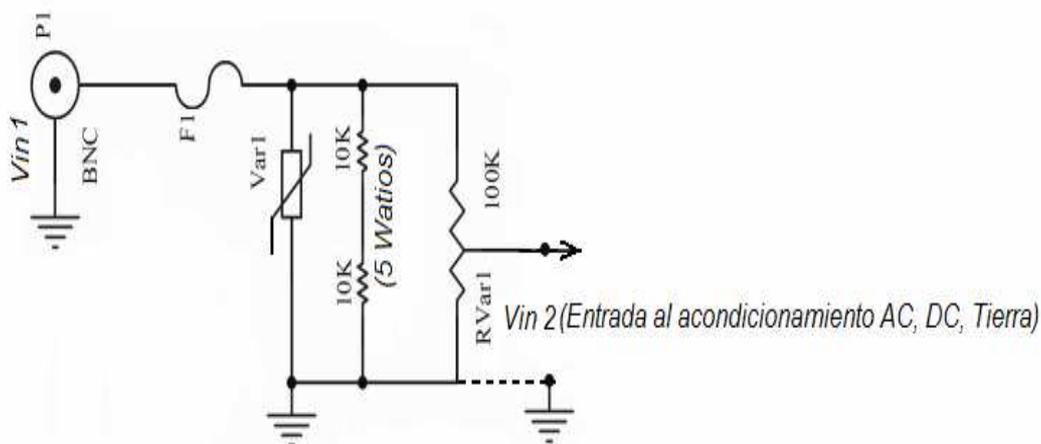


Figura 2.1 Circuito de Protecciones de entrada.

Finalmente la corriente sobre el varistor también se puede considerar despreciable, porque en condiciones de funcionamiento, sin sobre-voltajes, su resistencia es relativamente alta.

Por todo lo expuesto anteriormente, la máxima corriente que puede soportar la tarjeta de un canal a su entrada es de 840 [mA] . Entonces el **fusible** que se escogió para proteger la entrada de cada una de estas tarjetas es de 0.75 [A] .

2.2.2 DISEÑO DEL CIRCUITO DE CONTROL DE RELÉS QUE ACCIONAN LOS ACONDICIONAMIENTOS DE VOLTAJE Y LOS ACOPLAMIENTOS

El dsPIC mediante dos de sus salidas controla la selección de los circuitos de acondicionamiento y con otras dos salidas selecciona el tipo de accionamiento.

Las 2 salidas (2 bits) del dsPIC que controlan la selección del circuito de acondicionamiento ingresan a un DECODER/DEMUX, el que genera 4 salidas con estados lógicos mutuamente excluyentes. Éstas se conectan a las entradas de control de un juego de 4 interruptores bilaterales o también llamados llaves analógicas. De esta forma solo uno de los 4 interruptores se activará a la vez, y este accionará solo el relé correspondiente a la escala de voltaje escogida. Este mismo circuito y la misma lógica de funcionamiento se aplicará para la selección del acoplamiento (AC, DC o GND) en la entrada del circuito acondicionador de cada canal.

Los dos circuitos integrados más importantes que gestionan estas selecciones de acondicionamientos y acoplamientos, son: el decodificador binario 74HC139 y los interruptores bilaterales 74HC4066, cuyas características y función en el presente proyecto se describen a continuación:

2.2.2.1 Decodificador binario 74HCT139

Este decodificador binario cumple 2 funciones básicas: decodificar sus dos entradas binarias representadas en un número de 2 bits, en un juego de 4 salidas digitales de control. Adicionalmente (por la naturaleza de un decodificador), estos bits son mutuamente excluyentes, lo cual es necesario para el control de los interruptores bilaterales. Es necesario utilizar inversores después de estas salidas ya que el decodificador entrega una sola de sus salidas en estado activo, pero en cero voltios, es decir son salidas *activo-bajas*. Mientras que el integrado de interruptores bilaterales utiliza en sus entradas una lógica *activo-alta*. El inversor en cuestión, el 74HC240, tiene 4 entradas y 4 salidas (8 pares de entradas/salidas).

2.2.2.2 Interruptores bilaterales 74HC4066

El circuito integrado 74HC4066 contiene 4 interruptores bilaterales. Cada uno de estos consta de un par de transistores que permiten o no la comunicación directa entre 2 de sus salidas y es habilitado mediante una entrada independiente para cada uno de estos circuitos.

Cuando este interruptor se encuentra habilitado, su resistencia es de aproximadamente 50Ω , mientras que cuando se encuentra deshabilitado, su resistencia tiende a infinito. Todas estas compuertas son independientes, y al ser controladas por señales generadas en las salidas de un decodificador, nunca se seleccionarán más de un circuito acondicionador, ni tampoco más de un tipo de acoplamiento simultáneamente.

Un detalle a tener en cuenta es que los voltajes en sus entradas/salidas no pueden superar por mucho a V_{cc} ni hacerse negativos. Esto implicaría una conducción forzada de los transistores.

2.2.3 DISEÑO DE CIRCUITOS DE ACCIONAMIENTO DE ACONDICIONAMIENTOS DE VOLTAJE Y ACOPLAMIENTOS

Estos seleccionan los circuitos de voltaje y los acoplamientos en su entrada, para cada canal basados en abrir o cerrar circuitos mediante relés.

Para accionar los circuitos de acondicionamiento y accionar los tipos de acoplamiento en cada canal, se pensó en implementar a la entrada de toda la circuitería, interruptores bilaterales integrados para conectar o desconectar la entrada para que sea GND, AC o DC, así como también seleccionar las resistencias que dan la ganancia a cada circuito acondicionador. Pero en vez de dichos interruptores, al final se optó por usar relés de 5V, ya que por sus características eléctricas si son capaces de soportar los voltajes y corrientes que se van a tener a la entrada de cada canal.

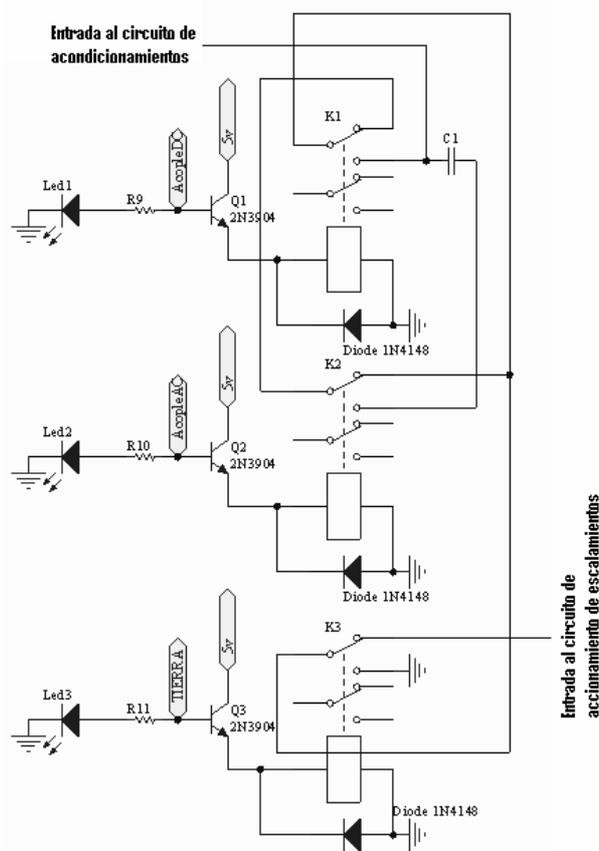


Figura 2.2 Circuito de Acoplamientos (AC, DC, Tierra).

Dichos relés serán controlados indirectamente por estados lógicos que provengan de 4 salidas del dsPIC (Como se explica en el Literal 2.2.2). Dos

para selección del circuito de acondicionamiento y dos para selección del tipo de acoplamiento.

2.2.3.1 Circuito de accionamiento de acoplamientos

Al igual que en los osciloscopios convencionales, para cada canal se debe poder seleccionar una entrada con acoplamiento AC, DC y conexión a tierra (GND). Con este propósito se diseñó el circuito de la Figura 2.2. Siguiendo las instrucciones del usuario, el dsPIC30F3011, a través de la circuitería explicada en el Literal 2.2.2, acciona los relés para lograr que la señal de entrada ingrese directamente (DC), o lo haga a través del capacitor C1 (AC), o vaya a tierra (GND).

2.2.3.2 Circuito de accionamiento de acondicionamientos de voltaje

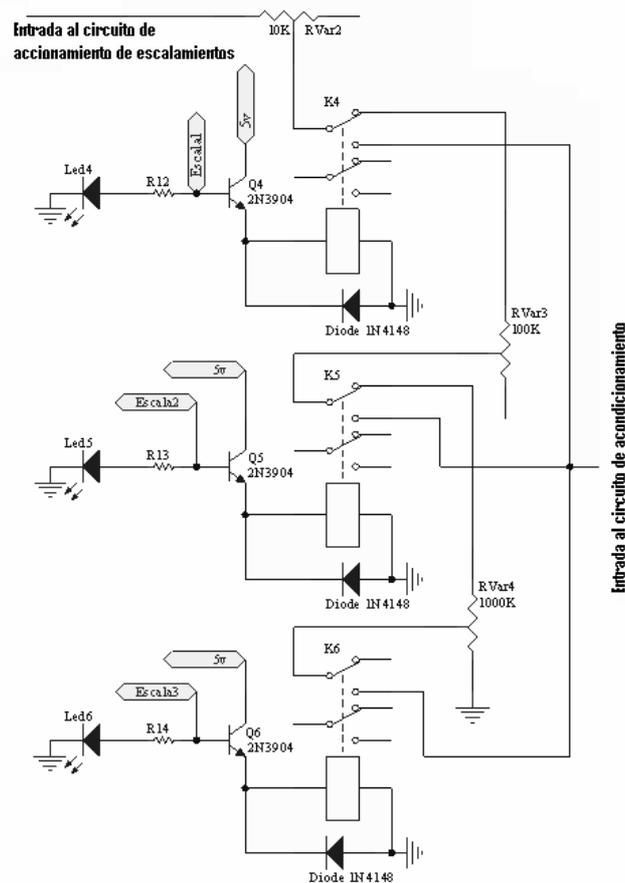


Figura 2.3 Circuito de escalamientos 2.5 V, 25 V, 250 V.

El programa desarrollado para el microcontrolador verifica los niveles de voltaje que ingresan al conversor AD del dsPIC y dependiendo de dichos niveles, envía órdenes al hardware (Figura 2.3) para direccionar o seleccionar el circuito acondicionador, sean:

- Circuito de escala mínima: -2.5V pico negativo, 2.5V pico positivo.
- Circuito de escala media: -25V pico negativo, 25V pico positivo.
- Circuito de escala máxima: -250V pico negativo, 250V pico positivo.

De esta forma se protege al sistema y se pueden apreciar estos voltajes de forma adecuada y sobre todo aprovechando al máximo en todas las escalas la resolución de 10 bits del conversor AD.

Para el diseño de este circuito, se debe considerar el valor de la resistencia de realimentación negativa (Figura 2.4) de la primera sub-etapa de la etapa de acondicionamiento, la cual se asumió como $R_{Var7} = 10 [k\Omega]$.

- Para el circuito de escala mínima: -2.5V pico negativo, 2.5V pico positivo:

Tomando en cuenta los valores picos positivos de voltaje, se sabe que la relación del voltaje de salida (V_{in_2}) respecto al de entrada (V_{RVar1}) para el circuito de escala mínima debe ser:

$$G1 = \frac{V_{in_2}}{V_{RVar1}} = \frac{2.5[V]}{2.5[V]/2}; \quad \underline{G1 = 2}$$

$$G1 = \frac{R_{Var7}}{R_{in_1}}; \quad R_{in_1} = \frac{R_{Var7}}{G1} = \frac{10 [k\Omega]}{2}; \quad \underline{R_{in_1} = 5 [k\Omega]}$$

$$R_{in_1} = R_{Var2}; \quad \underline{R_{Var2} = 5 [k\Omega]}$$

- Para el circuito de escala media: -25V pico negativo, 25V pico positivo:

Tomando en cuenta los valores picos positivos de voltaje, se sabe que la relación del voltaje de salida (V_{in_2}) respecto al de entrada (V_{RVar1}) para el circuito de escala media debe ser:

$$G2 = \frac{V_{in_2}}{V_{RVar1}} = \frac{2.5[V]}{25[V]/2}; \quad \underline{G2 = 0.2}$$

$$G2 = \frac{R_{Var7}}{Rin_2}; \quad Rin_2 = \frac{R_{Var7}}{G2} = \frac{10[k\Omega]}{0.2}; \quad \underline{Rin_2 = 50[k\Omega]}$$

$$Rin_2 = R_{Var2} + R_{Var3}; \quad \underline{R_{Var3} = Rin_2 - R_{Var2}};$$

Ya se tiene que: $R_{Var2} = 5[k\Omega]$, por lo tanto:

$$R_{Var3} = Rin_2 - R_{Var2} = 50[k\Omega] - 5[k\Omega]; \quad \boxed{R_{Var3} = 45[k\Omega]}$$

- Para el circuito de escala máxima: -250V pico negativo, 250V pico positivo:

Tomando en cuenta los valores picos positivos de voltaje, se sabe que la relación del voltaje de salida (V_{in_2}) respecto al de entrada (V_{RVar1}) para el circuito de escala máxima debe ser:

$$G3 = \frac{V_{in_2}}{V_{RVar1}} = \frac{2.5[V]}{250[V]/2}; \quad \underline{G3 = 0.02}$$

$$G3 = \frac{R_{Var7}}{Rin_3}; \quad Rin_3 = \frac{R_{Var7}}{G3} = \frac{10[k\Omega]}{0.02}; \quad \underline{Rin_3 = 500[k\Omega]}$$

$$Rin_3 = R_{Var2} + R_{Var3} + R_{Var4}; \quad \underline{R_{Var4} = Rin_3 - R_{Var2} - R_{Var3}};$$

Ya se tiene que: $R_{Var2} = 5[k\Omega]$ y que $R_{Var3} = 45[k\Omega]$ por lo tanto:

$$R_{Var4} = Rin_3 - R_{Var2} - R_{Var3} = 500[k\Omega] - 5[k\Omega] - 45[k\Omega]; \quad \boxed{R_{Var4} = 450[k\Omega]}$$

2.2.4 ETAPA DE ACONDICIONAMIENTO DE LAS SEÑALES DE VOLTAJE

Hay que tomar en cuenta que se trabajará con señales de hasta 40 kHz de frecuencia, así que se debió escoger amplificadores operacionales que tengan

un ancho de banda y respuesta de frecuencia mucho mayor a ese valor. En este caso se disponían de los tradicionales LM741, los OP37, y los no muy conocidos MAX477, que son los que mejor ancho de banda tienen, 300MHz. Después de hacer pruebas, los elegidos fueron los MAX477 (más adelante se verá que se requirió de otra etapa después del acondicionamiento, que es el filtrado *anti-aliasing*, lo cual está detallado en la sección 2.2.6).

Al probar el funcionamiento del amplificador operacional MAX477, se observó que, dada su altísima ganancia, se producían oscilaciones en su salida. También al amplificar con gran ganancia señales pequeñas, la distorsión que se obtenía a la salida era muy considerable. Esta distorsión era básicamente oscilaciones pequeñas sobre la señal original.

Al cambiar este operacional por un LM741, el comportamiento del circuito se acercaba más al esperado. Las oscilaciones ya no existían y por lo tanto se obtenía una señal mucho más limpia. Sin embargo, la ganancia era mucho menor que la del MAX477, y esto provocaba que las resistencias elegidas para la selección de escalas (rangos de acondicionamiento) prácticamente no provoquen diferencia alguna en la ganancia de este segundo amplificador, en especial a altas frecuencias. Esto implicaba tener una señal mucho mas limpia pero sin control sobre la selección de rangos.

Por otro lado, al utilizar el LM741 el ancho de banda se ve drásticamente reducido, ya que este operacional no esta preparado para trabajar a las frecuencias que tiene como límite el osciloscopio. Además, al tener un ancho de banda menor, la ganancia de este operacional es modificada de acuerdo a la frecuencia de la señal de entrada, haciendo que a partir de una frecuencia media se comporte de forma no-lineal.

Luego se optó por probar con otro operacional, en este caso, un OP37 de Texas Instruments, pero sin lograr un resultado satisfactorio.

En definitiva por sus características de estabilidad, ancho de banda y frecuencia máxima, se escogió el amplificador operacional MAX477.

2.2.4.1 Diseño de la etapa de acondicionamiento de las señales de voltaje

Básicamente esta etapa se conforma de dos sub-etapas, ambas son amplificadores inversores. El primero tiene una resistencia de realimentación $R_{Var7} = 10 [k\Omega]$, y su ganancia está determinada por la resistencia de entrada formada por la etapa anterior (circuito de accionamiento de acondicionamiento de voltaje). La siguiente etapa es un amplificador inversor de ganancia unitaria cuya relación de ganancia está formada por 2 resistencias de $10 [k\Omega]$, $R_{18} = R_{20} = 10 [k\Omega]$.

El circuito diseñado se muestra en la Figura 2.4.

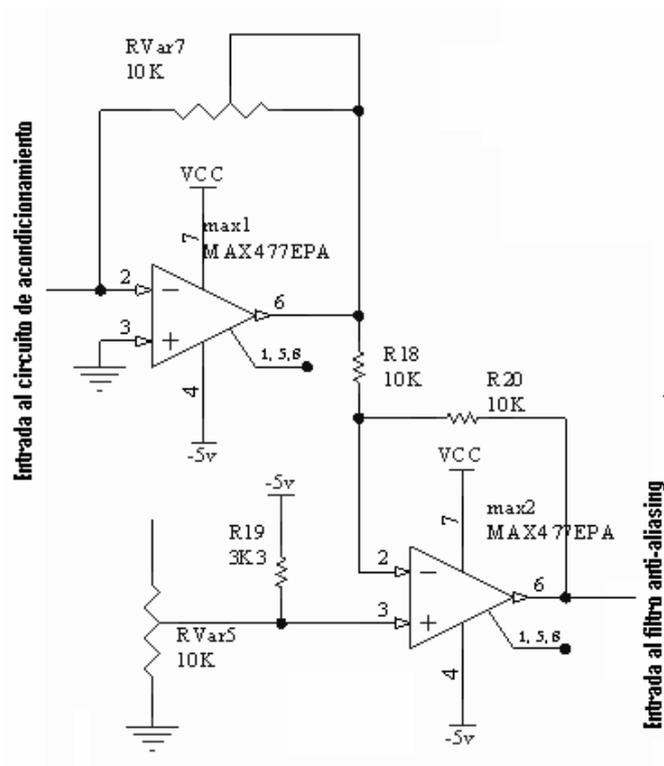


Figura 2.4 Circuito de acondicionamiento de las señales de voltaje.

2.2.4.2 Descripción del circuito diseñado

De la entrada BNC, y pasando por los relés de selección de acoplamiento y de selección de acondicionamiento, se tiene conexión al primer amplificador

operacional (max1). Este se encarga de invertir la señal de entrada debido a que se trata de una configuración de amplificador inversor.

La siguiente etapa se encarga de invertir nuevamente la señal y sumar una tensión media a la señal de entrada (2.5V), de modo que luego esta señal se encuentre centrada en el valor de tensión medio entre los rangos de entrada de los conversores A/D, que son de mínimo 0V y máximo 5V.

2.2.5 FILTRO ANTIALIASING A LA SALIDA DEL CIRCUITO DE ACONDICIONAMIENTO

El filtro *anti-aliasing* usado en este caso, fue una etapa adicional que no se creía necesaria en un inicio, pero después de haber diseñado y comprobado que se tenía a la salida del circuito acondicionador el efecto *aliasing* se vio la necesidad de implementar dicho filtro casi al final del proyecto, por eso no se habla del mismo en el Capítulo 1. A continuación se presenta el diseño del mismo, pero previamente un desarrollo introductorio referente al mismo.

2.2.5.1 Efecto aliasing

Al tomar muestras de una señal sin cumplir con lo establecido por el teorema de Nyquist; es decir, $F_s > 2 \cdot F_m$ (donde F_s es la frecuencia de muestreo y F_m es la frecuencia máxima de la señal a muestrear), se produce el efecto indeseado denominado *ALIASING* o *SOLAPAMIENTO*.

El *aliasing*, impide recuperar correctamente la señal original, dando como resultado otra (un "*alias*") con su misma forma, pero de frecuencia menor.

El efecto *aliasing* y la dispersión (o distanciamiento de un conjunto de valores con respecto a su valor medio) que introduce quedaron demostrados por los experimentos de Lagadec y Stockham.

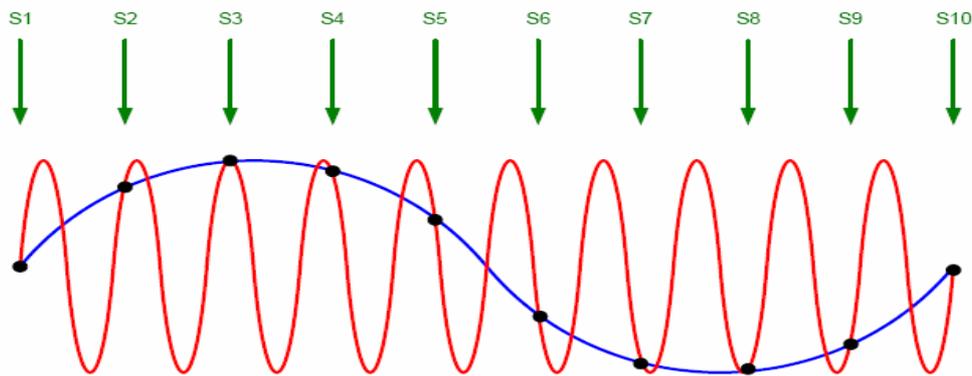


Figura 2.5 Demostrando aliasing, rojo es la verdadera onda, mientras azul es un alias.

En instrumentación es el ruido el que mayormente provoca *aliasing* y de ahí que conviene limitarlo mediante un filtro *anti-aliasing*.

2.2.5.2 Filtro anti-aliasing

Para eliminar el aliasing, los sistemas de digitalización incluyen filtros pasa bajos, que eliminan todas las frecuencias que sobrepasan la frecuencia máxima en la señal de entrada que se desea muestrear. El filtro pasa bajos para este uso concreto recibe el nombre de filtro *anti-aliasing*. Sin embargo, abusar de los filtros *anti-aliasing*, puede producir el mismo efecto que se quiere evitar. Cuando se conectan varios filtros en cadena (en el muestreo, en la conversión digital-analógica, etc.), un filtrado excesivo de una onda que ya cumplía con el requisito para su correcta transformación A/D puede degenerar y provocar que la onda final presente una pendiente marcada. Por esta desventaja del filtro *anti-aliasing* se ha generalizado la técnica conocida como sobre-muestreo de la señal¹.

El ejemplo de la Figura 2.5 demuestra claramente que, señales falsas (*aliasing*), pueden visualizarse en un ámbito digital. La onda roja es una forma de onda que está siendo monitoreada, mientras que la forma de onda que está bajo muestreo (véase flechas verdes que muestran a los puntos negros), muestra una perfecta sinusoidal (azul), que es un alias de la señal original.

¹ Tomado de: <http://www.azapedia.com/articulos/124/Frecuencia+de+muestreo.html>

Tenga en cuenta que es imposible decir que la señal azul es un alias. No hay nada que pueda hacerse después del muestreo para corregir el aliasing; de ahí que la solución es filtrar las altas frecuencias de la señal de entrada mediante un filtro pasa bajos.

2.2.5.3 Parámetros de un filtro pasa bajos

La Figura 2.6 muestra que hay cuatro parámetros fundamentales, que especifican a un filtro pasa bajos como son: $f_{\text{CUT-OFF}}$ (frecuencia de corte), f_{STOP} (frecuencia final), banda de transición, y M (Orden del Filtro).

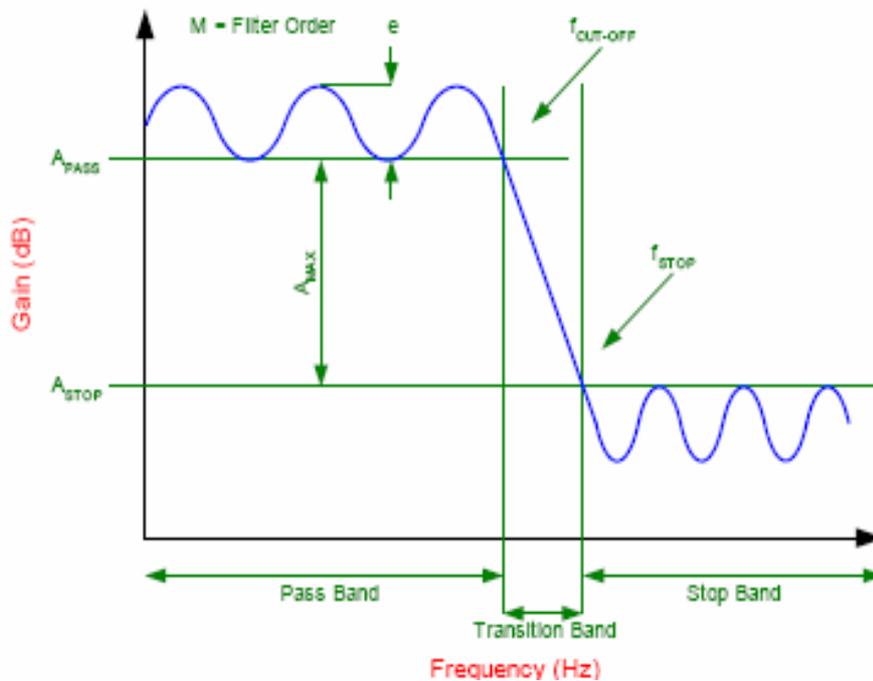


Figura 2.6 Parámetros fundamentales de un Filtro Pasa Bajos.

La frecuencia de corte $f_{\text{CUT-OFF}}$ de un filtro pasa bajos normalmente es definida como la de -3dB (filtros Butterworth y Bessel) o la frecuencia en que la respuesta del filtro deja el paso de banda (filtro Chebyshev).

Por definición, un filtro pasa bajos trabaja hasta la frecuencia de corte y atenúa las frecuencias más altas sobre la frecuencia de corte.

El orden del filtro es determinado por el número de polos en la función de transferencia (por ejemplo. 3 polos, tercer orden). En general, a mayor número de polos en un filtro, se tiene una menor banda de transición.

Idealmente, un filtro pasa bajos *anti-aliasing* debería cumplir con una respuesta, donde la banda de transición esté diseñada para ser tan pequeña como sea posible. Prácticamente hablando, esto puede ser muy difícil de conseguir. En el diseño de un filtro activo, cada dos polos requieren un amplificador operacional. Por ejemplo, si se diseña un filtro de orden 32, 16 amplificadores operacionales, 32 condensadores y hasta 64 resistencias serían necesarias para aplicar el circuito. Cada amplificador contribuiría a compensar y filtrar ruido en la banda de paso de la región de trabajo.

2.2.5.4 Elección de la frecuencia de muestreo y de la frecuencia del filtro

Gráficamente se muestra la relación entre frecuencia de muestreo y frecuencia del filtro en la Figura 2.7.

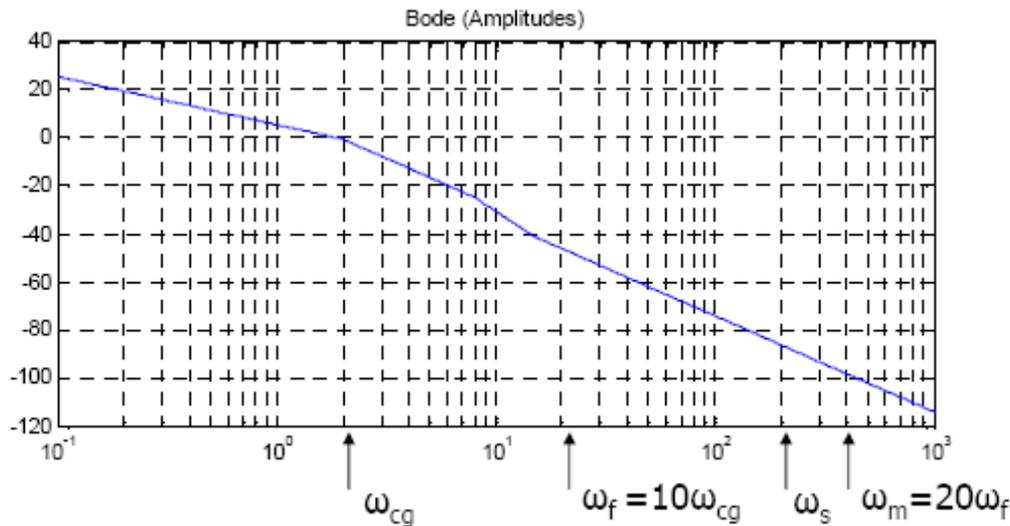


Figura 2.7 Relación entre frecuencia de muestreo y frecuencia del filtro¹.

En la práctica se acostumbra que la frecuencia de corte sea de 10 a 20 veces menor que la frecuencia de muestreo, todo esto para conseguir una atenuación suficiente antes de muestrear. Pero como se quiere filtrar también ruido, la frecuencia de corte será mayor que la máxima frecuencia a muestrear.

¹ Tomado de: <http://isa.uniovi.es/~arobles/ra2/pdf/mues.pdf>. Diapositiva # 15.

2.2.5.5 Filtro RC pasabajos utilizado

Si se considera un filtro paso bajo, la función de transferencia es igual a:

$$F(s) = \frac{V_{out}}{V_{in}} = \frac{1}{RC \left(s + \frac{1}{RC} \right)}$$

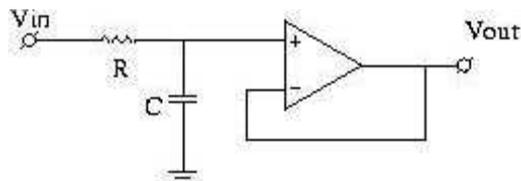


Figura 2.8 Circuito pasa bajo.

La frecuencia de corte será

$$f = \frac{1}{(2 \times \pi \times R \times C)}$$

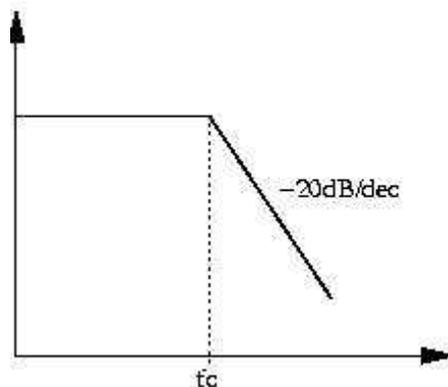


Figura 2.9 Respuesta de ganancia en el dominio de la frecuencia.

2.2.5.6 Filtro de segundo orden pasabajos utilizado

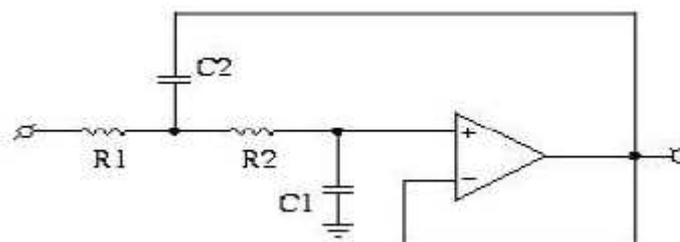


Figura 2.10 Filtro pasa bajos de segundo orden.

La Figura 2.10 muestra la configuración de un filtro pasa bajos de Segundo orden.

$$F(s) = \frac{1}{1 + [w_c \times C_1 \times (R_1 + R_2)] \times s + [w_c^2 \times R_1 \times R_2 \times C_1 \times C_2] \times s^2};$$

Donde los valores C_1 y C_2 son escogidos por el usuario, y $R_{1,2}$ se calculan así:

$$R_{1,2} = \frac{a_1 \times C_2 \pm \sqrt{a_1^2 \times C_2^2 - 4b_1 \times C_1 \times C_2}}{4\pi \times f_c \times C_1 \times C_2};$$

Donde los coeficientes a_1 y b_1 se calculan como sigue:

$$a_1 = w_c \times C_1 \times (R_1 + R_2); \quad b_1 = w_c^2 \times R_1 \times R_2 \times C_1 \times C_2;$$

Para obtener valores reales, se debe cumplir que:

$$C_2 \geq C_1 \frac{4 \times b_1}{a_1^2};$$

2.2.6 DISEÑO DEL FILTRO ANTI-ALIASING IMPLEMENTADO

El circuito *anti-aliasing* implementado está conformado por dos etapas: Un filtro pasa bajos de primer orden y a continuación, en serie, un filtro Butterworth pasa bajos de segundo orden, como muestra la Figura 2.11.

Para el diseño se va a tomar en cuenta la nomenclatura expuesta en el circuito de la Figura 2.11.

Etapas del filtro pasa-bajos de primer orden:

Se asumió: $C_2 = 1 [nF]$

$$R = (2 \times \pi \times f_c \times C_2)^{-1}; \quad R = (2 \times \pi \times 40000 \times 1 \times 10^{-9})^{-1}; \quad R = 3978.86 [\Omega]$$

Etapas del filtro pasa-bajos de segundo orden:

Siendo $C_4 = 0.01 [\mu F]$

Butterworth

| n | l | a_l | b_l | $k_l = \frac{f_{Cl}}{f_C}$ | Q_l |
|----|---|--------|--------|----------------------------|-------|
| 1 | 1 | 1.0000 | 0.0000 | 1.000 | — |
| 2 | 1 | 1.4142 | 1.0000 | 1.000 | 0.71 |
| 3 | 1 | 1.0000 | 0.0000 | 1.000 | — |
| | 2 | 1.0000 | 1.0000 | 1.272 | 1.00 |
| 4 | 1 | 1.8478 | 1.0000 | 0.719 | 0.54 |
| | 2 | 0.7654 | 1.0000 | 1.390 | 1.31 |
| 5 | 1 | 1.0000 | 0.0000 | 1.000 | — |
| | 2 | 1.6180 | 1.0000 | 0.859 | 0.62 |
| | 3 | 0.6180 | 1.0000 | 1.448 | 1.62 |
| 6 | 1 | 1.9319 | 1.0000 | 0.676 | 0.52 |
| | 2 | 1.4142 | 1.0000 | 1.000 | 0.71 |
| | 3 | 0.5176 | 1.0000 | 1.479 | 1.93 |
| 7 | 1 | 1.0000 | 0.0000 | 1.000 | — |
| | 2 | 1.8019 | 1.0000 | 0.745 | 0.55 |
| | 3 | 1.2470 | 1.0000 | 1.117 | 0.80 |
| | 4 | 0.4450 | 1.0000 | 1.499 | 2.25 |
| 8 | 1 | 1.9616 | 1.0000 | 0.661 | 0.51 |
| | 2 | 1.6629 | 1.0000 | 0.829 | 0.60 |
| | 3 | 1.1111 | 1.0000 | 1.206 | 0.90 |
| | 4 | 0.3902 | 1.0000 | 1.512 | 2.56 |
| 9 | 1 | 1.0000 | 0.0000 | 1.000 | — |
| | 2 | 1.8794 | 1.0000 | 0.703 | 0.53 |
| | 3 | 1.5321 | 1.0000 | 0.917 | 0.65 |
| | 4 | 1.0000 | 1.0000 | 1.272 | 1.00 |
| | 5 | 0.3473 | 1.0000 | 1.521 | 2.88 |
| 10 | 1 | 1.9754 | 1.0000 | 0.655 | 0.51 |
| | 2 | 1.7820 | 1.0000 | 0.756 | 0.56 |
| | 3 | 1.4142 | 1.0000 | 1.000 | 0.71 |
| | 4 | 0.9080 | 1.0000 | 1.322 | 1.10 |
| | 5 | 0.3129 | 1.0000 | 1.527 | 3.20 |

Tabla 2.2 Coeficientes del filtro Butterworth pasa-bajos de segundo orden¹.

De la Tabla 2.2 se tienen los coeficientes a_1^2 y b_1 para un filtro de segundo orden ($n = 2$):

$$\underline{b_1 = 1}; \quad \underline{a_1^2 = (1.4142)^2 = 2}$$

Para obtener valores reales se debe cumplir que: $C_3 \geq C_4 \frac{4 \times b_1}{a_1^2}$, entonces:

¹ Tomado de: <http://www.ulpgc.es/hege/almacen/download/29/29861/filtros.pdf>.

$$C_3 \geq C_4 \frac{4 \times b_1}{a_1^2}; \quad C_3 \geq 0.01 [\mu F] \times \frac{4 \times 1}{2}; \quad \underline{C_3 \geq 0.02 [\mu F]};$$

El valor estandar más próximo es: $C_3 = 0.022 [\mu F]$

Ya teniendo los valores de la frecuencia de corte f_c , las capacitancias C_4 y C_3 , y los coeficientes a_1^2 y b_1 entonces se puede calcular los valores de las resistencias R_{22} y R_{Var6} .

$$R_{22} = \frac{a_1 \times C_3 + \sqrt{a_1^2 \times C_3^2 - 4 \times b_1 \times C_4 \times C_3}}{4\pi \times f_c \times C_4 \times C_3};$$

$$R_{22} = \frac{1.4142 \times 0.022 [\mu F] + \sqrt{2 \times (0.022 [\mu F])^2 - 4 \times (1) \times (0.01 [\mu F]) \times (0.022 [\mu F])}}{4\pi \times 40 [kHz] \times (0.01 [\mu F]) \times (0.022 [\mu F])};$$

$$\boxed{R_{22} = 365.444 [\Omega]}$$

$$R_{Var6} = \frac{a_1 \times C_3 - \sqrt{a_1^2 \times C_3^2 - 4 \times b_1 \times C_4 \times C_3}}{4\pi \times f_c \times C_4 \times C_3};$$

$$R_{Var6} = \frac{1.4142 \times 0.022 [\mu F] - \sqrt{2 \times (0.022 [\mu F])^2 - 4 \times (1) \times (0.01 [\mu F]) \times (0.022 [\mu F])}}{4\pi \times 40 [kHz] \times (0.01 [\mu F]) \times (0.022 [\mu F])};$$

$$\boxed{R_{Var6} = 197.247 [\Omega]}$$

El circuito completo del filtro se muestra en la Figura 2.11.

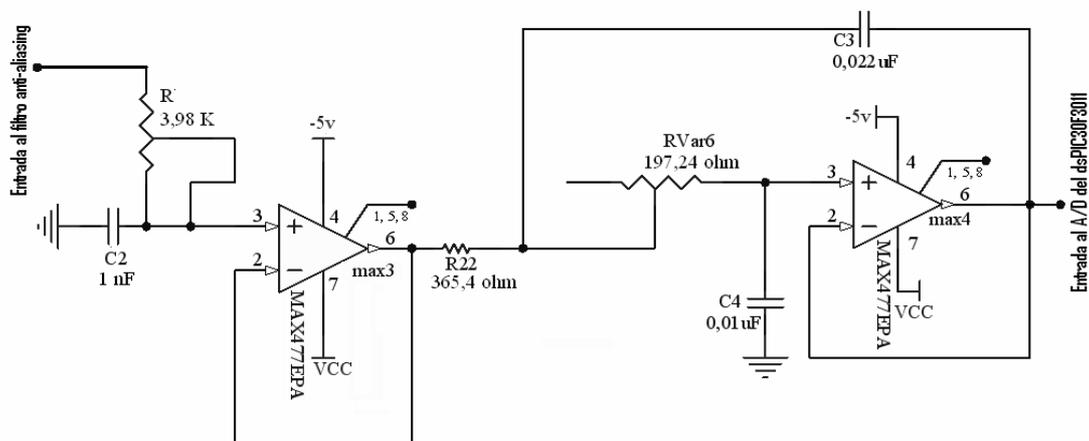


Figura 2.11 Filtro anti-aliasing implementado.

2.3 DISEÑO DE LA TARJETA DE CONTROL Y COMUNICACIÓN

La tarjeta de control y comunicación tiene como principal elemento al **dsPIC30F3011**, que es el que controla todo el envío y recepción de datos (TRAMAS) desde y hacia la PC, basado en las órdenes que recibe de la PC.

El elemento que sirve de interfaz de comunicación entre la PC y el dsPIC30F3011 es el módulo de comunicación USB Paralelo **USBMOD4**. Sin el USBMOD4 se haría necesario construir la interfaz y desarrollar todo el protocolo para convertir los datos que se tienen de forma paralela en el puerto del dsPIC a USB.

2.3.1 ADQUISICIÓN Y ADMINISTRACIÓN DE LOS DATOS DEL AD

Después de la etapa de acondicionamiento y filtro anti-aliasing se requiere un conversor análogo-digital de al menos 10 bits de resolución, debido al rango de escalas de voltaje que se va a manejar, desde 10 mV/DIV hasta 50 V/DIV.

Como se necesita de alta velocidad de muestreo y conversión, se requiere que el conversor tenga el método de conversión de aproximaciones sucesivas, con una velocidad de muestreo mínimo de 160000 muestras por segundo, para que, según el teorema de Nyquist, se pueda convertir frecuencias de hasta 40 kHz, el máximo fijado en el alcance.

En este caso, el módulo AD del dsPIC30F3011 posee un conversor A/D de aproximaciones sucesivas, con una resolución de 10 bits, y un rango máximo de 500000 muestras/seg (500 ksps).

La adquisición y administración de los datos del AD, básicamente consta de 3 partes:

- a) Adquisición de datos propiamente dicha (conversor AD).
- b) Procesamiento de los datos.
- c) Construcción de TRAMAS.

2.3.1.1 Descripción de las características más relevantes del dsPIC30F3011

Para este proyecto se escogió el microcontrolador híbrido (CPU y DSP) dsPIC30F3011, NO precisamente por las características de DSP (Procesador digital de señales) que posee, sino básicamente por la velocidad de su módulo conversor AD, que en este caso es de hasta 500000 muestras/segundo.

La arquitectura y características básicas del dsPIC, se resumen a continuación:

- **Arquitectura CPU**
 - Arquitectura de Hardware modificada.
 - Datos de 16 bits.
 - Instrucciones de 24 bits.
 - 24 Kbytes de espacio de memoria Flash de programa (8 Kbytes de palabras de instrucción).
 - 1 Kbyte de memoria RAM.
 - 1 Kbyte de memoria EEPROM (no volátil)
 - Arreglo de registros de 16x16 bits.
 - Operación de hasta 30 MIPS (mega-instrucciones por segundo).
 - 29 fuentes de interrupción
 - 3 fuentes de interrupción externas.
 - 8 niveles de prioridad, seleccionables para cada fuente de interrupción.
 - 4 fuentes de trampa de procesador.

- **Arquitectura DSP**
 - Extracción de datos dual.
 - Sobreescritura de acumulador para operaciones DSP.
 - Modo de direccionamiento de módulo y de bit reservado.
 - Dos acumuladores de 40 bits de ancho, con saturación lógica opcional.
 - Multiplicador-integrador de 17 bits x 17 bits de ciclo simple.
 - Todas las instrucciones DSP se hacen en un solo ciclo.

2.3.1.2 INTERRUPCIONES EN EL DSPIC30F3011

Este dispositivo posee 29 fuentes de interrupción en total, y 4 excepciones de procesador (trampas). Las interrupciones tienen un esquema arbitrario de prioridad y es posible programar la prioridad.

