



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**  
**FACULTAD DE INGENIERÍA**

**SISTEMA ELECTRÓNICO PARA EL MONITOREO DE  
FENÓMENOS SÍSMICOS  
UTILIZANDO LOS ACCELERÓMETROS TRI-AXIALES DSA1 Y  
DCA333**

**TESIS**

Que para obtener el título de  
**Ingeniero Eléctrico-Electrónico**

**P R E S E N T A**

Martínez Altamirano Jesús Manuel

**DIRECTOR DE TESIS**

M. I. Lauro Santiago Cruz



Ciudad Universitaria, Cd. Mx., 2016

---

## Agradecimientos

A Dios, porque mi vida ha carecido de desgracias, enfermedades y desamor, algunos de los ingredientes que justificarían no seguir combatiendo.

A mis padres Raymundo y Teresa, porque la educación germina en casa; ni instruyéndome en la Universidad puedo encontrar cosas más valiosas de las que estas humildes personas me han dotado. Por tener la facultad de otorgar profesionistas a la sociedad con recursos apenas suficientes, por el esfuerzo descomunal que realizan día a día, por sus inagotables palabras de aliento, por la vida.

A mis hermanos Mallely y Gabriel porque el apoyo se siente firme y constante, porque no tengo que pedirlo cuando ya me han ayudado, porque tenemos los mismos principios, porque tenemos la misma sangre.

A mis amigos y compañeros, sobre todo a aquellos que me han orientado en momentos tortuosos: a Omar, Ricardo, Colín, Diana, Adrián, Alejandro y Alicia, porque me han hablado con franqueza, reconociendo mis virtudes pero también señalando mis deficiencias para hacerme crecer como persona. Agradezco especialmente a Arcos, porque no importa cuán molesto pueda yo llegar a ser, nunca se rinde en su labor de ayudarme.

A todos mis maestros, 'mis padres académicos', porque me han colmado de conocimientos para aplicar en la tarea que me atañe como Ingeniero; a mi asesor de tesis el M.I. Lauro Santiago, por guiarme en esta larga travesía.

Al Instituto de Ingeniería por proporcionarme de recursos para la elaboración de este trabajo y, finalmente, a la Universidad Nacional Autónoma de México, mi alma máter, que me ha nutrido de conocimientos para responder a las necesidades que demande la sociedad.

---

# Contenido

Índice de figuras.....	IV
Índice de tablas.....	X
Prólogo.....	XI
<b>Capítulo 1. Marco teórico referencial.....</b>	<b>1</b>
1.1. La sismología en México.....	1
1.2. Ondas sísmicas y sus parámetros.....	3
1.3. Acelerógrafos y acelerogramas.....	11
<b>Capítulo 2. Generalidades del sistema.....</b>	<b>16</b>
2.1. Acelerómetro de Fuerza Balanceada.....	16
2.2. Acondicionamiento de señales.....	20
2.2.2. Filtros electrónicos.....	24
2.2.3. El amplificador de instrumentación.....	38
2.3. Conversión analógico-digital.....	39
2.3.1. Parámetros generales de un convertidor analógico-digital.....	41
2.3.2. Tipos de convertidor analógico-digital.....	44
2.3.3. Aliasing.....	49
2.3.4. Convertidor analógico digital sigma-delta.....	52
2.4. Protocolos de comunicación serial para dispositivos electrónicos.....	62
2.4.1. Comunicación serial asíncrona mediante UART.....	63
2.4.2. Protocolo SPI.....	69
2.4.3. Protocolo I2C.....	74
2.5. Memorias de estado sólido.....	80
2.5.1. Memoria RAM.....	82
2.5.2. Memoria flash.....	86
2.5.3. Tarjeta de memoria microSD.....	87
2.5.4. Comunicación con la memoria microSD.....	90
2.5.5. Sistema de archivos FAT.....	96
2.6. Sistema de Posicionamiento Global y especificación NMEA 0183.....	99
2.7. Microcontroladores.....	107
2.7.1. Arquitectura y funcionamiento de un microcontrolador.....	114
2.7.2. Conjunto de instrucciones.....	121
<b>Capítulo 3. Diseño del sistema.....</b>	<b>124</b>
3.1. Antecedentes del sistema electrónico para el monitoreo de fenómenos sísmicos.....	124
3.2. Hardware del sistema.....	128
3.2.1. Diseño de la etapa de acondicionamiento de la señal de entrada.....	128

3.2.2. Convertidor analógico-digital.....	139
3.2.3. Microcontrolador.....	147
3.2.4. Memoria principal.....	153
3.2.5. Memoria de pre-evento.....	154
3.2.6. Receptor GPS.....	158
3.2.7. Reloj de Tiempo Real.....	160
3.2.8. <i>Joystick</i> y <i>display</i> gráfico.....	164
3.3. Software del sistema.....	169
3.3.1. Estructura general del programa.....	170
3.3.2. Configuración de la frecuencia de muestreo.....	174
3.3.3. Digitalización y procesamiento de datos.....	177
3.3.4. Manejo de la memoria de pre-evento.....	183
3.3.5. Almacenamiento de datos en la memoria principal.....	186
3.3.6. Transferencia de datos entre la memoria de pre-evento y la memoria principal.....	193
3.3.7. Adquisición de fecha y hora.....	200
3.3.8. Manejo del joystick y navegación a través de menús.....	203
<b>Capítulo 4. Integración del sistema.....</b>	<b>205</b>
4.1. Fase 1.....	205
4.1.1. Pruebas al circuito de acondicionamiento de señal.....	206
4.1.2. Reconstrucción de la señal muestreada.....	217
4.2. Fase 2.....	219
4.2.1. Diseño de la tarjeta de circuito impreso.....	220
4.2.2. Integración del hardware.....	234
4.2.3. Pruebas al circuito de acondicionamiento de señal.....	239
4.2.4. Reconstrucción de la señal muestreada.....	249
<b>Capítulo 5. Resultados y conclusiones.....</b>	<b>252</b>
5.1. Resultados.....	252
5.2. Conclusiones.....	255
5.3. Recomendaciones.....	255
<b>Apéndice A: Glosario de términos.....</b>	<b>257</b>
<b>Referencia de figuras.....</b>	<b>259</b>
<b>Bibliografía.....</b>	<b>264</b>

---

# Índice de figuras

## Capítulo 1

Figura 1.1. Cinturón de fuego.....	1
Figura 1.2. Placas tectónicas que conforman la República Mexicana.....	2
Figura 1.3. Zonas sísmicas en México.....	3
Figura 1.4. Fenómenos relacionados con ondas elásticas.....	4
Figura 1.5. Onda P.....	5
Figura 1.6. Onda S.....	6
Figura 1.7. Onda Rayleigh.....	6
Figura 1.8. Onda Love.....	7
Figura 1.9. Sismograma.....	7
Figura 1.10. Sismógrafo.....	8
Figura 1.11. Sismógrafo moderno.....	9
Figura 1.12. Determinación del epicentro.....	11
Figura 1.13. Acelerograma del terremoto de 1985 en México.....	14
Figura 1.14. Aceleración, velocidad y desplazamiento de la tierra durante el terremoto de 1985.....	15

## Capítulo 2

Figura 2.1. Esquema de un FBA.....	17
Figura 2.2. Variación del factor de amplificación dinámico.....	19
Figura 2.3. Señales unipolares.....	22
Figura 2.4. Señales bipolares.....	22
Figura 2.5. Medición del voltaje en $Z_m$ .....	23
Figura 2.6. Medición de la corriente en $Z_m$ .....	23
Figura 2.7. Acoplamiento de impedancias.....	24
Figura 2.8. Filtro pasivo pasa bajos de primer orden.....	26
Figura 2.9. Respuesta en frecuencia del filtro pasa bajos.....	27
Figura 2.10. Filtro pasivo pasa altos de primer orden.....	28
Figura 2.11. Respuesta en frecuencia del filtro pasa altos.....	29
Figura 2.12. Filtro pasa banda.....	29
Figura 2.13. Respuesta en frecuencia del filtro pasa banda.....	30
Figura 2.14. Filtro pasivo pasa bajos de segundo orden.....	31
Figura 2.15. Filtro activo pasa altos de primer orden.....	32
Figura 2.16. Respuesta del filtro pasa bajos.....	33
Figura 2.17. Respuesta del filtro pasa altos.....	34
Figura 2.18. Respuesta del filtro pasa banda.....	34
Figura 2.19. Circuito genérico para la síntesis de filtros activos de segundo orden.....	35
Figura 2.20. Topología de realimentación múltiple y ganancia infinita o de <i>Rauch</i> .....	35

Figura 2.21. Filtro pasa bajos con la topología de <i>Rauch</i> .....	36
Figura 2.22. Topología del circuito general VCVS y Sallen-Key.....	37
Figura 2.23. Filtro pasa bajos con topología de Sallen-Key.....	37
Figura 2.24. Amplificador de instrumentación.....	38
Figura 2.25. Diagrama de bloques del proceso de conversión analógico-digital.....	40
Figura 2.26. Error de cuantificación en un convertidor analógico-digital ideal.....	41
Figura 2.27. Circuito <i>sample &amp; hold</i> .....	42
Figura 2.28. Error de offset.....	43
Figura 2.29. Error de monotonicidad.....	43
Figura 2.30. Error de linealidad.....	44
Figura 2.31. ADC de rampa.....	44
Figura 2.32. Convertidor de aproximaciones sucesivas.....	45
Figura 2.33. Curva de salida del ADC.....	46
Figura 2.34. Convertidor tipo flash.....	46
Figura 2.35. Convertidor de doble rampa.....	47
Figura 2.36. Forma de onda del convertidor de doble rampa.....	48
Figura 2.37. Convertidor voltaje-frecuencia.....	49
Figura 2.38. Muestreo de una señal analógica.....	50
Figura 2.39. Espectros de la señal analógica, el tren de impulsos y la señal muestreada.....	51
Figura 2.40. Espectros de la señal muestreada.....	51
Figura 2.41. Modulador Delta.....	53
Figura 2.42. Señales correspondientes a la modulación delta.....	54
Figura 2.43. Convertidor sigma-delta.....	55
Figura 2.44. Señales correspondientes al ADC sigma-delta.....	56
Figura 2.45. Modulador y demodulador delta.....	57
Figura 2.46. Reubicación del bloque integrador.....	57
Figura 2.47. Diagrama de bloques del modulador sigma-delta.....	58
Figura 2.48. Modulador sigma-delta en el dominio de Laplace.....	58
Figura 2.49. Modulador sigma-delta de tercer orden.....	59
Figura 2.50. Comportamiento de distintos moduladores sigma-delta.....	60
Figura 2.51. Conceptualización de la etapa de diezmado.....	61
Figura 2.52. Elección de la tasa de diezmado.....	62
Figura 2.53. Módulo UART.....	64
Figura 2.54. Generador de la tasa de transferencia de datos.....	65
Figura 2.55. Formato de la cadena de datos.....	67
Figura 2.56. Conector DB9 y DB25.....	67
Figura 2.57. Conexión de módem nulo.....	69
Figura 2.58. Bus SPI.....	70
Figura 2.59. Interconexión entre maestro y esclavo.....	71
Figura 2.60. Señales eléctricas en el bus SPI.....	72
Figura 2.61. Configuración de CPOL y CPHA.....	73
Figura 2.62. Bus I2C e interconexión de dispositivos.....	75
Figura 2.63. Condición de dato válido.....	75
Figura 2.64. Condiciones de inicio, parada e inicio repetido.....	76
Figura 2.65. Formato de los datos transmitidos en el bus I2C.....	77
Figura 2.66. Secuencia de datos al configurar al esclavo en modo escritura.....	78

Figura 2.67. Secuencia de datos al configurar al esclavo en modo lectura.....	78
Figura 2.68. Módulo I2C.....	79
Figura 2.69. Celda de memoria DRAM.....	83
Figura 2.70. Celda de memoria SRAM.....	83
Figura 2.71. Memoria RAM de 4x4 celdas y 6 MOSFETs por celda.....	85
Figura 2.72. Estructura de una memoria SRAM.....	86
Figura 2.73. Memoria microSD.....	89
Figura 2.74. Formato de comando válido en el modo SPI.....	91
Figura 2.75. Respuesta R1.....	92
Figura 2.76. Respuesta R2.....	92
Figura 2.77. Respuesta R3.....	93
Figura 2.78. Respuesta R7.....	93
Figura 2.79. Inicialización de la tarjeta de memoria SD.....	94
Figura 2.80. Lectura de un bloque de datos.....	95
Figura 2.81. Lectura de múltiples bloques de datos.....	95
Figura 2.82. Escritura de un bloque de datos.....	96
Figura 2.83. Escritura de múltiples bloques de datos.....	96
Figura 2.84. Sistema de archivos FAT.....	97
Figura 2.85. Red NAVSTAR.....	100
Figura 2.86. Estructura de la señal GPS.....	101
Figura 2.87. Estaciones del segmento de control.....	102
Figura 2.88. Intersección de tres esferas.....	103
Figura 2.89. Obtención de la distancia.....	105
Figura 2.90. Diagrama de bloques del microcontrolador MSP430.....	110
Figura 2.91. Arquitectura von Neumann.....	114
Figura 2.92. Arquitectura Harvard.....	115
Figura 2.93. Esquema general de una CPU.....	115

## Capítulo 3

Figura 3.1. Sistema DIGI.....	125
Figura 3.2. Diagrama de bloques del nuevo sistema.....	128
Figura 3.3. Adición de voltaje de directa al geófono.....	130
Figura 3.4. Filtro paso bajos.....	131
Figura 3.5. Circuito limitador de voltaje.....	131
Figura 3.6. Acondicionamiento de sensores pasivos.....	131
Figura 3.7. Circuito para acondicionar la salida de un sensor activo.....	132
Figura 3.8. Combinación de ambos circuitos.....	134
Figura 3.9. Amplificador de instrumentación.....	135
Figura 3.10. Configuración propuesta para ser utilizada con el ADC.....	136
Figura 3.11. Respuesta en frecuencia al variar R6.....	138
Figura 3.12. Circuito de acondicionamiento de señal.....	139
Figura 3.13. Programación de la tasa de sobre muestreo.....	140
Figura 3.14. Distribución de terminales del ADC.....	142
Figura 3.15. Ciclo de operación del ADC.....	144

Figura 3.16. Señales manejadas durante la transferencia de datos.....	144
Figura 3.17. Conexión entre convertidores y microcontrolador.....	147
Figura 3.18. Distribución de terminales del microcontrolador.....	149
Figura 3.19. Diagrama de bloques del microcontrolador.....	150
Figura 3.20. Conexión entre microcontrolador y periféricos.....	152
Figura 3.21. Memoria microSD.....	153
Figura 3.22. Interfaz entre microcontrolador y memoria.....	154
Figura 3.23. Distribución de terminales de la memoria RAM.....	155
Figura 3.24. Registro de modo de operación.....	157
Figura 3.25. Conexión entre memoria RAM y microcontrolador.....	158
Figura 3.26. Distribución de terminales del receptor GPS.....	160
Figura 3.27. Conexión entre receptor GPS y microcontrolador.....	160
Figura 3.28. Distribución de terminales del RTC.....	162
Figura 3.29. Mapa de memoria del RTC.....	163
Figura 3.30. Conexión entre RTC y microcontrolador.....	164
Figura 3.31. <i>Joystick</i> .....	165
Figura 3.32. Comportamiento ideal de un interruptor.....	165
Figura 3.33. Efecto de rebote.....	165
Figura 3.34. <i>Deboucer</i> por software.....	166
Figura 3.35. Conexión entre joystick y microcontrolador.....	166
Figura 3.36. <i>Display</i> .....	167
Figura 3.37. Conexión entre <i>display</i> y microcontrolador.....	169
Figura 3.38. Entorno de desarrollo de Arduino.....	170
Figura 3.39. Estructura general del programa.....	172
Figura 3.40. Registros del TCNT3.....	175
Figura 3.41. Especificaciones del modo de operación del contador.....	175
Figura 3.42. Especificaciones de la fuente de reloj para el contador.....	176
Figura 3.43. Opciones del ADC.....	178
Figura 3.44. Operación del ADC.....	179
Figura 3.45. Formato del dato convertido.....	180
Figura 3.46. Registro de modo de operación de la memoria RAM.....	184
Figura 3.47. Conjunto de instrucciones de la memoria RAM.....	186
Figura 3.48. Diagrama de flujo de la rutina de adquisición de datos.....	190
Figura 3.49. Organización de la memoria RAM.....	194
Figura 3.50. Almacenamiento de datos en la RAM. Tiempo de prueba = 20 [s].....	195
Figura 3.51. Almacenamiento de datos en la RAM. Tiempo de prueba = 50 [s].....	196
Figura 3.52. Almacenamiento de datos en la RAM. Tiempo de prueba = 90 [s].....	197
Figura 3.53. Diagrama de flujo de la rutina de transferencia de datos.....	199
Figura 3.54. Diagrama de flujo de la rutina de adquisición de la fecha.....	202
Figura 3.55. Menú principal.....	203
Figura 3.56. Diagrama de flujo del manejo del <i>joystick</i> .....	204

## Capítulo 4

Figura 4.1. Prototipo del sistema RAS-II.....	206
Figura 4.2. Circuito de acondicionamiento de señal para sensores activos.....	207

Figura 4.3. Sistema bajo prueba.....	209
Figura 4.4. Señales $vs1$ y $vs2$ .....	209
Figura 4.5. Voltajes $V_A$ , $V_B$ y $V_o$ .....	210
Figura 4.6. Filtro anti-aliasing.....	212
Figura 4.7. Señal $vs1$ y señal $V1$ .....	213
Figura 4.8. Respuesta a 2 [Hz].....	213
Figura 4.9. Respuesta a 4 [Hz].....	214
Figura 4.10. Respuesta a 8 [Hz].....	214
Figura 4.11. Respuesta a 12.4 [Hz].....	214
Figura 4.12. Respuesta a 16 [Hz].....	215
Figura 4.13. Respuesta a 20 [Hz].....	215
Figura 4.14. Respuesta a 40 [Hz].....	215
Figura 4.15. Respuesta a 124 [Hz].....	216
Figura 4.16. Señal original.....	217
Figura 4.17. Señal reconstruida.....	218
Figura 4.18. Plano de tierra de la capa superior ( <i>top layer</i> ).....	221
Figura 4.19. Plano de tierra de la capa inferior ( <i>bottom layer</i> ).....	221
Figura 4.20. Entrada de alimentación de la PCB.....	223
Figura 4.21. Capacitor de desacoplo de la memoria de pre-evento.....	223
Figura 4.22. División de la PCB en zonas de acuerdo a la frecuencia manejada.....	223
Figura 4.23. <i>Room "CH1"</i> .....	224
Figura 4.24. <i>Room "CH2"</i> .....	224
Figura 4.25. <i>Room "CH3"</i> .....	225
Figura 4.26. <i>Room "Digital"</i> .....	225
Figura 4.27. <i>Room "Supply"</i> .....	225
Figura 4.28. Marcas para identificar la posición correcta de los C.I.....	227
Figura 4.29. Ubicación del cristal.....	227
Figura 4.30. Ruteo de los componentes con señales críticas.....	229
Figura 4.31. Vista 3D de la figura 4.30.....	230
Figura 4.32. Cálculo de la anchura de las pistas.....	232
Figura 4.33. PCB del sistema RAS-II.....	234
Figura 4.34. PCB del sistema con todos los elementos integrados.....	235
Figura 4.35. Programador <i>AVRISP mkII</i> .....	235
Figura 4.36. Programación del microcontrolador.....	236
Figura 4.37. Iniciando programación del microcontrolador.....	236
Figura 4.38. Ventana <i>Device Programming</i> .....	237
Figura 4.39. Ventana emergente.....	237
Figura 4.40. Configuración de los fusibles.....	238
Figura 4.41. Programación de la memoria de programa.....	239
Figura 4.42. Circuito de acondicionamiento de señal para sensores activos.....	240
Figura 4.43. Sistema bajo prueba.....	241
Figura 4.44. Señal senoidal de 1 [V] pico y 3 [Hz].....	241
Figura 4.45. Oscilograma de las señales bajo prueba.....	242
Figura 4.46. Filtro anti-aliasing.....	243
Figura 4.47. Señal de entrada y señales $V1$ y $V2$ .....	245
Figura 4.48. Respuesta a 1 [Hz].....	245
Figura 4.49. Respuesta a 2 [Hz].....	245

---

Figura 4.50. Respuesta a 4 [Hz].....	246
Figura 4.51. Respuesta a 8 [Hz].....	246
Figura 4.52. Respuesta a 12.3 [Hz].....	246
Figura 4.53. Respuesta a 16 [Hz].....	247
Figura 4.54. Respuesta a 20 [Hz].....	247
Figura 4.55. Respuesta a 40 [Hz].....	247
Figura 4.56. Respuesta a 123 [Hz].....	248
Figura 4.57. Señal a digitalizar.....	249
Figura 4.58. Señal reconstruida.....	250

---

# Índice de tablas

## Capítulo 2

Tabla 2.1. Conector de 9 terminales.....	68
Tabla 2.2. Conector de 25 terminales.....	68
Tabla 2.3. Modos de operación para un microcontrolador ATMEGA328.....	73
Tabla 2.4. Características de las versiones de memoria SD.....	88
Tabla 2.5. Microcontroladores más comunes en el mercado.....	113

## Capítulo 3

Tabla 3.1. Características solicitadas para el nuevo equipo.....	127
Tabla 3.2. Frecuencias de corte al variar R6.....	137
Tabla 3.3. Valores de los MSB asociados al voltaje de entrada.....	145
Tabla 3.4. Terminales de la memoria microSD.....	153
Tabla 3.5. Conjunto de instrucciones de la memoria RAM.....	156
Tabla 3.6. Configuración del modo de operación.....	157
Tabla 3.7. Terminales del <i>display</i> .....	168
Tabla 3.8. Formato del mensaje RMC.....	200

## Capítulo 4

Tabla 4.1. Resultados del barrido de frecuencias, prototipo.....	216
Tabla 4.2. Resultados del barrido de frecuencias.....	248

---

# Prólogo

El territorio mexicano se ubica geográficamente en una de las regiones sísmicamente más activas del mundo, situado dentro del área conocida como el Cinturón de Fuego del Pacífico. Chiapas, Guerrero, Oaxaca y otros estados de la costa oeste son aquellos que presentan mayor actividad sísmica; la causa evidente, la interacción de las placas oceánicas de Cocos y Rivera que subducen con las placas Norteamericana y del Caribe sobre la costa del Pacífico.

Los terremotos son fenómenos naturales que propician una serie de perjuicios en las poblaciones de grupos humanos. Entre los costos materiales se encuentran la destrucción de viviendas, de líneas viales, carreteras y puentes; no obstante, otros costos inmateriales, como la pérdida de vidas humanas y lesiones causadas por la catástrofe, son resultado irreparable que un terremoto deja a su paso.

Aunque las zonas epicentrales se localizan en diversos puntos de la costa del Pacífico, es en las ciudades donde los efectos de un sismo son más notorios. La Ciudad de México es vulnerable ante estos fenómenos pues los sedimentos lacustres de la ciudad son altamente compresibles, lo que ha dado lugar a intrincados problemas de cimentación para la construcción de edificios elevados y de gran peso.

Distintas instituciones a nivel nacional participan en la investigación de fenómenos sísmicos con el objetivo de determinar las causas y los procesos que originan los terremotos, entender estos fenómenos y proponer medidas preventivas en caso de que se presenten. En el proceso de estas actividades y como pilar de inicio, se presenta la labor de medir el movimiento relativo de la Tierra durante el transcurso de un terremoto.

Por lo tanto, es necesario disponer de instrumentos apropiados para obtener información acerca de estos fenómenos naturales y de sistemas electrónicos embebidos que faciliten esta tarea. La electrónica ha ido ganando terreno en casi todas las actividades que desarrollamos y la instrumentación sísmica no es la excepción; actualmente modernos sismómetros manejados con electrónica sofisticada son utilizados para recabar los datos necesarios antes, durante y después de un terremoto.

Para instituciones cuyo objetivo es proporcionar la información necesaria para evaluar y prevenir el riesgo sísmico, es necesario contar con los equipos pertinentes que permitan el monitoreo, adquisición y almacenamiento de datos de forma autónoma; una de esas instituciones es la Coordinación de Sismología e Instrumentación Sísmica del Instituto de Ingeniería, que trabaja en la investigación de estos fenómenos naturales. Actualmente, dicha Coordinación cuenta con sensores sísmicos capaces de medir el movimiento de la Tierra y se pretende diseñar una interfaz que les permita digitalizar las señales y registrarlas en un medio de almacenamiento de datos.

Cabe mencionar que en México existe un reducido número de edificios instrumentados, a pesar de que el país es uno de los de mayor riesgo sísmico en el mundo. Entonces se presenta también la problemática de ubicar el punto medio entre el costo y la precisión que los instrumentos de medición sean capaces de ofrecer.

La instrumentación para registrar aceleraciones sísmicas en estructuras tiene costos elevados de instalación, adquisición, procesamiento y análisis de la información. Mediante el desarrollo de esta tesis se pretende obtener un sistema electrónico capaz de realizar acciones de adquisición, almacenamiento y monitoreo de señales sísmicas de forma autónoma, proporcionando también costos reducidos de manufactura.

El sistema a desarrollar contará con todos los elementos necesarios para su operación; conjuntará la funcionalidad de los circuitos electrónicos analógicos conformados por componentes discretos e integrados y, al mismo tiempo, utilizará módulos y circuitos de funcionamiento digital para dar robustez al mismo.

El diseño del sistema tendrá la calidad de ser un proyecto integral, puesto que para su desarrollo se hará uso de los fundamentos de electrónica a nivel de componentes discretos; en contraparte, y como apoyo a la gestión de las tareas internas, se utilizará un microcontrolador, empleando los fundamentos de la electrónica digital y del funcionamiento de la misma en componentes de alta escala de integración.

El sistema electrónico tendrá como función principal la adquisición y registro de las señales provenientes de los acelerómetros tri-axiales DSA1 y DCA333, utilizados para la obtención de la señal eléctrica. La señal proporcionada por los acelerómetros será procesada por mecanismos electrónicos analógicos de modo que se encuentre en un rango de tensión apropiado para el sistema. El paso siguiente será la digitalización de esta señal utilizando un convertidor analógico-digital de alto rendimiento en cuanto a precisión; el hecho de digitalizar las señales provenientes de los sensores implica la versatilidad para manejar la información y almacenarla en dispositivos que cumplen con este fin.

Dado que es imprescindible contar con una referencia precisa acerca de la fecha y hora en la que se presente un sismo, el instrumento contará con un receptor GPS para recabar esta información y proporcionarla en el momento que se le requiera. El instrumento también contará con un *display* gráfico y un *joystick* para la re-configuración y/o el despliegue de los parámetros que indicarán el estado del dispositivo, entre ellos: estado de la batería del dispositivo, fecha y hora, comprobación de las señales y el estado de la memoria de almacenamiento de datos.

Finalmente, el sistema contará con un módulo para el almacenamiento de datos, ofreciendo al usuario la facilidad de cambiar la memoria o extraer la existente para el vaciado de datos en su computadora personal. Adicionalmente, el sistema electrónico será totalmente portable, configurable en campo y no necesitará para su configuración algún periférico adicional.

Con base en lo mencionado anteriormente, la estructura de este trabajo se dividirá en cinco capítulos y una sección correspondiente a un apéndice y bibliografía. El desarrollo del trabajo se realiza de la siguiente manera:

**Capítulo 1.** En este capítulo se definen los conceptos básicos necesarios en el estudio de la sismología, se describe la actividad sísmica en nuestro país, se estudian algunos parámetros de las ondas sísmicas y se da una revisión de las características de los registradores sísmicos.

**Capítulo 2.** Se presentan los temas básicos y fundamentales que permiten comprender el proceso de desarrollo del sistema; asimismo, se abordan tópicos relacionados con los transductores utilizados para el registro de aceleraciones sísmicas, el acondicionamiento y digitalización de señales, la sincronización de tiempo con los datos, el almacenamiento de los datos en una microSD y la interfaz usuario – máquina.

**Capítulo 3.** Aquí se plantean las necesidades que debe cubrir el proyecto, presentando el diseño y desarrollándolo con base en ellas. El diseño cuenta de dos partes fundamentales, a saber: el hardware, integrado por todos los dispositivos y componentes interconectados, y el software, constituido por las funciones y rutinas programadas dentro del microcontrolador.

**Capítulo 4.** Se describen las pruebas realizadas al sistema, dividiéndose éstas en dos fases, la primera, correspondiente a las realizadas al prototipo, y la segunda, correspondiente a las realizadas al equipo final.

**Capítulo 5.** Se presentan los resultados obtenidos en el desarrollo de este trabajo, las conclusiones a las que se llegaron y algunas recomendaciones para mejorar el sistema desarrollado.

**Apéndice, referencia de figuras y bibliografía.** Al final del documento se incluye un apéndice asociado a un glosario de términos para facilitar la comprensión del trabajo, se incluye también una referencia de figuras para que el lector pueda remontarse a las ligas proporcionadas y tener acceso a éstas, finalmente, se incluye la bibliografía consultada para realizar este trabajo.

---

# Capítulo 1. Marco teórico referencial

En este capítulo se hará mención sobre los conocimientos básicos relacionados con el área de la sismología, para ello, se abordan temas como: la sismología en México, las ondas sísmicas presentes durante estos fenómenos y las características de los instrumentos empleados para su medición.

## 1.1 La sismología en México

México tiene un alto nivel de exposición al peligro sísmico por estar ubicado en el Cinturón (o Anillo) de Fuego del Pacífico (figura 1.1), considerado como una importante zona generadora de sismos. Los fenómenos sísmicos son impredecibles y, tras la ocurrencia de un terremoto, la principal causa de víctimas y daños en zonas urbanas se relaciona con el derrumbe parcial o total de las construcciones.

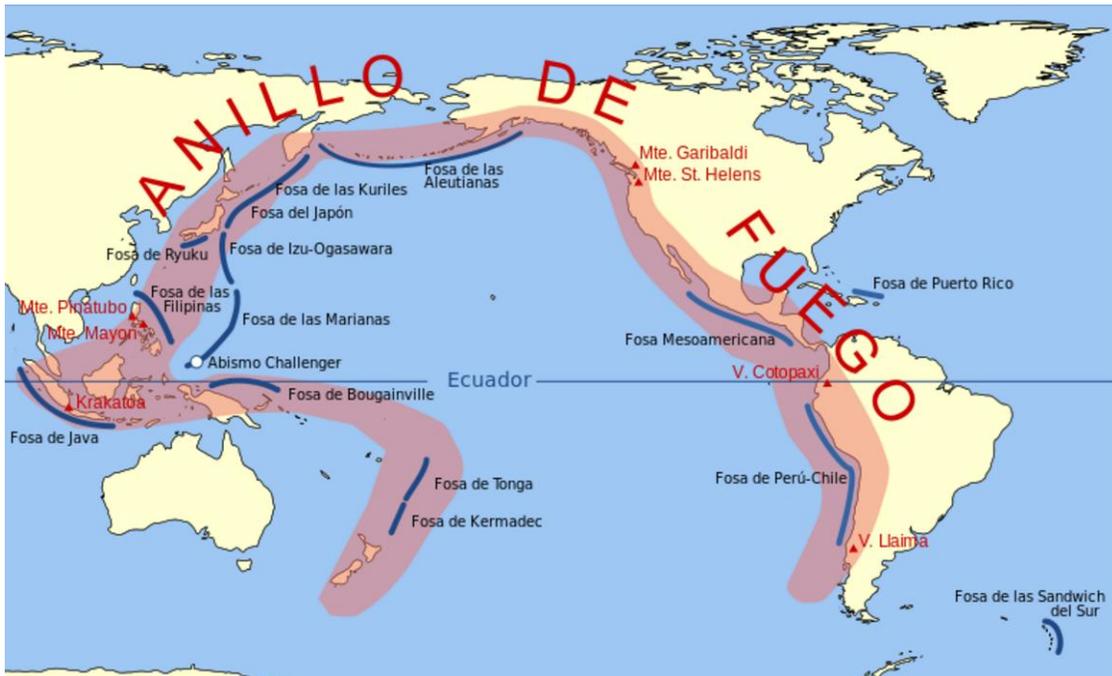


Figura 1.1. Cinturón de fuego.

Las zonas sísmicas de México se clasifican en función de la cantidad de sismos que se presentan. Debido a la intensa actividad entre las placas tectónicas ubicadas en el lado oeste del país, la zona de mayor actividad sísmica se encuentra en los estados de la costa del Pacífico. Gracias a los registros sísmicos de las redes de detección instaladas en territorio mexicano, se puede ver una intensa actividad sísmica que se concentra principalmente entre Chiapas, Oaxaca, Guerrero, Michoacán y Jalisco, así como en la parte norte de la península de Baja California, en la región de Mexicali.

La República Mexicana, geológicamente hablando, es el resultado de múltiples procesos tectónicos llevados a cabo durante su evolución. El territorio mexicano está situado sobre cinco placas litosféricas, en cuyos límites encontramos trincheras (zona de subducción), centros de expansión (zona de dorsal) y fallas transformantes. La mayor parte del territorio continental pertenece a la placa Norteamericana, mientras que la península de Baja California pertenece a la placa del Pacífico; en el litoral del Pacífico se tiene la micro placa de Rivera, la placa de Cocos, y la del Caribe como se observa en la figura 1.2.

Hasta hoy en día no hay ningún aparato o tecnología que nos permita predecir los sismos, ni el lugar, intensidad o duración, por lo que tenemos que estar preparados para prevenir mayores daños. El estudio de la actividad sísmica de México comenzó a principios del siglo, sin embargo, los antecedentes históricos de grandes sismos del país fueron registrados en un gran número de documentos. En 1910 se inauguró la red sismológica mexicana y desde esa fecha hasta nuestros días se ha mantenido una observación continua de los temblores cuyos registros se conservan en la Estación Sismológica de Tacubaya y en las instalaciones del Instituto de Geofísica de la UNAM, encargada de operar el Servicio Sismológico Nacional (SSN) y su red de 35 estaciones sismológicas. El SSN reporta en el país, en promedio, la ocurrencia de 4 sismos por día, de magnitud mayor a 3.0 en la escala de magnitud de momentos.



Figura 1.2. Placas tectónicas que conforman la República Mexicana.

Con fines de diseño antisísmico, la República Mexicana se dividió en cuatro zonas sísmicas, utilizándose los catálogos de sismos del país desde inicios de siglo. Las zonas sísmicas se aprecian en la figura 1.3.

- *Zona A.* Es una zona donde no se tienen registros históricos de sismos, no se han reportado sismos en los últimos 80 años y no se esperan aceleraciones del suelo mayores a un 10% de la aceleración de la gravedad a causa de temblores.
- *Zona B y C.* Son zonas intermedias, donde se registran sismos no tan frecuentemente, o son zonas afectadas por altas aceleraciones pero que no sobrepasan el 70% de la aceleración de la gravedad.
- *Zona D.* Es una zona donde se han reportado grandes sismos históricos, donde la ocurrencia de sismos es muy frecuente y las aceleraciones del suelo pueden sobrepasar el 70% de la aceleración de la gravedad.



Figura 1.3. Zonas sísmicas en México.

## 1.2. Ondas sísmicas y sus parámetros

Una onda sísmica es un tipo de onda elástica que se propaga a través de la corteza terrestre, que pueden ser generadas por movimientos telúricos naturales, por ejemplo, al chocar dos placas tectónicas, cuando surge un reajuste de las mismas o por actividad volcánica; también pueden ser generadas de manera artificial mediante la detonación de explosivos, con el uso de dispositivos electro-mecánicos que generan ondas sísmicas controladas o incluso por la actividad industrial.

En distintas ramas de la ingeniería se analizan fenómenos relacionados con la propagación de ondas elásticas, por ejemplo, en la Ingeniería Civil se utilizan modelos de escala reducida para estudiar el comportamiento de presas, puentes o puertos y perfeccionar los diseños antes de llevar a cabo la construcción real de la obra; en esta misma área, el análisis de la vibración ambiental (nombre genérico para describir una vibración persistente de la tierra debida a múltiples causas como el tránsito de automóviles, uso de maquinaria pesada, vientos y otros fenómenos) juega un papel importante en el diseño estructural de edificios. En el área de la Geotecnia se utilizan distintas técnicas, como la emisión acústica, para investigar las propiedades de las rocas por debajo del suelo y así diseñar las cimentaciones para diferentes estructuras civiles. En la Ingeniería Geológica por su parte, se utilizan técnicas como la prospección con fines de exploración para descubrir la existencia de yacimientos geológicos, petróleo, minerales, agua u otro elemento.

En cuanto a la Ingeniería Sismológica, se estudian distintos fenómenos pero no hay una definición exacta del rango de frecuencia de los sismos, sin embargo, a menudo se hablan de frecuencias no mayores a 100 [Hz] aunque las ondas de cuerpo (o internas) y las ondas superficiales que se propagan durante los sismos, poseen frecuencias por debajo de los 10 [Hz]; así mismo, se estudian otros fenómenos como los microsismos (sismos de muy poca intensidad que pueden ser ocasionados por modificaciones en el régimen fluvial de un río, desprendimiento de rocas en las laderas de una montaña, variaciones bruscas en la presión atmosférica debida a ciclones, entre otras) con frecuencias por debajo de 1 [Hz]. Se estudian también las oscilaciones libres de la Tierra que no son más que vibraciones naturales de toda la Tierra provocadas por grandes terremotos, las cuales, presentan frecuencias menores a  $10^{-2}$  [Hz]. Igualmente, las mareas terrestres son objeto de estudio, este fenómeno se ocasiona por la fuerza de atracción gravitatoria que la Luna y el Sol ejercen sobre la Tierra, presentando frecuencias por debajo de  $10^{-4}$  [Hz].

En la figura 1.4 se muestra el espectro en frecuencia de los fenómenos descritos anteriormente.

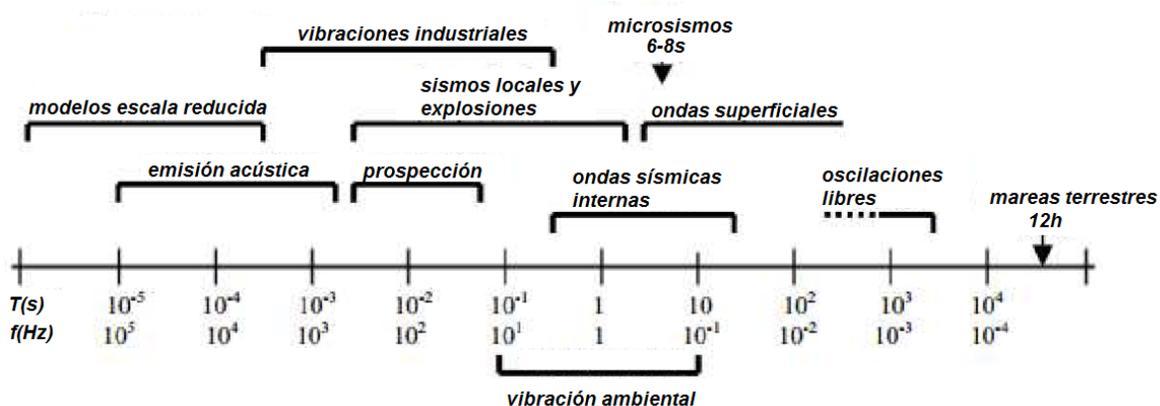


Figura 1.4 Fenómenos relacionados con ondas elásticas.

Las ondas sísmicas se dividen en dos grandes grupos: *ondas de cuerpo*, que se propagan por el interior del planeta, y *ondas superficiales*, que se generan por la interacción de las anteriores con la superficie de la Tierra, y se propagan por esta última.

### *Ondas de cuerpo*

Las ondas internas o de cuerpo se propagan en el interior de la Tierra. La velocidad de propagación de estas ondas es superior a la de las ondas superficiales. Las ondas de cuerpo se dividen a su vez en ondas *P* o *primarias* y ondas *S* o *secundarias*.

- *Ondas P*: Son ondas longitudinales o compresionales, lo cual significa que el suelo constantemente se dilata y expande alternadamente en la dirección de la propagación. Estas ondas generalmente viajan a una velocidad superior a la de las ondas S y pueden hacerlo a través de cualquier tipo de material. Las velocidades comunes de propagación de estas ondas son:  $330 \left[ \frac{m}{s} \right]$  en el aire,  $1450 \left[ \frac{m}{s} \right]$  en el agua y cerca de  $6000 \left[ \frac{m}{s} \right]$  en el granito, tipo de roca más abundante en la corteza continental (complementaria a la corteza oceánica).

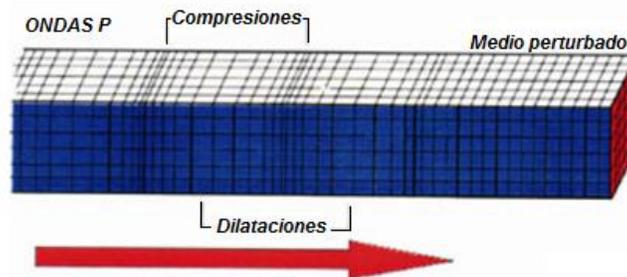


Figura 1.5. Onda P.

- *Ondas S*. Son ondas transversales o de corte, lo cual significa que el suelo es desplazado perpendicularmente a la dirección de propagación, alternadamente hacia un lado y hacia el otro. Las ondas S pueden viajar únicamente a través de sólidos debido a que los líquidos no pueden soportar esfuerzos de corte. Su velocidad es alrededor de 58% la de una onda P para cualquier material sólido. Usualmente la onda S tiene mayor amplitud que la P y se siente más fuerte que ésta; son las que producen la mayor parte de los daños durante los sismos.

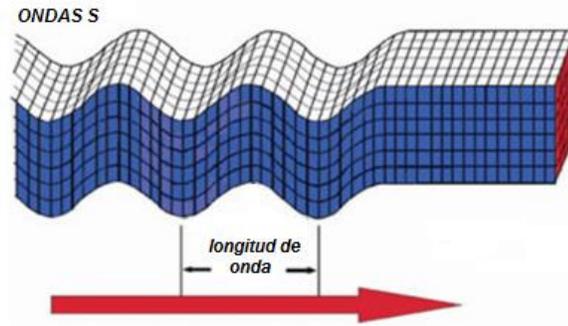


Figura 1.6. Onda S.

### Ondas superficiales

Las ondas superficiales reciben este nombre debido a que sólo se propagan por las capas superficiales de la Tierra, decreciendo su amplitud con la profundidad en la corteza terrestre. Dentro de este tipo de ondas se pueden diferenciar dos tipos de ondas: ondas *Rayleigh* y ondas *Love*.

- *Ondas Rayleigh*. La existencia de estas ondas fue predicha por John William Strutt, Lord Rayleigh, en 1885. Viajan como ondulaciones similares a aquellas encontradas en la superficie del agua, producen un movimiento parecido al de las olas en el mar, son más lentas que las ondas de cuerpo y su velocidad de propagación es casi un 90% de la velocidad de las ondas S.

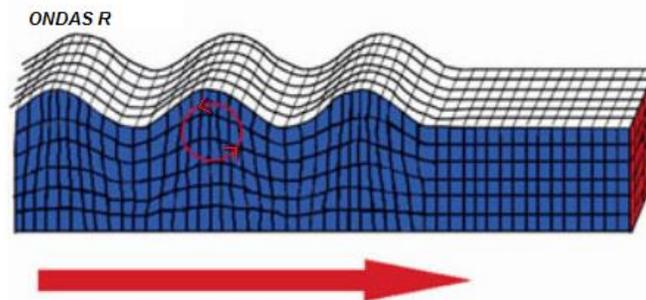


Figura 1.7. Onda Rayleigh.

- *Ondas Love*. Fueron bautizadas por Augustus Edward Hough Love, un matemático británico que creó un modelo de las ondas en 1911. Provocan cortes horizontales en la tierra, su velocidad es un 90% de la de las ondas S y es ligeramente superior a la velocidad de las ondas Rayleigh.

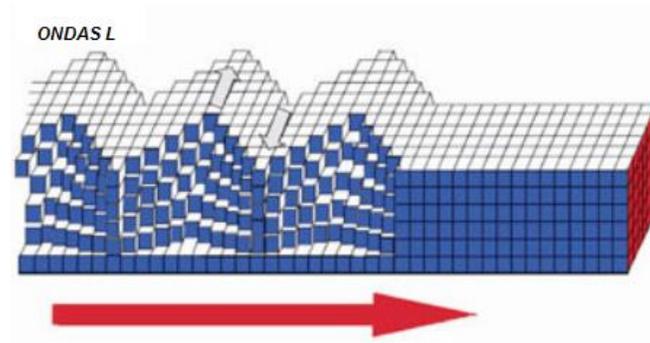


Figura 1.8. Onda Love.

La velocidad de propagación de las ondas sísmicas es distinta para cada una de ellas. Con ayuda de los sismogramas se pueden determinar ciertos parámetros que permiten cuantificar la energía de la Tierra liberada durante un sismo, siendo algunos de esos parámetros la amplitud máxima registrada por el sismógrafo y el retardo de tiempo entre la llegada de las ondas P y S.

En la figura 1.9 se muestra un ejemplo de un sismograma. En el extremo izquierdo se muestra la marca de tiempo asociada con la hora en que ocurrió el sismo. En la misma figura se muestra también el retardo de tiempo  $\Delta t$  entre el arribo de las ondas P y las ondas S, que para este caso es aproximadamente 4 [s], así mismo, se muestra la amplitud del desplazamiento máximo de la aguja del sismógrafo que en este ejemplo es aproximadamente 300 [ $\mu\text{m}$ ]. Vale la pena mencionar que la identificación entre las ondas P, S y superficiales no es una labor simple pues el tiempo de arribo entre unas y otras se superpone, por lo tanto, la interpretación de un sismograma sólo se puede realizar por parte de los especialistas en esta área.

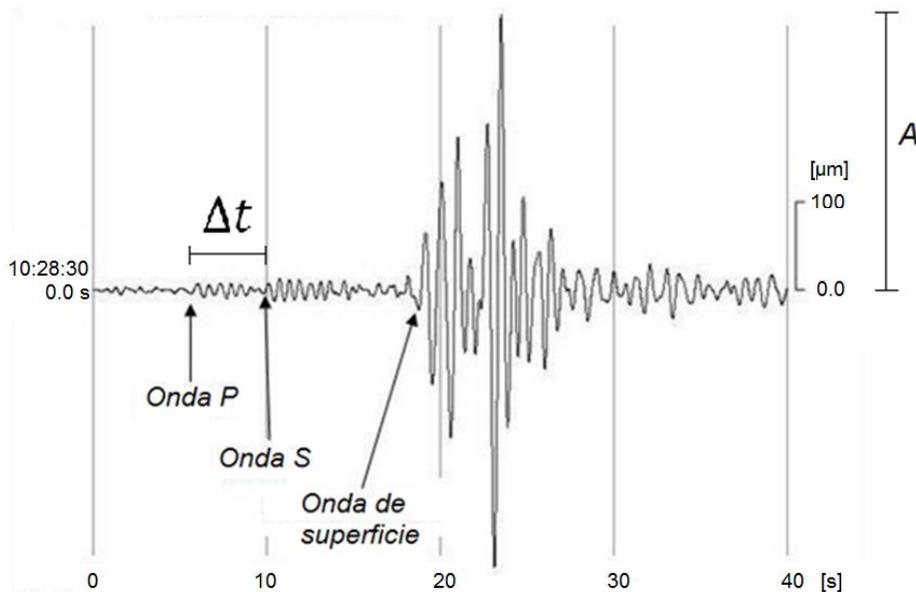


Figura 1.9. Sismograma.

En 1935 Charles Francis Richter con la colaboración de Beno Gutenberg, ambos investigadores del Instituto de Tecnología de California, desarrollaron una escala para estudiar terremotos cuyo registro se realizaba exclusivamente con un sismógrafo de Wood-Anderson, compuesto por un sistema masa-resorte-amortiguador, una aguja, un tambor de registro y un reloj. Un dibujo esquemático de este sismógrafo se muestra en la figura 1.10.

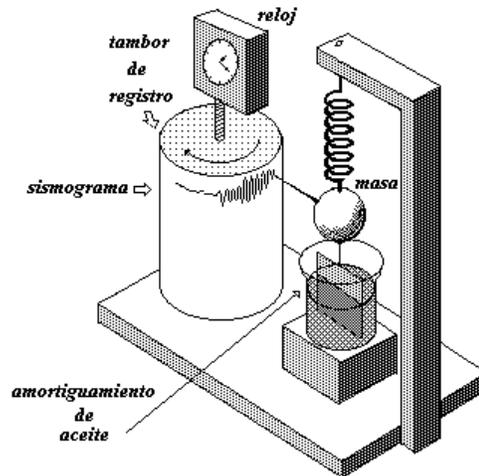


Figura 1.10. Sismógrafo.

Richter propuso una ecuación logarítmica para cuantificar la energía desprendida durante el terremoto:

$$M_L = \log_{10} A + 3 \log_{10}(8\Delta t) - 2.92 = \log_{10} \left( \frac{A \cdot \Delta t^3}{1.62} \right) \quad (1.1)$$

Donde:

$A$  = amplitud de las ondas en milímetros, tomada directamente del sismograma

$\Delta t$  = tiempo en segundos desde el inicio de las ondas P hasta el de las ondas S

$M_L$  = magnitud del terremoto

Richter arbitrariamente escogió un temblor de magnitud 0 para describir un terremoto que produciría un desplazamiento horizontal máximo de [1 $\mu$ m] en un sismograma trazado por un sismógrafo de torsión de Wood-Anderson localizado a 100 [km] de distancia del epicentro. La decisión de establecer el desplazamiento de 1[ $\mu$ m] tenía la intención de prevenir la asignación de magnitudes negativas, sin embargo, con el uso de sismógrafos modernos como el de la figura 1.11, se pueden obtener mediciones inferiores a 1 [ $\mu$ m], lo cual arroja resultados negativos al evaluarlos en esta escala.

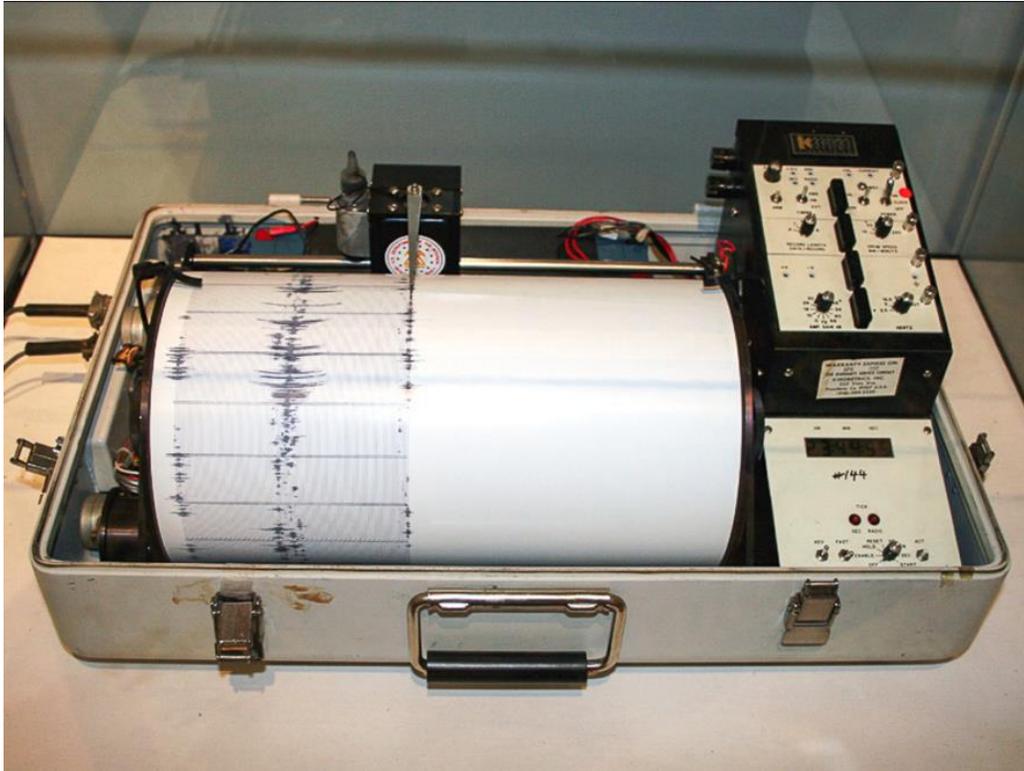


Figura 1.11. Sismógrafo moderno.

Vale la pena mencionar que la escala de Richter, también llamada escala de magnitud local ( $M_L$ ), no es la más usada actualmente pues presenta varios problemas; esta escala establece parámetros bastante específicos para poder utilizarla, por ejemplo: el epicentro debe localizarse a 100 [km], existen tablas y métodos para reformular el cálculo pero esto sólo es válido trabajando con distancias de hasta 600 [km], después de este valor los cálculos dejan de ser válidos; la escala únicamente puede determinar magnitudes hasta el valor de 6.8 pues dadas las limitaciones del sismógrafo de Wood-Anderson, llega un punto en el que la escala se satura y los sismos de magnitud superior a 6.8 aparentan ser muy similares. Otro inconveniente es que es difícil relacionar la escala con las características físicas del origen del terremoto.

La escala sismológica de magnitud de momento ( $M_W$ ) fue introducida en 1979 por Thomas C. Hanks y Hiroo Kanamori como la sucesora de la escala sismológica de Richter. La magnitud se basa en el momento sísmico del terremoto, que es igual a la rigidez de la Tierra, multiplicado por la cantidad media de deslizamiento en la falla y el tamaño del área que se deslizó. Algunas ventajas de esta escala son que no satura para valores altos y que coincide y continúa con los parámetros de la escala de Richter, por estas razones, la escala sismológica de magnitud de momento es la más usada para medir y comparar terremotos de grandes proporciones. A pesar de lo anterior, la escala de Richter es la que goza de mayor popularidad en la prensa y es común que ésta comunique la magnitud de un terremoto en la escala de Richter, cuando en realidad éste ha sido medido en la escala de magnitud de momento. En

algunos casos esto no constituye un error dada la coincidencia de parámetros entre ambas escalas.

Además de la magnitud de un sismo, analizando un sismograma también es posible localizar el epicentro de un terremoto al contar con una red de sismógrafos ubicados en distintas estaciones sísmicas sobre la Tierra. La técnica utilizada se basa en el principio de que toma un mayor tiempo el arribo de las ondas S que el de las ondas P hacia un punto de observación. Este atraso de tiempo es directamente proporcional a la distancia entre el punto de observación y el epicentro del terremoto. En términos de la velocidad de propagación de las ondas P y S, el retraso de tiempo puede expresarse como:

$$t_{SP} = \frac{d}{V_S} - \frac{d}{V_P} = \left( \frac{1}{V_S} - \frac{1}{V_P} \right) d \quad (1.2)$$

Donde:

$$\begin{aligned} V_p &= \text{velocidad promedio de las ondas P} \\ V_S &= \text{velocidad promedio de las ondas S} \\ d &= \text{distancia del epicentro al punto de observación} \end{aligned}$$

La distancia del punto de observación al epicentro puede obtenerse de:

$$d = \frac{t_{SP}}{\frac{1}{V_S} - \frac{1}{V_P}}$$

O aproximadamente:

$$d = \left( 8400 \frac{m}{s} \right) t_{SP} \quad (1.3)$$

La última expresión toma en cuenta que las velocidades de las ondas P y S son  $6 \left[ \frac{km}{s} \right]$  y  $3.5 \left[ \frac{km}{s} \right]$ , respectivamente. Tomando en cuenta el concepto de retardo entre éstas ondas, se puede obtener la distancia  $d$  y trazar una circunferencia cuyo radio es igual a esa distancia; al realizar el mismo proceso para dos estaciones sísmicas más y trazando igualmente circunferencias de radio igual a la distancia  $d$  calculada para cada estación, las circunferencias se intersectan en un punto y es precisamente ese punto, el lugar del epicentro. La figura 1.12 ilustra la obtención del epicentro.

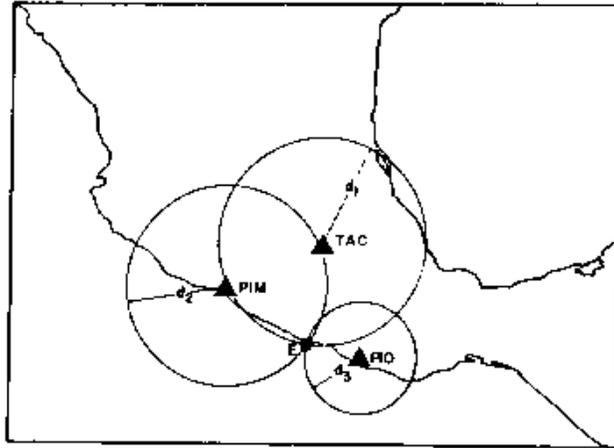


Figura 1.12. Determinación del epicentro.

### 1.3. Acelerógrafos y acelerogramas

A pesar de que los sismogramas obtenidos a través de los sismógrafos son muy útiles para estudiar algunas características de los terremotos y determinar sus epicentros y magnitudes, los sismógrafos son de uso limitado para aplicaciones de ingeniería. Hay varias razones para estas limitaciones. Primero, los sismógrafos no pueden grabar el movimiento generado por terremotos fuertes en regiones cercanas al epicentro, que son precisamente los movimientos que causan un alto daño y por lo tanto, son de interés en la ingeniería. Debido al gran factor de amplificación con el cual los sismógrafos magnifican el movimiento de la tierra, los terremotos fuertes desplazan la aguja del sismógrafo fuera del área del filme o papel de grabación.

La segunda razón tiene que ver con que los sismógrafos son invariablemente instalados en roca sólida, pues para estos instrumentos es de primordial interés medir movimientos de la corteza terrestre; por lo tanto, los sismógrafos no pueden proporcionar información acerca del suelo local. Finalmente, el diseño de los sismógrafos enfatiza la precisión en el tiempo de arribo de las ondas más que en la amplitud de éstas, sin embargo, en ingeniería estructural la amplitud del movimiento es de mayor interés que el tiempo en que las ondas arriban.

Para complementar el vacío que dejaron los sismógrafos, un tipo diferente de instrumento fue diseñado específicamente para registrar movimientos fuertes de la tierra y ser instalados en áreas urbanas u otras locaciones donde pudieran proporcionar información útil para aplicaciones de ingeniería. Este nuevo instrumento fue introducido en 1930 y es llamado acelerógrafo de movimiento fuerte.

#### *Acelerógrafos*

Los acelerógrafos están diseñados para medir directamente la variación de la aceleración de la tierra con respecto al tiempo. Hay dos razones por las cuales estos

instrumentos están diseñados para registrar aceleración y no desplazamiento de la tierra como en el caso de los sismógrafos. La primera razón es que los medidores de aceleración son más fácil de construir que los que miden desplazamiento para el rango de frecuencias de movimientos de tierra fuertes. La segunda razón tiene que ver con el hecho de que, en general, la integración numérica conlleva mucha más precisión que la diferenciación numérica, por lo tanto, es mejor registrar aceleración e integrar para obtener velocidad y desplazamiento que registrar desplazamiento y diferenciar para obtener velocidad y aceleración.

Como los sismógrafos, los acelerógrafos están compuestos de un péndulo, un amortiguador, un dispositivo de registro y un reloj interno; los acelerógrafos modernos también están equipados con un sistema de posicionamiento global. Los acelerógrafos, al igual que los sismómetros, registran tres componentes ortogonales de movimiento: dos horizontales y uno vertical. A diferencia de los sismómetros no hacen un registro continuo del movimiento de la tierra, puesto que los acelerógrafos tienen como fin registrar movimientos fuertes y éstos ocurren después de varios años en un sitio concreto; un registro continuo involucraría un uso ineficiente del medio de almacenamiento, por lo tanto, los acelerógrafos están implementados con un dispositivo que dispara el almacenamiento de datos cuando la aceleración de la tierra sobrepasa un nivel específico.

Como un terremoto desaparece después de algunos minutos, el dispositivo de disparo también es capaz de detener el registro de datos después de que la señal cae por debajo de niveles imperceptibles. La aceleración de la tierra que dispara el registro de datos es lo suficientemente pequeña para detectar todos los eventos importantes pero, al mismo tiempo, es lo suficientemente grande como para no disparar al equipo todo el tiempo debido a vibraciones ambientales. Dado que el registro de datos inicia al tener una aceleración notable, algunos acelerógrafos fallan en registrar la porción de la aceleración que toma lugar antes del inicio del registro. Los acelerógrafos modernos incorporan una memoria de estado sólido (conocida como memoria de pre-evento) que preserva pocos segundos de información antes de que el dispositivo de disparo inicie el registro permanente de datos. También, a diferencia de los sismógrafos, los acelerógrafos operan con baterías puesto que a menudo los terremotos interrumpen la energía eléctrica.

El sistema de registro en un acelerógrafo puede ser tanto analógico como digital. Se dice que el sistema de registro es del tipo analógico si los datos obtenidos en campo son grabados en forma analógica sobre un filme fotográfico o papel ahumado; en contraste, se dice que es del tipo digital si los datos son grabados de forma digital. En un grabador digital la señal del movimiento de la tierra es convertida a su forma digital por medio de un convertidor analógico-digital y grabada en una cinta magnética. Los grabadores analógicos son más simples y menos costosos que los digitales lo cual conlleva un reducido mantenimiento de campo, sin embargo, requieren un laborioso procesamiento de datos en un laboratorio.

Los registradores digitales extienden la respuesta en frecuencia y el rango dinámico del instrumento, también la extracción de los datos es relativamente sencilla y otorgan la posibilidad de procesar la información en una computadora, reduciendo considerablemente el tiempo y el esfuerzo requerido para el procesamiento de ésta. Entre las desventajas de los registradores digitales se encuentra el incremento de la complejidad y costo del sistema y la necesidad de personal capacitado y con amplia experiencia en el uso del instrumento. Otra desventaja es el mal funcionamiento de la cinta del cassette de registro, sin embargo, los acelerógrafos modernos reemplazan la cinta magnética por una memoria de estado sólido.

### *Acelerogramas*

Se llama acelerograma a la grabación obtenida de un acelerógrafo. El primer acelerograma registrado fue obtenido durante el terremoto de Long Beach al sur de California, el 10 de marzo de 1933. Un acelerograma moderno registra típicamente 4 parámetros: la aceleración horizontal en la dirección norte-sur, la aceleración vertical, la aceleración horizontal en la dirección este-oeste y la referencia de tiempo. Los acelerógrafos actuales manejan datos digitales, permitiendo procesar los mismos mediante una computadora. La digitalización se hace típicamente a una frecuencia de 200 muestras por segundo, no obstante, algunos acelerógrafos manejan también tasas de 50 y 100 muestras por segundo.

Existen distintas fuentes de error en el registro y digitalización de los datos que hace que la gráfica proporcionada por el acelerógrafo se desvíe con respecto de la aceleración real de la Tierra. Un problema común para los acelerógrafos es el ruido ambiental puesto que las ondas oceánicas, el tráfico, obras de construcción o cambios en la presión atmosférica son fuentes de ruido de alta y baja frecuencia que puede detectar el instrumento. Otras fuentes de error son: la ubicación incierta de la línea base del registro, la falta de uniformidad en la respuesta del instrumento y los errores introducidos en el registro y digitalización de los datos.

El error introducido por la localización incierta de la línea base del registro está siempre presente en las grabaciones de los acelerógrafos analógicos. Este error surge debido a que los acelerógrafos son disparados después de que la aceleración de la tierra sobrepase un nivel preestablecido y el registro se detiene al caer por debajo de cierto nivel de voltaje; como consecuencia, el acelerógrafo pierde la porción previa al momento de disparo, haciendo imposible la localización de la línea base de registro.

Para mitigar los efectos de los errores mencionados, se implementan diferentes acciones: mejorar la resolución del convertidor analógico-digital o incrementar el rango de frecuencia del instrumento. Los acelerogramas son típicamente procesados antes de hacer el análisis de las señales presentadas en éstos. El procesamiento normalmente consiste en la aplicación de un esquema de corrección para contrarrestar la respuesta no uniforme del instrumento, también el uso de un filtro paso-bajo o paso-alto para remover el ruido no deseado y por último, la integración para obtener la velocidad y desplazamiento de la tierra. Ordinariamente,

la corrección del instrumento es únicamente importante para frecuencias por arriba del rango de interés para la ingeniería; sin embargo, esta corrección puede ser significativamente importante para acelerógrafos situados en o cerca de los pilares de presas y puentes donde el movimiento de las estructuras puede contener componentes de alta frecuencia.

En la figura 1.13 se muestra el acelerograma, después de realizar la corrección de línea base, del terremoto del 19 de septiembre de 1985 en México; en orden descendente las gráficas representan la aceleración en la dirección Norte-Sur, la aceleración vertical y la aceleración en la dirección Este-Oeste. El valor de aceleración máxima de la tierra fue de 0.17g o 168 Gal en la dirección Este-Oeste.

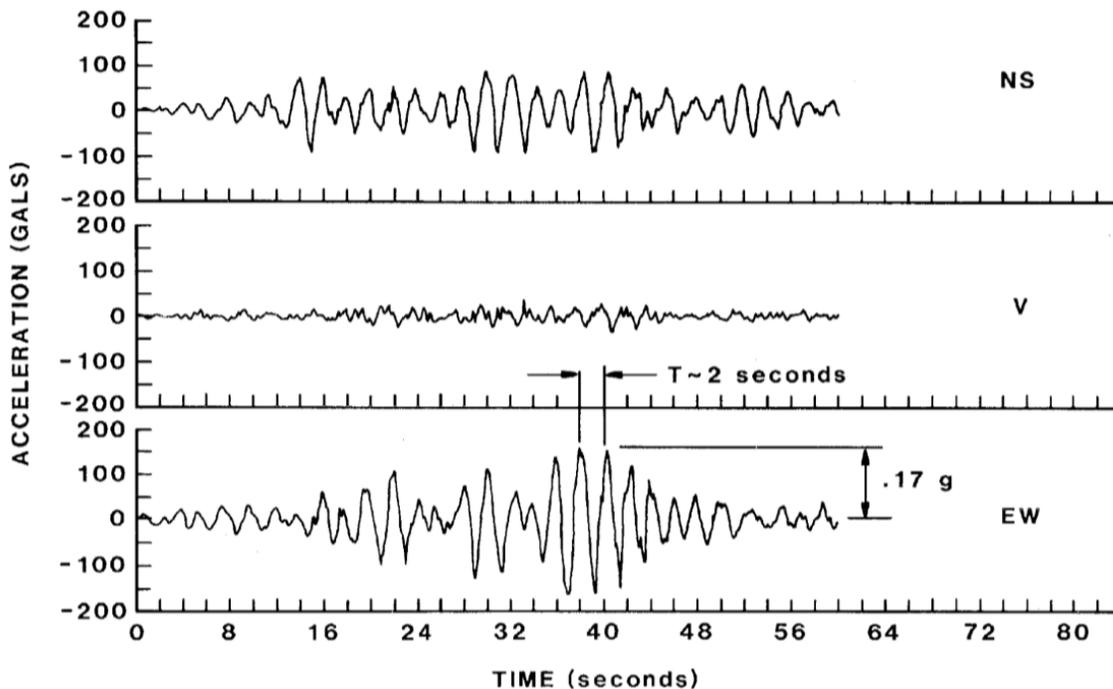


Figura 1.13. Acelerograma del terremoto de 1985 en México.

Gracias a la corrección de la línea base del registro hecha al acelerograma, al integrar y obtener a partir de la aceleración, la velocidad y el desplazamiento, las dos últimas gráficas comienzan a partir del valor cero ( $0 \left[ \frac{cm}{s} \right]$  y  $0 [cm]$ , respectivamente). Esto se puede apreciar en la figura 1.14 para los parámetros correspondientes a la dirección Este-Oeste durante el terremoto de 1985.

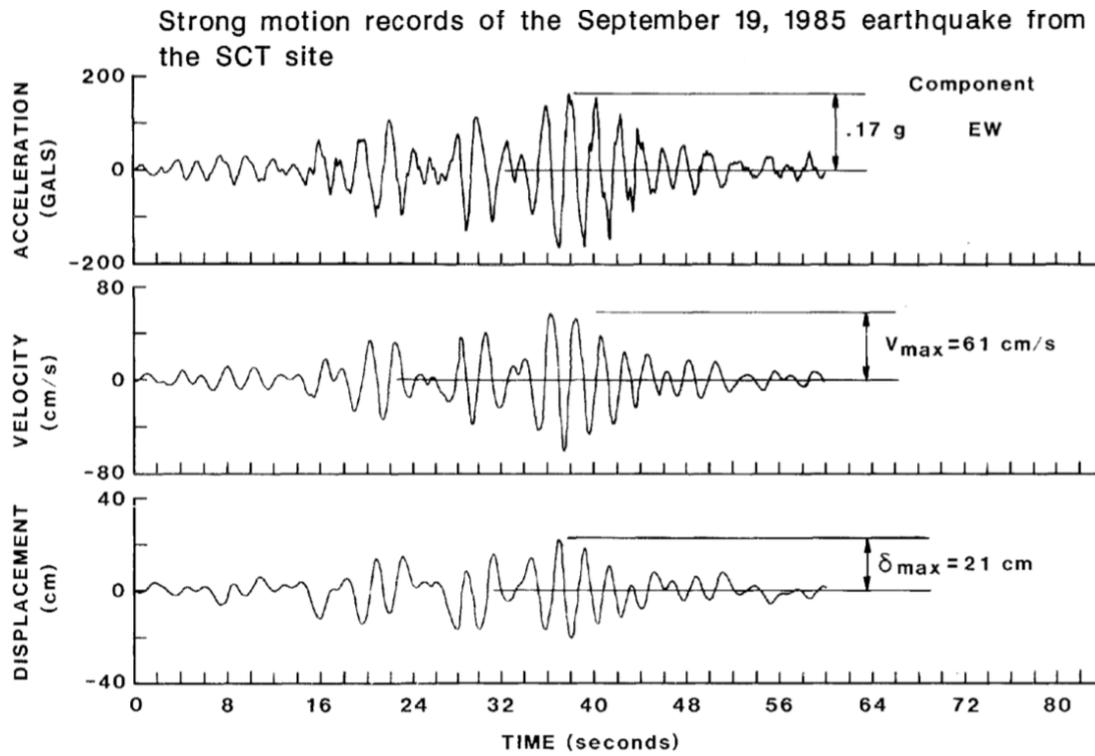


Figura 1.14. Aceleración, velocidad y desplazamiento de la tierra durante el terremoto de 1985.

---

# Capítulo 2.

# Generalidades del sistema

En este capítulo se hará una revisión de los conocimientos en electrónica necesarios para la elaboración del proyecto de tesis. Se abordarán distintos temas para sentar las bases teóricas del funcionamiento de los componentes electrónicos utilizados, se describirá la estructura interna de los componentes, los principios de funcionamiento de los mismos y se especificarán algunos de los parámetros que miden el rendimiento de los circuitos integrados.

Entre los elementos a describir en este capítulo se encuentran el acelerómetro de fuerza balanceada, el acondicionamiento para distinto tipo de señales, la conversión analógico-digital, los protocolos de comunicación utilizados para intercambiar información entre dispositivos electrónicos, las memorias de estado sólido para el almacenamiento de datos, el sistema de posicionamiento global y especificación NMEA 0183 y, finalmente, el uso de microcontroladores.

## 2.1. Acelerómetro de Fuerza Balanceada

Los elementos básicos de un acelerómetro de fuerza balanceada (*Force Balance Accelerometer, FBA*) son un péndulo, un transductor capacitivo, un actuador y un amplificador de señal. La figura 2.1 muestra el esquema de un *FBA*. Cuando el péndulo es llevado fuera de su posición de equilibrio bajo la acción de la aceleración del movimiento de la tierra, el capacitor detecta el desplazamiento y genera una corriente eléctrica que es amplificada y retroalimentada al actuador del péndulo; entonces, el actuador aplica una fuerza que regresa al péndulo a su condición de equilibrio. La corriente que se obtiene a la salida del amplificador alimenta al actuador y es proporcional a la aceleración del péndulo.

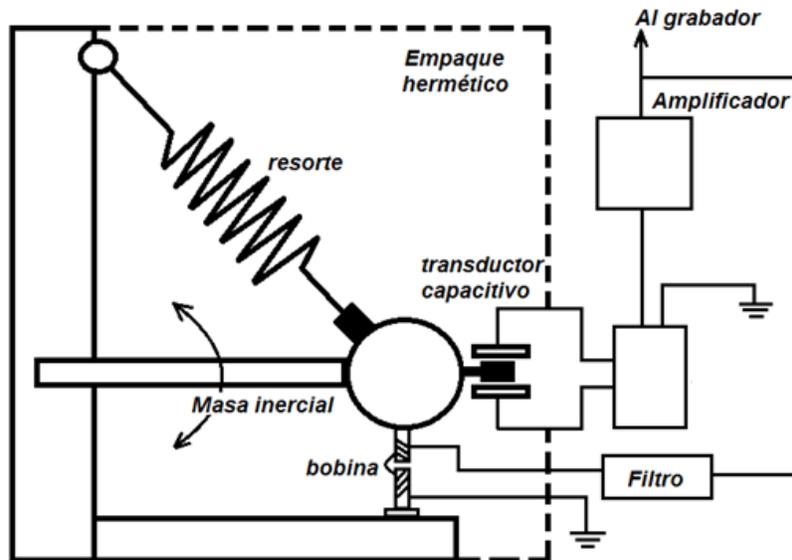


Figura 2.1. Esquema de un FBA.

La aceleración de la tierra se obtiene convirtiendo esta corriente en una señal analógica y registrándola en un medio de almacenamiento de datos, como una cinta magnética o una memoria de estado sólido en el caso de los acelerógrafos digitales. El mérito de los FBA está en que el desplazamiento del péndulo es tan pequeño que las características indeseables, como la no linealidad del resorte son insignificantes.

Como en el caso de los sismógrafos, el diseño de los acelerógrafos se basa en el análisis de la respuesta del sistema sujeto a la aceleración de la tierra, sin embargo, el periodo natural y el coeficiente de amortiguamiento son seleccionados de tal manera que la medición sea proporcional a la aceleración y no al desplazamiento, como en el caso de los sismógrafos.

A partir de la dinámica estructural<sup>1</sup>, se sabe que la respuesta al desplazamiento de un sistema de un grado de libertad sujeto a una aceleración senoidal de la tierra  $\ddot{u}_g(t) = \ddot{u}_0 \text{sen}(\omega_g t)$  está dada por:

$$u(t) = -\frac{1}{\omega_n^2} \ddot{u}_0 D \text{sen}(\omega_g t - \theta) \quad (2.1)$$

Donde:

$\ddot{u}_0$  = amplitud de la aceleración de la tierra

$\omega_n$  = frecuencia natural del sistema

$\omega_g$  = frecuencia de la excitación

$D$  = factor de amplificación dinámico

1. Villaverde R., *Fundamental concepts of earthquake engineering*, primera edición, CRC Press, USA, 2009, p. 156.

El factor de amplificación dinámico se define como:

$$D = \frac{1}{\sqrt{\left[1 - \left(\frac{\omega_g}{\omega_n}\right)^2\right]^2 + \left[2\xi\frac{\omega_g}{\omega_n}\right]^2}} \quad (2.2)$$

Ahora bien, como:

$$\ddot{u}_0 \text{sen}(\omega_g t - \theta) = \ddot{u}_0 \text{sen}\left[\omega_g \left(t - \frac{\theta}{\omega_g}\right)\right] = \ddot{u}_g \left(t - \frac{\theta}{\omega_g}\right) \quad (2.3)$$

Entonces, la respuesta al desplazamiento  $u(t)$  también puede ser expresada como:

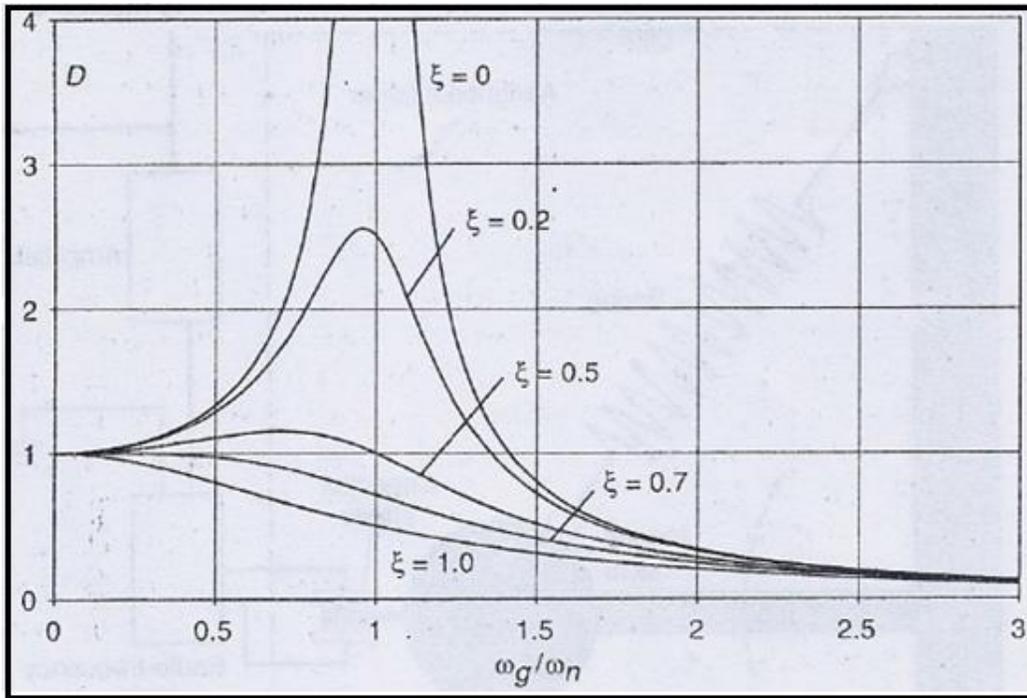
$$u(t) = -\frac{D}{\omega_n^2} \ddot{u}_g \left(t - \frac{\theta}{\omega_g}\right) \quad (2.4)$$

De lo cual se puede concluir que:

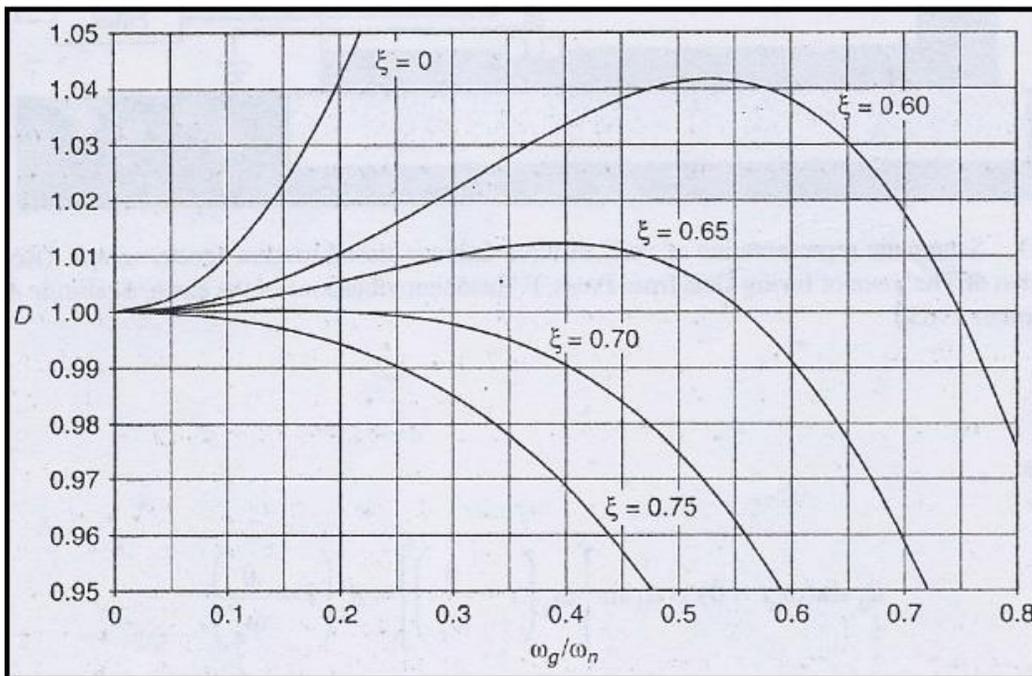
$$\ddot{u}_g \left(t - \frac{\theta}{\omega_g}\right) = -\frac{\omega_n^2}{D} u(t) \quad (2.5)$$

Se observa que debido a que  $\omega_n$  es constante, la aceleración de la tierra es directamente proporcional a la respuesta al desplazamiento del instrumento siempre y cuando el factor  $D$  también sea constante. En las figuras 2.1 (a) y (b) se observa la variación del factor de amplificación dinámico con respecto al coeficiente de amortiguamiento y la relación  $\frac{\omega_g}{\omega_n}$ . En la primera figura se observa que a medida que el coeficiente de amortiguamiento  $\xi$  decrementa, el valor de  $D$  varía en gran proporción y el instrumento deja de ser útil para estudiar el comportamiento de la aceleración de la tierra.

En la figura 2.2(b) se observa que el factor  $D$  es aproximadamente constante para valores de  $\xi$  cercanos a 0.7 y con valores del factor  $\frac{\omega_g}{\omega_n}$  en el rango de 0 a 0.8, con un error de  $\pm 5\%$ . La relación  $0 < \frac{\omega_g}{\omega_n} < 0.8$  es equivalente a:  $0 < \frac{T_n}{T_g} < 0.8$ , donde  $T_n$  y  $T_g$  denotan el periodo natural del instrumento y el periodo dominante de la aceleración de la tierra respectivamente. Si  $T_n = 0.1$  [s], entonces  $D$  es aproximadamente constante en el rango  $T_g > 0.1$  [s], lo cual cubre la mayoría de los casos prácticos de interés en ingeniería sísmica.



(a)  $0 < \frac{\omega_g}{\omega_n} < 3$



(b)  $0 < \frac{\omega_g}{\omega_n} < 0.8$

Figura 2.2. Variación del factor de amplificación dinámico.

Como se mencionó en el apartado 1.2, el espectro sísmico es muy amplio y por ello existen distintos sensores optimizados para ser utilizados en un cierto rango de frecuencia. Los acelerómetros de periodo corto son capaces de detectar señales de 0.1 a 100 [Hz], con una frecuencia natural en torno a 1 [Hz] (valores típicos). Este tipo de sensores son utilizados para el estudio de sismos de baja magnitud. Los acelerómetros de banda ancha detectan señales sísmicas en el rango de frecuencia de 0.01 a 50 [Hz] y son utilizados en estaciones sísmicas. Los acelerómetros de banda muy ancha detectan señales sísmicas comprendidas en el rango de frecuencia de 0.001 a 10 [Hz] y tienen como principal aplicación la investigación de la profundidad de la tierra.

La mayoría de los acelerógrafos son construidos con un periodo natural menor a 0.1 [s] y un coeficiente de amortiguamiento aproximadamente del 60%. Las características de los acelerógrafos comerciales varían dependiendo de los tipos disponibles, pero las características típicas son: frecuencia natural = 25 [Hz]; coeficiente de amortiguamiento = 60-70%; rango =  $\pm 2g$ ; aceleración de disparo = 0.05g; retardo de tiempo de inicio = 0.05-0.10 [s]; medidas = 40×40×20 [cm] y peso = 10 [kg].

Los acelerómetros utilizados por el sistema a desarrollar son los citados debajo:

- *Acelerómetro de fuerza balanceada DSA1 (Kinematics)*

Rango de escala completa:  $\pm 2g$

Frecuencia natural: 50 [Hz]

Coeficiente de amortiguamiento: 70%

Salida:  $\pm 2.5 [V]$  escala completa

Aceleración de inicio de registro: 0.01g

- *Acelerómetro de fuerza balanceada DCA333 (Terra Technology)*

Rango de escala completa:  $\pm 2g$

Frecuencia natural: 50 [Hz]

Coeficiente de amortiguamiento:  $0.7 \pm 0.15$

Salida de escala completa con respecto a la señal de pseudo-tierra:  $\pm 2.5[V]$

Salida con respecto a tierra:  $3.5V_{DC} \pm 2.5[V]$  (1 – 6[V])

Señal de pseudo-tierra:  $3.5V_{DC}$

## 2.2. Acondicionamiento de señales

Las señales que proporcionan los sensores, ya sean de velocidad, temperatura, presión, aceleración o cualquier otro tipo, en la mayoría de los casos necesitan un tipo de tratamiento antes de poder ser digitalizadas; este acondicionamiento de las señales generalmente suele ser la eliminación de ruido, el filtrado, la adición de un

voltaje de directa (denominado offset), la adaptación a los niveles de impedancia de entrada de la etapa siguiente o la amplificación de las mismas.

Para comenzar con el estudio del acondicionamiento de señales, vale la pena definir primero el significado de señal, en concreto, una señal eléctrica. Una señal eléctrica es un tipo de señal generada por un fenómeno electromagnético, cuya magnitud varía, llevando información sobre un proceso o magnitud física. Una de las clasificaciones de las señales eléctricas las define como: analógicas, discretas y digitales.

- Señal analógica: Está descrita por una función matemática continua tanto en su magnitud como en su representación a lo largo del tiempo. La mayoría de los fenómenos reales son ejemplos claros de señales analógicas: la intensidad de la luz, la presión, la temperatura, el sonido, etc.
- Señal discreta: Las señales discretas se caracterizan por estar definidas solamente para un conjunto finito de valores del tiempo. Se representan matemáticamente por secuencias numéricas. En la práctica suelen provenir de un muestreo periódico de una señal analógica.
- Señal digital: Son señales que toman valores discretos tanto en magnitud como en instantes determinados de tiempo. Su magnitud está dada por un código que en el caso de la electrónica digital, se representa únicamente con dos niveles de tensión (1 y 0).

Otra clasificación de las señales eléctricas es: señales unipolares y señales bipolares.

- Señal unipolar: Se mide entre una terminal y otra de referencia, dentro de esta clasificación existe otra subclasificación: señales unipolares puestas a tierra, unipolares flotantes y unipolares con tensión en modo común. En las primeras, la terminal de referencia está conectada a tierra, en las segundas, la terminal de referencia es independiente de tierra y se puede conectar a tierra sin afectar el circuito. El tercer tipo de señales se presentan cuando existe un voltaje entre la terminal de tierra y la terminal de referencia y no se puede conectar ésta última a tierra. En la figura 2.3 se muestran respectivamente: una señal unipolar puesta a tierra, unipolar flotante y unipolar con tensión en modo común.

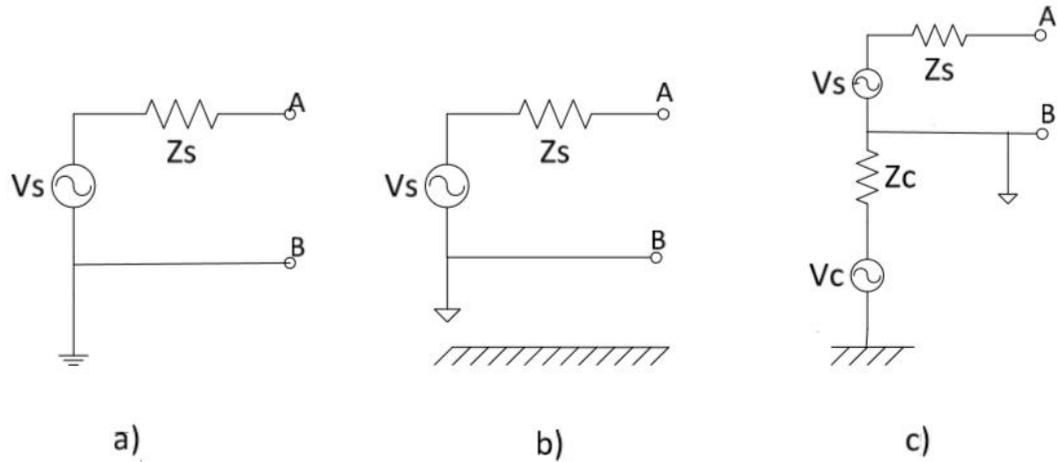


Figura 2.3. Señales unipolares.

- Señal bipolar: La medición se realiza entre dos terminales independientes de la terminal de referencia, que a su vez puede estar o no conectada a tierra, la polaridad con que se tome la señal es irrelevante ya que sólo cambia el signo. En este grupo de señales también existe la misma subclasificación que para las señales unipolares. En la figura 2.4 se muestran respectivamente: una señal bipolar puesta a tierra, una señal bipolar flotante y una señal bipolar con tensión en modo común.

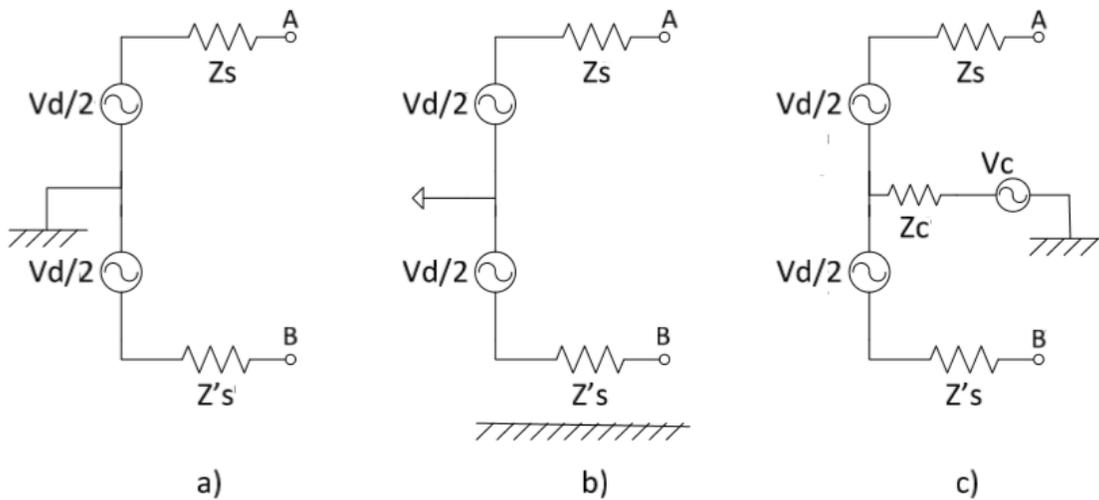


Figura 2.4. Señales bipolares.

*Señales de alta y baja impedancia.* Al interconectar dos elementos de un sistema electrónico hay dos posibles situaciones: La primera sería, que la tensión o corriente de entrada de un elemento coincida con la correspondiente a la salida del elemento precedente, y la segunda, que se transfiera la máxima potencia de un elemento a otro. En ambos casos es necesario adaptar la impedancia de entrada del circuito a conectar.

Para plantear el primer caso, se presenta el esquema de la figura 2.5. En el diagrama mostrado, se debe evitar que al conectar el circuito B al circuito A, haya una atenuación en el voltaje proveniente del primer circuito, obteniendo el voltaje en la carga  $Z_m$ :

$$V_m = \frac{Z_m}{Z_m + Z_o} V_o \quad (2.6)$$

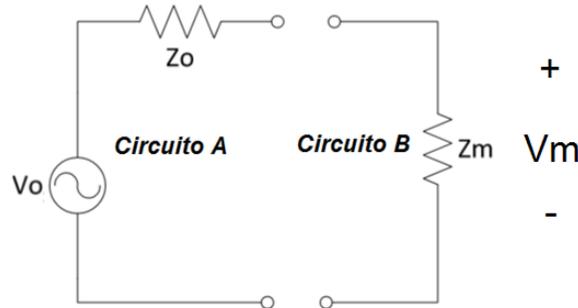


Figura 2.5. Medición del voltaje en  $Z_m$ .

Para que el voltaje en la carga  $Z_m$  sea aproximadamente el mismo que  $V_o$ , el valor de la impedancia de entrada del segundo circuito ( $Z_m$ ) debe ser mucho mayor que la impedancia de salida  $Z_o$  del circuito A.

En el caso de que lo que se desee es evitar la atenuación de la corriente, se debe tomar en cuenta la figura 2.6. En este caso la corriente de entrada  $i_o$  debe ser la misma que la corriente que circula por  $Z_m$ .

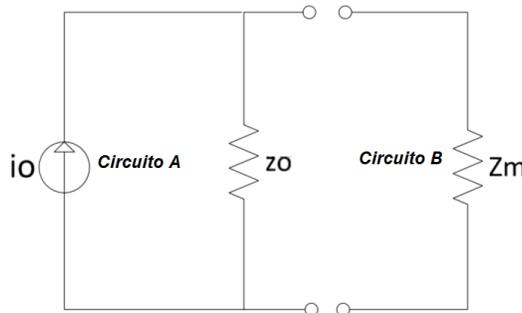


Figura 2.6. Medición de la corriente en  $Z_m$ .

En este caso se obtiene la siguiente expresión:

$$i_m = \frac{Z_o}{Z_o + Z_m} i_o \quad (2.7)$$

Para que la corriente no se vea atenuada, el valor de la impedancia de entrada  $Z_m$  debe ser mucho menor que la impedancia de salida  $Z_o$  del circuito A.

Para el último caso, en el que lo que se necesita es transferir la máxima potencia desde un circuito A hacia una carga y partiendo del esquema de la figura 2.7, para obtener el valor de  $R_m$  que haga máxima la potencia que se le entrega, es necesario considerar la corriente que pasa por esta misma, la cual se muestra en la ecuación (2.8). Por otra parte, la potencia (P) suministrada a la resistencia  $R_m$  se obtiene mediante la expresión 2.9.

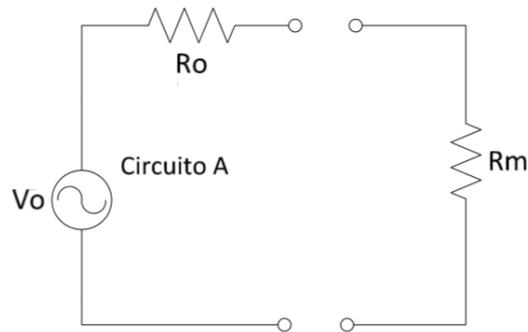


Figura 2.7. Acoplamiento de impedancias.

$$i = \frac{V_o}{R_m + R_o} \quad (2.8)$$

$$P = R_m * \left( \frac{V_o}{R_m + R_o} \right)^2 \quad (2.9)$$

Al derivar la ecuación (2.9) con respecto a  $R_m$  y utilizando el criterio de la primera derivada se puede obtener el máximo relativo para esta función:

$$\frac{dP}{dR_m} = V_o^2 \frac{(R_o + R_m)^2 - 2(R_o + R_m)R_m}{(R_m + R_o)^4} = 0 \quad (2.10)$$

$$(R_o + R_m)^2 - 2(R_o + R_m)R_m = 0 \quad (2.11)$$

De la última expresión se obtiene que  $R_m = R_o$ , por lo tanto, para lograr una máxima transferencia de potencia a la carga, el valor de  $R_m$  tendrá que ser el mismo valor que el de  $R_o$ .

### 2.2.1. Filtros electrónicos

#### *Clasificación de los filtros electrónicos*

Puede definirse un filtro como cualquier dispositivo que modifica de un modo determinado una señal que pasa a través de él. A menudo se reserva el término filtro para un dispositivo selector de frecuencia, es decir, aquellos que permiten el paso de la señal en ciertas bandas de frecuencia y discriminan la señal en otras bandas. Aunque existen muchos filtros de interés práctico que no cumplen esa función, por lo que se prefiere la definición más amplia, la mayoría de filtros son

selectores de frecuencia. La excepción la constituyen los *filtros pasatodo* que, sin alterar la amplitud, modifican la fase.

Hay diversas clasificaciones de los filtros. Cuando la señal es una magnitud eléctrica (corriente o tensión) es un filtro electrónico. Existen también filtros mecánicos, filtros acústicos, filtros ópticos, etc.

Otra clasificación es: filtros lineales y filtros no lineales, según que su comportamiento pueda o no modelarse matemáticamente con ecuaciones lineales. Un ejemplo de filtro no lineal es un comparador de tensión, otro, un rectificador.

Una clasificación más es: filtros analógicos y filtros digitales. Los filtros analógicos son aquéllos en los cuales la señal puede tomar cualquier valor dentro de un intervalo, y los digitales corresponden al caso en que la señal toma sólo valores discretos. También pueden clasificarse en filtros continuos y filtros discretos o muestreados, según que la señal se considere en todo instante o en instantes discretos. Dado que los filtros digitales en la práctica son siempre muestreados, el nombre *filtro digital* se refiere habitualmente a filtros discretos digitales. Sin embargo, existen filtros discretos no digitales, como los filtros de capacidades conmutadas.

Finalmente, los filtros también pueden clasificarse en filtros activos o filtros pasivos, según empleen o no fuentes controladas (elementos activos, tal como amplificadores y sus derivados). Los filtros eléctricos pasivos se implementan en general con inductores y capacitores. Dado que los inductores son elementos voluminosos, pesados y costosos, el empleo de filtros pasivos es poco conveniente excepto en frecuencias bastante altas. Los inductores pueden eliminarse mediante el uso de amplificadores y técnicas de realimentación.

Los filtros son utilizados en la etapa de acondicionamiento de señal de los circuitos eléctricos y electrónicos. Estos filtros tienen diferentes funciones, entre ellas:

(a) Acondicionamiento de la señal de entrada al sistema

- Eliminar interferencias y ruido de las señales procedentes de sensores.
- Limitación del ancho de banda al rango útil del sistema.
- Eliminación de frecuencias superiores antes de la conversión analógico-digital (anti-aliasing).
- Sintonización de señales útiles (p. ej. demodulación).

(b) Acondicionamiento de la señal de salida

- Eliminación de armónicos innecesarios (p. ej. Alisado de los convertidores digital-analógico).
- Supresión de ruido e interferencias que haya introducido el sistema.

Los filtros se pueden clasificar según su respuesta en frecuencia, teniéndose de esta manera, cuatro categorías diferentes:

- *Filtro pasa bajos*: Permite el paso de señales eléctricas cuya frecuencia es baja.
- *Filtro pasa altos*: Atenúa las señales cuya frecuencia es baja; por otro lado, permite el paso de señales con componentes de alta frecuencia.
- *Filtro pasa banda*: En este tipo de filtros existen dos frecuencias de corte, una inferior y otra superior. Este filtro sólo permite el paso de señales cuya frecuencia se encuentra entre las frecuencias de corte antes mencionadas.
- *Filtro supresor de banda*: Este filtro elimina en su salida todas las señales que tengan una frecuencia comprendida entre una frecuencia de corte inferior y una frecuencia de corte superior.

Algunas definiciones que vale la pena hacer al tratar el tema de los filtros electrónicos son:

*Octava*: Dos frecuencias están separadas una octava si una de ellas es el doble de la otra.

*Década*: Dos frecuencias están separadas una década si una de ellas es 10 veces la otra.

*Frecuencia de corte*: Es la frecuencia en la que se tiene el 70.7% (-3dB) del valor de la ganancia total de voltaje.

*Banda de paso*: Es el rango de frecuencias que el filtro deja pasar desde la entrada hasta la salida con una atenuación máxima de 3 dB.

*Banda de rechazo*: Es el rango de frecuencias que el filtro atenúa más de 3 dB.

### *Filtros pasivos*

Como se ha mencionado, los filtros pasivos son aquellos cuya implementación práctica requiere únicamente de elementos pasivos, que no necesitan ninguna fuente de alimentación, los cuales pueden ser resistencias, capacitores e inductores.

La figura 2.8 muestra la implementación de un filtro pasivo pasa bajos de primer orden.

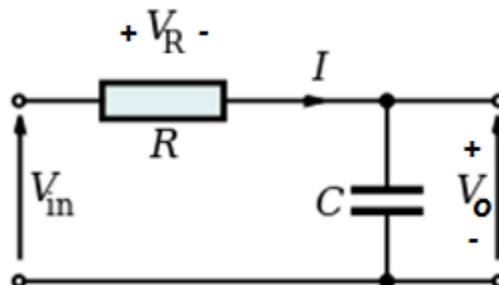


Figura 2.8. Filtro pasivo pasa bajos de primer orden.

La relación entre el voltaje de salida  $V_o$  y el voltaje de entrada  $V_{in}$  se puede obtener al observar que el circuito es un divisor de tensión, de esta manera:

$$V_o(S) = \frac{\frac{1}{CS}}{R + \frac{1}{CS}} V_{in}(S) = \frac{1}{1 + RCS} V_{in}(S) \quad (2.12)$$

La función de transferencia es:

$$\frac{V_o(S)}{V_{in}(S)} = \frac{1}{1 + RCS} = \frac{\frac{1}{RC}}{s + \frac{1}{RC}} \quad (2.13)$$

De la ecuación (2.13) se observa que en la función de transferencia existe la presencia de un polo en la frecuencia  $\omega_c = \frac{1}{RC}$ , denominada frecuencia de corte. El efecto de un polo en la función de transferencia, es que la ganancia de voltaje decae a una razón de 20 dB por década a partir de la frecuencia de corte. En la figura 2.9 se muestra el diagrama de bode tanto de ganancia como de fase del filtro pasa bajos de la figura 2.8; en cuanto a la ganancia, se observa que a lo largo de la banda de paso, la señal no tiene ninguna atenuación y todas las componentes de baja frecuencia pasan desde la entrada del filtro hasta la salida del mismo. A partir de la frecuencia de corte las señales con componentes de alta frecuencia, ubicadas en la banda de rechazo, sufren una atenuación debida a la acción del filtro electrónico. En cuanto a la fase, una señal de baja frecuencia sufre poca modificación al salir del filtro, cosa que no sucede al aumentar la frecuencia de la señal pues cuando esto sucede, la fase de la señal se modifica con el aumento de la frecuencia hasta llegar a desfasarse  $90^\circ$ .

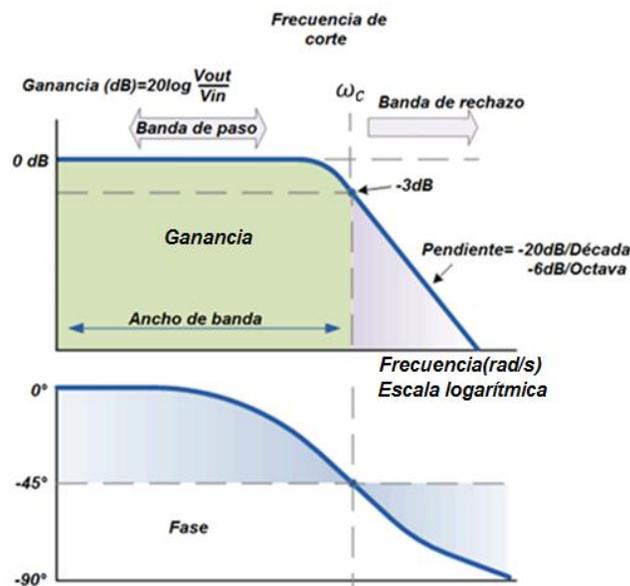


Figura 2.9. Respuesta en frecuencia del filtro pasa bajos.

La figura 2.10 muestra un filtro pasa altos, obtenido al intercambiar la posición de los elementos del circuito analizado anteriormente y tomando esta vez el voltaje en las terminales de la resistencia en lugar de en las terminales del capacitor.

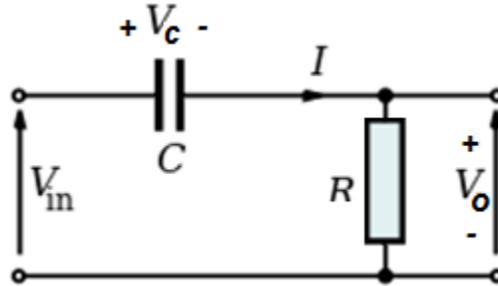


Figura 2.10. Filtro pasivo pasa altos de primer orden.

La relación entre el voltaje de entrada  $V_{in}$  y el voltaje de salida  $V_o$  para este circuito se puede determinar observando que el arreglo es un divisor de tensión, de esta manera:

$$V_o(S) = \frac{R}{R + \frac{1}{CS}} V_{in}(S) = \frac{RCS}{1 + RCS} V_{in}(S) \quad (2.14)$$

La función de transferencia es:

$$\frac{V_o(S)}{V_{in}(S)} = \frac{RCS}{1 + RCS} = \left( \frac{1}{RC} \right) \frac{S}{S + \frac{1}{RC}} \quad (2.15)$$

La ecuación (2.15) indica la presencia de un cero y un polo; el cero se encuentra en el origen ( $\omega = 0$ ) y tiene una contribución de un aumento de 20 dB por década en la ganancia de voltaje del filtro. Al aumentar la frecuencia y llegar a la frecuencia de corte  $\omega = \frac{1}{RC}$ , el efecto en la ganancia del cero se contrarresta con el efecto del polo, y la ganancia permanece constante a partir de esta frecuencia. En la figura 2.11 se muestra el diagrama de Bode, tanto de ganancia como de fase, del filtro pasa altos. Se observa que a bajas frecuencias, el filtro responde con una atenuación de las señales en este rango (banda de rechazo); a medida que la frecuencia aumenta hasta llegar a la frecuencia de corte, el filtro permite el paso de las señales de altas frecuencias sin atenuar su magnitud en voltaje. Por otro lado, las señales con componentes de baja frecuencia tienen un desfase de  $90^\circ$  al salir del filtro, mientras que aquellas con componentes de alta frecuencia no sufren desfase en la transición de la entrada a la salida del filtro electrónico.

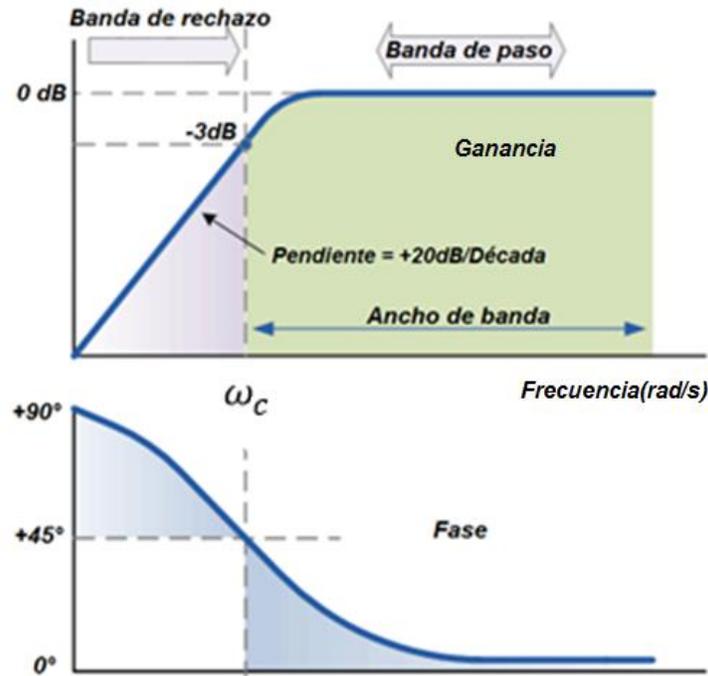


Figura 2.11. Respuesta en frecuencia del filtro pasa altos.

La forma más simple de construir un filtro paso banda es interconectando un filtro pasivo paso altos y un filtro pasivo paso bajas, como los mostrados en la figura 2.12. Se debe realizar un acoplamiento de impedancias (por ejemplo con un amplificador operacional) de manera que el segundo circuito no genere un efecto de sobrecarga al primer circuito.