El CPU es el responsable de la lectura de la tabla de vectores de interrupción (IVT) y transferencia del contenido de una dirección en el vector de interrupción hacia el contador de programa.

El vector de interrupción es transferido desde el bus de datos del programa hacia el contador de programa, mediante un multiplexor de 24 bits de ancho en la entrada del contador de programa.

La tabla de vectores de interrupción (IVT) y la tabla de vectores de interrupción alternativa (AIVT) se ubican cerca del inicio (0x000004) de la memoria de programa.

El controlador de interrupciones es responsable por el procesamiento de las interrupciones y las excepciones del procesador. Las interrupciones periféricas y trampas son habilitadas, priorizadas y controladas usando Registros de Funciones Especiales (SFR), y son:

- IFS0<15:0>, IFS1<15:0>, IFS2<15:0>

Todas las banderas de interrupción son mantenidas en estos tres registros. Estas banderas son activadas (se ponen en 1), por sus respectivos periféricos o señales externas, y son limpiadas (se ponen en 0), mediante software.

- IEC0<15:0>, IEC1<15:0>, IEC2<15:0>

Los bits de control de habilitación de todas las interrupciones están en estos tres registros. Estos bits son usados para habilitar las interrupciones de forma individual.

- IPC0<15:0> hasta IPC11<15:0>
El nivel de prioridad de atención de cada una de las interrupciones, es asignado en estos doce registros.
- IPL<3:0>
El nivel de prioridad es explícitamente almacenado en los bits IPL. IPL<3> está en el registro CORCON, mientras que los bits IPL<2:0> están presentes en el registro SR (registro de estado).
- INTCON1<15:0>, INTCON2<15:0>
Las funciones de control global de interrupciones son derivadas de estos dos registros. INTCON1 contiene el control y el estado de las banderas para las excepciones del procesador. INTCON2 controla los requerimientos de señal de las interrupciones externas y el uso de la tabla de vectores alternativa.

En el caso del presente proyecto, solo se usan 3 fuentes de interrupción, dos externas y una interna y son:

- Interrupción externa INT0 (pin 17, RE8). Bit INT0EP del registro INTCON2, bit INT0IE del registro IEC0 y bit INT0IF del registro IFS0. Usada cuando el dsPIC recibe un dato cualquiera desde la PC.
- Interrupción externa INT2 (pin 18, RD1). Bit INT2EP del registro INTCON2, bit INT2IE del registro IEC1 y bit INT2IF del registro IFS1. Usada cuando se termina de enviar una trama hacia la PC (solo en el modo de almacenamiento).
- Interrupción del AD (al terminar una conversión). Bit ADIE del registro IEC0 y bit ADIF del registro IFS0.

2.3.1.3 Módulo AD de 10 bits del dsPIC30F3011

Este módulo posee las siguientes características:

- 4 conversores individuales de entrada S/H (muestreo y retención).
- Método de aproximaciones sucesivas.

- Máximo de 500000 muestras/segundo.
- 9 canales de entrada.
- Conversión disponible durante los modos Idle y Sleep.

Para más información sobre las características del dsPIC30F3011, por favor referirse al datasheet, incluido en el ANEXO A.

2.3.2 MODULO DE COMUNICACIÓN USBMOD4

Para la etapa de comunicación, se planteó usar la interfaz USB, para poder ver formas de onda en tiempo real de hasta 40 kHz. Puesto que cada dato del AD, al ser de 10 bits de resolución, necesariamente se debe separar en 2 partes, entonces, por cada dato del AD, se deberá enviar hacia la PC, 2 bytes. Consecuentemente, para un máximo de 40 kHz se necesita de un mínimo de 80000 muestras por segundo (según el teorema de Nyquist), por lo que se requiere de mínimo 160000 bytes/segundo como velocidad de transmisión.

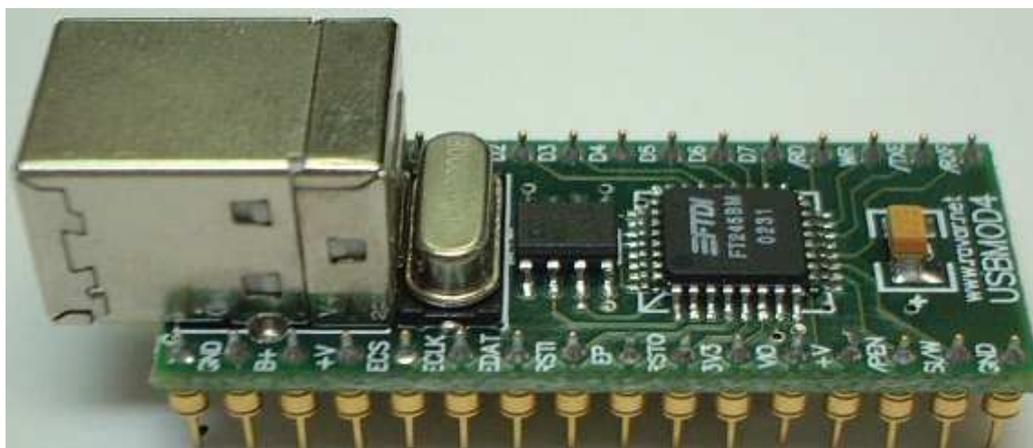


Figura 2.12 Módulo de comunicación USBMOD4.

En este caso, el módulo de comunicación USBMOD4, tiene una velocidad de recepción y transmisión de máximo 1 Mbyte/seg.

La comunicación entre la PC y el dsPIC30F3011 se realiza utilizando como interfaz al convertidor de protocolo de comunicación USBMOD4, que se puede apreciar en la Figura 2.12. Este es el que permite que cualquier byte que se

tenga como salida en un puerto del dsPIC, se pueda enviar vía USB, en este caso hacia una PC a través de sus 8 pines de datos, o viceversa.

2.3.2.1 Descripción de características del USBMOD4

El USBMOD4 es un módulo integrado para transferencia de datos desde o hacia un periférico o una PC, a una velocidad de 8 Megabits por segundo (1 Megabyte por segundo). Está basado en un circuito integrado para transferencia de datos rápida de paralelo a USB y viceversa, llamado FTDI FT245BM USB FIFO.

Las características más importantes del módulo son:

- ✓ Módulo de 32 pines.
- ✓ High speed, con velocidad máxima de transmisión y recepción de 1 Mbyte/seg.
- ✓ Cristal de 6 MHz.
- ✓ Conector USB tipo B.
- ✓ No requiere de elementos pasivos externos.
- ✓ Transferencia de datos de USB a paralelo y viceversa.

2.3.2.2 Distribución de pines y modo de uso del USBMOD4

En el **ANEXO A** está la información completa de la distribución de pines del módulo USBMOD4, pero para el caso de esta aplicación, en la Tabla 2.3 solo se mencionan aquellos pines que son utilizados.

| # de PIN | SEÑAL | TIPO | DESCRIPCIÓN |
|----------|-------|--------------|--|
| 1 | G | alimentación | Pin de tierra (referencia 0 V) |
| 2 | GND | alimentación | Pin de tierra (referencia 0 V) |
| 3 | B+ | alimentación | Bus de alimentación USB (+5V) |
| 4 | +V | alimentación | Pin de alimentación de +4.4 a 5.25 V |
| | | | NOTA: No se necesita de alimentación externa cuando se está en el modo de "alimentación del BUS" |

| # de PIN | SEÑAL | TIPO | DESCRIPCIÓN |
|----------|-------|----------------|---|
| 9 | EP | Entrada | Este pin es el llamado de Enumeración de energía (Enumeration Power), este se conecta al pin RSTO para operación en modo de "alimentación del BUS" |
| 10 | RSTO | Salida | Salida del generador de Reset interno. Se mantiene en alta impedancia por 2ms, después de que VCC>3.5V y arranca el reloj interno, entonces el pin se enclava a la salida de 3.3V del regulador interno. |
| 12 | VIO | alimentación | Alimentación externa de 3.0 a 5.25 V para los pines de interfaz FIFO, 10... 12, 14 ...16 y 18... 25. Se conecta a una fuente externa lógica de 3.3 V cuando así se necesite, en cualquier otro caso se conecta a +V para manejar niveles CMOS de 5 V. |
| 13 | +V | alimentación | Pin de alimentación de +4.4 a 5.25 V |
| | | | NOTA: No se necesita de alimentación externa cuando se está en el modo de "alimentación del BUS" |
| 16 | GND | alimentación | Pin de tierra (referencia 0 V) |
| 17 | /RXF | Salida | Cuando está en alto, NO ha llegado ningún dato a la memoria FIFO o acaba de vaciarse la misma. Cuando está en bajo significa que llegó un dato a la FIFO. Este dato puede ser leído (en los pines D0 a D7) poniendo en cero el pin /RD. |
| 18 | /TXE | Salida | Mientras está en bajo, no se puede escribir ningún dato en la FIFO. Cuando se pone en bajo el pin WR y además /TXE está en alto, ya puede ser escrito el dato en la FIFO, el cual se va inmediatamente vía USB. |
| 19 | WR | Entrada | Escribe el byte que está en los pines del D0 al D7 en el Buffer FIFO de transmisión, cuando se pone en cero. |
| 20 | /RD | Entrada | Habilita el byte que está desde D0 a D7 en la memoria FIFO cuando está en bajo. Saca el siguiente byte (si está disponible) desde el Buffer FIFO de recepción cuando pasa de bajo a alto. |
| 21 | D7 | Entrada/salida | Bus de un bit bidireccional # 7 |
| 22 | D6 | Entrada/salida | Bus de un bit bidireccional # 6 |
| 23 | D5 | Entrada/salida | Bus de un bit bidireccional # 5 |
| 24 | D4 | Entrada/salida | Bus de un bit bidireccional # 4 |
| 25 | D3 | Entrada/salida | Bus de un bit bidireccional # 3 |
| 26 | D2 | Entrada/salida | Bus de un bit bidireccional # 2 |
| 27 | D1 | Entrada/salida | Bus de un bit bidireccional # 1 |
| 28 | D0 | Entrada/salida | Bus de un bit bidireccional # 0 |
| 29 | GND | alimentación | Pin de tierra (referencia 0 V) |
| 32 | G | alimentación | Pin de tierra (referencia 0 V) |

Tabla 2.3 Distribución de pines del módulo USBMOD4 (pines más importantes).

2.3.2.3 Instalación de controladores del USBMOD4

Para instalar los controladores del módulo FT245BM del USBMOD4, primero se debe descargarlos de la página de FTDI chip¹.

Se pueden descargar en forma de archivo comprimido *.zip, para instalarlo con la opción de instalación de hardware de Windows (para este caso), o como archivo *.exe de instalación automática.

Los pasos para la instalación de dichos controladores, se detalla en el **ANEXO C**.

2.3.3 DESCRIPCIÓN DE FUNCIONAMIENTO DE LA TARJETA DE CONTROL Y COMUNICACIÓN

Cabe destacar que todas las órdenes son manejadas o accionadas desde la interfaz en la PC, existen algunas que ejecuta el dsPIC, y otras que se ejecutan en la misma PC. En este caso se exponen aquellas que ejecuta el dsPIC.

2.3.3.1 Requerimientos del control

Todas las órdenes se manejarán o accionarán desde la interfaz gráfica en la PC que simula el osciloscopio, pero las acciones de control se repartirán entre el microcontrolador y la interfaz, dependiendo de cuan óptimo resulte para cada tipo de orden.

Para manejar las órdenes desde la PC, se pensó en dos métodos:

El primero consiste en que cada cierto tiempo, específicamente, cada vez que se termine de graficar una pantalla, el microcontrolador deje de enviar datos hacia la PC, e inicie una comunicación para leer el estado de algunas “banderas”, tantas como órdenes maneje la interfaz (cada orden de la interfaz

¹ Tomado de: <http://www.ftdichip.com/Drivers/VCP.htm>

tendría su correspondiente bandera), y entonces ejecute dichas órdenes antes de empezar a graficar la siguiente pantalla.

El segundo consiste en ejecutar las órdenes en forma inmediata, mediante la utilización de una o algunas interrupciones externas del microcontrolador, sin tener que esperar que se termine de graficar una pantalla o, en general, sin tener que leer el estado de las “banderas” desde la PC cada cierto tiempo. Pero este método requiere del hardware necesario que genere una señal que indique cada vez que la PC quiera enviar algo hacia el microcontrolador, señal que debe conectarse a la interrupción externa del microcontrolador (el dsPIC en este caso).

Se utilizó el segundo método, debido a que el módulo de comunicación usado (USBMOD4), si posee dichas “banderas” o indicadores de recepción (RXF), y de transmisión (/TXE), los cuales se conectan a 2 pines con interrupción externa en el dsPIC30F3011.

2.3.3.2 Conversión AD

Las salidas de los filtros anti-aliasing, son las que ingresan directamente a las entradas del convertidor A/D, que son: RB0 (pin 2), RB1 (pin 3) y RB2 (pin 4) para los canales 1, 2 y 3 respectivamente. El modo de conversión que se programó en el dsPIC, es el que usa el muestreo simultáneo; es decir, que cada pin de entrada está conectado a un solo canal S/H (muestreo / retención) del módulo A/D del dsPIC, es decir, RB0, usa el muestreador / retenedor del CH1, RB1 usa el muestreador / retenedor del CH2, RB2 usa el muestreador / retenedor del CH3, y RB3 usa el muestreador / retenedor del CH0. Al ser independientes, se puede tomar hasta 4 lecturas (porque posee 4 convertidores independientes) de manera simultánea, en este caso se usa las de los canales CH1, CH2 y CH3.

2.3.3.3 Tratamiento de los datos resultantes de la conversión AD

Al tener ya los datos de los 3 canales, se procede primero a una revisión para ver si está activo el TRIGGER por software con cada dato que llega a la PC.

Los modos de trabajo del trigger se pueden programar desde la interfaz gráfica. Los modos son: respecto a cual de los tres canales disparar las formas de onda en pantalla, el nivel de disparo automático (fijo) o manual (variable) y el flanco de disparo de subida (positivo) o de bajada (negativo).

Luego, con o sin TRIGGER aplicado, se tiene un algoritmo que decide los datos de que canales serán acondicionados y enviados, en el caso del modo en *tiempo real*, o cuales serán acondicionados, almacenados y enviados, en el caso de modo de *almacenamiento* (mayores detalles de estos modos, se cubre en los Capítulos 3 y 4).

Al tratarse de un muestreo simultáneo, el conversor cuando está activo, está siempre tomando 4 lecturas del AD. (En el presente caso se necesita solo de 3 lecturas, porque son 3 canales, pero solo se puede programar muestreo simultáneo para 1, 2 y 4 entradas). Se prefirió no variar esa característica, así se estén usando solo uno o dos canales, ya que tomar 1, 2 o 3 lecturas, o en otras palabras usar 1, 2 o 3 canales, en tiempo resulta casi lo mismo.

2.3.3.4 Control de la comunicación entre el dsPIC y el USBMOD4

El dsPIC30F3011 se comunica con el módulo USBMOD4 vía paralelo. Para los datos (tramas) el USBMOD4 utiliza 8 de sus salidas desde el D0 a D7 que están conectados a RF0 (pin 30), RF1 (pin 29), RE0 (pin 38), RE1 (pin 37), RE2 (pin 36), RE3 (pin 35), RE4 (pin 34) y RE5 (pin 33) del dsPIC, respectivamente.

Para el control de la comunicación se usan /RD, WR, /TXE y /RXF, conectados al dsPIC en RD2 (pin 22), RD3 (pin 19), INT2 (pin 18) y INT0 (pin 17) respectivamente como se indica en la Figura 2.13.

La bandera de transmisión del módulo USBMOD4 en el pin /TXE se la utiliza solo en el modo de *almacenamiento* de tramas, debido a que en el modo en *tiempo real* siempre se está transmitiendo dichas tramas, y si se detuviera el envío de cada una, mediante la interrupción externa INT2 (en el pin 18),

generaría retardos críticos, lo cual sería perjudicial en este modo. Por otro lado, en el modo de “almacenamiento”, no resulta perjudicial debido a que se trata de datos ya almacenados y la velocidad de comunicación no es un cuello de botella como lo es en el modo en “tiempo real”. El uso de esta bandera confirma que se termina de enviar una trama, para empezar a enviar la siguiente.

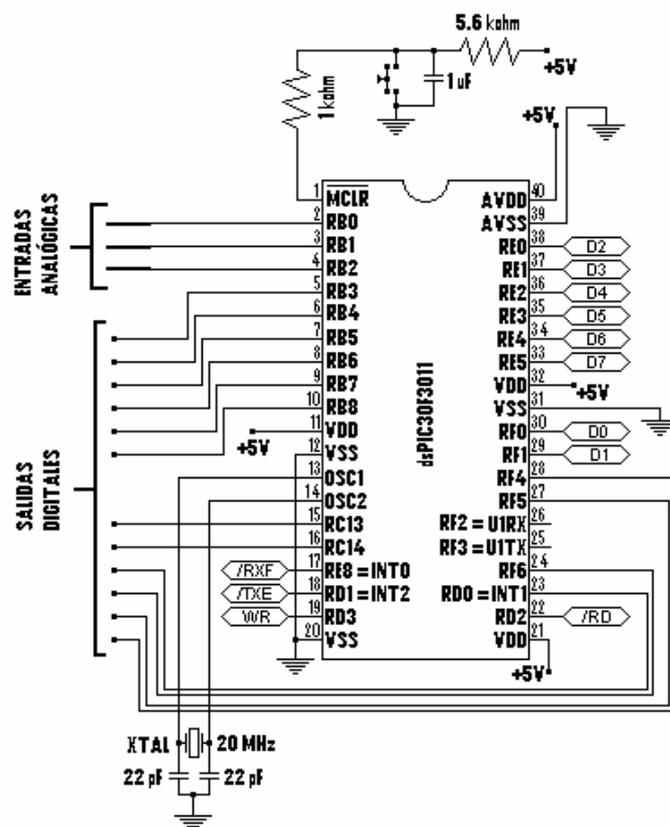


Figura 2.13 Conexión de líneas de datos y comunicación entre el dsPIC30F3011 y el USBMOD4.

Para el correcto funcionamiento del dsPIC (Figura 2.13), se utilizó un cristal de 20 MHz (para este microcontrolador se pueden usar cristales desde 4 MHz hasta 25 MHz). El máximo que se encontró en el mercado fue de 24 MHz, pero no funcionó de ninguna manera. Y para el acoplamiento se escogió 2 capacitores de 22 pF (valor dentro del rango recomendado por el fabricante), que son valores suficientes para que el oscilador no sea ni muy lento, ni tampoco llegue a ser inestable.

2.3.3.5 Control de los circuitos de acondicionamiento y de los acoplamientos

Las líneas de control son aquellas salidas del dsPIC que determinan qué circuito de acondicionamiento se activa, dependiendo de la amplitud de la señal entrante, y también controlan, el acoplamiento que se asigna a la entrada del circuito acondicionador que se haya seleccionado; las opciones típicas son: GND, DC y AC.

Las líneas encargadas de controlar el accionar de los circuitos de acondicionamiento y los acoplamientos son las siguientes:

Para el canal 1 se tiene:

Los estados lógicos de RB3 (pin 5) y RB4 (pin 6) son los que determinan que circuito acondicionador actúa a la entrada del CANAL 1, donde RB3 corresponde al bit LSB, y RB4 al bit MSB.

Los estados lógicos de RC14 (pin 16) y RC13 (pin 15) son los que determinan el acoplamiento, es decir, si la entrada será DC, AC, o GND. En este caso RC14 corresponde al bit LSB y RC13 al bit MSB.

Para los canales 2 y 3 se aplica la lógica que se indica en la Tabla 2.3.

| CANAL | # de pin | identificación | Función |
|-------|----------|----------------|---|
| 1 | 5 | RB3 (LSB) | Control de enclavamiento de circuitos de acondicionamiento del canal 1 |
| | 6 | RB4 (MSB) | |
| | 16 | RC14 (LSB) | Control de enclavamiento de tipo de acoplamiento (AC, DC o GND) para el canal 1 |
| | 15 | RC13 (MSB) | |
| 2 | 7 | RB5 (LSB) | Control de enclavamiento de circuitos de acondicionamiento del canal 2 |
| | 8 | RB6 (MSB) | |
| | 24 | RF6 (LSB) | Control de enclavamiento de tipo de acoplamiento (AC, DC o GND) para el canal 2 |
| | 23 | RD0 (MSB) | |
| 3 | 9 | RB7 (LSB) | Control de enclavamiento de circuitos de acondicionamiento del canal 3 |
| | 10 | RB8 (MSB) | |
| | 28 | RF4 (LSB) | Control de enclavamiento de tipo de acoplamiento (AC, DC o GND) para el canal 3 |
| | 27 | RF5 (MSB) | |

Tabla 2.4 Función de las salidas del dsPIC para el control de los circuitos de acondicionamiento y acoplamiento.

Los 12 pines explicados en la Tabla 2.4 son las salidas de la tarjeta de control y comunicación, mientras que las entradas del módulo conversor AD son: RB0 (pin 2) que es la entrada analógica del canal 1, RB1 (pin 3) que es la entrada analógica del canal 2; y, RB2 (pin 4) que es la entrada analógica del canal 3.

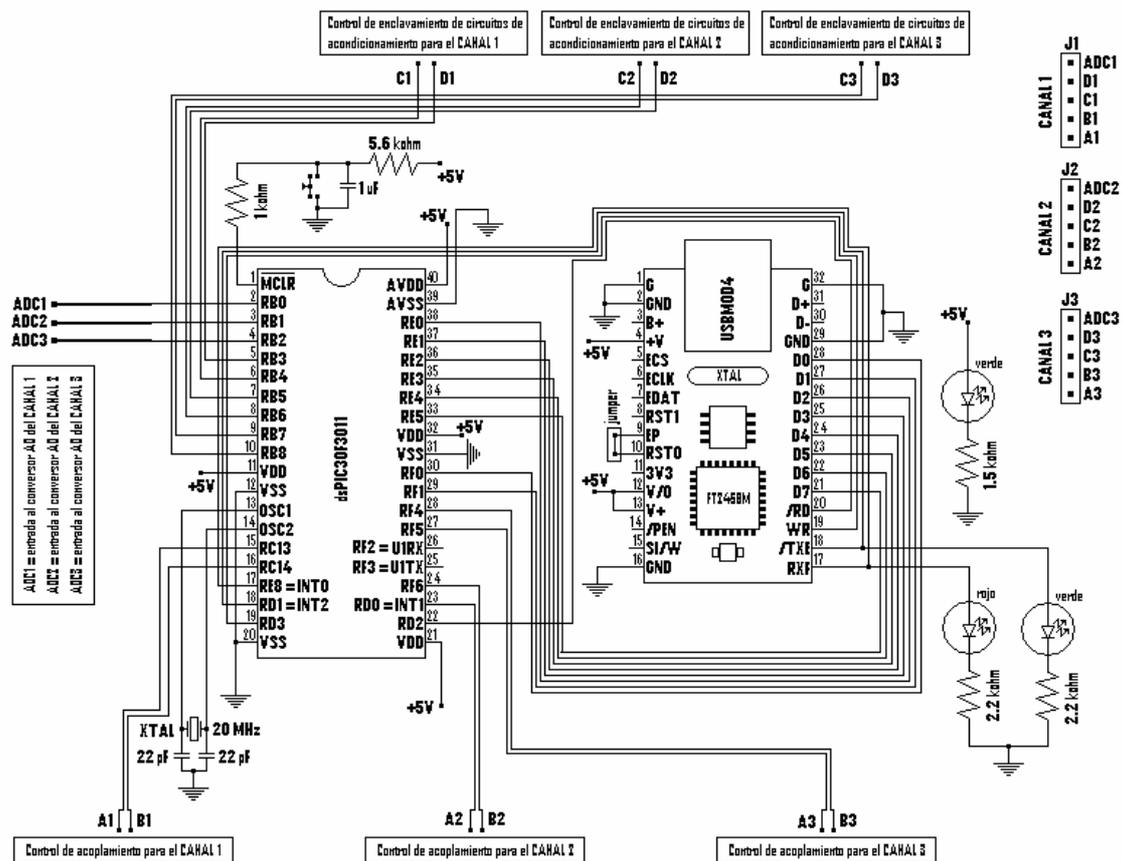


Figura 2.14 Circuito de la tarjeta de control y comunicación.

La Figura 2.14 muestra el circuito completo de la tarjeta de control y comunicación.

Una cosa adicional que vale destacar es que los valores de las resistencias de los leds indicadores de la comunicación se los debía escoger con valores iguales o mayores a 1 kohm, debido a que se comprobó que con valores menores, se consume mucha corriente, y la comunicación no funcionaba. En esta caso se escogió resistencias de 2.2 kohm.

2.4 DIAGRAMAS CIRCUITALES DEFINITIVOS

DIAGRAMA DE LA TARJETA DE CONTROL Y COMUNICACIÓN.

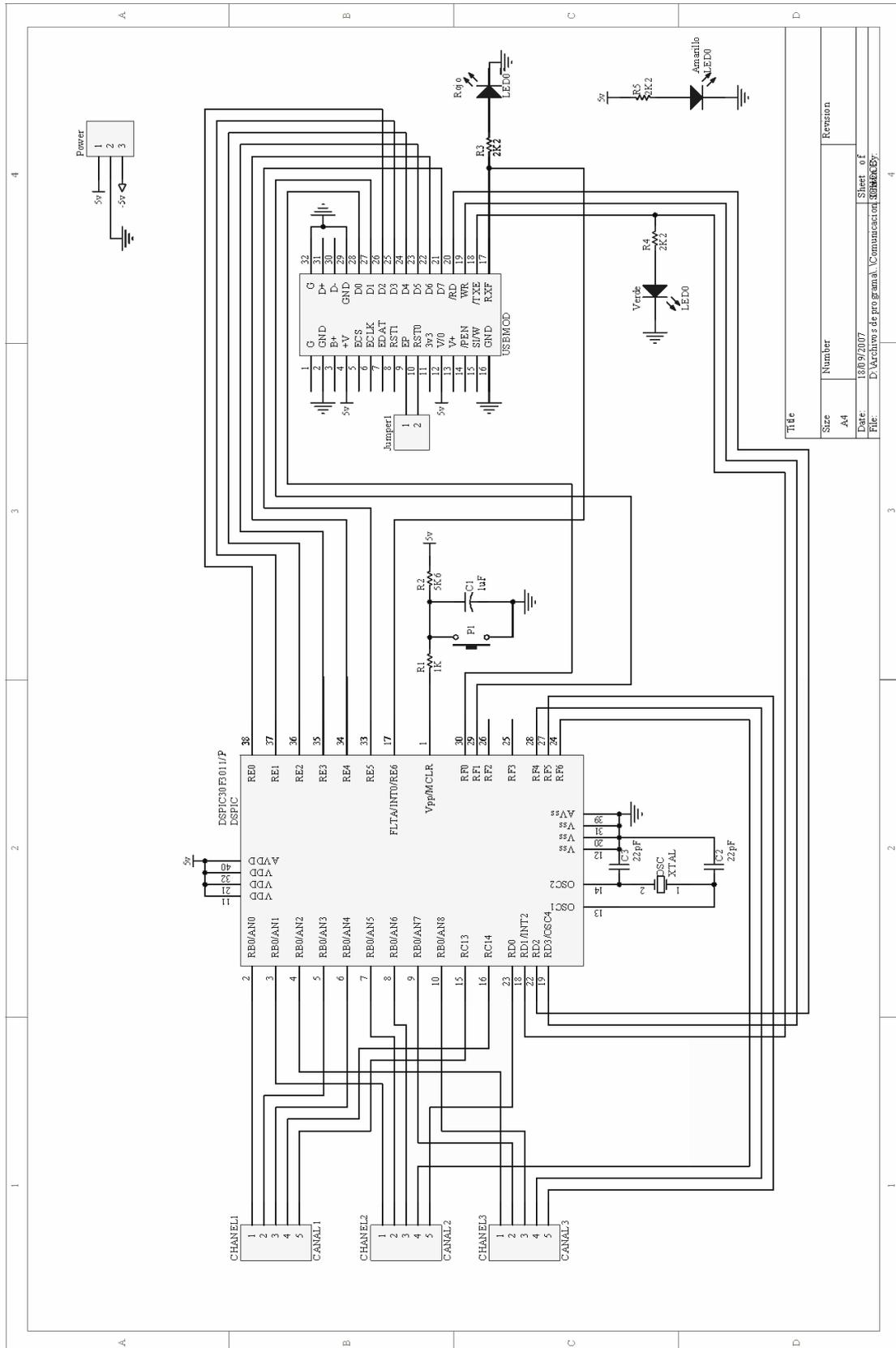
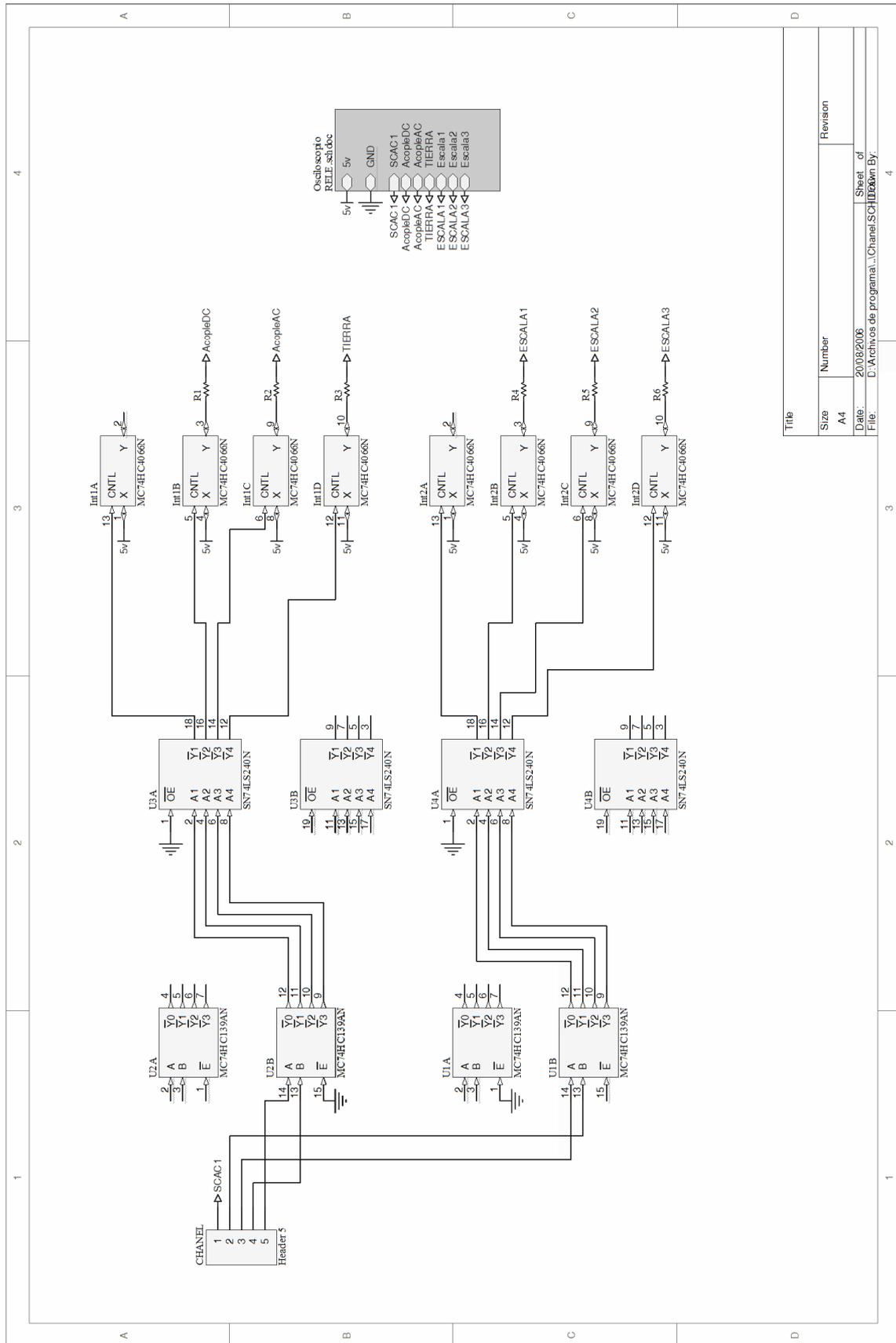
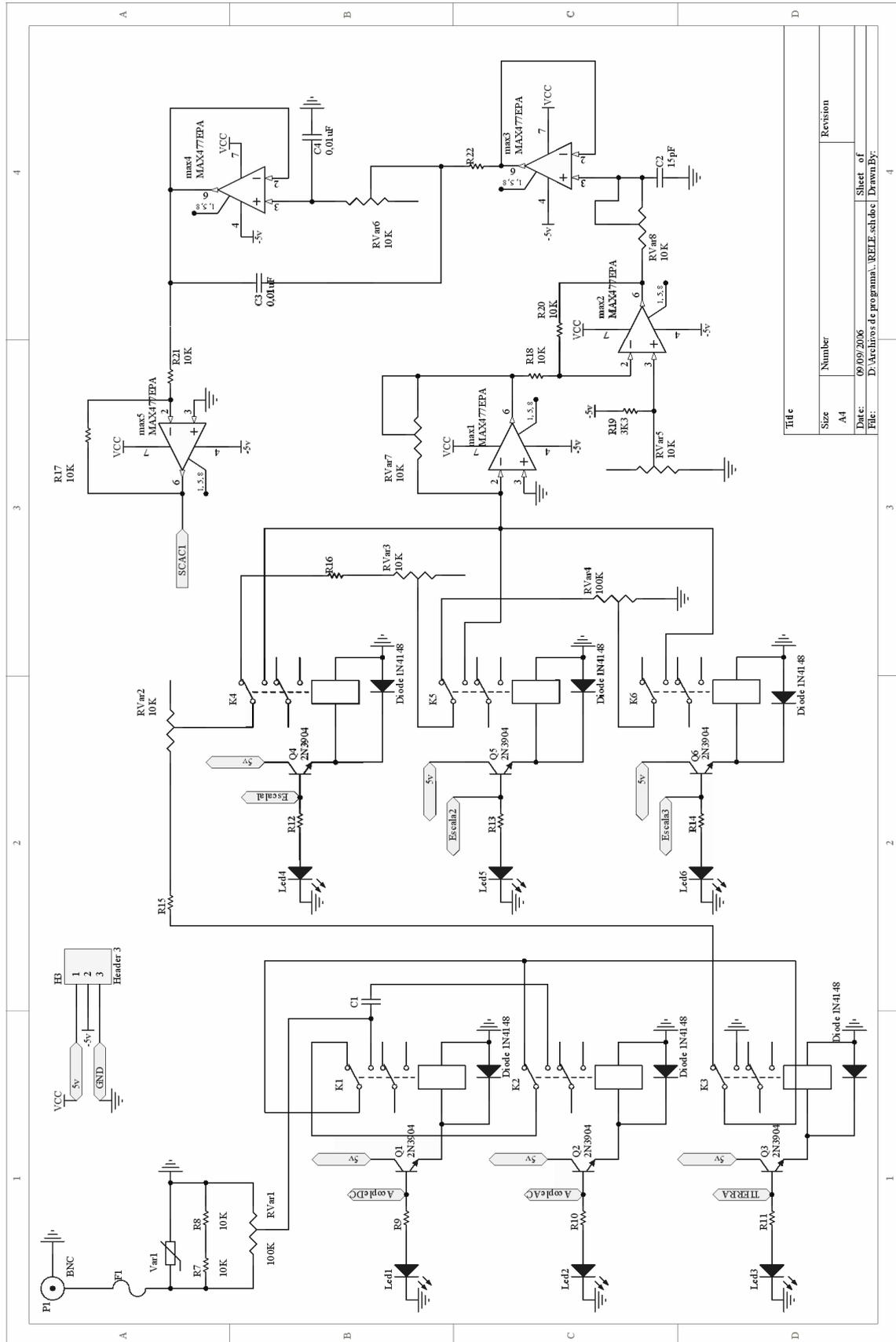


DIAGRAMA DEL CIRCUITO DE ACCIONAMIENTOS PARA CADA CANAL.



| | |
|------------|---|
| Title | |
| Size | Number |
| A4 | |
| Date: | Revision |
| 20/09/2006 | |
| Sheet of | File: |
| 4 | D:\Archivos de programa\1\Channel\SCH\HD060n By |

DIAGRAMA DEL CIRCUITO DE ACONDICIONAMIENTO Y FILTRADO DE CADA CANAL.



2.5 CONSUMO DE POTENCIA

2.5.1 REQUERIMIENTOS DE ENERGÍA

Se requirió de una fuente que provee niveles de -5V, 0V y +5V, para la alimentación de todas las etapas. Para esto se estudiaron 2 posibilidades; con fuente externa, o solo con el voltaje de +5V que provee el puerto USB de la PC.

Estudiando cuanto va a consumir tanto la etapa de control y comunicación, así como las de cada canal, se vio que la solución es una fuente externa (tomando en cuenta que un puerto USB es capaz que suministrar un máximo de 500 mA de corriente por cada dispositivo conectado al bus).

2.5.2 CÁLCULO DEL CONSUMO DE POTENCIA

Para decidir si es necesario alimentar la placa con una fuente externa (o si basta con la potencia entregada por el puerto USB) se debe estudiar el consumo de los diferentes componentes activos que integran el osciloscopio, los cuales se presentan en la Tabla 2.5 (todos los valores están en mA).

En esta tabla se incluyen la tarjeta de control y comunicación, y, también las tarjetas de los 3 canales, pero se omitieron algunos componentes cuyo consumo es despreciable (74HCT240, 74HCT139).

| ELEMENTO | CONSUMO DE CORRIENTE POR UNIDAD | | CANTIDAD | CONSUMO DE CORRIENTE TOTAL | |
|--------------|---------------------------------|--------|--------------|----------------------------|--------|
| | mA | | | mA | |
| | TÍPICA | MÁXIMA | | TÍPICA | MÁXIMA |
| dsPIC30F3011 | 200 | 250 | 1 | 200 | 250 |
| USBMOD4 | 400 | 500 | 1 | 400 | 500 |
| MAX477 | 100 | 120 | 15 | 1500 | 1800 |
| 74HC4066 | 20 | 20 | 6 | 120 | 120 |
| Relés de 5V | 100 | 100 | 6 | 600 | 600 |
| | | | TOTAL | 2820 | 3270 |

Tabla 2.5 Consumo de potencia de componentes activos.

Cabe indicar que en el caso de los relés, a pesar que en total son 18, para el consumo de energía se toman en cuenta solo 6, ya que siempre en cada uno de los tres canales, estarán activados solo 2 relés, uno para el acondicionamiento y otro para el acoplamiento.

A partir de los valores máximos totales de la Tabla 2.5 se puede concluir que para una alimentación de 5V el consumo de potencia máximo del osciloscopio es aproximadamente de 16 watts.

$$P_{\max} = 5V * I_{\max} = 5 [V] * 3.27 [A]; \quad P_{\max} = 16.35 [watts]$$

Ya habiendo descrito el diseño del hardware del osciloscopio para PC, en el siguiente capítulo se tratará todo lo que se refiere al diseño del Firmware (programa del dsPIC) del osciloscopio.

CAPÍTULO 3

DESARROLLO DEL PROGRAMA DEL dsPIC30F3011 (FIRMWARE)

DESARROLLO DEL PROGRAMA DEL DSPIC30F3011

3.1 INTRODUCCIÓN AL DESARROLLO DEL FIRMWARE

Para el presente proyecto se escogió un circuito integrado específico que es un controlador digital de señales, el **dsPIC30F3011**, por sus características de tratamiento de señales y por la velocidad de conversión de su módulo conversor AD. Para este microcontrolador, se utilizó la herramienta de programación en lenguaje C, MPLAB C30.

A continuación se describe brevemente las características de esta herramienta de programación, y se detalla el programa para el **dsPIC** que se diseñó para el control y funcionamiento del osciloscopio.

3.2 HERRAMIENTA DE PROGRAMACIÓN MPLAB C30

El programa que controla todo el funcionamiento del osciloscopio (en base a órdenes que se le envía desde la interfaz de la PC), se diseñó en lenguaje C con la ayuda de la herramienta MPLAB C30.

El MPLAB C30 es una herramienta de programación bajo la norma ANSI x3. 159-1989 que incluye una extensión en lenguaje C para el compilador del programa MPLAB IDE 6.30 o superior (que normalmente solo permite desarrollar código en lenguaje Assembler), brindando aplicaciones de control para dsPICs y DSC (Digital Signal Controllers).

El compilador funciona como aplicación de Windows que provee una plataforma para el diseño en código C. Este compilador es parte de los compiladores GCC (GNU Compiler Collection) de la FSF (Free Software Foundation).

3.3 IDENTIFICACIÓN DE TODAS LAS ÓRDENES EMITIDAS DESDE LA PC, CON LA INTERRUPCIÓN EXTERNA INT0.

Como se mencionó en el Literal 3.2, todo el funcionamiento de la comunicación y sus diferentes modos, es controlado por el dsPIC, pero todo en base a órdenes que le envía la PC. Un número de 8 bits (un byte) distingue a cada orden.

Al dsPIC le pueden llegar 46 comandos (órdenes) diferentes más 2 bytes (LSB y MSB) que corresponden al valor del nivel del trigger. Por lo mismo, el dsPIC debe poder identificar cual de los 46 comandos está llegando, o si se trata del valor del nivel del trigger. Para realizar esto se diseñó la rutina que se muestra en las Figuras 3.1 y 3.2.

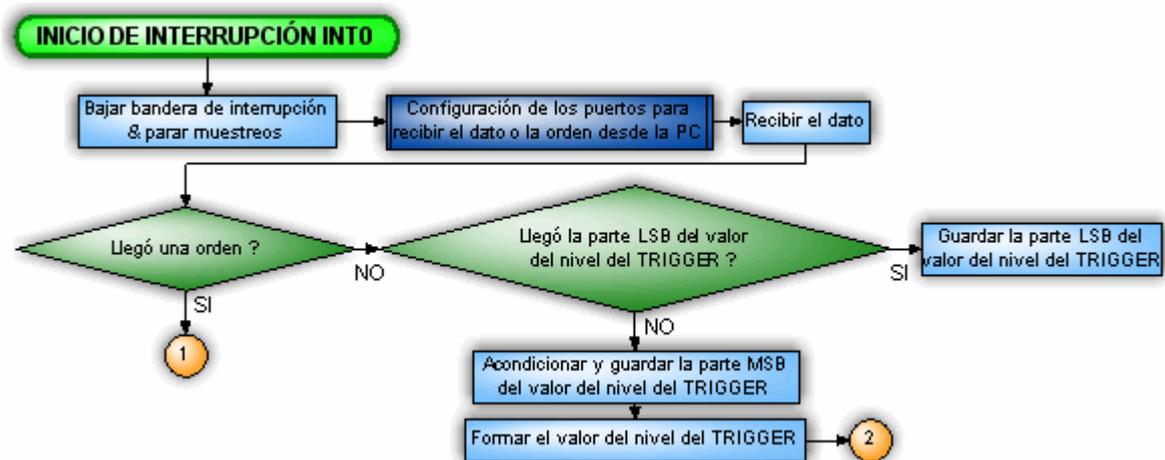


Figura 3.1 Inicio de la interrupción INT0.