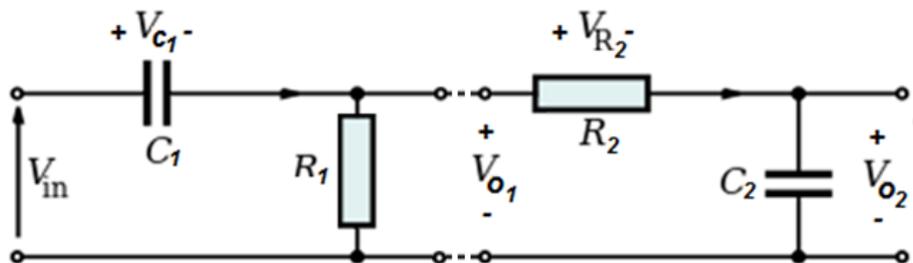


Figura 2.12. Filtro pasa banda.

La relación entre el voltaje de entrada  $V_{in}$  y el voltaje de salida  $V_{o2}$  está dada por:

$$\frac{V_{o2}(S)}{V_{in}(S)} = \frac{\frac{1}{C_2 S}}{R_2 + \frac{1}{C_2 S}} \left( \frac{R_1}{R_1 + \frac{1}{C_1 S}} \right) = \left( \frac{1}{R_1 C_1 R_2 C_2} \right) \left( \frac{1}{S + \frac{1}{R_2 C_2}} \right) \left( \frac{S}{S + \frac{1}{R_1 C_1}} \right) \quad (2.16)$$

La función de transferencia de la ecuación (2.16) indica la presencia de un cero y dos polos; como se observa en el diagrama de ganancia de la figura 2.13, la primer contribución es debida al cero, éste se sitúa en el origen y es por ello que la ganancia

aumenta 20 dB por década. El filtro posee dos frecuencias de corte, una inferior  $\omega_L$  y otra superior  $\omega_H$ . Al aumentar la frecuencia hasta llegar a la frecuencia  $\omega_L$ , el efecto del cero en el origen y el primer polo se contrarrestan y la ganancia del sistema es constante. Prosiguiendo con el aumento de frecuencia hasta llegar a la frecuencia  $\omega_H$ , el segundo polo contribuye a la ganancia con un decaimiento de 20 dB por década, a partir de esta frecuencia el filtro atenúa las componentes de más alta frecuencia. En el caso de los filtros pasa banda, el ancho de banda  $BW$  se define como:  $BW = \omega_H - \omega_L$ .

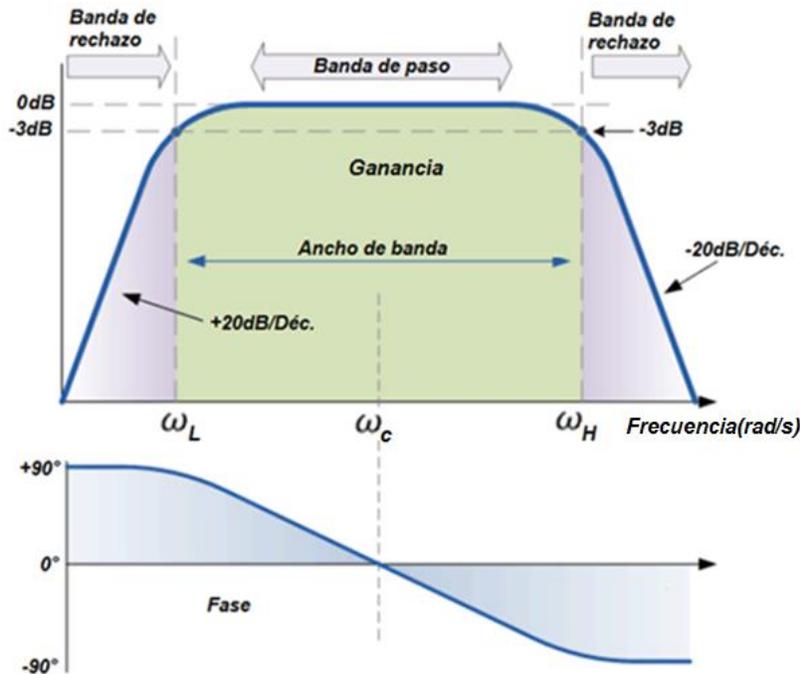


Figura 2.13. Respuesta en frecuencia del filtro pasa banda.

El filtro electrónico permite el paso de las señales cuyas frecuencias se sitúan entre las dos frecuencias de corte, por otro lado, atenúa aquellas señales cuyas componentes en frecuencia se sitúan fuera de éste rango. La frecuencia situada en el punto medio del rango antes indicado se denomina frecuencia central  $\omega_c$ ; surge otro parámetro conocido como factor de calidad  $Q$  que indica en qué medida el filtro selecciona las frecuencias que deja pasar. Si el factor de calidad es grande, únicamente un rango de frecuencias muy estrecho podrán pasar a través del filtro mientras que al tener un factor de calidad bajo, un rango más amplio de frecuencias pueden pasar a través de él. El factor de calidad se define como  $Q = \omega_c / BW$ .

En cuanto a la fase, las señales de baja frecuencia sufren un desfase de  $+90^\circ$ , a medida que la frecuencia de la señal aumenta este desfase disminuye hasta llegar a la frecuencia central donde es nulo. Si la frecuencia de la señal aumenta, el desfase de la señal cambia de dirección hasta llegar a  $-90^\circ$  para las frecuencias más altas.

El filtro rechazo de banda es otro tipo de filtro selector de frecuencia cuyo funcionamiento es exactamente el opuesto al filtro pasa banda. El filtro rechazo de banda permite el paso de todas las señales excepto aquellas cuyas frecuencias se encuentran en un rango determinado; si el rango de frecuencias es muy estrecho, al filtro también se le denomina filtro notch (voz inglesa de muesca). Al igual que el filtro pasa banda, su función de transferencia es de segundo orden (dos polos), teniendo dos frecuencias de corte  $\omega_L$  y  $\omega_H$ ; entonces el filtro permite el paso de frecuencias desde DC (Direct Current) hasta la primer frecuencia de corte  $\omega_L$ , bloquea las frecuencias entre el rango que comprenden las dos frecuencias de corte y, finalmente, permite el paso de las frecuencias superiores a la frecuencia  $\omega_H$ .

Es posible conectar más de un filtro en cascada para lograr obtener un filtro de orden superior, ello implica que la pendiente del filtro sea más pronunciada, asemejándose más a un filtro ideal y por lo tanto, mejorando la respuesta en frecuencia; no obstante, al realizar esto, los circuitos pueden representar una carga para los circuitos predecesores atenuando la señal a la salida del filtro. También es posible implementar filtros de orden superior como el filtro de segundo orden pasa bajos de la figura 2.14, sin embargo, es necesario utilizar bobinas, las cuales son más difícil de obtener y por lo tanto el costo del filtro se incrementa.

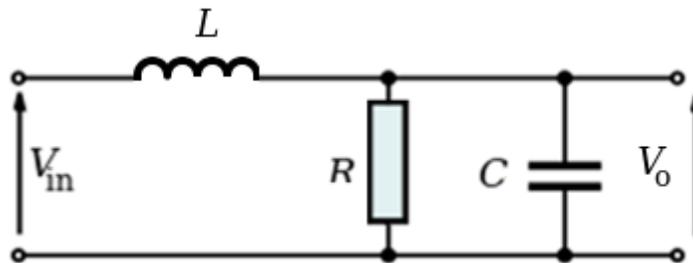


Figura 2.14. Filtro pasivo pasa bajos de segundo orden.

Los filtros activos surgieron porque presentan ciertas ventajas con respecto a los filtros pasivos, algunas de ellas son las siguientes:

- No utilizan bobinas en los circuitos; marcando la ventaja más notable pues las bobinas son muy difícil de conseguir o diseñar.
- Adaptan impedancias.
- Es fácil crear filtros difíciles o complejos mediante la colocación de varios filtros simples conectados en cascada.
- Es posible contar con ganancia al utilizar elementos activos.
- Son baratos.

Para la aplicación que tiene como objetivo este trabajo, no se necesita ni amplificación, ni el uso de bobinas, además el filtro se reduce a uno de primer orden debido a las prestaciones del convertidor analógico digital que se describirá posteriormente, es por ello que los filtros activos no serán estudiados a profundidad, sin embargo, se hará mención de ellos a continuación.

*Filtros activos*

Los filtros activos más sencillos de implementar son los de primer orden, los cuales típicamente se implementan utilizando amplificadores operacionales. El circuito de la figura 2.15 muestra un filtro pasa altos de primer orden.

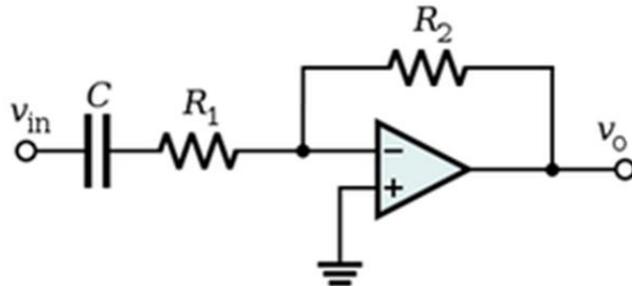


Figura 2.15. Filtro activo pasa altos de primer orden.

La función de transferencia del circuito anterior es la siguiente:

$$\frac{V_o}{V_{in}} = -\frac{R_2}{R_1 + \frac{1}{CS}} = -\frac{R_2 CS}{R_1 CS + 1} = -\left(\frac{R_2}{R_1}\right) \frac{S}{S + \frac{1}{R_1 C}} \quad (2.17)$$

La ecuación (2.17) indica la presencia de un cero en el origen y un polo en la frecuencia  $\omega = \frac{1}{R_1 C}$ , se deduce que el comportamiento en frecuencia es igual al mostrado en la figura 2.11. La diferencia entre este filtro y el de la figura 2.10 es que la señal es invertida y además la ganancia se modifica por el factor  $\frac{R_2}{R_1}$ . De esta manera, si  $R_2 > R_1$  la señal a la salida del filtro se verá amplificada, si por el contrario  $R_2 < R_1$ , la señal será atenuada al pasar a través del filtro. En el caso en que  $R_1$  y  $R_2$  sean iguales, el filtro no afectará en magnitud a la señal de entrada.

Los filtros activos de segundo orden presentan una mejor respuesta en frecuencia al contar con pendientes más pronunciadas y atenuar en mayor medida los rangos de frecuencia que no son de interés. Tomando como ejemplo el circuito de la figura 2.15, al colocar dos filtros de primer orden en cascada, se dará lugar a un filtro de segundo orden; si los valores de las resistencias y los capacitores son iguales, el sistema contará con un polo doble y el decaimiento de ganancia será de 40 [dB] por década, si por el contrario, los valores de las resistencias y capacitores son distintos, se tendrán dos frecuencias de corte; al llegar a la primera, el filtro tendrá una atenuación de 20 dB por década mientras que al llegar a la segunda frecuencia de corte, el decaimiento de la ganancia será del doble debido a la acción del segundo polo.

Un tercer caso es cuando el sistema cuenta con polos complejos conjugados. La función de transferencia de un filtro pasa bajos de segundo orden es de la forma:

$$H(S) = \frac{H_o \omega_0^2}{S^2 + 2\xi \omega_0 S + \omega_0^2} \quad (2.18)$$

Para el caso de los filtros pasa altos y pasa banda, las funciones de transferencia están dadas por las ecuaciones 2.19 y 2.20 respectivamente.

$$H(S) = \frac{H_0 S^2}{S^2 + 2\xi\omega_0 S + \omega_0^2} \quad (2.19)$$

$$H(S) = \frac{H_0 2\xi\omega_0 S}{S^2 + 2\xi\omega_0 S + \omega_0^2} \quad (2.20)$$

La respuesta en magnitud de la función de transferencia de los filtros pasa bajos, pasa altos y pasa banda, se muestran respectivamente en las figuras 2.16, 2.17 y 2.18. En estas gráficas se aprecia que la respuesta en frecuencia depende del coeficiente de amortiguamiento  $\xi$ . En ellas se aprecia que mientras mayor sea, la elongación tiende a disminuirse y se habla de un sistema sobre amortiguado. Para el caso del filtro pasa bajos, el sistema proporciona una ganancia relativamente constante para las componentes de baja frecuencia; sin embargo, al elevar la frecuencia de la señal de entrada y llegar a la frecuencia de corte, el sistema comenzará a atenuar la señal a una razón de 40 dB por década. Para el filtro pasa altas, las frecuencias más bajas son atenuadas y el sistema proporciona un aumento en la ganancia de 40 dB por década hasta llegar a la frecuencia de corte, en ese punto la ganancia es constante excepto en el codo. Para el último caso, todas las señales son atenuadas excepto aquellas cuya frecuencia se sitúa entre un rango determinado.

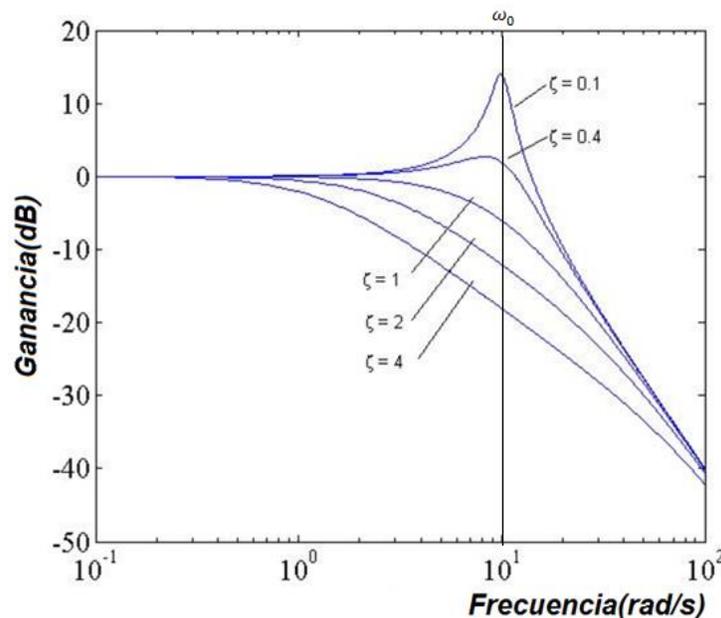


Figura 2.16. Respuesta del filtro pasa bajos.

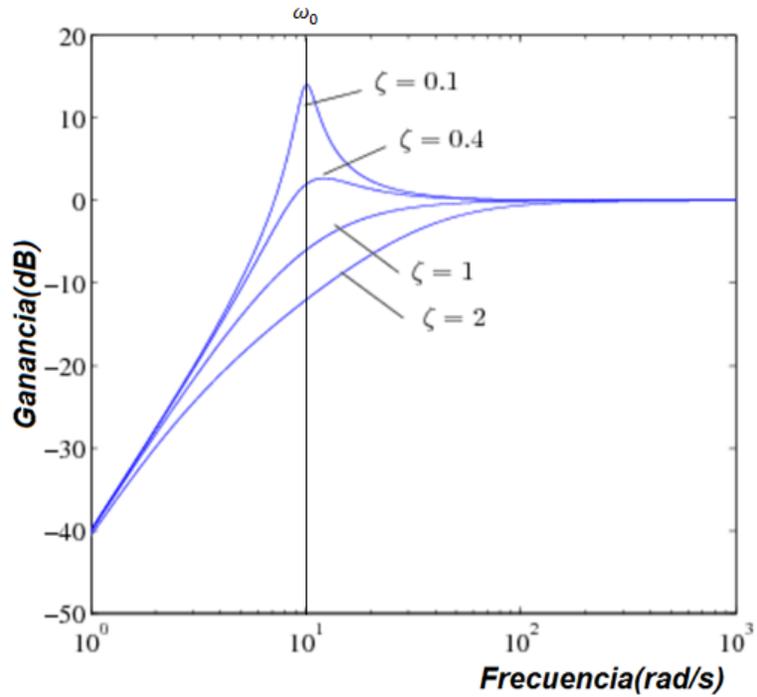


Figura 2.17. Respuesta del filtro pasa altos.

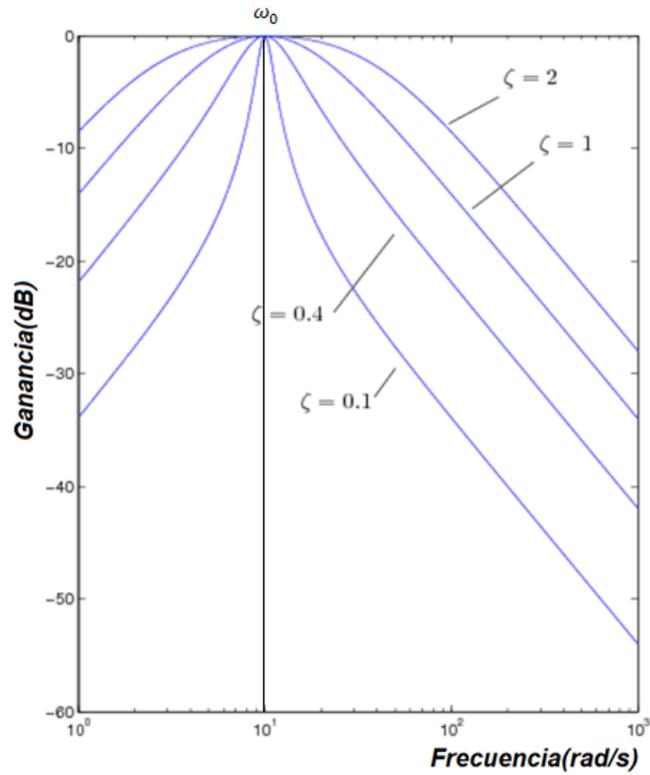


Figura 2.18. Respuesta del filtro pasa banda.

Las realizaciones prácticas de los filtros activos de segundo orden se basan en la estructura general de la figura 2.19. De ella se deducen las dos topologías más utilizadas: *Rauch* y *Sallen-Key*.

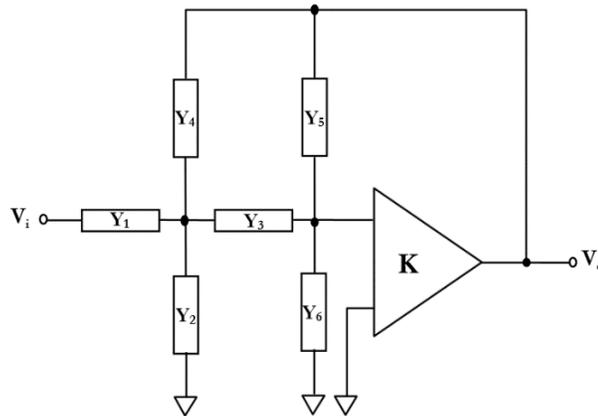


Figura 2.19. Circuito genérico para la síntesis de filtros activos de segundo orden.

El bloque de ganancia  $K$  es una fuente de voltaje controlada por voltaje (*VCVS*, *Voltage Controlled Voltage Source*). Cada admitancia  $Y_i$  cumple con la relación  $Y_i = 1/Z_i$ . La función de transferencia para esta configuración es:

$$H(S) = \frac{KY_1Y_3}{[Y_5(Y_1+Y_2+Y_3+Y_4)+Y_3Y_4](1-K)+Y_6(Y_1+Y_2+Y_3+Y_4)+Y_3(Y_1+Y_2)} \quad (2.21)$$

*Topología de realimentación múltiple y ganancia infinita o de Rauch.*

El *filtro de realimentación múltiple y ganancia infinita* o de *Rauch* se sintetiza a partir del circuito anterior y de la ecuación (2.21), considerando que: la ganancia es negativa e infinita, de este modo, se emplea un amplificador operacional en lazo abierto y la admitancia  $Y_6$  se selecciona de manera que no aparece en el circuito. A partir de estas consideraciones surge el circuito de la figura 2.20.

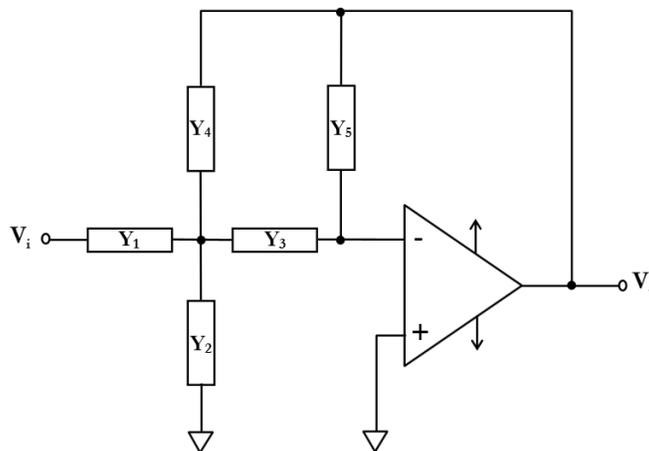


Figura 2.20. *Topología de realimentación múltiple y ganancia infinita* o de *Rauch*.

La función de transferencia resultante de la topología de *Rauch* es:

$$H(S) = \frac{-Y_1 Y_3}{Y_5(Y_1 + Y_2 + Y_3 + Y_4) + Y_3 Y_4} \quad (2.22)$$

Las distintas realizaciones prácticas de los tipos de respuestas de los filtros surgen de los casos particulares de esta función de transferencia, en concreto, del carácter resistivo o capacitivo de las impedancias, por ejemplo, el circuito de la figura 2.21 muestra la implementación de un filtro pasa bajas.

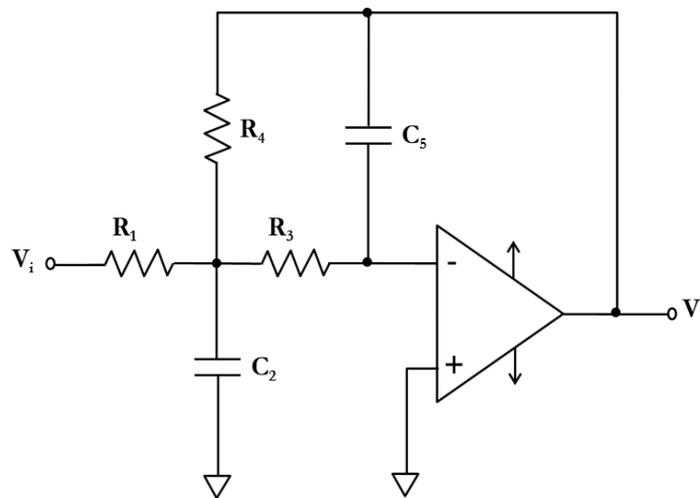


Figura 2.21. Filtro pasa bajas con la topología de *Rauch*.

La función de transferencia del circuito anterior está dada por:

$$H(S) = \frac{-\frac{1}{R_1 R_3 C_2 C_5}}{S^2 + S \frac{1}{C_2} \left( \frac{1}{R_1} + \frac{1}{R_3} + \frac{1}{R_4} \right) + \frac{1}{R_3 R_4 C_2 C_5}} \quad (2.23)$$

Las ecuaciones de diseño se obtienen comparando la ecuación anterior con la genérica pasa bajas (ecuación 2.18), de este modo, comparando término a término ambas ecuaciones, se pueden elegir el valor de los componentes para que proporcionen la frecuencia de corte y coeficiente de amortiguamiento deseados.

### Topología VCVS y de Sallen-Key

El circuito general VCVS (*Voltage Controlled Voltage Source*) se obtiene aplicando al circuito general dos reglas:  $Z_5 = 1/Y_5$  se considera infinita y  $K$  se toma de una configuración no inversora. Un caso particular, la topología Sallen-Key considera además a  $Z_2 = 1/Y_2$  también como infinita. En el circuito de la figura 2.21 se indican las acciones anteriores sobre las impedancias.

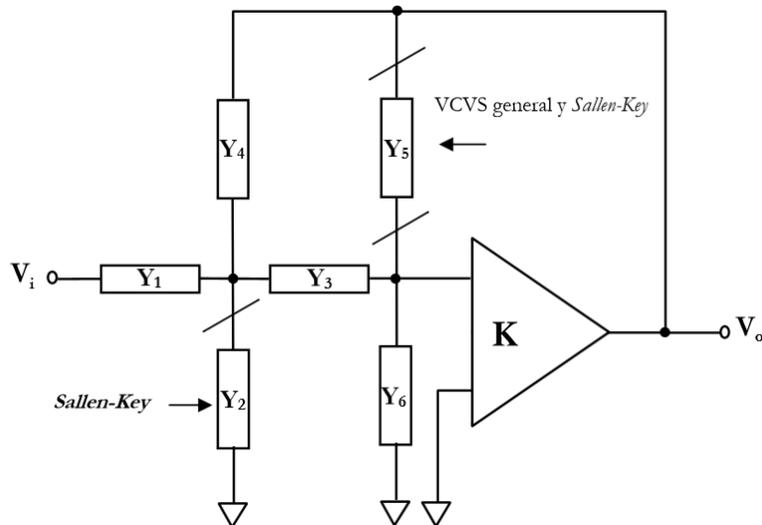


Figura 2.22. Topología del circuito general VCVS y Sallen-Key.

La función de transferencia para la topología *Sallen-Key* es:

$$H(S) = \frac{KY_1Y_3}{(Y_1+Y_3+Y_4)Y_6+Y_1Y_3+(1-K)Y_3Y_4} \quad (2.24)$$

La realización de un filtro pasa bajos se muestra en la figura 2.23.

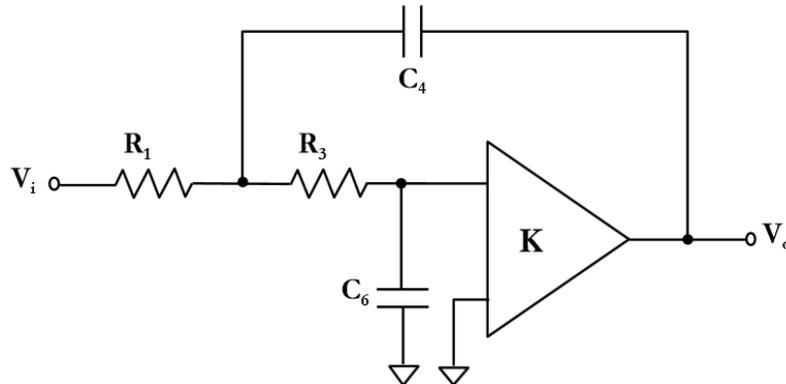


Figura 2.23. Filtro pasa bajos con topología de Sallen-Key.

La función de transferencia se obtiene sustituyendo cada valor de impedancia:

$$H(S) = \frac{K \frac{1}{R_1 R_3 C_4 C_6}}{S^2 + \left[ \frac{1}{R_1} + \frac{1}{R_3} + (1-K) \frac{1}{R_3} \frac{C_4}{C_6} \right] \frac{1}{C_4} S + \frac{1}{R_1 R_3 C_4 C_6}} \quad (2.25)$$

Nuevamente, las ecuaciones de diseño se obtienen comparando la ecuación 2.25 con la genérica pasa bajos.

### 2.2.2. El amplificador de instrumentación

Al trabajar con señales analógicas el acoplamiento entre sensores y una etapa de acondicionamiento, introduce ruido eléctrico a las señales que necesitan estudiarse. Si se usan cables largos, el ruido inducido por un ambiente eléctricamente ruidoso puede también producir señales no deseadas en la entrada de las etapas siguientes. Con la finalidad de eliminar este ruido, se emplean amplificadores de instrumentación.

Un amplificador de instrumentación es un tipo de amplificador diferencial que ha sido equipado con *buffers* (dispositivo electrónico que sirve para acoplar impedancias) de entrada, lo que lo convierte en un amplificador especialmente adecuado para su uso en equipos de medición y prueba. Las características adicionales que lo hacen óptimo para estas aplicaciones son: nivel muy bajo de offset de corriente directa (DC, Direct Current), bajo ruido, alta ganancia de lazo abierto, alta razón de rechazo en modo común (CMRR, Common-Mode Rejection Ratio) y alta impedancia de entrada. Los amplificadores de instrumentación son usados cuando se requiere una gran precisión y estabilidad del circuito tanto a corto como a largo plazo.

El amplificador de instrumentación está compuesto típicamente por 3 amplificadores operacionales, a saber, un *buffer* en cada entrada y un amplificador diferencial en la salida. El circuito comúnmente usado es el que se muestra en la figura 2.24.

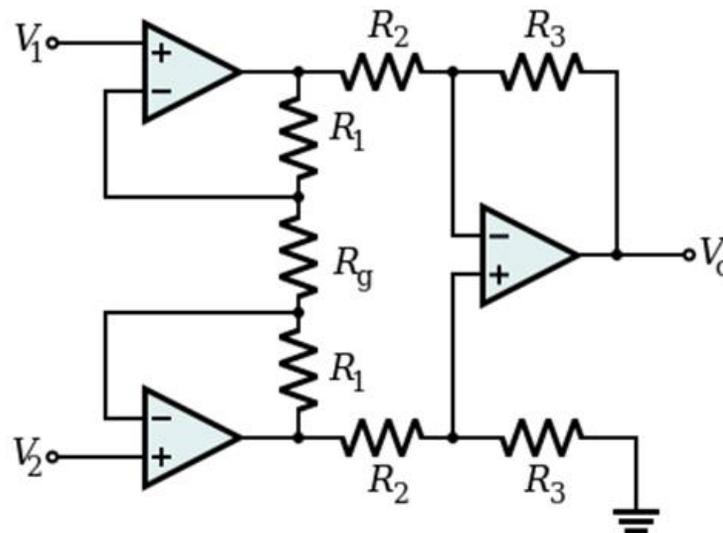


Figura 2.24. Amplificador de instrumentación.

La función de transferencia que relaciona el voltaje de salida  $V_o$  y el voltaje diferencial de entrada  $V_2 - V_1$  es:

$$\frac{V_o}{V_2 - V_1} = \left(1 + \frac{2R_1}{R_g}\right) \frac{R_3}{R_2} \quad (2.26)$$

El amplificador ubicado en la parte derecha de la figura 2.24 es el amplificador diferencial estándar cuya ganancia es  $R_3/R_2$ . Los dos amplificadores a la izquierda son los *buffers*. Removiendo  $R_g$  (circuito abierto) los dos amplificadores operacionales de entrada simplemente tienen ganancia unitaria y la ganancia total únicamente será aquella debida a  $R_3$  y  $R_2$ . La ganancia podría ser incrementada colocando resistencias entre las entradas inversoras de los dos primeros amplificadores operacionales y tierra, sin embargo, un método más elegante es colocando la resistencia  $R_g$ , esto incrementa la ganancia en modo diferencial  $A_{md}$  del par, a la vez que la ganancia en modo común  $A_{mc}$  disminuye, por lo tanto, la razón de rechazo en modo común incrementa; este último parámetro se define como:

$$CMRR = \frac{A_{md}}{A_{mc}} \quad (2.27)$$

El hecho de tener una baja  $A_{mc}$  y una elevada  $A_{md}$  asegura que las señales en modo común, como el ruido eléctrico ambiental acoplado al circuito, no son amplificadas, al mismo tiempo, únicamente las señales en modo diferencial (como las provenientes de los sensores) son amplificadas en gran medida. Al analizar la ecuación (2.27), se observa que el caso ideal es cuando  $A_{md}$  tiende a infinito mientras que  $A_{mc}$  es igual a cero, esto en consecuencia origina una elevada  $CMRR$ .

### 2.3. Conversión analógico-digital

En la actualidad, el procesamiento de señales se hace por medio de microcontroladores que permiten el tratamiento de los datos a grandes velocidades, así como la ejecución de otras tareas. En el medio en el que vivimos, las variables físicas como temperatura, presión y velocidad, por citar algunos ejemplos, son continuas en el tiempo, y para poder medirlas mediante un dispositivo electrónico digital, se necesita realizar una conversión del dominio analógico al digital.

En la figura 2.25 se presenta un diagrama de bloques que ilustra el proceso de conversión analógico-digital. El proceso de conversión analógico-digital cuenta básicamente de tres partes: *muestreo*, *cuantificación* y *codificación*.

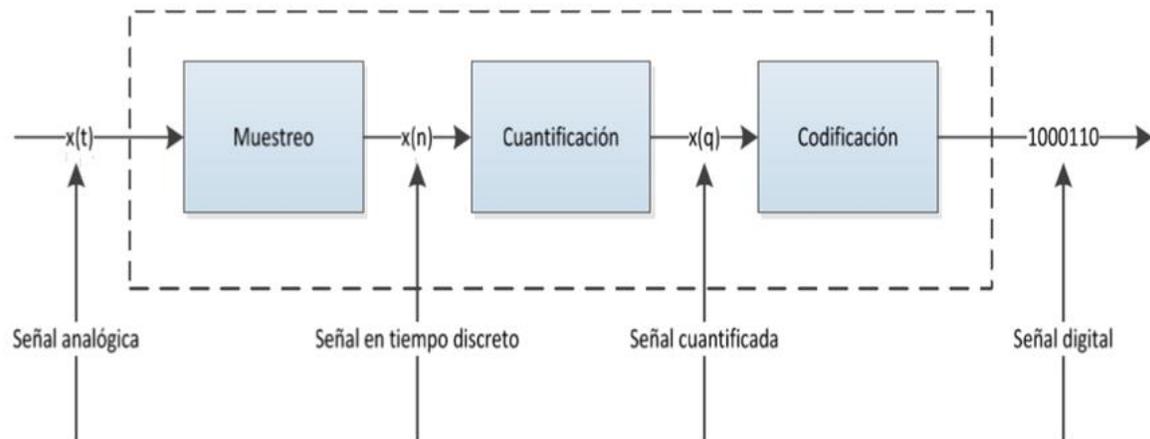


Figura 2.25. Diagrama de bloques del proceso de conversión analógico-digital.

El *muestreo* consiste en tomar muestras periódicas de la amplitud de la onda. La velocidad con que se toman estas muestras, es decir, el número de muestras por segundo, es lo que se conoce como frecuencia de muestreo y está establecida por el teorema de Nyquist, que indica que la frecuencia de muestreo  $F_s$  debe ser por lo menos el doble de la frecuencia máxima  $F_m$  de la señal a muestrear, esto es:

$$F_s \geq 2F_m \quad (2.28)$$

La *cuantificación* es la etapa en la que se convierte una sucesión de muestras de amplitud continua en una sucesión de valores discretos, preestablecidos según el código utilizado. Durante este proceso se mide el nivel de tensión de cada una de las muestras obtenidas en el proceso de muestreo y se les atribuye un valor finito (discreto) de amplitud, seleccionado por aproximación dentro de un margen de niveles previamente fijado. Los valores preestablecidos para ajustar la cuantificación se eligen en función de la propia resolución que utilice el código empleado durante la codificación. Si el nivel obtenido no coincide exactamente con ninguno, se toma como valor el inferior más próximo. En este momento, la señal analógica (que puede tomar cualquier valor) se convierte en una señal discreta, ya que los valores que están preestablecidos son finitos.

La diferencia que existe entre la señal de entrada y la señal cuantificada se conoce como error de cuantificación, esto es, la medida en la que ha sido necesario cambiar el valor de una muestra para igualarlo a su nivel de cuantificación más próximo. Esta diferencia puede ser interpretada en la práctica como una señal indeseada añadida a la señal original, motivo por el cual se le denomina ruido de cuantificación cuando se produce la señal tras el proceso de decodificación digital. La figura 2.26 muestra la gráfica de una señal cuantificada (pendiente escalonada) y una señal analógica (pendiente lineal), además del error de cuantificación introducido en este proceso.

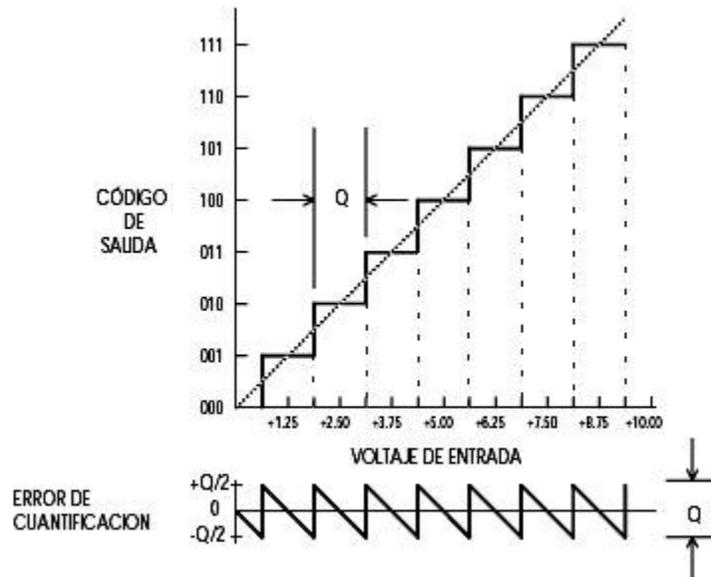


Figura 2.26. Error de cuantificación en un convertidor analógico-digital ideal.

En la figura 2.26 se aprecia que el error máximo se presenta cuando la muestra se encuentra exactamente entre dos niveles de cuantificación. Cada paso de cuantificación  $Q$  tiene el valor de un bit menos significativo, por lo que el error de cuantificación máximo corresponde a  $Q/2$ .

El último proceso en la conversión analógico-digital se denomina *codificación* y consiste en la traducción de la señal ya cuantificada al sistema binario, mediante códigos preestablecidos. Finalmente la señal analógica quedará transformada en un tren de impulsos digitales.

### 2.3.1. Parámetros generales de un convertidor analógico-digital

Existen algunas características que deben de ser consideradas para la correcta elección de un convertidor analógico digital (*ADC, Analog to Digital Converter*), tales como: la resolución, la frecuencia de muestreo, la máxima entrada de escala completa, el rango dinámico, la monotonicidad, la impedancia de entrada, el offset, el tiempo de conversión, el tiempo de adquisición, el tiempo de asentamiento, el *slew rate* (velocidad de respuesta), la precisión y la linealidad. Los parámetros citados se describen a continuación.

- *Resolución*, se entiende como el voltaje necesario (señal analógica) para lograr que en la salida (señal digital) haya un cambio del bit menos significativo (*LSB, Least Significant Bit*). Si  $n$  es el número de bits del ADC y  $V_{ref}$  es el voltaje de referencia, la resolución del ADC está dada por:

$$Resolución = \frac{V_{ref}}{2^n - 1} \quad (2.29)$$

- *Frecuencia de muestreo*, es el número de muestras que se pueden tomar por segundo. Este valor se expresa en [Hz] o en muestras por segundo (mps).
- *Máxima entrada de escala completa*, se refiere a la máxima entrada de voltaje que el dispositivo puede convertir sin que éste se sature.
- *Rango dinámico*, se define como la relación entre el valor más grande y el valor más pequeño que el Convertidor Analógico Digital puede aceptar. Por ejemplo, si el valor más grande es 4 [V] y el más chico es 1 [V], el rango dinámico es  $4/1 = 4$ ; en decibeles,  $20 \log (4/1) = 12$  dB. En algunos convertidores, los bits menos significativos contienen ruido, por lo que el rango dinámico se define como la relación entre la entrada más grande de voltaje y el nivel de ruido del convertidor.
- *Impedancia de entrada*, este parámetro idealmente debe ser tan grande como sea posible para no alterar la señal proveniente del sensor. Un valor típico es 1 [MΩ].
- *Tiempo de asentamiento*, es el intervalo de tiempo entre la retención de la señal y el asentamiento definitivo de la señal.
- *Slew rate*, es la velocidad a la cual el valor de la salida del sistema sample & hold, converge al valor muestreado. El proceso de conversión analógico-digital requiere que la señal analógica de entrada permanezca en un valor constante de tal forma que el ADC pueda realizar su tarea en forma adecuada. Aparece aquí un elemento llamado *sample & hold* (muestreo y retén), que toma una muestra de la señal seleccionada y mantiene su valor durante el tiempo que dura la conversión analógico-digital. El circuito básico de *sample & hold* se muestra en la figura 2.27; este circuito utiliza un capacitor para mantener la tensión a muestrear. El interruptor controlado electrónicamente se utiliza para realizar la carga del capacitor con el voltaje analógico de entrada, posteriormente el interruptor se abre y el capacitor retiene el voltaje de la señal.

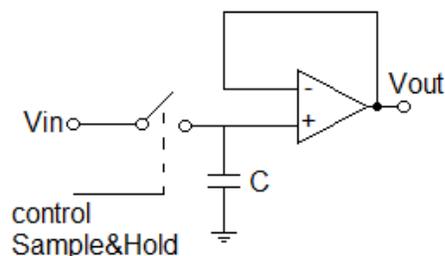


Figura 2.27. Circuito sample &amp; hold.

- *Precisión*, se refiere a la dispersión del conjunto de valores obtenidos de mediciones repetidas de una magnitud por medio del convertidor analógico

digital. En donde cuanto menor sea la dispersión de los datos, mayor será la precisión de la lectura.

- *Offset*, es el nivel de voltaje de DC presente en la salida del convertidor analógico digital, esto debido a la no linealidad de los componentes del ADC. La figura 2.28 muestra la función de transferencia de un ADC ideal y uno real, como se observa, el nivel de voltaje de DC introduce otro tipo de error: el error de offset, éste se caracteriza por asignar un código a cada muestra conformado por el valor real de la muestra más un nivel determinado de voltaje de DC.

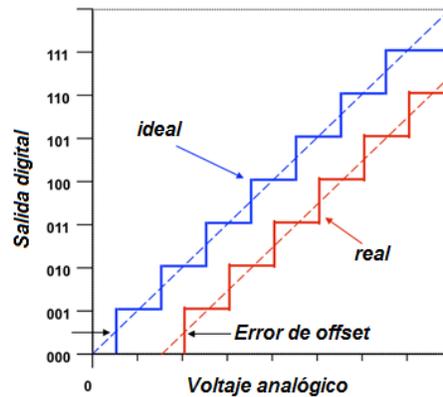


Figura 2.28. Error de offset.

- *Monotonicidad*, se considera que un convertidor es monótonico cuando a un incremento de tensión en la entrada le corresponda un incremento en la salida, y para una disminución de la entrada, se tiene el correspondiente descenso a la salida. Si un convertidor no es monótonico, el resultado es la pérdida del código. En la figura 2.29 se presenta un caso de error de monotonicidad debido a la correspondencia no uniforme entre el aumento de la señal analógica y el correspondiente código digital.

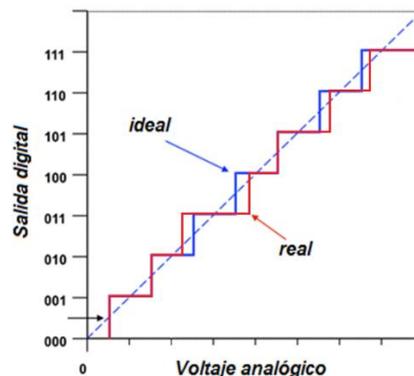


Figura 2.29. Error de monotonicidad.

- *Linealidad*, se refiere a que la señal de salida del dispositivo conservará las características (amplitud y frecuencia) de la señal de entrada. En la figura

2.29 se muestra el error de linealidad introducido por el ADC al no tener un comportamiento lineal durante su conversión.

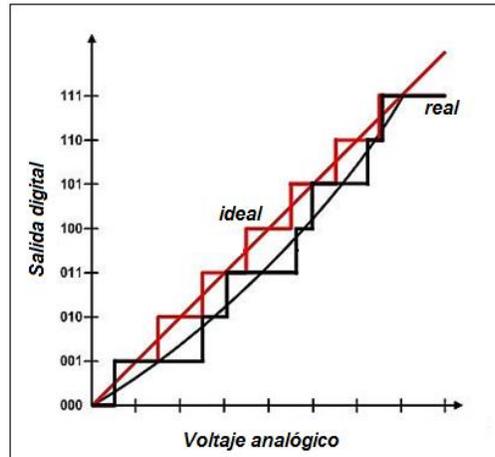


Figura 2.30. Error de linealidad.

### 2.3.2. Tipos de convertidor analógico-digital

Los dispositivos ADC convierten un nivel de tensión analógico en una palabra digital correspondiente. Si  $n$  es el número de bits obtenidos de la palabra, esto significa que habrán  $2^n$  niveles de tensión diferentes. Todo convertidor ADC debe procurar que la palabra digital obtenida a la salida sea un reflejo lo más exacto posible del valor analógico correspondiente.

Se usan un gran número de métodos para convertir señales analógicas a la forma digital, los más usados son: rampa, aproximaciones sucesivas, paralelo (flash), doble rampa y voltaje a frecuencia.

#### Convertidor ADC de rampa

Se basa en la comparación de la señal analógica de entrada con una señal de rampa definida con precisión. El esquema se muestra en la figura 2.31.

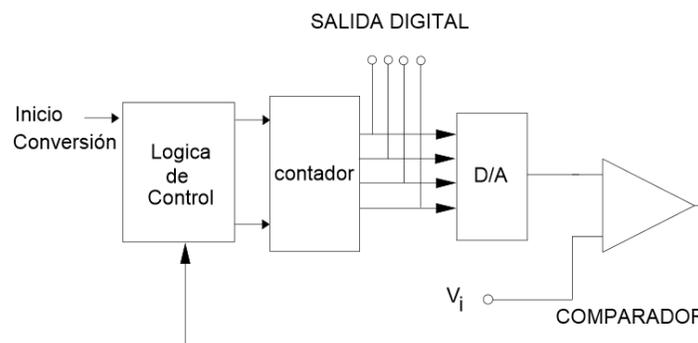


Figura 2.31. ADC de rampa.

El proceso de conversión comienza activando un pulso de inicio en la lógica de control, con esta acción el contador se inicializará en cero, entregando en sus salidas el código binario del cero digital. La secuencia pasa directamente como entrada paralelo al convertidor digital-analógico (*DAC, Digital to Analog Converter*) que corresponde con 0 [V] a la salida. Esta señal es usada como entrada de referencia a un comparador, el cual compara la magnitud de la señal analógica de entrada con el valor entregado por el conversor. Del valor que proporcione el comparador dependerá que el contador continúe contando, o bien, se detenga, pues si el comparador entrega un “1” lógico, entonces el reloj continuará alimentando al contador; de lo contrario, al entregar un “0” lógico, el contador se detendrá.

La lógica del comparador establece que si la señal de entrada es mayor que la referencia, entonces el comparador responderá con un “1” y se incrementa la cuenta digital en 1 digital y así sucesivamente, la cuenta sólo se detendrá cuando la respuesta del DAC sea mayor que la entrada de la señal analógica, en ese caso, el reloj se detendrá y se tendrá la salida digital del valor de cuenta anterior.

#### *Convertidor ADC por aproximaciones sucesivas*

La conversión por aproximaciones sucesivas se utiliza ampliamente debido a su combinación de alta resolución y velocidad. A diferencia del convertidor de rampa, contiene un contador programable que se incrementa o decrementa de acuerdo a la influencia del bit de mayor peso o bit más significativo (*MSB, Most Significant Bit*) dentro del registro de aproximaciones sucesivas (*SAR, Successive Approximation Register*). Con la consideración mencionada, no es necesario contar  $2^n$  veces como lo hacía el contador tipo rampa, ahora la cuenta máxima solo es de  $n$  veces. La figura 2.32 muestra un diagrama a bloques del convertidor de aproximaciones sucesivas.

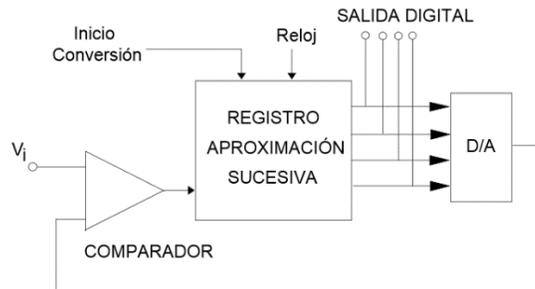


Figura 2.32. Convertidor de aproximaciones sucesivas.

El SAR pone el MSB en “1” y todos los restantes en “0”. La cantidad es tomada por el DAC de tal manera que su equivalente analógico se compara con la señal de entrada. Si la salida del DAC es mayor que la entrada, se elimina el “1” del bit MSB y se pone en “1” el bit inmediatamente anterior, con todos los demás bits en “0”, esto se realiza sucesivamente hasta que se logre encontrar una secuencia analógica pero que resulta ser menor que la entrada de la señal, cuando ocurra esto, el bit mantendrá su valor y se pone a “1” el bit anterior. El procedimiento se repite hasta

terminarse de probar un “1” en cada bit del contador. Lo anterior equivale a un tanteo digital, a medida que se avanza, el procedimiento se va estabilizando hasta llegar a un valor que corresponda con la medición de la señal de entrada. La figura 2.33 muestra las salidas características de este tipo de convertidor.

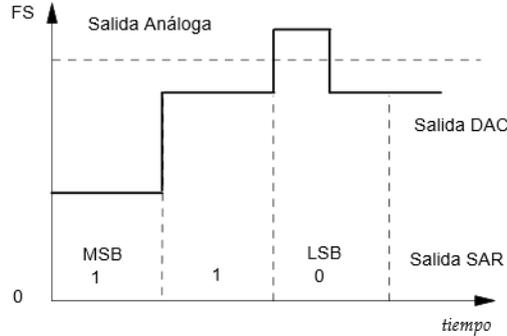


Figura 2.33. Curva de salida del ADC.

El ADC de aproximaciones sucesivas es de los más utilizados, es posible encontrar modelos capaces de suministrar 16 bits en la salida y realizar la conversión en un tiempo de unas decenas de microsegundos. Los modelos de 12 y 8 bits son los más comunes y ofrecen una elevada velocidad a un precio bajo.

*Convertidor ADC digital paralelo (flash)*

Los convertidores de tipo flash o conversión directa parten de una concepción radicalmente opuesta: la velocidad es el objetivo básico de esta arquitectura y el costo que se debe pagar por ello es un circuito muy complejo aunque sencillo a nivel de concepto. Dos señales participan en la etapa de entrada, la propia señal analógica que se debe convertir y una señal de referencia. En la configuración básica, la señal analógica se aplica a las entradas no inversoras de un cierto número de amplificadores operacionales que, utilizados como comparadores, están dispuestos en paralelo a la entrada de un decodificador. Este convertidor se muestra en la figura 2.34.

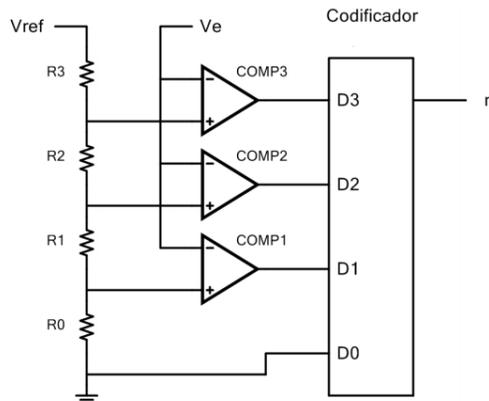


Figura 2.34. Convertidor tipo flash.

A la entrada inversora de cada comparador se aplica la tensión de referencia, que a su vez ataca una red de resistencias de valor idéntico y dispuestas en serie. El resultado es la diferencia de tensión entre dos comparadores sucesivos de un bit menos significativo (LSB, Least Significant Bit). La complejidad de la arquitectura flash se deriva precisamente del elevado número de comparadores necesarios a medida que aumenta el número de bits que se desea obtener a la salida. El número de éstos es  $2^{n-1}$ , donde  $n$  es el número de bits de salida; no es de extrañar que los convertidores de tipo flash vean limitada su resolución por su elevada integración.

El resultado de utilizar un elevado número de comparadores es que no existe ningún convertidor flash que ofrezca una resolución de 16 bits, y que más allá no son prácticos teniendo en cuenta el tamaño del chip, el correcto funcionamiento de los comparadores e incluso el precio. Este tipo de convertidor es utilizado, únicamente debido a la velocidad de conversión, en el campo de las telecomunicaciones, los instrumentos de medida y, en general, en el tratamiento de señales rápidas como las de video.

### Convertidor ADC de doble rampa

Este tipo de convertidores tienen como punto fuerte la precisión, y al mismo tiempo, sólo pueden aplicarse a señales cuyo nivel oscile de forma muy lenta (una tasa de muestreo típica es de 10 muestras por segundo). Estos dispositivos se utilizan en instrumentos de medición como los multímetros digitales. El esquema de un convertidor de doble rampa se muestra en la figura 2.35.

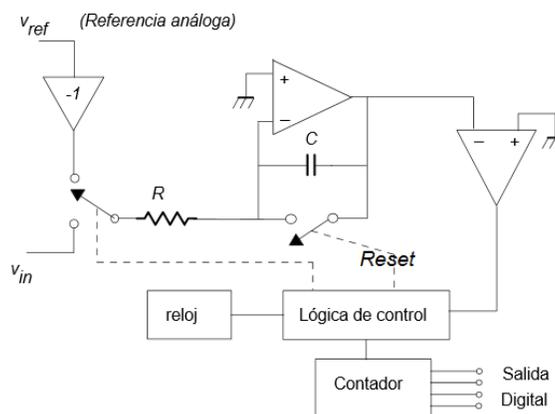


Figura 2.35. Convertidor de doble rampa.

El clásico convertidor de doble rampa está conformado por dos secciones principales: un circuito que adquiere y digitaliza la entrada, produciendo un intervalo de tiempo, y un contador que convierte el resultado en un valor digital.

Se utilizan dos señales, la señal de entrada a digitalizar y un voltaje de referencia negativo. La señal de entrada es integrada durante un tiempo determinado ( $T_{carga}$ ) que usualmente corresponde al tiempo de cuenta máximo del contador, al final de este intervalo de tiempo, el dispositivo reinicia su contador y aplica el voltaje de

referencia negativo para descargar el capacitor hasta obtener una salida igual a cero, lo cual detiene el contador. El comportamiento del dispositivo se muestra en la figura 2.35.

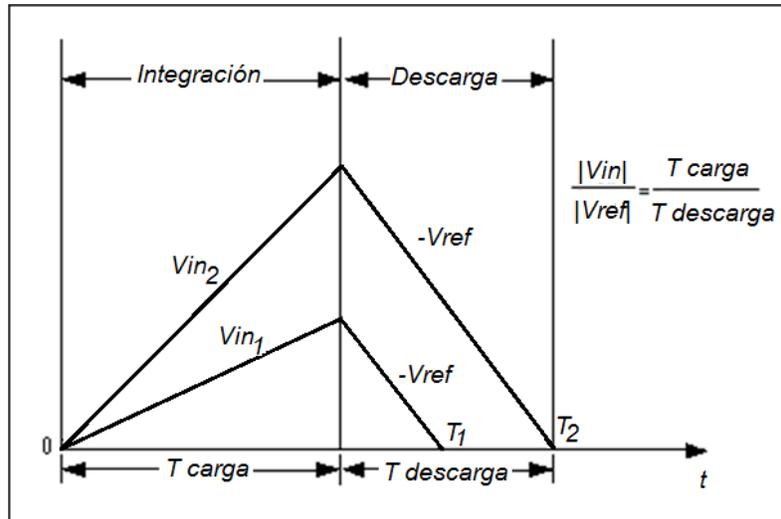


Figura 2.36. Forma de onda del convertidor de doble rampa.

Debido a que la carga ganada por el capacitor durante el tiempo  $T_{carga}$  es la misma perdida durante el tiempo  $T_{descarga}$ , se cumple la siguiente relación:

$$\frac{|V_{in}|}{|V_{ref}|} = \frac{T_{carga}}{T_{descarga}} \quad (2.30)$$

El valor digital correspondiente a la señal de entrada es proporcional al tiempo de descarga del capacitor, este mismo tiempo es aquel durante el cual la lógica de control permite el funcionamiento del contador y de donde se obtiene el código final.

### Convertidor voltaje-frecuencia

En este tipo de convertidores el voltaje continuo de entrada se transforma en un conjunto de pulsos cuya frecuencia es proporcional a la magnitud del voltaje de alimentación. Los pulsos se cuentan mediante un contador electrónico durante un intervalo de tiempo específico y la cuenta resultante se exhibe como una representación digital del voltaje. El esquema es el que se muestra en la figura 2.37.

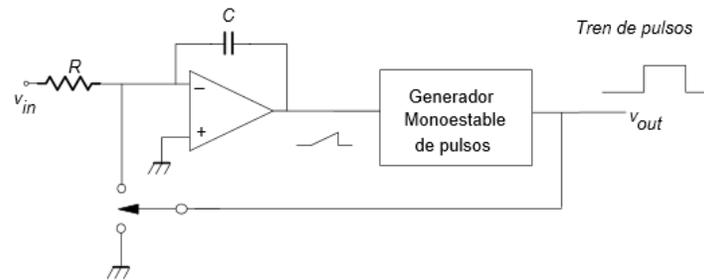


Figura 2.37. Convertidor voltaje-frecuencia.

Cuando se aplica un voltaje de entrada el integrador genera un voltaje de salida de rampa con una pendiente proporcional al voltaje aplicado, esta rampa se aplica a un generador monoestable que genera un pulso de amplitud definido por el voltaje de entrada de rampa. El pulso es realimentado a un conmutador que descarga el condensador integrador, terminando así la rampa.

Su utilización es adecuada en ambientes ruidosos por su alta inmunidad al ruido y exactitud. Las frecuencias típicas del convertidor voltaje-frecuencia están entre 10 [kHz] y 1 [MHz].

### 2.3.3. Aliasing

El teorema de muestreo de Nyquist enuncia que la reconstrucción exacta de una señal periódica continua a partir de sus muestras, es matemáticamente posible si la señal está limitada en banda y la tasa de muestreo es superior al doble de su ancho de banda. Matemáticamente, el teorema establece que si la frecuencia más alta contenida en una señal analógica  $x(t)$  es  $F_m$  y la señal se muestrea a una tasa  $F_s > 2F_m$ <sup>1</sup>, entonces  $x(t)$  se puede recuperar totalmente a partir de sus muestras.

En el proceso de conversión analógico-digital, el fenómeno aliasing se presenta al muestrear una señal analógica a una tasa  $F_s$  menor a  $2F_m$ . Este fenómeno provoca que al reconstruir una señal a partir de las muestras tomadas sea imposible la coincidencia entre la señal reconstruida y la señal original, teniendo entonces una pérdida de información.

A manera de ejemplo, la señal mostrada en la parte superior de la figura 2.38 representa una señal analógica cuya frecuencia es igual a 1 [Hz]. La señal ubicada al centro de esta imagen fue reconstruida a partir de las muestras obtenidas en un proceso de muestreo previo, estableciendo  $F_s = 12$  [Hz]. Como se observa, la señal reconstruida es igual a la señal que la originó, la relación es biunívoca y no existe pérdida de información. Por otro lado, la señal mostrada en la parte inferior de esta imagen fue reconstruida con las muestras tomadas a una tasa  $F_s = 0.66$  [Hz], esta vez la frecuencia es inferior a la indicada por el teorema de Nyquist y por ende la señal reconstruida y la señal original no se corresponden.

<sup>1</sup> Manolakis D., Proakis J., *Digital Signal Processing: Principles, algorithms and applications*, 3a. ed., Prentice-Hall, USA, p.29

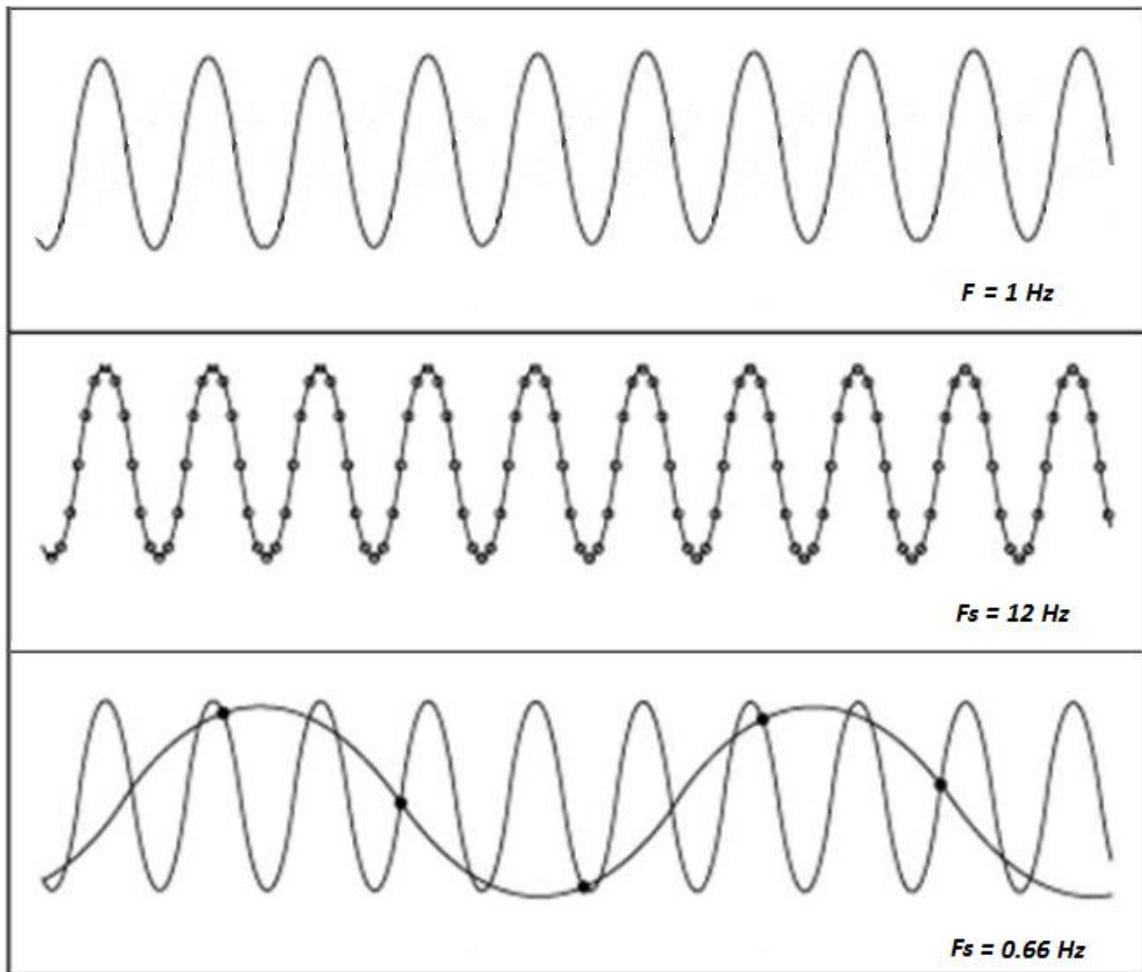


Figura 2.38. Muestreo de una señal analógica.

En el dominio de la frecuencia, el fenómeno de aliasing también puede ser observado. Considerando una señal continua, ésta puede ser muestreada a partir de la ecuación (2.31), donde se observa que la señal discreta  $x[n]$  es igual a la multiplicación de un tren de impulsos periódicos por la señal  $x(t)$  continua.

$$x[n] = \sum_{n=0}^{\infty} \delta[t - nT]x(t) \quad (2.31)$$

Los respectivos espectros se muestran en la figura 2.39. Se puede observar que el espectro de una función muestreada es la repetición periódica del espectro de  $x(t)$ , con un ancho de banda  $F_m$ , el cual si es menor que  $F_s$ , entonces no se presentan traslapos.

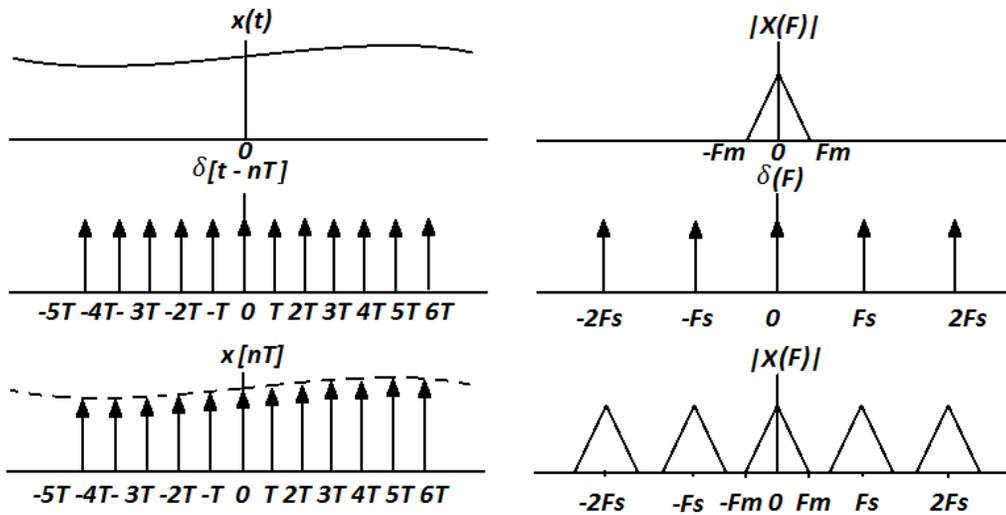


Figura 2.39. Espectros de la señal analógica, el tren de impulsos y la señal muestreada.

A medida que la frecuencia de muestreo se reduce, los espectros se aproximan entre sí. El punto límite para poder reconstruir aún la señal a partir de las muestras es cuando  $F_s = 2F_m$ , en ese momento los espectros se encuentran uno al lado del otro como se observa en el esquema intermedio de la figura 2.40. Al reducir la frecuencia de muestreo por debajo de  $2F_m$  los espectros se traslapan, se tiene pérdida de información y se presenta el fenómeno de aliasing. Esta situación se observa en el último diagrama de la misma figura.

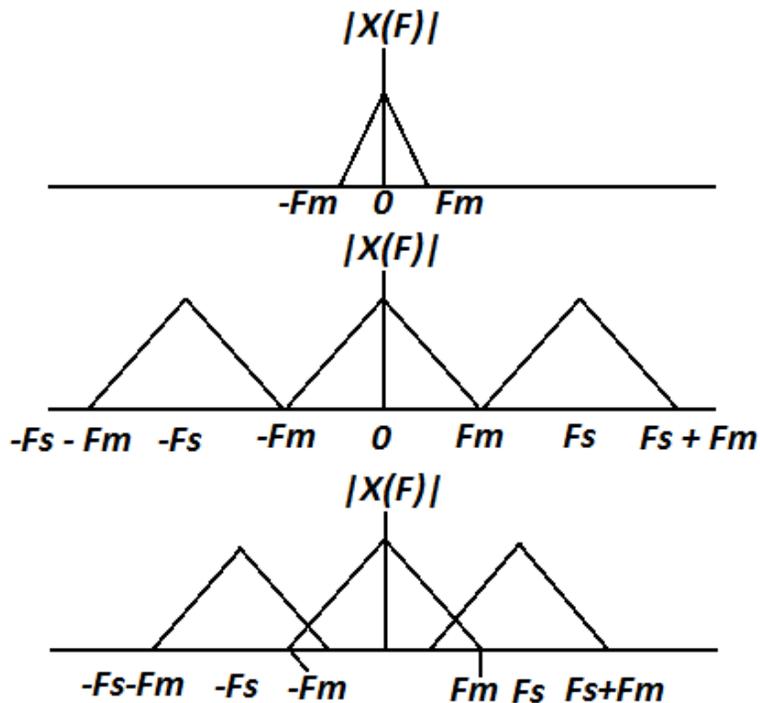


Figura 2.40. Espectros de la señal muestreada.

La manera de realizar una correcta digitalización de una señal analógica y evitar el fenómeno de aliasing, consiste en limitar en banda la señal de interés, en otras palabras, se deben eliminar las componentes de más alta frecuencia mediante filtrado, de manera que la señal reconstruida a partir de las muestras sea una copia fiel de la señal que le dio origen.

#### 2.3.4. Convertidor analógico digital sigma-delta

El convertidor analógico-digital sigma-delta es un dispositivo que opera bajo el principio de sobremuestreo. Los convertidores de sobremuestreo utilizan una frecuencia de muestreo superior a la indicada por el teorema de Nyquist. El éxito de tales convertidores se debe a que un circuito que implementa esta técnica de operación, puede ser realizado usando sistemas de bajo costo del tipo CMOS (*Complementary Metal-Oxide-Semiconductor*), una de las familias lógicas empleadas en la fabricación de circuitos integrados, además de esto, la combinación de sobremuestreo y técnicas de modulación sigma-delta, proporcionan la más grande de sus ventajas: obtención de muy alta resolución y poca sensibilidad frente al ruido de cuantificación.

En particular, los convertidores de sobremuestreo disminuyen los requerimientos en cuanto a circuitería para la limitación en banda de una señal analógica. Este tipo de convertidores realizan un muestreo a alta frecuencia y en consecuencia, el filtro anti-aliasing puede ser implementado mediante un filtro pasivo de primer orden. Para el caso del convertidor analógico-digital sigma-delta, las operaciones críticas de filtrado se trasladan al plano digital donde éstas resultan más robustas e inmunes a las imperfecciones del filtro pasivo.

Como se observa en el espectro de frecuencia de la señal muestreada, ubicado al centro de la figura 2.40, utilizando una frecuencia de muestreo igual a la de Nyquist, el filtro anti-aliasing debe limitar perfectamente en banda la señal de manera que éste no permita el paso de frecuencias superiores a  $F_m$  [Hz]. Si el filtro paso bajas no responde adecuadamente, se presentarían solapamientos y se presentaría aliasing. Utilizando sobremuestreo se tendrá una condición parecida a la mostrada en el extremo inferior derecho de la figura 2.39. Los espectros están más alejados y por lo tanto la transición entre la banda de paso y la banda de bloqueo del filtro puede ser más prolongada.

#### *Modulador delta*

El convertidor analógico-digital sigma-delta utiliza una variación de la modulación delta; la modulación, como es sabido, radica en variar una característica de una onda portadora de acuerdo a una señal que proporciona información. En el modulador delta, la señal analógica de interés es modulada digitalmente con ayuda de un cuantizador de un solo bit, es decir, se tienen únicamente dos estados lógicos: “alto” y “bajo”. La figura 2.41 muestra el diagrama de bloques de la implementación práctica de un modulador delta.

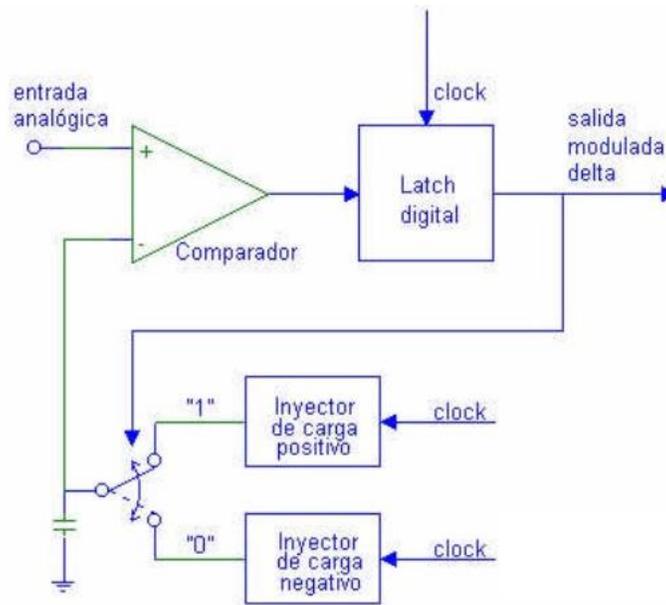


Figura 2.41. Modulador Delta.

Como se observa en la figura anterior, el primer elemento que participa en la modulación delta es un comparador, dicho elemento discierne qué voltaje es mayor: el correspondiente a la señal analógica o el asociado a un capacitor involucrado en el circuito del modulador. El componente siguiente es un *latch*, el cual, recibe el resultado de la comparación y otorga un “1” o “0” como salida digital y, al mismo tiempo, es utilizado para gobernar un conmutador electrónico. Si la señal analógica es superior al voltaje del capacitor, el gobernador electrónico permitirá la inyección de carga positiva a cada ciclo de reloj, de esta manera, el voltaje en el capacitor aumentará progresivamente hasta alcanzar el nivel de la señal analógica. En el momento en que el voltaje del capacitor supere el correspondiente a la señal de entrada, el conmutador electrónico cambiará de posición y permitirá la inyección de carga negativa, disminuyendo de esta manera la tensión en el capacitor y facilitando el seguimiento de la señal analógica. La salida de cada bit se realiza a la frecuencia de reloj del *latch* y es este quien define la tasa de muestreo.

La figura 2.42 muestra las señales asociadas al proceso de modulación delta. Originalmente el voltaje de la señal de entrada y del capacitor pueden ser equivalentes y el conmutador electrónico cambiará de posición para cargar y descargar el capacitor. En el momento que el voltaje de la señal de entrada aumente, el comparador arrojará un “1” y el conmutador permitirá la inyección continua de carga positiva al capacitor, la carga será continua hasta que los voltajes nuevamente sean equivalentes. La pendiente escalonada en la figura, representa un aumento de 1 [V] por cada ciclo de reloj, también se observa que el cambio de nivel de tensión entre la señal analógica y la señal modulada no se realiza a la misma velocidad, esto se debe al *slew rate*, que es un parámetro limitante en la operación de este tipo de convertidores. La pendiente escalonada sigue aumentando hasta alcanzar el equilibrio. En la gráfica mostrada, al permanecer la

señal analógica constante, el voltaje en el capacitor oscila entre 9 y 10 [V] mientras que la salida modulada consiste en una secuencia alterna de “1”s y “0”s.

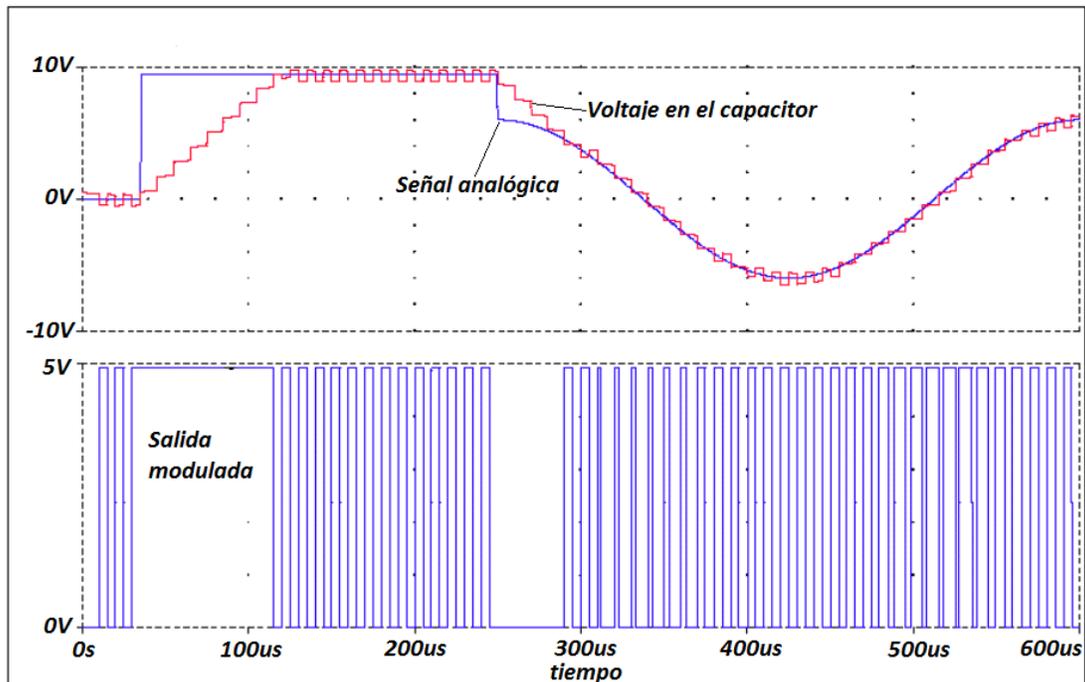


Figura 2.42. Señales correspondientes a la modulación delta.

Se observa que la señal modulada consiste en una cadena de bits que se comportan de acuerdo al cambio presentado en la señal entrante. Si la señal de entrada se incrementa, la salida consiste en una secuencia con más unos que ceros, por el contrario, cuando la señal disminuye, la señal modulada contiene más ceros que unos y, finalmente, cuando la señal analógica se mantiene en un nivel constante, la salida del modulador es una cadena de unos y ceros alternadamente. Del comportamiento de la salida modulada surge el nombre de delta, símbolo utilizado para referirse al cambio en una variable, en muchos casos también para referirse a la pendiente, o razón de cambio de una señal como en este caso.

### *Modulador sigma-delta*

Como se mencionó, el convertidor analógico-digital sigma-delta utiliza una variación del modulador delta. La figura 2.43 muestra el diagrama de bloques de la implementación práctica de este tipo de convertidor, el cual, tiene un comportamiento similar al modulador delta descrito anteriormente, sin embargo, hay dos cambios. En primer lugar el voltaje del capacitor es comparado con tierra en vez de con la señal analógica de entrada, por otro lado, la operación del conmutador electrónico se invierte, permitiendo el suministro de carga negativa al capacitor cuando la salida digital es un uno y suministrando carga positiva cuando la salida es un cero.

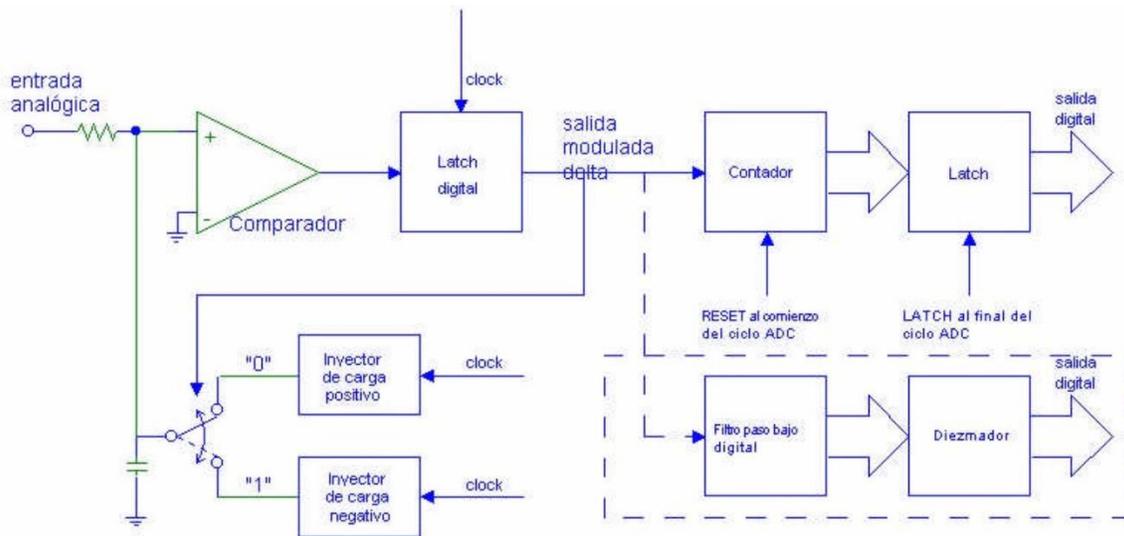


Figura 2.43. Convertidor sigma-delta.

Cuando la señal de entrada es positiva, el comparador arrojará un "1" y mediante un pulso de reloj este valor será transferido por el *latch* al conmutador electrónico. El conmutador, al recibir un "1", permitirá la inyección de carga negativa hacia el capacitor y el voltaje del capacitor disminuirá. En el momento que el voltaje del capacitor se haga negativo, el conmutador aplicará una inyección de carga positiva aumentando el voltaje del capacitor y el bit de salida será un "0".

En la figura 2.44 se observan las señales correspondientes al modulador sigma-delta. El número relativo de unos y ceros está relacionado ahora con el nivel de voltaje de entrada y no con la pendiente como en el circuito anterior. Si la tensión de entrada es positiva, la salida estará compuesta por más unos que ceros. Este exceso de unos es necesario para generar la inyección negativa de carga que compensa la señal positiva de entrada. De la misma forma, si la tensión de entrada es negativa, la señal digital de salida estará compuesta por más ceros que unos, que provocan la inyección de carga positiva. Si la señal de entrada es cero, a la salida se generará un mismo número de unos y ceros.

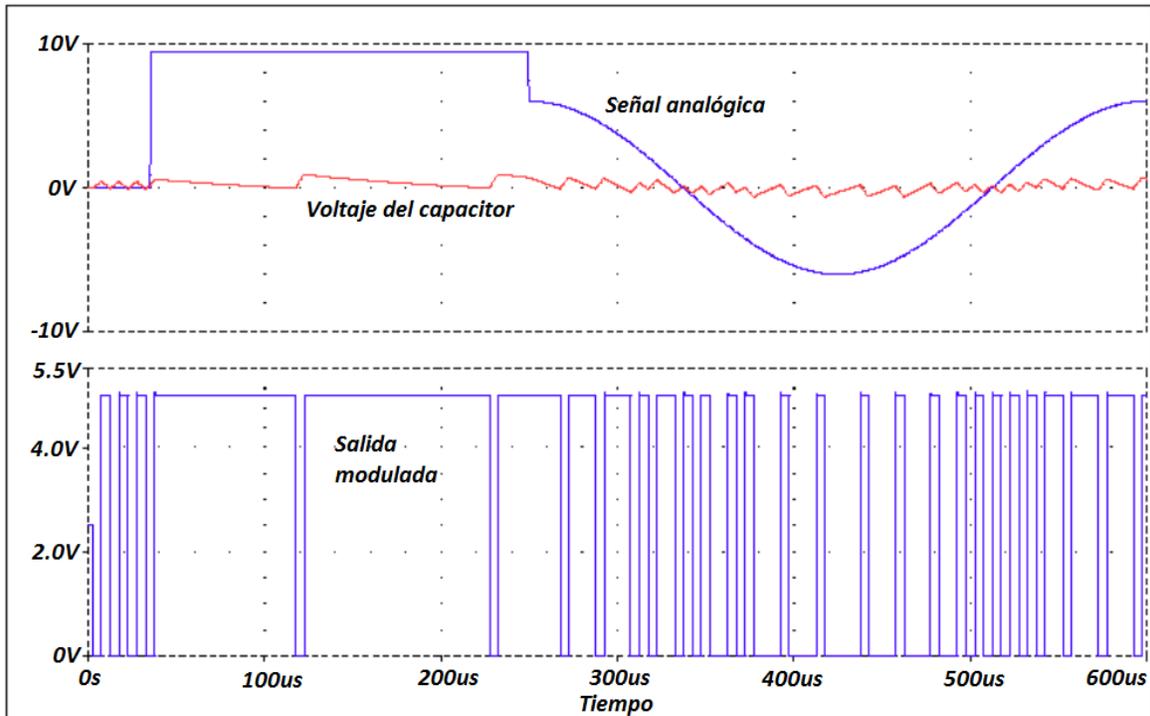


Figura 2.44. Señales correspondientes al ADC sigma-delta.

El hecho de que ahora el número de unos y ceros se relacione con el nivel de voltaje de la señal de entrada, trae como consecuencia una mayor sencillez para realizar la conversión. Un primer método para realizar la conversión es utilizando un contador que utilice la señal modulada como fuente de reloj, de este manera, a medida que el contador reciba más unos, el valor de cuenta será mayor correspondiéndose con una lectura de voltaje elevada. A manera de ejemplo, se puede crear un convertidor analógico-digital de 8 bits, así, el número 256 ( $2^8$ ) corresponderá a la máxima tensión de entrada mientras que el valor 0 corresponderá a la tensión más pequeña de entrada.

Los unos y ceros producidos por el modulador delta también son muy fáciles de convertir en una señal analógica idéntica a la señal de entrada, lo único que se necesita es un filtro RC. Este método de transformar una cadena de bits simples en la forma de onda original es importante por varias razones. En primer lugar, se puede sustituir el contador en el convertidor sigma-delta y en lugar de contar simplemente los pulsos del modulador, la señal digital se pasa por un filtro digital y luego se diezma para reducir la velocidad de muestreo.

#### *Modelado de ruido (Noise shaping)*

Con la finalidad de comprender una de las utilidades (alta resolución e inmunidad al ruido de cuantificación) más grandes que proporciona un convertidor analógico-digital sigma-delta, se obtendrá la función de transferencia del convertidor, partiendo del diagrama de bloques de la modulación delta.

Un mecanismo para digitalizar una señal analógica es usando un modulador delta seguido por un demodulador delta. El diagrama de bloques de la implementación de este mecanismo se muestra en la figura 2.45. Para reconstruir la señal analógica a partir de la señal modulada es necesario un demodulador, el cual, está conformado por un integrador que actúa como decodificador y un filtro pasa bajas para eliminar el ruido de cuantificación.

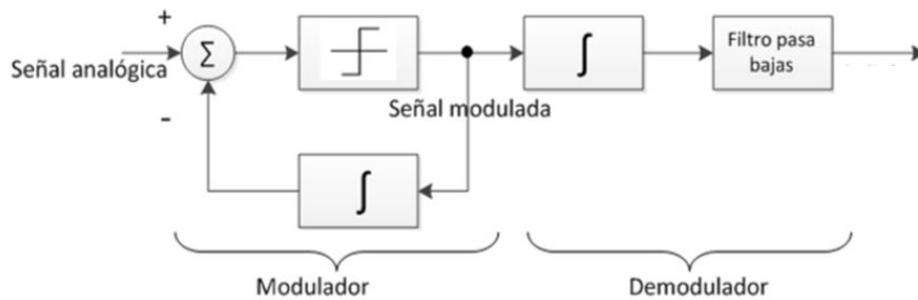


Figura 2.45. Modulador y demodulador delta.

Dado que la integración es una operación lineal (cumple con las propiedades de aditividad y homogeneidad), este operador se puede reubicar en una posición anterior a la del modulador delta sin alterar las características generales de entrada y salida. La reubicación del bloque integrador se muestra en la figura 2.46.

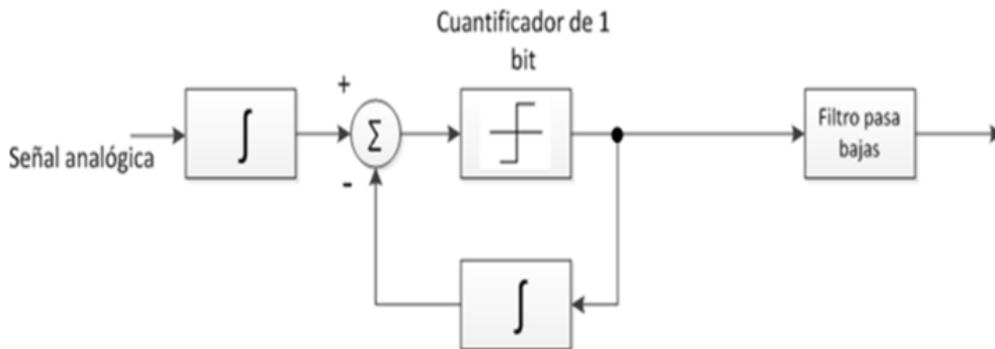


Figura 2.46. Reubicación del bloque integrador.

De la figura anterior se observa que el diagrama de bloques puede ser simplificado si la operación de integración se coloca frente al punto suma. Esta acción da como resultado el diagrama de bloques del convertidor analógico-digital sigma-delta mostrado en la figura 2.47.

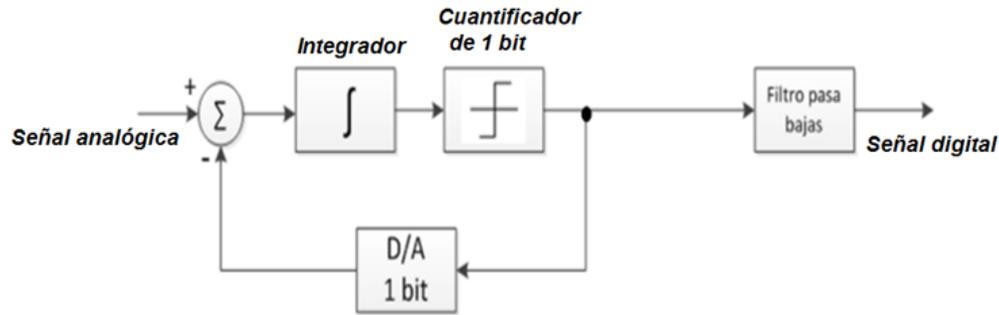


Figura 2.47. Diagrama de bloques del modulador sigma-delta.

A partir de la figura 2.47 se puede obtener el diagrama de bloques del modulador sigma-delta en el dominio de Laplace. Este diagrama se muestra en la figura 2.48.



Figura 2.48. Modulador sigma-delta en el dominio de Laplace.

En el diagrama de la figura anterior el cuantificador es sustituido por un punto suma al que se le agrega el ruido de cuantificación  $N(S)$ . El integrador es sustituido por su equivalente en el dominio de Laplace y el convertidor analógico-digital de 1 bit puede omitirse ya que éste únicamente sirve como interfaz de la salida modulada digitalmente hacia el punto suma de comparación analógica. A partir de la figura 2.48 se puede obtener la función de transferencia tanto para la entrada analógica  $X(S)$  como para el ruido de cuantificación  $N(S)$ . Para la entrada analógica  $X(S)$ , con  $N(S) = 0$  y  $X(S) \neq 0$ , se tiene:

$$Y(S) = \frac{1}{s} [X(S) - Y(S)] \quad (2.32)$$

$$\frac{Y(S)}{X(S)} = \frac{1}{s+1} \quad (2.33)$$

Como se observa, el modulador sigma-delta se comporta como un filtro pasa bajas ante la entrada analógica  $X(S)$ .

Por otra parte, al hacer  $X(S) = 0$  y  $N(S) \neq 0$ , de la figura 2.48 se obtiene la salida  $Y(S)$  en función del ruido de cuantificación  $N(S)$  de la siguiente manera:

$$Y(S) = -Y(S) \frac{1}{s} + N(S) \quad (2.34)$$

$$\frac{Y(S)}{N(S)} = \frac{S}{S+1} \quad (2.35)$$

La función de transferencia obtenida corresponde a la función de transferencia de un filtro pasa altas.

Del análisis anterior se puede concluir que el modulador sigma-delta se comporta como un filtro pasa bajas para la señal de entrada y como un filtro pasa altas para el ruido de cuantificación, gracias a la utilización del bloque integrador. La unión del bloque integrador y la técnica de sobremuestreo permite que la magnitud del ruido de cuantificación sólo sea apreciable a frecuencias tan altas como la frecuencia de sobremuestreo mientras que en la banda base (baja frecuencia) el ruido de cuantificación es imperceptible. Esta técnica se denomina *noise shaping* o modelado de ruido.

Conectando en cascada más de un modulador sigma-delta es posible disminuir aún más el ruido de cuantificación en la banda base. El funcionamiento de los moduladores de orden superior es exactamente el mismo que el descrito para el modulador de primer orden, sólo que la señal que pasa al siguiente modulador será el término de error del modulador actual, siendo este error la diferencia entre la salida del integrador y la salida del cuantificador. Por lo tanto, los moduladores de orden superior, llevan a cabo un mayor número de operaciones de diferencia del error producido por el cuantificador, y como consecuencia de esto se tendrá una mayor atenuación a bajas frecuencias del ruido de cuantificación, eliminando más ruido en la banda base. El diagrama de bloques de un modulador sigma-delta de tercer orden se muestra en la figura 2.49.

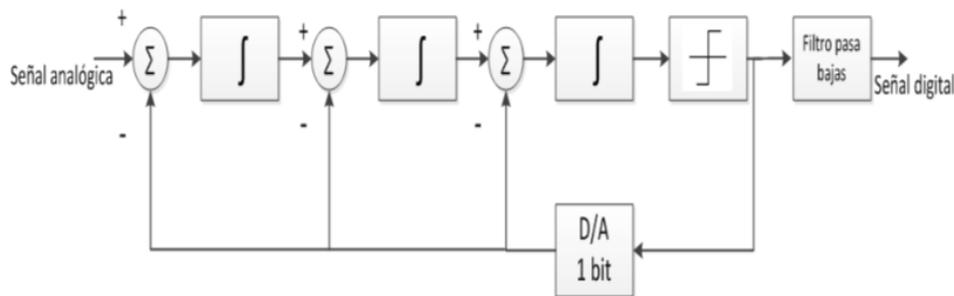


Figura 2.49. Modulador sigma-delta de tercer orden.

En la figura 2.50 se observa el comportamiento del ruido de cuantificación a medida que aumenta la frecuencia; se muestran tres moduladores sigma-delta de distinto orden. A frecuencias cercanas a la frecuencia de sobremuestreo  $f_s$  (altas frecuencias), el ruido de cuantificación es mayor para un modulador sigma-delta de tercer orden en comparación con uno de primer orden, por otro lado, a frecuencias cercanas a la frecuencia de conversión de datos  $f_D$  (bajas frecuencias) del convertidor analógico-digital sigma-delta, el ruido de cuantificación es menor para un modulador de tercer orden en comparación con uno de primer orden.

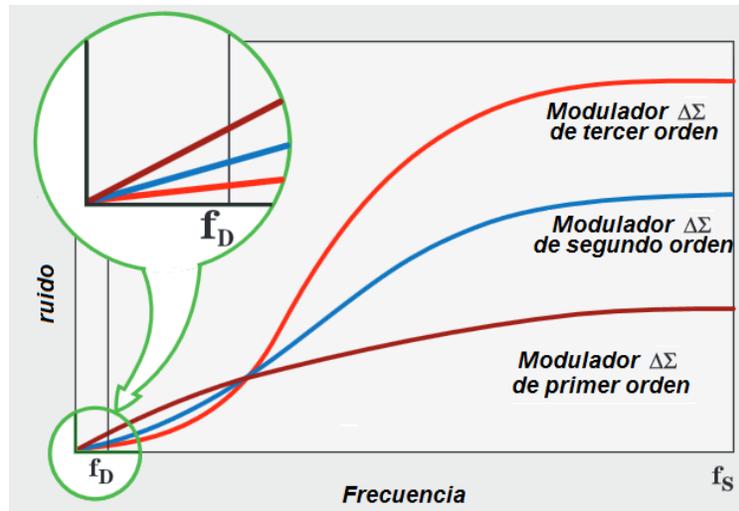


Figura 2.50. Comportamiento de distintos moduladores sigma-delta.

La utilización de un modulador sigma-delta de orden superior trae consigo una disminución del ruido en la banda base y, por lo tanto, la capacidad para medir variaciones de voltaje más pequeñas.

### *Bloque digital*

Una vez que la señal se obtiene del modulador sigma-delta, esta reside en el dominio digital y mediante el uso de un filtro se puede obtener una secuencia de bits relacionados con la señal de entrada. También se realiza un segundo tratamiento a la señal el cual consiste en eliminar la redundancia de datos ocasionada por la alta tasa de muestreo. Los dos elementos dentro del bloque digital son un filtro y un *diezmador*, sin embargo, a menudo suele incorporarse un solo elemento que realiza las dos funciones. El filtro y el *diezmador* se describen a continuación.

### *Filtro digital*

En la primer etapa del bloque digital se implementa un filtro pasa bajos en el dominio discreto, su función radica en permitir el paso de la señal y suprimir el ruido de cuantificación; la entrada al filtro la constituye una cadena de dígitos binarios. A la salida del filtro discreto se tiene una versión digitalizada de la señal analógica de entrada al convertidor, siendo la tasa de salida de datos del filtro, la misma que la tasa de muestreo  $f_s$ . En teoría, la señal digitalizada obtenida después de esta etapa ya puede ser manejada por dispositivos electrónicos digitales, sin embargo, existen dos razones importantes por las que esto no es posible en la realidad:

1. El tratamiento de la señal digital requeriría un procesador muy rápido.
2. Existe una redundancia de datos y la mayoría de ellos no proporcionan una información útil.

Es debido a estas dos razones que después del filtrado digital prosigue una segunda etapa encargada de eliminar aquellos datos que no son útiles.

*Diezmador*

La segunda función del bloque digital es el diezmado. La palabra “diezmar” fue originalmente usada por el ejército romano para referirse a la acción de asesinar a 10 hombres de cada grupo culpable de motín. En el caso del diezmado digital, la eliminación de muestras es radicalmente mayor. En el circuito diezmador, el número de datos de la señal digitalizada es reducido al eliminar una gran porción de ellos, descartando las muestras innecesarias.

Previamente la señal digitalizada se conformaba de valores muy cercanos entre sí que parecían reconstruir perfectamente a la señal analógica original; después de eliminar el mayor número de muestras, la señal parecería ser pobre y haber perdido información, sin embargo, de acuerdo al teorema de Nyquist, bastan solo las muestras necesarias para poder reconstruir una señal a partir de ellas. De hecho, estas muestras contienen exactamente la misma información que el conjunto previo, con la diferencia de ocupar menos espacio en memoria a la hora de ser almacenadas. La figura 2.51 ejemplifica el uso del diezmador para reducir la tasa de sobremuestreo.

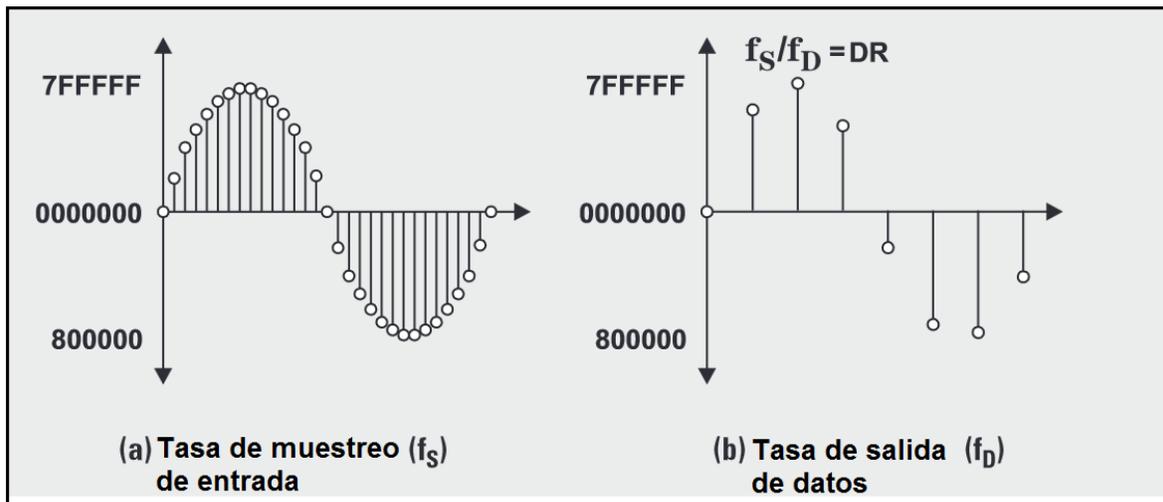


Figura 2.51. Conceptualización de la etapa de diezmado.

Las dos variables importantes utilizadas en este tipo de ADC son la frecuencia o tasa de muestreo  $f_s$  y la tasa de salida de datos del filtro/diezmador  $f_D$ . La relación entre ellas se denomina tasa de diezmado ( $DR$ , *Decimation Rate*) y se define de la siguiente manera:

$$DR = \frac{f_s}{f_D} \tag{2.36}$$

La tasa de diezmado es igual al número de muestras del modulador por dato de salida. El rango típico de la tasa de diezmado para algunos de los ADCs sigma-delta comerciales se encuentra entre 64 y 32768. Algunos convertidores permiten al diseñador programar la tasa de salida de datos mediante el ajuste de la tasa de diezmado.

La elección de la tasa de salida de datos afecta directamente a la resolución que el convertidor analógico-digital puede proporcionar. Tomando como referencia la figura 2.52, puede observarse que al elegir una baja tasa de salida de datos se tiene una gran relación señal a ruido: el ruido de cuantificación en la banda base es mínimo y los bits están libres de ruido de cuantificación. Por otro lado, al elegir una elevada tasa de salida de datos y aproximarse a la frecuencia de sobremuestreo, el ruido de cuantificación es mayor y el nivel de ruido está por encima de los bits menos significativos, esta situación conlleva a que los bits libres de ruido o el número efectivo de bits, ENOB (*Effective Number Of Bits*), se vea reducido.

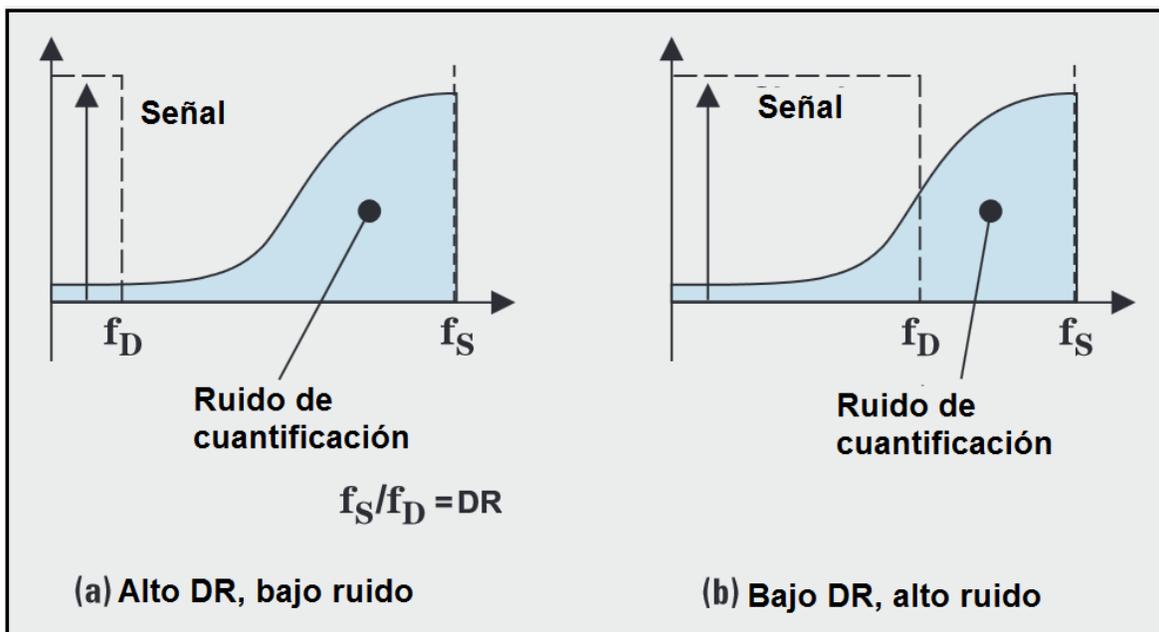


Figura 2.52. Elección de la tasa de diezmado.

Para elegir la tasa de salida de datos correcta, se debe tomar en cuenta que la relación entre la tasa de salida de datos  $f_D$  y la tasa de diezmado  $DR$  es inversamente proporcional, por lo tanto, al elegir una alta  $DR$ , el ruido de cuantificación es bajo, mientras que al elegir una baja  $DR$ , el ruido de cuantificación es alto.

#### 2.4. Protocolos de comunicación serial para dispositivos electrónicos

Los dispositivos electrónicos como los microcontroladores utilizan estándares de comunicación para intercambiar información, ya sea con otros microcontroladores

o con dispositivos que admitan el mismo tipo de comunicación. Un protocolo de comunicación es un conjunto de normas que permite a dos o más entidades de un sistema de comunicación, transmitir información mediante algún medio físico, generalmente un conjunto de conductores eléctricos denominado bus de comunicación. El protocolo de comunicación se implementa mediante un hardware especializado que permite la generación de las señales eléctricas y el software encargado de proporcionar la información codificada en un formato preestablecido.

La comunicación serial se utiliza en aquellos casos en donde el costo de utilizar múltiples conductores hacen impráctica la comunicación en paralelo. La comunicación serial es el proceso de envío de datos de un bit a la vez sobre el bus de comunicación. La principal ventaja entre la comunicación serial con respecto a la comunicación paralelo, radica en que la primera requiere de un número más pequeño de líneas de transmisión de datos, sumado a esto, la evolución en el desarrollo de circuitos integrados permite alcanzar tasas de transmisión de datos lo suficientemente altas para realizar la comunicación entre dispositivos a una velocidad adecuada.

Algunos de los protocolos de comunicación que utilizan los microcontroladores más populares en el mercado, son la comunicación serial asíncrona mediante UART (*Universal Asynchronous Receiver-Transmitter*), el estándar SPI (*Serial Peripheral Interface*), I2C (*Inter-Integrated Circuit*), entre otros. A continuación se describirán estos protocolos.

#### 2.4.1. Comunicación serial asíncrona mediante UART

La diferencia principal entre una interfaz síncrona (como SPI o I2C) y una asíncrona (como UART), es la forma en la que la información es pasada del transmisor al receptor. Los periféricos síncronos necesitan una línea física dedicada a la señal de reloj, proporcionando la sincronización entre dispositivos. Los periféricos asíncronos, por el contrario, no usan una señal de reloj. Se agregan bits de inicio y de parada, además de un formato específico a una tasa de transferencia fija.

##### *UART*

El módulo UART es un elemento de hardware que controla los puertos y dispositivos seriales, permite comunicación serial asíncrona con otros dispositivos o componentes tales como computadoras, convertidores analógico-digitales, módems, entre otros. Los elementos más importantes que constituyen un módulo UART son el registro de transmisión, el registro de recepción y un generador de la tasa de transmisión de datos o *baud rate*. El diagrama de bloques de un módulo UART se muestra en la figura 2.53.

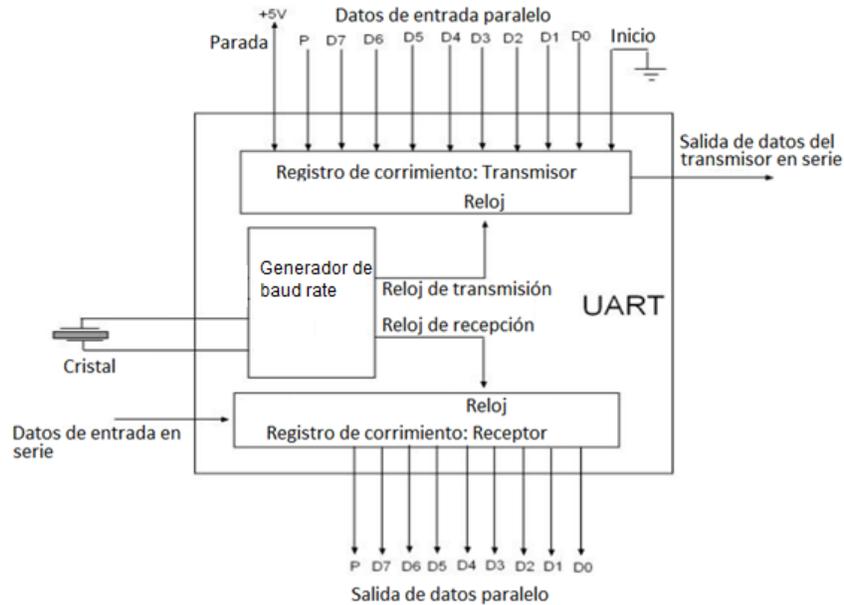


Figura 2.53. Módulo UART.

Tanto el registro de transmisión como el registro de recepción son propiamente registros de corrimiento. Un registro de corrimiento es un arreglo de *flip-flops* colocados uno enseguida del otro, de esta manera, la salida del *flip-flop* anterior representa la entrada del elemento siguiente. El arreglo permite que el bit de entrada al primer *flip-flop* sea recorrido hasta llegar al último elemento con cada pulso de reloj.

Un registro de corrimiento puede convertir un dato de entrada de tipo paralelo a un formato de tipo serial para su transmisión mediante un solo conductor hacia otro dispositivo electrónico. El elemento que incluye dos registros de corrimiento y la lógica de control es un módulo UART. Un módulo UART incluye un registro de corrimiento de entrada de datos serial a salida de datos paralelo (Receptor) y otro registro de corrimiento de entrada de datos paralelo a salida de datos serial (Transmisor).

El proceso para generar la tasa de transferencia de datos es el siguiente: el microprocesador recibe una señal de reloj desde una fuente externa como un oscilador, a su vez, el microcontrolador produce una señal de reloj de entrada al módulo UART a una frecuencia programada. El módulo UART contiene un generador de *baud rate* programable que toma una señal de entrada y aplica un divisor con valores que van desde 1 hasta  $(2^{16} - 1)$  para producir la fuente de reloj BCLK. La frecuencia de BCLK es 16 veces el *baud rate*.

Para configurar el valor del divisor del generador de *baud rate* se utilizan dos registros de 8 bits (DLH y DLL) denominados *latch's de división*. DLH (*Division Latch High*) corresponde a los bits más significativos mientras que DLL (*Division Latch Low*) corresponde a los bits menos significativos del valor cargado a estos registros.