Cada vez que la PC envía un byte hacia el módulo, primero lo recepta el USBMOD4, el cual genera un cambio de estado lógico en /TXE (pin 17) que está conectado a la interrupción externa INT0 del dsPIC, por lo que se genera una interrupción en el mismo. Al iniciarse la rutina de interrupción, el dsPIC toma el byte del respectivo puerto y lo identifica de entre 46 (bytes) posibilidades de que sea una orden, o que sea uno de los 2 bytes que corresponden a las 2 partes del valor del trigger en el modo manual.

A continuación se explica en lenguaje estructurado el algoritmo de inicio de aplicación de la interrupción de recepción INT0

Inicio de interrupción INT0:**Bajar la bandera de interrupción y parar muestreos:**

Encerar la bandera de la interrupción INT0

Parar muestreos

Configuración de los puertos para recibir la orden o el dato desde la PC:

Configurar los puertos PORTE y PORTF para poder receptar los 8 bits de la orden o el dato desde la PC (en este punto ya están en el puerto paralelo -D0 a D7- del USBMOD4).

Recibir el dato:

Poner en 1 el pin de recepción (/RD) del USBMOD4: LATDbits.LATD2 = 1. Para que el dato esté disponible automáticamente en los puertos descritos anteriormente.

Llegó una orden?:

Si llegó una orden:

Empezar a identificar de cual se trata

Si No llegó una orden:

Revisar que parte del nivel del trigger llegó

Llegó la parte LSB del valor del nivel del TRIGGER?:

Si llegó:

Guardar la parte LSB del valor del nivel del TRIGGER:

Almacenar el dato

Reconfigurar los puertos como salidas

Salir de la interrupción

Si No llegó:

Acondicionar y guardar la parte MSB del valor del nivel del TRIGGER:

Acondicionar y guardar la parte más significativa del valor del nivel del TRIGGER

Formar el valor del nivel del TRIGGER:

Formar el valor del nivel del TRIGGER sumando las dos partes guardadas previamente

Reconfigurar los puertos como salidas

Salir de la interrupción

Fin del algoritmo.

En la Figura 3.2 se muestra brevemente la forma en que se detectan y aplican la primera y última órdenes de control del osciloscopio, a continuación se detalla en lenguaje estructurado, el algoritmo completo de las 46 órdenes.

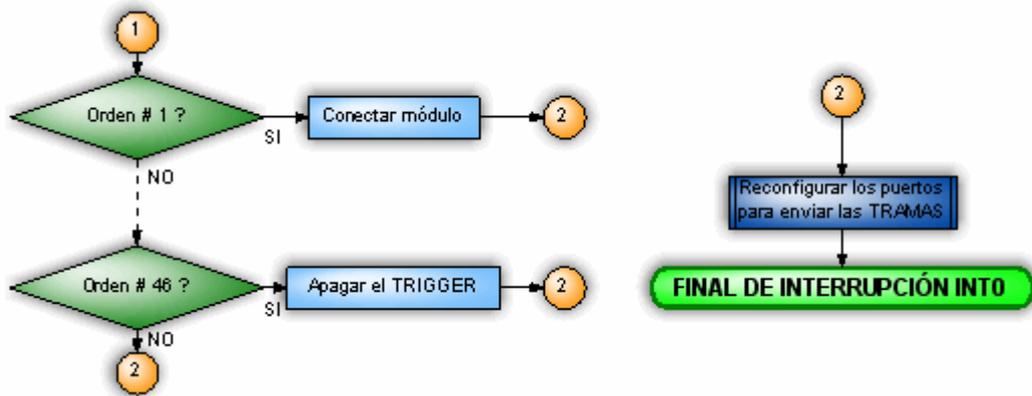


Figura 3.2 Identificación de todas las órdenes.

Algoritmo de discernimiento y aplicación de las 46 órdenes de control

Orden # 1?

Si es ésta orden:

Conectar módulo:

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 2?

Si es ésta orden:

Desconectar módulo:

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 3?

Si es ésta orden:

Accionar la escala máxima del canal 1:

Enclavar circuito de escala máxima del canal 1

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 4?

Si es ésta orden:

Accionar la escala media del canal 1:

Enclavar circuito de escala media del canal 1

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 5?

Si es ésta orden:

Accionar la escala mínima del canal 1:

Enclavar circuito de escala mínima del canal 1

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 6?

Si es ésta orden:

Accionar la escala máxima del canal 2:

Enclavar circuito de escala máxima del canal 2

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 7?

Si es ésta orden:

Accionar la escala media del canal 2:

Enclavar circuito de escala media del canal 2

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 8?

Si es ésta orden:

Accionar la escala mínima del canal 2:

Enclavar circuito de escala mínima del canal 2

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 9?

Si es ésta orden:

Accionar la escala máxima del canal 3:

Enclavar circuito de escala máxima del canal 3

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 10?

Si es ésta orden:

Accionar la escala media del canal 3:

Enclavar circuito de escala media del canal 3

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 11?

Si es ésta orden:

Accionar la escala mínima del canal 3:

Enclavar circuito de escala mínima del canal 3

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 12?

Si es ésta orden:

Aplicar acoplamiento GND del canal 1:

Conectar a tierra la entrada del circuito acondicionador del canal 1

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 13?

Si es ésta orden:

Aplicar acoplamiento DC del canal 1:

Conectar la entrada BNC del canal 1 a la entrada de su circuito acondicionador

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 14?

Si es ésta orden:

Aplicar acoplamiento AC del canal 1:

Conectar la entrada BNC del canal 1 a través de un capacitor hacia su circuito acondicionador

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 15?

Si es ésta orden:

Aplicar acoplamiento GND del canal 2:

Conectar a tierra la entrada del circuito acondicionador del canal 2

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 16?

Si es ésta orden:

Aplicar acoplamiento DC del canal 2:

Conectar la entrada BNC del canal 2 a la entrada de su circuito acondicionador
Reconfigurar puertos
Finalizar interrupción INT0

No es ésta orden:

Orden # 17?

Si es ésta orden:

Aplicar acoplamiento AC del canal 2:

Conectar la entrada BNC del canal 2 a través de un capacitor hacia su circuito acondicionador
Reconfigurar puertos
Finalizar interrupción INT0

No es ésta orden:

Orden # 18?

Si es ésta orden:

Aplicar acoplamiento GND del canal 3:

Conectar a tierra la entrada del circuito acondicionador del canal 3
Reconfigurar puertos
Finalizar interrupción INT0

No es ésta orden:

Orden # 19?

Si es ésta orden:

Aplicar acoplamiento DC del canal 3:

Conectar la entrada BNC del canal 3 a la entrada de su circuito acondicionador
Reconfigurar puertos
Finalizar interrupción INT0

No es ésta orden:

Orden # 20?

Si es ésta orden:

Aplicar acoplamiento AC del canal 3:

Conectar la entrada BNC del canal 3 a través de un capacitor hacia su circuito acondicionador
Reconfigurar puertos
Finalizar interrupción INT0

No es ésta orden:

Orden # 21?

Si es ésta orden:

Aplicar escala de 500 ms/DIV

Aplicar modo de toma de datos en tiempo real

Reconfigurar puertos
Finalizar interrupción INT0

No es ésta orden:

Orden # 22?

Si es ésta orden:

Aplicar escala de 200 ms/DIV

Aplicar modo de toma de datos en tiempo real

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 23?

Si es ésta orden:

Aplicar escala de 100 ms/DIV

Aplicar modo de toma de datos en tiempo real

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 24?

Si es ésta orden:

Aplicar escala de 50 ms/DIV

Aplicar modo de toma de datos en tiempo real

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 25?

Si es ésta orden:

Aplicar escala de 20 ms/DIV

Aplicar modo de toma de datos en tiempo real

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 26?

Si es ésta orden:

Aplicar escala de 10 ms/DIV

Aplicar modo de toma de datos en tiempo real

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 27?

Si es ésta orden:

Aplicar escala de 5 ms/DIV

Aplicar modo de toma de datos en tiempo real

Reconfigurar puertos
Finalizar interrupción INT0

No es ésta orden:

Orden # 28?

Si es ésta orden:

Aplicar escala de 2 ms/DIV
Aplicar modo de toma de datos en tiempo real
Reconfigurar puertos
Finalizar interrupción INT0

No es ésta orden:

Orden # 29?

Si es ésta orden:

Aplicar escala de 1 ms/DIV
Aplicar modo de toma de datos en tiempo real
Reconfigurar puertos
Finalizar interrupción INT0

No es ésta orden:

Orden # 30?

Si es ésta orden:

Aplicar escala de 0.5 ms/DIV
Aplicar modo de toma de datos en tiempo real
Reconfigurar puertos
Finalizar interrupción INT0

No es ésta orden:

Orden # 31?

Si es ésta orden:

Aplicar escala de 0.2 ms/DIV
Aplicar modo de toma de datos en tiempo real
Reconfigurar puertos
Finalizar interrupción INT0

No es ésta orden:

Orden # 32?

Si es ésta orden:

Aplicar escala de 0.1 ms/DIV
Aplicar modo de toma de datos de almacenamiento
Reconfigurar puertos
Finalizar interrupción INT0

No es ésta orden:

Orden # 33?

Si es ésta orden:

Aplicar escala de 50 μ s/DIV

Aplicar modo de toma de datos de almacenamiento

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 34?

Si es ésta orden:

Aplicar escala de 20 μ s/DIV

Aplicar modo de toma de datos de almacenamiento

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 35?

Si es ésta orden:

Aplicar escala de 10 μ s/DIV

Aplicar modo de toma de datos de almacenamiento

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 36?

Si es ésta orden:

Apagar osciloscopio

Detener el conversor AD

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 37?

Si es ésta orden:

Encender osciloscopio

Arrancar el conversor AD

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 38?

Si es ésta orden:

Resetear el número de tramas enviadas hacia la PC

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 39?

Si es ésta orden:

Indicar que va a llegar la parte LSB del valor del nivel del trigger

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 40?

Si es ésta orden:

Aplicar trigger con respecto al canal 1

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 41?

Si es ésta orden:

Aplicar trigger con respecto al canal 2

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 42?

Si es ésta orden:

Aplicar trigger con respecto al canal 3

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 43?

Si es ésta orden:

Aplicar trigger en flanco de subida

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 44?

Si es ésta orden:

Aplicar trigger en flanco de bajada

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 45?

Si es ésta orden:

Encender el trigger

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Orden # 46?

Si es ésta orden:

Encender el trigger

Encerar el número de tramas a enviarse a la PC

Reconfigurar puertos

Finalizar interrupción INT0

No es ésta orden:

Reconfigurar puertos

Finalizar interrupción INT0

Fin del algoritmo.

Para las órdenes de acoplamiento de los 3 canales, se aplican niveles lógicos en las salidas RC13 (pin 15) y RC14 (pin 16) para el canal 1, RD0 (pin 23) y RF6 (pin 24) para el canal 2 y, RF5 (pin 27) y RF4 (pin 28) para el canal 3.

Las órdenes de aplicación de las escalas de tiempo más grandes, desde 0.2 ms/DIV hasta 500 ms/DIV tienen una particularidad; desde la escala de 0.2 ms/DIV hasta la del 5 ms/DIV se mantiene la máxima velocidad de muestreo en tiempo real (aprox. 21.74µs/muestra), mientras que para las escalas desde 10 ms/DIV hasta la de 500 ms/DIV se aplica un retardo entre muestra y muestra, para obtener una velocidad de muestreo efectiva mucho menor (1.25ms/muestra). Esto se tuvo que hacer debido a que el buffer de recepción del puerto USB de la PC y la pila de eventos del Visual Basic, se desbordan cuando se tratan de dibujar más de 4000 muestras por pantalla. Por ejemplo, para la escala de 10 ms/DIV, se hubiera tenido que graficar 4600 muestras en pantalla a velocidad de muestreo normal en tiempo real, lo cual supera el máximo permisible por el buffer del puerto y la pila de eventos cuando se trabaja con el MSCComm en el Visual Basic 6.0.

Para las escalas más pequeñas de tiempo, desde 10 µs/DIV hasta 0.1 ms/DIV se activan el modo de almacenamiento, que tiene una velocidad de muestreo aproximada de 11 µs/muestra.

3.4 CONEXIÓN Y ENCENDIDO DEL DISPOSITIVO

El algoritmo mostrado en la Figura 3.3 muestra el procedimiento para conectar, desconectar, encender y apagar al osciloscopio.

Al iniciar la conexión, la PC envía un número varias veces a todos los puertos COM posibles, o al seleccionado en forma manual (ver Capítulo 4). Entonces, si ya se ha conectado el módulo a la PC, el dsPIC al recibir solo dicho número, enviará como respuesta otro número hacia la PC, con lo cual la PC ha reconocido la existencia del módulo y se considera que ya inició la comunicación con el mismo, caso contrario, simplemente terminará un número definido de búsquedas y se quedará en espera hasta que se vuelva a iniciar la búsqueda manualmente y la PC encuentre el módulo cuando este finalmente sea conectado. Después de haber reconocido al módulo y establecer ya comunicación, se queda el programa en un lazo esperando que el usuario encienda el osciloscopio, luego de lo cual se sale de dicho lazo e inicia el programa principal. Dentro de dicho lazo volverá a permanecer el programa, cuando el usuario apague el osciloscopio.

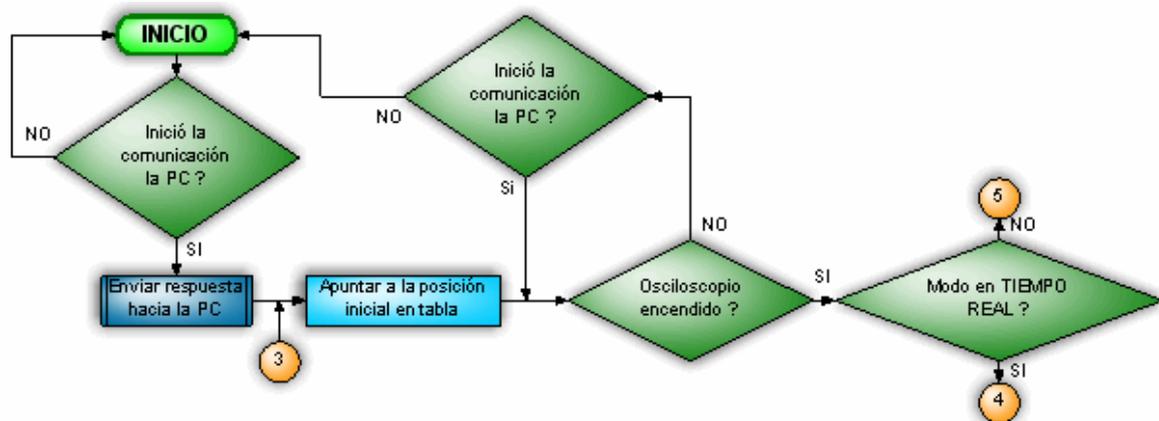


Figura 3.3 Conexión, desconexión, encendido y apagado del osciloscopio.

A continuación se presenta en lenguaje estructurado el algoritmo de conexión, desconexión, encendido y apagado del osciloscopio

INICIO:

Inició la comunicación la PC?

Si inició la comunicación:

Enviar respuesta hacia la PC

Apuntar a la posición inicial en tabla

Osciloscopio encendido?

Si está encendido:

Modo en TIEMPO REAL?

Si es éste modo:

Adquirir datos en modo de tiempo real

No es éste modo:

Adquirir datos en modo de almacenamiento

No está encendido:

Inició la comunicación la PC?

Si inició la comunicación:

Reviso nuevamente el encendido del osciloscopio

No inició la comunicación:

Retornar al INICIO

No inició la comunicación:

Retornar al INICIO

Fin del algoritmo.

3.5 MODO EN TIEMPO REAL

En este modo cada dato de 10 bits del conversor AD se le divide en 2 partes de 5 bits. A cada parte se le coloca su respectivo encabezado (se convierten en TRAMAS de 8 bits) e inmediatamente se envía cada trama hacia la PC. Para poder comprender esta parte, se debe tomar en cuenta que al referirse a “**escalas pequeñas**”, se trata de las escalas de tiempo desde 10 μ s/DIV hasta 0.1 ms/DIV; y, al referirse a “**escalas grandes**”, se trata de las escalas de tiempo desde 0.2 ms/DIV hasta 500 ms/DIV.

Los modos de toma de datos están determinados por el valor de la variable “**TOMA_DE_DATOS**”, cuando **TOMA_DE_DATOS = 1**, se activa el modo de toma de datos en tiempo real, y cuando **TOMA_DE_DATOS = 2**, se activa el modo de toma de datos de almacenamiento, este último es el caso de las escalas pequeñas.

3.5.1 ADQUISICIÓN DE DATOS PARA MODO DE TIEMPO REAL

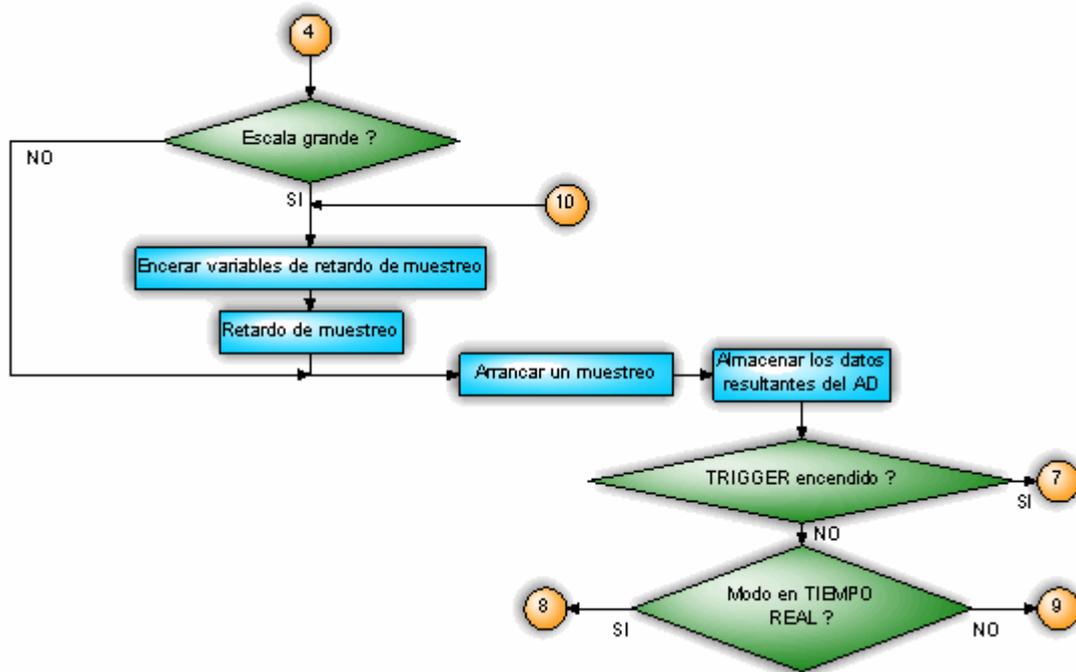


Figura 3.4 Algoritmo de adquisición de datos para modo de TIEMPO REAL.

A continuación se aplica en lenguaje estructurado el algoritmo de adquisición de datos para modo de TIEMPO REAL

Escala grande?

Si es escala grande:

Encerar variables de retardo de muestreo

Poner en cero las variables que generan el retardo entre muestreos

Retardo de muestreo:

Empieza a correr el retardo

Arrancar un muestreo:

Ordenar que empiece otro muestreo

Almacenar los datos resultantes del AD:

Almacenar dato del canal 1

Almacenar dato del canal 2

Almacenar dato del canal 3

TRIGGER encendido?

Si está encendido:

Algoritmo de comprobación previa a aplicación del trigger

No está encendido:

Modo en TIEMPO REAL?

Si es este modo:

Algoritmo de lectura y preparación de datos para modo en TIEMPO REAL (Figura 3.5)

No es este modo:

Algoritmo de lectura y preparación de datos para el modo de ALMACENAMIENTO

No es escala grande:

Arrancar un muestreo (y el algoritmo se repite como arriba, pero solo desde esta línea).

Ordenar que empiece otro muestreo

Fin del algoritmo.

3.5.2 LECTURA Y PREPARACIÓN DE DATOS PARA MODO DE TIEMPO REAL

Después de tomar cada lectura del AD, se preparan los valores previos a su acondicionamiento y envío como se muestra en la Figura 3.5. La variable “**TRAMA_C**” es la que porta las cabeceras que se añadirán a las 2 partes en que se dividirá el dato que en su momento esté guardado en la variable “**REG_AUX**”.

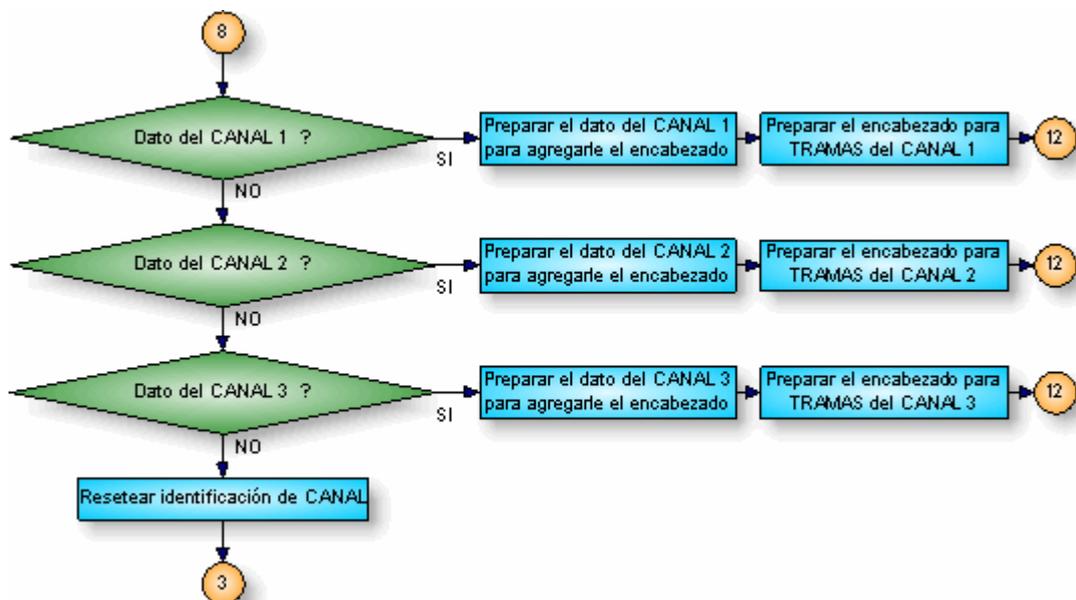


Figura 3.5 Algoritmo de lectura y preparación de datos para modo de TIEMPO REAL.

3.5.3 ACONDICIONAMIENTO Y ENVÍO DE TRAMAS PARA MODO DE TIEMPO REAL

| CANAL | Parte del dato | CABECERA | | | DATOS | | | | |
|-------|----------------|---|--|-------|---------------|-------|-------|-------|-------|
| | | Bit que indica a que parte del dato corresponde (LSB o MSB) | Bits que indican a que canal pertenece el dato | | Bits de datos | | | | |
| | | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| 1 | MSB | 1 | 0 | 0 | x | x | x | x | x |
| | LSB | 0 | 0 | 0 | x | x | x | x | x |
| 2 | MSB | 1 | 0 | 1 | x | x | x | x | x |
| | LSB | 0 | 0 | 1 | x | x | x | x | x |
| 3 | MSB | 1 | 1 | 0 | x | x | x | x | x |
| | LSB | 0 | 1 | 0 | x | x | x | x | x |

Tabla 3.1 Estructura de TRAMAS y las 6 posibilidades de la cabecera.

La Tabla 3.1 indica la estructura de las TRAMAS, y también los valores que toma la cabecera en cada caso.

La Figura 3.6 resume todo el proceso de acondicionamiento y envío de TRAMAS hacia la PC en modo de TIEMPO REAL.

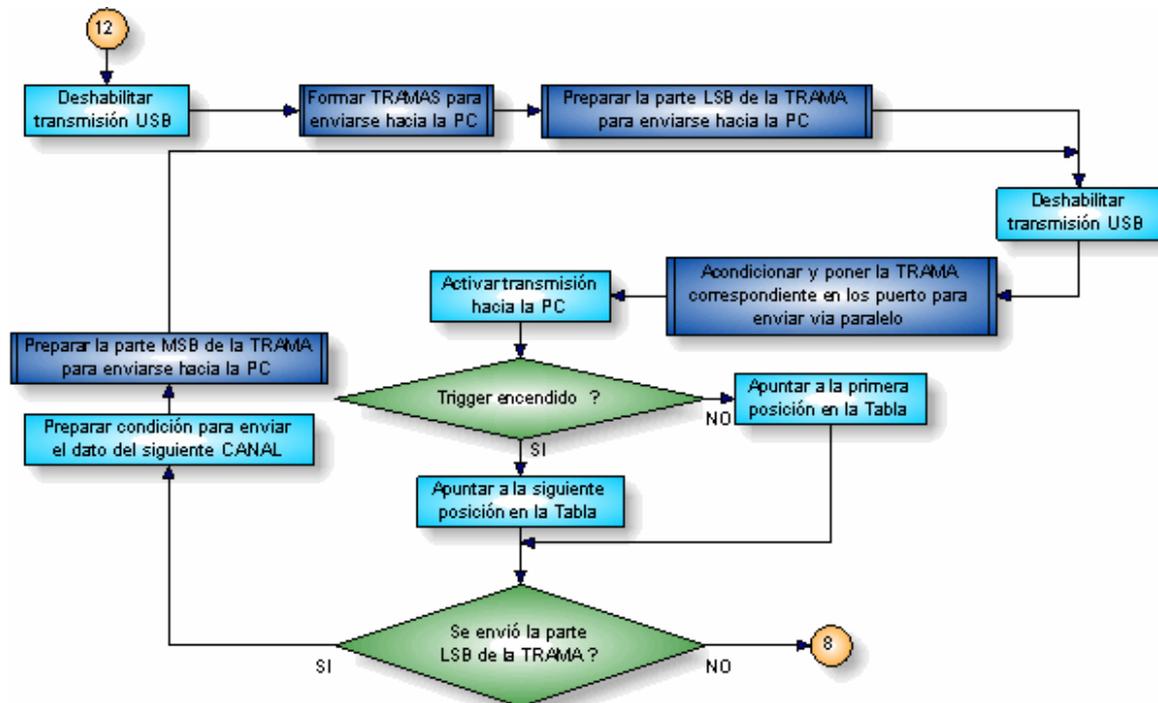


Figura 3.6 Algoritmo de acondicionamiento y envío de tramas para modo de TIEMPO REAL.

A continuación se aplica en lenguaje estructurado el algoritmo de acondicionamiento y envío de tramas para modo de TIEMPO REAL

Deshabilitar transmisión USB:

Formar TRAMAS para enviarse hacia la PC:

Se divide el dato en "REG_AUX", en dos partes (LSB y MSB) de 5 bits y con los bits de control construir la TRAMA a enviar hacia la PC:

Formar la parte menos significativa (LSB)

Formar la parte más significativa (MSB)

Preparar la parte LSB de la TRAMA para enviarse hacia la PC:

Definir variable que distingue entre las dos partes "LSB_O_MSB = 1"

Deshabilitar transmisión USB

Inhabilitar la transmisión vía USB de cualquier byte hacia la PC (Poner la salida RD3 del dsPIC en 1)

Acondicionar y poner la TRAMA correspondiente en los puertos para enviar vía paralelo

Activar transmisión hacia la PC

Habilitar la transmisión vía USB de cualquier byte hacia la PC (Poner la salida RD3 del dsPIC en 0)

Trigger encendido?

Si trigger encendido:

Apuntar a la siguiente posición en la tabla

Se envió la parte LSB de la TRAMA?

Si se envió la parte LSB de la trama:

Preparar condición para enviar el dato del siguiente canal

Preparar la parte MSB de la trama para enviarse hacia la PC

*Regresar a: **Deshabilitar transmisión USB** (y continúa igual desde ahí el algoritmo)*

No se envió la parte LSB de la trama:

Regresar a iniciar el algoritmo de lectura y preparación de datos en modo de TIEMPO REAL

Si trigger **No** está encendido:

Apuntar a la primera posición en la Tabla

*Regresa a: **Se envió la parte LSB de la TRAMA?** (y continúa igual el algoritmo desde ahí)*

Fin del algoritmo.

3.6 MODO DE ALMACENAMIENTO DE DATOS

En este modo, a diferencia del de tiempo real, cuando ya se tiene formada una trama a ser enviada a la PC, no se la envía directamente, sino que primero se la almacena en la memoria RAM del dsPIC.

El procedimiento mostrado en las Figuras 3.7 y 3.8 y 3.9 se describe muy brevemente así: Se toma un dato, se forma la trama, y en vez de enviar a la PC dicha trama, se la almacena en una tabla en memoria RAM. Inmediatamente se toma otro dato, se forma otra trama, y se la almacena, así sucesivamente hasta el momento en que se llene con el número máximo de tramas programado para almacenar en la tabla en memoria RAM, luego de lo cual las tramas almacenadas empiezan a ser enviadas hacia la PC.

3.6.1 ADQUISICIÓN DE DATOS PARA MODO DE ALMACENAMIENTO

El modo de almacenamiento requiere de otros elementos que no tiene el modo en tiempo real. Uno de esos elementos es la tabla donde se van a almacenar las tramas antes de ser enviadas hacia la PC, para lo cual se va a hacer uso de la memoria RAM del dsPIC30F3011.

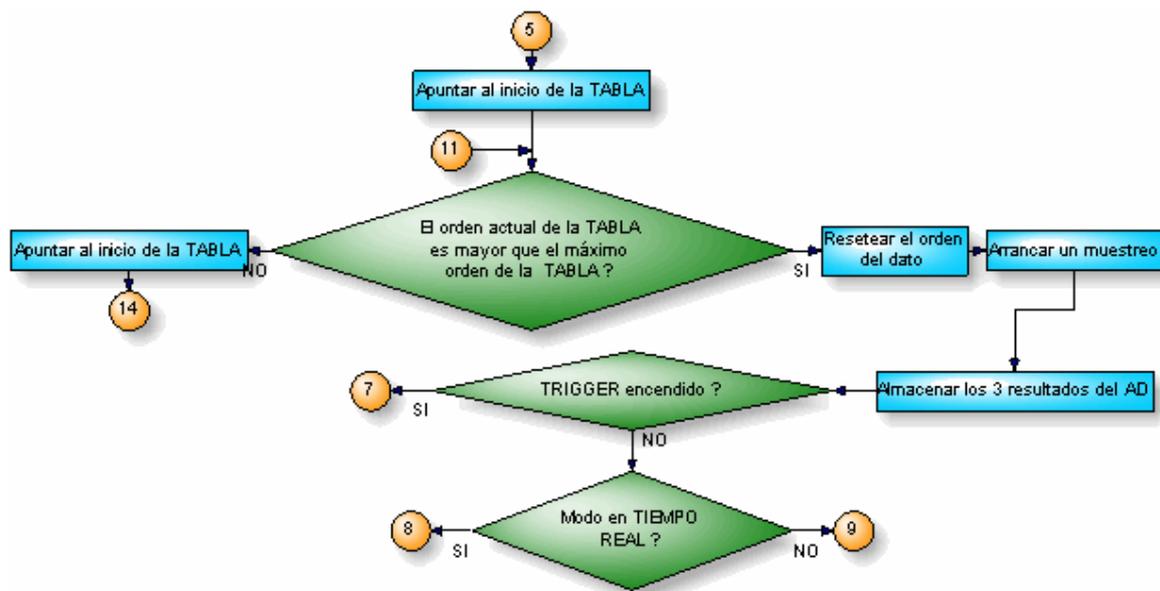


Figura 3.7 Algoritmo de adquisición de datos para modo de ALMACENAMIENTO.

A continuación se aplica en lenguaje estructurado el algoritmo de adquisición de datos para modo de ALMACENAMIENTO

Apuntar al inicio de la TABLA:

El orden actual de la TABLA es mayor que el máximo orden de la TABLA?

Si es mayor:

Resetear el orden del dato

Arrancar un muestreo:

Se empieza otro muestreo

Almacenar los 3 resultados del AD:

Almacenar el dato del canal 1

Almacenar el dato del canal 2

Almacenar el dato del canal 3

TRIGGER encendido?

Si está encendido:

Algoritmo de comprobación previo a la aplicación del trigger (Figura 3.12).

No está encendido:

Modo en TIEMPO REAL?

Si:

Algoritmo de adquisición de datos para modo en TIEMPO REAL (Figura 3.4)

No:

Algoritmo de adquisición de datos para modo de ALMACENAMIENTO (Figura 3.7)

No es mayor:

Apuntar al inicio de la TABLA

Algoritmo de envío de TRAMAS para el modo de ALMACENAMIENTO (Figura 3.10)

Fin del algoritmo.

3.6.2 LECTURA Y PREPARACIÓN DE DATOS PARA MODO DE ALMACENAMIENTO

La Figura 3.8 muestra la preparación de los datos para después transformarlos en tramas y almacenarlas, solo esta parte es análoga al procedimiento de lectura y preparación en el método de tiempo real.

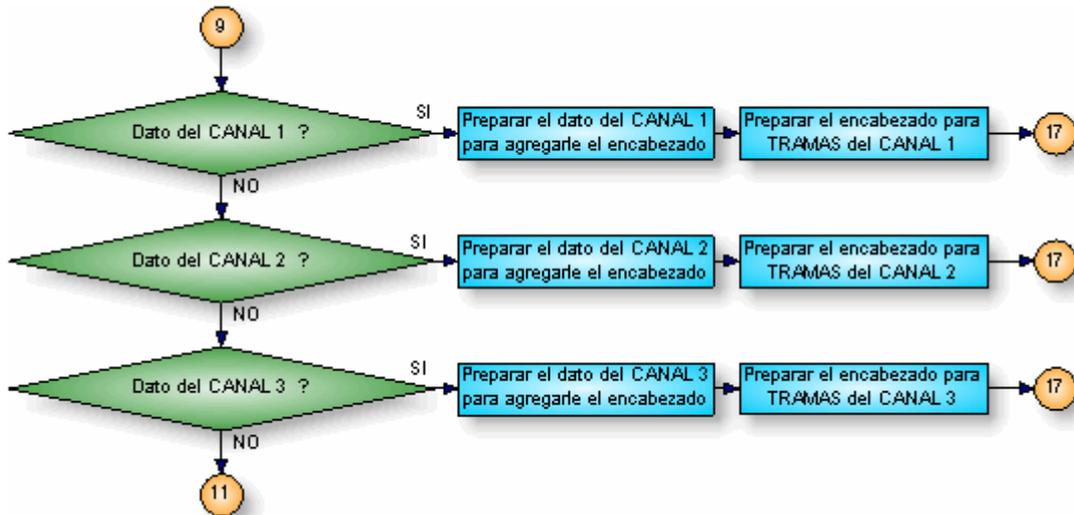


Figura 3.8 Algoritmo de lectura y preparación de datos para el modo de ALMACENAMIENTO.

3.6.3 ACONDICIONAMIENTO Y REGISTRO DE TRAMAS PARA MODO DE ALMACENAMIENTO

Para cada lectura que se toma del conversor AD, el procedimiento de acondicionamiento es el mismo que se describe en el Literal 3.5.3 hasta el momento en que se obtiene “TRAMA_LSB_CANAL” y “TRAMA_MSB_CANAL”. Después de esto, los valores de las 2 variables no se envían inmediatamente, sino que se las almacena en una tabla en memoria RAM, llamada “MATRIZ_TRAMAS”, se apunta a la siguiente posición de tabla, y salta a leer el dato del siguiente canal.

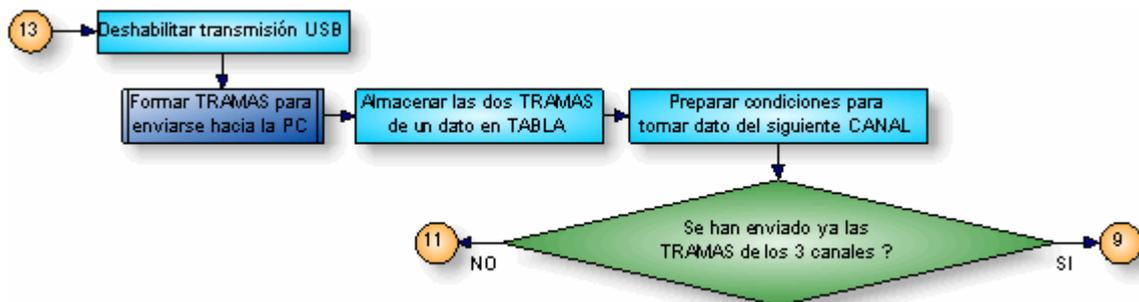


Figura 3.9 Algoritmo de acondicionamiento y registro de tramas para modo de ALMACENAMIENTO.

A continuación se aplica en lenguaje estructurado el algoritmo de acondicionamiento y registro de TRAMAS para modo de ALMACENAMIENTO

Deshabilitar transmisión USB:

Inhabilitar la transmisión vía USB de cualquier byte hacia la PC (Poner la salida RD3 del dsPIC en 1)

Formar TRAMAS para enviarse hacia la PC:

Se divide el dato en "REG_AUX", en dos partes (LSB y MSB) de 5 bits y con los bits de control se obtiene la TRAMA a enviar hacia la PC, estas dos partes son:

Formar la parte menos significativa (LSB)

Formar la parte más significativa (MSB)

Almacenar las dos TRAMAS de un dato en TABLA:

Almacenar las dos partes en la tabla "MATRIZ_TRAMAS []" en posiciones seguidas en la matriz, dadas por la variable "POSICION")

Preparar condiciones para tomar dato del siguiente CANAL:

Preparar para recoger el dato del siguiente canal

Se han enviado ya las TRAMAS de los 3 canales?

Si se enviaron:

Leer y preparar datos para el modo de ALMACENAMIENTO (Figura 3.8)

No se enviaron:

Adquirir datos para modo de ALMACENAMIENTO (Figura 3.7)

No resetear la posición en tabla

Fin del algoritmo.

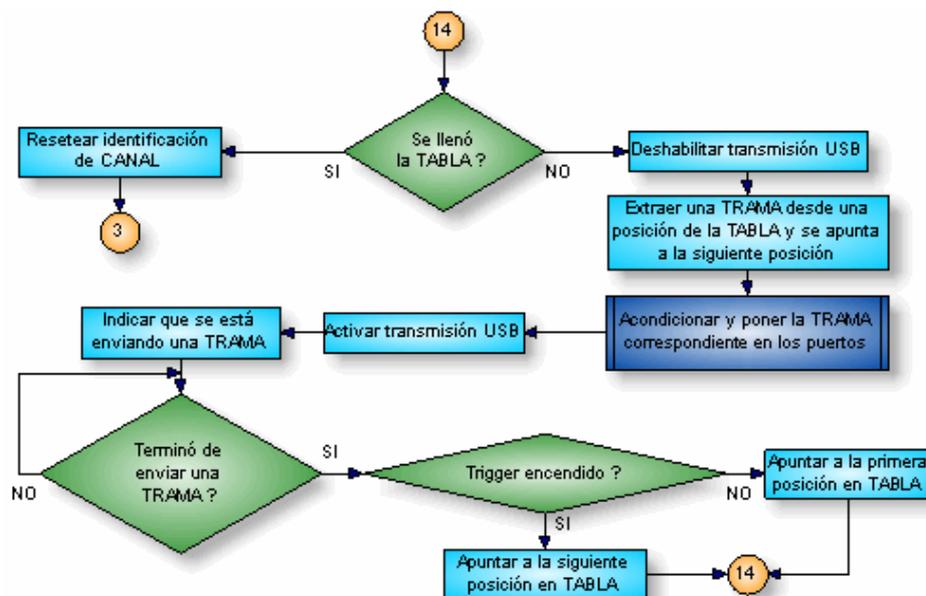
3.6.4 ENVÍO DE TRAMAS PARA EL MODO DE ALMACENAMIENTO

Figura 3.10 Algoritmo de envío de tramas para el modo de ALMACENAMIENTO.

En el algoritmo de la Figura 3.10 se ve el proceso de envío de tramas. Recordar que la frase, “la **TABLA**”, se refiere a la matriz “MATRIZ_TRAMAS[]”

A continuación se aplica en lenguaje estructurado el algoritmo de envío de tramas para el modo de ALMACENAMIENTO.

No se llena todavía la tabla?

No se llenó:

Deshabilitar transmisión USB:

Inhabilitar la transmisión vía USB de cualquier byte hacia la PC. Poner la salida RD3 del dsPIC en 1)

Extraer una trama desde una posición de la TABLA y se apunta a la siguiente posición:

Extraer la TRAMA en orden.

Guardarla en el registro “TRAMA”.

Prepara para extraer la siguiente

Acondicionar y poner la TRAMA correspondiente en los puertos

Activar transmisión USB:

Habilitar la transmisión vía USB de cualquier byte hacia la PC. Poner la salida RD3 del dsPIC en 0)

Indicar que se está enviando una TRAMA

Terminó de enviar una TRAMA?

Si terminó:

TRIGGER encendido?

Si:

Apuntar a la siguiente posición en TABLA

Regresar a comprobar: Orden de la tabla >= Máximo orden de la tabla?

No:

Apuntar a la primera posición en TABLA

Avanza una posición en la TABLA

Regresar a comprobar: No se llena todavía la tabla?

No terminó:

Regresar a comprobar: Terminó de enviar una TRAMA?

Si se llenó:

Resetear identificación de canal:

Identificar el canal 1

Fin del algoritmo.

3.6.5 USO DE LA INTERRUPCIÓN EXTERNA INT2 DEL DSPIC PARA EL MODO DE ALMACENAMIENTO

Al terminar de enviarse una trama desde el dsPIC hacia la PC, se genera la interrupción de transmisión indicada en la Figura 3.11.



Figura 3.11 Confirmación de envío de cada trama en el modo de ALMACENAMIENTO.

Se genera un cambio de estado lógico en la salida /TXE (pin 18) del USBMOD4, el cual está conectado a la interrupción externa INT2 del dsPIC. Con esto se confirma el envío correcto de cada trama. También el programa principal para salir de su lazo de impedimento de adquisición de nuevos datos, identifica que ya se ha enviado una TRAMA, haciendo la variable "ENVIO_TRAMA = 1".

3.7 ALGORITMO DEL TRIGGER

Después de los algoritmos de adquisición de datos en ambos modos (tiempo real y almacenamiento) se inicia el algoritmo del trigger (Figura 3.12).

En el caso de este algoritmo, **solo se detallará en lenguaje estructurado la comprobación previa para el canal 1**, ya que es análoga para los otros dos canales.