El esquema del proceso de generación de la tasa de transferencia de datos se ilustra en la figura 2.54.

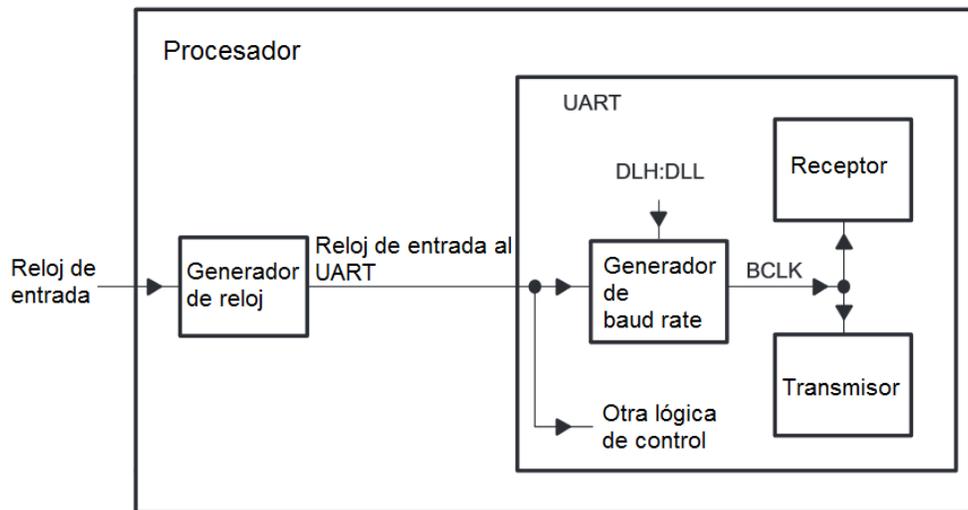


Figura 2.54. Generador de la tasa de transferencia de datos.

La fórmula para obtener el valor del divisor cargado a los registros DLH y DLL se obtiene a partir de la ecuación 2.37.

$$Divisor = \frac{\text{Frecuencia del reloj de entrada al UART}}{\text{Baud rate deseado} \times 16} \quad (2.37)$$

Normalmente la fuente de reloj de entrada al módulo UART es la misma que utiliza el microcontrolador para su operación, es decir, un oscilador que incluye un cristal de cuarzo. El reloj se elige de tal manera que se puedan obtener tasas de transmisión de datos de 50, 300, 600, 2400, 4800, 9600, 19200, 38400, 57600 y 115200.

Debido a que la comunicación mediante UART es un método de comunicación asíncrona, la cadena de datos de información suele incluir bits de inicio y de parada para indicar el inicio y el final de la transmisión de un dato, más allá de utilizar una señal de reloj para la sincronización de los datos. De esta manera, ambos dispositivos (el dispositivo transmisor y el dispositivo receptor) deben configurarse para tener la misma tasa de transferencia de datos. Una gran desviación de este parámetro causará que tanto el dispositivo transmisor como el dispositivo receptor muestreen el bit incorrecto y por lo tanto, la información sea errónea. Por ejemplo, si un UART es programado a una tasa de 9600 bits/s utilizando un reloj de 1.8432 [MHz] y se comunica con otro dispositivo configurado a la misma tasa de transmisión de datos, sin embargo, su reloj opera a una frecuencia de 1.8063 [MHz] (-2%), la tasa de transferencia de datos asociada a esta fuente de reloj es (utilizando la ecuación 2.37) 9408 bits/s. En tal caso, la información enviada por el transmisor será distinta a la recibida por el receptor.

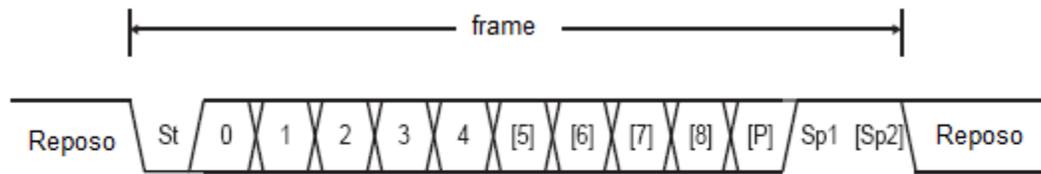
Un método que la mayoría de los módulos UART utilizan para disminuir la desviación de la tasa de transferencia de datos, es sobremuestrear el dato entrante a 16 veces el *baud rate*, en otras palabras, cada bit de información se detecta durante 16 ciclos de reloj de BCLK. Este método permite que la variación de la frecuencia de las fuentes de reloj de los dispositivos, sea de hasta un  $\pm 5\%$  del valor estrictamente esperado.

### *Protocolo serial RS-232*

Un uso común que se le da al módulo UART es comunicar a un microcontrolador con una computadora usando el puerto serial y el protocolo RS232. Este estándar define las características eléctricas de las señales, los parámetros de tiempo, su nomenclatura, las dimensiones físicas y la distribución de terminales de los conectores.

En años precedentes las computadoras contaban con el puerto serial y a pesar de que muchos sistemas modernos lo han abandonado completamente y adoptado conexiones por USB (*Universal Serial Bus*), el puerto serial sigue siendo usado en la industria, en los equipos de instrumentación y en módulos embebidos, debido a su sencillez en comparación con protocolos seriales modernos como USB. En el protocolo RS-232, el dispositivo a conectar es llamado DCE (*Data Communication Equipment*) y la computadora es llamada DTE (*Data Terminal Equipment*), esta nomenclatura fue introducida en 1962 por *Radio Sector* de EIA (*Electronic Industries Alliance*) en la estandarización de comunicación entre teletipos y módems.

En el protocolo RS-232, antes de cada byte de información, se envía un bit de inicio, el cual es un bit con valor de 0, después se envía la palabra digital (5, 6, 7 u 8 bits) y finalmente se envía un bit de parada para indicar que el byte ha sido completado. Algunas veces también se manda un bit de paridad para verificación de errores. Un bit de paridad es un dígito binario que indica si el número de bits con un valor de 1 en un conjunto de bits es par o impar. Los bits de paridad conforman el método de detección de errores más simple. Existen dos variantes de este método, bit de paridad par y bit de paridad impar. En el caso de la paridad par, se cuenta el número de unos. Si el total es impar, el bit de paridad se establece en uno y por tanto la suma del total anterior con este bit de paridad, daría par. Si el conteo de bits uno es par, entonces el bit de paridad (par) se deja en 0, pues ya es par. En el caso de la paridad impar, la situación es la contraria. El formato de la cadena de datos (*frame*) transmitido mediante el protocolo RS232 se muestra en la figura 2.55.



- St** Start bit: Bit de inicio (bajo)
- (n)** Bits de datos (0 a 8)
- P** Bit de paridad
- Sp** Stop bit: Bit de parada (alto)

Figura 2.55. Formato de la cadena de datos.

El bus de comunicación está conformado por una línea de transmisión (TX) y una línea de recepción (RX). Cuando el bus se encuentra libre, es decir, no se está transmitiendo ningún dato, ambas líneas se mantienen en un estado de *reposo*, típicamente un valor lógico alto. Para iniciar la comunicación, el primer bit enviado es el de inicio, a continuación, se envía la palabra digital correspondiente a la información intercambiada entre ambos dispositivos. En el caso de trabajar con bit de paridad, se envía este bit y en seguida se envía el bit de parada. El estándar RS232 permite al usuario indicar si se trabajará con bit de paridad y si se contará con uno o dos bits de parada. Después del envío del o los bits de parada, el bus regresa al estado de *reposo* a la espera de un nuevo dato.

### *El conector serial*

El conector externo para un puerto serial puede ser de 9 o de 25 terminales (figura 2.56). Originalmente el uso de un puerto serial era para conectar un modem a la computadora y la asignación de las terminales refleja este hecho. La configuración de terminales se muestra en las tablas 2.1 y 2.2.



Figura 2.56. Conector DB9 y DB25.

<b>PIN</b>	<b>ABREV.</b>	<b>NOMBRE</b>	<b>FUNCIONAMIENTO</b>
1	CD	Carrier Detect	Determina si el modem está conectado a una línea telefónica en funcionamiento.
2	RD	Receive Data	La computadora recibe la información enviada por el modem.
3	TD	Transmit Data	La computadora envía información al modem.
4	DTR	Data Terminal Ready	Esta señal le dice al MODEM que el UART está listo para establecer una conexión.
5	SG	Signal Ground	Tierra
6	DSR	Data Set Ready	Esta señal le dice al UART que el MODEM está listo para establecer una conexión.
7	RTS	Request To Send	Esta línea le informa al MODEM que el UART está listo para intercambiar datos.
8	CTS	Clear To Send	Esta línea indica que el MODEM está listo para intercambiar datos.
9	RI	Ring Indicator	Una vez que una llamada ha tomado lugar, la computadora reconoce por esta señal (enviada por el modem) que una llamada es detectada.

Tabla 2.1. Conector de 9 terminales.

<b>PIN</b>	<b>ABREV.</b>	<b>NOMBRE</b>	<b>FUNCIONAMIENTO</b>
1		No utilizado	
2	TD	Transmit Data	La computadora envía información al modem.
3	RD	Receive Data	La computadora recibe la información enviada por el modem.
4	RTS	Request To Send	Esta línea le informa al MODEM que el UART está listo para intercambiar datos.
5	CTS	Clear To Send	Esta línea indica que el MODEM está listo para intercambiar datos.
6	DSR	Data Set Ready	Esta señal le dice al UART que el MODEM está listo para establecer una conexión.
7	SG	Signal Ground	Tierra
8	CD	Carrier Detect	Determina si el modem está conectado a una línea telefónica en funcionamiento.
9-19		No utilizado	
20	DTR	Data Terminal Ready	Esta señal le dice al MODEM que el UART está listo para establecer una conexión.
21		No utilizado	
22	RI	Ring Indicator	Una vez que una llamada ha tomado lugar, la computadora reconoce por esta señal (enviada por el modem) que una llamada es detectada.
23-25		No utilizado	

Tabla 2.2. Conector de 25 terminales.

En la conexión típica, es decir, entre un DTE y un DCE, se tiene un control sobre el flujo de intercambio de datos y se utilizan todas las terminales, sin embargo, existe también otro tipo de conexión denominada *conexión de modem nulo* utilizada para conectar dos DTE. Este tipo de conexión proporciona una manera económica de

conectar dos computadoras, o bien, una computadora y un microcontrolador de un sistema electrónico. La conexión de módem nulo únicamente requiere el uso de las terminales TD, RD y SG. En este modo de conexión la terminal TD de una computadora está conectada con la terminal RD de la otra, la terminal DTR está interconectado con la terminal DSR y con CD en ambos extremos. Las terminales RTS y CTS por su parte, dejan de ser importantes pues ambas computadoras se comunican a la misma velocidad, el control de flujo no es necesario de tal forma que estas dos terminales están también interconectadas. La figura 2.57 muestra el esquema de la conexión de módem nulo.

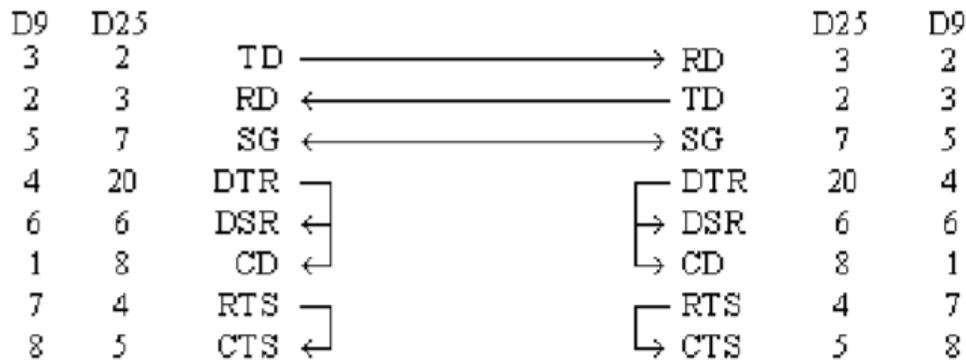


Figura 2.57. Conexión de módem nulo.

En cuanto a las especificaciones eléctricas, se definen las siguientes:

*Rango de voltaje de la señal transmitida*

“0” lógico: +5 a +15 [V]

“1” lógico: -5 a -15 [V]

*Rango de voltaje de la señal recibida*

“0” lógico: +3 a +13 [V]

“1” lógico: -3 a -13 [V]

*Formato de datos*

*Bit de inicio: “0”*

*Bit de parada: “1”*

*Paridad: par, impar o ninguna*

*Bits de parada: 1 o 2*

*Distancia máxima de los conductores: 50 [ft]*

#### 2.4.2. Protocolo SPI

El protocolo SPI define un bus de comunicación con cuatro líneas y una tierra común para la transferencia de datos entre dispositivos electrónicos, fue desarrollado por Motorola y posteriormente fue adoptado por otros fabricantes de circuitos

electrónicos. Dos líneas son utilizadas para la transferencia de datos, una para la sincronización durante el envío de datos y la última está dedicada a la activación del circuito integrado. En el protocolo SPI se tiene un dispositivo denominado *maestro* que controla la actividad en el bus y uno o más dispositivos denominados *esclavos* que operan de acuerdo a las instrucciones dictadas por el dispositivo *maestro*. En la mayoría de los casos el *maestro* suele ser un microcontrolador mientras que los *esclavos* son periféricos, sensores o convertidores analógico-digitales.

Las cuatro líneas en el bus de datos son las siguientes: la línea MOSI (*Master Output - Slave Input*) permite la transferencia de datos desde el *maestro* hacia el *esclavo*. La línea MISO (*Master Input - Slave Output*) por su parte, posibilita el envío de datos desde el *esclavo* hacia el *maestro*. La utilización de ambas líneas facilitan una comunicación *full dúplex*, es decir, los dispositivos pueden enviar y recibir información al mismo tiempo. Puesto que el protocolo está diseñado para que un solo *maestro* controle uno o más *esclavos*, una línea denominada SS (*Slave Select*) es incorporada y su propósito es activar aquel *esclavo* con el que se establecerá la comunicación. Es importante hacer notar que una desventaja del protocolo SPI es que en el caso de utilizar múltiples *esclavos* será necesario utilizar múltiples líneas para la selección de cada circuito integrado.

La última línea es una señal de reloj, comúnmente denominada SCK, cuyas siglas hacen referencia a *Serial Clock*. El uso de esta línea es la razón por la cual la comunicación se denomina síncrona, ya que la transferencia de datos entre *maestro* y *esclavo* se realiza únicamente al detectar un flanco de subida o bajada de la señal de reloj; en ausencia de una señal de reloj la comunicación no se realiza. La figura 2.58 muestra un esquema del bus SPI y la interconexión de los dispositivos.

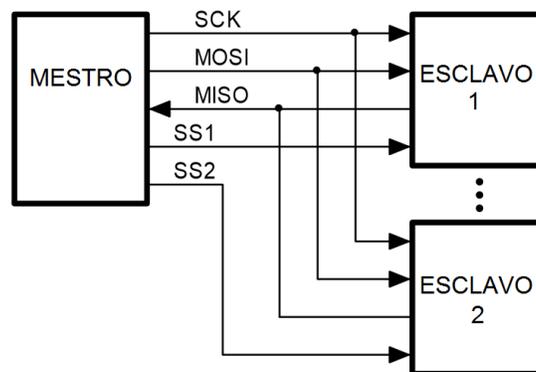


Figura 2.58. Bus SPI.

El protocolo SPI es una alternativa ante la comunicación serial asíncrona mediante un módulo UART, ya que en esta última se requieren bits adicionales para indicar el inicio y el final de la transmisión de la información, además, si los dispositivos que se están comunicando no están configurados a la misma tasa de transferencia de datos, la información enviada y recibida no concordará. Adicionalmente, hablando

en cuestiones de hardware, en la comunicación mediante el protocolo SPI la implementación se ve reducida a la utilización de un simple registro de corrimiento.

La figura 2.59 muestra la interconexión entre un *maestro* y un *esclavo*. El sistema completo consiste de dos registros de corrimiento y un generador de reloj. El dispositivo *maestro* inicia el ciclo de comunicación cuando mantiene una señal baja en la terminal SS. Tanto el maestro como el esclavo preparan el dato que va a ser enviado en sus respectivos registros de corrimiento y el *maestro* genera los pulsos de reloj en la línea SCK para intercambiar datos. Los datos siempre son recorridos en la dirección de *maestro* a *esclavo* en la línea MOSI y, de *esclavo* a *maestro* en la línea MISO. Después del envío de cada paquete de datos, el *maestro* detiene la fuente de reloj y posteriormente deshabilitará al *esclavo* manteniendo una señal alta en la terminal SS. Vale la pena señalar que la interfaz SPI no ejerce ningún control sobre la línea SS y ésta debe ser controlada mediante software para indicarle al microcontrolador cuando debe activar y desactivar la selección de un *esclavo*.

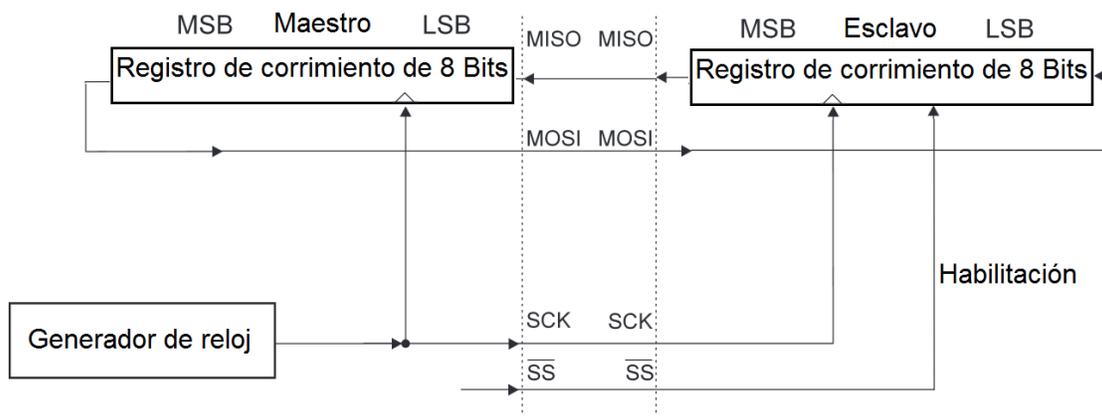


Figura 2.59. Interconexión entre maestro y esclavo.

La figura 2.60 muestra un ejemplo de las señales eléctricas observadas en el bus SPI. En esta figura se observa que la señal CS es mantenida en un nivel bajo a lo largo de toda la transferencia de datos. En la línea SCK se aprecia la línea de reloj que permite la sincronización en la transferencia de los datos en cada flanco de subida (transición de un valor bajo a alto). En este mismo ejemplo, los datos son enviados desde el LSB hasta el MSB y se observa que a través de la línea MOSI el *maestro* envía el valor 0x53 y, posteriormente, el *esclavo* responde con el valor 0x46 enviado a través de la línea MISO. Este ejemplo ilustra el acceso al registro de control de un reloj de tiempo real, en cuyo caso, el microcontrolador primero envía el comando para acceder al registro de control y en consecuencia, el reloj de tiempo real envía el estado del registro.

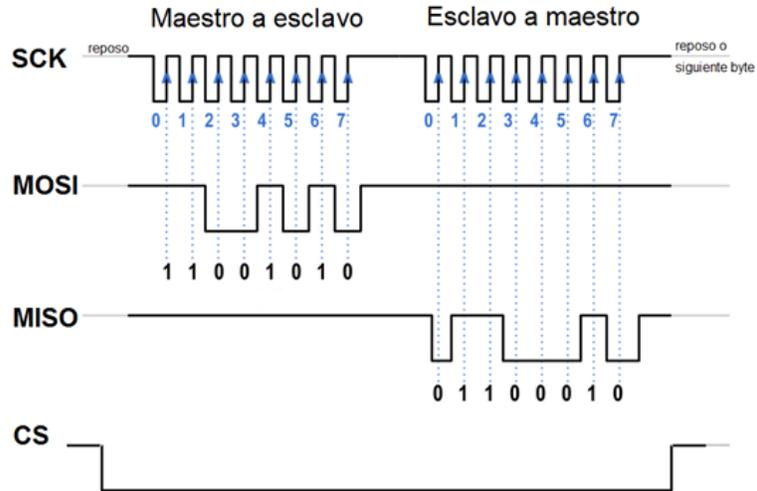


Figura 2.60. Señales eléctricas en el bus SPI.

### *Polaridad y fase de la señal de reloj*

Además de configurar la frecuencia de reloj del protocolo SPI, la cual se obtiene mediante un generador de reloj basado en el cristal utilizado por el mismo microcontrolador, el *maestro* también debe establecer la polaridad y fase con la que se utilizará la señal de reloj. Comúnmente estos parámetros se designan con las siglas CPOL (*Clock Polarity*) y CPHA (*Clock Phase*). La operación de la fuente de reloj al configurar CPOL y CPHA se resume de la siguiente manera:

- Cuando CPOL=0, el valor de estado inactivo (o reposo) del reloj es un valor bajo, mientras que el valor de estado activo es alto.
  - Para CPHA=0, el dato recibido es capturado en el flanco de subida, mientras que cada bit del dato transmitido sale durante el flanco de bajada.
  - Para CPHA=1, el dato recibido es capturado en el flanco de bajada y cada bit transmitido se sincroniza con el flanco de subida.
- Cuando CPOL=1, el valor de estado inactivo del reloj es un valor alto, mientras que el valor de estado activo es bajo.
  - Para CPHA=0, el dato recibido es capturado en el flanco de bajada y la salida de cada bit transmitido se sincroniza con cada flanco de subida.
  - Para CPHA=1, el dato entrante se captura en el flanco de subida y cada bit saliente se sincroniza con el flanco de bajada.

La figura 2.61 ilustra las señales del bus SPI cuando se configuran CPOL y CPHA. De la figura y conforme a lo señalado anteriormente, se puede concluir que CPHA=0 significa muestrear el dato recibido en el primer flanco del reloj mientras que CPHA=1 significa muestrear en el segundo flanco, independientemente del valor de CPOL. En otras palabras, CPHA=0 significa transmitir el dato en la transición del estado activo a estado inactivo y CPHA=1 lo contrario. Una observación importante

es que si la transmisión ocurre en un flanco particular, entonces la captura se realizará en el flanco opuesto, por ejemplo, si la transmisión ocurre durante el flanco de bajada, entonces la recepción se realizará en el flanco de subida y viceversa. Las líneas MISO y MOSI son estables durante la mitad de un ciclo de reloj, esto añade mayor flexibilidad durante la comunicación entre dispositivos.

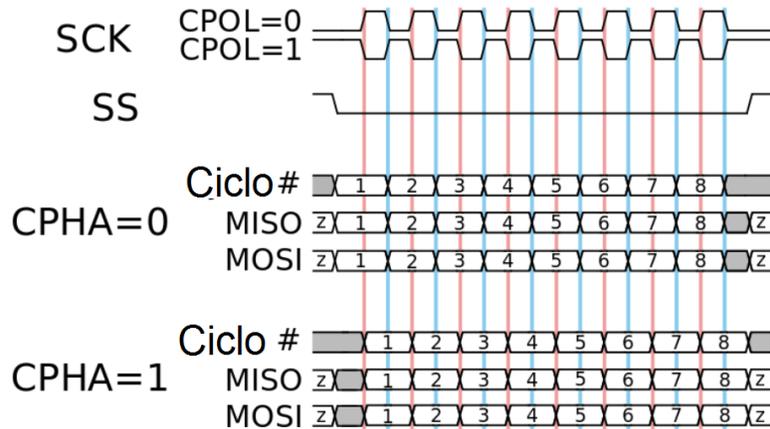


Figura 2.61. Configuración de CPOL y CPHA.

La manera de seleccionar los dos parámetros citados anteriormente da lugar a los distintos *modos* de la señal de reloj del protocolo SPI. A manera de ejemplo, para el caso de un microcontrolador ATMEGA328, los modos de operación de la fuente de reloj se resumen en la tabla 2.3.

Mode	CPOL	CPHA
SPI_MODE0	0	0
SPI_MODE1	0	1
SPI_MODE2	1	0
SPI_MODE3	1	1

Tabla 2.3. Modos de operación para un microcontrolador ATMEGA328.

Finalmente, vale la pena mencionar que en el mercado existe una infinidad de dispositivos electrónicos que soportan comunicación serial mediante SPI, sin embargo, como los fabricantes y sus estándares son diversos, los dispositivos operan cada uno con distintos parámetros: polaridad, fase, frecuencia máxima de transmisión de datos y el formato de transferencia de datos (si el primer bit enviado es el MSB o el LSB), este hecho requiere especial atención para elegir únicamente aquellos dispositivos que trabajen de manera similar y que puedan comunicarse mediante el mismo bus y con la misma fuente de reloj. Esta característica marca uno de los inconvenientes de este protocolo, no obstante, la velocidad de

transmisión de datos del protocolo SPI, que corresponde a algunos [MHz], hacen de él un método de comunicación muy utilizado entre dispositivos electrónicos.

#### 2.4.3. Protocolo I2C

El protocolo I2C define un bus de datos serial desarrollado por Phillips Semiconductors en 1982. Es utilizado principalmente para la comunicación entre diferentes partes de un circuito, por ejemplo, entre un microcontrolador y periféricos. El sistema original fue desarrollado con el fin de controlar varios circuitos integrados en televisiones de manera sencilla. La compañía ATMEL que fabrica entre otras cosas, microcontroladores, por motivos de licencia introdujo la designación TWI (*Two-Wired-Interface*) actualmente utilizada por algunos otros fabricantes. Desde el punto de vista técnico, TWI e I2C son idénticos.

El bus I2C está diseñado como un bus *maestro-esclavo*, cada dispositivo conectado puede actuar como *esclavo* o *maestro*, en cuyo caso regirá el funcionamiento del bus. Se tienen dos líneas: SCL (*Serial Clock*) y SDA (*Serial Data*), la primera corresponde a la línea de reloj para la sincronización durante la transferencia de datos, mientras que la segunda es utilizada para enviar la dirección del dispositivo a elegir y los datos que se desean transferir. La señal de reloj siempre es generada por el *maestro* en curso, aunque los *esclavos* pueden forzar la línea SCL a tener un valor bajo en un momento crítico para generar un retardo de tiempo y evitar que el *maestro* continúe con el envío de datos.

A diferencia de las conexiones en el bus del módulo UART y el bus SPI, al bus I2C se conectan únicamente circuitos integrados con salida *open drain* u *open collector*, lo cual conlleva a la conexión de una resistencia de *pull-up* en cada línea del bus. Esta característica crea un circuito denominado *Wired-AND*, es decir, se crea una función lógica AND en el mismo alambre o conductor. Al disponer de una resistencia de *pull-up*, se elimina la *contención del bus* (una característica indeseada en un bus de comunicación, caracterizada por poseer múltiples dispositivos conectados a éste e intentar ejercer dominio del bus al transmitir datos al mismo tiempo), pues los dispositivos conectados a la línea sólo pueden forzarla a tomar un valor bajo, de esta manera, si un circuito está enviando un "0" lógico y un intruso está enviando al mismo tiempo un "1" lógico, se respetará el primer valor puesto que se forzará a la corriente a circular por el transistor "*open collector*" del primer circuito integrado y el valor en la línea será "0". La utilidad de emplear resistencias de *pull-up* radica en la eliminación del daño causado por una excesiva disipación de potencia en el sistema. En la figura 2.62 se muestra un esquema del bus I2C y de la conexión de múltiples dispositivos, cada línea del bus cuenta con su propia resistencia de *pull-up*: R1 y R2.

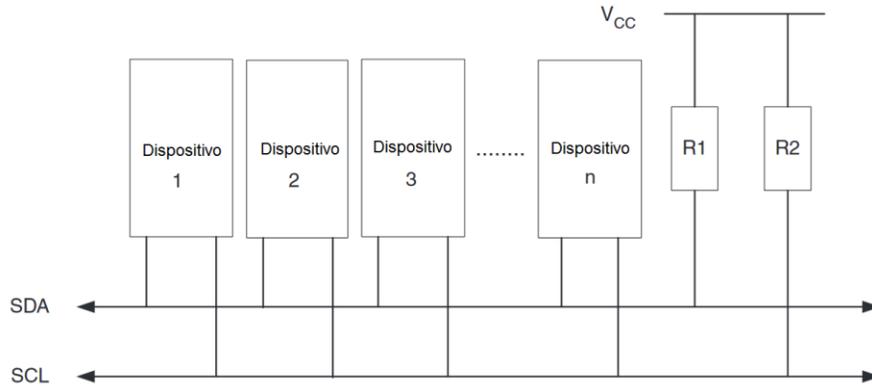


Figura 2.62. Bus I2C e interconexión de dispositivos.

### Terminología

El protocolo I2C utiliza la siguiente terminología:

- **Maestro:** Es el dispositivo que inicia y termina una transmisión de datos. El maestro también genera la señal de reloj.
- **Esclavo:** Es el dispositivo direccionado por el *maestro*.
- **Transmisor:** Es el dispositivo en turno, que se encuentra escribiendo datos en el bus.
- **Receptor:** Es el dispositivo en turno, que se encuentra leyendo datos del bus.

### Transferencia de datos

La transferencia de datos únicamente es posible cuando el bus no se encuentra ocupado y ambas líneas, SCL y SDA, cuentan con un valor lógico alto. Los bits enviados a través de la línea SDA por el transmisor, pueden clasificarse en bits de información y bits de control. Cada bit enviado por la línea SDA va acompañado por un ciclo de reloj, sin embargo, mientras que los bits de control son utilizados para generar condiciones de inicio y terminación del envío de datos, los bits de información deben mantenerse “estables” mientras la señal de reloj se mantiene en un nivel lógico alto, esto es, mientras la señal de reloj se encuentra en un nivel alto, no debe haber ningún flanco de subida o bajada, o los bits serán interpretados como bits de control. Únicamente en esta condición los datos son válidos. La figura 2.63 muestra la condición de validez en los datos.

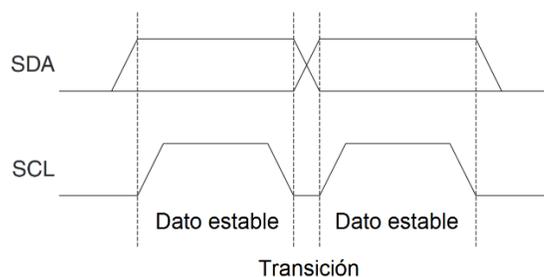


Figura 2.63. Condición de dato válido.

Como se observa en la figura anterior, en todo momento en que la línea SCL toma un valor alto, la línea SDA mantiene estabilidad ya sea tomando un valor alto o bajo pero sin presentarse ninguna transición. Por otro lado, los bits en la línea de datos SDA, sólo presentan la transición de alto a bajo o viceversa cuando la señal de reloj se encuentra en un valor bajo.

El *maestro* inicia y termina todas las transmisiones de datos. Para iniciar la transmisión, el *maestro* genera una condición de inicio (*start condition*) y para terminarla genera una condición de parada (*stop condition*). Entre una condición de inicio y una condición de parada, el bus se considera ocupado y ningún otro maestro debe intentar ejercer dominio de éste. Se tiene una tercera condición denominada *repeated start* que se traduce literalmente como inicio repetido y tiene lugar cuando un *maestro* desea iniciar una nueva transferencia de datos sin abandonar el dominio del bus. Después de generarse esta condición, el bus se considera ocupado hasta generar una condición de parada. Las tres condiciones se ilustran en la figura 2.64.

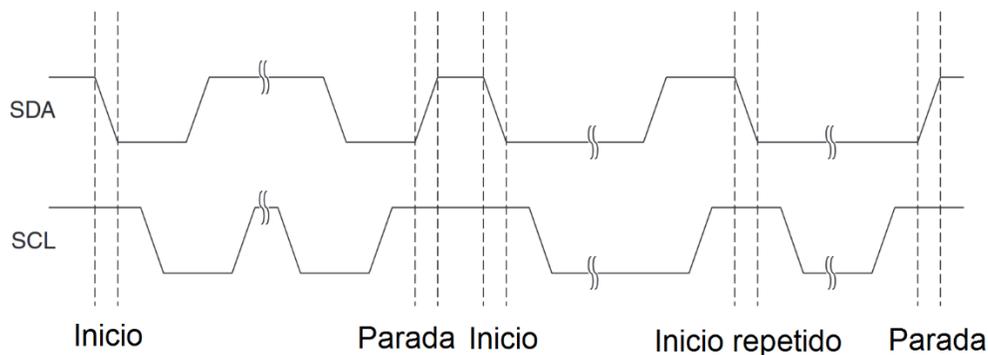


Figura 2.64. Condiciones de inicio, parada e inicio repetido.

Como se observa en la figura anterior, una transición de alto a bajo en la línea SDA, mientras la línea SCL se mantiene en alto, define una condición de inicio. La condición de parada se presenta cuando un cambio de bajo a alto en la línea SDA tiene lugar mientras la línea SCL mantiene un valor alto. La condición de inicio repetido tiene lugar después de haberse generado una condición de inicio y las características en las líneas es exactamente el mismo que para una condición de inicio.

#### *Formato de los datos de direcciones*

Toda cadena (denominada *frame*) está compuesta por nueve bits, de los cuáles, 7 corresponden a la dirección del dispositivo que va a ser direccionado, un bit de control de lectura/escritura (R/W, Read/Write) y un bit de reconocimiento. Al colocar en un valor alto el bit de lectura/escritura, se realizará una operación de lectura por parte del maestro y, al colocar este bit en un valor bajo, se procederá con una operación de escritura. La utilización de 7 bits para indicar la dirección del dispositivo permite interactuar con 128 dispositivos conectados al bus, sin embargo, la dirección 0x00000000 suele estar reservada para realizar un *llamado general*, cuando esto sucede, todos los dispositivos colocan el bit de reconocimiento en un

valor bajo y reciben todos el mismo mensaje del *maestro*, el cual, será leído por cada uno de ellos. El *llamado general* sólo se utiliza en el modo de lectura puesto que los dispositivos no pueden escribir datos en el bus al mismo tiempo.

Como se mencionó, los dispositivos esclavos envían un bit de reconocimiento (*ACK*, *Acknowledge*). Cuando un *esclavo* reconoce que está siendo direccionado, éste debe indicarlo poniendo la línea SDA en un valor bajo en el noveno ciclo de reloj de la línea SCL. Si el esclavo direccionado se encuentra ocupado, o por alguna razón no puede procesar la petición del *maestro*, la línea SDA se mantendrá en un nivel alto. El *maestro* puede después transmitir una condición de parada o de inicio repetido para gestionar una nueva transmisión.

### Formato de los datos de información

Toda cadena está compuesta por nueve bits, de los cuáles, 8 corresponden a los bits de información enviada o recibida y un bit de reconocimiento. Después de que un dispositivo ha sido direccionado, aún durante la transferencia de datos (ya sea en modo lectura o escritura), el esclavo está obligado a generar un bit de reconocimiento *ACK*, poniendo la línea SDA en bajo en el noveno ciclo de reloj de la señal SCL. El formato de los datos de direcciones y de información enviado a través del bus I2C se muestra en la figura 2.65.

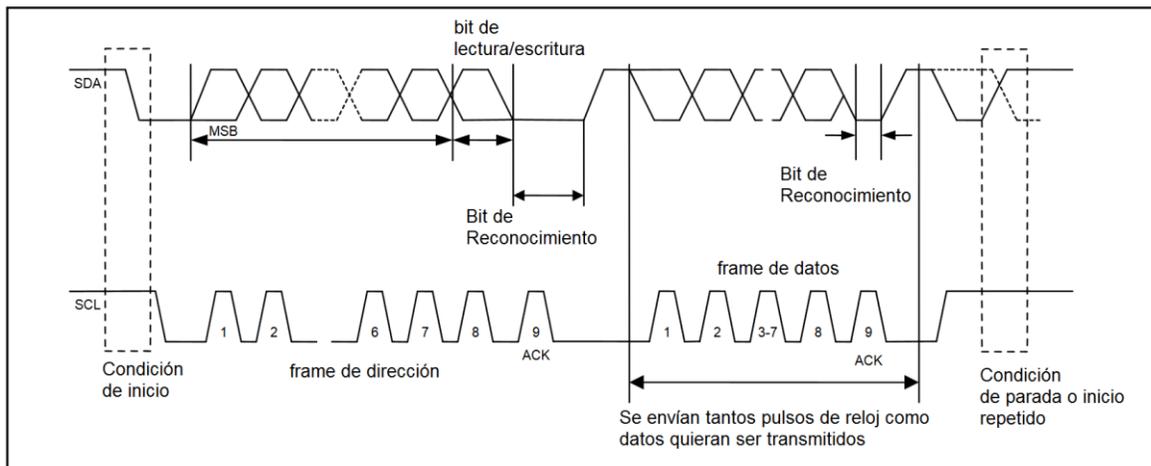


Figura 2.65. Formato de los datos transmitidos en el bus I2C.

### Transferencia de datos de un maestro transmisor a un esclavo receptor

Cuando se configura al *esclavo* para operar en modo escritura, primeramente el *maestro* deberá generar una condición de inicio, a continuación, enviará la dirección del dispositivo con el que quiere establecer una comunicación seguido por el bit R/W con un valor bajo. Si el dispositivo reconoce que está siendo direccionado deberá responder colocando el bit *ACK* en un valor bajo. En seguida, el *maestro* enviará el primer byte de datos y el *esclavo* generará el bit *ACK* en el noveno ciclo de reloj. El *maestro* enviará el segundo byte de información y así sucesivamente hasta que el

*maestro* genere una condición de parada. El proceso descrito se muestra gráficamente en la figura 2.66.

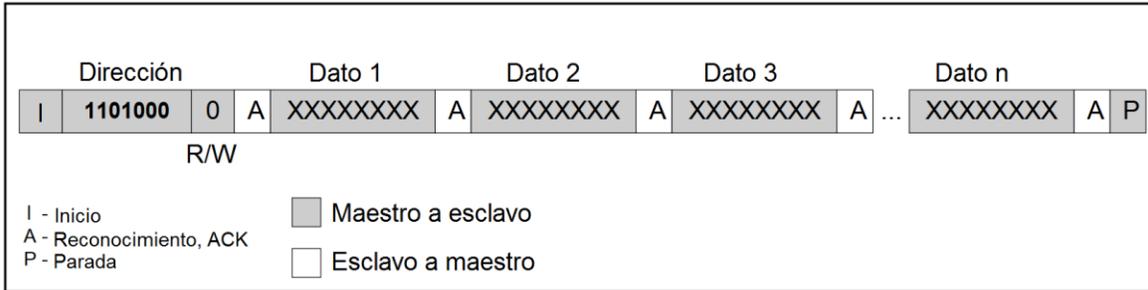


Figura 2.66. Secuencia de datos al configurar al esclavo en modo escritura.

*Transferencia de datos de un esclavo transmisor a un maestro receptor*

Cuando se configura al esclavo para operar en modo lectura, en primer lugar el *maestro* deberá generar una condición de inicio, a continuación, enviará la dirección del dispositivo con el que quiere establecer una comunicación seguido por el bit R/W con un valor alto. Si el dispositivo reconoce el direccionamiento deberá responder colocando el bit ACK en un valor bajo. En el siguiente paso los papeles del *maestro* y el *esclavo* se invierten, esta vez el esclavo colocará los datos en el bus y el *maestro* leerá estos datos, además, al finalizar la transmisión de cada byte por parte del *esclavo*, el maestro generará el bit de reconocimiento ACK en el noveno ciclo de reloj de la señal SCL. Para finalizar la lectura de datos, el *maestro* debe mantener la señal de reconocimiento en alto indicando al *esclavo* que ya no desea recibir más datos y que éste debe dejar de transmitirlos. El proceso descrito anteriormente se ilustra en la figura 2.67.

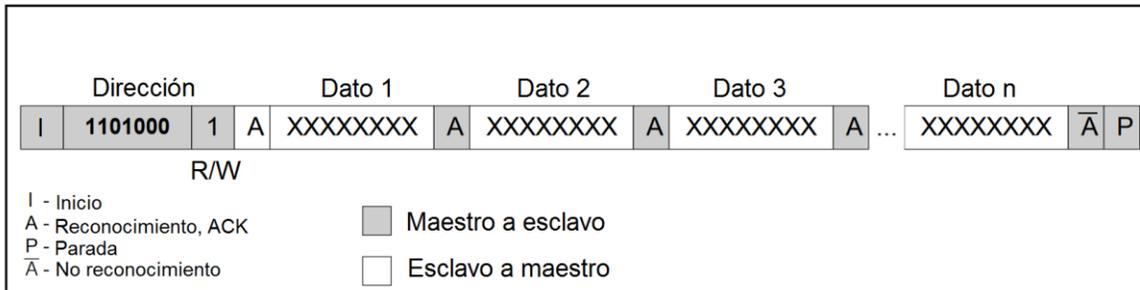


Figura 2.67. Secuencia de datos al configurar al esclavo en modo lectura.

**Módulo I2C**

En cuanto al hardware utilizado para la implementación del protocolo I2C, se utiliza un módulo compuesto de distintos bloques con tareas específicas pero que en conjunto trabajan para generar las señales en el bus. La figura 2.68 muestra el diagrama de bloques de un módulo general para la generación de señales en el protocolo I2C.

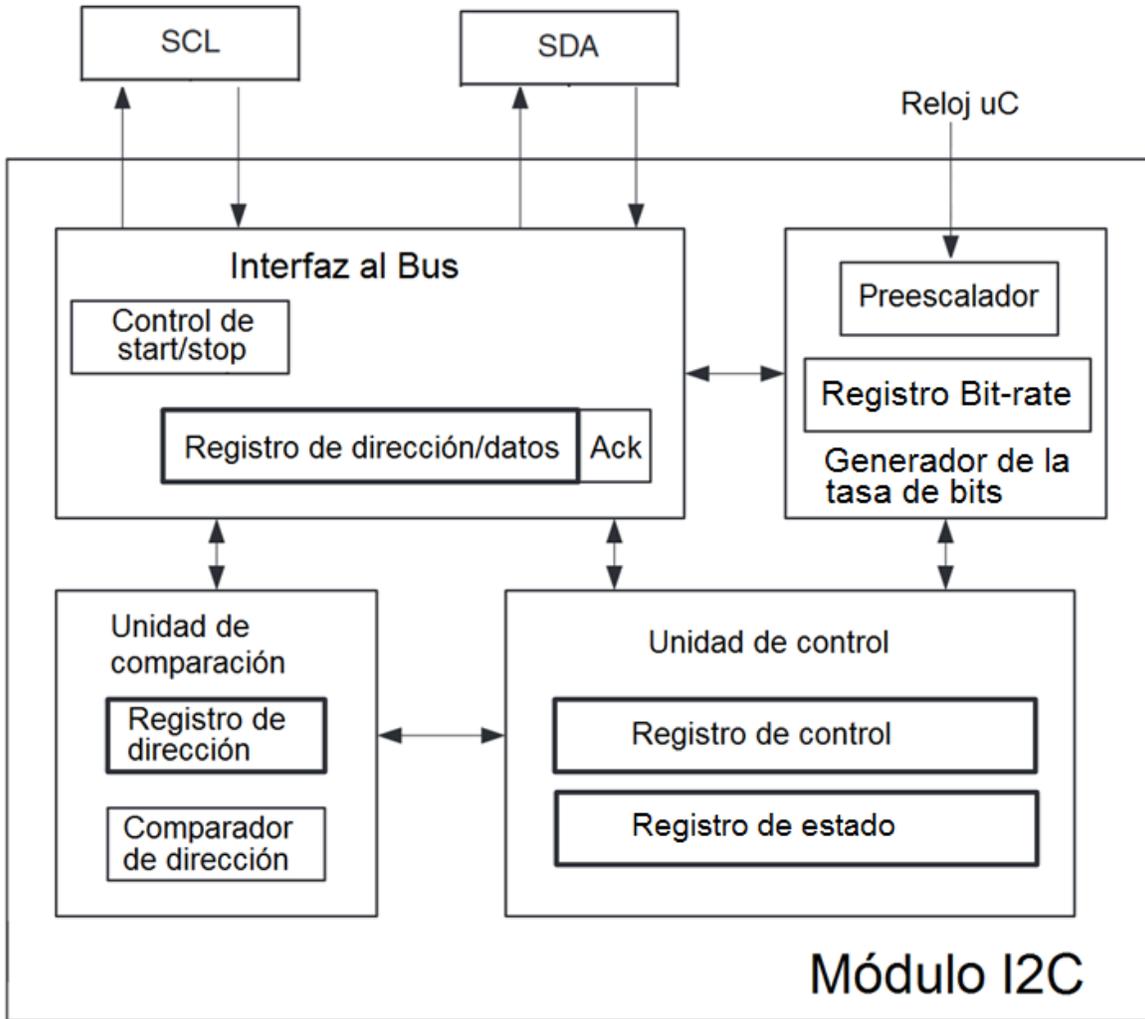


Figura 2.68. Módulo I2C.

Las terminales SCL y SDA son el contacto directo con los conductores eléctricos en la placa de circuito impreso para enlazar múltiples dispositivos y establecer la comunicación entre ellos. Algunos microcontroladores permiten ejercer un control sobre el *slew rate* y además incluyen un filtro de picos para la operación correcta durante el envío y recepción de datos.

El generador de la tasa de bits (*Bit-rate*) o tasa de transferencia de datos, es el encargado de proporcionar la señal de reloj a los registros internos del módulo I2C y también a los periféricos conectados al bus a través de la línea SCL. Cuenta con un pre-escalador que adquiere la señal de reloj del microcontrolador y permite disminuir la frecuencia dividiéndola en alguna fracción de la original. El generador de la tasa de bits incluye un registro para configurar la frecuencia de la señal SCL.

La frecuencia para el protocolo I2C es de 100 [kHz] en el modo estándar y 400 [kHz] en el modo rápido. Existen también el modo rápido *plus* (1 [MHz]) y el modo de alta velocidad (3.4 [MHz]), pero son menos utilizados.

La unidad de comparación realiza una verificación de dirección cuando un maestro envía este dato a través del bus. Cuenta con un registro que almacena la dirección del propio dispositivo y un elemento con pequeños bloques digitales para realizar la comparación entre la dirección almacenada y la recibida por la línea SDA.

La interfaz con el bus incluye un elemento para generar las condiciones de inicio y parada, este elemento ejerce el control de las señales de modo que el sistema pueda transmitir bits de control sobre la línea SDA. El otro elemento que conforma la interfaz con el bus es un registro de corrimiento de salida serial y con 8 posiciones de entrada paralelo, adicionalmente, se incluye una novena posición correspondiente al estado del bit de reconocimiento ACK. El usuario puede manipular mediante software el registro de corrimiento para indicar el dato que será enviado, o la dirección y el bit R/W para transmitir por el bus. La novena posición no puede ser manipulada cuando el dispositivo actúa como transmisor, pero al operar como receptor este bit puede ser modificado mediante la unidad de control.

La unidad de control básicamente tiene como tarea gestionar el estado del bit ACK. Cuando el dispositivo actúa como transmisor, mediante el registro de estado se puede monitorear la condición del bit de reconocimiento enviado por el transmisor y saber si éste aún se encuentra sincronizado. Cuando el dispositivo trabaja como receptor, mediante el registro de control se puede indicar el estado del bit ACK para que de esta manera responda después de recibir cada byte de datos.

## **2.5. Memorias de estado sólido**

Las memorias de estado sólido son aquellas construidas a partir de circuitos o dispositivos construidos totalmente de materiales semiconductores y en los que los electrones, u otros portadores de carga, están confinados enteramente dentro de un material sólido. El término “electrónica de estado sólido” se utiliza a menudo para contrastar con las tecnologías anteriores de vacío y dispositivos electromecánicos como relés, interruptores, discos duros y otros con partes móviles.

Durante años, la solución para el almacenamiento de datos fue utilizar elementos electromecánicos o magnéticos; ante las limitaciones que presentaban estos elementos, como retardos en los movimientos mecánicos de platos giratorios y cabezales, alta sensibilidad a los golpes, alto consumo de energía, generación de calor y ruido, entre otros; se desarrollaron entonces dispositivos de almacenamiento de estado sólido capaces de solucionar todos estos inconvenientes.

Entre las ventajas proporcionadas por las memorias de estado sólido se encuentran: menor tiempo de acceso a la memoria, mayor rapidez de lectura/escritura, menor consumo de energía y de calor, mayor seguridad de almacenamiento de datos y

menor peso y dimensiones. Sin embargo, se tienen también inconvenientes, tales como: menor tiempo de vida útil (los ciclos de lectura/escritura son limitados), menor probabilidad de recuperación de datos cuando la unidad sufre algún daño y, vulnerabilidad ante la pérdida abrupta de energía eléctrica.

Las memorias de estado sólido pueden clasificarse en función de las operaciones que se pueden realizar sobre ellas, de esta manera, se tienen memorias de sólo lectura, memorias de sobre todo lectura y memorias de lectura/escritura. Los ejemplos de estos tipos de memorias son los siguientes:

- Memorias de sólo lectura
  - *ROM (Read Only Memory)*. Se usan principalmente como medio de almacenamiento utilizado en ordenadores y dispositivos electrónicos, que permite sólo la lectura de información y no su escritura, independientemente de la presencia o no de una fuente de energía.
  - *PROM (Programmable Read Only Memory)*. Estas memorias son utilizadas para grabar datos permanentes en cantidades menores a las ROM. Los datos en la PROM pueden ser cambiados con un programador especial.
- Memorias de sobre todo lectura
  - *EPROM (Erasable Programmable Read Only Memory)*. Se pueden escribir varias veces de forma eléctrica, sin embargo, el borrado de los contenidos es completo y a través de la exposición a rayos ultravioleta, para ello, suelen tener una ventanilla en la cubierta del circuito integrado.
  - *EEPROM (Electrically Erasable Programmable Read Only Memory)*. A diferencia de las EPROM, estas memorias se pueden borrar selectivamente byte a byte con corriente eléctrica.
  - *Memoria flash*. Está basada en la memoria EEPROM pero permite el borrado de bloques enteros de información en una misma operación, utilizando un menor tiempo que en el caso de las EEPROM. También suele ser más barata y con mayor densidad para el almacenamiento de datos.
- Memorias de lectura/escritura
  - *DRAM (Dynamic Random Acces Memory)*. Permiten la lectura y escritura de datos, éstos son almacenados en un capacitor que con el paso del tiempo tiende a perder la carga eléctrica, en consecuencia, se requiere de un circuito que recupere el valor original almacenado.

- *SRAM (Static Random Access Memory)*. A diferencia de la DRAM, los datos se almacenan en un arreglo de transistores y por tal motivo no requieren de un mecanismo para refrescar la información.

Las memorias de estado sólido se pueden clasificar también en dos tipos de acuerdo a su capacidad para perder o retener los datos almacenados en ausencia de una fuente de polarización. Estos dos tipos de memoria son la volátil y la no volátil, respectivamente. Entre las memorias del tipo volátil se encuentran la SRAM y DRAM. Por otro lado, ejemplos de memoria de tipo no volátil son la memoria flash, ROM, PROM, EPROM y EEPROM.

### 2.5.1. Memoria RAM

La memoria de acceso aleatorio (RAM) se utiliza para almacenar datos que serán utilizados por un sistema electrónico durante su operación y por lo tanto, el acceso debe realizarse lo más rápido posible. En la memoria RAM se almacenan datos e instrucciones que utilizará la unidad central de procesamiento (microprocesador). Una característica de las memorias de acceso aleatorio es que pueden ser escritas o leídas en cualquier orden, independientemente de cuál fue la última posición de memoria accedida.

Las dos variantes principales de memoria RAM son: DRAM (*Dynamic Random Acces Memory*) y SRAM (*Static Random Acces Memory*).

#### *DRAM y SRAM*

La memoria DRAM es un tipo de tecnología basada en capacitores, los cuales pierden su carga progresivamente y por lo tanto necesitan de un circuito de *refresco* que, cada cierto periodo de tiempo, reponga el nivel de voltaje del capacitor. La ventaja de este tipo de memorias es la posibilidad de construir memorias con una gran densidad de posiciones. Por otro lado, la memoria SRAM utiliza otro tipo de construcción que elimina la necesidad de utilizar un circuito de refresco. En este tipo de memorias cada bit de información se almacena en cuatro transistores que conforman un circuito biestable (elemento de memoria capaz de permanecer en uno de dos estados posibles durante un tiempo indefinido) para almacenar un 0 o un 1. Se puede concluir que la diferencia básica entre una DRAM y una SRAM es la tecnología de construcción para los elementos de memoria.

La celda de memoria es el bloque de construcción mínimo de una memoria computacional. La celda de memoria es un circuito electrónico que almacena un bit de información y se debe establecer (*set*) para almacenar un 1 y re-establecer (*reset*) para almacenar un 0. Su valor se mantiene hasta que sea cambiado por un proceso de establecimiento (*set*) o re-establecimiento (*reset*). El bit de información almacenado puede conocerse mediante un proceso de lectura.

La celda de memoria de la SRAM consiste de un circuito *flip-flop*, usualmente implementado con MOSFETs, requiriendo muy poco consumo de energía cuando

no está siendo accedida. En cuanto a la celda de memoria DRAM, se basa en el almacenamiento de carga en un capacitor. Cargando y descargando este capacitor se puede almacenar un 1 o un 0, sin embargo, este capacitor se descarga lentamente a lo largo del tiempo y necesita ser refrescado periódicamente. Debido a este proceso de refresco, la DRAM utiliza mayor consumo de potencia aunque puede proporcionar mayor densidad de almacenamiento por la simplicidad de la celda con respecto a la utilizada por la SRAM. Las figuras 2.69 y 2.70 muestran una celda de memoria DRAM y SRAM respectivamente.

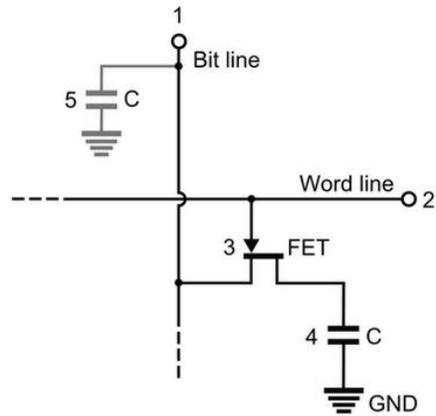


Figura 2.69. Celda de memoria DRAM.

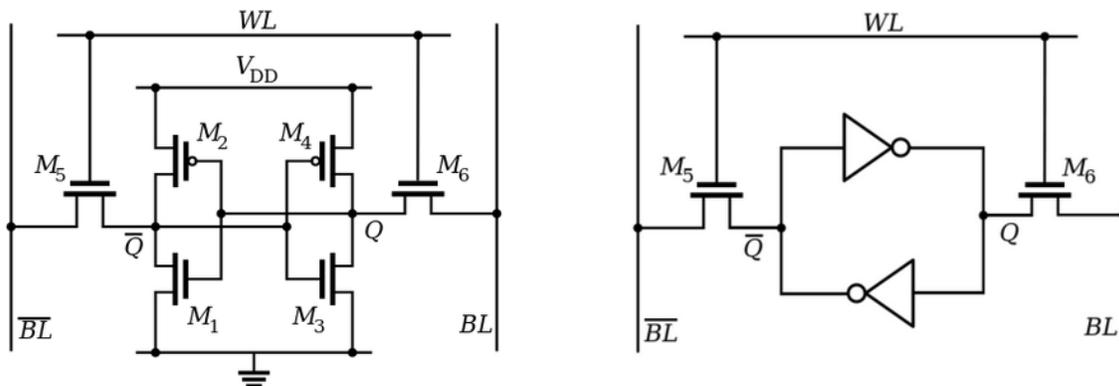


Figura 2.70. Celda de memoria SRAM.

### Funcionamiento de la celda DRAM

- Reposo. El elemento de almacenamiento de una DRAM es el capacitor marcado con el número 4 en la figura 2.69. La carga del capacitor se degrada con el tiempo, de manera que su valor debe ser refrescado. El MOSFET de canal n, marcado con el número 3 en la figura antes citada, actúa como una compuerta para permitir la lectura o escritura cuando el MOSFET se encuentra conduciendo. En el proceso de reposo, ninguna de las líneas (*Word line* y *bit line*) es activada.

- Lectura. Para realizar una operación de lectura, la línea “Word line” mantiene un nivel lógico alto en la compuerta del MOSFET, que lo hace conducir y la carga del capacitor es transferida a la línea “bit line”. La línea “bit line” posee una capacitancia parásita (indicada con el número 5 en la figura 2.69) que drenará parte de la carga de (4) y hará más lento el proceso de lectura. Si el capacitor que almacena el bit de información no cuenta con el valor de capacitancia adecuado, la línea “bit line” no aumentará hasta el valor lógico alto. Puesto que el proceso de lectura degrada la carga del capacitor de almacenamiento de información, su valor es re-escrito después de cada lectura.
- Escritura. Es el proceso más sencillo. Conlleva la escritura de un valor lógico alto o bajo en la línea “bit line”, que corresponde al bit que quiere ser almacenado. Se activa la línea “Word line” para activar la conducción del MOSFET y por lo tanto el valor en “bit line” pasa directamente al capacitor de almacenamiento de información. La única tarea después de esto, consiste en mantener activada la línea “Word line” por el tiempo suficiente para que el capacitor pueda ser totalmente cargado o descargado.

#### *Funcionamiento de la celda SRAM*

- Reposo. El principio de funcionamiento de la celda SRAM puede entenderse más fácilmente si se observa que los MOSFETs M1 a M4 constituyen compuertas lógicas: dos inversores acoplados cruzadamente. Un valor 1 a la entrada del primer inversor se convierte en un 0 a su salida, a su vez, ésta alimenta a la entrada del segundo inversor que a su salida genera y mantiene el 1 original. Esto crea un estado que no cambia a lo largo del tiempo. Si la línea de control (Word line) no está activada, los transistores de acceso M5 y M6 desconectan la celda de la línea de datos (bit line). Las dos compuertas formadas por M1 a M4 mantendrán los datos almacenados, en tanto dure la alimentación eléctrica.
- Lectura. Para leer el contenido de la celda almacenado en el lazo, los MOSFETs M5 y M6 deben activarse recibiendo voltaje en sus compuertas mediante la línea “Word line”. En ese momento comienzan a conducir y los valores Q y su complemento son transmitidos a las líneas de datos (bit line y su complemento).
- Escritura. El proceso de escritura es similar al de lectura, la diferencia es que ahora el nuevo valor que quiere ser almacenado en la celda es escrito en la línea “bit line” y en su complemento. Después, los MOSFETs M5 y M6 se activan para pasar la información hacia el lazo. Existen dos posibles casos:
  - a) Si el valor en el lazo y el valor escrito es el mismo, no existe ningún cambio.

- b) Si el valor en el lazo y el valor escrito son diferentes, se debe sobrescribir en el lazo para eliminar el bit anterior y conservar el bit escrito actualmente. Para que esto sea realizado, las características de los MOSFETs M5 y M6 son diferentes respecto de los otros; permiten mayor flujo de corriente a través de ellos para que el voltaje dominante sea el escrito en la línea "bit line".

La estructura de las celdas fue diseñada para ser duplicadas en las direcciones x y y con el propósito de crear arreglos de celdas que permitan el almacenamiento de grandes cantidades de bits. En la figura 2.71 se muestra una memoria RAM hipotética de 4 renglones y 4 columnas, cuya capacidad permitiría almacenar 16 bits de información.

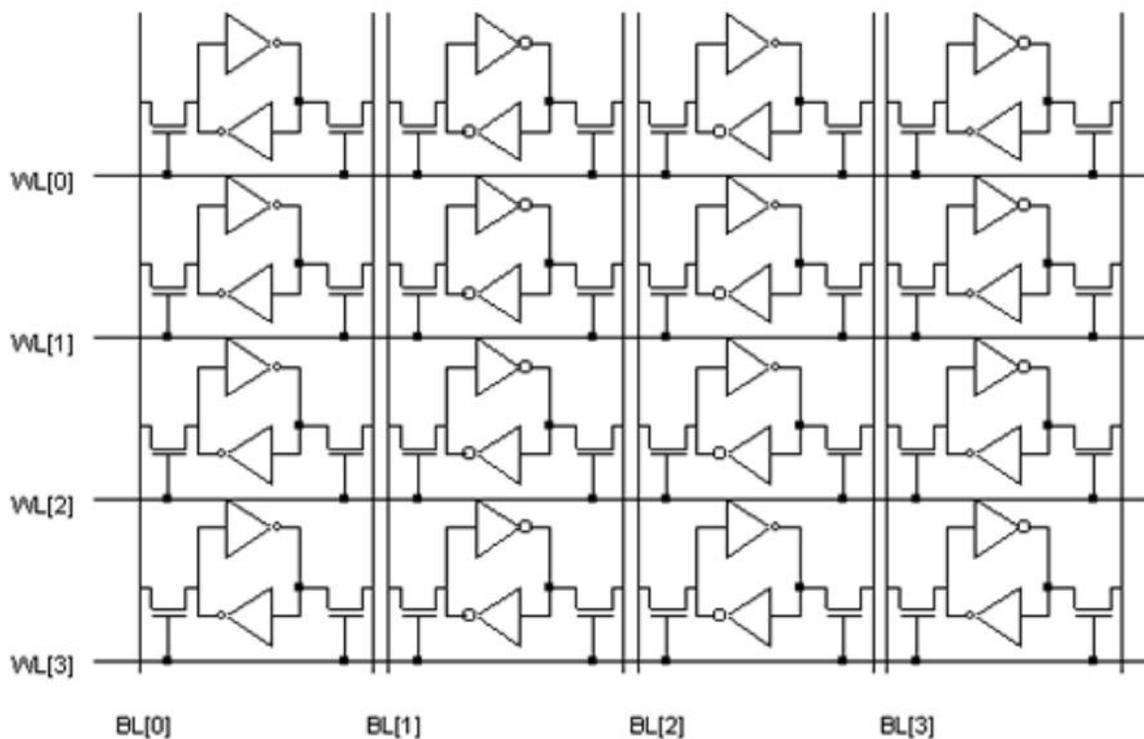


Figura 2.71. Memoria RAM de 4x4 celdas y 6 MOSFETs por celda.

Las celdas en una memoria RAM generalmente están dispuestas en una matriz bidimensional que utiliza dos decodificadores y un multiplexor para acceder a la celda deseada. El esquema de una memoria RAM se muestra en la figura 2.72. Cada posición dentro de la matriz cuenta con una dirección de memoria que puede ser especificada a través del bus de direcciones conformado por 8 conductores. El bus de direcciones se divide en dos partes. Los 4 bits más significativos son llevados hacia un decodificador de renglón mientras que los 4 bits menos significativos se llevan hacia un decodificador y multiplexor de columna. El decodificador de renglón es un decodificador de 4 a 16 y permite la habilitación de la línea "Word line" en cada renglón. Durante una operación de lectura los 16 bits del renglón seleccionado son pasados al multiplexor de columna 16 a 1 y el bit se elige de acuerdo a los 4

bits menos significativos del dato de direcciones. Posteriormente, el dato es enviado al buffer de entrada salida (I/O, Input/Output) para su posterior envío al circuito externo. El decodificador de columna también es 4 a 16 y durante la operación de escritura, habilita una columna y el dato es almacenado en la celda que interseca con el renglón habilitado.

Adicionalmente, se suele incluir un bloque de control que incluye las señales ce (chip enable o chip select) para habilitar el funcionamiento del circuito integrado, we (write enable) para habilitar la escritura al dispositivo y oe (output enable) para habilitar la salida de datos en el buffer de tres estados (tri-state).

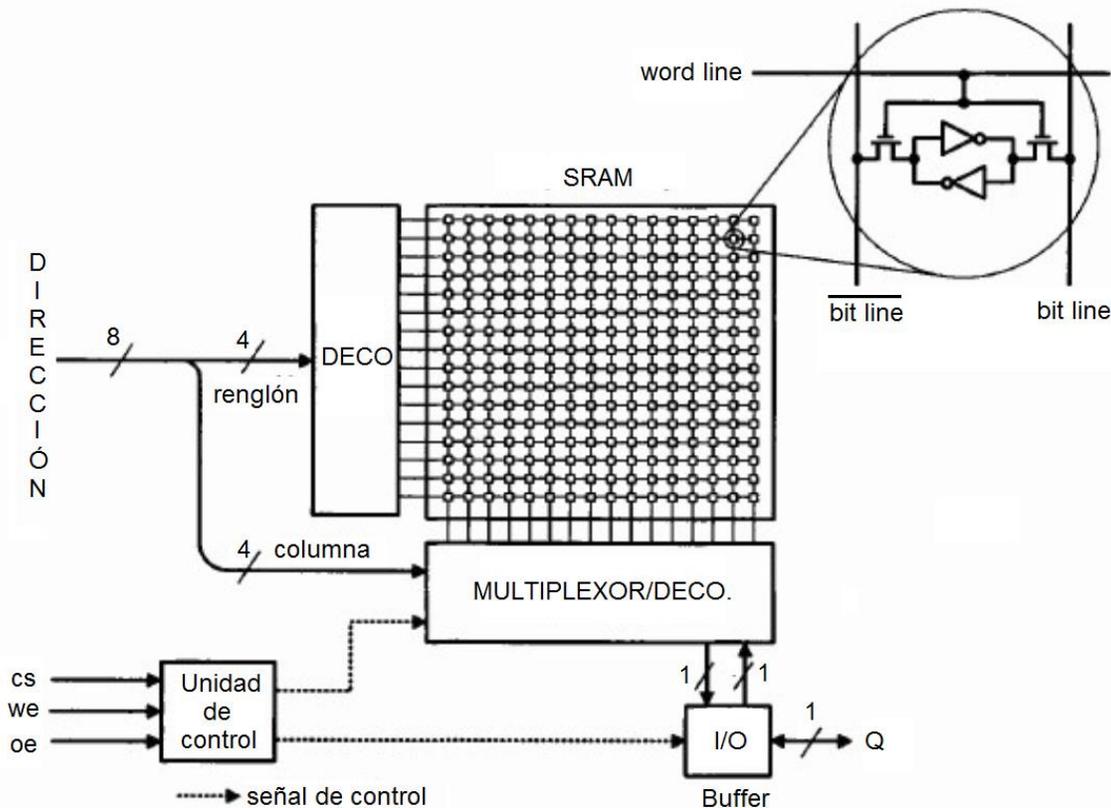


Figura 2.72. Estructura de una memoria SRAM.

### 2.5.2. Memoria flash

La memoria flash es un medio de almacenamiento electrónico no volátil. Fue introducida por el Dr. Fujio Masuoka en el año 1980 mientras trabajaba para la compañía Toshiba. Este tipo de memoria es la evolución de la memoria EEPROM y se clasifica básicamente de acuerdo al tipo de compuertas que utiliza para el almacenamiento de los datos, de esta manera, existe la memoria flash de tipo NAND y la memoria flash de tipo NOR.

La información en la memoria flash de tipo NAND puede ser escrita en bloques completos con una sola operación, esta característica proporciona la ventaja

principal con respecto a su predecesora EEPROM, aumentando la velocidad de escritura y lectura de datos. La memoria flash de tipo NOR por otro lado no facilita la escritura de datos en bloques, sin embargo, a diferencia de la memoria flash de tipo NAND, permite la escritura y lectura independiente de un solo byte en alguna posición de la memoria. La memoria flash de tipo NAND permite mayor densidad de almacenamiento ya que puede almacenar más datos en un espacio de silicio más pequeño, ahorrando el costo de fabricación por bit de almacenamiento.

La memoria flash es principalmente utilizada cuando se desea realizar un almacenamiento masivo de información, como en el caso de las *tarjetas de memoria*, la *memoria flash USB* y la reciente *unidad de estado sólido (SSD, Solid State Drive)*. La falta de versatilidad para el acceso aleatorio de bajo nivel a una posición específica de la memoria, ha evitado que la memoria flash remplace a la memoria ROM como unidad de memoria principal en los sistemas informáticos. Una desventaja de la memoria flash es que sólo permite un número limitado de escrituras y borrados, generalmente entre 10 000 y 1 000 000 de ciclos.

La utilización de la tecnología NAND y NOR para almacenar la información está dividida entre los múltiples fabricantes, especialmente a la hora de elegir un sistema de archivos para el manejo de estas memorias. El sistema de archivos JFFS (*Journalling Flash File System*) por ejemplo, está basado en una estructura de registros para usarse sobre memorias flash de tipo NOR, sin embargo, en la actualidad se emplea el sistema de archivos FAT32 utilizado en casi todas las *tarjetas de memoria* y *memorias flash USB* ante la compatibilidad con el sistema operativo Windows.

Finalmente, a pesar de que en términos técnicos una memoria flash es un tipo de EEPROM, el término “EEPROM” es generalmente usado para referirse específicamente a la memoria EEPROM que es borrada en pequeños bloques, generalmente bytes. Debido a que los ciclos de borrado son más lentos, los accesos en grandes bloques de la memoria flash le confieren una significativa ventaja en cuanto a velocidad. El término “flash” se atribuye al hecho de que en este tipo de memorias el proceso de borrado es tan rápido como el flash de una cámara fotográfica.

### 2.5.3. Tarjeta de memoria microSD

Una *tarjeta de memoria* es un medio de almacenamiento de datos que conserva la información haciendo uso de una memoria flash y de un controlador que sirve como interfaz para la comunicación con esta. Existe una amplia variedad de formatos de tarjeta de memoria ofrecida por múltiples fabricantes. La *PC Card* fue uno de los primeros formatos de tarjeta de memoria, lanzada en el año 1990; ésta aún existe pero se usa principalmente para aplicaciones industriales y para conectar dispositivos de entrada/salida como el módem. Posteriormente aparecieron una serie de formatos de memoria más pequeños que la *PC Card*, entre ellos: *Multi Media Card (MMC)*, *Compact Flash*, *Smart Media*, *Secure Digital (SD)*, *miniSD* y *microSD*.

El estándar SD fue desarrollado por SandDisk, Panasonic y Toshiba en 1990 como una mejora evolutiva del formato MMC. El estándar SD es manejado por la *Asociación de tarjetas SD (SD Association)*, conformada por las tres compañías citadas anteriormente, y en la que también participan diversos fabricantes de dispositivos electrónicos. Inicialmente el formato SD compitió y coexistió con otros formatos, sin embargo, con el paso del tiempo se ha posicionado como el más utilizado en la actualidad, sobre todo después de la aparición de la miniSD y microSD, ofreciendo un reducido tamaño y una elevada densidad de almacenamiento ideal para aplicaciones portátiles.

El formato SD incluye tres versiones de tarjetas disponibles en tres tamaños. Las tres versiones son:

- *Standard Capacity (SDSC)*. Capacidad estándar.
- *High Capacity (SDHC)*. Alta capacidad.
- *Extended Capacity (SDXC)*. Capacidad extendida.

Las características de estas versiones se resumen en la tabla 2.4.

Versión	Capacidad	Sistema de archivos	Velocidad del Bus
SDSC	Menor a 2 GB	FAT12 FAT16	NS HS
SDHC	Mayor a 2 GB Hasta 32 GB	FAT 32	NS HS UHS-1
SDXC	Mayor a 32 GB Hasta 2 TB	exFAT	NS HS UHS-I

Tabla 2.4. Características de las versiones de memoria SD.

Las velocidades del Bus son las siguientes:

- *NS (Normal Speed)*. Velocidad estándar: 12.5 MB/s.
- *HS (High Speed)*. Velocidad alta: 25 MB/s.
- *UHS-I (Ultra High Speed)*. Ultra-alta velocidad I: 50 MB/s.

Los tres tamaños disponibles son:

- SD estándar.
- miniSD.
- microSD.

La *tarjeta de memoria SD* está básicamente conformada por una memoria flash y un controlador. Las operaciones de borrado, lectura, escritura y control de errores son completadas dentro de la tarjeta de memoria. Los datos son transferidos entre la tarjeta de memoria y el *host* (controlador anfitrión) mediante bloques de datos de 512 bytes, de esta manera, las tarjetas de memoria SD pueden ser consideradas como unidades de disco duro de pequeñas dimensiones y menor capacidad de almacenamiento.

El diagrama de bloques de una tarjeta de memoria microSD se muestra en la figura 2.73.

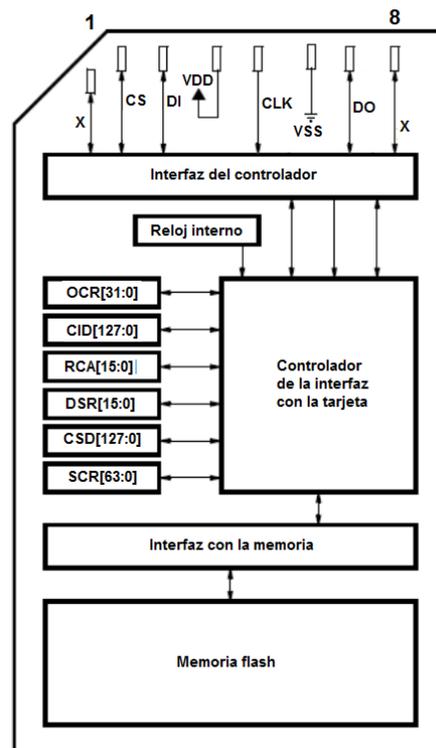


Figura 2.73. Memoria microSD.

La memoria microSD consiste de una interfaz de 8 terminales, un controlador de la tarjeta, una interfaz con la memoria y una memoria flash. La interfaz de 8 terminales permite el intercambio de datos entre un sistema conectado y el controlador de la tarjeta. El controlador puede leer y escribir datos de o hacia la memoria flash usando la interfaz con la memoria. Además de los elementos indicados se cuenta con diversos registros internos que almacenan el estado de la tarjeta. La tarjeta de memoria SD cuenta con 6 registros y un registro de estado. Los 6 registros son: OCR (*Operation Conditions Register*), CID (*Card Identification*), CSD (*Card Specific Information*), RCA (*Relative Card Address*), DSR (*Driver Stage Register*) y SCR (*SD Card Special Features*).

El controlador responde a dos tipos de solicitudes hechas por el host: de control y de datos. Las solicitudes de control establecen la operación del controlador y se permite el acceso a los registros de la tarjeta de memoria SD. Las solicitudes de datos se utilizan tanto para leer datos de la memoria flash como para escribir datos en ella.

#### 2.5.4. Comunicación con la memoria microSD

Las tarjetas de memoria microSD soportan dos tipos de comunicación, una de ellas se realiza empleando el estándar de comunicación SD desarrollado por la *SD Association* y, la segunda, es la comunicación en modo SPI. El estándar de comunicación SD cuenta con numerosas patentes y sólo se puede obtener la información técnica del estándar mediante una licencia adquirida a través de la *SD Association*, sin embargo, todas las tarjetas de memoria SD soportan el antiguo modo de comunicación SPI que utiliza únicamente 4 señales para la comunicación con la tarjeta, estas señales son: señal de reloj (CLK), dato de entrada (DI), dato de salida (DO) y selección del circuito integrado (CS). La utilización de estas señales facilita la interfaz con una amplia variedad de microcontroladores que emplea la misma señalización en el protocolo SPI. Como es de pensarse, el modo de comunicación mediante SPI no ofrece las mismas características que el modo SD, la más notable es la reducción de la tasa de transferencia de datos.

El modo de comunicación SPI, como todos los estándares, se rige por ciertos criterios, sin embargo, la documentación en los medios de información es limitada y simplemente se especifican criterios generales para entablar una comunicación con una tarjeta de memoria SD. Estos criterios generales para la comunicación mediante el modo SPI con una tarjeta de memoria SD son los siguientes:

1. Todo comando válido consta de 48 bits.
2. Existen comandos de control y de datos.
3. Los comandos de detección de errores no son soportados.
4. Todas las operaciones son controladas por el dispositivo maestro mediante el manejo de la señal CS.
5. A cada comando enviado por el maestro le corresponde una respuesta generada por la tarjeta de memoria SD, exceptuando el comando CMD0.
6. La longitud de los bloques de datos es de 512 Bytes.

El formato de un comando en el modo de comunicación SPI se muestra en la figura 2.74. Un comando inicia con el MSB. El bit 47 es un bit de inicio siempre con valor 0, el bit 46 es el bit de transmisión y tiene un valor 1 para comandos y 0 para respuestas. Los bits 45:40 corresponden al número de comando transmitido, consta de 6 bits y puede tener algún valor entre 0 y 63. Los 32 bits siguientes especifican el argumento del comando, sin embargo, es importante destacar que no todos los comandos poseen un argumento. A continuación se tienen los bits 7:1 concernientes al código CRC (*Cyclic Redundancy Code*); este código es un código de detección de errores empleado para detectar cambios accidentales en los datos. Por default, la tarjeta de memoria SD ignora los bits del CRC para todos los

comandos, exceptuando el comando CMD8. Finalmente, se tiene un bit de parada con el valor 1.



Figura 2.74. Formato de comando válido en el modo SPI.

Los comandos más importantes para la comunicación con una tarjeta de memoria SD utilizando el modo SPI son los siguientes:

- CMD0. Reinicia todas las tarjetas a un estado de reposo o espera (*idle state*).
- CMD8. Envía a la tarjeta de memoria las condiciones de operación del bus, como el voltaje manejado por el maestro. La respuesta para este comando es la R7.
- CMD17. El envío de este comando permite la lectura de un bloque de datos; para el caso de las tarjetas de memoria SDSC el tamaño del bloque se establece con el comando *SET BLOCK LEN* (CMD16), mientras que para las tarjetas SDXC el bloque es de 512 Bytes. La respuesta para este comando es la R1.
- CMD18. Con el uso de este comando se permite la lectura de múltiples bloques de datos hasta la interrupción ocasionada por el envío del comando *STOP TRANSMISSION* (CMD12). La respuesta para este comando es la R1.
- CMD24. El envío de este comando permite la escritura de un bloque de datos en la tarjeta de memoria SD. El tamaño del bloque de datos es de 512 Bytes para las tarjetas SDXC y se fija con el comando CMD16 para las tarjetas SDSC. La respuesta para este comando es la R1.
- CMD25. Facilita la escritura de múltiples bloques de datos en la tarjeta de memoria SD hasta la interrupción generada con el comando CMD12. La respuesta para este comando es la R1.
- CMD58. Mediante el envío de este comando se permite la lectura del registro OCR (*Operation Condition Register*) de la tarjeta de memoria SD para identificar si ésta cumple con el rango de voltaje suministrado por el maestro. Cuando existe una incompatibilidad con el rango de voltaje el proceso de inicialización de la tarjeta SD se detiene. La respuesta para este comando es la R3.

- ACMD41. Envía al maestro información acerca de la capacidad soportada, además, envía el contenido del registro OCR para verificar si el proceso de inicialización ha terminado. La respuesta para este comando es la R3.

Las respuestas son generadas por la tarjeta de memoria al recibir un comando por parte del maestro, la longitud de esta depende del tipo de respuesta generada. Todas las respuestas comienzan con un bit de inicio siempre con un valor 0, a continuación se tiene el bit de dirección de transmisión, que para el caso de las respuestas siempre es 1. Los bits subsiguientes varían conforme al tipo de respuesta generada, y finalmente, se tiene el bit de parada siempre con un valor 1. Las respuestas utilizadas en el modo de comunicación SPI son las siguientes:

- R1 (Respuesta normal de comando). La respuesta R1 consta de 48 bits. El MSB corresponde al bit de inicio, el bit 46 corresponde al bit de dirección de transmisión. Los bits 45:40 indican el número de comando al cual se dirige la respuesta, este valor puede ser cualquiera entre 0 y 63. Los 32 bits 39:8 proporcionan información correspondiente al estado de la tarjeta, los bits 7:0 contienen el CRC y, el LSB es un bit de parada. El formato de la respuesta R1 se muestra en la figura 2.75.



Figura 2.75. Respuesta R1.

- R2 (Estado de los registros CID y CSD). La respuesta R2 consta de 136 bits. Los bits 135 y 134 corresponden respectivamente al bit de inicio y bit de dirección. Los bits 133:128 están reservados con el valor 0b111111. El contenido de las posiciones 127:1 varía dependiendo del comando recibido; contienen la información del registro CID de la tarjeta de memoria como respuesta a los comandos CMD2 y CMD10, por otro lado, contienen la información del registro CSD de la tarjeta de memoria como respuesta al comando CMD9. Como se puede observar en la figura 2.76, los registros CID y CSD constan de 128 bits, sin embargo, el LSB que está reservado es omitido. El último bit de la respuesta R2 es el bit de parada. La figura 2.76 muestra el formato de esta respuesta.



Figura 2.76. Respuesta R2.

- R3 (Estado del registro OCR). La respuesta R3 consta de 48 bits. Los bits 47 y 46 corresponden respectivamente al bit de inicio y bit de dirección. Los bits 45:40 están reservados con el valor 0b111111, los 32 bits siguientes, cuyas posiciones comprenden los bits 39:8 contienen el estado del registro OCR. Los bits 7:1 están reservados con el valor 0b1111111 y, finalmente, el LSB es el bit de parada con valor 1. El formato de la cadena de datos de la respuesta R3 se muestra en la figura 2.77.

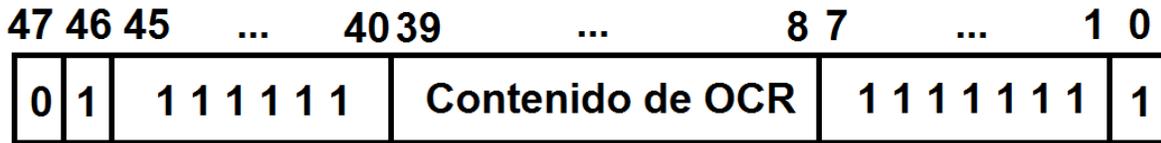


Figura 2.77. Respuesta R3.

- R7 (Condiciones de la interfaz entre la tarjeta de memoria y el maestro). Esta respuesta también consta de 48 bits. Los bits 47 y 46 corresponden respectivamente a los bits de inicio y de dirección. Los bits 45:40 indican el número de comando al cual va dirigida la respuesta, los 20 bits siguientes cuyas posiciones comprenden la 39:20 están reservados con el valor 0x00000. Los 4 bits 19:16 contienen la información sobre el rango de alimentación de la tarjeta como respuesta ante el comando CMD8. Los 8 bits 15:8 contienen un patrón de verificación. Los bits 7:0 contienen el CRC y, finalmente, el LSB corresponde al bit de parada. El formato de la respuesta R7 se muestra en la figura siguiente.



Figura 2.78. Respuesta R7.

*Inicialización de la tarjeta de memoria SD*

Para que la tarjeta de memoria pueda ser utilizada y se puedan realizar operaciones de lectura y escritura de datos, primeramente se debe seguir un procedimiento de inicialización. Cuando la tarjeta de memoria SD es energizada, originalmente opera en el modo de comunicación SD, por este motivo debe ser activada en el modo SPI. Para inicializar la tarjeta de memoria en el modo SPI la señal CS debe ser mantenida con un nivel lógico bajo mientras el maestro envía el comando de reinicio CMD0. Si la tarjeta de memoria se activa correctamente en el modo SPI, transmitirá la respuesta R1.

El paso siguiente consiste en el envío de un comando para verificar la compatibilidad del bus. Mediante la generación del comando CMD8 por parte del maestro, se envía información a la tarjeta de memoria acerca de las condiciones de operación del bus, entre ellas el voltaje de operación. La tarjeta de memoria a su vez, mediante la

respuesta R7, envía la información sobre el voltaje soportado; el maestro analiza la compatibilidad y en caso de existir, se continúa con el proceso de inicialización. En este caso se habla de la segunda versión de las tarjetas de memoria. Cuando la respuesta tiene un formato ilegal, entonces se habla de la versión 1 de tarjetas de memoria SD, en consecuencia, se puede enviar el comando CMD58 que también verifica la compatibilidad de voltajes, si el voltaje es correcto se continúa con el proceso de inicialización de la tarjeta.

El comando ACMD41 es utilizado para verificar a través del bit "in\_idle\_state", si la tarjeta ha completado el proceso de inicialización.

El diagrama de flujo para la inicialización de una tarjeta de memoria SD se muestra en la figura 2.79.

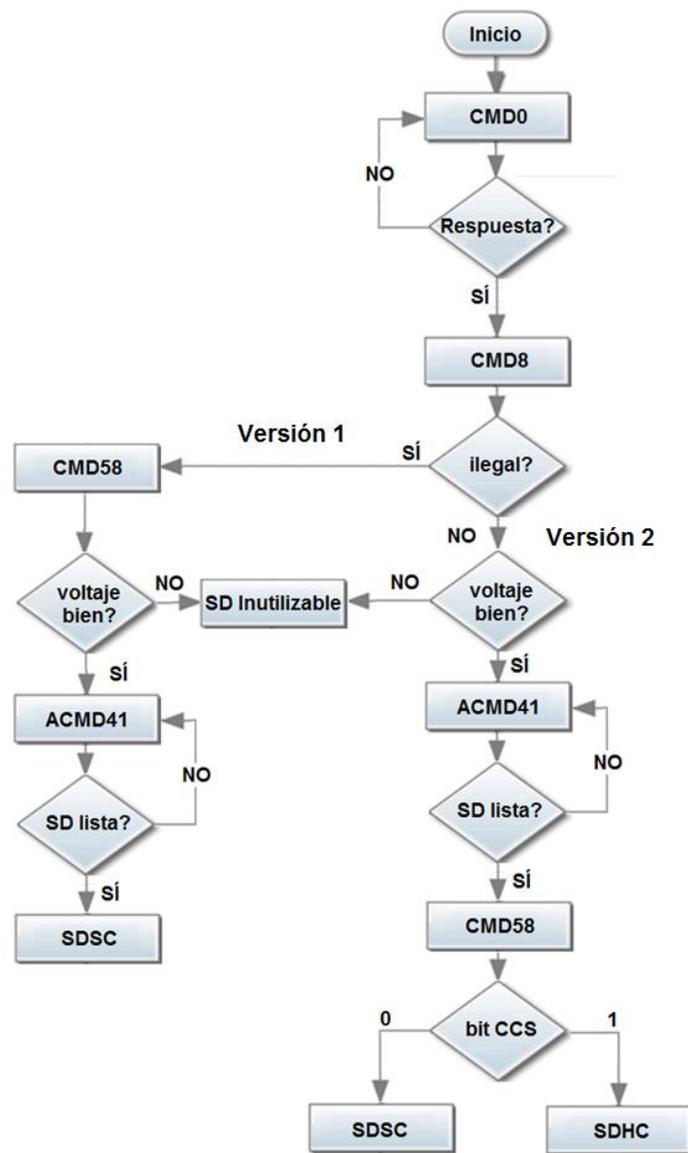


Figura 2.79. Inicialización de la tarjeta de memoria SD.

*Lectura de datos*

- Lectura de un bloque de datos. Para realizar una operación de lectura de un bloque de datos, en primer lugar el maestro debe enviar a la tarjeta de memoria el comando CMD17, a continuación, ésta generará la respuesta R1 indicando que el comando es válido (cuando no es válido se activa el bit de error, *illegal-command*, en el registro de estado de la tarjeta y no se toma acción alguna, lo mismo sucede en todas las operaciones que se describen en seguida al recibir un comando inválido), después de esto, la tarjeta de memoria enviará un bloque de datos a través del bus de comunicación. Un bloque de datos está acompañado por el CRC que consta de 16 bits. En el modo SPI el CRC es ignorado. La figura 2.80 muestra esquemáticamente una operación de lectura de un bloque.

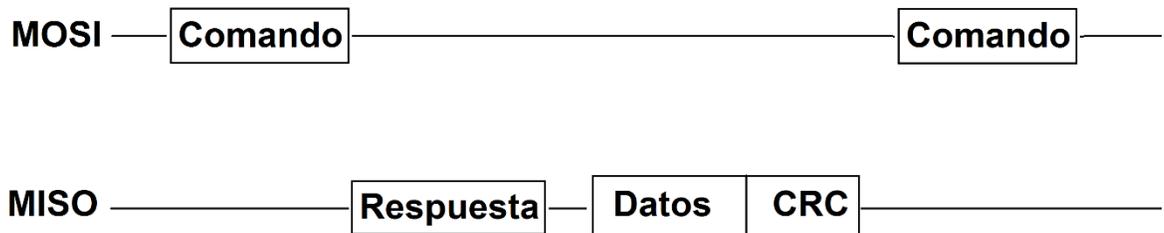


Figura 2.80. Lectura de un bloque de datos.

- Lectura de múltiples bloques de datos. Para realizar una operación de lectura de múltiples bloques de datos se sigue el siguiente procedimiento: primero el maestro envía el comando CMD18, en seguida la tarjeta de memoria contesta con la respuesta R1, verificando que el comando es válido, a continuación envía los bloques de datos. Cada bloque de datos está acompañado del CRC y la transmisión continua hasta que el maestro genere el comando CMD12, en ese momento la tarjeta responde al comando de parada y detiene el envío de datos. La figura 2.81 ejemplifica esquemáticamente la operación descrita.

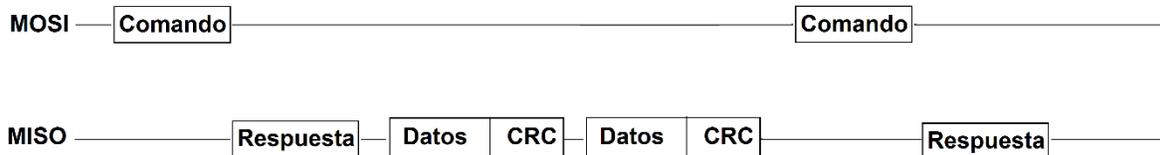


Figura 2.81. Lectura de múltiples bloques de datos.

*Escritura de datos*

- Escritura de un bloque de datos. Para realizar una operación de escritura de un bloque de datos, el maestro envía el comando CMD24 a la tarjeta de memoria, ésta verifica la validez del comando y contesta con la respuesta R1, señalando que se tiene un comando válido; a continuación, la tarjeta de

memoria esperará por un bloque de datos enviados por el *maestro*. Un bloque de datos enviado por el *maestro* debe ser antecedido por un bit de inicio (*Start Block Token*), el *maestro* envía el bloque de datos y en seguida la tarjeta de memoria envía una respuesta de datos (*Data Response*). En los instantes siguientes, la tarjeta de memoria se encontrará escribiendo el dato y enviará la señal *busy* indicando que se encuentra realizando operaciones de escritura. La figura 2.82 ejemplifica esquemáticamente una operación de escritura de datos de un bloque.

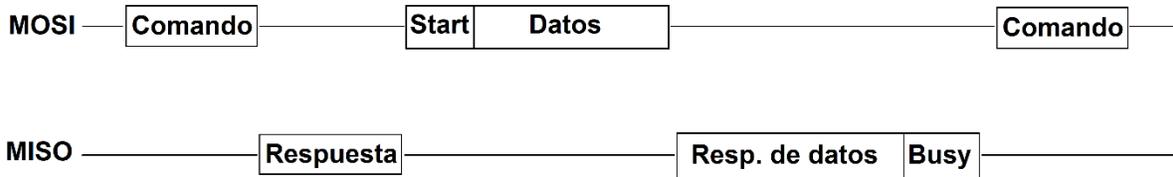


Figura 2.82. Escritura de un bloque de datos.

- Escritura de múltiples bloques de datos. Para realizar una operación de escritura de múltiples bloques de datos se sigue el siguiente procedimiento: el maestro envía el comando CMD25, la tarjeta contesta con la respuesta R1 verificando la validez del comando, el maestro envía el bloque de datos antecedido por el bit de inicio, la tarjeta envía una respuesta de datos finalizada con el bit *busy*. Para finalizar la transacción el maestro envía el comando CMD12, la tarjeta envía una respuesta a este comando y la transferencia de datos se termina. La figura 2.83 ejemplifica el proceso descrito.

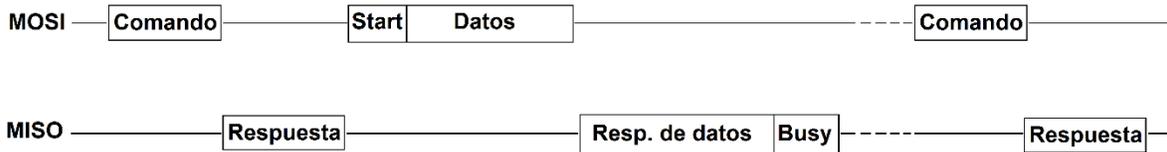


Figura 2.83. Escritura de múltiples bloques de datos.

### 2.5.5. Sistema de archivos FAT

El núcleo de la tarjeta de memoria SD es una memoria flash, que por ser un medio de almacenamiento de datos, direccionable por bloques, requiere de un sistema de archivos para ser utilizada. La mayoría de las tarjetas de memoria (incluyendo las que no son SD) vienen pre-formateadas con una partición que contiene un sistema de archivos. En general, todos los dispositivos con capacidades masivas de almacenamiento de datos requieren de un sistema de archivos para hacer eficiente la organización y acceso a los mismos.

Debido a que el formato de tarjeta de memoria predominante en el mercado es el SD en sus diferentes tamaños, la mayoría de las tarjetas de memoria utilizan el sistema de archivos FAT (File Allocation Table). El sistema de archivos FAT tuvo

sus orígenes a finales de 1970 y principios de 1980 y fue el sistema de archivos soportado por el sistema operativo MS-DOS de Microsoft. El sistema de archivos FAT ha evolucionado, pues en principio era utilizado para el manejo de información en los discos flexibles cuya capacidad era muy limitada, sin embargo, con el paso de los años FAT se adaptó a los nuevos dispositivos que cuentan con mayor capacidad de manejo de datos.

Actualmente muchas memorias SD utilizan el sistema de archivos FAT32 y esto debido al continuo mejoramiento en cuanto a la capacidad de almacenamiento de las tarjetas de memoria, no obstante lo anterior, siguen existiendo aunque cada vez entran más en desuso, los sistemas de archivos FAT16 y FAT12. La diferencia básica entre éstos tres sistemas de archivos radica en el número de bits de entrada para acceder a la *Tabla de asignación de archivos* o FAT; como es de suponerse, las versiones más antiguas utilizan una menor cantidad de bits, puesto que las capacidades de almacenamiento son inferiores, de esta manera, FAT12, FAT16 y FAT32 utilizan respectivamente 12, 16 y 32 bits para acceder a la FAT.

En cuanto a hardware se refiere, los datos se manejan en pequeños bloques constituidos por 512 bytes; cada uno de estos bloques recibe el nombre de *sector*. Por otro lado, en términos de software, el sistema de archivos segmenta el medio de almacenamiento de datos en unidades mínimas denominadas *clústers* y constituyen la unidad de almacenamiento más pequeña que se puede asignar a un archivo y direccionar en la FAT. Un clúster agrupa un número determinado de sectores contiguos, este número siempre es una potencia de 2 y entonces el número de sectores abarcados en un clúster puede ser: 1, 2, 4, 8, etc.

El sistema de archivos FAT está conformado por cuatro regiones, estas regiones son las siguientes:

- R0. *Boot sector* (*Sector de arranque*).
- R1. *FAT*.
- R2. *Directorio raíz* (excepto para FAT32 donde está compartida con R3).
- R3. *Región de datos*.

La figura 2.84 ilustra las regiones del sistema de archivos FAT.

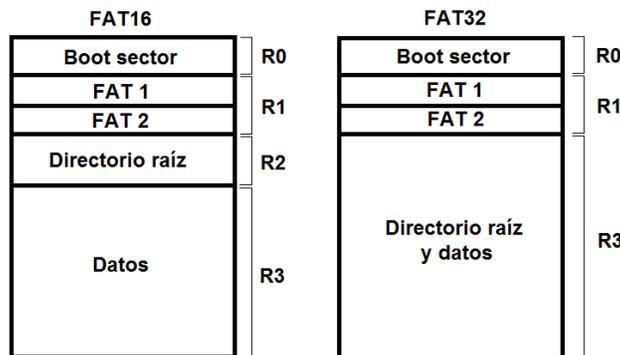


Figura 2.84. Sistema de archivos FAT.

### *Boot sector*

La primera estructura de datos del sistema de archivos FAT es el *Boot sector* o sector de arranque, localizado en el primer sector del volumen. El sector de arranque contiene información que es de vital utilidad para tener acceso al volumen, alguna de esta información es la siguiente: número de bytes por sector (casi siempre 512), número de sectores por clúster, número de sectores reservados, número de FAT's, número total de sectores en toda la unidad, etc.

### *FAT*

La tabla FAT contiene la referencia a cada uno de los *clústers* dentro de la unidad de almacenamiento manejada, cada posición de la FAT se corresponde directamente con un clúster dentro del volumen. Cualquier posición en la tabla puede contar únicamente con los siguientes tres valores.

1. La marca especial 0 para indicar que se trata de un clúster libre o vacío, en otras palabras, el clúster al cual hace referencia no contiene datos.
2. Una marca especial para indicar que se trata del último clúster de un archivo almacenado en la unidad, o sea, que el clúster al cual hace referencia contiene la parte final del archivo. Para el caso de FAT32 la marca es algún valor entre 0xFFFFFFFF8 y 0xFFFFFFFF.
3. Cualquier otro valor se interpreta como el clúster siguiente de un archivo. Por ejemplo, si una posición de la FAT contiene el valor 356, entonces, en el clúster 356 se almacena la parte contigua de un archivo.

Por motivos de seguridad, la región FAT contiene dos tablas FAT en donde la segunda es una copia idéntica de la primera.

### *Directorio raíz*

La región del directorio raíz se utiliza tanto en el sistema de archivos FAT16 como en FAT32, sin embargo, en FAT16 el espacio en la unidad es definido mientras que en FAT32 el directorio raíz es variable y depende de la cantidad de directorios y archivos creados. En el caso de FAT32 el directorio raíz inicia a partir del primer clúster de la región de datos.

En esta zona se localiza la disposición de los datos presentes en la unidad, dichos datos son almacenados en una estructura de carpetas y archivos. Los directorios son tipos especiales de archivos cuyo atributo de directorio se encuentra activado. El contenido de este archivo tipo directorio son las entradas correspondientes a cada elemento alojado en él.

La entrada para algún archivo o carpeta consta de 32 bytes, de los cuales, ocho corresponden al nombre del archivo, tres a la extensión del archivo, uno al byte de atributos, ocho son reservados, dos bytes para la parte más significativa del primer clúster, cuatro bytes reservados, dos bytes para la parte menos significativa del

primer clúster y finalmente, cuatro para el tamaño del archivo. Esta información es utilizada por el sistema operativo para localizar el archivo deseado; obteniendo el número del primer clúster de un archivo encontrará el resto entrando a la tabla FAT.

### *Región de datos*

En esta región se encuentra la información correspondiente a los archivos y carpetas almacenadas por el usuario. Todos los datos almacenados tienen el formato "*Little endian*", es decir, que el byte menos significativo se almacena en la primera localidad de memoria. Vale la pena mencionar que los archivos son almacenados en cantidades enteras de clústers, de esta manera, si un archivo cuya tamaño es igual a 2 KB es almacenado en un clúster de 32 KB, 30KB quedarán sin utilizarse.

## **2.6. Sistema de Posicionamiento Global y especificación NMEA 0183**

Todos los sistemas utilizados para el monitoreo de fenómenos sísmicos utilizan un elemento para la generación de la referencia de tiempo de los eventos registrados. Para realizar esta actividad es común utilizar un receptor GPS (Sistema de Posicionamiento Global), el cual, a través del enlace con un número determinado de satélites que orbitan alrededor de la Tierra, ofrece una referencia de tiempo caracterizada por su alta precisión. El conjunto de datos proporcionados por el GPS recibe el nombre de *efemérides*.

El Sistema de Posicionamiento Global, es un sistema de navegación compuesto de una red de satélites, radio bases y receptores; su objetivo es la determinación de las coordenadas espaciales de un punto situado sobre la superficie terrestre, aún si este se encuentra en movimiento y a cualquier hora del día. La utilización en sistemas para el monitoreo de actividad sísmica se debe a que además de las coordenadas proporcionadas, también se puede obtener el *Tiempo Universal Coordinado* (en inglés *Coordinated Universal Time, UTC*), que es el estándar de tiempo utilizado mundialmente para registrar los eventos sísmicos. El tiempo que maneja el sistema GPS está definido por el reloj atómico de Cesio de la Estación de Control Maestra localizada en Colorado Springs en Colorado, Estados Unidos.

GPS está conformado por tres partes o segmentos principales:

- *Segmento espacial*, conformado por satélites que orbitan alrededor de la Tierra.
- *Segmento de control*, compuesto por estaciones ubicadas cerca del ecuador terrestre para controlar a los satélites.
- *Segmento de usuario*, que engloba a cualquier persona u organización que reciba y utilice las señales GPS.

### Segmento espacial

El sistema espacial consiste de una red de 24 satélites, denominada *NAVSTAR* (*Navigation System Using Timing and Ranging*) que orbitan alrededor de la Tierra a una altura aproximada de 20 200 [km]. Los satélites se ubican en 6 órbitas prácticamente circulares, con una inclinación de  $55^\circ$  con respecto al plano del Ecuador y con una distribución aproximadamente uniforme; con 4 satélites en cada órbita. El periodo de rotación alrededor de la Tierra es de aproximadamente 12 horas y también existen más satélites que se encuentran desactivados y disponibles como reemplazo.

El segmento espacial está diseñado de tal forma que se pueda contar con un mínimo de 4 satélites visibles por encima de un ángulo de elevación de  $15^\circ$  en cualquier punto de la superficie terrestre, durante las 24 horas del día. El tiempo máximo de observación de un satélite es de hasta 4 horas y 15 minutos. Para la mayoría de las aplicaciones, el número mínimo de satélites visibles deberá ser de cuatro. La experiencia ha demostrado que la mayor parte del tiempo hay por lo menos 5 satélites visibles por encima de los  $15^\circ$ , y muy a menudo hay 6 o 7 satélites visibles. En la figura 2.85 se muestra esquemáticamente la red de satélites NAVSTAR.

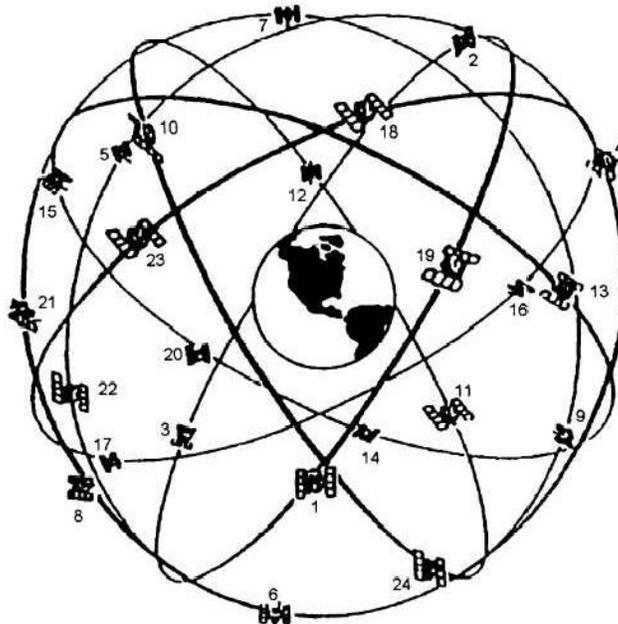


Figura 2.85. Red NAVSTAR.

Cada satélite GPS lleva a bordo varios relojes muy precisos. Estos relojes operan en una frecuencia fundamental de 10.23 [MHz], la cual se emplea para generar las señales de tiempo transmitidas por el satélite. Los satélites transmiten constantemente en dos ondas portadoras. Estas ondas se encuentran en la banda L (utilizada para transmisiones de radio comprendiendo el rango entre 1000 [MHz] y 3000 [MHz]) y viajan a la Tierra a la velocidad de la luz. Dichas ondas portadoras

se derivan de la frecuencia fundamental generada por un reloj atómico muy preciso. Las dos ondas portadoras son:

- Portadora L1. Tiene una frecuencia de 1575.42 [MHz], con utilización en aplicaciones civiles.
- Portadora L2. Tiene una frecuencia de 1227.60 [MHz], con utilización en aplicaciones militares.

La portadora L1 es modulada por dos códigos: el código C/A (*Coarse Acquisition*) o Adquisición Gruesa a una frecuencia de 1.023 [MHz] y el código P (*Precision*) o de Precisión a una frecuencia de 10.23 [MHz]. L2 es modulada únicamente con el código P a una frecuencia de 10.32 [MHz]. Las frecuencias de señales portadoras y moduladoras se obtienen ambas de la frecuencia del reloj que el satélite lleva a bordo tal como se esquematiza en la figura 2.86.

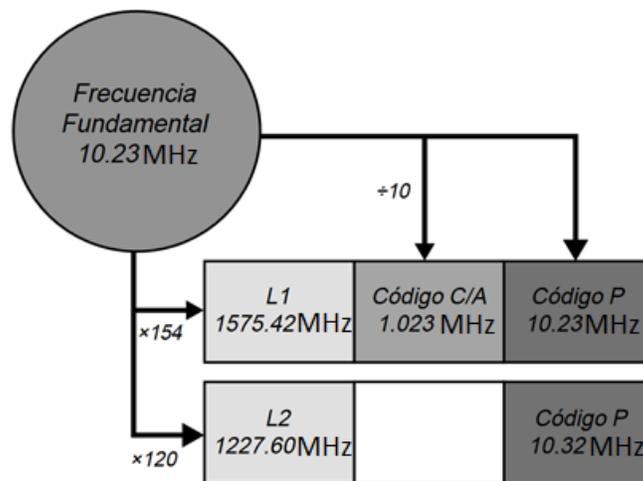


Figura 2.86. Estructura de la señal GPS.

La identificación de un satélite en particular puede hacerse gracias a su PRN (*Pseudo Random Noise*) o *Ruido Pseudo Aleatorio*, denominado de esta manera porque debido a su naturaleza aparenta ser una señal de ruido aleatoria, sin embargo, en realidad se trata de un código generado con algoritmos complejos.

### Segmento de control

El segmento de control consiste de una Estación de Control Maestro y 4 estaciones de observación. En la figura 2.87 se puede observar la ubicación geográfica de cada estación; la Estación de Control Maestro se localiza en Colorado Springs mientras que las estaciones de observación se ubican en regiones cercanas al ecuador terrestre.



Figura 2.87. Estaciones del segmento de control.

El segmento de Control rastrea los satélites GPS, actualiza su posición orbital y calibra y sincroniza sus relojes. Otra función importante consiste en determinar la órbita de cada satélite y predecir su trayectoria para las siguientes 24 horas. Esta información es cargada a cada satélite y posteriormente transmitida desde allí. Esto permite al receptor GPS conocer la ubicación de cada satélite.

Las señales de los satélites son leídas desde las estaciones: Ascensión, Diego García, Hawaii y Kwajalein. Estas mediciones son entonces enviadas a la Estación de Control Maestro, donde son procesadas en cada satélite. La información es enviada posteriormente a las cuatro estaciones de observación equipadas con antenas de tierra y de allí cargadas a los satélites.

### *Segmento de usuario*

El segmento de usuarios comprende a cualquiera que reciba las señales GPS con un receptor, determinando su posición y/o la hora. Algunas aplicaciones típicas dentro del segmento usuario son: la navegación en tierra para excursionistas, ubicación de vehículos, topografía, navegación marítima y aérea, control de maquinaria, etc.

### *Funcionamiento*

GPS depende de que cada satélite en la constelación transmita su posición exacta y una señal de tiempo extremadamente precisa a los receptores en tierra. Dada esta información, los receptores GPS pueden calcular su distancia al satélite, y combinando esta información de cuatro satélites, el receptor puede calcular su posición exacta usando un proceso llamado trilateración.