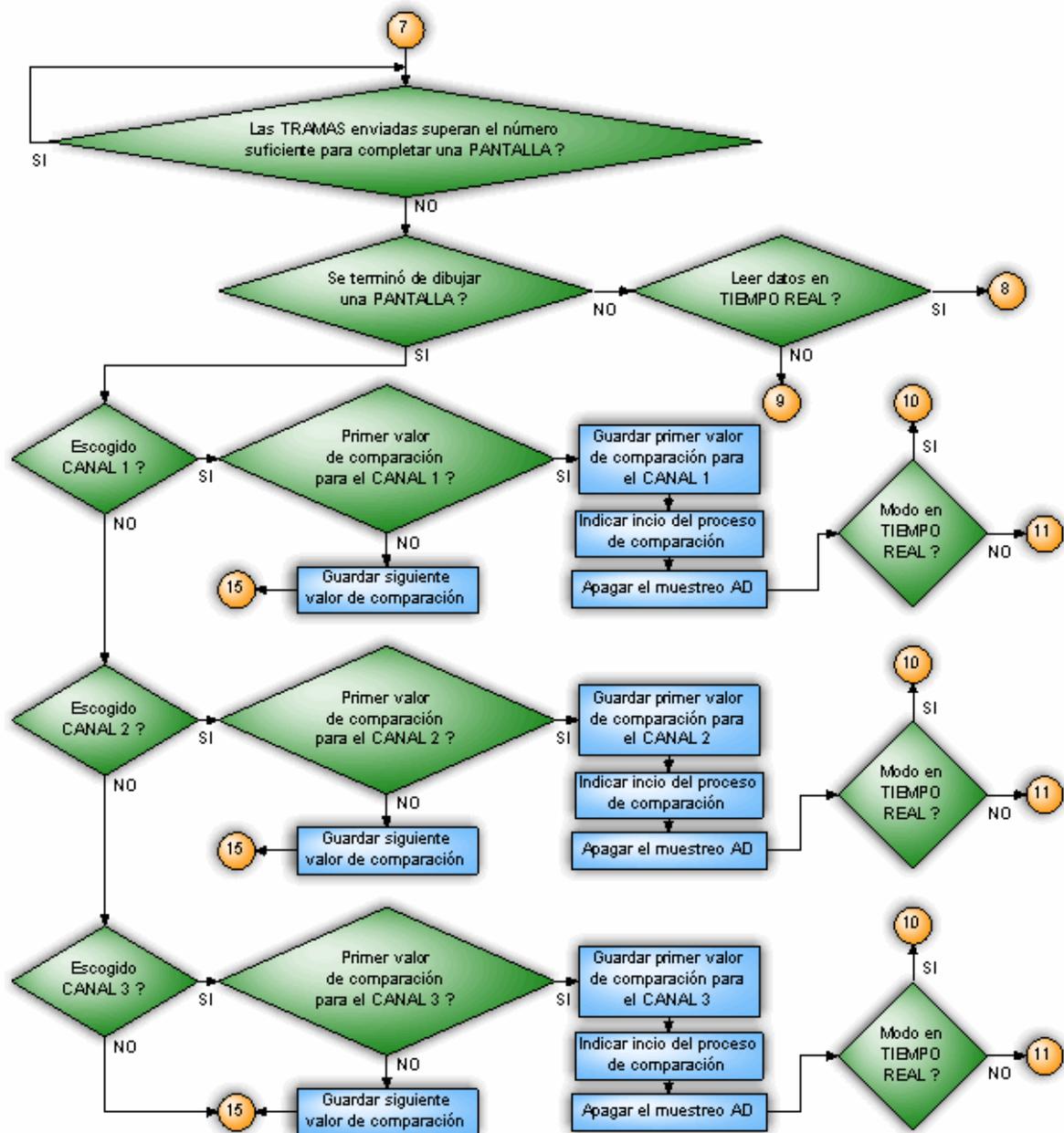


Figura 3.12 Algoritmo de comprobación previa a la aplicación del trigger.

A continuación se aplica en lenguaje estructurado el algoritmo de comprobación previo a la aplicación del trigger

Las TRAMAS enviadas superan el número suficiente para completar una PANTALLA?

Si superan el número suficiente:

Empezar de nuevo.

No superan el número suficiente:

Se terminó de dibujar una pantalla?

(Esto lo indica la PC, enviando una orden cuya variable de identificación en el programa principal es "FIN_PANTALLA")

Si se terminó:

(FIN_PANTALLA = 1)

Escogido CANAL 1?

Si:

Primer valor de comparación para el canal 1?

Si:

Guardar primer valor de comparación para el canal 1:

Almacenar en "VALOR1" el dato del AD

Indicar el inicio del proceso de comparación

Apagar el muestreo AD:

Encerar bit SAMP del registro ADCON

Saltar al algoritmo de adquisición de datos según el modo en el que esté trabajando el osciloscopio (TIEMPO REAL o ALMACENAMIENTO)

No:

Guardar siguiente valor de comparación:

Almacenar en "VALOR2" el dato del AD

Aplicar trigger (Figura 3.13)

No se terminó:

Leer datos en TIEMPO REAL?

Si:

Ejecutar algoritmo de lectura y preparación de datos para modo de TIEMPO REAL (Figura 3.5)

No:

Ejecutar algoritmo de lectura y preparación de datos para modo de ALMACENAMIENTO (Figura 3.8)

Fin del algoritmo.

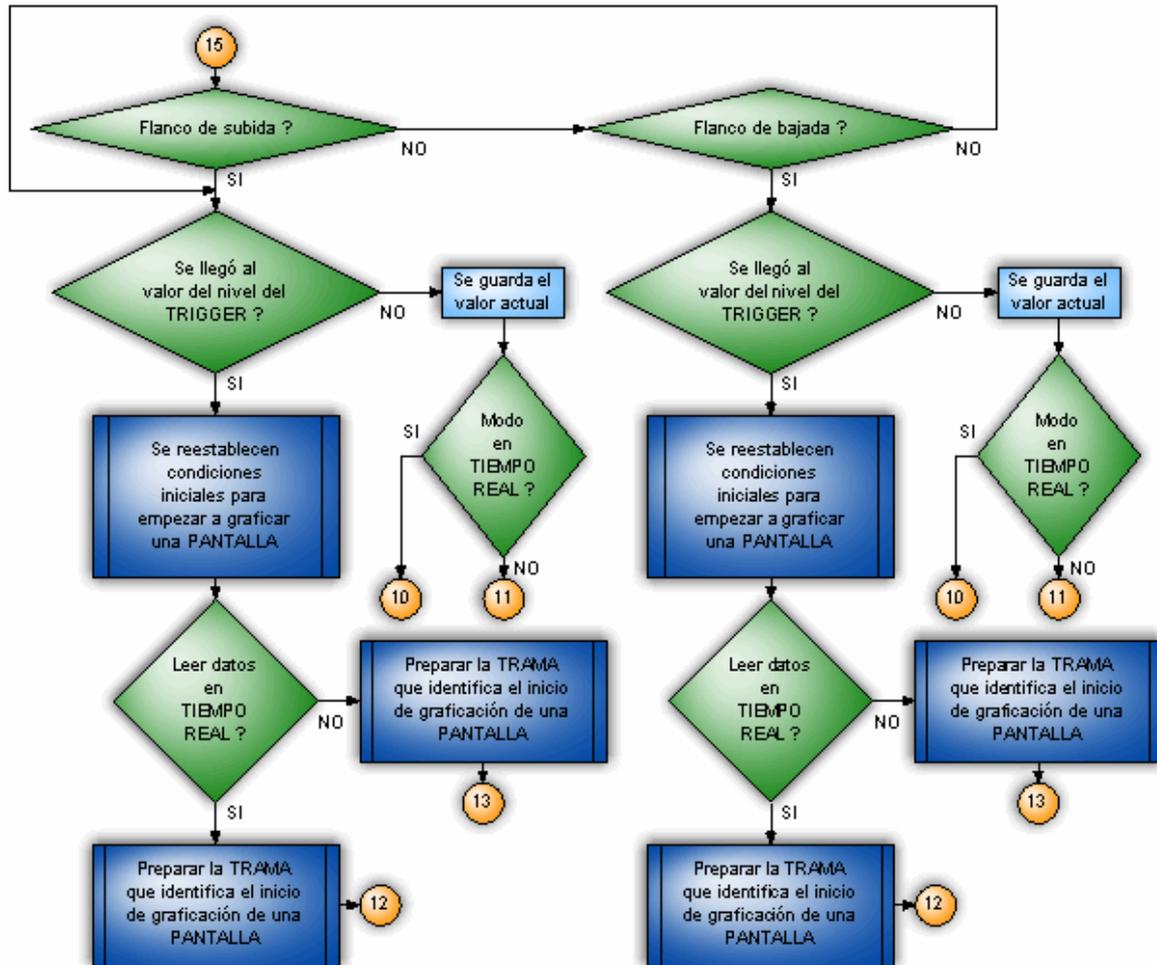


Figura 3.13 Aplicación del trigger.

Cabe destacar que mientras se está aplicando el algoritmo del trigger, no se envía ninguna trama hacia la PC, hasta que el convertor AD obtenga el dato que cumpla los parámetros de canal, nivel y flanco del trigger.

La Figura 3.13 muestra la revisión y aplicación de los parámetros del trigger elegidos por el usuario. El primer parámetro que ya fue revisado sería el canal con cuyos valores se aplicará el trigger. Para esta siguiente parte empieza revisando que flanco fue escogido, revisando el valor de la variable “TRIGGER_FLANCO”. Si es 1, aplicará el algoritmo de flanco de subida (positivo), si es 2, aplicará el algoritmo de flanco de bajada (negativo).

Después de haber elegido el flanco, se hace la comparación de los valores de “VALOR1” y “VALOR2” con el del nivel del trigger, “NIVEL_TRIGGER”. Para el

caso del flanco de subida, se debe cumplir que $VALOR1 < NIVEL_TRIGGER \leq VALOR2$, para el flanco de bajada debe cumplirse que $VALOR2 \leq NIVEL_TRIGGER < VALOR1$, con lo cual se abandona el algoritmo del trigger y se empiezan a enviar normalmente las tramas hacia la PC, hasta que vuelva a terminar de graficarse una pantalla; caso contrario, al no cumplirse la condición del respectivo caso de flanco, se hace $VALOR1 = VALOR2$, para la siguiente comparación del trigger, y se salta a recibir la siguiente lectura del AD del canal elegido.

Esto es todo cuanto se puede describir del proceso de comunicación y control basado en las órdenes que recibe el dsPIC30F3011 desde la PC. A continuación, en el Capítulo 4, se presenta un análisis del diseño de la interfaz gráfica y sus funciones principales en el control del osciloscopio.

CAPÍTULO 4

DESARROLLO DE LA INTERFAZ DEL OSCILOSCOPIO EN LA PC

DESARROLLO DE LA INTERFAZ DEL OSCILOSCOPIO EN LA PC

4.1 REQUERIMIENTOS BÁSICOS PARA EL FUNCIONAMIENTO DEL SOFTWARE

Para poder tener una interfaz gráfica amigable, que simule o se asemeje bastante al HMI de un osciloscopio comercial y sobre todo que sea intuitiva para el usuario, la primera idea fue diseñarla con algún lenguaje orientado a objetos, y que se maneje en base a eventos, debido a que un osciloscopio, al ser un instrumento de medida, requiere de que el usuario programe o realice alguna acción discreta (botones, contactos o pulsantes) o continua (perillas, tornillo, scroll, etc) para provocar una variación en los parámetros de medida del mismo. Por todo esto para el desarrollo de la interfaz, se eligió el lenguaje de programación VISUAL BASIC 6.0, que además requiere de comandos relativamente simples para diseñar una interfaz gráfica.

La comunicación de la interfaz diseñada, con el módulo de adquisición de señales se la realiza vía USB con la ayuda del módulo USBMOD4, el cual es un módulo integrado para transferencia de datos sobre la interfaz USB a velocidad máxima de 1 Megabyte por segundo, lo cual se había descrito anteriormente en el Literal 2.3.2.1 del Capítulo 2.

Cuando se usa este módulo, se debe instalar unos controladores VCP (ver Capítulo 2, Literal 2.3.2.3), se llaman así por sus siglas en inglés, *Virtual Com Port*. Estos controladores simplifican la programación en Visual Basic, debido a que hacen uso del control MSComm de Visual Basic, y manejan toda la comunicación vía USB, pero con los mismos parámetros y facilidades que si se estuviera usando un puerto COM normal. Ya habiendo instalado los controladores VCP, cada vez que se conecta el módulo, Windows agrega a su lista de periféricos y dispositivos conectados el *Virtual Com Port* (ver Figura 4.1 (a)), esto se puede visualizar en Windows XP, cuando se conecte el módulo y se seleccione en el menú Inicio > Panel de Control > Rendimiento y Mantenimiento >

Sistema > Hardware > Administrador de dispositivos, en la parte que dice *Puertos (COM & LPT)* como muestra la Figura 4.1 (b). Cuando se desconecta el módulo, Windows automáticamente quita el VCP de la lista del Administrador de dispositivos.

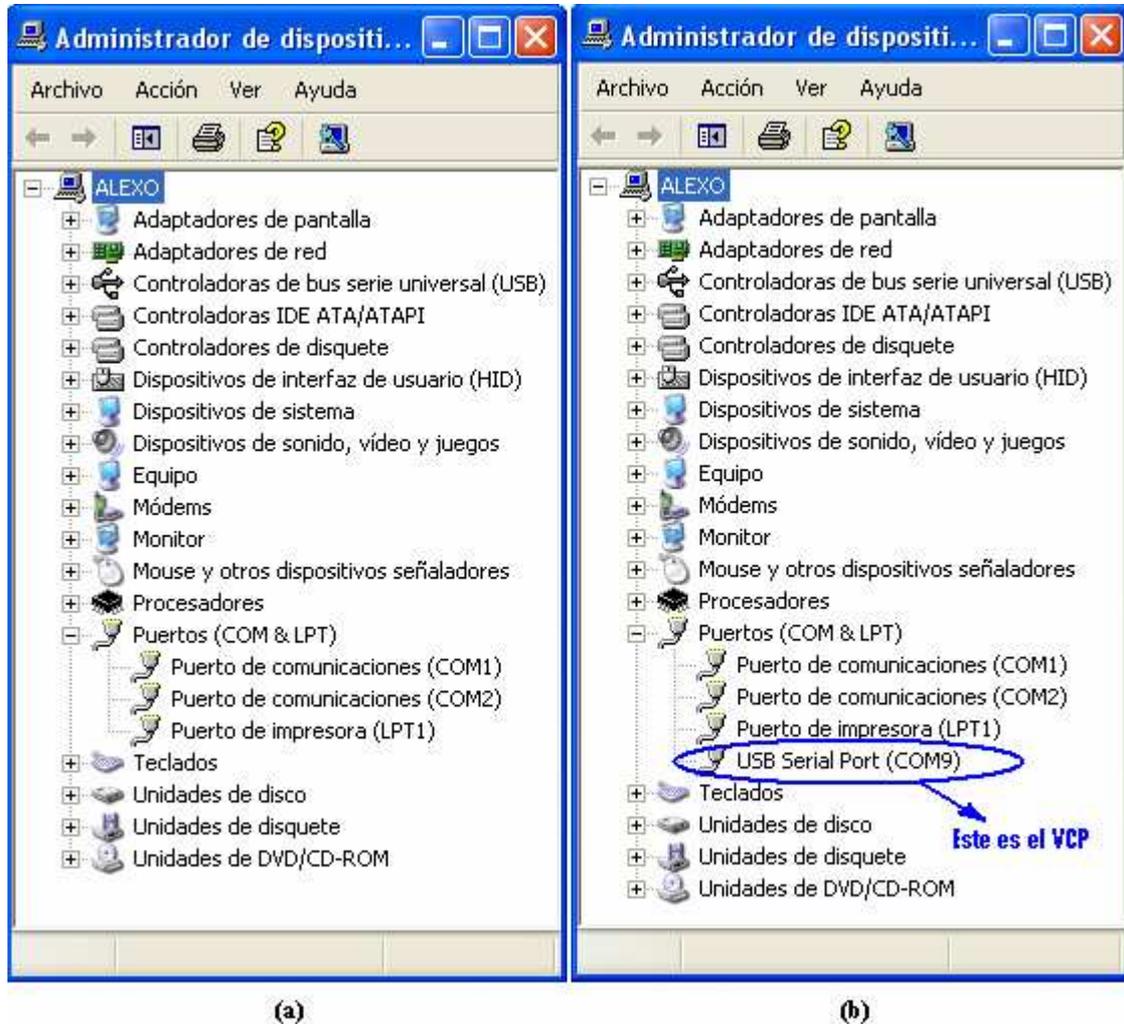


Figura 4.1 Ventana de Administrador de dispositivos en Windows XP.

El número del puerto COM varía dependiendo del número de puertos COM existentes en la PC y del número de módulos USBMOD4 conectados. Dicho número es el que se debe definir en el control MSComm del Visual Basic, para poder tener comunicación.

Otra parte importante para el mejor funcionamiento en velocidad, es definir las características del VCP, esto se hace, haciendo clic derecho sobre el "USB Serial Port" en la lista, se elige "Propiedades". Se abre entonces una ventana como la que muestra la Figura 4.2, y en la pestaña "Port Settings", se debe poner las

propiedades tal como muestra la figura, pero muy especialmente en “*Bits per second*”, debe estar a la máxima velocidad que es de 921600 bits/seg.



Figura 4.2 Ventana de Propiedades del *Virtual Com Port*.

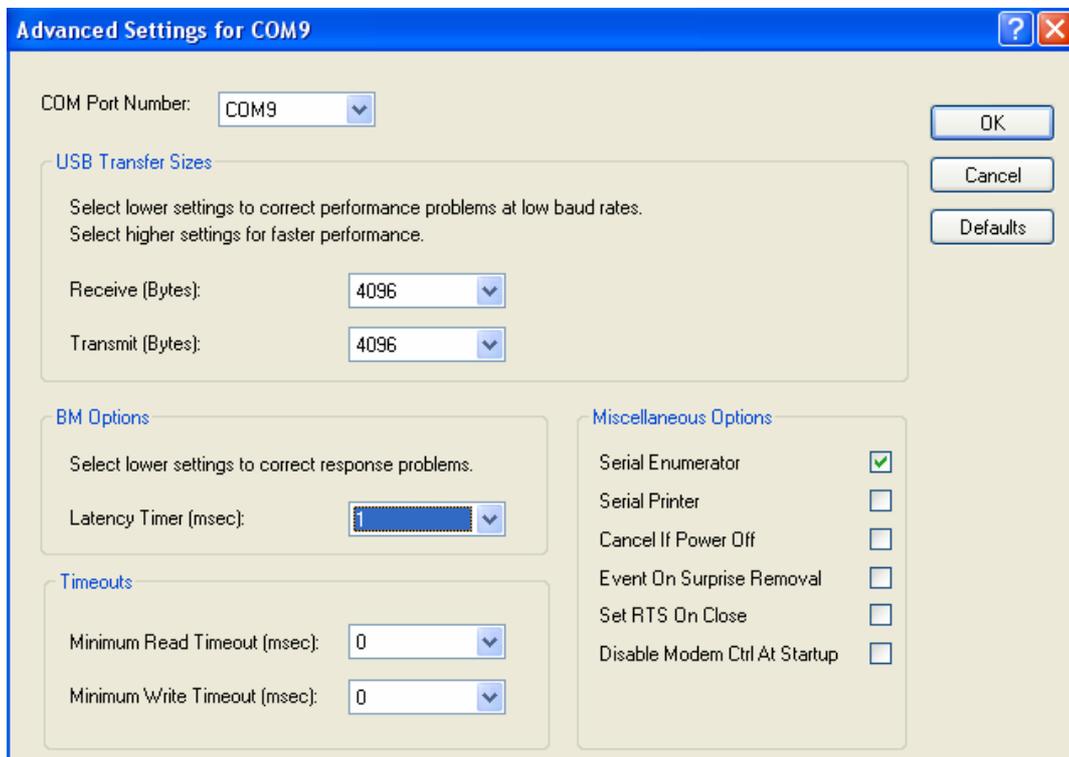


Figura 4.3 Ventana *Advanced Settings*.

Después presionando el botón *Advanced*, se abre la ventana que se presenta en la Figura 4.3. En esta se ponen los parámetros tal como se muestra en dicha figura, pero es esencial poner el *Latency Timer (msec)* en el mínimo valor de 1, después se presiona el botón *OK*, y en la ventana de Propiedades, se presiona el botón *Aceptar* y con eso el VCP queda configurado para trabajar de manera óptima.

4.2 PRESENTACIÓN DE LA INTERFAZ DE USUARIO

La interfaz en la pantalla de a PC debe simular el HMI de un osciloscopio convencional, por eso se tomó varios ejemplos de osciloscopios convencionales y se diseñó la interfaz que a continuación se presenta en la Figura 4.4.

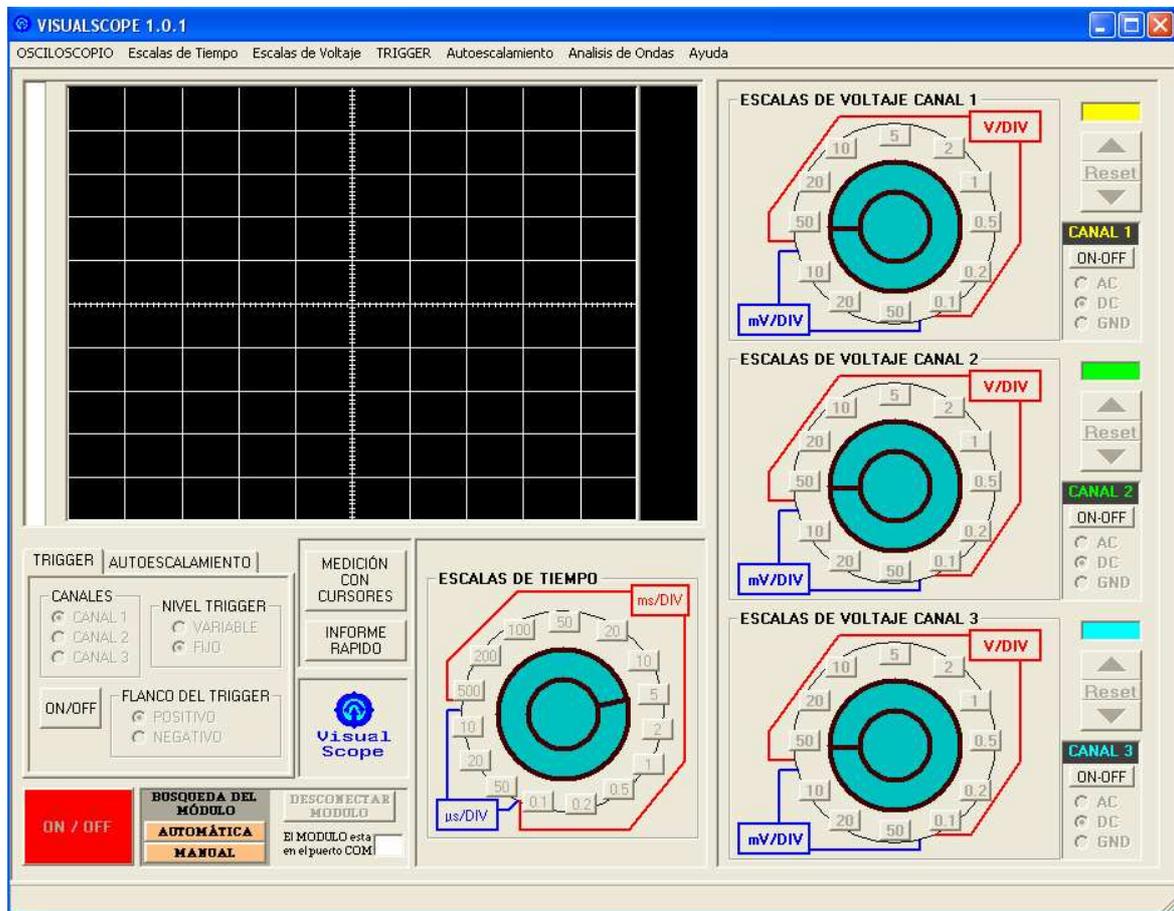


Figura 4.4 Interfaz gráfica del osciloscopio.

Como se puede ver, la interfaz consta de todos los controles de los parámetros básicos de un osciloscopio, todos incluidos también en el menú.

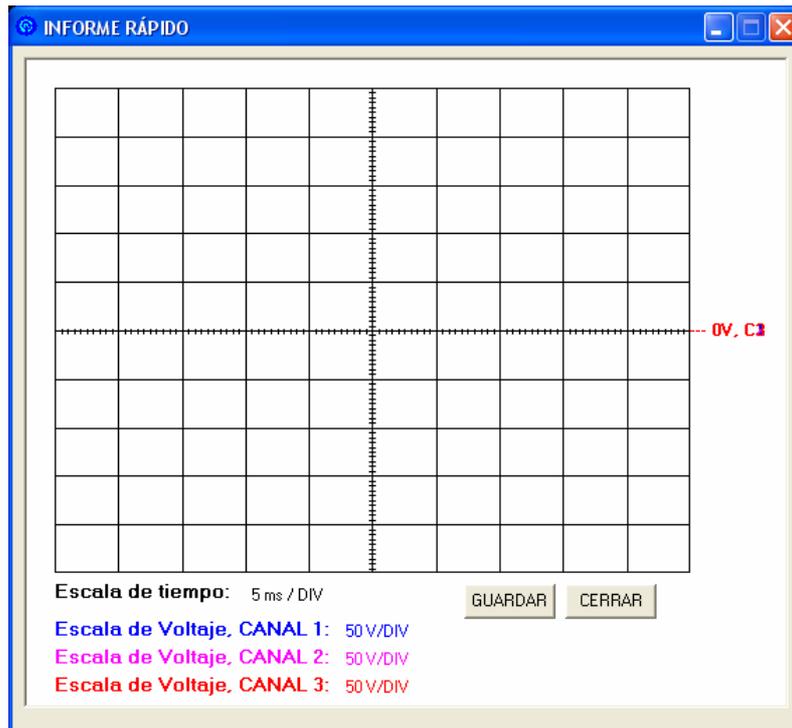


Figura 4.5 Ventana de “INFORME RÁPIDO”.

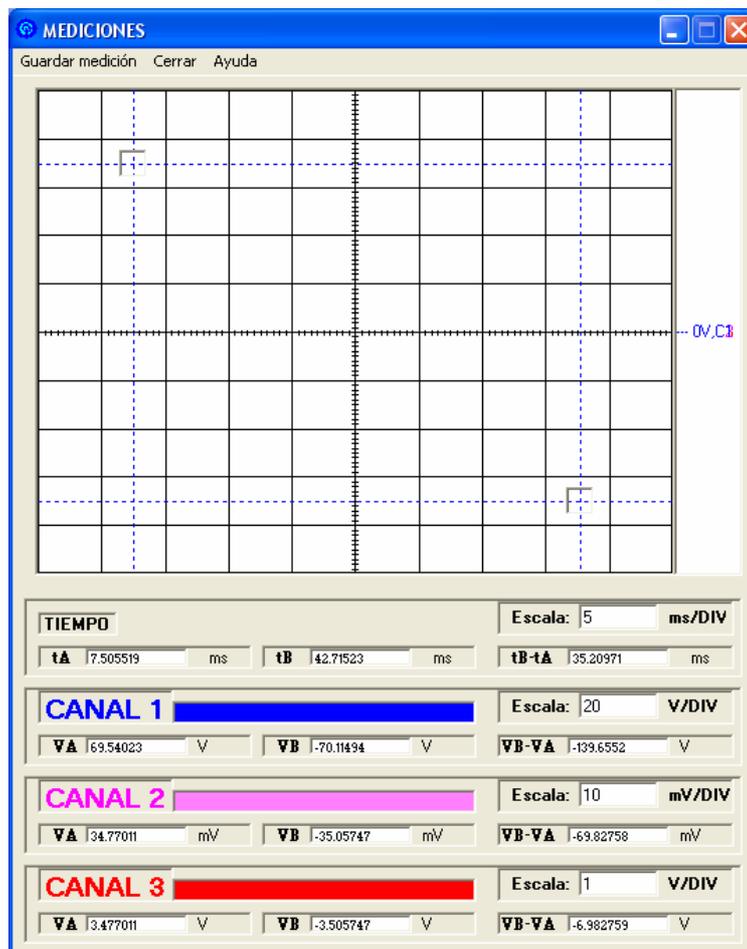


Figura 4.6 Ventana de “MEDICIÓN CON CURSORES”.

Las otras funciones de interfaz de usuario que caben destacar son las de “**MEDICIÓN CON CURSORES**” e “**INFORME RÁPIDO.**” La ventana de “**INFORME RÁPIDO**” (Figura 4.5), permite obtener una “fotografía” de las señales en pantalla, presentando para cada canal su respectiva escala. La ventana de “**MEDICIÓN CON CURSORES**” (Figura 4.6), nos permite obtener medidas diferenciales de voltaje y tiempo, en base a cursores en la pantalla. En ambos casos, tanto en el **INFORME RÁPIDO** como en la **MEDICIÓN CON CURSORES**, se puede guardar el resultado en un archivo gráfico de mapa de bits (BMP).

4.3 EVENTOS Y SUBROUTINAS MÁS IMPORTANTES PARA EL CONTROL DEL OSCILOSCOPIO

Como se mencionó en el Literal 4.1, Visual Basic es un lenguaje de programación orientado a objetos, y el funcionamiento de los programas creados en Visual Basic, se basa en eventos de dichos objetos. A continuación se exponen todos aquellos eventos y subrutinas que controlan el funcionamiento del osciloscopio.

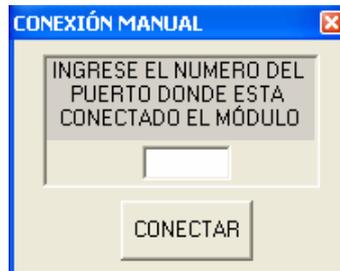


Figura 4.7 Ventana de conexión manual.

4.3.1 CONEXIÓN MANUAL DEL MÓDULO

Al presionar el botón “**MANUAL**” ubicado en la parte inferior izquierda de la interfaz gráfica principal, se abre una ventana mostrada en la Figura 4.7. Esta ventana permite conectar el módulo, si se conoce el número del puerto COM virtual que está usando el módulo (En el Literal 4.1 se puede ver como averiguar dicho número en Windows XP).

Al presionar el botón “**CONECTAR**” se genera una búsqueda del módulo un número determinado de veces en el puerto COM definido previamente. Básicamente la PC envía un número fijo al **VCP** (*Virtual Com Port*) y espera como

respuesta otro número con lo cual se confirma la conexión del módulo y aparecerá una ventana de información que indica que el módulo fue encontrado. Caso contrario, si después de hacer todos los envíos para establecer conexión, no se obtiene respuesta, aparecerá en pantalla una ventana de información que indica que el módulo no fue encontrado y los posibles procedimientos a seguir para intentar nuevamente la conexión.

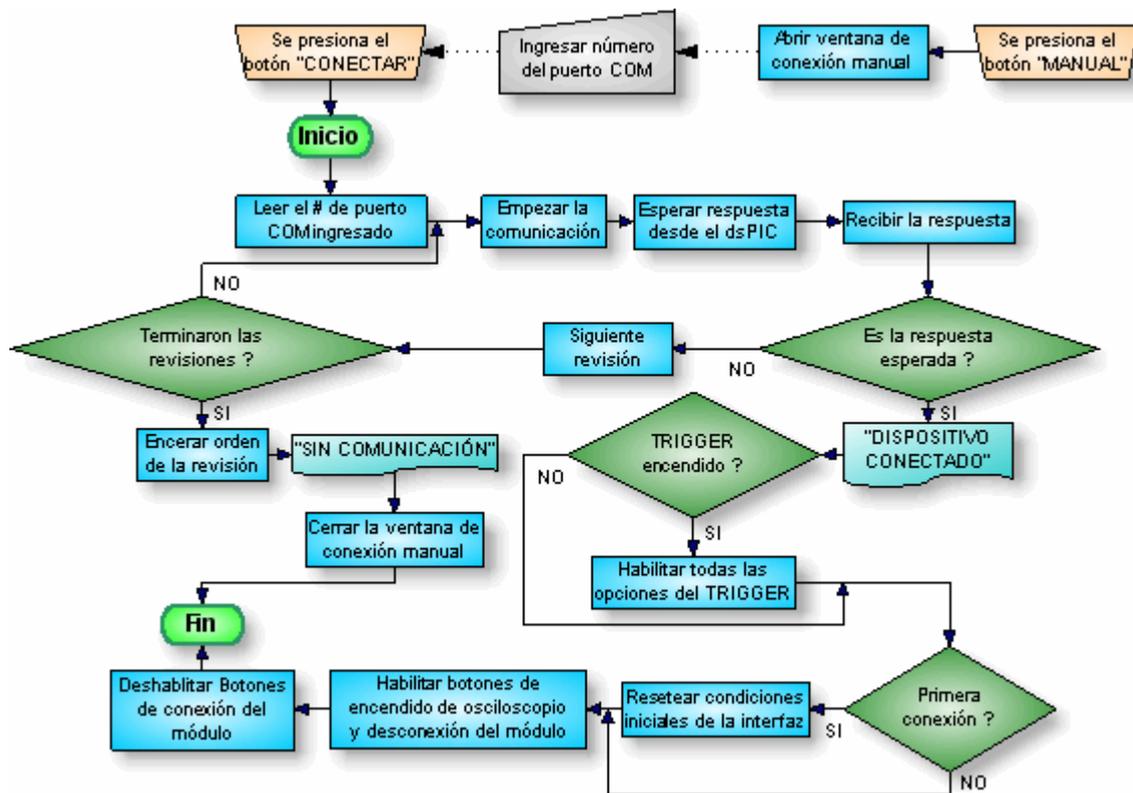


Figura 4.8 Algoritmos de conexión manual del módulo.

Para cuando ya se establece la conexión se habilitan los botones de encendido del osciloscopio y desconexión del módulo, y, se deshabilitan los botones de conexión del módulo. Solo en el caso de la primera vez que se establece la conexión, se aplican las condiciones iniciales o condiciones por defecto de los parámetros del osciloscopio. Todo este proceso se puede apreciar en la Figura 4.8.

4.3.2 CONEXIÓN AUTOMÁTICA DEL MÓDULO

La Figura 4.9 muestra este proceso. Como se puede apreciar es similar a la búsqueda manual descrita en el literal anterior, con la diferencia que el usuario no

necesita ingresar el número del puerto COM, sino que automáticamente al presionar el botón **"AUTOMÁTICA"**, la PC busca el módulo en todos los posibles puertos COM (del 1 al 255) bajo el mismo principio anterior.

Al inicio, el número del VCP (Puerto COM Virtual) es 0, por esto, se inicia con el bloque **Incrementar número del VCP**, el cual incrementa en 1 el número correspondiente al puerto COM (del 1 al 255 en una misma revisión).

Al encontrar el módulo, presenta el número del puerto en la interfaz, debajo del botón **"DESCONECTAR MÓDULO"**. Si no lo encuentra, aparece una ventana de información que indica que el módulo no fue encontrado.

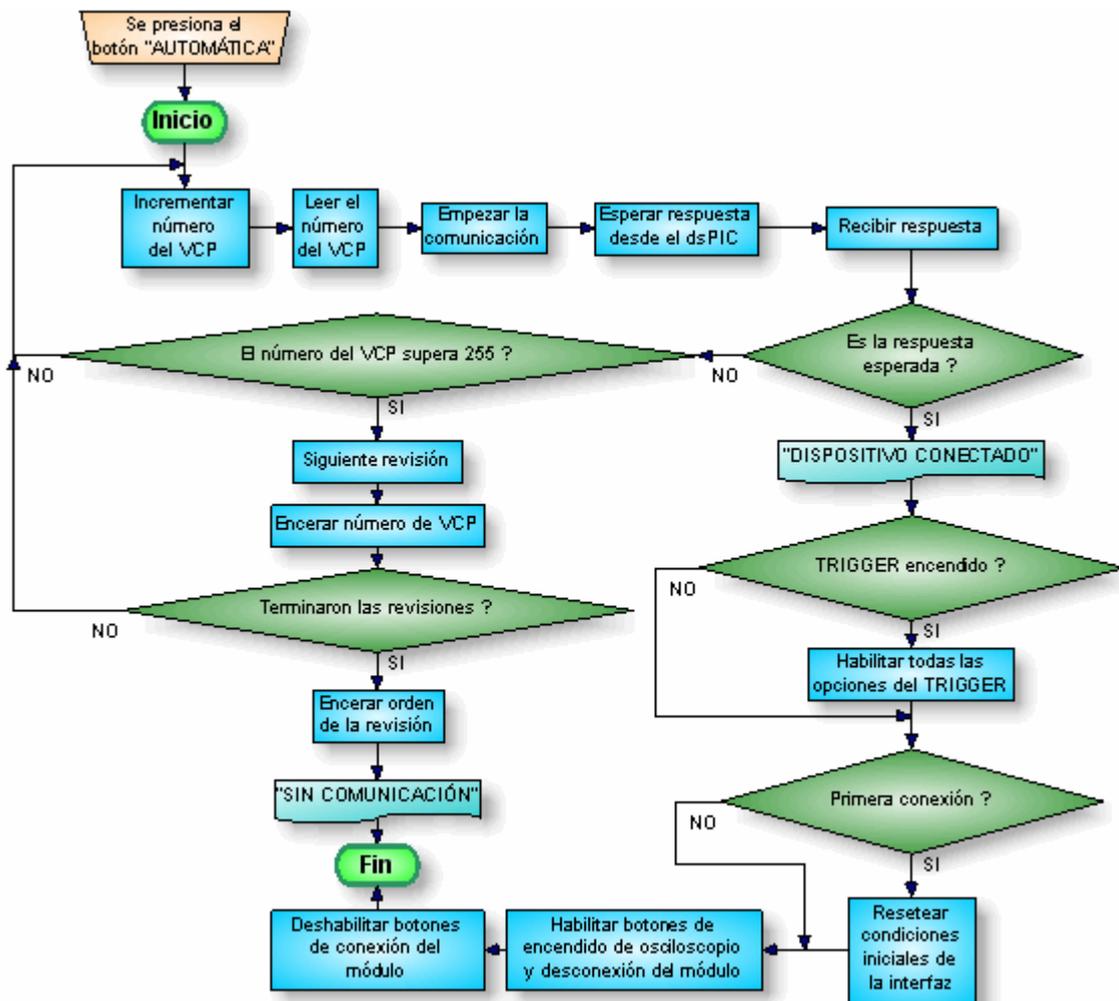


Figura 4.9 Conexión automática del módulo.

4.3.3 DESCONEXIÓN DEL MÓDULO

Al presionar el botón “**DESCONECTAR MÓDULO**” (Figura 4.10) lo primero es aplicar el evento de apagado del osciloscopio, se deshabilitan los controles de parámetros del trigger y se envía el número 100 al dsPIC, con lo cual el dsPIC se queda en un bucle infinito fuera del programa principal y se resetea el valor del puerto a su valor por defecto en Visual Basic (en código es así: `MSComm1.Commport = 1`).



Figura 4.10 Desconexión del módulo.

4.3.4 ENCENDIDO Y APAGADO DEL OSCILOSCOPIO

El botón “**ON/OFF**” en el extremo inferior izquierdo de la interfaz es el que controla el apagado y encendido del osciloscopio (Figura 4.11). Básicamente el encendido del osciloscopio consiste en enviar el número 24 hacia el dsPIC, el mismo que se sale de un bucle infinito y entonces empieza a enviar tramas. En el apagado en cambio, envía el número 25, el cual ordena al programa del dsPIC que entre en ese mismo bucle infinito con lo cual deja de enviar tramas. Para el apagado no se pone al dsPIC en modo de bajo consumo (SLEEP) debido a que no tiene mucho sentido el ahorro de energía en este caso, ya que el módulo es alimentado por una fuente que utiliza la red de 120 Vac y no en base a baterías en cuyo caso, si tendría sentido ahorrar energía.

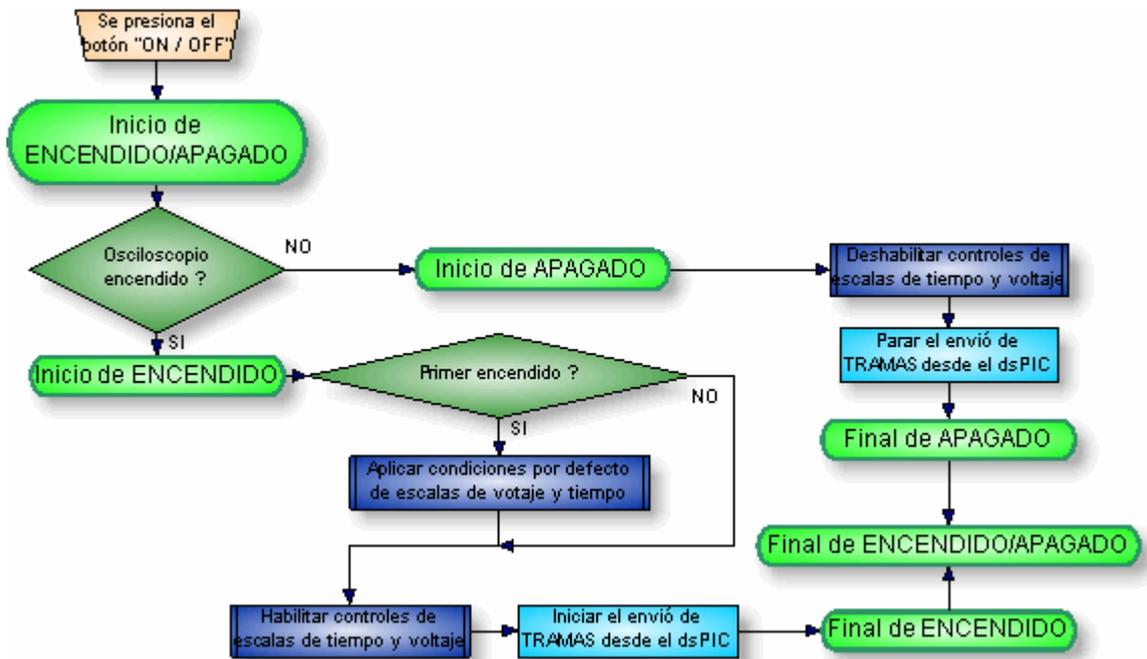


Figura 4.11 Algoritmo de encendido y apagado del osciloscopio.

A continuación se presenta en lenguaje estructurado el algoritmo de encendido y apagado del osciloscopio:

Deshabilitar controles de escalas de tiempo y voltaje:

- Deshabilitar controles de encendido/apagado de canales
- Deshabilitar controles de acoplamientos (AC, DC y GND)
- Deshabilitar todos los botones de escalas de tiempo y voltajes
- Deshabilitar los 3 cursores para la referencia 0 voltios de cada canal.

Fin de tarea.

Parar el envío de TRAMAS desde el dsPIC:

- Enviar hacia el dsPIC la orden de entrar en un lazo infinito, y con esto dejar de enviar TRAMAS hacia la PC.

Fin de tarea

Aplicar condiciones por defecto de escalas de voltaje y tiempo:

- Seleccionar la escala de tiempo de 5 ms/DIV
- Seleccionar la escala de voltaje de 50 V/DIV.

Fin de tarea.

Habilitar controles de escalas de tiempo y voltaje:

- Habilitar controles de encendido/apagado de canales
- Habilitar controles de acoplamientos (AC, DC y GND)

Habilitar todos los botones de escalas de tiempo y voltajes

Habilitar los 3 cursores para la referencia 0 voltios de cada canal.

Fin de tarea.

Iniciar el envío de TRAMAS desde el dsPIC:

Enviar hacia el dsPIC la orden para salirse del lazo infinito y continuar enviando TRAMAS hacia la PC.

Fin de tarea

Fin del algoritmo.

4.3.5 ENCENDIDO Y APAGADO DE LOS CANALES

El algoritmo presentado en la Figura 4.12 es el que describe el funcionamiento del encendido y apagado específicamente del canal 1, el procedimiento es análogo para los canales 2 y 3.

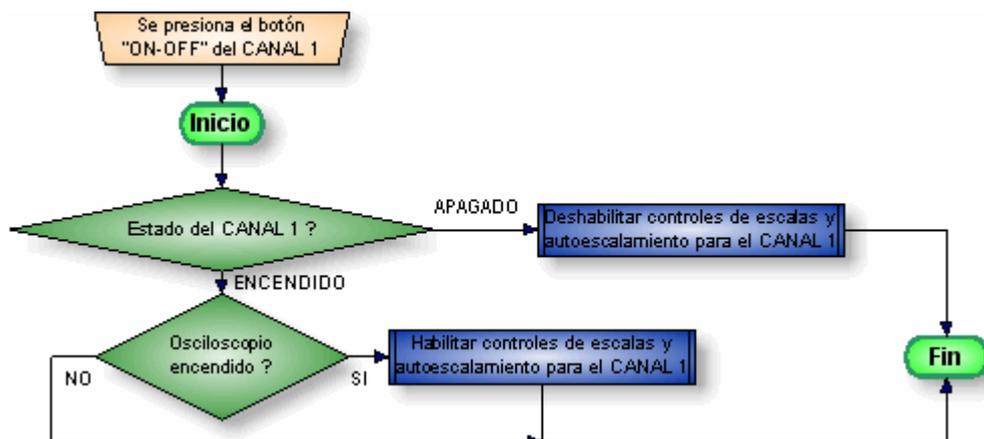


Figura 4.12 Algoritmo de encendido y apagado del canal 1.

A continuación se presenta en lenguaje estructurado el algoritmo de encendido y apagado del CANAL 1:

Si CANAL 1 APAGADO?

Deshabilitar controles y autoescalamiento para el CANAL 1:

Deshabilitar controles de acoplamientos

Deshabilitar cursores de posición de referencia 0 V

Deshabilitar botones de escalas de voltaje

Deshabilitar botones de autoescalamiento para el CANAL 1

Final de tarea

Si CANAL 1 ENCENDIDO?

Osciloscopio encendido?

Si esta encendido:

Habilitar controles y autoescalamiento para el CANAL 1:

Habilitar controles de acoplamientos

Habilitar cursores de posición de referencia 0 V

Habilitar botones de escalas de voltaje

Habilitar botones de autoescalamiento para el CANAL 1

Final de tarea

No está encendido:

Final de ENCENDIDO/APAGADO para el CANAL 1

Fin del algoritmo.

4.3.6 ESCALAS DE TIEMPO

En general todas las escalas de tiempo se aplican con el mismo algoritmo, variando entre ellas solo el número de muestras por pantalla, y el número de la orden hacia el dsPIC. Por esto, la Figura 4.13 presenta a manera de ejemplo el algoritmo del evento de selección de la escala de tiempo de 5 ms/DIV.



Figura 4.13 Evento de escalamiento en tiempo para la escala de 5 ms/DIV.

A continuación se presenta en lenguaje estructurado el evento de escala de 5 ms/DIV:

Determinar el número de muestras en PANTALLA:

Definir cuantas muestras se deben mostrar en pantalla, en este caso por ejemplo son 2300 muestras

Fin de tarea

Ordenar aplicación de características para esta escala:

Enviar la orden al dsPIC, para que el mismo determine:

Toma de datos en modo de TIEMPO REAL (como en este caso) o de ALMACENAMIENTO (para las escalas pequeñas)

Definir retardo si la toma de datos requiere del retardo (para el caso de las escalas grandes).

En el caso específico de este ejemplo, para 5 ms/DIV es modo en TIEMPO REAL y sin retardo

Fin de tarea

Fin del algoritmo.

4.3.7 ESCALAS DE VOLTAJE

A continuación se describen los eventos y subrutinas más importantes en el control del escalamiento de voltaje.

En esta sección es necesario explicar que se hará referencia a los circuitos de acondicionamiento en un canal de la siguiente forma:

- **Escala máxima:** Circuito acondicionador para mínimo -250V pico negativo y máximo 250V pico positivo.
 - o **Subescalas** de la escala máxima: 10 V/DIV, 20 V/DIV y 50 V/DIV.

- **Escala media:** Circuito acondicionador para mínimo -25V pico negativo y máximo 25V pico positivo.
 - o **Subescalas** de la escala media: 1 V/DIV, 2 V/DIV y 5 V/DIV.

- **Escala mínima:** Circuito acondicionador para mínimo -2.5V pico negativo y máximo 2.5V pico positivo.
 - o **Subescalas** de la escala mínima: 10 mV/DIV, 20 mV/DIV, 50 mV/DIV, 0.1 V/DIV, 0.2 V/DIV y 0.5 /DIV.

También es necesario tener en cuenta otros términos para poder comprender esta sección:

Escalamiento normal: Se refiere a que cuando se elige una subescala de voltaje en la interfaz de un canal, dicho canal se enclavará en el circuito acondicionador que le corresponde a dicha subescala. Por ejemplo si presiono el botón de la subescala 20 mV/DIV se terminará enclavando el circuito acondicionador de escala mínima.

Escalamiento de protección o escalamiento paralelo: Se refiere a la protección por software que aplica la interfaz de usuario para proteger al osciloscopio, en los casos en que la entrada de voltaje de cualquiera de los canales llegue a sobrepasar los valores máximos para los que fue dimensionado el circuito de acondicionamiento que se encuentre enclavado en ese momento. Es básicamente como un escalamiento paralelo que a diferencia del escalamiento normal, escala matemáticamente los valores de la señal de entrada que sobrepasen los valores máximos permitidos y evitar enclavar desde un circuito acondicionador mayor a otro menor, pudiendo provocar el daño del osciloscopio.

4.3.7.1 Eventos de escalamiento de voltaje para el circuito de acondicionamiento de escala máxima



Figura 4.14 Evento de escalamiento de voltaje en 10 V/DIV para el canal 1.

En la Figura 4.14 se muestra como ejemplo de aplicación de una subescala del circuito de acondicionamiento de escala máxima, la escala de 10 V/DIV del canal 1.

A continuación se presenta en lenguaje estructurado el evento de escala 10 V/DIV en el CANAL 1:

Aplicación de condiciones para inicio de escalamiento de protección:

Aplicar condiciones de interfaz, tanto en la interfaz gráfica principal como en el menú
Definir valores para saber en que escala estuvo y en que escala se encuentra

Fin de tarea

Enclavar circuito de escala máxima de voltaje:

Enviar la orden para que el dsPIC accione el relé del circuito de máxima escala en este caso

Fin de tarea

Determinar constante de resolución vertical:

Calcular la constante de resolución vertical que dimensionará las señales en el eje vertical según la escala

Fin de tarea

Indicar en que circuito se enclavo:

Definir valor de la variable que indica el circuito que está enclavado

Fin de tarea

Indicar que NO se aplicó el escalamiento de protección:

Defino el valor de la variable que indica que no se aplicó el escalamiento de protección

Fin de tarea

Fin del algoritmo.

Para el caso del presente proyecto el término enclavar significa que un relé cierra un circuito.

4.3.7.2 Eventos de escalamiento de voltaje para el circuito de acondicionamiento de escala media

Como ejemplo de este caso se tomó la aplicación de la escala de 1 V/DIV en el canal 1 (Figura 4.15).

El evento se inicia presionando el botón de 1 V/DIV de las escalas de voltaje del canal 1. Y la descripción de las distintas tareas de este se explican a continuación en lenguaje estructurado.

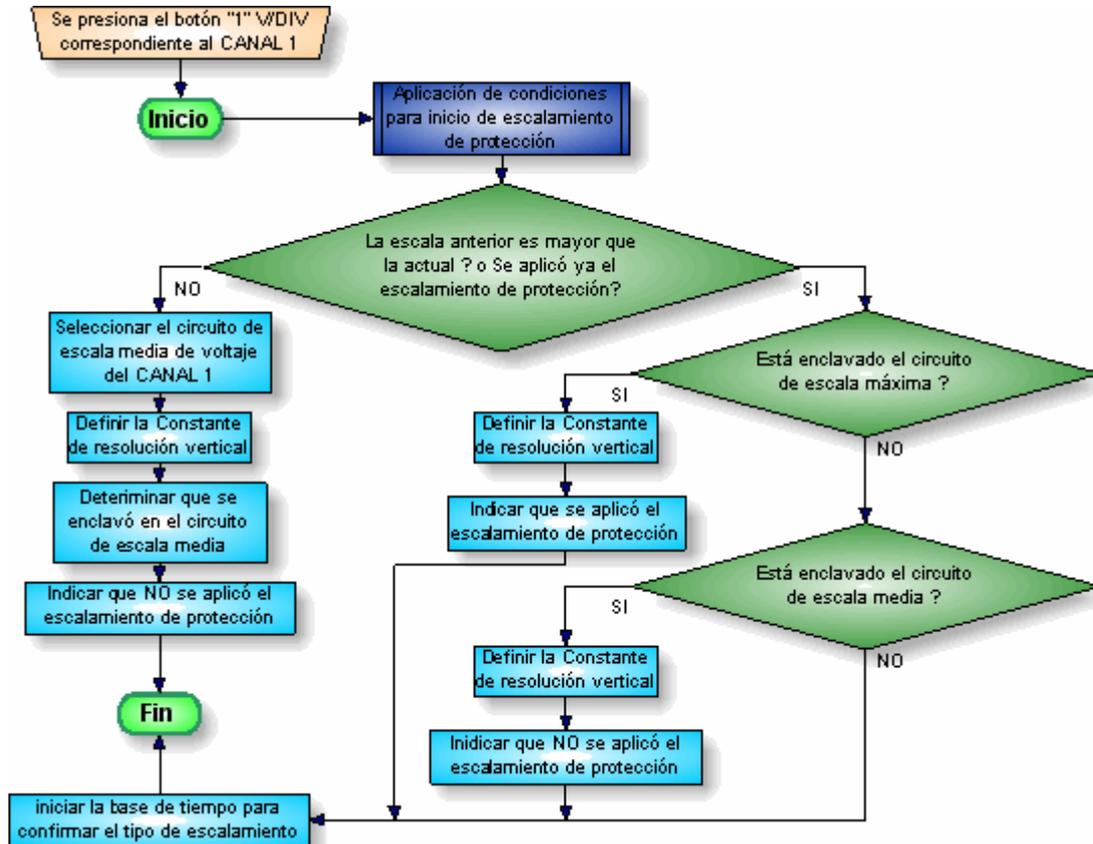


Figura 4.15 Evento de escalamiento de voltaje en 1 V/DIV para el canal 1.

A continuación se presenta en lenguaje estructurado el evento de escala 1 V/DIV para el CANAL 1:

Aplicación de condiciones para inicio de escalamiento de protección:

Aplicar propiedades de interfaz gráfica y del menú para ésta escala.

Definir mediante variables en que escala estuvo y la escala actual (necesario para aplicar el escalamiento de protección)

Fin de tarea

Seleccionar el circuito de escala media de voltaje para el canal 1:

Ordenar al dsPIC, que accione el relé que selecciona el circuito de escala media del canal

1

Fin de tarea

Definir la constante de resolución vertical:

Determinar la constante de resolución vertical que dimensionará las señales en el eje vertical según la escala

Fin de tarea**Determinar que se enclavó en el circuito de escala media:**

Hacer saber al programa que se seleccionó el circuito de escala media

Fin de tarea

Fin del algoritmo.

Para lo que se denomina el escalamiento de protección, es siempre importante saber cuando se pasa de una escala mayor (refiriéndose como escala a los circuitos de acondicionamiento, no a las subescalas o escalas de la interfaz) a una menor o si está aplicado el escalamiento de protección, en cualquiera de los dos casos.

4.3.7.3 Eventos de escalamiento de voltaje para el circuito de acondicionamiento de escala mínima

En la Figura 4.16 se presenta el algoritmo para este caso. A manera de ejemplo se escogió en este caso el evento de selección de la escala de 10 mV/DIV en el canal 1.

El evento inicia presionando el botón correspondiente a la escala de 10 mV/DIV en el canal 1. Primero define las variables de identificación de circuito de acondicionamiento correspondiente, se ve si la escala anterior es mayor a la actual. Si no se cumple, se enclava el circuito de escala mínima de voltaje, se define el valor de la constante de dimensionamiento vertical, y se determina que no se aplicó el escalamiento de protección. En el caso que si se cumpla, se revisa cual fue la escala anterior en base al mismo solo provisionalmente se determina la constante de dimensionamiento vertical, se aplica el escalamiento de protección y se arranca una base de tiempo (de 1 segundo) para confirmar el

Solo en el caso de que la escala anterior sea la misma que la actual, como muestra la Figura 4.17, se revisa el valor que indica en que circuito está enclavado el canal, para aplicar o no dicho escalamiento de protección.

4.3.7.4 Evento de final de base de tiempo de escalamiento de protección

Como se ve en las Figuras 4.18 y 4.19 para el caso del canal 1, al finalizar la base de tiempo de escalamiento de protección, se revisa una variable (se verá con más detalle en el Literal 4.3.11.3) que indica que los valores que van a graficarse en pantalla sobrepasan o no los límites de la misma, en caso de sobrepasar dichos límites se aplica escalamiento de protección, caso contrario se revisa todas las subescalas de escala media y escala mínima para determinar el valor de la constante de dimensionamiento vertical según la subescala seleccionada, y, enclavar el circuito correspondiente de escala media o escala mínima. En la Figura 4.18 están las subescalas de escala media y en la Figura 4.29 se muestran las subescalas de escala mínima.

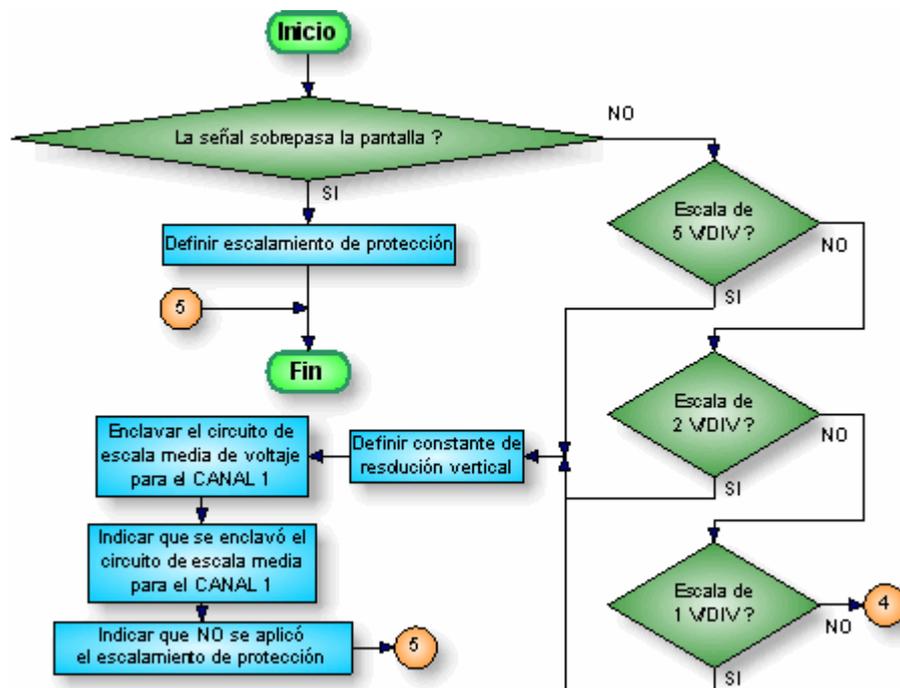


Figura 4.18 Evento de finalización de base de tiempo de escalamiento de protección.

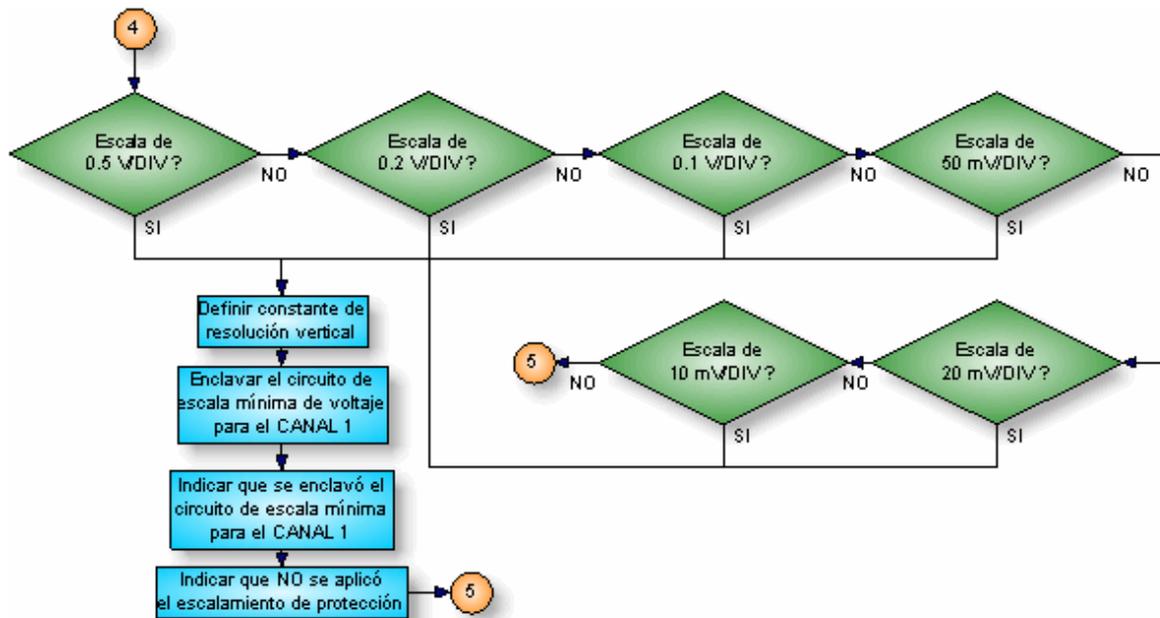


Figura 4.19 Evento de finalización de base de tiempo de escalamiento de protección (continuación).

4.3.8 EVENTOS DE ACOPLAMIENTOS (AC, DC, GND) PARA UN CANAL

La Figura 4.20 muestra los eventos de acoplamientos para el canal 1. Al presionar el botón "AC", se envía al dsPIC la orden de enclavamiento de la entrada del circuito de acondicionamiento, a través de un capacitor para eliminar cualquier componente de DC. Presionando el botón "DC", se envía al dsPIC la orden que indica enclavar el circuito de acondicionamiento a la entrada BNC del canal 1. Y, presionando el botón "GND", se envía la orden de enclavamiento de la entrada del circuito de acondicionamiento a tierra (nivel 0 voltios).

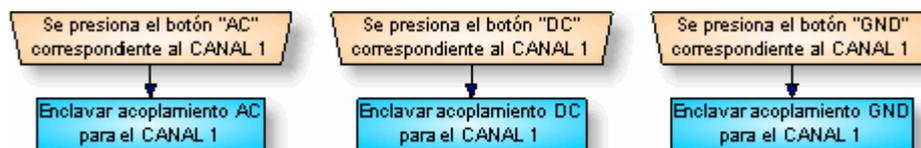


Figura 4.20 Eventos de acoplamientos para el canal 1.

4.3.9 AUTOESCALAMIENTO

El autoescalamiento en voltaje se refiere al proceso por el cual presionando un botón, el osciloscopio ajusta la representación gráfica de la señal que se está midiendo en pantalla, de tal forma que esta se vea completa desde su valor

máximo hasta su valor mínimo en la escala vertical en pantalla buscando la mejor apreciación visual para el usuario.

Para esta parte del autoescalamiento vale la pena una explicación. En los diagramas de flujo se mencionará frecuentemente las frases **rango positivo de escala** o **rango positivo de subescala**, y, **rango negativo de escala** o **rango negativo de subescala**, los cuales significan:

Rango positivo de escala o de subescala: Se refiere a todos aquellos valores de voltaje positivo que están dentro del rango de una escala o subescala, pero que no son comunes o son mayores a los valores de escalas o subescalas menores. Por ejemplo: para la escala de -250V a 250V, el rango positivo de escala son los valores mayores que 25V y menores o iguales a 250V.

Rango negativo de escala o de subescala: Se refiere a todos aquellos valores de voltaje negativo que están dentro del rango de una escala o subescala, pero que no son comunes o son menores a los valores de escalas o subescalas menores. Por ejemplo: para la escala de -250V a 250V, el rango negativo de escala son los valores menores que -25V y mayores o iguales a -250V.

A continuación se presentan todos aquellos algoritmos que controlan la aplicación del autoescalamiento solo para el canal 1, debido a que de manera análoga están diseñados para los otros 3 canales, variando entre ellos solo las respectivas órdenes de aplicación de los circuitos de acondicionamiento.

4.3.9.1 Evento de autoescalamiento para un canal

La Figura 4.21 nos muestra la generación del evento de autoescalamiento para el canal 1 al presionar el botón "**AUTOESCALAMIENTO_CANAL_1**" (botón amarillo) en la pestaña "**AUTOESCALAMIENTO**".

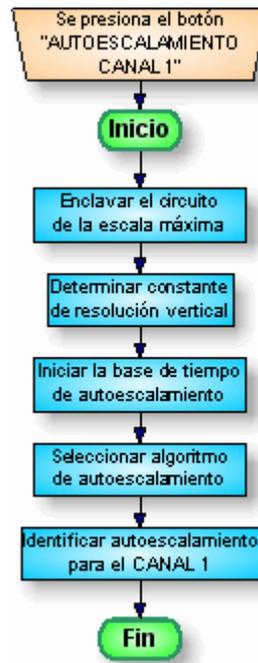


Figura 4.21 Evento de autoescalamiento para el canal 1.

A continuación se presenta en lenguaje estructurado el evento de autoescalamiento para el canal 1:

Enclavar el circuito de la escala máxima:

Ordenar al dsPIC, que cierre el relé correspondiente al circuito acondicionador de escala máxima del canal 1.

Fin de tarea

Determinar constante de resolución vertical:

Definir la constante para calcular los valores a graficarse en pantalla dependiendo de que subescala está seleccionada, y que circuito acondicionador fue seleccionado

Fin de tarea

Iniciar la base de tiempo de autoescalamiento:

Iniciar una base de tiempo de 1 segundo, donde todos los datos que van llegando a la PC son escalados según la subescala seleccionada y después son comparados entre sí, para obtener al final de la base de tiempo el valor mínimo y el máximo de los mismos en resolución de 10 bits

Fin de tarea

Seleccionar algoritmo de autoescalamiento:

Identificar con una variable, que se va a aplicar el algoritmo de autoescalamiento, ya que la base de tiempo no solo se usa para este proceso, sino también se la aprovecha para

calcular el valor del nivel del trigger automático (ver Literales 4.3.10.5 y 4.3.10.6), entonces el programa al finalizar dicha base de tiempo debe identificar si va a aplicar el algoritmo para calcular el valor del nivel del trigger fijo, o el algoritmo de autoescalamiento

Fin de tarea

Identificar autoescalamiento para el CANAL 1:

Determinar que canal va a ser autoescalado, para que el programa sepa a que datos debe aplicarles el algoritmo de comparación cada vez que la PC recibe un dato durante este proceso y también para que a final de la base de tiempo, sepa que señal será autoescalada en la pantalla, en este caso es la señal del canal 1

Fin de tarea

Fin del algoritmo.

4.3.9.2 Evento de finalización de tiempo de autoescalamiento

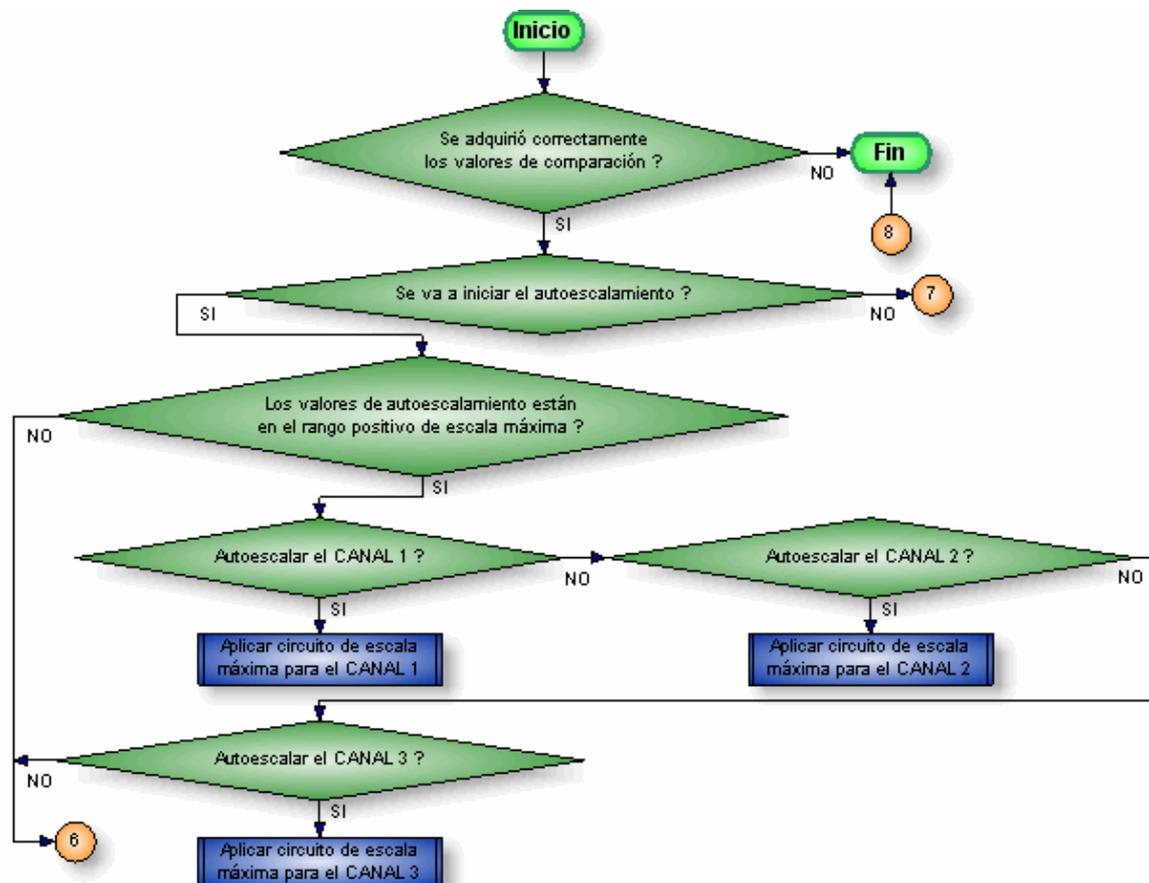


Figura 4.22 Evento de finalización de tiempo de autoescalamiento.

Al finalizar la base de tiempo que se arranca al presionar el botón de autoescalamiento, en este caso específicamente del canal 1, se comparan dos variables (llamadas “AUTOESCAL_MAXIMO” y “AUTOESCAL_MINIMO”) que

definen su valor en cada evento de recepción de una trama (ver Literal 4.3.11.4) desde el dsPIC. En dichas variables al finalizar la base de tiempo de 1 segundo, quedarán solo los valores mínimo y máximo de la señal que está sensando en este caso el canal 1.

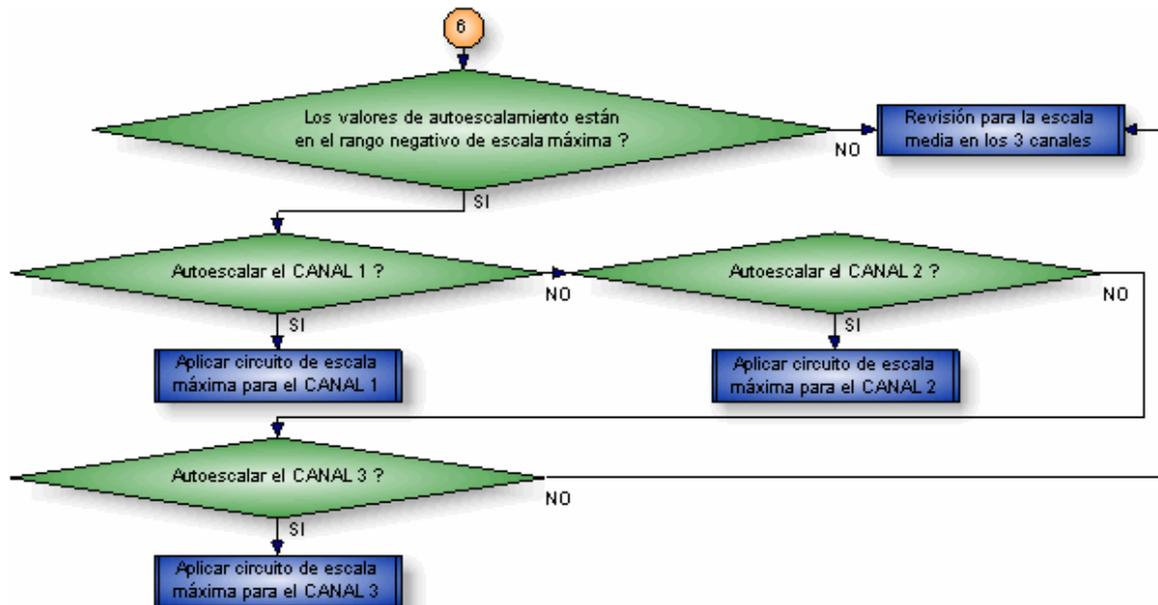


Figura 4.23 Comprobación final de evento de autoescalamiento.

Al inicio de este evento, como se ve en la Figura 4.22, se compara dichos valores máximo y mínimo entre ellos solo por seguridad (debiendo cumplirse lógicamente que el valor máximo debe ser mayor que el mínimo). Al cumplirse la condición de seguridad descrita anteriormente, se prosigue a determinar si el evento de base de tiempo fue generado para autoescalamiento o para el cálculo del nivel fijo del trigger; Al cumplirse que fue generado para autoescalamiento, se comprueba si la señal tiene valores que estén dentro del rango de máxima escala, es decir, que el valor máximo represente valores de voltaje mayores a 25V y menores o igual a 250V, que comparados con la escala máxima (porque ahí se enclava en un inicio para iniciar el autoescalamiento) representan los valores de 562 para 25V y 1023 para 250V en resolución de 10 bits, si esto se cumple, es condición suficiente para enclavar el circuito acondicionador de escala máxima para el canal 1. En caso de no cumplirse, se revisa el valor mínimo, que análogamente al caso anterior presenta valores de 0 para -250V y 461 para -25V. Si esto se cumple, es condición suficiente para enclavar el circuito acondicionador de máxima escala para el canal 1 (Figura 4.23).

4.3.9.3 Aplicación del circuito de escala máxima para un canal

Al aplicarse el circuito de escala máxima (-250V a 250V), ahora solo queda determinar la subescala correspondiente a este circuito acondicionador, determinando el valor de la constante de dimensionamiento vertical, la cual sirve para calcular el valor definitivo a graficarse en la pantalla, como muestra el algoritmo de la Figura 4.24.



Figura 4.24 Aplicación del circuito de escala máxima para el canal 1.

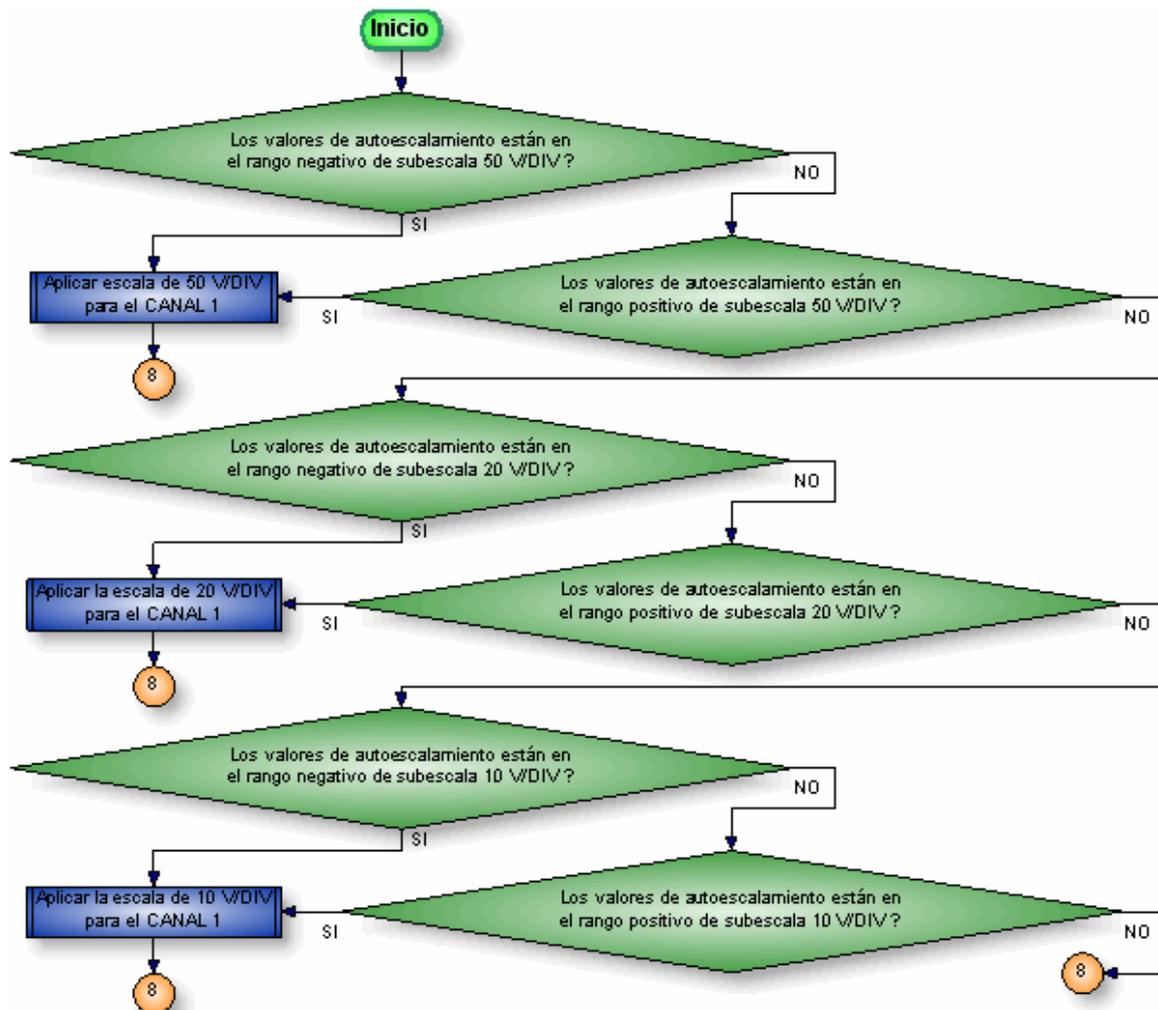


Figura 4.25 Algoritmo de revisión de subescalas de la escala máxima para el canal 1.

4.3.9.4 Revisión de sub-escalas de la escala máxima para un canal

En este caso ya se encuentra enclavado el circuito de escala máxima y en base a este se hace las comparaciones de los valores máximo y mínimo para determinar que subescala debe aplicarse como muestra la Figura 4.25. Las equivalencias en resolución de 10 bits para dichas subescalas se presentan en la Tabla 4.1.

| | Subescala (V/DIV) | Voltaje (V) | Valor límite de 10 bits |
|--|----------------------|----------------|-------------------------|
| | | | AUTOESCAL_MAXIMO |
| | 50 | 250 | 1023 |
| | 20 | 100 | 716 |
| | 10 | 50 | 613 |
| Límite superior de escala media | 5 | 25 | 562 |
| | | | AUTOESCAL_MINIMO |
| Límite inferior de escala media | 5 | -25 | 461 |
| | 10 | -50 | 410 |
| | 20 | -100 | 307 |
| | 50 | -250 | 0 |

Tabla 4.1 Equivalencias de valores límite en 10 bits de resolución vertical, para las subescalas de la escala máxima (-250V a 250V).

4.3.9.5 Revisión de sub-escalas de la escala media para los 3 canales

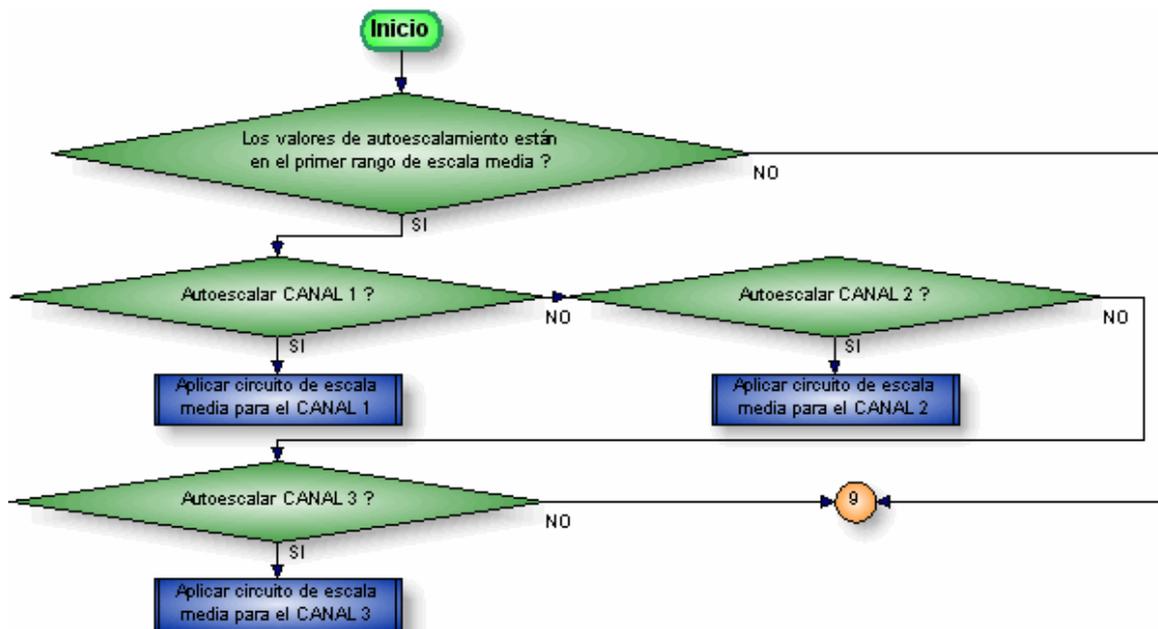


Figura 4.26 Inicio de revisión de subescalas para escala media.

En la Figura 4.26 se muestra como se da este proceso. Primero se revisa en que límite están los valores máximo y mínimo de la señal, tomados durante el tiempo

(1 segundo) de autoescalamiento, Los valores de 516 a 562 del valor máximo, representan los límites de 2.5V hasta 25V respectivamente, y el valor de 461 para el mínimo representa el límite inferior de escala media de -25V.

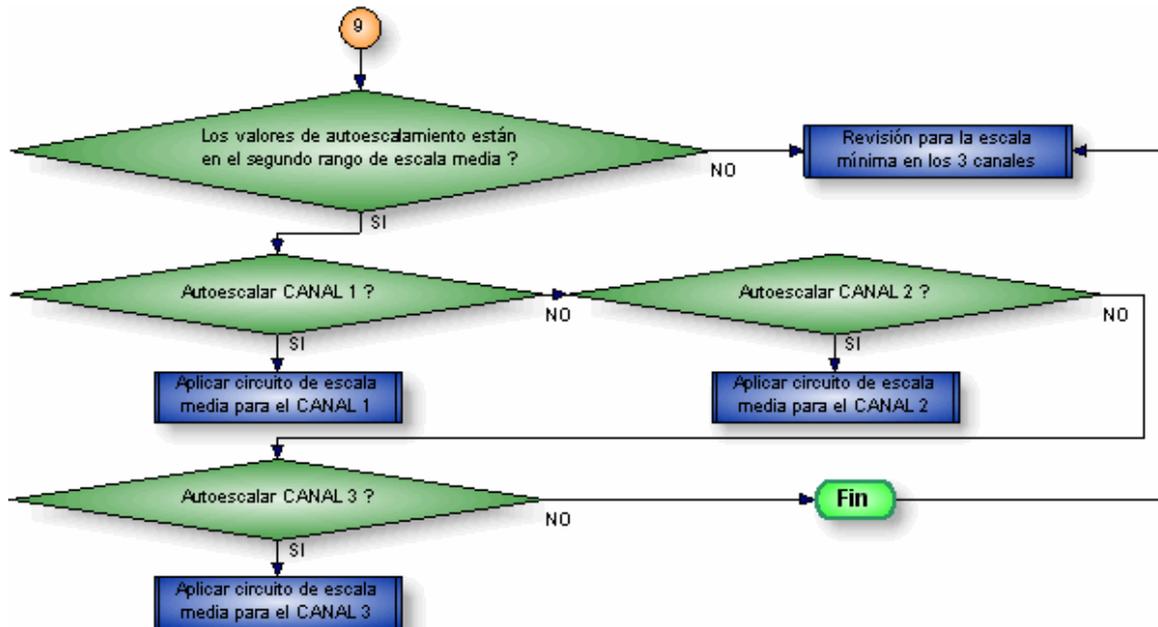


Figura 4.27 Revisión de subescalas para escala media (continuación).

Al cumplirse que dichos valores estén en esos rangos, se aplicaría el circuito acondicionador de escala media; caso contrario, se revisa el otro caso (Figura 4.27), en el que debe cumplirse que el valor mínimo debe estar entre 461 y 507 y el máximo debe ser menor o igual a 562, donde 461 y 507 representan los límites de -2.5V, y -25V respectivamente, y 562 representa el límite superior de 25V. Si este caso se cumple, entonces se aplica el circuito acondicionador de escala media, caso contrario continúa con la revisión para el caso de la escala mínima.

4.3.9.6 Aplicación del circuito de escala media para un canal

Como muestra la Figura 4.28, el circuito de media escala del canal 1 se aplica solo enviando el número 2 al dsPIC, para después de eso empezar el algoritmo de aplicación de subescalas de la escala media.



Figura 4.28 Aplicación del circuito acondicionador de escala media para el canal 1.