Si uno conoce la distancia a un satélite uno sabe que su posición se encuentra sobre una esfera con centro en el satélite y con radio igual a la distancia conocida. Si se obtiene la misma información de un segundo satélite se puede dar por hecho que la posible posición es aquella región que tienen en común ambas esferas. Agregando

un tercer satélite se puede precisar aún más la ubicación a la región donde las tres esferas se cruzan.

Para determinar cuál de los dos puntos (ver figura 2.88) representa la ubicación correcta, se puede tomar una cuarta medida, pero generalmente uno de los dos puntos obtenidos de tres satélites representa una posición absurda (por ejemplo el espacio abierto) o con movimiento imposiblemente rápido, por lo cual se puede eliminar sin necesidad de la cuarta medida. Sin embargo, la cuarta medida aún es necesaria.

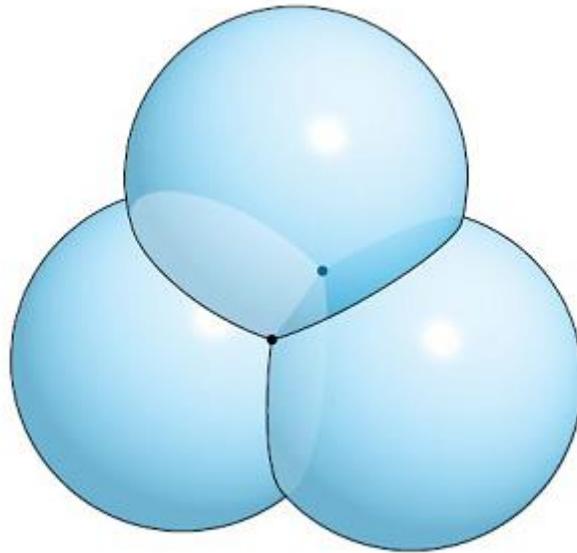


Figura 2.88. Intersección de tres esferas.

La distancia a los satélites se calcula midiendo el tiempo que toma a la señal el llegar del satélite al receptor. Debido a que las señales de radio viajan a la velocidad de la luz, los tiempos en tránsito de cada satélite al receptor son extremadamente pequeños y se necesitan dispositivos de cronometraje muy preciso para medirlos con exactitud, por lo cual surge la necesidad de llevar relojes atómicos en los satélites. Sin embargo, los receptores GPS no cuentan con relojes atómicos, lo que introduce errores en ese lado del sistema, y aún errores de cronometraje pequeños pueden resultar en grandes errores de posición.

Si se realizan cuatro mediciones de distancia del receptor GPS hacia cuatro satélites diferentes, las 4 esferas con radio igual a las medidas tomadas, sólo intersectan en un solo punto. Debido a que el error del receptor es el mismo para las cuatro medidas, un procesador en el receptor puede calcular una corrección que haga que las cuatro esferas “*crucen*”, y aplicar la corrección a las medidas para obtener la posición correcta. La cuarta medida también permite al sistema calcular una posición en tres dimensiones que incluye no solo la latitud y longitud, sino también la altitud. Las medidas de altitud, no obstante, se reportan con referencia a un modelo matemático de la tierra para poder expresarla normalmente en relación al

nivel del mar. El modelo es sólo una aproximación por lo cual las medidas de altitud son menos precisas que las de latitud y longitud, y los errores son diferentes en distintas partes del planeta.

### *Cálculo de la distancia*

Cuando se diseñó el sistema GPS se estableció que el código C/A fuera de libre adquisición, es decir, no reservado para uso militar, por lo tanto el problema a resolver radica en medir la distancia entre los satélites y este código. Se presenta un problema similar al que resuelven los distanciómetros electrónicos, estos aparatos emiten una onda homogénea de frecuencia conocida, la cual se refleja en un prisma colocado en el otro extremo del segmento a medir; el rebote es recibido por el aparato, que mide el desfase, lo convierte en tiempo y posteriormente en distancia equivalente.

Para calcular la distancia que hay entre un satélite en particular y el receptor GPS que está siendo utilizado, se requiere primero obtener el tiempo que la señal demora en arribar desde el transmisor hasta el receptor, para ello, ambas partes requieren contar con “relojes” de alta precisión para que el cálculo de la distancia entre ambos puntos no difiera mucho de la realidad. Los elementos utilizados como relojes son osciladores de frecuencias muy estables capaces de señalar medidas de tiempo del orden de  $10^{-14}$  [s] en los satélites y  $10^{-8}$  [s] en los receptores.

Como se ha mencionado, existe una frecuencia fundamental generada por el oscilador del satélite, de ella se deriva la portadora L1 y L2, también los códigos C/A y P. Tomando como ejemplo los receptores de uso civil, éstos utilizan la portadora L1 modulada bajo la señal C/A. El código consiste de una cadena digital de unos y ceros. Al multiplicar la onda portadora por el código, esta no se afecta en presencia de un uno lógico pero, la señal se invierte en la presencia de un cero, el resultado es una onda deformada, el PRN (que aparentemente se trata de un ruido aleatorio), que es captada por el receptor.

Cada satélite cuenta con un C/A diferente, generando una modulación propia y exclusiva. La señal resultante representa un PRN distintivo de cada satélite, pero además, el PRN se repite cada milisegundo y le corresponde un instante determinado que debe ser común a todo el sistema (toda la constelación de satélites). Por su parte, los receptores GPS tienen almacenadas en un elemento de memoria, las réplicas de todos los PRN del sistema, así, cuando recibe la emisión satelital puede efectuar el reconocimiento del satélite correspondiente.

Tanto el receptor como el satélite generan independientemente el PRN en un instante determinado de tiempo; cuando el receptor GPS tiene visibilidad del satélite, recibe el PRN con algún desfase de tiempo  $\Delta t$ , con respecto al que él mismo generó (figura 2.89). Para obtener la distancia entre ambos puntos se multiplica el desfase de tiempo  $\Delta t$  por la velocidad de la luz,  $3 \times 10^8 \left[ \frac{m}{s} \right]$ . El resultado obtenido es una *pseudodistancia*, lo cual es lógico puesto que la sincronización entre los

relojes del satélite y del receptor no puede ser perfecta, basta tener en cuenta la diferencia existente entre la precisión de cada uno de ellos. La consecuencia es que la distancia calculada difiere en una longitud  $\delta_e$  con respecto a la distancia real.

Puesto que el receptor GPS siempre contará con el mismo error debido a su reloj, independientemente del satélite con el que se enlace, se puede obtener la distancia a un cuarto satélite y la ubicación exacta del receptor GPS se obtendrá al resolver un sistema de cuatro ecuaciones y 4 incógnitas, a saber: las coordenadas  $x$ ,  $y$ ,  $z$  y la distancia de error  $\delta_e$ .

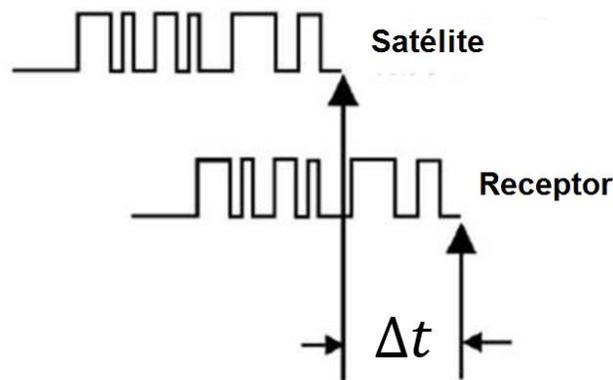


Figura 2.89. Obtención de la distancia.

Otros errores tienen influencia en las medidas de las señales de los satélites GPS, entre ellos están la interferencia atmosférica y reflejos de obstáculos en la tierra como árboles y edificios. En cuanto a la interferencia atmosférica se puede mencionar la disminución de la velocidad de la señal al pasar por la ionósfera, (recordando que la velocidad de la luz solo es constante en el vacío), efecto similar a la refracción producida cuando la luz atraviesa un bloque de vidrio. También afecta la variación de la densidad de la ionosfera ocasionada por el Sol. Durante la noche la influencia es mínima, además la densidad de la ionosfera varía con los ciclos solares (actividad de las manchas solares). Igualmente el vapor de agua contenido en la atmósfera puede afectar la señal GPS. Otra fuente de error es el ocasionado por la multitrayectoria de la señal. La señal del satélite no viaja directamente al receptor GPS sino que llega primero a un objeto reflectante como lagos y edificios y posteriormente arriba al receptor GPS. La manera de disminuir este error es con la utilización de una antena externa, incluyendo además un plano de tierra, el cual evita que señales espurias lleguen a la antena.

### *Especificación NMEA 0183*

Cada fabricante tiene sus propios formatos para proporcionar las *efemérides* y por lo tanto, al utilizar distintas versiones de receptores se presenta la limitante de falta de compatibilidad para el compartimiento de la información; para vencer esta limitante, varios grupos de investigación se han dado a la tarea de desarrollar

estándares de usuario. Algunos de los protocolos estándares son: RINEX, NGS-SP3, RTCM SC-104 y NMEA 0183, de éstos, el NMEA 0183 resulta ser un estándar ampliamente usado por diversos fabricantes de receptores GPS.

NMEA es la abreviatura de *National Marine Electronics Association*. Es una asociación fundada en el año 1957 por un grupo de fabricantes de electrónica, con la finalidad de obtener un sistema de comunicación común entre las diferentes marcas de electrónica naval. Poco a poco se fueron sumando todos los fabricantes a este estándar, además de organizaciones oficiales y gubernamentales. NMEA fue creado para el intercambio de información digital entre productos electrónicos marinos.

El estándar NMEA 0183 utiliza un protocolo de comunicaciones serie ASCII, el cual define como se transmiten los datos en una cadena de un “hablador” a varios “oyentes” a la vez, los cuales son transmitidos mediante una comunicación serial.

Las características generales de los datos proporcionados son las siguientes:

- Los datos son enviados en forma de cadenas.
- Cada cadena comienza con el signo “\$” y termina con un retorno de carro <CR> y siguiente línea <LF> (ASCII 13 y 10 respectivamente).
- El signo “\$” está seguido por un campo de 5 caracteres, generalmente en mayúsculas, que identifica al hablante (los dos primeros caracteres), al tipo de dato y el formato de los campos sucesivos (los últimos tres caracteres).
- El último campo de cualquier cadena es una suma de control, precedido por un carácter delimitador “\*”.

Los mensajes manejados por el estándar NMEA 0183 son los siguientes:

- *GGA, Global Positioning System Fixed Data*. Proporciona datos referentes al tiempo y la posición fijos.
- *GLL, Geographic Position*. Proporciona datos de latitud, longitud, hora UTC de fijación de la posición y el estado.
- *GSA, Active Satellites*. Contiene el modo de funcionamiento del receptor GPS y los satélites utilizados.
- *GSV, Satellites in View*. Contiene el número de satélites GPS en vista, número de identificación de los satélites, elevación y valores SNR (*Signal to Noise Ratio*).

- MSS, *MSK receiver signal*. Contiene información referente a la relación señal a ruido, intensidad de la señal, frecuencia.
- RMC, *Recommended Minimum Specific Data*. Contiene información referente a la hora UTC, fecha, rumbo y datos de posición.
- VTG, *Course over ground and ground speed*. Contiene información del curso y velocidad con respecto al suelo.

La cadena siguiente es un ejemplo del formato manejado por el estándar NMEA 0183:

```
$GPRMC,044235.000,A,4322.0289,N,00824.5210,W,0.39,65.46,020911,,A*44
```

La información proporcionada por esta cadena es la siguiente:

- \$GPRMC, son las letras con las cuales se identifica al *Recommended Minimum Specific Data* o RMC, uno de los mensajes de salida del protocolo NMEA, que proporciona los datos mínimos del GPS como fecha, hora y ubicación.
- 044235.000, corresponde a la hora (04:42:35).
- A, indica que la localización del receptor GPS se ha obtenido correctamente.
- 4322.0289, corresponde a la longitud (43°22.0289').
- N, indica norte.
- 00824.5210, corresponde a la latitud (8°24.5210').
- W, indica oeste.
- 0.39, velocidad en nudos (1 nudo = 1853 metros/hora).
- 65.46, es la orientación en grados.
- 020911, es la fecha (02/09/11).
- \*44, se refiere a la suma verificadora (*check sum*), la cual está formada por la operación XOR de todos los datos de la cadena incluyendo las separaciones “,” pero exceptuando los limitadores “\$” y “\*”.

## 2.7. Microcontroladores

La electrónica desarrollada en la actualidad tiene un sin número de aplicaciones, su versatilidad permite su uso en áreas tan ajenas entre sí como el control de sistemas de distribución de energía eléctrica, sistemas electrónicos para el control de aeronaves o su utilización en aparatos médicos. Los sistemas electrónicos se conforman por un conjunto de circuitos que interactúan entre sí para lograr un fin común, sin embargo, el elemento principal de este conjunto de circuitos electrónicos es el microcontrolador.

La palabra microcontrolador (generalmente abreviado  $\mu C$ , UC o MCU), está conformada por dos términos: “micro” por pequeño y “controlador” debido a que se utiliza para maniobrar o controlar procesos definidos mediante la programación. Un microcontrolador es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Un microcontrolador está compuesto por las tres partes fundamentales de una computadora: unidad central de procesamiento (CPU, *Central Processing Unit*), memoria y periféricos de entrada/salida.

Los microcontroladores pueden utilizarse para muchos fines y cada uno puede contar con distinto tipo de arquitectura, no obstante, una de las clasificaciones más importantes tiene que ver con el ancho de palabra utilizado por éste. El ancho, tamaño o longitud de palabra es el tamaño en bits de los registros de la CPU y del bus de datos que se utilizan para realizar operaciones aritméticas. De esta manera, existen MCU de 4, 8, 16 y 32 bits. Entre mayor sea este número, el microcontrolador contará con mayor facilidad para realizar operaciones y su potencial incrementará. Para un microcontrolador con un reducido tamaño de palabra le tomará más tiempo realizar una operación que a uno de 32 bits, puesto que el primero deberá descomponer la operación en operaciones más simples para obtener el valor final.

Propiamente dicho, el procesador construido dentro del microcontrolador es el responsable de ejecutar los programas grabados. Surge entonces otro dispositivo electrónico que vale la pena mencionar, este dispositivo es el microprocesador. La tarea de un microprocesador es ejecutar las instrucciones dictaminadas por un determinado programa, llevar el control del flujo de ejecución del programa y realizar los cálculos necesarios cuando sea necesario. Un microprocesador no puede operar solo para llevar a cabo una determinada tarea, necesita de una memoria donde se almacene el programa informático que deberá ejecutar (EPROM, EEPROM, FLASH), una memoria para el almacenamiento de las variables que se utilizarán a lo largo de la ejecución del programa (DRAM, SRAM), controladores para el manejo de puertos de entrada y de salida, además, para interactuar con otros circuitos integrados, generalmente también se requiere de varias interfaces de comunicación serial para enviar y recibir información.

La diferencia entre un microprocesador y un microcontrolador es que este último ya contiene todos los componentes que le permiten operar de manera autónoma para controlar procesos, mientras que un microprocesador requiere de la conexión de periféricos para poder operar. El microcontrolador, además de la CPU, incluye elementos de memoria, temporizadores, controladores de puertos de entrada/salida, módulos para interfaces de comunicación y algunos incluyen también convertidores analógico-digitales. La lista siguiente muestra los elementos que la mayoría de los microcontroladores incluyen.

- *CPU*. El procesador o CPU ejecuta las instrucciones programadas en lenguaje de bajo nivel, realizando operaciones aritméticas y lógicas. Está compuesto por una Unidad Aritmético-Lógica (ALU), una unidad de control y un número variable de registros como: contador de programa (*program*

*counter*), apuntador de pila (*stack pointer*), registro de estado (*status register*), registro de instrucción, registros de uso general, etc.

- *Memoria*. Se utiliza para almacenar el código de programación. Dependiendo del tipo de arquitectura, las variables manejadas por el procesador (datos) pueden almacenarse en la misma memoria o puede utilizarse una segunda memoria para este fin.
- *Entradas/salidas digitales (I/O, Input/Output)*. Los puertos de entrada/salida se utilizan para realizar una interfaz entre el microcontrolador y los dispositivos periféricos conectados a él. El número de puertos varía dependiendo del microcontrolador utilizado.
- *Temporizadores/contadores (Timer/Counter)*. La mayoría de los microcontroladores incluyen más de un temporizador/contador que pueden ser utilizados para marcar intervalos de tiempo o contar eventos. La mayoría de microcontroladores poseen también varias unidades de PWM (*Pulse-Width Modulation*), las cuales pueden controlar motores haciendo variar su velocidad, mediante el envío de una señal eléctrica con ciclo de trabajo variable.
- *Convertidores analógico-digitales*. Además de las interfaces digitales, los microcontroladores suelen incluir convertidores analógico-digitales de una resolución variable, dependiendo del microcontrolador, generalmente de 8 a 12 bits. También se incluyen características de comparadores analógicos y en algunos casos, también convertidores digital-analógicos.
- *Interfaces de comunicación serial*. Los microcontroladores incluyen por lo menos una interfaz serial para cargar el programa informático y para la comunicación con una PC (*Personal Computer*). Puesto que existe un amplio número de interfaces, la mayoría suele incluir las más utilizadas por la industria electrónica como: SPI, I2C, comunicación serie mediante UART, etc.
- *Perro guardián (Watchdog Timer)*. Puesto que muchos microcontroladores son utilizados en sistemas de seguridad donde la mínima falla es crítica, se utiliza el watchdog. Este elemento reinicia al microcontrolador ante un colapso en el flujo de ejecución del programa.
- *Controlador de interrupciones*. Las interrupciones son útiles para cambiar el flujo normal de ejecución del programa al presentarse eventos internos o externos, como el término de una conversión analógico-digital o la pulsación de un botón.

Un ejemplo de la estructura de un microcontrolador se muestra en la figura 2.90. El diagrama de bloques corresponde al microcontrolador MSP430 de 16 bits de Texas

Instruments. En el orden de arriba hacia abajo y dirección izquierda a derecha se pueden observar en la figura las unidades de: reloj, CPU, interfaz JTAG (*Joint Test Action Group*) utilizado comúnmente para la depuración de aplicaciones en microcontroladores, memoria Flash/ROM, bus de dirección de memoria (MAB, *Memory Address Bus*), bus de datos de memoria (*Memory Data Bus*), temporizador *Watchdog*, memoria RAM, una interfaz entre el bus MDB y los controladores periféricos (Bus. Conv.) y, finalmente, los controladores de periféricos.

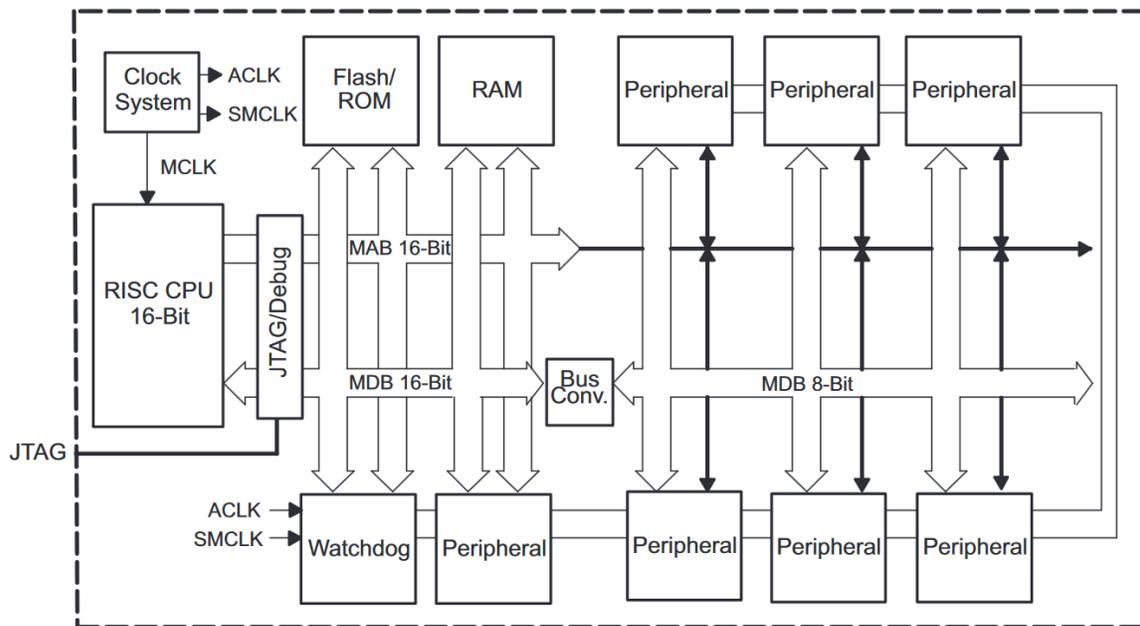


Figura 2.90. Diagrama de bloques del microcontrolador MSP430.

Una última característica que no se puede dejar de lado son los fusibles (*fuses*) o bits de configuración, éstos permiten configurar ciertas funciones en algunos microcontroladores como el tipo de oscilador seleccionado, la protección de código, la habilitación del *Watchdog*, habilitación del reinicio maestro, habilitación de reinicio por bajo voltaje en la fuente de polarización, habilitación de un depurador como JTAG, entre otras. Los bits de configuración se almacenan en un tipo de memoria no volátil y de escasos 3 bytes que pueden ser programados mediante un software especial.

El mercado de los microcontroladores es muy extenso y para elegir uno en particular, que satisfaga las necesidades de alguna aplicación, se deben considerar una multitud de factores. Los siguientes factores tienen que ver con el desarrollo, disponibilidad y costos de utilizar un microcontrolador específico:

- Se debe considerar si existe documentación y herramientas de desarrollo disponibles para utilizar un microcontrolador en particular, además se debe saber si éstas son de fácil acceso o si se deben pagar numerosas cantidades de dinero para adquirir el software y documentación con los cuales se puede comenzar a trabajar con el microcontrolador.

- La disponibilidad de obtención del microcontrolador por parte de los fabricantes y la popularidad entre los usuarios relacionados con el área de la electrónica. Cuanto más amplia sea su oferta en el mercado y la comunidad de usuarios sea mayor, será mucho más fácil, por un lado, buscar un remplazo en un periodo muy corto de tiempo (sin que el proyecto se detenga por falta de proveedores de fácil acceso) y por el otro lado, la búsqueda de soluciones ante problemas frecuentes con el uso del microcontrolador.
- El costo y la facilidad de explotación de los productos desarrollados con base en un microcontrolador son factores importantes a tomar en cuenta. Para poder comercializar un equipo se deben tomar precauciones como el hecho de poder ejercer un dominio total sobre la comercialización de éste, puesto que se basa en un microcontrolador y/o software provistos por un tercero. Si estos factores no se toman en cuenta podría caerse en un problema legal si el fabricante tiene restricciones sobre el uso de sus productos.

En cuanto a los factores técnicos, se deben considerar como mínimo los siguientes:

- *Ancho de palabra.* Si la aplicación requiere el procesamiento de grandes cantidades de datos además de rapidez para la ejecución de los cálculos, se debe optar por un microcontrolador de una longitud de palabra grande, sin embargo, los contras se ven reflejados en la elevación de los precios al momento de ser adquiridos. Si por el contrario, la aplicación no requiere manejar operaciones complejas y los procesos a controlar son más simples, es conveniente optar por un microcontrolador más modesto, en cuanto a costo y longitud de palabra, pero que al mismo tiempo satisfaga los requerimientos mínimos.
- *Puertos de entrada/salida.* Cuando se ha elegido el ancho de palabra adecuado y el MCU satisfaga los requisitos de procesamiento de datos, también se debe considerar el número de periféricos que serán conectados al microcontrolador. El microcontrolador cuenta con un número limitado de terminales de entrada/salida, por lo que si este factor no se ha considerado, podría incurrirse en el problema de que las terminales no son suficientes para controlar todos los periféricos como: LEDs, *displays*, interruptores, motores, etc. Aunque es posible solucionar este problema mediante el empleo de multiplexores, la consecuencia es el incremento en el costo y densidad de componentes del sistema y la reducción de velocidad para la ejecución de una tarea.
- *Memoria.* En este aspecto se debe considerar tanto la memoria de programa como la memoria de datos. Si la memoria de programa (típicamente se utiliza una memoria flash) es limitada, el código de programación no podrá almacenarse y por lo tanto la electrónica no tendrá utilidad alguna. Por otra parte, también debe haber una cierta holgura en la capacidad de la memoria

volátil o de datos (RAM), de manera que los cálculos puedan realizarse sin problemas y el microcontrolador no colapse ante un exceso en el manejo de información como suele ocurrir con el uso desmesurado de arreglos en la programación del código. Otra memoria que suele incluirse en un microcontrolador es una EEPROM, cuya utilidad recae en el almacenamiento de valores que serán constantes a lo largo de la ejecución del programa, este tipo de memoria puede ser útil si el programa requerirá números de serie o cadenas extensas de datos que no cambiarán con el tiempo.

- *Interfaces de comunicación.* Debe asegurarse que el microcontrolador utilizado cuente con los módulos necesarios para interactuar con otros circuitos electrónicos mediante la comunicación serial, aunque en la actualidad casi todos incluyen estas interfaces, suelen haber excepciones, sobre todo al tratarse de microcontroladores de propósito general y con ancho de palabra limitado.
- *Consumo de energía.* Algunas aplicaciones requieren que el microcontrolador entre en acción únicamente ante la ocurrencia de un evento, como la pulsación de un interruptor, de esta manera, a lo largo del tiempo que no se le requiera, el sistema óptimamente debe de disminuir el consumo de energía eléctrica. Con esto en mente, se debe optar por aquel microcontrolador que requiera menos consumo de energía para su operación, incluso existen algunos que pueden operar en un modo de bajo consumo, mejorando entonces las cualidades de todo el proyecto.

La tabla 2.5 muestra algunos de los microcontroladores existentes en el mercado.

Empresa	8 bits	16 bits	32 bits
Atmel	Familia AVR 89SXXX 8051	---	SAM7(ARM7TDMI) SAM3(ARM Cortex- M3) SAM9(ARM926) AVR32
Freescale	68HC05 68HC08 68HC11 HCS08	68HC12 68HCS12 68HCX12 68HC16	683XX Power PC ColdFire
Intel	MCS-48(familia 8048) MCS51(familia 8051) 8XC251	MCS96 MXS296	---
INFINEON	C500 XC800	C100 XC100 XE100	T1130 TC116X TC1167X
Microchip	10F2XX 12CXX 12FXX 16CXX 16FXX	PIC24F PIC24H DSPIC30FXX DSPIC33F	PIC32
NXP Semiconductors	80C51 80C552 80C31	PXAC37 XA-G30 XA-S3	Cortex M0 Cortex M3 ARM7 ARM9 LPC1300
Renesas	78K H8	H8S 78K0R R8C	RX V850 SuperH SH-Mobile
ST Microelectronics	STM8L STM8S STM8AF STM8T	---	STM32(ARM7)
Texas Instruments	TMS370	MSP430	C2000 Cortex M3(ARM) TMS570(ARM)
Zilog	Z51F0410 Z51F0811 Z51F3220 Z51F6412	Z16F2810 Z16F2811 Z16F3211 Z16F6411	---
ARM	---	---	Cortex A series Cortex R series Cortex M series

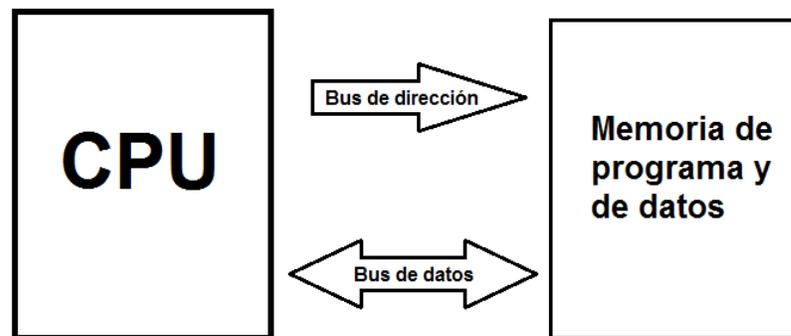
Tabla 2.5. Microcontroladores más comunes en el mercado.

### 2.7.1. Arquitectura y funcionamiento de un microcontrolador

#### Arquitectura

Cada fabricante de microcontroladores dispone de un diseño característico en la construcción de un microcontrolador, no obstante, todos tienen una estructura muy parecida. Los microcontroladores basan su estructura general en dos tipos de arquitecturas:

- *Arquitectura von Neumann*: En este tipo de arquitectura, propuesto por John von Neumann en 1945, el programa informático y los datos son almacenados juntos en una sola memoria y su acceso se realiza mediante una sola ruta (compuesta de un bus para direccionar la memoria y un bus para transferir datos a la memoria). La desventaja que esto impone es un conflicto conocido como *efecto de cuello de botella*, que es originado al intentar leer una instrucción del programa y un dato, ambos compartiendo el mismo camino para ser procesados. El resultado es una disminución de la velocidad en la operación del microcontrolador. La figura 2.91 muestra un esquema de este tipo de arquitectura.



#### ARQUITECTURA VON NEUMANN

Figura 2.91. Arquitectura von Neumann.

- *Arquitectura Harvard*: El término proviene de la computadora Harvard Mark I. Este tipo de arquitectura demanda que el programa y los datos sean almacenados en memorias diferentes y con buses independientes. Como consecuencia, los accesos al código del programa no entran en conflicto con los accesos a los datos, mejorando el rendimiento general del microcontrolador. Una diferencia de esta arquitectura es que requiere mayor hardware al necesitar buses de datos separados y dos elementos de memoria. La figura 2.92 muestra un esquema de este tipo de arquitectura.

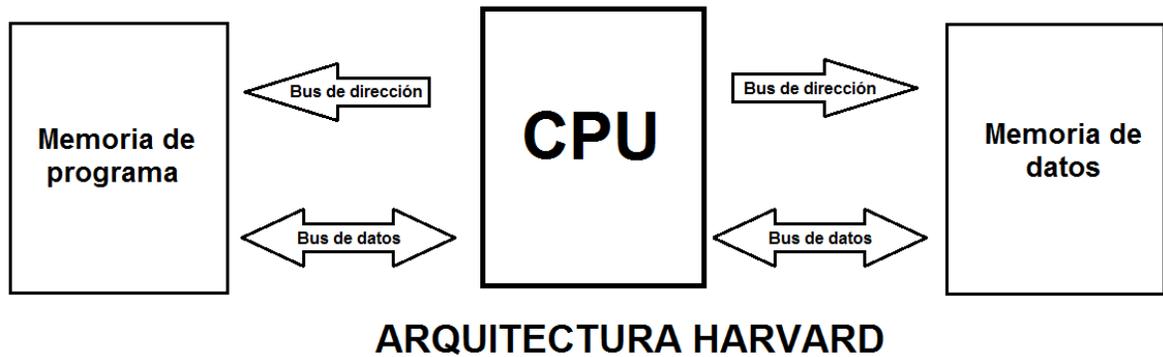


Figura 2.92. Arquitectura Harvard.

El núcleo del microcontrolador es el procesador o Unidad Central de Procesamiento, cuyos componentes fundamentales son la Unidad Aritmético Lógica, la Unidad de Control y una serie de registros que proporcionan los operandos a la ALU y almacenan los resultados arrojados por esta misma unidad. La figura 2.93 muestra un esquema de la estructura general de una Unidad Central de Procesamiento.

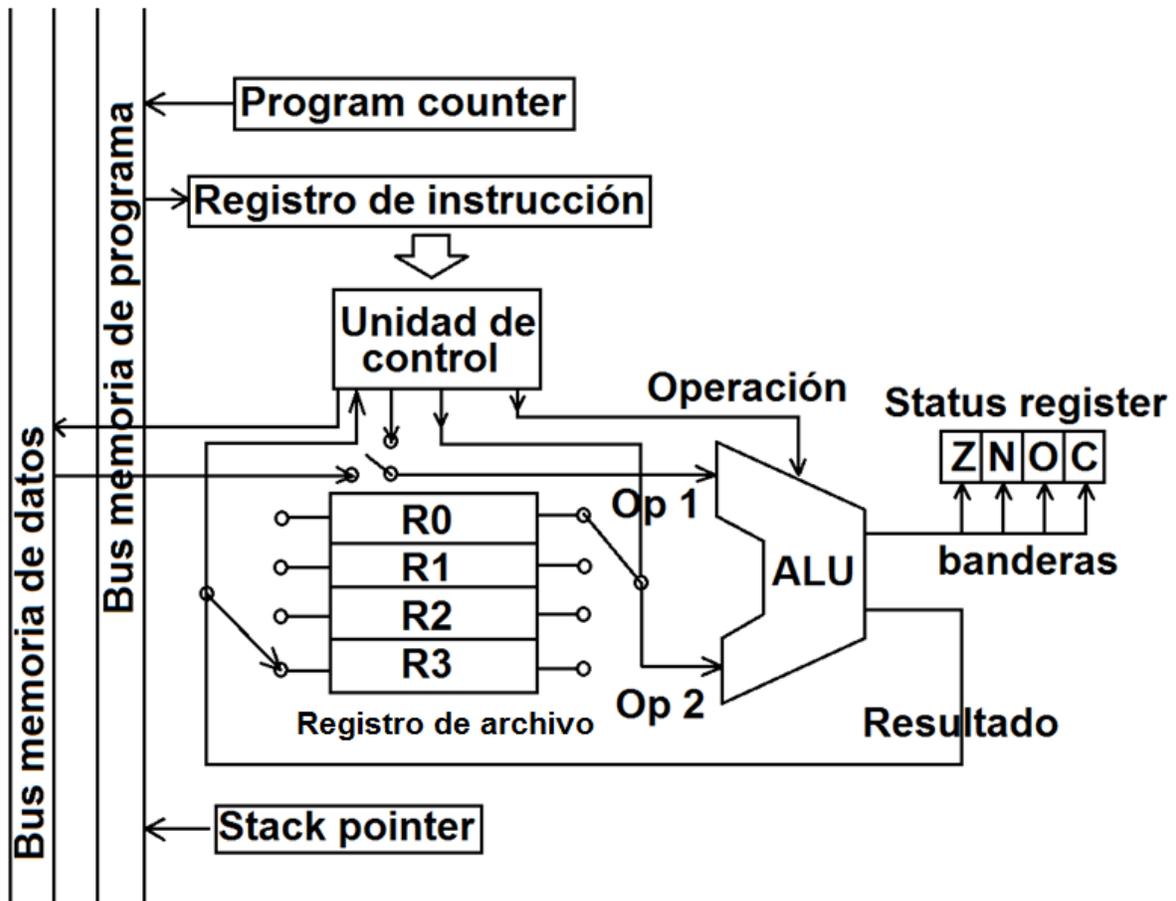


Figura 2.93. Esquema general de una CPU.

Antes de describir el funcionamiento completo de la CPU, vale la pena describir por separado algunos de los elementos que la conforman. Como se ha mencionado, los tres elementos básicos que conforman la CPU son la ALU, la Unidad de Control y una serie de registros.

- *ALU.* El núcleo de la CPU es la Unidad Aritmético Lógica, que es utilizada para realizar operaciones aritméticas simples como sumas, restas y multiplicaciones, además de operaciones lógicas como AND, NOT, OR, etc. Varias líneas de control le proporcionan a la ALU los operandos y le indican que operación se realizará, entonces, la ALU toma dos operandos fuente como entrada y arroja como salida un resultado destino. Los operandos fuente pueden ser tomados tanto de la memoria de datos como de los registros auxiliares, de la misma manera, el destino puede realimentarse a los registros o ser llevado a la memoria de datos. Además del resultado, la ALU proporciona otra información de interés que se almacena en el *Status Register*.
- *Unidad de Control.* Su función es buscar las instrucciones en la memoria de programa, decodificarlas (interpretarlas) y ejecutarlas. Su tarea radica en determinar qué operación debe ser ejecutada y configurar las rutas necesarias para tal fin. Para realizar su función primero, con ayuda del *Program Counter*, obtiene la dirección de la siguiente instrucción, a continuación, la Unidad de Control carga dicha instrucción al *Registro de Instrucción*, la decodifica y establece las rutas para proporcionar los operandos y operación a la ALU. La configuración de las rutas incluye proporcionar los operandos de registros auxiliares o de la memoria de datos, del mismo modo, después de obtener el resultado proporciona las rutas para almacenarlo en los registros o en la memoria de datos, según se requiera.
- *Program Counter.* El *contador de programa* también llamado *Instruction Pointer (Apuntador de Programa)*, es un registro del procesador que dependiendo del diseño del procesador, apunta a la dirección de la instrucción que está siendo ejecutada o bien, a la instrucción siguiente que va a ser ejecutada. El *contador de programa* es incrementado automáticamente en cada ciclo de instrucción (periodo que tarda la CPU en ejecutar una instrucción de lenguaje máquina) en un número variable de bytes dependiendo del tamaño de la instrucción ejecutada, de tal manera que las instrucciones son leídas de la memoria de programa. Ciertas instrucciones tales como las llamadas y retornos de subrutinas interrumpen la secuencia al colocar un nuevo valor en el *contador de programa*, es cuando se hace uso del *Stack Pointer*.
- *Stack pointer.* La pila o *Stack* es una porción de memoria consecutiva en el espacio de memoria de datos que es usada por la CPU para almacenar direcciones y posiblemente el contenido de algunos registros durante la ejecución de una subrutina o una llamada de una rutina de servicio de

interrupción, *ISR (Interrupt Service Routine)*. La pila es accedida al utilizar los comandos *PUSH* (poner en el *Stack*) y *POP* (remover del *Stack*). Para señalar la posición en la pila se cuenta con el *Stack Pointer*, este decrementa periódicamente partiendo de la más alta dirección de memoria hacia el valor más bajo de dirección de memoria, en otras palabras, el *Stack Pointer* inicia al final de la memoria de datos y finaliza al principio de la misma. La razón de este modo de operación es que generalmente los datos son almacenados al principio de la memoria de datos, entonces, poniendo el *Stack Pointer* al final se evita la colisión. El microcontrolador usa el *Stack Pointer* durante el llamado de una subrutina o interrupción, de esta manera, siempre que el programa sea interrumpido, el *Stack* cuenta con los valores necesarios para continuar con la ejecución del programa después de haber realizado la subrutina o interrupción.

- *Status Register*. Este registro almacena el valor de algunas variables o banderas, que al conocer su estado, se puede tomar alguna acción para cambiar la secuencia de ejecución del programa. Las banderas suelen ser: *Z (Zero)*, que se establece (toma un valor alto) cuando el resultado de una operación es cero; *N (Negative)*, se establece cuando el resultado de una operación es negativo, esto es, el MSB del resultado es 1; *O (Overflow)*, se establece cuando el resultado de alguna operación aritmética sobrepasa el rango manejado y *C (Carry)*, que se establece cuando el resultado de una operación genera un bit de acarreo. El programa puede valerse de estas banderas para tomar una acción en concreto, por ejemplo, el *mnemónico* (en lenguaje ensamblador, palabra que sustituye un código de operación) *JC* evalúa el valor de la bandera *C* y en caso de que éste contenga un valor 1, se realiza un salto hacia una etiqueta especificada en el programa.
- *Registro de instrucción*. El registro de instrucción almacena la instrucción que se está ejecutando. En los microcontroladores simples cada instrucción a ser ejecutada es cargada al registro durante el tiempo que es decodificada, preparada y ejecutada. Los microcontroladores más sofisticados utilizan una técnica denominada *segmentación* (en inglés *pipelining*) dividida en partes, en donde en cada sección se realiza un trabajo parcial para decodificar, preparar y ejecutar las instrucciones, pasando a las etapas consecuentes con menor carga de trabajo y ejecutando la instrucción final en un menor tiempo.
- *Registro de archivo*. El registro de archivo contiene los registros de trabajo de la CPU. Puede consistir de un conjunto de registros de propósito general (generalmente de 16 a 32, pero puede variar), cada uno de los cuales puede contener la fuente o destino de alguna operación. También puede estar conformado por registros dedicados como un acumulador utilizado para operaciones aritmético-lógicas, o un registro indexado, utilizado para cierto tipo de modos de direccionamiento. En cualquier caso, la CPU puede tomar los operandos de estos registros y posteriormente almacenar el resultado en el mismo lugar. No es obligatorio que las fuentes y destinos sean

almacenados en estos registros, de manera alternativa la información puede provenir de la memoria de datos y el resultado ser almacenado en esta misma memoria, sin embargo, los accesos a la memoria de datos generan mayores retardos de tiempo.

### *Funcionamiento*

El principio de operación del microcontrolador se basa en la ejecución de una secuencia de instrucciones almacenadas en la memoria de programa, esta serie de instrucciones son asignadas en el *programa fuente*. El programa se basa en una secuencia de *códigos de operación (op code)* que simbolizan instrucciones de *lenguaje máquina*, o en otras palabras, códigos fácilmente entendibles por el microcontrolador que trabaja con números binarios únicamente.

Para establecer las instrucciones en el programa fuente se utiliza alguna herramienta de software y se describe el programa mediante instrucciones establecidas por parte del programador. Existen dos formas o lenguajes para realizar esta tarea: *lenguaje de bajo nivel* y *lenguaje de alto nivel*. El lenguaje de alto nivel le permite al programador indicar las instrucciones que el microcontrolador ejecutará, valiéndose de sentencias que son más familiares al lenguaje que utilizamos los seres humanos. Las instrucciones son similares a las que una persona le indica a un tercero, por ejemplo, para preparar una limonada; utilizando este ejemplo los pasos podrían ser: primero lava los limones, después corta los limones, mientras los limones tengan jugo exprime los limones, etc. Al conjunto de pasos realizados para lograr una tarea se le denomina *algoritmo*. Los programas escritos en lenguaje de alto nivel permiten a otros programadores descubrir intuitivamente los pasos que se están siguiendo en la ejecución del programa. Algunos ejemplos de lenguajes de alto nivel son: C, C++, Java, Phyton, etc.

El apelativo lenguaje de bajo nivel no quiere decir que el potencial sea menor sino que las instrucciones son más parecidas al lenguaje nativo de las computadoras que operan con dígitos binarios. Con el lenguaje de bajo nivel el programador tiene un control más completo del microcontrolador haciendo un uso más eficiente de sus recursos y del tiempo necesario para realizar las operaciones. El inconveniente de este lenguaje es que la programación no se puede realizar tan intuitivamente y se debe tener un mayor conocimiento de la arquitectura del microcontrolador.

El lenguaje de bajo nivel puede ser: *lenguaje de código máquina* y *lenguaje ensamblador*. El primero es el único lenguaje que el microcontrolador entiende de forma nativa, consiste únicamente de números que representan instrucciones y por lo tanto, es raro que sea utilizado por un programador. El *lenguaje ensamblador* emplea *nemónicos* que son palabras que se corresponden con un código numérico y que el programador puede asociar con dicha operación, por ejemplo, el *nemónico MOV* se corresponde con una instrucción utilizada para mover un valor a un determinado registro.

El programa fuente, ya sea escrito en lenguaje de alto nivel o de bajo nivel, debe ser “transformado” a otro tipo de programa que sí pueda ser entendido por el microcontrolador. Existen herramientas utilizadas para tal fin, estas son los *intérpretes* y *traductores*. Los *intérpretes* únicamente analizan el programa fuente y permiten ejecutarlo instrucción por instrucción, sin embargo, no se genera otro tipo de programa para ser guardado en la memoria del microcontrolador. Por otro lado, los *traductores* cumplen con dos funciones, la primera consiste en analizar el programa y verificar que no existen errores, por ejemplo de sintaxis y, el segundo paso consiste en la síntesis, que es la generación de un *programa objeto* el cuál sí puede ser manejado por el microcontrolador.

Cuando el traductor genera un programa objeto a partir de un programa fuente escrito en lenguaje de alto nivel, se denomina *compilador* y, si el traductor genera un programa objeto a partir de un programa fuente escrito en lenguaje ensamblador se denomina *ensamblador*. Existe otra herramienta utilizada para corregir o depurar errores (en inglés *bug* que significa bicho) del programa, esta herramienta se denomina *depurador* (en inglés *debugger*) y facilita al programador ejecutar el programa instrucción por instrucción, pausar el programa para analizar el estado de los registros, modificar el contenido de éstos, entre otras cosas.

Una vez que el programa fuente es compilado sin errores y se ha generado el *programa objeto*, éste último es guardado en la memoria de programa del microcontrolador a través de un *grabador*, que es una pieza de hardware que a través de señales eléctricas provenientes de una PC almacena el programa. Muchos fabricantes de microcontroladores incluyen el hardware para la programación del microcontrolador y el propio microcontrolador en un solo circuito que fácilmente se conecta a la PC.

Cuando el microcontrolador ya se encuentra ejecutando el programa almacenado en su memoria, las instrucciones almacenadas son ejecutadas siguiendo los siguientes pasos: leer, decodificar, ejecutar y escribir (*fetch, decode, execute y writeback*).

#### *Fetch (lectura)*

El primer paso consiste en leer una instrucción de la memoria de programa. La localización en la memoria de programa es determinada por el *Contador de Programa (PC)* que almacena un número que identifica la dirección donde se sitúa la siguiente instrucción a ejecutar. El *PC* es incrementado en una cantidad de bytes que varía dependiendo del tipo de instrucción leída, de esta manera contará con la dirección de la siguiente instrucción que será ejecutada. Por lo regular la instrucción leída de la memoria de programa es retenida durante el tiempo en que es procesada, para el caso de los microcontroladores más sofisticados se hace uso del *pipelining*.

### *Decode (decodificación)*

En el paso de decodificación la instrucción es interpretada a un valor que tiene significado para la CPU. La manera en que el valor de la instrucción numérica es interpretado está definida por el *conjunto de instrucciones* del microcontrolador, un grupo de instrucciones que son características de un microcontrolador. El tipo de valor numérico indica qué operación debe realizarse. El valor numérico también proporciona información requerida para esa instrucción, por ejemplo, los operandos para una operación de adición. Los operandos se pueden dar como un valor constante (llamado valor inmediato), o como un lugar para localizar un valor, que dependiendo del *modo de direccionamiento* puede ser un registro o una dirección de la memoria de datos.

### *Execute (ejecución)*

Después de los pasos de lectura y decodificación se realiza la ejecución de la instrucción. Durante este paso, varias unidades del CPU son conectadas de tal manera que pueden realizar la operación deseada. Si, por ejemplo, una operación de adición fue solicitada, la ALU será conectada a un conjunto de entradas y un conjunto de salidas. Las entradas proporcionan los números a ser sumados y las salidas contendrán el resultado de la operación. Si la operación produce un resultado demasiado grande para poder ser manejado por la CPU, la bandera *Overflow (O)* se establece.

### *Writeback (escritura)*

El paso final consiste en la escritura de los resultados en algún tipo de almacenamiento temporal de datos. Muy a menudo los resultados son escritos en algún registro interno de la CPU para acceso rápido en subsecuentes operaciones, sin embargo, puede que en cambio el dato sea almacenado en la memoria de datos. También, algunos tipos de instrucciones manipulan el *Contador de programa* en lugar de directamente producir un resultado, este tipo de instrucciones son llamados *saltos*, como la instrucción *JC* mencionada algunas páginas atrás. Los *saltos* facilitan comportamientos como ciclos de repetición, la ejecución condicional de una sección del código y *funciones* en programas.

Después de la ejecución de la instrucción y la escritura de los datos resultantes, el proceso entero se repite con el siguiente ciclo de instrucción, normalmente leyendo la siguiente instrucción en secuencia debido al valor incrementado en el *Contador de Programa*. Si la instrucción era un *salto*, el *Contador de Programa* será modificado para contener la dirección de la instrucción a la cual se saltó y la ejecución del programa continúa normalmente.

### 2.7.2. Conjunto de instrucciones

Debido a que existen múltiples fabricantes de microcontroladores, además de distintos tipos de arquitecturas, incluso tratándose de microcontroladores de un mismo fabricante, existen especificaciones de lenguaje propias de un microcontrolador en particular. Un microcontrolador cuenta con un número determinado de instrucciones que puede realizar.

La arquitectura del conjunto de instrucciones o más comúnmente: conjunto de instrucciones (ISA, *Instruction Set Architecture*), especifica las instrucciones que una CPU es capaz de entender y ejecutar. Las especificaciones incluyen el tipo de datos utilizados por el microcontrolador, las instrucciones, los registros, la arquitectura de memoria y las interrupciones. Los dos modelos de ISA son: *RISC* y *CISC*.

- *RISC (Reduced Instruction Set Computer)*. Traducido como Computadora de Conjunto de Instrucciones Reducido, es un tipo de diseño de CPU que se caracteriza por poseer un repertorio de instrucciones compacto y de tamaño fijo, además sólo se accede a la memoria de datos al realizar una operación de carga o almacenamiento.
- *CISC (Complex Instruction Set Computer)*. Traducido como Computadora de Conjunto de Instrucciones Compleja, es un modelo de arquitectura de CPU que se caracteriza por poseer un amplio conjunto de instrucciones y que permite operaciones complejas entre operandos, situados en la memoria de datos o en los registros internos del microcontrolador.

La diferencia entre estos dos modelos de arquitectura de CPU es que *RISC* está a favor de conjuntos de instrucciones más pequeños y simples, facilitando el *pipelining* y el *paralelismo* utilizado para ejecutar más de una instrucción por ciclo de reloj, además de reducir los accesos a memoria, todo esto conlleva a una mayor velocidad en la operación del microcontrolador. Por otro lado, *CISC* simplifica el trabajo del compilador, puesto que su conjunto de instrucciones complejo realiza un mayor número de operaciones con una sola instrucción.

Entre las instrucciones más ampliamente manejadas por los microcontroladores se encuentran las siguientes:

- Transferencia de datos. Los datos se copian de un origen a un destino, normalmente sin afectar las banderas *O*, *Z*, *N* y *C*. Algunos de los nemónicos característicos son: *move*, que copia el contenido de la memoria de datos o de un registro a otro registro; *clear* que pone todos los bits del destino a cero; *set*, que pone todos los bits del destino a uno; *push*, que introduce un dato en la cabecera del *stack*; *pop* que saca un dato de la cabecera del *stack*, etc.
- Instrucciones aritméticas. Son realizadas por la ALU y suelen modificar las banderas del *Status Register*. Algunos de los nemónicos característicos son:

*add*, para realizar una suma; *addc*, para realizar una suma con acarreo; *sub*, para realizar una resta; *inc*, para incrementar en una unidad el registro; *dec*, para decrementar en una unidad un registro, etc.

- Instrucciones de comparación. Son utilizadas para evaluar un registro y compararlo con un valor determinado. Se utilizan antes de una bifurcación concebida para destinar flujos de ejecución del programa diferentes, de acuerdo a una condición. Algunos nemónicos característicos suelen ser: *comp*, que resta dos operandos pero sin almacenar el resultado, únicamente se modifican las banderas del *Status Register*, otro nemónico es *test*, que compara un valor dictado por el programador con el valor cero.
- Instrucciones lógicas. Realizan operaciones booleanas bit a bit entre dos operandos, también modifican las banderas de estado. Las operaciones suelen ser: *and*, *or*, *xor* y *not*.
- Instrucciones de desplazamiento. Son utilizados para realizar un desplazamiento de los bits almacenados en algún registro. Dos nemónicos usados frecuentemente son: *shift*, que desplaza los bits ya sea a la derecha o izquierda y, *rotate*, que reordena los bits de algún registro realizando una operación de rotación.
- Instrucciones de bits. Manejan bits independientes dentro de un registro. Algunos nemónicos son: *bit test*, que comprueba si un bit específico del registro contiene un valor lógico alto; *bit clear*, que pone un bit a cero y *bit set* que pone un bit a uno.
- Instrucciones de control. Se utilizan para modificar la secuencia normal de ejecución de un programa, pueden ser saltos condicionales e incondicionales, llamadas a subrutinas y llamadas de interrupciones. Algunos de los nemónicos manejados son: *jump* o *branch*, que son saltos incondicionales y redirigen el programa a una línea de código marcada por una etiqueta; *jc*, *jn*, *jo*, *jz*, que son saltos condicionales y redirigen el programa a una línea de código marcada por una etiqueta siempre y cuando las banderas *C*, *N*, *O* y *Z* sean establecidas. Los nemónicos *call* y *ret* se utilizan para ejecutar una subrutina, por otro lado, el nemónico *reti* es utilizado para señalar el final de una rutina de servicio de interrupción y retornar a la línea de código donde se generó la interrupción.
- Otras. Existen otros nemónicos que suelen ser utilizados para otro tipo de operaciones, entre ellos se encuentran: *nop*, que no realiza operación alguna pero genera un retardo de tiempo; *enable*, que permite habilitar las interrupciones y *disable*, que deshabilita las interrupciones, entre otras.

Finalmente, vale la pena mencionar que no existe una manera estandarizada para nombrar los nemónicos y su operación en código máquina asociado, varía de un fabricante a otro.

---

# Capítulo 3. Diseño del sistema

En este capítulo se describirá el procedimiento realizado a lo largo del proceso de diseño del sistema para el monitoreo de fenómenos sísmicos. Se describirán los elementos de los cuáles está constituido el sistema, la justificación de su elección y las especificaciones técnicas necesarias para el funcionamiento de cada elemento constitutivo. El capítulo se divide en dos secciones; la primera tiene que ver con el hardware del sistema, que consiste de los circuitos integrados y módulos electrónicos que implementan el funcionamiento físico del equipo, a través de la manipulación de señales eléctricas gestionadas por parte del microcontrolador. La segunda parte está asociada con el software del sistema y se describe el programa informático cargado al microcontrolador, este programa se encuentra segmentado en pequeñas rutinas, se describe cada una de ellas así como el algoritmo utilizado para solucionar cada tarea. El punto de partida del sistema para el monitoreo de fenómenos sísmicos lo marca el desarrollo de un sistema similar desarrollado para una aplicación diferente y es el que se describe a continuación.

## 3.1. Antecedentes del sistema electrónico para el monitoreo de fenómenos sísmicos

El sistema DIGI (DIGITIZER) es un sistema desarrollado para una aplicación de sismología cuyo principal objetivo es el almacenamiento continuo de datos; el sistema utiliza tres sensores pasivos GS-11D de la compañía Geospace. DIGI está diseñado para digitalizar la señal de tres geófonos que no requieren de una fuente de polarización para emitir una señal eléctrica. Los datos digitalizados son almacenados en una memoria microSD, incluyendo también la hora de inicio de la prueba, ésta se obtiene de un módulo GPS.