4.3.9.7 Revisión de sub-escalas de la escala media para un canal

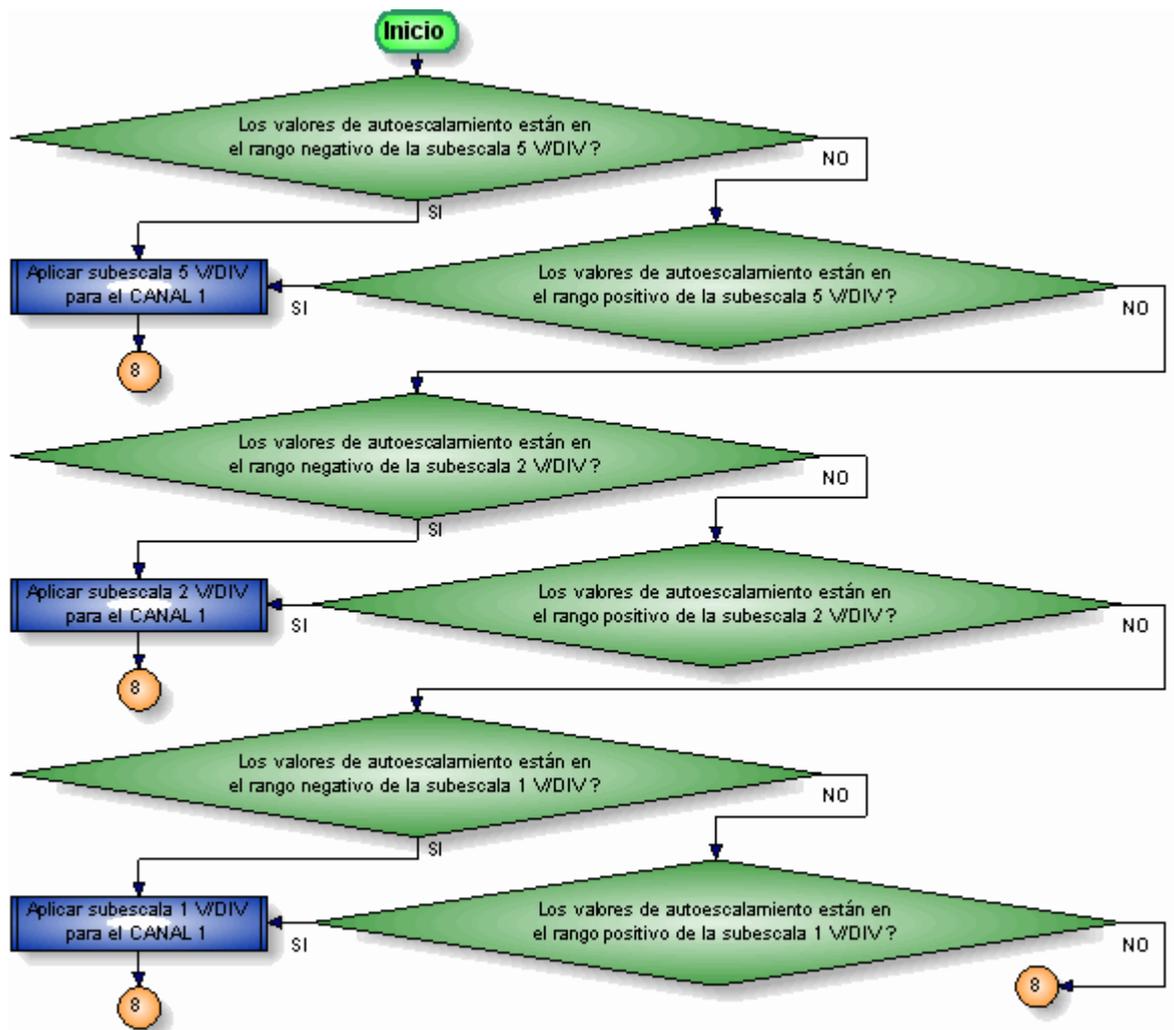


Figura 4.29 Revisión de subescalas de la escala media para el canal 1.

En este punto empieza la revisión de, que subescala del circuito acondicionador (Figura 4.29) de la escala media se debe aplicar, dependiendo de los valores máximo y mínimo de la señal, pero hay que tomar en cuenta que estos valores fueron tomados cuando todavía estaba enclavado el circuito acondicionador de la

escala máxima, así que los límites de las variables a revisar son como muestra la Tabla 4.2.

| | Subescala (V/DIV) | Voltaje (V) | Valor límite de 10 bits |
|---|----------------------|----------------|-------------------------|
| | | | AUTOESCAL_MAXIMO |
| | 5 | 25 | 562 |
| | 2 | 10 | 531 |
| | 1 | 5 | 521 |
| Límite superior de escala mínima | 0.5 | 2.5 | 516 |
| | | | AUTOESCAL_MINIMO |
| Límite inferior de escala mínima | 0.5 | -2.5 | 507 |
| | 1 | -5 | 502 |
| | 2 | -10 | 492 |
| | 5 | -25 | 461 |

Tabla 4.2 Equivalencias de valores máximos en 10 bits de resolución vertical, para las subescalas de la escala media (-25V a 25V) en base a la escala máxima.

4.3.9.8 Revisión de sub-escalas de la escala mínima para los 3 canales

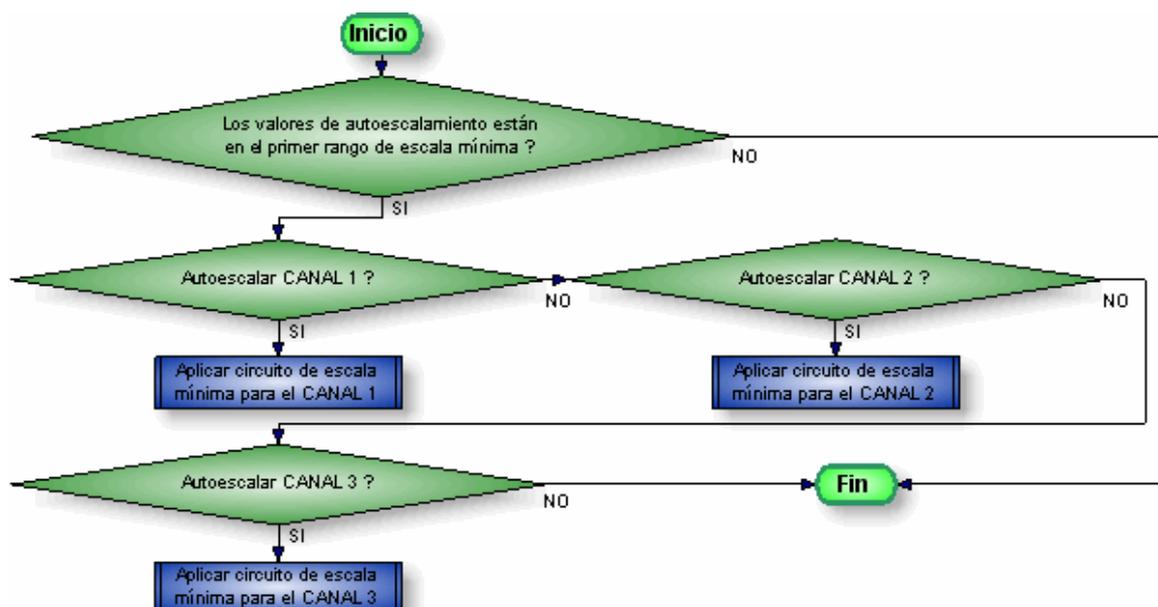


Figura 4.30 Inicio de revisión de subescalas de la escala mínima.

Como los valores máximo y mínimo de la señal fueron tomados en primera instancia en base a la escala máxima, se tiene que las equivalencias en 10 bits para los límites de -2.5V a 2.5V con respecto a la escala máxima son de 507 a 516 respectivamente. Entonces la Figura 4.30 muestra la comparación inicial que se hace en base a dichas variables, para aplicar el circuito de mínima escala en el caso que se cumpla con el rango mencionado, en el canal que se esté autoescalando. La frase “primer rango de escala mínima”, se refiere

precisamente a las equivalencias de los límites -2.5V a 2.5V en base a la escala máxima.

4.3.9.9 Aplicación del circuito de escala mínima para un canal



Figura 4.31 Aplicación del circuito acondicionador de escala mínima para el canal 1.

La Figura 4.31 describe la aplicación del circuito de escala mínima específicamente para el canal 1. La PC envía la orden al dsPIC para enclavar su circuito acondicionador de escala mínima (-2.5V a 2.5V), pero aquí se debió implementar otra base de tiempo para autoescalar con respecto a la escala mínima, debido a que si vemos los valores de las equivalencias en resolución de 10 bits de los valores máximos de la escala mínima respecto a la escala máxima, son muy pequeños, por lo tanto para el caso de sus subescalas dichas equivalencias serían demasiado pequeñas, e incluso no se podría ni hacer aproximaciones por que se salen de la resolución de 10 bits que se tiene. Entonces, después de enclavar el circuito acondicionador de escala mínima, se arranca la base de tiempo de autoescalamiento de mínima escala, se define el canal que se está autoescalando, en este caso el canal 1 y finaliza el evento de autoescalamiento.

4.3.9.10 Evento de finalización de base de tiempo para autoescalar con la mínima escala para los 3 canales

Al finalizar el evento de la base de tiempo de autoescalamiento para la escala mínima (Figura 4.32), Lo que se debe hacer es solo determinar que canal se está autoescalando para después empezar con la revisión que determina la selección final de la subescala.

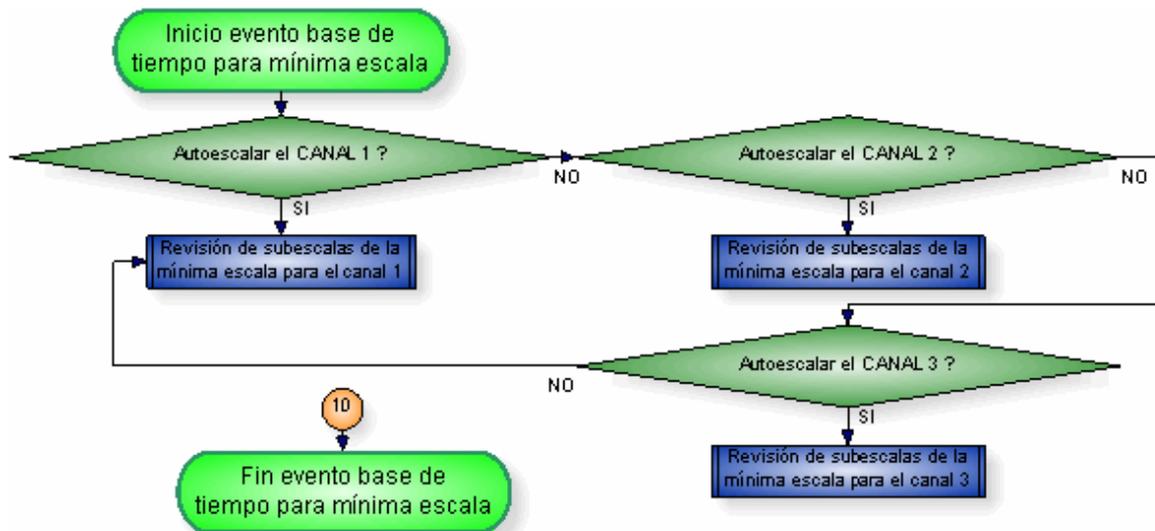


Figura 4.32 Evento de finalización de base de tiempo para autoescalar con la mínima escala.

4.3.9.11 Revisión de sub-escalas de la escala mínima para un canal

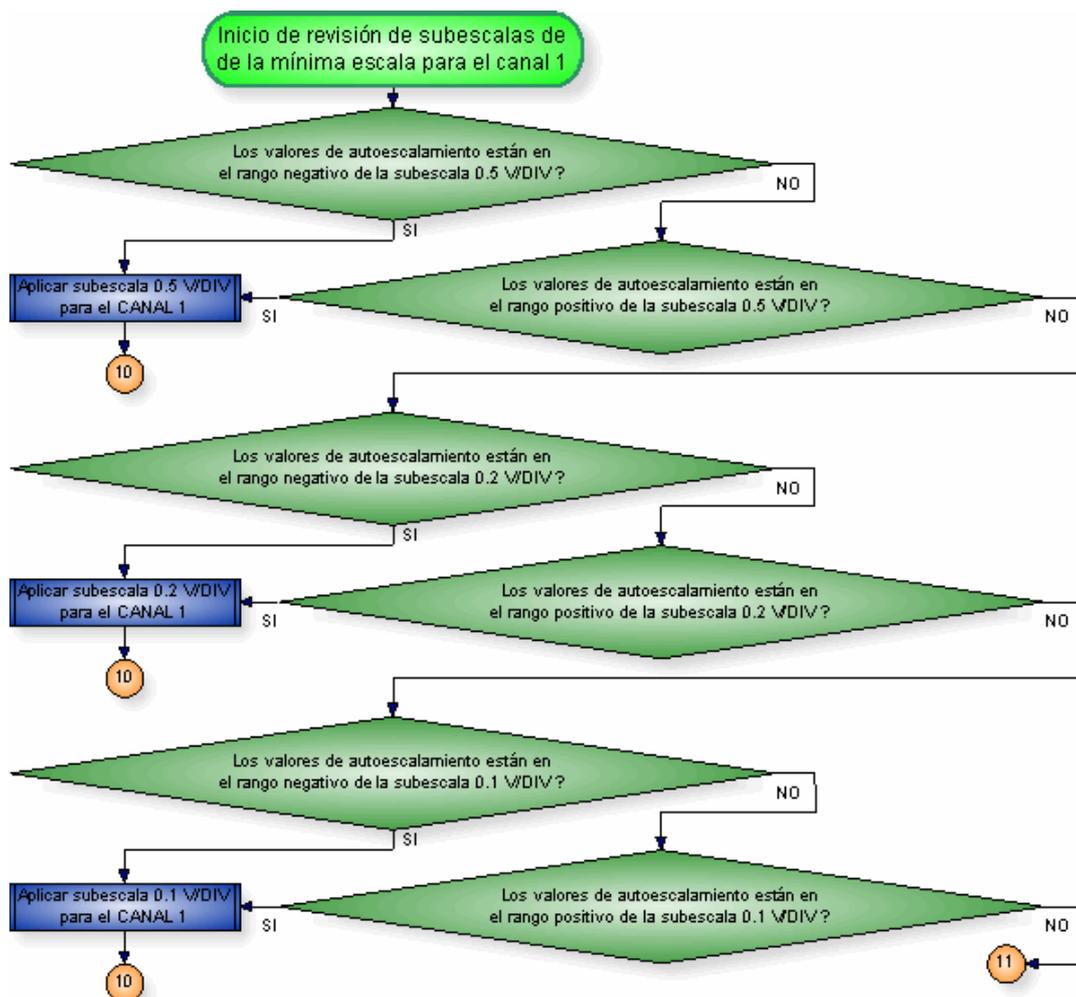


Figura 4.33 Revisión de subescalas del circuito acondicionador de escala mínima para el canal 1.

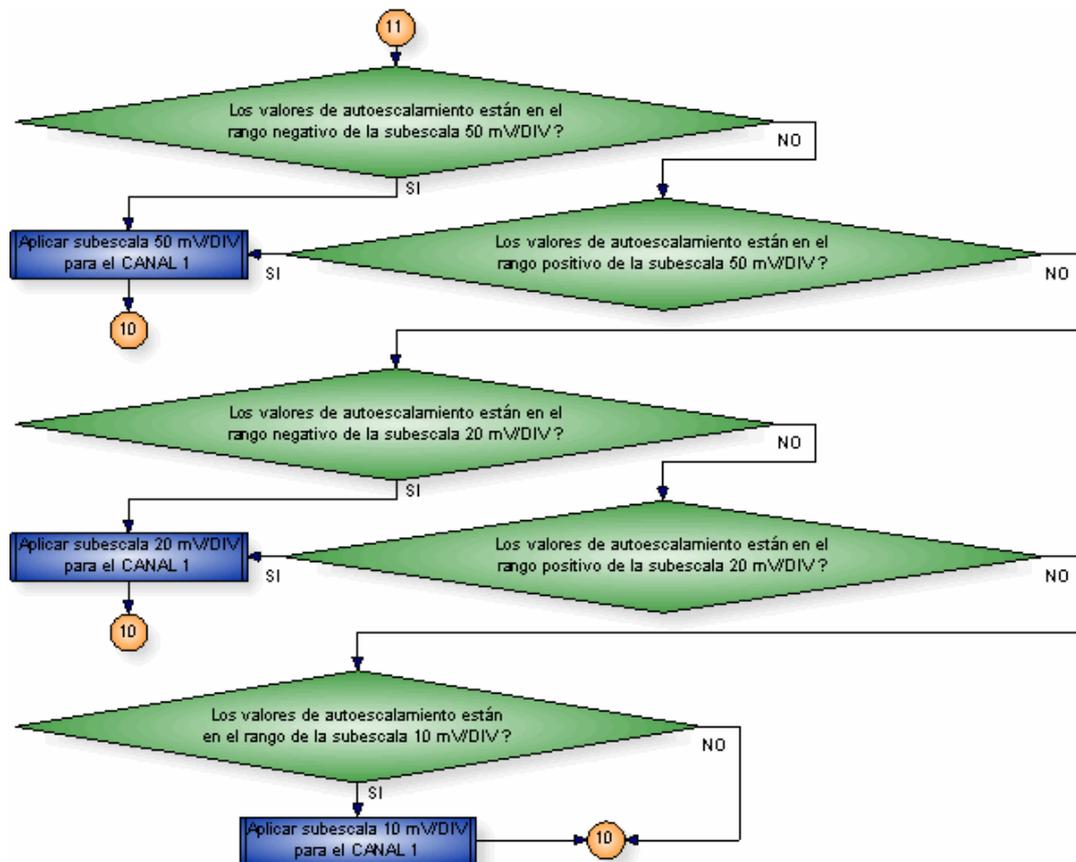


Figura 4.34 Revisión de subescalas del circuito acondicionador de escala mínima para el canal 1 (continuación).

La selección final de subescala para el circuito acondicionador de escala mínima se presenta en la Figura 4.33 y la Figura 4.34. Y se basa en los valores límite de las equivalencias en 10 bits como muestra la Tabla 4.3.

| Subescala (mV/DIV) | Voltaje (V) | Valor límite de 10 bits |
|--------------------|-------------|-------------------------|
| | | AUTOESCAL_MAXIMO |
| 500 | 2.5 | 1023 |
| 200 | 1 | 716 |
| 100 | 0.5 | 613 |
| 50 | 0.25 | 562 |
| 20 | 0.1 | 531 |
| 10 | 0.05 | 521 |
| | | AUTOESCAL_MINIMO |
| 10 | -0.05 | 502 |
| 20 | -0.1 | 492 |
| 50 | -0.25 | 461 |
| 100 | -5 | 410 |
| 200 | -10 | 307 |
| 500 | -25 | 0 |

Tabla 4.3 Equivalencias de valores máximos en 10 bits de resolución vertical, para las subescalas de la escala mínima (-2.5V a 2.5V) en base a la misma escala mínima.

4.3.10 TRIGGER

El TRIGGER no es más que una ayuda de visualización para el usuario que presentan los osciloscopios, consiste en fijar el inicio de graficación de una señal en pantalla, en base principalmente a tres parámetros: selección de señal (canal), nivel de voltaje y flanco de la señal.

A continuación se presentan todos los eventos y algoritmos que controlan todos los parámetros para la aplicación del trigger por software.

4.3.10.1 Evento de encendido/apagado del trigger

Al presionar el botón "ON/OFF" del trigger, se genera el evento descrito en el diagrama de flujo de la Figura 4.35. Al tratarse de un mismo botón para encendido y apagado, primero se comprueba si se está apagando o prendiendo el trigger.

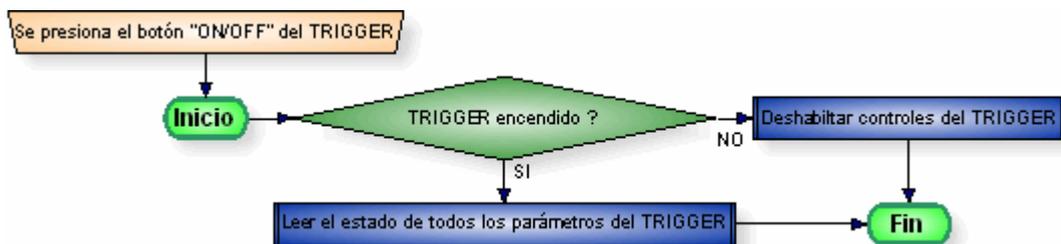


Figura 4.35 Evento de encendido y apagado del trigger.

A continuación se presenta en lenguaje estructurado el evento de encendido y apagado del trigger:

Deshabilitar controles del TRIGGER:

- Deshabilitar controles de selección de canal (canal 1, canal 2 o canal 3) para el trigger
- Deshabilitar controles de selección de nivel (variable o fijo) para el trigger
- Deshabilitar controles de selección de flanco (positivo o negativo) para el trigger

Fin de tarea

Leer el estado de todos los parámetros del TRIGGER:

- Leer el estado de los parámetros de selección de canal del trigger
- Leer el estado de los parámetros de selección de nivel del trigger
- Leer el estado de los parámetros de selección de flanco del trigger

Fin de tarea

Fin del algoritmo.

4.3.10.2 Evento de aplicación del trigger respecto a un canal

Como ejemplo la Figura 4.36 presenta el algoritmo del evento de selección del trigger respecto al canal 1. Al seleccionar el canal 1 para aplicar el trigger, primero se comprueba (solo por seguridad, si llegara a suceder que si se apaga el trigger, los controles de sus parámetros no se deshabiliten) si el trigger está encendido, en cuyo caso la PC envía al dsPIC la orden de aplicar su algoritmo de trigger solo con datos resultantes del canal 1.

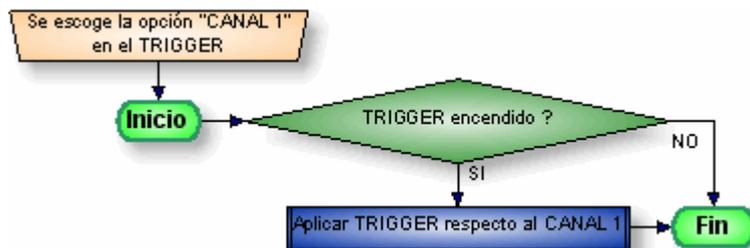


Figura 4.36 Evento de aplicación del trigger respecto al canal 1.

4.3.10.3 Evento de aplicación del trigger con flanco de subida (positivo)

Al seleccionar la opción de flanco positivo (Figura 4.37) para el trigger, y después de comprobar si el trigger está encendido (por la razón explicada en el literal anterior), la PC envía al dsPIC la orden para aplicar su algoritmo del trigger en base a la idea de empezar a enviar tramas hacia la PC, solo cuando el valor de un dato resultante del conversor AD (del canal que esté seleccionado) sea mayor que el inmediatamente anterior e igual al valor del nivel del trigger.

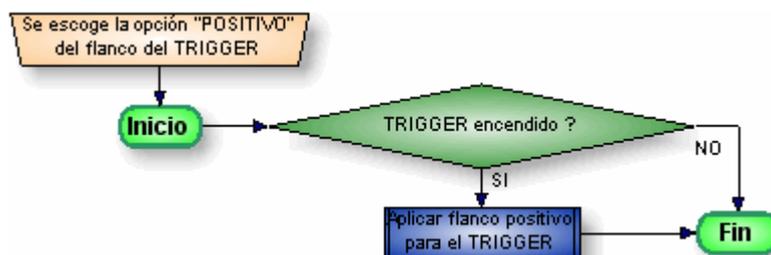


Figura 4.37 Evento de aplicación del trigger con flanco de subida

En el caso de seleccionar flanco negativo, en cambio el dsPIC empezará a enviar tramas a la PC, solo cuando el valor de un dato resultante del conversor AD sea menor que el inmediatamente anterior e igual al valor del nivel del trigger.

4.3.10.4 Evento de aplicación de nivel variable (manual) para el trigger

En este caso se habilitan (Figura 4.38) la herramienta de variación manual del valor del nivel del trigger, en este caso consiste en un scroll de variación continua ubicado al lado izquierdo de la pantalla.

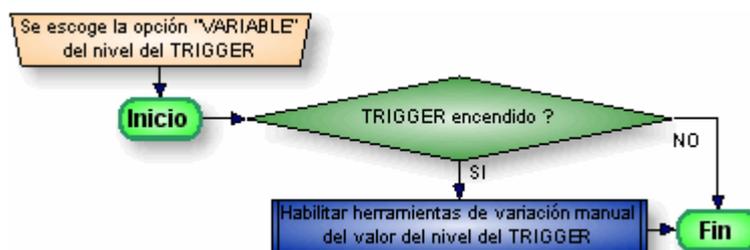


Figura 4.38 Evento de aplicación de nivel variable (manual) para el trigger.

4.3.10.5 Evento de aplicación de nivel fijo (automático) para el trigger

En este caso se arranca una base de tiempo (Figura 4.39), cuyo fin es tomar valores de datos llegados a la PC, y aplicarles un algoritmo para determinar el valor máximo y mínimo de los datos llegados durante esa base de tiempo, y calcular el valor medio entre los dos para tomarlo como valor de nivel de trigger automático.

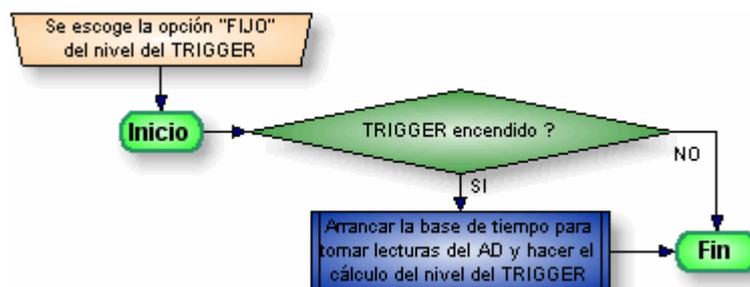


Figura 4.39 Evento de aplicación de nivel fijo para el trigger.

4.3.10.6 Finalización de base de tiempo y cálculo del nivel para el trigger en modo fijo (automático)

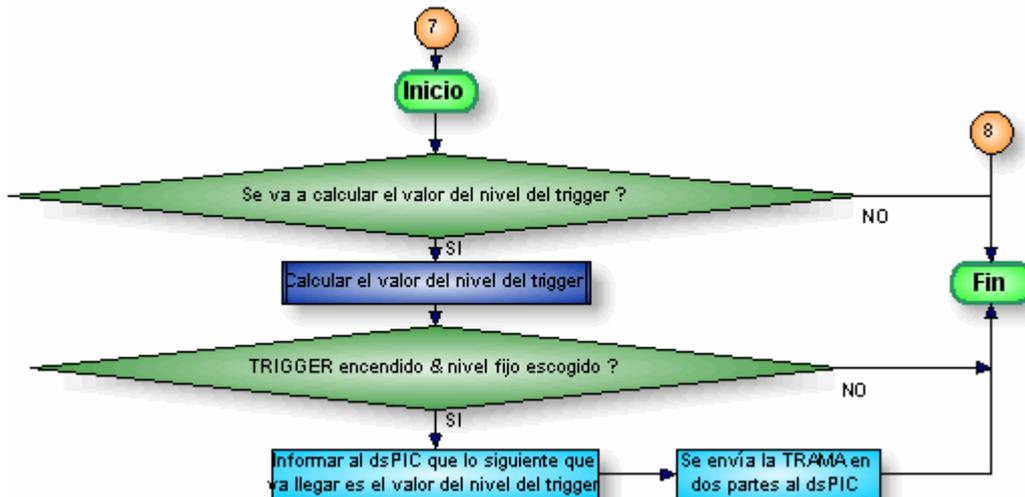


Figura 4.40 Cálculo del nivel del trigger en modo fijo (automático).

Al culminar la base de tiempo (que como se había indicado en el literal 4.3.9.2 es el mismo objeto temporizador que se usa para el inicio del autoescalamiento) primero se comprueba que, dicha base de tiempo, se inició para el cálculo del nivel automático del trigger y no para el autoescalamiento, en cuyo caso lo primero es calcular el valor del nivel del trigger como se muestra en el algoritmo de la Figura 4.40, en base a las mismas variables usadas para el autoescalamiento, “AUTOESCAL_MINIMO” y “AUTOESCAL_MAXIMO” (talvez los nombres de estas variables no tengan sentido para esta parte, pero se las aprovechó, porque almacenan los mismos valores que nos interesan también en este caso). Lo que se hace es sacar un promedio entre ambos valores y ese será el valor del nivel del trigger, pero éste, como es un valor de 10 bits, se lo divide en 2 partes de 5 bits, para poder ser enviadas una tras otra hacia el dsPIC.

Después se comprueba que esté encendido el trigger, y que esté elegido en la interfaz el nivel de trigger fijo, con lo cual se envía un número identificador que le informa al dsPIC que los 2 siguientes datos que le van a llegar son las 2 partes del valor del nivel del trigger, llamadas “LSB” y “MSB” en ese orden. Y finalmente, se envían al dsPIC las dos partes de 5 bits del valor del nivel del trigger (LSB y MSB).

4.3.11 RECEPCIÓN DE TRAMAS

A continuación se detallan todas las subrutinas y algoritmos que comprenden el evento de recepción de tramas en la PC.

4.3.11.1 Inicio del evento de recepción de tramas

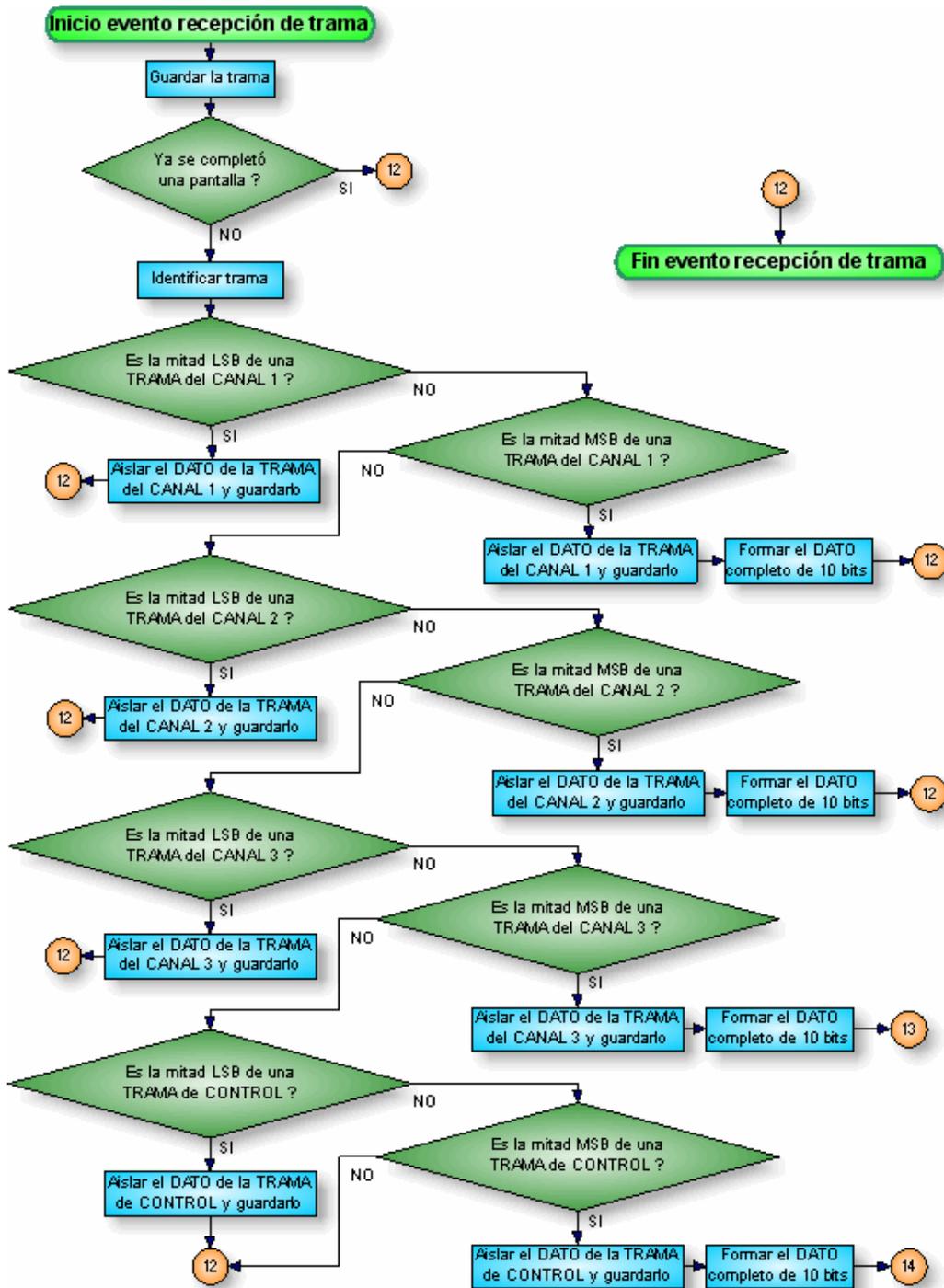


Figura 4.41 Inicio de evento de recepción de tramas.

Al iniciar el evento de recepción de tramas, lo primero es guardar la trama que llega en la variable "TRAMA" (Figura 4.41). Se revisa si ya fue completada una pantalla, en caso de no completarse todavía, se pueden recibir tramas para graficarlas, pero cuando ya se ha completado, se desechan las tramas que vayan llegando. Esto con la finalidad de que pase un tiempo de espera de 10 milisegundos entre pantalla y pantalla, durante el cual no se grafique nada en pantalla, para evitar el desbordamiento del buffer de recepción del puerto USB de la PC.

A continuación se aplica una comparación para el reconocimiento posterior de cada dato enmascarado en la trama, es decir, para identificar a que canal pertenece y si tiene enmascarada la parte LSB o la parte MSB del dato.

Después para cada caso se aplica el algoritmo que aisle al dato LSB de la trama, y el momento que llegue su respectiva parte MSB, también se aísla el dato respectivo y con los dos se aplica el algoritmo respectivo para formar el dato completo de 10 bits. El algoritmo para obtener dicho dato es así:

$$\text{Dato completo} = 32 \times \text{Dato MSB} + \text{Dato LSB}$$

Así para todos los casos, incluso para cuando llegan datos de control.

4.3.11.2 Inicio de trazado de la(s) señal(es)

Para graficar las señales en pantalla, primero se debe ver si se va a empezar a graficar una, en cuyo caso se debe ordenar limpiar la pantalla (Figura 4.42). Después de eso en cualquier caso, inicia la subrutina de graficación de las señales en pantalla.



Figura 4.42 Limpieza de pantalla para empezar a graficar otra.

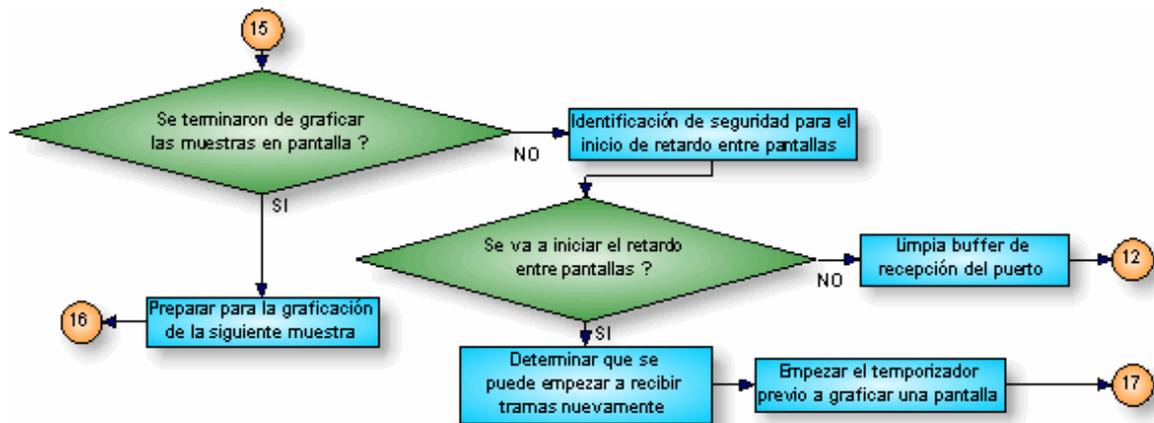


Figura 4.43 Inicio de graficación de la(s) señal(es).

A continuación se presenta en lenguaje estructurado el algoritmo de inicio de graficación de la(s) señal(es):

Se terminaron de graficar las muestras en pantalla?

Si se terminaron de graficar:

Preparar para la graficación de la siguiente pantalla:

Incrementar el orden del dato a graficarse en pantalla (para que el siguiente dato se grafique en la siguiente posición en pantalla del eje del tiempo)

Fin de tarea

Saltar a ver que canales están encendidos para graficar solo esas señales.

No se terminaron de graficar:

Identificación de seguridad para inicio de retardo entre pantallas:

Incrementar el valor una variable para asegurar que el retardo entre graficación de pantallas se inicie solo una vez cada fin de graficación de pantalla

Fin de tarea

Se va a iniciar el retardo entre pantallas ?

Si se va a iniciar:

Determinar que se puede empezar a recibir tramas nuevamente:

Definir mediante una variable que el programa puede empezar a recibir tramas para poder empezar a graficar la siguiente pantalla

Fin de tarea

Empezar el temporizador previo a graficar una pantalla:

Empezar el evento de retardo entre pantallas de 10 milisegundos

Fin de tarea

Saltar a revisar autoescalamiento

No se va a iniciar, o ya se inicio una vez:

Limpiar buffer de recepción del puerto:

Limpiar cualquier valor que haya sido recibido después de los que ya fueron graficados en pantalla y antes de empezar a graficar la siguiente

Fin de tarea

Fin de evento de recepción de tramas

Fin del algoritmo.

4.3.11.3 Trazado y definición de condiciones de escalamiento

Para empezar a graficar una señal, primero se comprueba si está encendido el canal por donde está ingresando, como muestra la Figura 4.44, donde se ve el caso de cuando está encendido el canal 1, el procedimiento es análogo para los otros dos canales (Figuras 4.45 y 4.46) Si no está encendido el canal 1, se salta a revisar si está encendido el canal 2 y así sucesivamente.

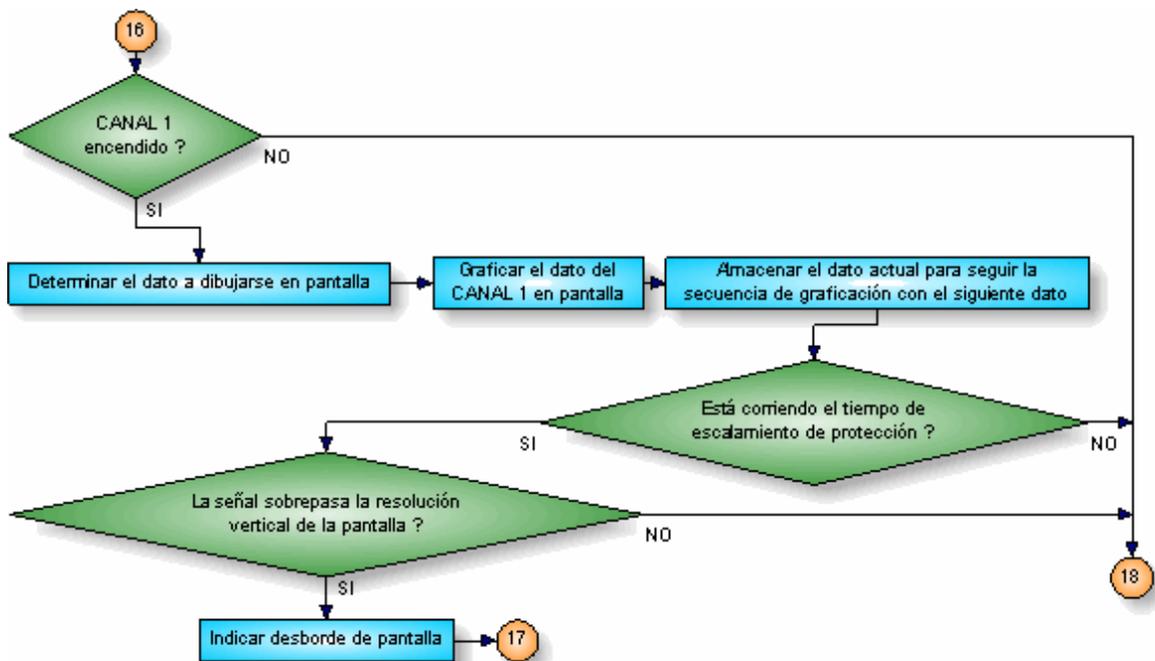


Figura 4.44 Graficación de señales para el canal 1 y definición de condiciones de escalamiento.

En caso de estar encendido el canal 1, primero se determina el dato a dibujarse en pantalla con la operación:

$$\text{DATO1} = ke1 \times (\text{DATOCANAL1} - 511.5) + 511.5 + \text{MOVER_EJE_1}$$

Donde:

DATO1 es el valor a graficarse en pantalla para el canal 1.

ke1 es la constante de resolución vertical para el canal 1.

DATOCANAL1 es el dato resultante del conversor AD para el canal 1.

MOVER_EJE_1 es el valor que representa la posición vertical en pantalla del nivel de referencia 0 voltios, según donde lo haya movido el usuario.

Continuando con el procedimiento, se grafica el valor correspondiente (DATO1) en pantalla, y se lo almacena para ser usado en el trazo del siguiente valor en pantalla.

A continuación se verifica si está corriendo la base de tiempo del escalamiento de protección, en cuyo caso se comprueba si el valor a graficarse en pantalla sobrepasa por alto o por bajo en algún momento a la pantalla, y se identifica este hecho con una variable (DESBORDE_ESCALA_C1), la cual se usa como uno de los parámetros para la aplicación o no del escalamiento de protección. En caso de no estar corriendo dicha base de tiempo, se procede a verificar si se ordenó procedimiento de autoescalamiento para alguno de los canales.

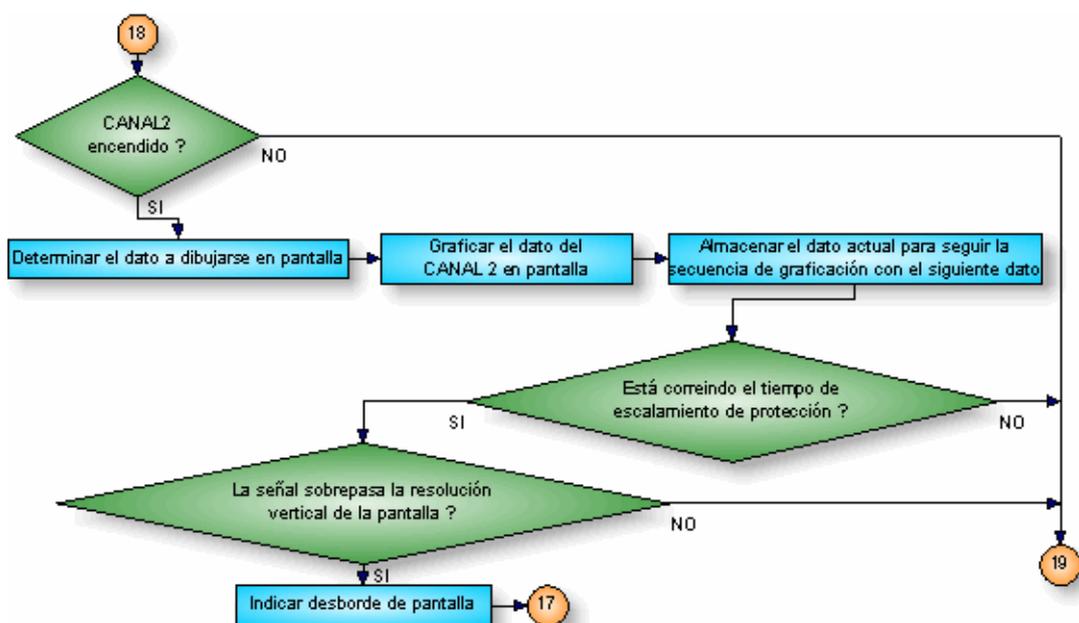


Figura 4.45 Graficación de señales para el canal 2 y definición de condiciones de escalamiento.

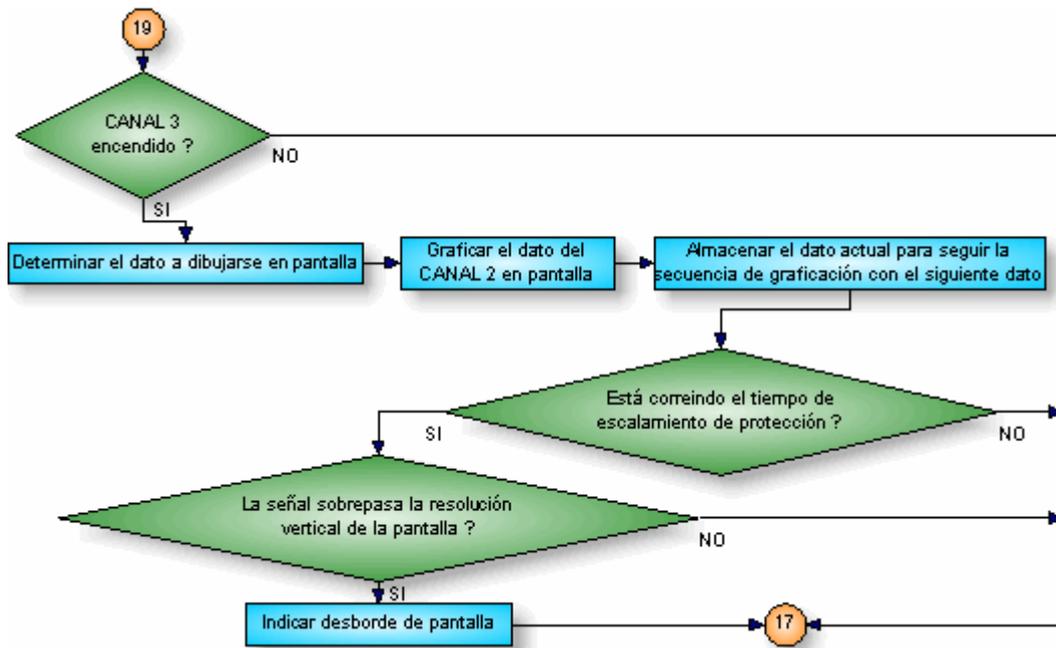


Figura 4.46 Graficación de señales para el canal 3 y definición de condiciones de escalamiento.

4.3.11.4 Definición de condiciones de autoescalamiento

Continuando con los algoritmos del evento de recepción de tramas, se tiene la definición de condiciones de autoescalamiento como se muestra en la Figura 4.47. Primero se determina si es que se está en proceso de autoescalamiento de alguno de los 3 canales, y, si es el caso, se revisa, de que canal se trata. Habiendo determinado de que canal se trata y suponiendo que es el canal 1, se compara si el dato que llegó en ese momento, es mayor que el valor máximo de los datos que han llegado antes, en cuyo caso se actualiza este valor máximo, en caso de no cumplirse esto, se revisa y si este dato es menor que el valor mínimo de los datos que han llegado antes, en cuyo caso se actualiza este valor mínimo. Si acaso no se cumple ninguna de las dos condiciones, las variables que van almacenando los valores máximo y mínimo quedan igual que antes.

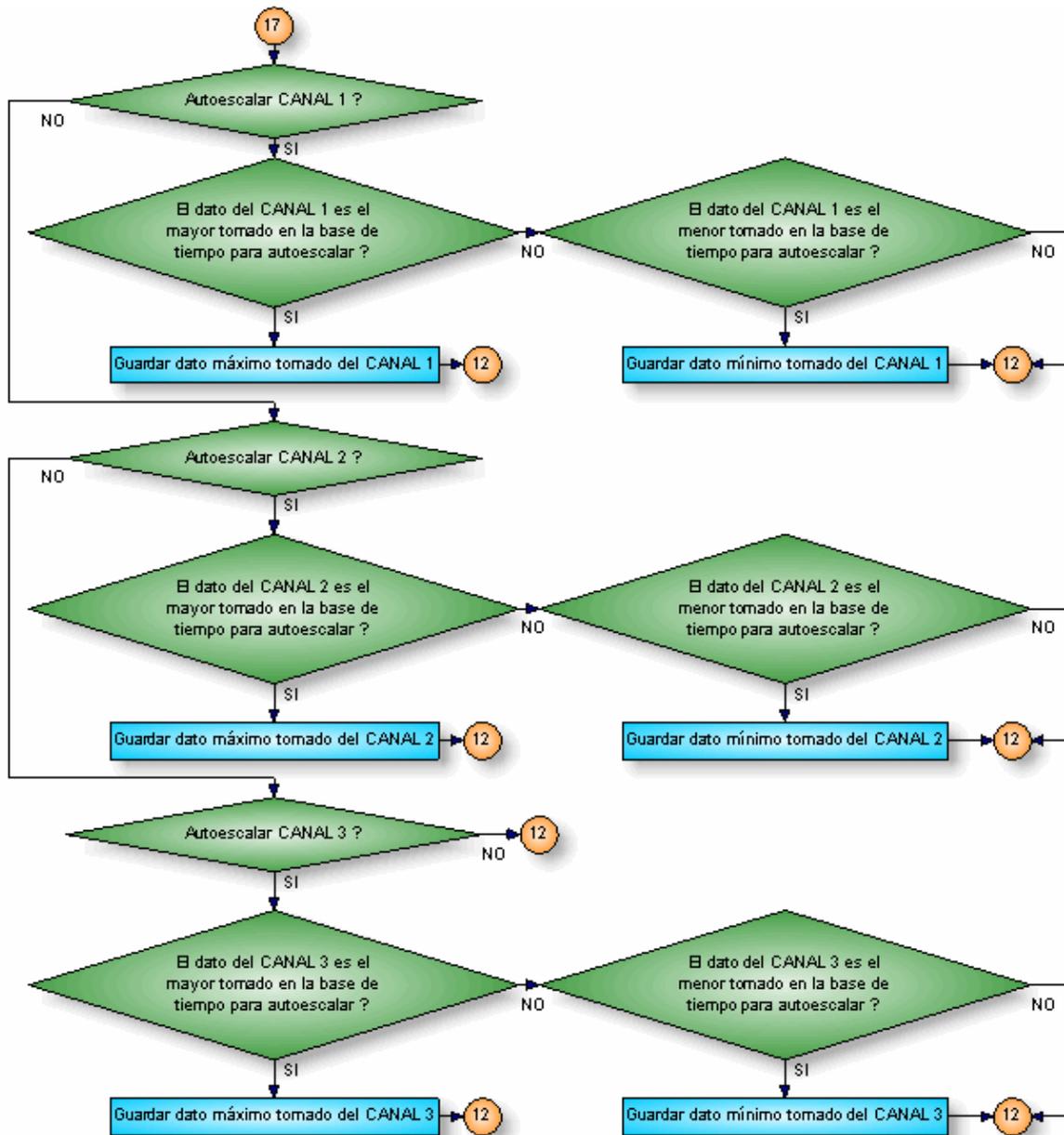


Figura 4.47 Definición de condiciones de autoescalamiento.

En los capítulos anteriores se trató ya todo lo referente al diseño del Hardware y del Firmware del presente proyecto. En este capítulo se detalló los procesos de cada uno de los eventos y subrutinas más importantes que permiten el control y accionamiento de todos los parámetros del osciloscopio desde la interfaz gráfica en la PC. Entonces, habiendo cubierto ya detalladamente todas las partes que conforman el osciloscopio, en el siguiente capítulo se detallan las pruebas y resultados obtenidos del funcionamiento del presente proyecto en laboratorio.

CAPÍTULO 5

PRUEBAS Y RESULTADOS DEL OSCILOSCOPIO

PRUEBAS Y RESULTADOS DEL OSCILOSCOPIO

5.1 ELEMENTOS IMPORTANTES A CONSIDERAR PREVIOS A LAS PRUEBAS

5.1.1 RESUMEN DE INCOVENIENTES Y SOLUCIONES DE LA ETAPA DE DISEÑO

En un inicio, mientras se comprobaba el funcionamiento de todos y cada uno de los componentes, se detectó que algunas veces, cuando un componente no actuaba como se esperaba, el problema provenía del microcontrolador. El dsPIC30F3011 no realizaba lo que se le solicitaba. Ciertos pines del PIC que debían cambiar de valor no lo hacían. Otras veces, la forma en que se enviaban las tramas, estaba incorrecta, debido a que algunos de los pines que estaban utilizándose como puerto para enviar las tramas se habían programado de forma incorrecta y en vez de actuar como salidas, lo hacían como entradas. Algunos de estos comportamientos anómalos se debieron a una mala definición o programación de los puertos de entrada/salida del microcontrolador, especialmente en el caso específico de los puertos de datos, los cuales, la mayoría del tiempo están como salidas, pero en ciertos momentos (cuando la PC envía órdenes), deben configurarse como entradas.

En el diseño e implementación del hardware de acondicionamiento de cada canal también se tuvieron algunos errores, como por ejemplo, el haber elegido al inicio amplificadores operacionales cuyo ancho de banda era muy limitado, lo cual no permite tener una apreciación real de la forma de onda de entrada. El otro error fue tratar de habilitar directamente a los relés con señales directo desde el dsPIC (errores que hacían ver como que el dsPIC fallaba cuando en realidad no era así), lo cual fue solucionado cambiando a transistores BJT.

Una vez que se realizó los cambios correspondientes en el hardware, y se reprogramó el microcontrolador con la configuración correcta, el funcionamiento ya no tuvo problemas.

5.1.2 CALIBRACIÓN

Una vez que el osciloscopio estuvo funcionando, era necesario en primer lugar realizar la calibración del equipo.

Puesto que la PC ya podía graficar las variables medidas, restaba hacer los cálculos y calibraciones correspondientes para obtener los valores y tiempos correctos según las escalas en pantalla. La etapa de entrada de cada canal tiene un potenciómetro multi-vuelta de alta precisión que permite ajustar el nivel de tensión antes de entrar al conversor analógico digital. Con este potenciómetro se calibró tal que, aplicando una misma señal de voltaje en las entradas de los tres canales, estos muestren la misma forma de onda y con la misma magnitud bajo iguales condiciones de escalamiento.

Se realizaron varias mediciones aplicando señales desde un generador de funciones. Las señales utilizadas fueron de diferente forma de onda, de diferente amplitud y frecuencia. Las calibraciones del osciloscopio para PC se hicieron con la ayuda del osciloscopio comercial TEKTRONIX TDS210.

En cuanto al tiempo de muestreo, se hizo un análisis basado en las características del conversor AD del dsPIC30F3011 y en las mediciones hechas en laboratorio. Para el caso del módulo conversor AD del dsPIC30F3011 (ver ANEXO A) el tiempo mínimo de adquisición de datos (T_{AD}) es de 165 [ns], y el tiempo de conversión (T_{CONV}) requiere de $12 \cdot T_{AD}$, así que $T_{CONV} = 1.98[\mu s]$, por lo tanto el tiempo de muestreo del AD (T_m) es: $T_m = T_{AD} + T_{CONV}$, de lo cual: $T_m = 2.145 [\mu s]$. Pero el Tiempo de muestreo efectivo (T_e) obtenido en este caso es de 5.5[μs] de lo cual, Si se calcula $T_e - T_m$, da un retardo para convertir cada dato, de 3.355[μs]. Pero, como cada dato para transmitir a la PC se lo divide en 2 partes, el tiempo de muestreo mínimo real del sistema (T_R) es el doble del efectivo, $T_R = 2T_e = 11 [\mu s]$. Esto en modo de almacenamiento, pero para modo en tiempo real se determinó en las pruebas que $T_R = 22 [\mu s]$.

5.2 ESPECIFICACIONES DEL OSCILOSCOPIO

A continuación en la Tabla 5.1 se presenta un resumen de las especificaciones finales del osciloscopio armado.

| Especificación | Valor |
|---------------------------------------|-------------------------------------|
| Resolución vertical | 10 bits |
| Conexión a la PC | USB 1.1 (USB 2.0 compatible) |
| Velocidad máxima de muestreo efectiva | 90900 SPS |
| Entradas | 3 canales con entradas BNC |
| Rangos de tensión | ± 170 V |
| Precisión de tensión | 3% |
| Tamaño del buffer | 32K |
| Impedancia de entrada | 1 M Ω |
| Alimentación eléctrica | -5VDC, 0, 5VDC |
| Consumo de potencia | 10-11W |
| Dimensiones | 180mm x 124mm x 104mm |
| Software | Windows 98/2000/XP |
| Aislamiento | No dispone de aislamiento eléctrico |
| Señal de calibración | No dispone de señal de calibración |

Tabla 5.1 Especificaciones del osciloscopio.

Cada una de las especificaciones es limitada o definida por un componente particular, a saber:

- Resolución vertical – Limitada por el conversor AD del dsPIC.
- Conexión con la PC - Limitada por la velocidad del puerto USB de la PC.
- Velocidad máxima de muestreo - Limitada por la frecuencia del oscilador (en este caso, un cristal de 20 Mhz) y por la frecuencia máxima de trabajo del ADC (500 khz).
- Precisión de Amplitud - Definido por el error de cuantización del ADC (± 1 LSB) más un error (estimado) que aparece en la etapa de entrada.
- Tamaño del buffer - Limitado por el ancho de los bytes en el buffer de recepción de la PC.
- Impedancia de entrada - Definido por la impedancia de entrada de los amplificadores operacionales en la etapa de entrada.
- Alimentación eléctrica - Definido por la fuente de PC que alimenta el módulo.
- Consumo de potencia - Definido por el consumo de los componentes activos, analizados en el Capítulo 2.
- Dimensiones - Definido por el tamaño de la carcasa.
- Software - Definido por las características del lenguaje Visual Basic 6.0.

5.3 REALIZACIÓN DE LAS PRUEBAS

A continuación se describe las pruebas que se diseñaron para probar el correcto funcionamiento del equipo.

5.3.1 EQUIPO QUE SE UTILIZÓ EN LA REALIZACIÓN DE LAS PRUEBAS

En total se utilizaron 10 elementos, descritos a continuación:

- Un osciloscopio digital (TEKTRONIX TDS 210).
- Dos generadores de señales.
- 4 puntas de prueba simples X1 (2 para las salidas de los generadores de señales y 2 para conectar al módulo).
- 2 puntas de prueba X10.
- Software WAVESTAR versión 1.0 (para bajar a la PC las formas de onda obtenidas en el osciloscopio).

5.3.2 INCONVENIENTES Y LIMITACIONES DEL PROCESO DE REALIZACIÓN DE LAS PRUEBAS

Lo ideal hubiera sido disponer de otro generador de señales a parte de los dos utilizados, para así poder probar simultáneamente los tres canales del osciloscopio. Pero, tampoco se disponía de 3 canales en el otro osciloscopio comercial para hacer comparaciones simultáneas. Sin embargo de este inconveniente, se puede afirmar que las pruebas realizadas y los resultados obtenidos son suficientes para evaluar los beneficios y limitaciones del osciloscopio diseñado.

5.4 RESULTADOS DE LAS PRUEBAS REALIZADAS

5.4.1 PRUEBA CON UNA ONDA TRIANGULAR A 60Hz Y AMPLITUD 2.5V

- FORMA DE ONDA EN EL OSCILOSCOPIO DIGITAL TEKTRONIX TDS 210

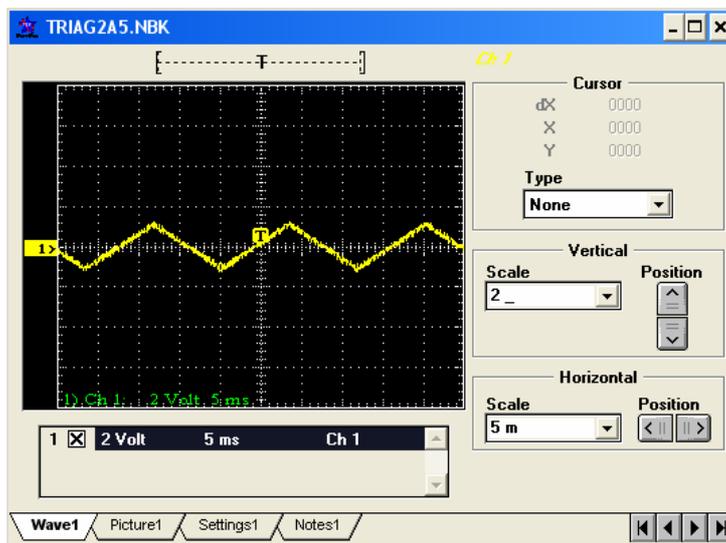


Figura 5.1 Onda triangular a 60 Hz y 2.5V de amplitud, tomada con el osciloscopio TDS210.

- FORMA DE ONDA EN EL OSCILOSCOPIO A PRUEBA

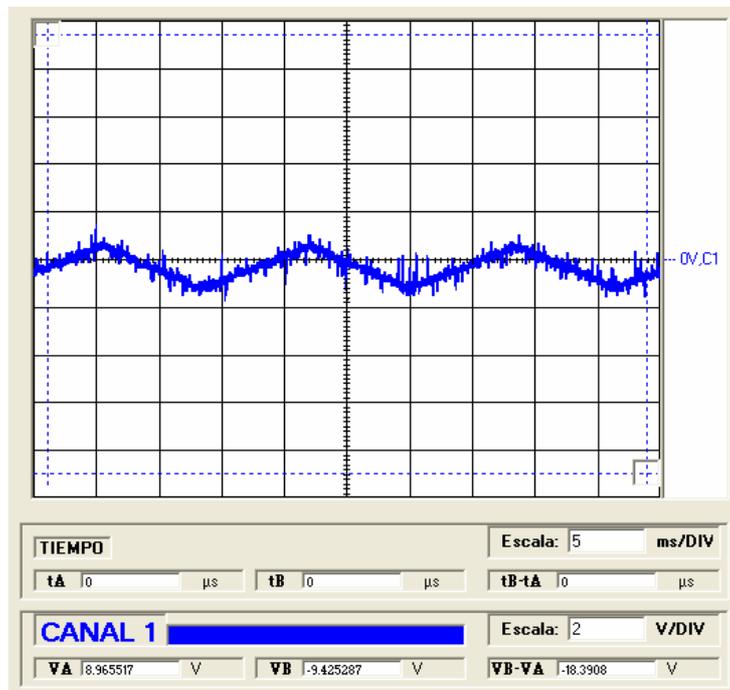


Figura 5.2 Onda triangular a 60 Hz y 2.5V de amplitud, tomada con el osciloscopio para PC.

5.4.2 PRUEBA CON UNA ONDA A 1KHz Y AMPLITUD 2.5V

➤ FORMA DE ONDA EN EL OSCILOSCOPIO DIGITAL TEKTRONIX TDS 210

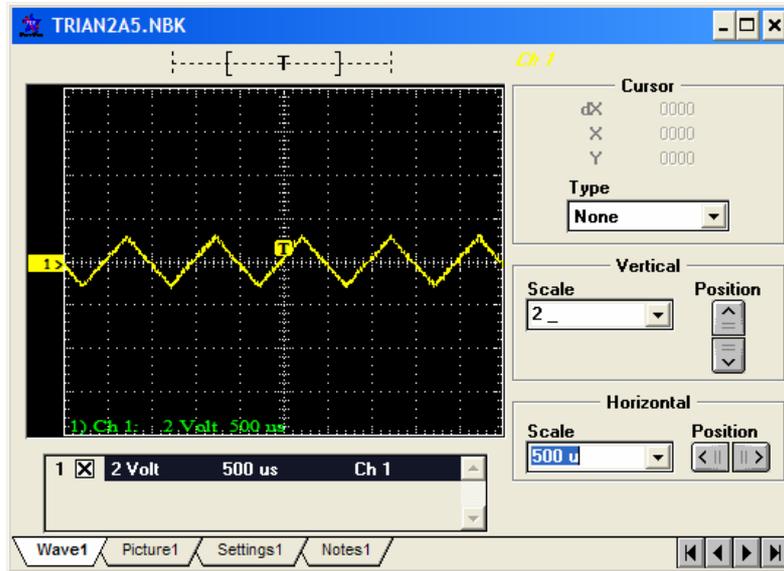


Figura 5.3 Onda triangular a 1 kHz y 2.5V de amplitud, tomada con el osciloscopio TDS210.

➤ FORMA DE ONDA EN EL OSCILOSCOPIO A PRUEBA

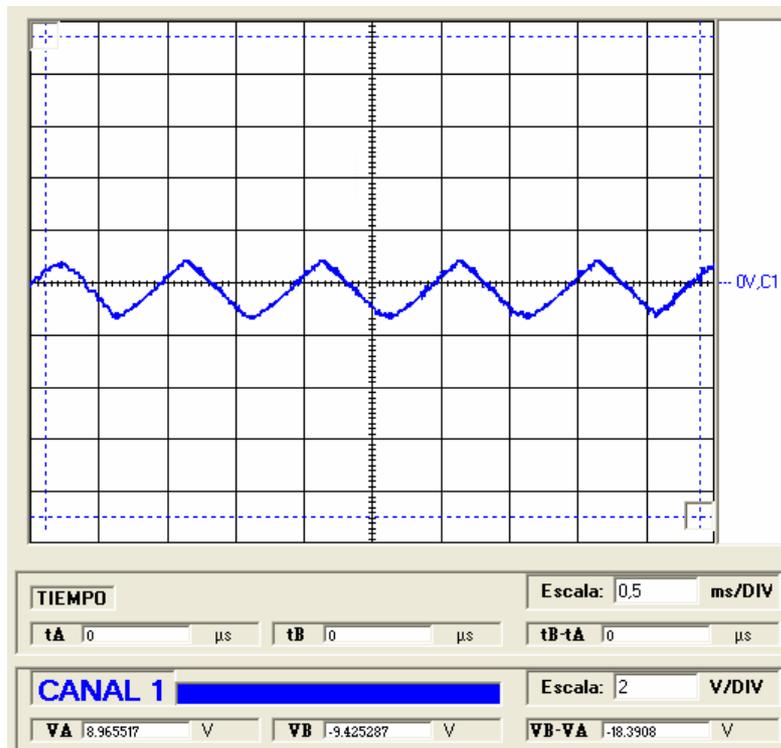


Figura 5. 4 Onda triangular a 1 kHz y 2.5V de amplitud tomada con el osciloscopio para PC.

5.4.3 PRUEBA CON UNA ONDA A 5KHz Y AMPLITUD 2.5V

➤ FORMA DE ONDA EN EL OSCILOSCOPIO DIGITAL TEKTRONIX TDS 210

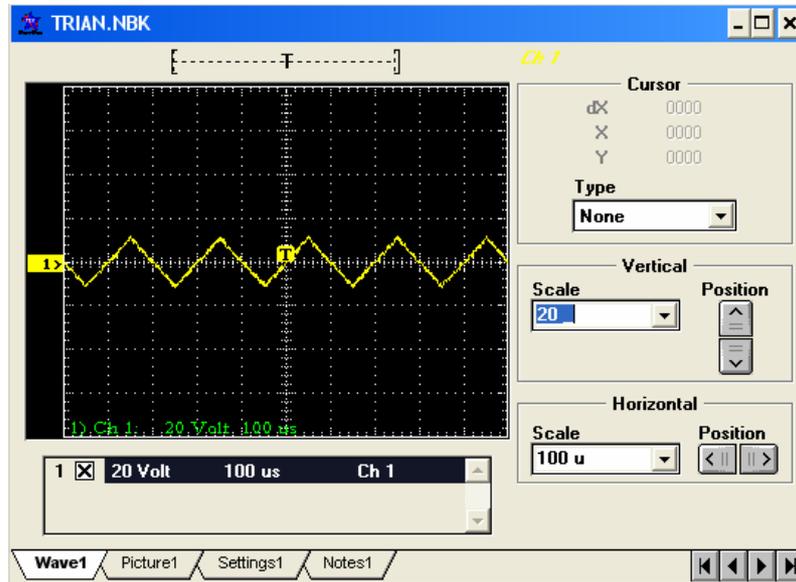


Figura 5.5 Onda triangular a 5 kHz y 2.5V de amplitud tomada con el osciloscopio TDS210.

➤ FORMA DE ONDA EN EL OSCILOSCOPIO A PRUEBA

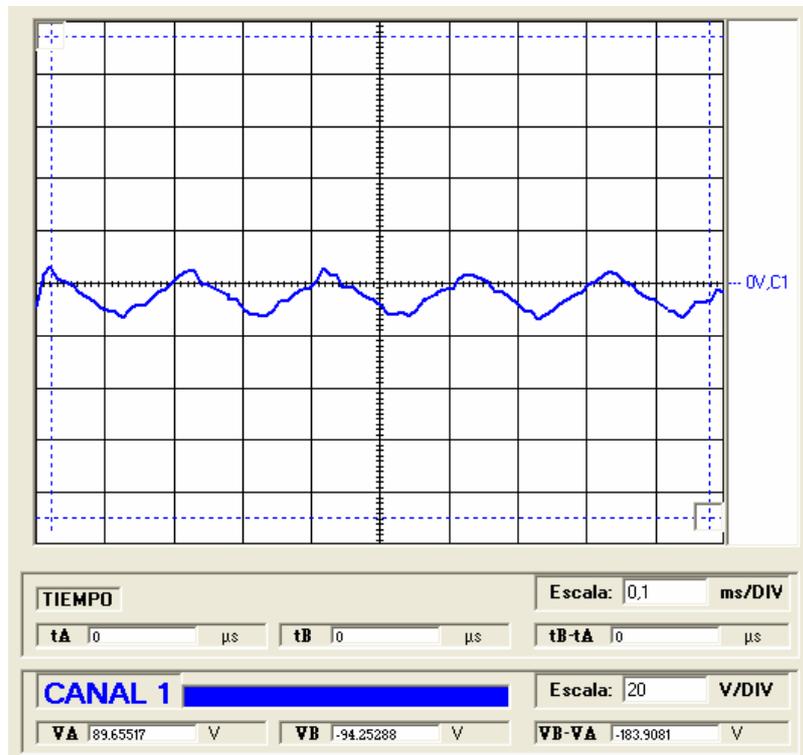


Figura 5.6 Onda triangular a 5 kHz y 2.5V de amplitud tomada con el osciloscopio para PC.

5.4.4 PRUEBA CON UNA ONDA A 10KHz Y AMPLITUD 25V

- FORMA DE ONDA EN EL OSCILOSCOPIO DIGITAL TEKTRONIX TDS 210

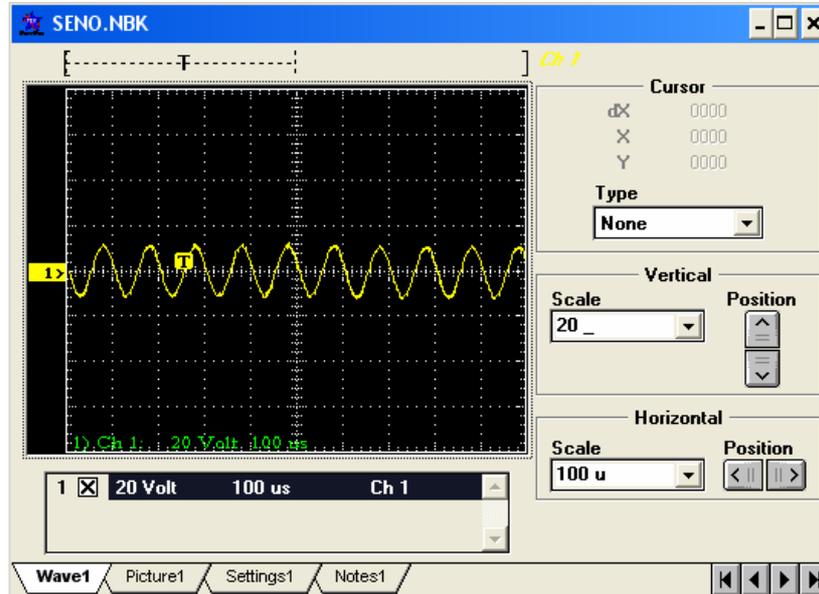


Figura 5.7 Onda senoidal a 10 kHz y 25V de amplitud tomada con el osciloscopio TDS210.

- FORMA DE ONDA EN EL OSCILOSCOPIO A PRUEBA

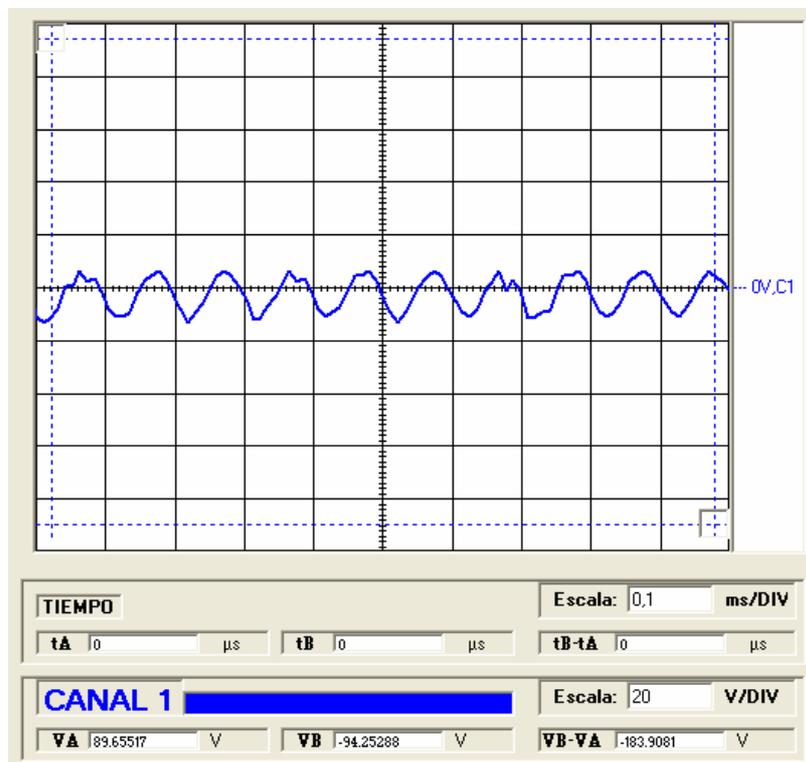


Figura 5.8 Onda senoidal a 10 kHz y 25V de amplitud tomada con el osciloscopio para PC.

5.4.5 PRUEBA CON UNA ONDA A 20KHz Y AMPLITUD 25V

➤ FORMA DE ONDA EN EL OSCILOSCOPIO DIGITAL TEKTRONIX TDS 210

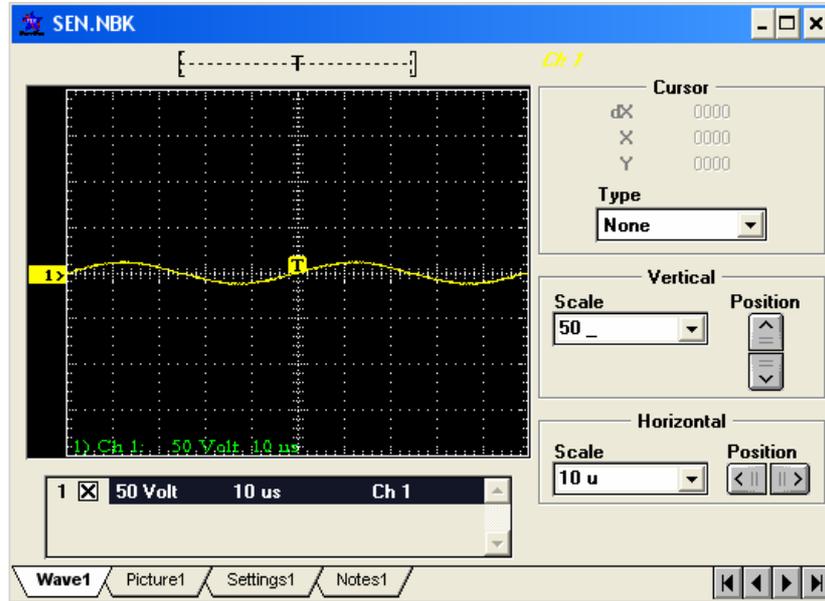


Figura 5.9 Onda senoidal a 20 kHz y 25V de amplitud tomada con el osciloscopio TDS210.

➤ FORMA DE ONDA EN EL OSCILOSCOPIO A PRUEBA

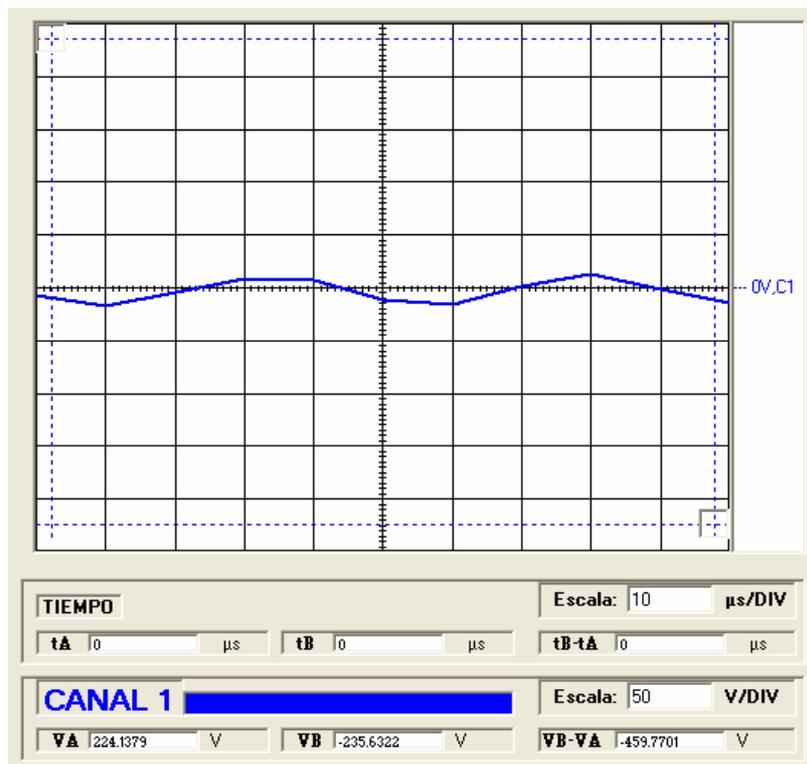


Figura 5.10 Onda senoidal a 20 kHz y 25V de amplitud tomada con el osciloscopio para PC.

5.4.6 PRUEBA CON UNA ONDA A 30KHz Y AMPLITUD 25V

➤ FORMA DE ONDA EN EL OSCILOSCOPIO DIGITAL TEKTRONIX TDS 210

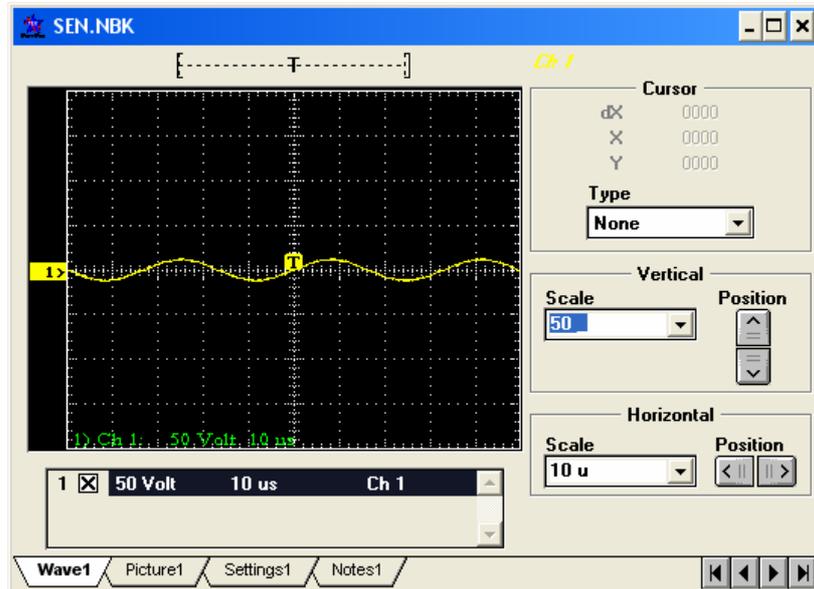


Figura 5.11 Onda senoidal a 30 kHz y 25V de amplitud tomada con el osciloscopio TDS210.

➤ FORMA DE ONDA EN EL OSCILOSCOPIO A PRUEBA

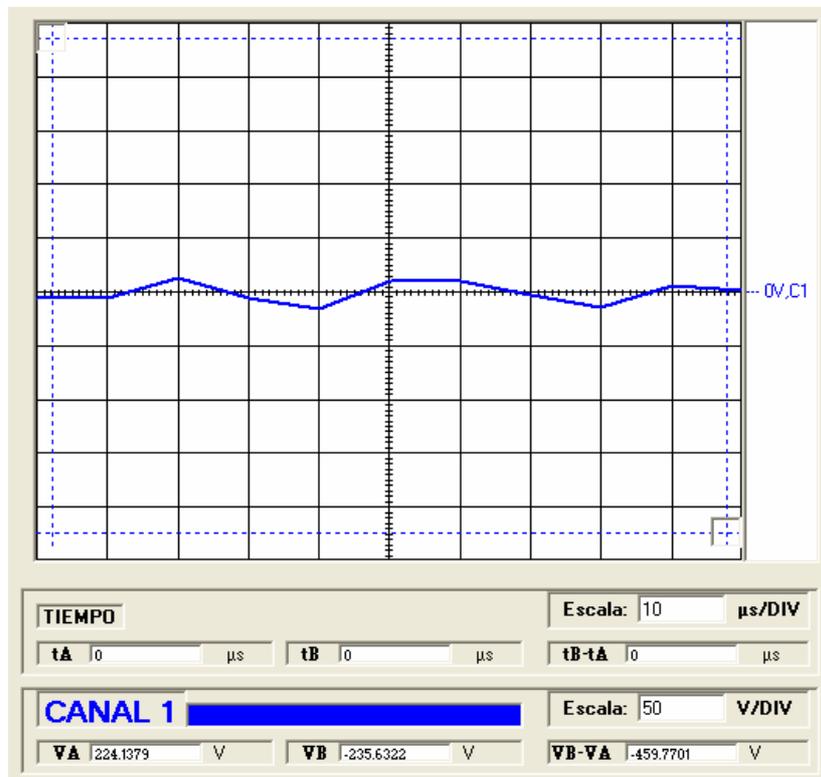


Figura 5.12 Onda senoidal a 30 kHz y 25V de amplitud tomada con el osciloscopio para PC.

5.4.7 PRUEBA CON UNA ONDA A 40KHz Y AMPLITUD 25V

➤ FORMA DE ONDA EN EL OSCILOSCOPIO DIGITAL TEKTRONIX TDS 210

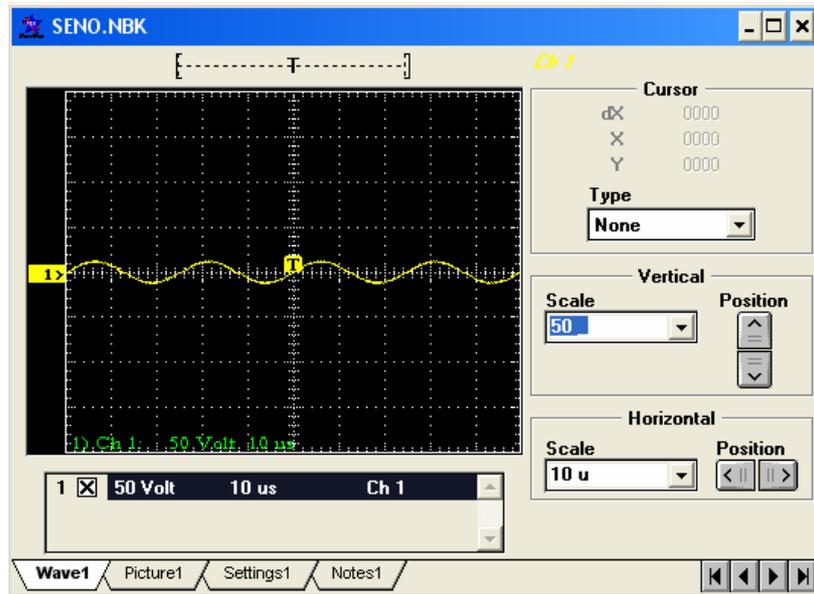


Figura 5.13 Onda senoidal a 40 kHz y 25V de amplitud tomada con el osciloscopio TDS210.

➤ FORMA DE ONDA EN EL OSCILOSCOPIO A PRUEBA

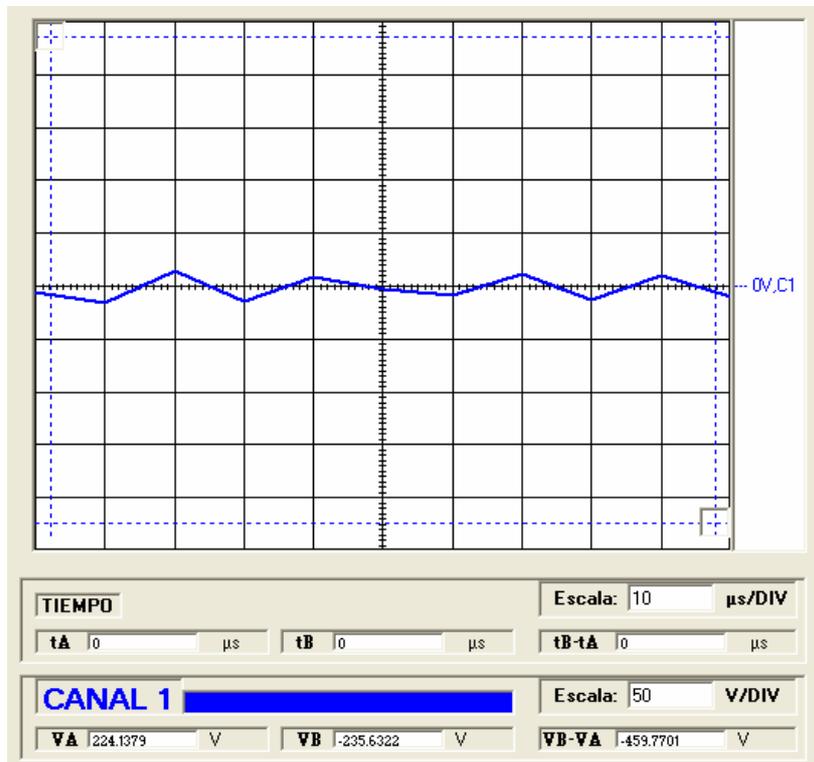


Figura 5.14 Onda senoidal a 40 kHz y 25V de amplitud tomada con el osciloscopio para PC.

5.4.8 PRUEBA CON DOS FORMAS DE ONDA, UNA SENOIDAL DE 1KHz Y UNA CUADRADA DE 10KHz, AMBAS CON AMPLITUD DE 25V

➤ **FORMA DE ONDA EN EL OSCILOSCOPIO DIGITAL TEKTRONIX TDS 210**

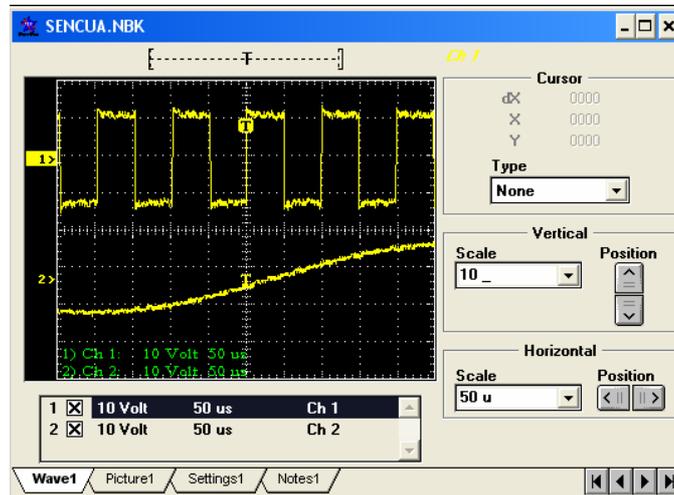


Figura 5.15 Onda senoidal a 1 kHz y una cuadrada a 10 kHz, las dos con 25V de amplitud tomadas con el osciloscopio TDS210.

➤ **FORMA DE ONDA EN EL OSCILOSCOPIO A PRUEBA**

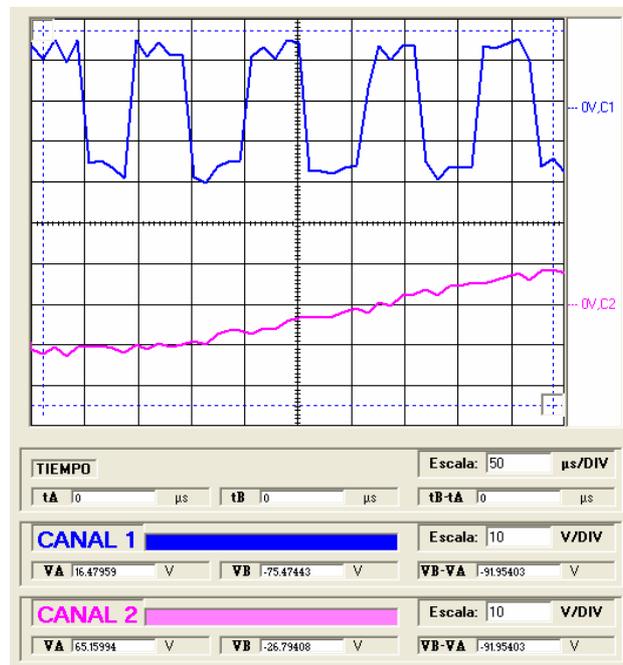


Figura 5.16 Onda senoidal a 1 kHz y una cuadrada a 10 kHz, las dos con 25V de amplitud tomadas con el osciloscopio para PC.

5.4.9 PRUEBA CON DOS FORMAS DE ONDA, UNA CUADRADA DE 60Hz Y UNA TRIANGULAR DE 5KHz, AMBAS CON AMPLITUD DE 25V

➤ **FORMA DE ONDA EN EL OSCILOSCOPIO DIGITAL TEKTRONIX TDS 210**

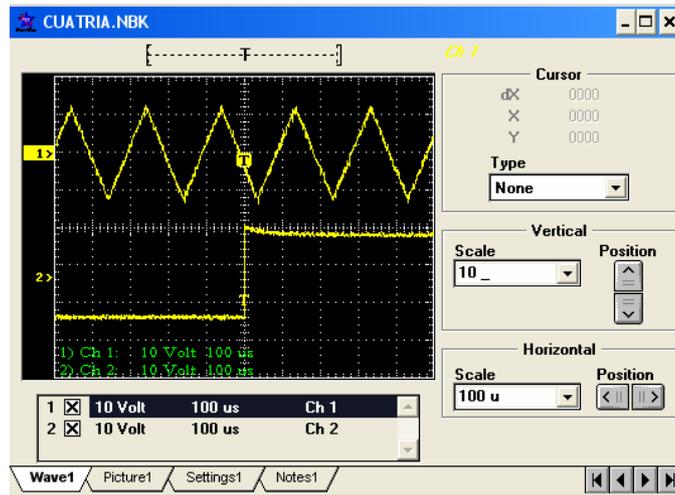


Figura 5.17 Onda cuadrada a 60 Hz y una triangular a 5 kHz, las dos con 25V de amplitud tomadas con el osciloscopio TDS210.

➤ **FORMA DE ONDA EN EL OSCILOSCOPIO A PRUEBA**

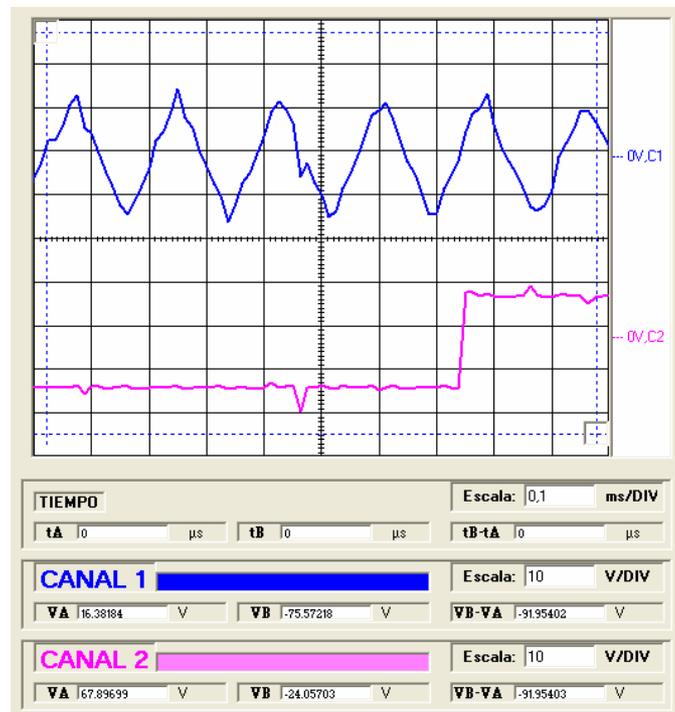


Figura 5.18 Onda cuadrada a 60 Hz y una triangular a 5 kHz, las dos con 25V de amplitud tomadas con el osciloscopio para PC.

5.4.10 PRUEBA CON DOS FORMAS DE ONDA, UNA CUADRADA DE 20KHz Y UNA CUADRADA DE 40KHz, AMBAS CON AMPLITUD DE 25V

➤ **FORMA DE ONDA EN EL OSCILOSCOPIO DIGITAL TEKTRONIX TDS 210**

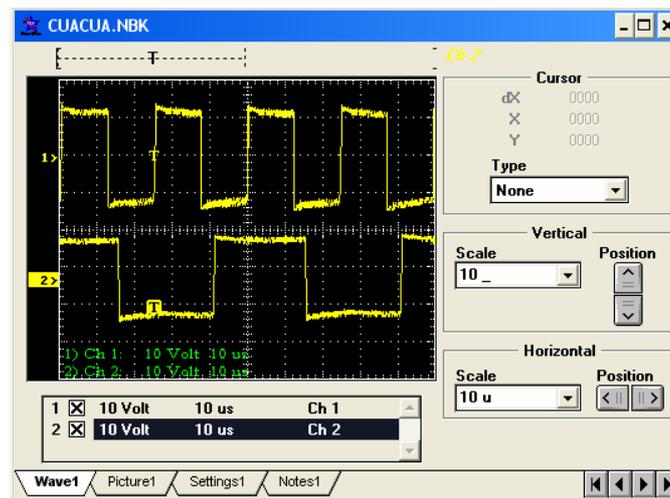


Figura 5.19 Onda cuadrada a 20 KHz y una cuadrada a 40 kHz, las dos con 25V de amplitud tomadas con el osciloscopio TDS210.

➤ **FORMA DE ONDA EN EL OSCILOSCOPIO A PRUEBA**

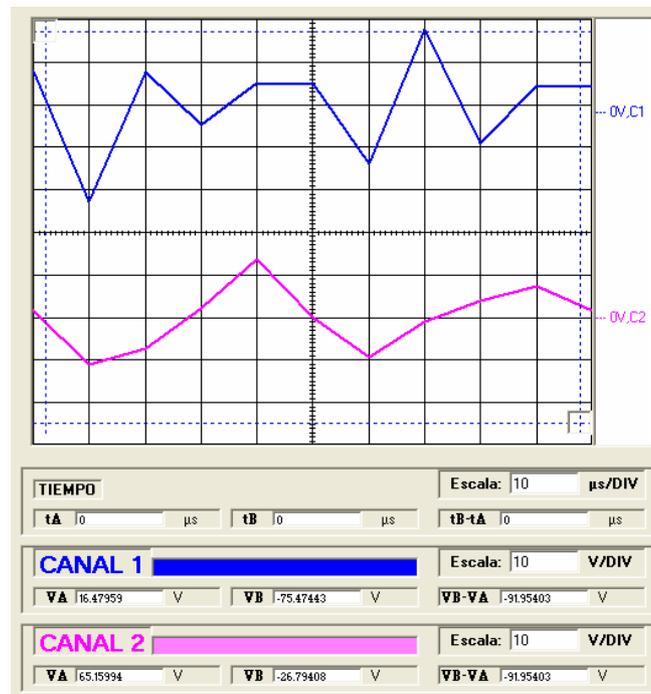


Figura 5.20 Onda cuadrada a 20 KHz y una cuadrada a 40 kHz, las dos con 25V de amplitud tomadas con el osciloscopio para PC.

5.5 ANÁLISIS DE RESULTADOS Y ERRORES

5.5.1 IDENTIFICACIÓN DE ERRORES EN EL MUESTREO

En un sistema de conversión analógica-digital, se presentan algunos tipos de errores, ya analizados en el Literal 1.6 del Capítulo 1. Pero el más notable en el presente proyecto es el **error de cuantización**. Este error es notorio e importante en ésta aplicación, ya que las señales de voltaje, las cuales van a ser objeto de medición, son fenómenos electromagnéticos que ocurren en la naturaleza en una forma continua. Estas señales, al ser digitalizadas, ocupan un número finito de pasos discretos, a pesar de que la conversión analógica-digital del microcontrolador que éste proyecto posee es de 10 bits de resolución.

5.5.2 ANÁLISIS DE RESULTADOS Y ERRORES DE MEDICIÓN CON RESPECTO AL OSCILOSCOPIO COMERCIAL

En general se pudo apreciar que a frecuencias de hasta 10 KHz, la forma de onda de las señales de entrada es bastante aceptable comparadas con las de un osciloscopio comercial; pero, como explica el teorema de Nyquist, a frecuencias mayores a 10 kHz, las señales se deterioran mucho, debido a que se tiene solo 8 o menos muestras por período.

5.5.2.1 Errores en la medición de frecuencia

En cuanto a la medición de frecuencia, vamos a analizar 2 casos, la toma de datos en modo en TIEMPO REAL y la toma de datos en modo de ALMACENAMIENTO.

a) Errores en TIEMPO REAL

Se toma como ejemplo el caso planteado en el Literal 5.4.2, donde se mide una forma de onda triangular de 2.5V pico-pico y frecuencia de 1kHz.

El osciloscopio digital **TDS 210** mide **1.008 kHz** de frecuencia.

El **osciloscopio para PC** da 1.01545 ms de período, es decir, **984.79 Hz** de frecuencia.

Sabiendo que se considera a f_{TDS} el **valor real** y a f_{PC} el **valor medido**.

Entonces:

El **Error absoluto** es:

$$EA_f = |f_{PC} - f_{TDS}|; \quad EA_f = |(984.79 - 1008)[Hz]; \quad \boxed{EA_f = 23.21[Hz]}$$

El **Error relativo** es:

$$ER_f = \frac{EA_f}{f_{TDS}} = \frac{23.21}{1008}; \quad \boxed{ER_f = 0.02303}$$

El **Error porcentual** es:

$$EP_f = ER_f \times 100 = 0.02303 \times 100; \quad \boxed{EP_f = 2.3\%}$$

b) Errores en modo de ALMACENAMIENTO

Se toma como ejemplo el caso planteado en el Literal 5.4.8, donde se mide una forma de onda cuadrada de 25V pico-pico y frecuencia de 10kHz.

El osciloscopio digital **TDS 210** mide **10.025 kHz** de frecuencia.

El **osciloscopio para PC** da 101.55 μ s de período, es decir, **9.847 kHz** de frecuencia.

Entonces:

El **Error absoluto** es:

$$EA_f = |f_{PC} - f_{TDS}|; \quad EA_f = |(9.847 - 10.025)[kHz]; \quad \boxed{EA_f = 178[Hz]}$$

El **Error relativo** es:

$$ER_f = \frac{EA_f}{f_{TDS}} = \frac{0.178}{10.025}; \quad \boxed{ER_f = 0.01775}$$

El **Error porcentual** es:

$$EP_f = ER_f \times 100 = 0.01775 \times 100; \quad \boxed{EP_f = 1.77\%}$$

c) Errores promedio totales en la medición de frecuencia

Después de haber presentado un ejemplo del cálculo de los errores en la medición de frecuencia para cada modo de toma de datos (en los 2 Literales anteriores), en la Tabla 5.2 se presenta un resumen de los resultados del cálculo de los errores de todas las pruebas realizadas y el cálculo del error promedio total en la medición de frecuencia.

| PRUEBA # | Literal | Medición de frecuencia del osciloscopio TDS 210 | Medición de período del osciloscopio a prueba | Medición de frecuencia del osciloscopio a prueba | Error absoluto | Error relativo | Error porcentual |
|--|---------|---|---|--|----------------|----------------|------------------|
| | | kHz | ms | kHz | kHz | | % |
| 1 | 5.4.1 | 0,06 | 17,21854 | 0,05808 | 0,00192 | 0,03205 | 3,20511 |
| 2 | 5.4.2 | 1,008 | 1,01545 | 0,98479 | 0,02321 | 0,02303 | 2,30307 |
| 3 | 5.4.3 | 5,021 | 0,20088 | 4,97802 | 0,04298 | 0,00856 | 0,85596 |
| 4 | 5.4.4 | 10,001 | 0,10155 | 9,84785 | 0,15315 | 0,01531 | 1,53134 |
| 5 | 5.4.5 | 20,032 | 0,05055 | 19,78166 | 0,25034 | 0,01250 | 1,24971 |
| 6 | 5.4.6 | 30,018 | 0,03377 | 29,60787 | 0,41013 | 0,01366 | 1,36627 |
| 7 | 5.4.7 | 40,01 | 0,02539 | 39,39133 | 0,61867 | 0,01546 | 1,54630 |
| 8 | 5.4.8 | 1,008 | 1,00442 | 0,99560 | 0,01240 | 0,01230 | 1,22972 |
| | | 10,025 | 0,10155 | 9,84785 | 0,17715 | 0,01767 | 1,76708 |
| 9 | 5.4.9 | 0,06 | 17,21854 | 0,05808 | 0,00192 | 0,03205 | 3,20511 |
| | | 5,028 | 0,20309 | 4,92391 | 0,10409 | 0,02070 | 2,07014 |
| 10 | 5.4.10 | 20,032 | 0,05331 | 18,75778 | 1,27422 | 0,06361 | 6,36090 |
| | | 40,01 | 0,02561 | 39,05173 | 0,95827 | 0,02395 | 2,39508 |
| Error promedio porcentual total de frecuencia | | | | | | | 2,24 |

Tabla 5.2 Errores en la medición de frecuencia.

Los errores de medición de la frecuencia se puede deducir que se deben a factores como:

- El número de muestras por ciclo de señal: El número de muestras por cada ciclo nunca será suficiente para reproducir una señal análoga.

Consecuentemente, al tratar de graficar una señal que se reconstruye de una señal digital que ya tiene un error, el resultado acarrea el error.

- Se puede notar que los errores de medición de frecuencia se incrementan cuando la frecuencia es más alta. Esto debido a que mientras más alta es la frecuencia a graficarse, menor es el número de muestras posibles para mostrarse en pantalla.
- Resolución horizontal del monitor: Las medidas del período de las señales, Se hacen con una resolución horizontal de aproximadamente 453 píxeles, lo cual limita la exactitud de la medición.

5.5.2.2 Errores en la medición de voltaje

En cuanto a la medición de voltaje se analiza 3 casos, uno para cada circuito acondicionador que se tienen en cada canal. Para escala mínima de -2.5V a 2.5V, escala media de -25V a 25V, y, escala máxima de -250V a 250V.

a) Errores en el circuito acondicionador de escala mínima

Se toma como ejemplo el caso del Literal 5.4.2, donde se mide una forma de onda triangular de 2.5V pico-pico y frecuencia de 1kHz.

El osciloscopio digital **TDS 210** mide **2.5Vpp**.

El **osciloscopio para PC** mide **2.471Vpp**.

Sabiendo que se considera a V_{TDS} el **valor real** y a V_{PC} el **valor medido**.

Entonces:

El **Error absoluto** es:

$$EA_V = |V_{PC} - V_{TDS}|; \quad EA_V = |(2.471 - 2.5)|[V]; \quad \boxed{EA_V = 0.029[V]}$$

El **Error relativo** es:

$$ER_V = \frac{EA_V}{V_{TDS}} = \frac{0.029}{2.5}; \quad \boxed{ER_V = 0.0116}$$

El **Error porcentual** es:

$$EP_V = ER_V \times 100 = 0.0116 \times 100;$$

$$EP_V = 1.16\%$$

b) Errores en el circuito acondicionador de escala media

Se toma como ejemplo el caso planteado en el Literal 5.4.8, donde se mide una forma de onda cuadrada de 25V pico-pico y frecuencia de 10kHz.

El osciloscopio digital **TDS 210** mide **25Vpp**.

El **osciloscopio para PC** mide **24.713Vpp**.

Entonces:

El **Error absoluto** es:

$$EA_V = |V_{PC} - V_{TDS}|; \quad EA_V = |(24.713 - 25)|[V];$$

$$EA_V = 0.287[V]$$

El **Error relativo** es:

$$ER_V = \frac{EA_V}{V_{TDS}} = \frac{0.287}{25};$$

$$ER_V = 0.0112$$

El **Error porcentual** es:

$$EP_V = ER_V \times 100 = 0.0112 \times 100;$$

$$EP_V = 1.12\%$$

c) Errores en el circuito acondicionador de escala máxima

Se toma como ejemplo el caso planteado en el Literal 5.4.7, donde se mide una forma de onda senoidal de 25V pico-pico y frecuencia de 40kHz.

El osciloscopio digital **TDS 210** mide **25Vpp**.

El **osciloscopio para PC** mide **27.298Vpp**.

Entonces:

El **Error absoluto** es:

$$EA_V = |V_{PC} - V_{TDS}|; \quad EA_V = |(27.298 - 25)|[V]; \quad \boxed{EA_V = 2.298[V]}$$

El **Error relativo** es:

$$ER_V = \frac{EA_V}{V_{TDS}} = \frac{2.298}{25}; \quad \boxed{ER_V = 0.0919}$$

El **Error porcentual** es:

$$EP_V = ER_V \times 100 = 0.0919 \times 100; \quad \boxed{EP_V = 9.19\%}$$

d) Errores promedio totales en la medición de voltaje

Después de haber calculado los errores de voltaje para cada uno de los 3 circuitos de acondicionamiento (que posee cada canal), en la Tabla 5.3 se presenta un resumen de los errores de todas las pruebas realizadas y el cálculo del error promedio total en la medición de voltaje.

| PRUEBA # | Literal | Medición de voltaje pico-pico del osciloscopio TDS 210 | Medición de voltaje pico-pico del osciloscopio a prueba | Error absoluto | Error relativo | Error porcentual |
|---|---------|--|---|----------------|----------------|------------------|
| | | V | V | V | | % |
| 1 | 5.4.1 | 2,5 | 2,643 | 0,143 | 0,057 | 5,720 |
| 2 | 5.4.2 | 2,5 | 2,471 | 0,029 | 0,012 | 1,160 |
| 3 | 5.4.3 | 25 | 22,988 | 2,012 | 0,080 | 8,048 |
| 4 | 5.4.4 | 25 | 21,839 | 3,161 | 0,126 | 12,644 |
| 5 | 5.4.5 | 25 | 27,298 | 2,298 | 0,092 | 9,192 |
| 6 | 5.4.6 | 25 | 28,735 | 3,735 | 0,149 | 14,940 |
| 7 | 5.4.7 | 25 | 27,298 | 2,298 | 0,092 | 9,192 |
| 8 | 5.4.8 | 25 | 25,860 | 0,860 | 0,034 | 3,440 |
| | | 25 | 24,713 | 0,287 | 0,011 | 1,150 |
| 9 | 5.4.9 | 25 | 25,574 | 0,574 | 0,023 | 2,296 |
| | | 25 | 25,287 | 0,287 | 0,011 | 1,148 |
| 10 | 5.4.10 | 25 | 26,149 | 1,149 | 0,046 | 4,596 |
| | | 25 | 24,425 | 0,575 | 0,023 | 2,300 |
| Error promedio porcentual total de voltaje | | | | | | 5,83 |

Tabla 5.3 Errores en la medición de voltaje.

Los errores de medición de voltaje se pueden deber a:

- Resolución vertical digital: La resolución del conversor A/D es de 10 bits, es decir que solo hay 1024 posibilidades discretas de definir un valor de voltaje en todo el rango vertical.
- Resolución vertical del monitor: La resolución vertical del monitor en este caso es de aproximadamente 348 píxeles, lo cual limita la exactitud de la medición en sentido vertical.

5.6 COSTOS DEL PROYECTO

5.6.1 COSTO DE LA LICENCIA DEL SOFTWARE VISUAL BASIC 6.0

Se tiene que el costo de la licencia del programa Microsoft Visual Basic 6.0 Professional es de \$695.91¹. Además, cada computador se estima debe costar en promedio \$600, pero este es un elemento que se asume ya posee el usuario, así que no se considera para el análisis.

5.6.2 COSTO DE DISEÑO, DESARROLLO Y CONSTRUCCIÓN

En cuanto al firmware y el software, considerando el tiempo Invertido (1 año y medio) y herramientas utilizadas para el diseño y desarrollo de los mismos, y considerando también el tiempo invertido (1 año) para el diseño y construcción de todo el hardware, el costo por diseño, desarrollo y construcción del primer prototipo del presente proyecto, incluyendo Hardware y Software se estima en **\$2000**.

5.6.3 COSTOS DE LOS ELEMENTOS DEL HARDWARE

Los costos listados a continuación incluyen todos los elementos, activos y pasivos que conforman el módulo, y demás elementos externos del dispositivo. Los costos se dividieron en 4 partes: Chasis y fuente, tarjeta de control y comunicación, tarjetas de los canales; y, software.

¹ Se puede comprobar en: <http://www.jigantic.com/product.aspx?item=50535&refID=206&sku=S54300>

5.6.3.1 Costos de chasis y fuente

| Etapa | Descripción | Cantidad | Precio unitario | Precio total |
|----------------------------------|----------------------------------|----------|-----------------|--------------|
| | | | \$ | \$ |
| Fuente | Fuente para PC de 250W, 6A | 1 | 13,00 | 13,00 |
| Interruptor para carcaza | Switch balancin PND | 1 | 0,55 | 0,55 |
| Carcaza | Carcaza | 1 | 5,60 | 5,60 |
| Conectores BNC | Jack BNC para chasis | 3 | 1,20 | 3,60 |
| cableado, espadines y conectores | Cableado, espadines y conectores | 1 | 7,00 | 7,00 |
| Chasis y Fuente, TOTAL | | | | 29,75 |

Tabla 5.4 Detalle de los costos de los elementos para chasis y fuente.

5.6.3.2 Costos de la tarjeta de comunicación y control

| Etapa | Descripción | Cantidad | Precio unitario | Precio total |
|---|--|----------|-----------------|--------------|
| | | | \$ | \$ |
| Placa | Tarjeta | 1 | 15,00 | 15,00 |
| Control | Pulsante | 1 | 0,20 | 0,20 |
| | Resistencia de 1 kohm, 1/4W | 1 | 0,02 | 0,02 |
| | Resistencia de 5.6 kohm, 1/4W | 1 | 0,02 | 0,02 |
| | Cristal de 20 MHz | 1 | 0,50 | 0,50 |
| | Capacitor electrolítico de 1uF, 50V | 1 | 0,20 | 0,20 |
| | Capacitor cerámico de 22pF | 2 | 0,05 | 0,10 |
| | Zócalo 40 pines | 1 | 0,50 | 0,50 |
| | dsPIC30F3011 | 1 | 20,00 | 20,00 |
| Comunicación | Jumper | 1 | 0,25 | 0,25 |
| | Resistencia 2.2 kohm, 1/4W | 3 | 0,02 | 0,06 |
| | Diodo led 5mm | 3 | 0,10 | 0,30 |
| | Zócalo 40 pines | 1 | 0,50 | 0,50 |
| | Módulo integrado de transferencia de datos USBMOD4 | 1 | 50,00 | 50,00 |
| Tarjeta de comunicación y control, TOTAL | | | | 87,65 |

Tabla 5.5 Detalle de los costos referentes a la tarjeta de comunicación y control.

5.6.3.3 Costos de las tarjetas de los canales

| Etapa | Descripción | Cantidad | Precio unitario | Precio total |
|---|--|----------|-----------------|--------------|
| | | | \$ | \$ |
| Placa | Tarjeta | 3 | 29,00 | 87,00 |
| Entrada | Portafusible BAK. | 3 | 0,18 | 0,54 |
| | Fusible 2A, 250V | 3 | 0,04 | 0,12 |
| | Varistor 180V, 10A | 3 | 0,24 | 0,72 |
| | Resistencia 10kohm, 5W | 6 | 0,20 | 1,20 |
| | Potenciómetro multivuelta de precisión de 100 kohm | 3 | 1,68 | 5,04 |
| | Capacitor de poliester de 0.1 pF, 250V | 3 | 0,15 | 0,45 |
| Circuitos de selección de acondicionamiento de entrada y de acoplamientos | Rele de 5Vdc, 0.5A, 120Vac | 18 | 1,60 | 28,80 |
| | Diodo rectificador 1N007 | 18 | 0,05 | 0,90 |
| | Transistor BJT, NPN 123AP | 18 | 0,07 | 1,26 |
| | Resistencia de 100 ohm, 1/4W | 18 | 0,02 | 0,36 |
| | Decodificador integrado 2/4, 74HC139N | 6 | 1,00 | 6,00 |
| | Inversor digital integrado 74HC240N | 6 | 0,57 | 3,42 |
| | Llave lógica 74HC4066AP | 6 | 0,55 | 3,30 |
| Acondicionamiento de señal de voltaje | Amplificadores operacionales MAX477 | 15 | 3,80 | 57,00 |
| | Resistencia de 10 kohm, 1/4W | 12 | 0,02 | 0,24 |
| | Resistencia de 150 ohm, 1/4W | 3 | 0,02 | 0,06 |
| | Resistencia de 3.3 kohm, 1/4W | 3 | 0,02 | 0,06 |
| | Resistencia de 2.7 kohm, 1/2W | 6 | 0,02 | 0,12 |
| | Capacitor cerámico 0.01uF, 16V | 6 | 0,06 | 0,36 |
| | Capacitor de 1.5pF, 100V | 3 | 0,15 | 0,45 |
| | Potenciómetro multivuelta de precisión de 10 kohm | 6 | 1,68 | 10,08 |
| | Potenciómetro multivuelta de precisión de 100 kohm | 12 | 1,68 | 20,16 |
| | Potenciómetro multivuelta de precisión de 1 Mohm | 3 | 1,68 | 5,04 |
| | Zócalo 8 pines | 15 | 0,09 | 1,35 |
| | Zócalo 14 pines | 6 | 0,10 | 0,60 |
| | Zócalo 16 pines | 6 | 0,10 | 0,60 |
| Zócalo de 20 pines | 6 | 0,10 | 0,60 | |

| | |
|---------------------------------------|---------------|
| Tarjetas de los canales, TOTAL | 235,83 |
|---------------------------------------|---------------|

Tabla 5.6 Detalle de los costos referentes a las tarjetas de los 3 canales.

Siendo este costo total de Hardware y Software con precios actualizados a Septiembre del año 2007.

5.6.4 ANÁLISIS DEL COSTO TOTAL DEL PROYECTO

Considerando todo lo expuesto en los Literales 5.6.1, 5.6.2, y, 5.6.3, se tiene que el costo total del primer prototipo del osciloscopio para PC diseñado, es de **\$3049.14**, como lo muestra el resumen de la Tabla 5.7.

| RUBRO | | COSTO |
|---|-----------------------------------|----------------|
| Licencia Visual Basic 6.0 | | 695,91 |
| Diseño, desarrollo y construcción (Software y Hardware) | | 2000,00 |
| Elementos de construcción del módulo | Chasis y fuente | 29,75 |
| | Tarjeta de comunicación y control | 87,65 |
| | Tarjetas de los canales | 235,83 |
| Costo total del primer prototipo | | 3049,14 |

Tabla 5.7 Costo total del primer prototipo.

Pero, una vez adquirida la licencia de Visual Basic 6.0, restando el costo de diseño y desarrollo, en la producción de más prototipos, solo queda la etapa de construcción, trabajo que se estima en **\$50**. A este rubro se debe añadir el estimado del software de la interfaz del osciloscopio. Para calcular este costo, se tomó como referencia el costo de productos similares y se consideró que se trata de un software de aplicación muy específica. Sumando estas consideraciones se estimó para el **VISUALSCOPE 1.0.1** un costo de **\$50**. También se debe prorratear y añadir el valor de la licencia para el número de máquinas en las que se instalará el software de control. Bajo el supuesto de que se instalará en 3 máquinas, por cada máquina se tiene entonces un costo de licencia de **\$231.97**. Considerando todo esto, el costo calculado por máquina del prototipo es de **\$685.20** como lo muestra la Tabla 5.8.

| RUBRO | | COSTO |
|--|-----------------------------------|---------------|
| Licencia Visual Basic 6.0 (por cada máquina) | | 231,97 |
| Software VISUALSCOPE 1.0.1 | | 50,00 |
| Construcción del módulo | | 50,00 |
| Elementos de construcción del módulo | Chasis y fuente | 29,75 |
| | Tarjeta de comunicación y control | 87,65 |
| | Tarjetas de los canales | 235,83 |
| Costo total del primer prototipo | | 685,20 |

Tabla 5.8 Costo por módulo para 3 máquinas.

Finalmente, si consideramos que para otras máquinas fuera de este caso, solo necesitamos el software **VISUALSCOPE 1.0.1**, que es la aplicación ya diseñada, entonces no se considera el costo de la licencia del Visual Basic 6.0, así como muestra la Tabla 5.9.

| RUBRO | | COSTO |
|--|-----------------------------------|---------------|
| Software VISUALSCOPE 1.0.1 | | 50,00 |
| Construcción del módulo | | 50,00 |
| Elementos de construcción del módulo | Chasis y fuente | 29,75 |
| | Tarjeta de comunicación y control | 87,65 |
| | Tarjetas de los canales | 235,83 |
| Costo total definitivo del osciloscopio para PC | | 453,23 |

Tabla 5.9 Costo mínimo por módulo.

5.7 COMPARACIÓN CON OTROS EQUIPOS SIMILARES DISPONIBLES EN EL MERCADO

En general en el mercado se pueden encontrar osciloscopios analógicos o digitales con mejores características en velocidad y ancho de banda, las cuales no son estrictamente necesarias para aplicaciones industriales, donde se trabaja con 60 y 120 Hz.

Lo que se quiere hacer con el presente proyecto es tener un dispositivo de medida que brinde las características más básicas y necesarias para un laboratorio de instrumentación.

Las características en amplitud de la señal de entrada de los osciloscopios comerciales, son bastante similares al del presente proyecto.

En cuanto a número de canales, este módulo posee 3 canales lo cual es una característica favorable y muy necesaria, son muy pocos los osciloscopios comerciales que tienen más de 2 canales. Y, la resolución vertical del presente proyecto es de 10 bits, frente a los 8 bits que presentan casi todos los osciloscopios digitales y osciloscopios para PC que existen en el mercado.

Si se compara con osciloscopios analógicos y digitales, la diferencia de precio es bastante notable, como por ejemplo:

- Osciloscopio digital **PHILIPS PM3323** (Figura 5.21), 10 bits de resolución vertical, **€850 (\$1252)**.

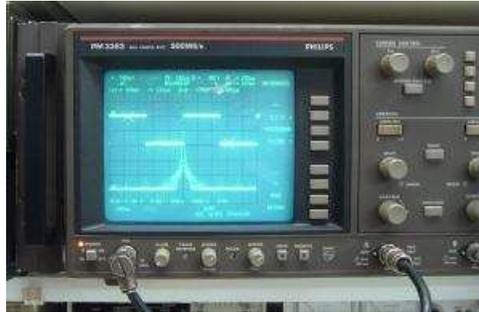


Figura 5.21 Osciloscopio PHILIPS PM3323.

- Osciloscopio digital **GDS-810S** (Figura 5.22), 8 bits de resolución vertical, **\$1250**.



Figura 5.22 Osciloscopio GDS-810S.

Se toma en este caso 2 ejemplos de osciloscopios para PC con sus características más importantes, los precios incluyen el software del equipo.

- Osciloscopio digital de almacenamiento para PC, **ETC M621/E** (Figura 5.23), de 8 bits de resolución vertical y voltaje máximo de entrada de +- 100 V, **\$799**.



Figura 5.23 Osciloscopio para PC, ETC M621/E.

- Osciloscopio digital de almacenamiento para PC, **ETC M522** (Figura 5.24), de 8 bits de resolución vertical y voltaje máximo de entrada de +- 200V, **\$760**.



Figura 5.24 Osciloscopio para PC, ETC M522.

En estos ejemplos se ve que el más barato en el mercado cuesta **\$760**, entonces en costos, la implementación en 3 o más máquinas del presente osciloscopio es económicamente una excelente opción porque se trata de **\$685.20** o menos. Mucho más beneficioso resulta al no considerar el costo de la licencia del Visual Basic 6.0 ya que su costo final llegaría hasta **\$453.23** como se ve en la Tabla 5.9.

Por todo lo expuesto anteriormente el osciloscopio que aquí se ha construido es una buena alternativa económica frente a sus rivales o que no tienen el ancho de banda de los comerciales. Ancho de banda que de todas maneras es excesivo para aplicaciones industriales.

CAPÍTULO 6

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES Y RECOMENDACIONES

Después de las pruebas realizadas, a continuación se extraen las conclusiones de los resultados obtenidos. Más adelante se listan ciertas recomendaciones aprovechando las experiencias obtenidas durante la realización de este proyecto.

6.1 CONCLUSIONES

La resolución de conversión del módulo conversor A/D del dsPIC es de 10 bits por lo que cada lectura del A/D se dividió en 2 partes de 5 bits. Esto se hizo por varias razones, principalmente limitaciones en el hardware. El puerto USB de la PC tiene un buffer de 8 bits de ancho, igual que en el dsPIC y el USBMOD4, que gestiona la comunicación paralela con el dsPIC con solo con 8 bits. Por las razones expuestas, al tener que enviarse 2 tramas por cada dato, se reduce prácticamente la velocidad efectiva de la comunicación a la mitad.

Un objetivo del proyecto fue la visualización en “tiempo real” de señales de voltaje en una HMI desplegada en la pantalla del computador. No solo que se consiguió esto sino, mejor aún, mostrar 3 canales frente a los típicos dos canales que muestran los osciloscopios digitales tradicionales.

Se probó hacer un programa que solo envíe datos hacia la PC en lenguaje C y otro con la misma función en lenguaje Assembler. El código desarrollado resultó casi cuatro veces más denso que en lenguaje C. De esto se puede concluir que la programación de un microcontrolador o en este caso específico, de un dsPIC, resulta más simple para el programador si se la realiza en código de medio o alto nivel. Por esta razón se seleccionó programarlo en lenguaje C.

Sin embargo, la utilización del lenguaje C, introdujo líneas de código muchas veces innecesarias en el programa, generando retardos críticos en el resultado final y haciendo que el sistema trabaje más lentamente. En conclusión, si se

hubiese contado con la herramienta de software para poderlo programar en lenguaje Assembler, se lo hubiese preferido para ganar en velocidad.

El trigger por software implementado presenta todavía inconvenientes, a veces se frena y otras veces funciona perfectamente. Las razones pueden ser: Insuficientes datos enviados a la PC para completar una pantalla, desbordamiento del buffer de recepción del puerto USB y no se lo está limpiando a tiempo, o la orden de inicio de trazado de cada pantalla no está llegando en forma regular a la PC. Este es un problema pendiente del cual lamentablemente no se pudo determinar su causa, pero es un fenómeno que no ocurre con frecuencia y el resultado global es satisfactorio.

El autoescalamiento implementado en este osciloscopio resultó ser una herramienta necesaria. Por facilidad, este algoritmo se lo implementó en el programa de la interfaz directamente. Se probó también implementarlo en el Firmware del dsPIC, pero se comprobó que el resultado es el mismo, debido a que el proceso de autoescalamiento no requiere o no depende de la velocidad de lectura de los datos.

La idea de diseñar la HMI con la plataforma de desarrollo LAB-VIEW, quizás debería probarse y evaluar velocidad de programación vs. velocidad de la aplicación. Pero si bien no se usó un software especializado para diseño de interfaces gráficas de instrumentación, se puede decir que VISUAL BASIC 6.0 permitió la creación de una interfaz gráfica intuitiva y bastante satisfactoria.

Varias personas que utilizaron el osciloscopio, manifestaron que su interfaz gráfica simula bastante bien al HMI de un osciloscopio tradicional y, por tanto, el manejo del mismo es bastante sencillo e intuitivo.

En cuanto a las pruebas, los resultados de las mismas indican que a las frecuencias altas, como se tiene solo un poco más de 2 muestras por período, como en el caso de una señal de 40 kHz, la señal no se recupera bien debido a que el teorema de Nyquist se aplica muy cerca al límite. Para señales de

frecuencia igual o menor a 10 kHz, se obtiene formas de onda bastante buenas, debido a que se tiene 8 o más muestras por período.

Si bien las especificaciones completas del proyecto no pudieron ser alcanzadas en esta primera instancia, los resultados obtenidos son muy satisfactorios y el prototipo puede ser fácilmente extendido para cumplir con los objetivos propuestos inicialmente.

Por último, la etapa más difícil del proyecto fue sin duda el comienzo, pues se carecía de un rumbo definido a seguir y existía gran dificultad para concretarlo, por miedo a estar perdiendo el tiempo o de llegar a un callejón sin salida. Pero, ya en etapas posteriores, con las ideas claras y los objetivos inmediatos definidos resultó más interesante y ameno el trabajo. Es más, sería una buena idea seguir trabajando en este proyecto, para llegar a solucionar los defectos y limitaciones que tiene este prototipo.

6.2 RECOMENDACIONES

Podría resultar una ventaja utilizar una placa universal en vez de un circuito impreso ya que el hacer un cambio de último momento en cualquier conexión solo implicaría soldar o desoldar un elemento o un cable. Esto en una placa impresa podría ser imposible de llevar a cabo, a menos que se utilicen cables para reemplazar la pista de cobre, lo cual en cierto momento se debió hacer, pero afortunadamente no resultó un cambio crítico.

Para la alimentación del osciloscopio, siempre será recomendable que no se conecte directamente a la red (120Vac), sino que se conecte a través de un transformador de aislamiento (relación de transformación unitaria), para evitar cualquier daño por inconsistencia de tierras.

La utilización de un lenguaje de bajo nivel es bastante necesaria, en el caso del diseño de un equipo de medición y prueba de instrumentación, debido a que lo que se necesita en estos casos es optimizar al máximo la velocidad de tratamiento de los datos, y sobre todo en la parte de adquisición y

acondicionamiento de los datos. En este caso lamentablemente no se pudo disponer de la herramienta necesaria para este fin, por lo que se recomienda que la Institución haga un esfuerzo en conseguirla.

Existe un detalle difícil de descubrir al principio con el MPLAB C30 cuando se realizan ciertas operaciones binarias de movimiento de bits en un registro (como SWAP o Rotation left no carry). La peculiaridad consiste en que al aplicar dichas operaciones a un registro, en el resultado algunos bits se pierden sin razón, o dicho registro se encera. Después se descubrió que antes de aplicar dichas operaciones a ese registro, el mismo se debe igualar a otro, y entonces si se puede aplicar dichas operaciones sin problema.

En un osciloscopio se toman datos para obtener una gráfica, en magnitud y tiempo, de cualquier tipo de señal eléctrica en el tiempo. Esto hace esencial que se escoja una herramienta que ayude en dicha tarea. VISUAL-BASIC demostró que puede ser considerado para este fin.

El presente proyecto podría ser modificado de muchas maneras, incluyendo el rediseño del Firmware del dsPIC, y agregando otros elementos al hardware, como convertidores AD externos, memorias RAM externas para almacenamiento de los datos, circuitos de trigger por hardware, incluyendo aislamientos en la fuente de alimentación, e incluso usando amplificadores operacionales aislados (estos últimos no fueron utilizados debido a su enorme costo comparado con los usados). Aunque se encarecería el costo del equipo, las ventajas que se podrían lograr, bien podrían compensar el costo.

Una de las herramientas que se utilizaron para la realización de este proyecto fue el acceso a Internet que hoy en día es imprescindible, no solo para un proyecto de este tipo puesto que la red cuenta con incontables recursos e información, mucho más que cualquier biblioteca local.

BIBLIOGRAFÍA

Libros:

- **LOPEZ MONTENEGRO Héctor**; Utilización del computador como un osciloscopio de dos canales. Escuela Politécnica Nacional. Facultad de Ingeniería Eléctrica, Quito. 1999.
- **TANDASO CANDO, Juan Carlos**; Obtención de oscilogramas bajo formato NTSC. Escuela Politécnica Nacional. Facultad de Ingeniería Eléctrica, Quito. 1999.

Páginas WEB:

- <http://www.recursovisualbasic.com.ar>
- <http://www.telecable.es>
- <http://www.vbasic.astalaweb.com>
- <http://www.todopic.com.ar>
- <http://www.manolo.auna.com>
- <http://www.microchip.com>
- <http://www.ate.uniovi.es>
- <http://www.moetronix.com/dspic/index.htm>
- <http://www.micropic.es>

- http://www.precma.it/ELEXOL_USBMOD4.htm
- <http://www.ftdichip.com/Drivers/VCP.htm>
- <http://www.ee.washington.edu/conselec/CE/sp95reports/guterman/final/scope1.htm>
- http://www.labc.usb.ve/mgimenez/Lab_Circ_Electronicos_Guia_Teorica/Cap8.pdf
- <http://eureka.ya.com/elektron/osciloscopioCASERO.htm>
- <http://www.depi.itchihuahua.edu.mx/lfcorral/previo.pdf>
- <http://www.ucm.es/info/electron/laboratorio/instrumentos/osc/osc.html>
- <http://www.nbb.cornell.edu/neurobio/land/PROJECTS/VideoScope/>
- <http://www.todoelectronica.com/Revistas/REVISTAS.htm>
- <http://kudelsko.free.fr/oscilloscope/sommaire.htm>
- <http://www.terra.es/tecnologia/articulo/html/tec4782.htm>
- <http://www.maxim-ic.com/parts.cfm?p=MAX477>
- <http://es.wikipedia.org/wiki/Aliasing>
- <http://cnx.org/content/m12971/latest/>
- http://es.wikipedia.org/wiki/Teorema_de_muestreo_de_Nyquist-Shannon

ANEXOS