Las especificaciones técnicas del sistema DIGI son las siguientes:

1. Alimentación: 12 [V].
2. Rango de voltaje aceptado:  $\pm 625$  [mV].
3. Ganancia de amplificación de las señales de entrada: 12 [dB].
4. Tasa de muestreo configurable. Tres tasas de muestreo programables: 50, 100 y 200 muestras por segundo (mps).

5. Tres canales para la entrada de señal.
6. Resolución de los canales: 21 bits.
7. Filtro pasivo anti-aliasing con frecuencia de corte igual a 20 [Hz].
8. Obtención de la referencia de tiempo a través de un módulo GPS.
9. Almacenamiento de datos en memoria microSD extraíble.
10. Interfaz hombre-máquina mediante *joystick* y *display* de cristal líquido (*Liquid Crystal Display*, LCD).

La figura 3.1 muestra el sistema DIGI.

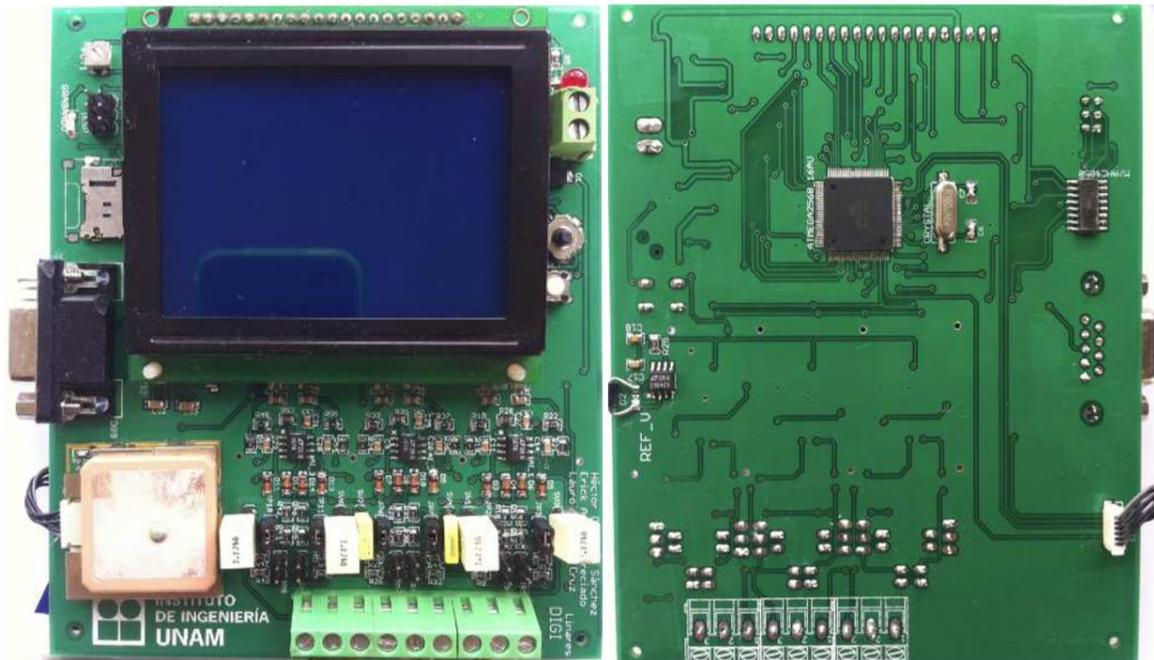


Figura 3.1. Sistema DIGI.

DIGI admite la conexión de sensores cuya salida se encuentre en el rango de  $\pm 625$  [mV], los sensores son conectados a una etapa de acondicionamiento de señal, donde se realiza la suma de un voltaje de directa con el propósito de manejar señales positivas y negativas pero sin la necesidad de utilizar una fuente de polarización doble. En la etapa de acondicionamiento también se realiza un filtrado con ayuda de un filtro pasivo de primer orden, con frecuencia de corte igual a 9 [Hz], no obstante, el usuario es capaz de cambiar la frecuencia de corte del filtro al intercambiar 3 pares de capacitores manualmente. La última etapa del circuito de acondicionamiento de señal del DIGI consiste de un bloque de amplificación, que proporciona una ganancia de 12 [dB].

La digitalización de los datos se realiza con la ayuda de tres convertidores analógico-digitales sigma-delta cuya resolución es igual a 21 bits. El usuario puede configurar la tasa de muestreo y elegir entre 50, 100 y 200 mps. Los datos digitalizados son almacenados en una tarjeta de memoria microSD, el sistema admite tarjetas de memoria de capacidad estándar (hasta 2 GB).

DIGI emplea un módulo GPS para adquirir la fecha y hora de la prueba, además, para que el usuario pueda interactuar con el equipo, el sistema cuenta con un *display* gráfico que le proporciona la información requerida a través de una serie de menús desplegados en él. DIGI incorpora también un *joystick* permitiendo al usuario configurar el sistema con ayuda del *display*. El sistema cuenta con un puerto DB9 para la comunicación serial con otro sistema, sin embargo, el puerto no está habilitado.

La secuencia de pasos a seguir para poder trabajar con el sistema DIGI es la siguiente: el usuario conecta los sensores que serán utilizados para la prueba, a continuación, configura la tasa de muestreo con ayuda del *joystick* y el LCD. Para verificar que la hora proveniente del GPS es correcta, se puede acceder a una opción que despliega la hora en el LCD, de la misma manera, el sistema permite graficar las señales en el LCD para verificar que los sensores han sido conectados correctamente. Para iniciar una prueba, el sistema brinda la facilidad de elegir la duración que ésta tendrá, se permite la elección en número de días o en número de horas. El usuario debe establecer el periodo de ejecución de la prueba y una vez establecida, el sistema comenzará con el almacenamiento de datos en la memoria microSD. Al iniciar el almacenamiento de datos, primeramente el sistema adquiere la fecha proporcionada por el GPS, la almacena en la memoria microSD y posteriormente almacena los datos correspondientes a los tres canales. Todos los datos son almacenados directamente en formato binario para optimizar el uso de la tarjeta de memoria. La prueba termina en el momento en que se llegue a la conclusión del periodo de tiempo fijado por el usuario.

El conjunto conformado por los sensores y el sistema de digitalización de datos, DIGI, constituyen propiamente un sismómetro electrónico moderno cuyo monitoreo de datos se realiza de manera continua, generando un historial a lo largo de todo el tiempo de operación del sistema.

El equipo descrito anteriormente es útil en aplicaciones en las cuales los investigadores requieren un historial de datos provenientes de los geófonos, sin interrupciones durante el periodo de prueba. El prototipo fue probado y cumplió cabalmente con su propósito, sin embargo, surgió la inquietud de diseñar un sistema análogo que operara con acelerómetros y que sirviera para el estudio de movimientos fuertes (*strong motion*); este nuevo equipo requeriría de un sistema de disparo para la inicialización del almacenamiento de datos y su área de aplicación sería el estudio estructural de distintas zonas sobre la corteza terrestre, en otras palabras, el nuevo equipo se asemejaría más a un acelerógrafo.

La Coordinación de Sismología e Instrumentación Sísmica solicitó a la Coordinación de Electrónica del Instituto de Ingeniería el desarrollo de nuevo equipo, que debe contar con las siguientes características:

CARACTERÍSTICAS	RANGO	UNIDADES
<i>Disparo por umbral en los 3 canales</i>	<i>0 a 10, con incrementos de 0.5</i>	<i>[Gal]</i>
<i>Filtro anti-aliasing</i>	<i>Frecuencia de corte: 12</i>	<i>[Hz]</i>
<i>Frecuencias de muestreo</i>	<i>200 fijo</i>	<i>muestras por segundo</i>
<i>Pre evento</i>	<i>50 fijo</i>	<i>[s]</i>
<i>Post evento</i>	<i>60 fijo</i>	<i>[s]</i>
<i>Capacidad de re disparo</i>		
<i>Tiempo de la primera muestra en el archivo de datos</i>		
<i>Ajuste automático del reloj por GPS</i>		
<i>Ajuste de reloj manual por teclado</i>		
<i>Display para despliegue de parámetros</i>		
<i>Despliegue de reloj en display</i>		
<i>Teclado para ajuste de parámetros</i>		
<i>Tarjeta de memoria de datos intercambiable</i>		
<i>Visualización del número de eventos sucedidos</i>		
<i>Visualización del voltaje de la batería</i>		
<i>Visualización de la cantidad de memoria usada y disponible</i>		
<i>Visualización de las señales numéricas (offsets) de los 3 canales</i>		

Tabla 3.1. Características solicitadas para el nuevo equipo.

Se requiere que todos los ajustes de parámetros se puedan realizar sin ayuda de una computadora, utilizando sólo el teclado y el display del adquirente, se pretende que la rutina de revisión sea muy simple y que consista sólo en:

1. Parar la adquisición.
2. Verificar el voltaje de la batería.
3. Verificar o en su caso ajustar el tiempo del reloj con año, mes, día, hora, minuto y segundo.
4. Verificar las señales de los canales.
5. Retirar la memoria con eventos.

6. *Instalar una memoria en blanco.*
7. *Reiniciar la adquisición.*

Los acelerómetros utilizados por los usuarios son dos: el DCA333 de la empresa Terra Technology y el DSA1 de Kinematics. Con base en los nuevos requerimientos, en los apartados siguientes se describe el re-diseño del nuevo equipo, partiendo de la estructura general del sistema DIGI.

### 3.2. Hardware del sistema

Todos los componentes electrónicos, incluyendo los acelerómetros, circuitos integrados, elementos pasivos, módulos electrónicos, conectores y conductores eléctricos representan el hardware del equipo que trabajan en conjunto, manipulando señales eléctricas, para cubrir los requerimientos solicitados. La figura 3.2 muestra los bloques de los que está compuesto el sistema final.

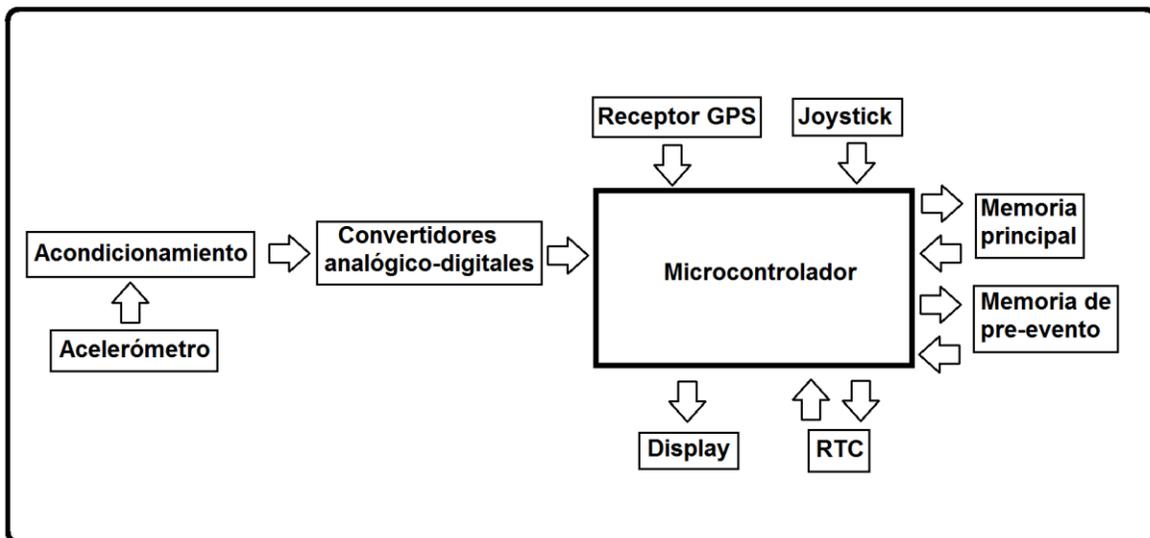


Figura 3.2. Diagrama de bloques del nuevo sistema.

El funcionamiento detallado de cada uno de estos bloques será realizado a continuación.

#### 3.2.1. Diseño de la etapa de acondicionamiento de la señal de entrada.

La etapa de acondicionamiento de señal de entrada tiene como propósito adecuar el voltaje de salida de los sensores al voltaje manejado por el sistema electrónico a desarrollar, especialmente con el rango de voltaje de los convertidores analógico-digitales, para ello, es necesario especificar en primer lugar el voltaje de alimentación del sistema, el rango de voltaje de entrada de los convertidores analógico digitales y el rango de voltaje de salida de los acelerómetros.

En cuanto a la alimentación del sistema, éste debe brindar la versatilidad para ser alimentado por baterías y eliminar la dependencia de una conexión al sistema de distribución de energía eléctrica, previniendo el mal funcionamiento frente a cortos circuitos o apagones inesperados. En la Coordinación de Sismología e Instrumentación Sísmica utilizan baterías de 12 [V] para alimentar sus equipos, por lo tanto, se eligió este valor de tensión para el funcionamiento del equipo.

Los convertidores analógico-digitales y, en general, la mayoría de los circuitos integrados modernos, funcionan a partir de 3.3 [V]; por otro lado, una cantidad grande de estos circuitos operan con un voltaje nominal de 5 [V]. Al elegir el nivel de tensión de 5 [V] se logra la compatibilidad entre la mayoría de los circuitos y en caso de ser necesario disminuir el voltaje, se puede optar por un arreglo de elementos pasivos que cubran esa necesidad; en otro caso, al elegir el nivel de tensión de 3.3 [V] se dificulta la versatilidad de conexión entre componentes.

Las características de los acelerómetros DCA333 y DSA1 se describieron en la sección 2.1, donde se señaló lo siguiente:

- DSA1. Salida de escala completa =  $\pm 2.5$  [V].
- DCA333. Salida de escala completa con respecto a la señal de pseudo-tierra =  $\pm 2.5$  [V].

Dicho lo anterior, el nuevo sistema, referido de ahora en adelante como *Registrador de Aceleraciones Sísmicas del Instituto de Ingeniería* o RAS-II, opera con un voltaje de alimentación nominal de 12 [V], además, incluye un regulador de tensión con voltaje de salida nominal de 5 [V] para alimentar a otros circuitos integrados.

Como puede apreciarse, la salida de los acelerómetros utilizados para esta aplicación tiene una variación positiva y negativa, de modo que al interconectar los acelerómetros con el sistema RAS-II, que opera únicamente con voltajes positivos, la variación de la señal en el sentido negativo se ve eliminada. Para solucionar este conflicto, la etapa de acondicionamiento de señal incluye un bloque para la adición de  $2.5 [V_{DC}]$  de corriente directa que se suma al voltaje proporcionado por los acelerómetros. Después de sumar la componente de directa, la señal tiene una variación de  $2.5 [V_{DC}] \pm 2.5 [V]$ , es decir, la señal varía en el rango de  $[0-5] [V]$  empleando tanto la variación positiva como negativa de los acelerómetros.

Como medida para aprovechar la utilización de sensores pasivos como los geófonos GS-11D de la aplicación anterior y, ahora, la inclusión de los acelerómetros DCA333 y DSA1, el sistema posibilita el uso de dos circuitos para utilizar sensores pasivos y activos.

### Sensores pasivos

En primer lugar, los sensores pasivos no requieren de una fuente de polarización, además, para el caso del geófono GS-11D la salida es del tipo unipolar flotante, es decir, la terminal de referencia es independiente de tierra.

Los sensores pasivos generalmente tienen un funcionamiento electromecánico, por ejemplo, para el caso del geófono, se utiliza un cilindro metálico en donde se aloja una bobina, por otro lado, se tiene una masa y una bobina suspendidas por un resorte. Cuando se somete al sensor a un movimiento vertical, la masa tiende a moverse y con ella la bobina, esta última pasa por las líneas de campo magnético del imán y con ello se genera una corriente eléctrica. Como el movimiento de la masa puede darse en dos sentidos, a cada una de las dos terminales del sensor le corresponde la corriente generada en el movimiento ascendente y en el movimiento descendente respectivamente.

De la misma manera que para el caso de los acelerómetros, la salida de los geófonos es una señal de voltaje con variación positiva y negativa, por este motivo, se debe agregar una componente de corriente directa para aprovechar todo el rango de variación de la salida de los sensores. El voltaje sumado corresponde a  $2.5 [V_{DC}]$  para lograr una excursión simétrica de la señal en el rango de voltaje utilizado por el circuito de acondicionamiento, que es de 0 a 5 [V]. Se optó por utilizar una configuración de divisor de tensión con dos resistencias del mismo valor en el orden de los  $[k\Omega]$ , a su vez, a la salida del divisor se conectó el geófono a través de dos resistencias de un alto valor ( $1 [M\Omega]$ ) para prevenir la circulación de corriente por las ramas del circuito y en consecuencia se degradara el nivel de tensión del divisor. El circuito se muestra en la figura 3.3.

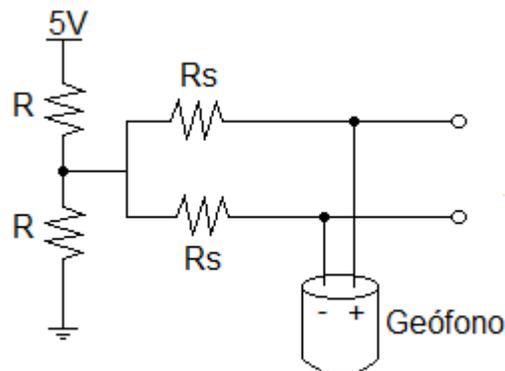


Figura 3.3. Adición de voltaje de directa al geófono.

Además de la adición de la componente de directa, es necesario realizar un filtrado a la señal para que ésta pueda ser digitalizada posteriormente y se evite el fenómeno de *aliasing*, en otras palabras, se necesita limitar en banda a la señal. Dado que el proceso de digitalización se realiza utilizando un convertidor analógico-

digital sigma-delta que utiliza el método de sobremuestreo, la banda de transición del filtro no tiene que ser tan estrecha y el decaimiento en la ganancia puede realizarse de forma suave con un filtro paso bajos de primer orden. Con esto en mente, se realiza el filtrado con el circuito mostrado en la figura 3.4.

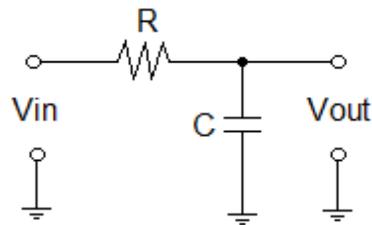


Figura 3.4. Filtro paso bajos.

Previendo de posibles daños al circuito de acondicionamiento de señal, el sistema dispone también de un arreglo de diodos de señal pequeña para limitar los niveles de voltaje de entrada a las etapas subsiguientes. El arreglo se muestra en la figura 3.5. Cuando el usuario introduce una señal superior a  $5 \text{ [V]} + V_D$ , donde  $V_D$  es el voltaje del diodo, el diodo 1 limita el voltaje de la señal a  $5 \text{ [V]}$ , por otro lado, cuando el usuario introduce una señal por debajo de  $0 \text{ [V]} - V_D$ , el diodo 2 limita el voltaje de la señal a  $0 \text{ [V]}$ .

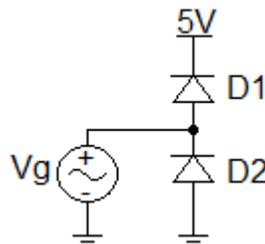


Figura 3.5. Circuito limitador de voltaje.

Finalmente, la primera etapa de acondicionamiento de señal para sensores pasivos se muestra en la figura 3.6.

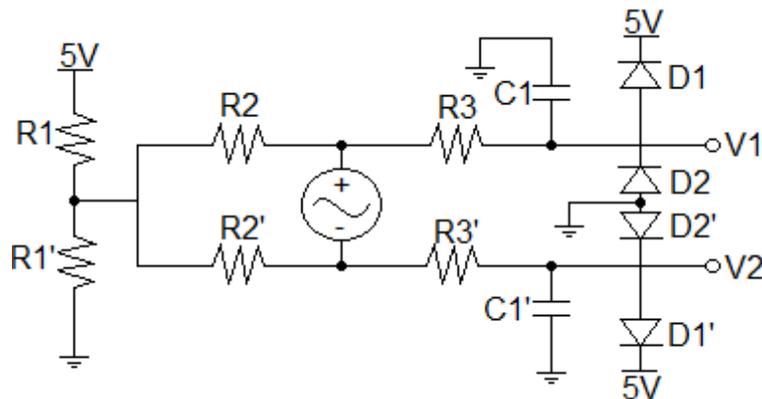


Figura 3.6. Acondicionamiento de sensores pasivos.

### Sensores activos

Los sensores activos requieren una fuente de polarización para poder operar, además, el tipo de salida de los sensores es diferente a la correspondiente a los geófonos GS-11D. Para el caso del sensor DSA1 la salida es unipolar con tensión en modo común mientras que para el sensor DCA333 la salida es unipolar puesta a tierra.

Un circuito utilizado para adicionar una componente de corriente directa y realizar un filtrado a la señal proveniente de un sensor activo es el mostrado en la figura 3.7.

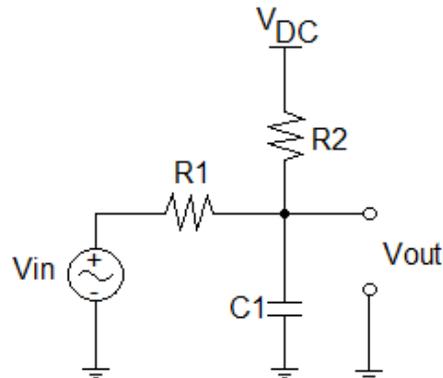


Figura 3.7. Circuito para acondicionar la salida de un sensor activo.

El circuito de la figura anterior cuenta únicamente con elementos lineales, motivo por el cual puede ser analizado por el teorema o principio de superposición que establece que:

**Teorema de superposición para circuitos eléctricos.** *En cualquier circuito resistivo lineal que contenga dos o más fuentes independientes, cualquier voltaje (o corriente) del circuito puede calcularse como la suma algebraica de todos los voltajes (o corrientes) individuales originados por cada fuente independiente actuando por sí sola y con todas las demás fuentes desactivadas.*<sup>1</sup>

De esta manera, para el circuito mostrado, considerando primero el efecto de la fuente de corriente directa y posteriormente el efecto de la fuente de corriente alterna, y resolviendo para el voltaje de salida  $V_{out}$ , se tiene:

$$V_{out} = \left(\frac{R_1}{R_1+R_2}\right)V_{DC} + \frac{1}{C_1 R_1} \left(\frac{1}{s + \frac{R_1+R_2}{C_1 R_1 R_2}}\right)V_{in} \quad (3.1)$$

Como se observa, el voltaje de salida  $V_{out}$  está conformado por una componente de DC (corriente directa) y una componente de AC (corriente alterna) modificada por la respuesta pasa bajos de un filtro de primer orden.

1. Johnson D., Hilburn J., *Análisis básico de circuitos eléctricos*, cuarta edición, Prentice Hall, USA, 1991, p. 131.

Para obtener una componente de directa de 2.5 [V], partiendo del voltaje regulado de 5 [V] con el cual cuenta el RAS-II, las resistencias R1 y R2 son del mismo valor, de esta manera, como se ha explicado, a la salida de los sensores se suma la componente de DC y se logra una excursión simétrica de la señal resultante que varía en el rango de [0-5] [V].

Para el caso del filtro de primer orden y después de establecer que  $R_1 = R_2$ , la frecuencia de corte está dada por:

$$f = \frac{2}{2\pi C_1 R_1} \text{ [Hz]} \quad (3.2)$$

De este modo, conociendo el ancho de banda de interés (que para el caso de esta aplicación es 12 [Hz]) y estableciendo el valor de R1 o C1, el valor del otro parámetro puede ser obtenido a partir de la ecuación (3.2). En el caso del circuito de acondicionamiento del RAS-II se calculó el valor de C1 después de fijar el valor de R1. La razón de fijar el valor de R1 se describe a continuación.

#### *Arreglo para la utilización de sensores pasivos y activos*

Al ofrecer la versatilidad de trabajar con dos tipos de sensores, la densidad de componentes del sistema se ve incrementada y en consecuencia, el costo, espacio, número de pistas conductoras y tiempo necesarios para la manufactura del equipo. La solución es simplificar los circuitos en uno solo, reutilizando los componentes que son compatibles en ambos circuitos pero facilitando al usuario el ajuste en la utilización de uno u otro tipo de sensor.

Para lograr el fin antes mencionado, el circuito general incluye cuatro conectores que el usuario puede manipular manualmente para optar por trabajar con sensores pasivos, o bien, con sensores activos. El circuito resultante de la combinación de ambos circuitos se muestra en la figura 3.8. En primer lugar, para operar con sensores pasivos, los conectores J2 y J3 deben ser conectados al mismo tiempo que J1 y J4 permanecen desconectados. Para operar con sensores activos, sucede exactamente lo opuesto, J1 y J4 deben mantenerse conectados mientras que J2 y J3 permanecen sin conectar. En esta figura se aprecia que el filtro es compartido para ambos circuitos, es decir, se comparten los elementos R3 y C1. Si el valor del capacitor C1 se fijara, las resistencias R3, R3', R4 y R4' tendrían que ser modificados para establecer una frecuencia de corte en particular. El circuito diseñado opta por fijar R3 (y en consecuencia R3', R4 y R4') y proporcionar conectores adecuados para el intercambio de los capacitores C1 y C1' y, de esta manera, cambiar la respuesta del filtro.

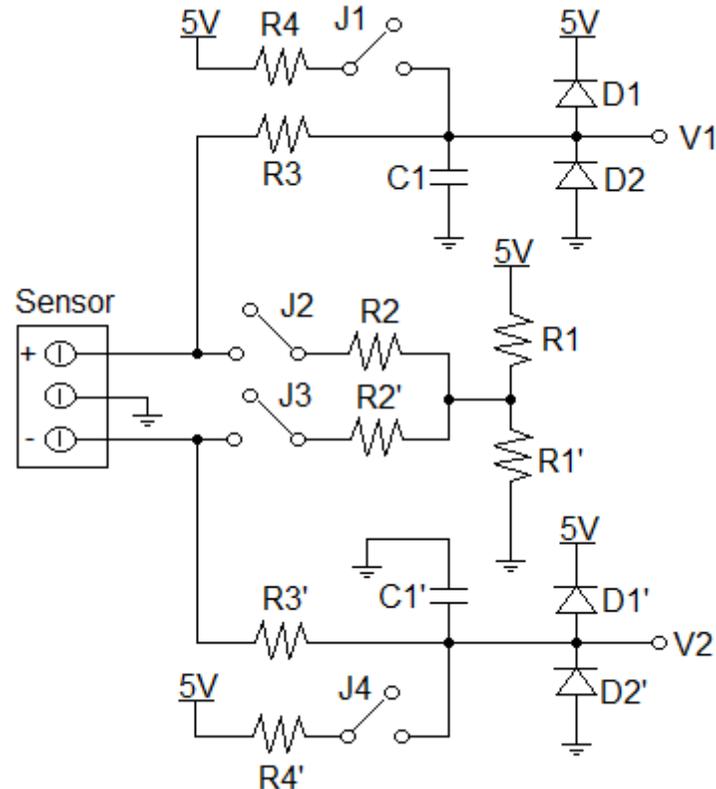


Figura 3.8. Combinación de ambos circuitos.

### *Acoplamiento de impedancias*

La siguiente etapa en el proceso de acondicionamiento de señal incluye un circuito para el acoplamiento de impedancias, que es de gran importancia sobre todo al utilizar sensores con alta impedancia de salida, esto último trae como consecuencia la degradación de la señal entrante y un efecto de sobrecarga a las etapas siguientes. Cuando los sensores poseen una baja impedancia de salida éstos pueden ser conectados directamente al convertidor analógico-digital, sin embargo, esto no sucede en todos los casos y es mejor brindar un correcto acoplamiento de impedancias entre el circuito generador de las señales y las entradas al convertidor analógico-digital.

Como dato previo, el convertidor analógico-digital utilizado posee una entrada de tipo diferencial, de modo que el resultado de la conversión analógico-digital corresponde al valor de voltaje diferencial presente en sus entradas. La entrada del convertidor analógico-digital realiza la función de un amplificador diferencial; tomando esta característica en cuenta, el acoplamiento de impedancias se realiza a través de dos buffers y una red que, en conjunto con el amplificador diferencial del ADC, conforman una configuración como la mostrada en la figura 3.9.

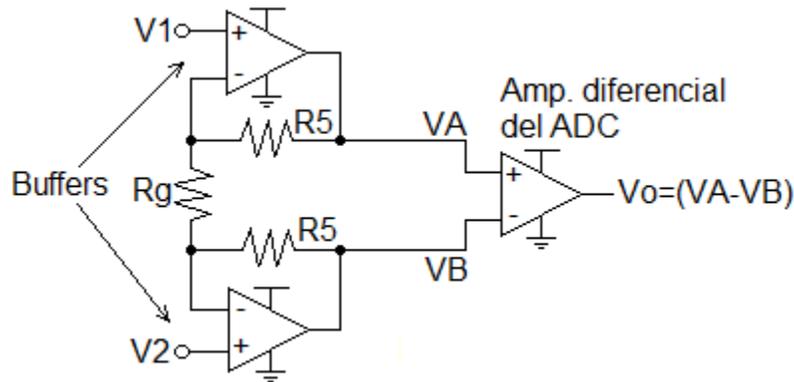


Figura 3.9. Amplificador de instrumentación.

La topología mostrada en la figura 3.9 corresponde a la de un amplificador de instrumentación y los valores de tensión para  $V_A$ ,  $V_B$  y  $V_o = (V_A - V_B)$  están dadas por las siguientes expresiones:

$$V_A = \left(1 + \frac{R_5}{R_g}\right) V_1 - \left(\frac{R_5}{R_g}\right) V_2 \quad (3.3)$$

$$V_B = -\left(\frac{R_5}{R_g}\right) V_1 + \left(1 + \frac{R_5}{R_g}\right) V_2 \quad (3.4)$$

$$V_o = \left(1 + 2 \frac{R_5}{R_g}\right) (V_1 - V_2) \quad (3.5)$$

De la expresión (3.5) se observa que el voltaje de salida es directamente proporcional al voltaje diferencial de entrada ( $V_1 - V_2$ ) y la constante de proporcionalidad es el factor  $1 + \frac{2R_5}{R_g}$ , de esta manera, si  $R_g$  es mucho menor que  $R_5$ , se tendrá una gran ganancia de voltaje; si  $R_g$  es mayor a  $R_5$  se tendrá una ganancia entre 1 y 3 y, finalmente, si  $R_g$  es igual a  $R_5$  se tendrá una ganancia de 3.

Para el caso de los acelerómetros DCA333 y DSA1 el nivel de voltaje en sus salidas no requiere amplificación pues su variación de  $\pm 2.5$  [V] es ideal para interconectar al sistema sin modificación, por este motivo se eligió  $R_g = 2R_5$ . Con la elección hecha se tiene una ganancia de 2, sin embargo, al conjuntar la primera y segunda etapas de acondicionamiento de señal, se tiene una ganancia general de 1, de este modo, la variación sin amplificación de  $\pm 2.5$  [V] de los acelerómetros que se suma a la componente de directa de  $2.5$  [ $V_{DC}$ ] se convierte en una señal idónea para ser manejada en el rango de entrada al convertidor analógico-digital.

El fabricante del convertidor analógico-digital propone además la inclusión de una red de realimentación como la mostrada en la figura 3.10.

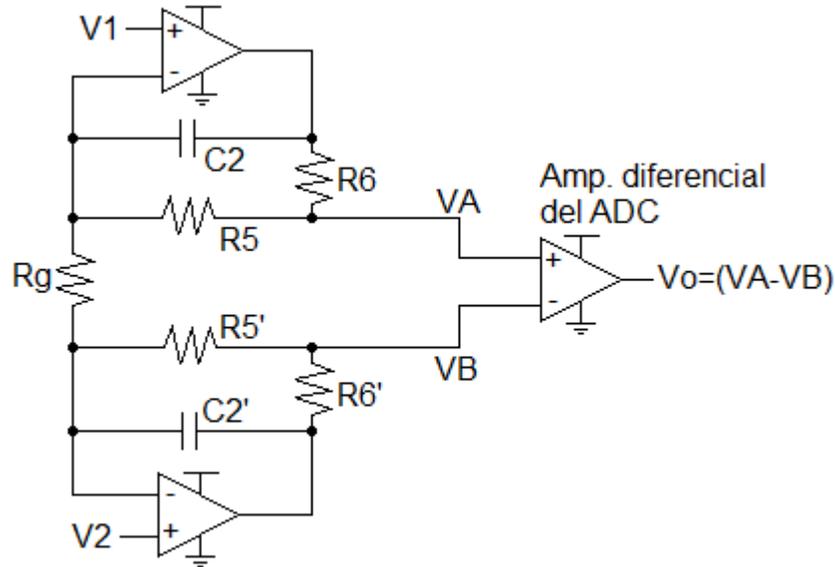


Figura 3.10. Configuración propuesta para ser utilizada con el ADC.

Para el circuito de la figura 3.10, las expresiones para los voltajes VA, VB y el voltaje diferencial  $V_o = (V_A - V_B)$ , considerando  $C_2 = C_2'$ ,  $R_5 = R_5'$  y  $R_6 = R_6'$ , están dados por las siguientes expresiones:

$$V_A = \left( \frac{R_5 + R_6}{R_5 R_6} \right) \left( \frac{S + \frac{R_5 + R_g}{C_2 R_g (R_5 + R_6)}}{S^2 + S \left( \frac{R_5 + R_6}{R_5 R_6} \right) + \frac{1}{C_2 R_5 R_6}} \right) V_1 - \left( \frac{1}{C_2 R_6 R_g} \right) \left( \frac{1}{S^2 + S \left( \frac{R_5 + R_6}{R_5 R_6} \right) + \frac{1}{C_2 R_5 R_6}} \right) V_2 \quad (3.6)$$

$$V_B = - \left( \frac{1}{C_2 R_6 R_g} \right) \left( \frac{1}{S^2 + S \left( \frac{R_5 + R_6}{R_5 R_6} \right) + \frac{1}{C_2 R_5 R_6}} \right) V_1 + \left( \frac{R_5 + R_6}{R_5 R_6} \right) \left( \frac{S + \frac{R_5 + R_g}{C_2 R_g (R_5 + R_6)}}{S^2 + S \left( \frac{R_5 + R_6}{R_5 R_6} \right) + \frac{1}{C_2 R_5 R_6}} \right) V_2 \quad (3.7)$$

$$V_o = V_A - V_B = \left( \frac{R_5 + R_6}{R_5 R_6} \right) \left( \frac{S + \frac{2R_5 + R_g}{C_2 R_g (R_5 + R_6)}}{S^2 + S \left( \frac{R_5 + R_6}{R_5 R_6} \right) + \frac{1}{C_2 R_5 R_6}} \right) (V_1 - V_2) \quad (3.8)$$

Por inspección se puede observar que al incluir los componentes C2 y R6 en la red de realimentación, la respuesta en frecuencia puede ser modificada, cosa que no ocurre si se utiliza la topología convencional de la figura 3.9. A partir de la expresión (3.8) se observa que la función de transferencia  $V_o / (V_A - V_B)$  incluye un cero y dos polos, que debido a los valores utilizados para R5, R6 y C2, se trata de un par de polos complejos conjugados.

Alguno de los dos parámetros C2 o R6 puede ser parametrizado para observar cómo cambia la respuesta en frecuencia al variarse el valor de dicho parámetro. Se eligió fijar el valor de C2 y parametrizar R6 puesto que existe una mayor variedad de valores de resistencias que valores de capacitores.

De acuerdo al valor de los componentes R5, R6 y C2 utilizados en el RAS-II, los polos de la ecuación (3.8) son complejos conjugados, cuyo polinomio característico es:  $S^2 + 2\xi\omega_0 S + \omega_0^2$ , donde  $\omega_0$  es la frecuencia de corte en radianes por segundo y  $\xi$  es el coeficiente de amortiguamiento. Igualando el denominador de la ecuación (3.8) con el polinomio característico de un par de polos complejos conjugados, se sabe que la frecuencia de corte  $f_p$  (en [Hz]) de los polos está dada por:

$$f_p = \sqrt{\frac{1}{4\pi^2 C_2 R_5 R_6}} \text{ [Hz]} \quad (3.9)$$

Por otro lado, la frecuencia de corte del cero  $f_z$  de la ecuación (3.8), está dada por:

$$f_z = \frac{1}{2\pi C_2 R_g (R_5 + R_6)} \text{ [Hz]} \quad (3.10)$$

En cuanto al valor de las resistencias R5 y Rg, estos parámetros ya han sido establecidos, además, se fija el valor de C2 a 0.1 [ $\mu$ F], por lo tanto, la única variable es el valor de R6. En la tabla 3.2 se muestran las frecuencias de corte del cero y del par de polos complejos conjugados al modificar el valor de R6.

R6 [ $\Omega$ ]	$f_p$ [Hz]	$f_z$ [Hz]
1000	0.4	1.27k
100	1.2	1.98k
10	4.1	2.10k
1	12.99	2.12k

Tabla 3.2. Frecuencias de corte al variar R6.

De la tabla anterior se observa que la frecuencia del cero se encuentra lejos de la banda base y su contribución no es de importancia, por otro lado, la elección de R6 puede adecuarse para satisfacer uno de los requerimientos del sistema, que consiste en la limitación del ancho de banda a 12 [Hz].

La figura 3.11 muestra la respuesta en frecuencia, tanto en magnitud (normalizada) como en fase, del circuito de la figura 3.10 al parametrizar R6. El sobresalto observado en la respuesta en magnitud se debe a que el coeficiente de amortiguamiento es muy cercano a cero debido a los valores utilizados para R5 y R6.

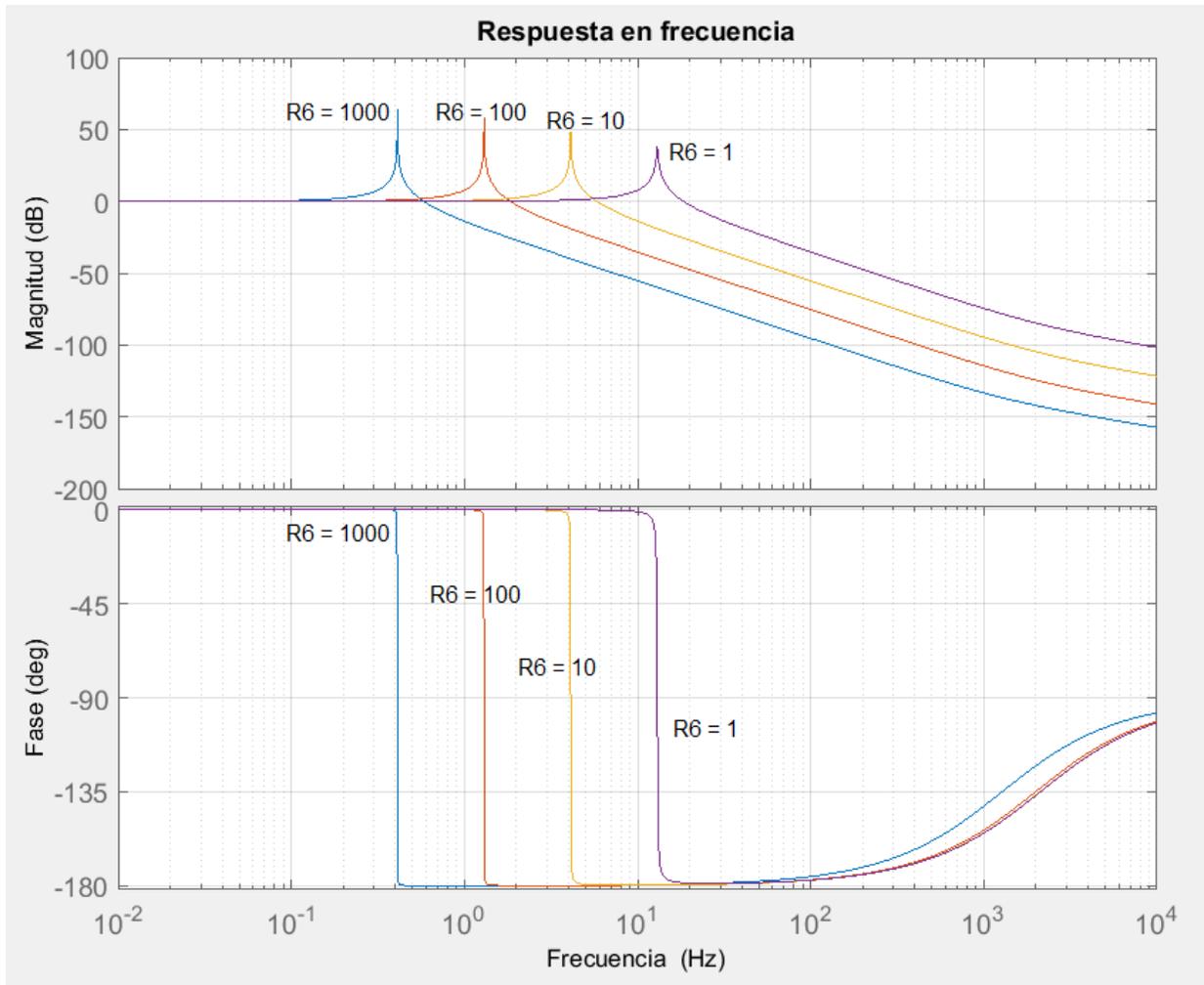


Figura 3.11. Respuesta en frecuencia al variar  $R_6$ .

Con base en la tabla 3.2 y la figura 3.11, se eligió el valor de  $R_6$  que ajusta el ancho de banda del amplificador de instrumentación al rango de frecuencias de interés para los usuarios.

El circuito final para el acondicionamiento de señales se muestra en la figura 3.12.

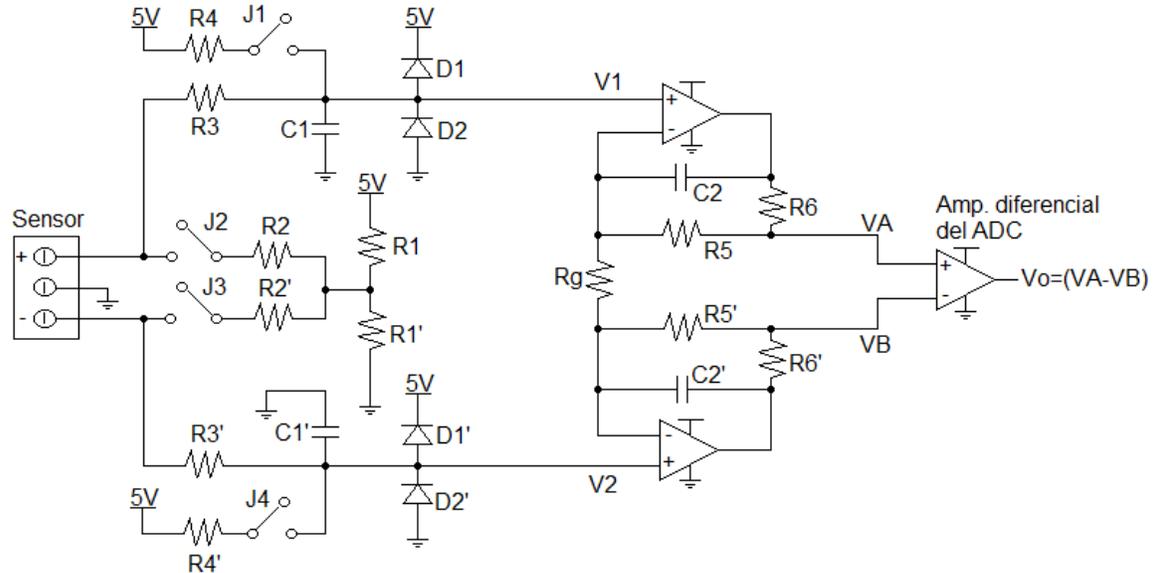


Figura 3.12. Circuito de acondicionamiento de señal.

Finalmente, el diseño físico del sistema incluye tres circuitos como el de la figura 3.12, que corresponden a las señales en las direcciones vertical, transversal y longitudinal que proporcionan los acelerómetros.

### 3.2.2. Convertidor analógico-digital

Después de lograr que las tensiones provenientes de los acelerómetros sean compatibles con las entradas del ADC, las señales en los tres canales entran al proceso de conversión analógico-digital. La conversión se realiza con ayuda de tres convertidores analógico-digitales con arquitectura sigma-delta del fabricante Linear Technology. Los convertidores, que por su esquema de modulación sigma-delta (de tercer orden) ofrecen una alta resolución, se eligieron por características que los distinguen de otros convertidores analógico-digitales, entre ellas: costo, alta resolución, elevada tasa de conversión y salida de datos, tipo de comunicación, tamaño y número de terminales. Algunas de las características generales de los convertidores utilizados son las siguientes:

#### Rangos máximos absolutos

- Voltaje de polarización ( $V_{cc}$ ):  $-0.3 [V]$  a  $6 [V]$
- Voltaje en las terminales de entrada (señal):  $-0.3 [V]$  a  $(V_{cc} + 0.3) [V]$
- Voltaje en las terminales de referencia:  $-0.3 [V]$  a  $(V_{cc} + 0.3) [V]$

#### Características de voltaje

- Voltaje positivo de entrada ( $IN^+$ ):  $-0.3 [V]$  a  $(V_{cc} + 0.3) [V]$
- Voltaje negativo de entrada ( $IN^-$ ):  $-0.3 [V]$  a  $(V_{cc} + 0.3) [V]$
- Rango del voltaje diferencial de entrada ( $IN^+ - IN^-$ ):  $-V_{REF}/2$  a  $V_{REF}/2$
- Voltaje positivo de referencia ( $REF^+$ ):  $0.1 [V]$  a  $V_{cc}$
- Voltaje negativo de referencia ( $REF^-$ ):  $GND$  a  $(V_{cc} - 0.1) [V]$
- Voltaje diferencial de referencia ( $V_{REF}$ ): mínimo =  $0.1 [V]$ ; máximo =  $V_{cc}$

**Características eléctricas**

- Resolución: mínima = 17 bits; máxima = 24 bits
- Nivel de ruido: mínimo = 200 [nV]; máximo = 23 [ $\mu$ V]

**Características de consumo**

- Corriente de polarización (conversión): 8 [mA]
- Corriente de polarización (modo de reposo): 8 [ $\mu$ A]

**Características de tiempo**

- Tiempo de conversión: mínimo = 284 [ $\mu$ s]; máximo = 145 [ms]
- Tiempo de salida de datos: 35.3 [ $\mu$ s]
- Tasa de muestreo: 1.8 [Msps] (Mega muestras por segundo)

**Características generales**

- Temperatura de operación: -40 [°C] a 85 [°C]
- Tipo de comunicación: SPI.

La velocidad/resolución de un convertidor puede ser elegida y programada de entre 11 posibles combinaciones (pero una repetida), utilizando la línea MOSI del Bus SPI. El ajuste de la tasa de sobre muestreo (OSR, Oversampling Rate) fija la velocidad de conversión de datos y, al mismo tiempo, la resolución del convertidor analógico-digital. La figura 3.13, tomada del manual de usuario del ADC utilizado, muestra las opciones disponibles para la configuración de la tasa de sobre muestreo.

OSR4	OSR3	OSR2	OSR1	OSR0	CONVERSION RATE		RMS NOISE	ENOB	OSR
					INTERNAL 9MHz CLOCK	EXTERNAL 10.24MHz CLOCK			
X	0	0	0	1	3.52kHz	4kHz	23 $\mu$ V	17	64
X	0	0	1	0	1.76kHz	2kHz	3.5 $\mu$ V	20	128
0	0	0	0	0	880Hz	1kHz	2 $\mu$ V	21.3	256*
X	0	0	1	1	880Hz	1kHz	2 $\mu$ V	21.3	256
X	0	1	0	0	440Hz	500Hz	1.4 $\mu$ V	21.8	512
X	0	1	0	1	220Hz	250Hz	1 $\mu$ V	22.4	1024
X	0	1	1	0	110Hz	125Hz	750nV	22.9	2048
X	0	1	1	1	55Hz	62.5Hz	510nV	23.4	4096
X	1	0	0	0	27.5Hz	31.25Hz	375nV	24	8192
X	1	0	0	1	13.75Hz	15.625Hz	250nV	24.4	16384
X	1	1	1	1	6.875Hz	7.8125Hz	200nV	24.6	32768**

\*\*Address allows tying SDI HIGH \*Additional address to allow tying SDI LOW

Figura 3.13. Programación de la tasa de sobre muestreo.

Para configurar la tasa de sobremuestreo, un microcontrolador debe generar los bits OSR 4:0 y enviarlos por la línea MOSI del Bus SPI al convertidor. Dependiendo del valor de OSR elegido, la velocidad de conversión y resolución quedan fijas y la relación entre estos parámetros es inversamente proporcional, de modo que al aumentar la velocidad en la conversión de los datos, la resolución, medida con el número efectivo de bits (ENOB), se ve reducida. De acuerdo con el fabricante, el

ruido del ADC aumenta en un factor de  $\sqrt{2}$  cuando la OSR disminuye en un factor de 2, esto sucede para todos los casos excepto con las dos tasas de conversión más altas (1.76 y 3.52 [kHz]), en donde el nivel de ruido del ADC se ve incrementado por múltiples factores que rompen con la regla.

La velocidad de conversión de datos puede ser ajustada de modo que se adecue a los requerimientos del usuario. Al elegir una tasa de conversión de datos suficientemente mayor a la frecuencia de muestreo que el usuario desea implementar, éste puede decidir en qué momento activar el circuito integrado y leer el valor digitalizado, de modo que se ajuste a una frecuencia de muestreo determinada.

El consumo de potencia se puede optimizar eligiendo una frecuencia (esta es la frecuencia de muestreo requerida por el usuario) apropiada para la activación del circuito integrado. El ADC tiene un consumo de corriente de 8 [mA] durante el tiempo de conversión que es independiente de la OSR elegida; una vez que el valor se tiene listo, el convertidor entra en un estado de reposo con un consumo de 8 [ $\mu$ A] en tanto se mantenga inactiva la selección del circuito integrado, de esta manera, a medida que la frecuencia con que se activa el circuito integrado sea disminuida, se requerirá menor uso de potencia.

Una de las ventajas que ofrecen los convertidores analógico-digitales sigma-delta con respecto a otro tipo de convertidores, es el esquema de sobremuestreo y filtrado digital en el mismo chip, lo cual simplifica los requerimientos del filtro anti aliasing previo al convertidor. La tasa de sobremuestreo de este convertidor es 1.8 [MHz]  $\pm$  5%, sujeta a variaciones de polarización y temperatura. La tasa de sobremuestreo es constante e independiente de la OSR elegida que, de acuerdo al fabricante, permite que la linealidad, error de offset y error de escala completa sean constantes. La relación entre la tasa de sobremuestreo, OSR y tasa de conversión de datos está dada por la siguiente expresión:

$$tasa\ de\ conversi3n = \frac{tasa\ de\ sobre\ muestreo}{8 \times OSR} = \frac{1.8\ [MHz]}{8 \times OSR} \quad (3.11)$$

El convertidor analógico-digital cuenta con 16 terminales, de las cuales, 4 corresponden a la conexión a tierra (GND). El tipo de empaquetado utilizado es el SSOP de 16 terminales, éstas se muestran en la figura 3.14. Las terminales del ADC se describen a continuación.

*GND (terminales 1, 8, 9 y 16).* Corresponden con la conexión a tierra del circuito integrado, el uso de múltiples conexiones promueve el flujo de corriente adecuado hacia tierra y al mismo tiempo, el desacoplo de  $V_{cc}$  (utilizando capacitores de desacoplo).

*Vcc (terminal 2).* Corresponde a la terminal de entrada para el voltaje de alimentación. El fabricante recomienda conectar un capacitor de desacoplo de 10

[ $\mu\text{F}$ ] en paralelo con uno de 0.1 [ $\mu\text{F}$ ] lo más cercano posible a la terminal de alimentación del circuito integrado.

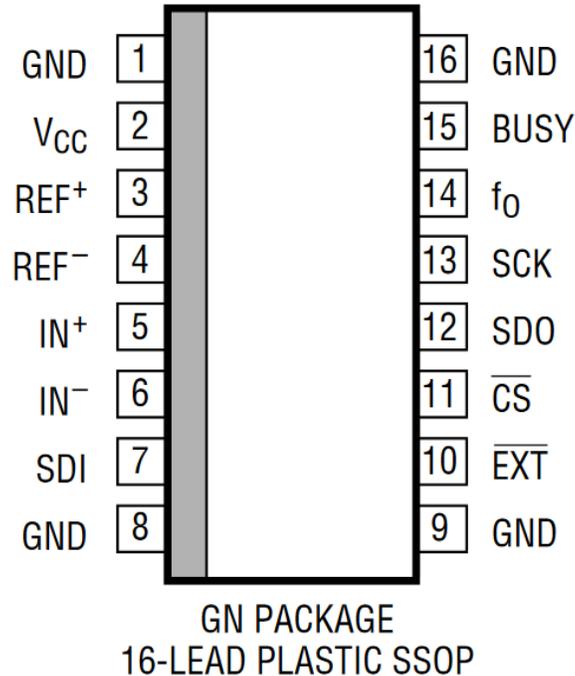


Figura 3.14. Distribución de terminales del ADC.

*REF<sup>+</sup>* (terminal 3), *REF<sup>-</sup>* (terminal 4). Corresponden a las terminales de entrada para las tensiones de referencia. El voltaje en cualquiera de las dos terminales puede tener cualquier valor entre GND y ( $V_{CC}+0.3$  [V]), siempre y cuando el voltaje de referencia positivo sea superior al voltaje de referencia negativo al menos por un nivel de tensión de 0.1 [V].

*IN<sup>+</sup>* (terminal 5), *IN<sup>-</sup>* (terminal 6). Corresponde a las terminales del voltaje diferencial de entrada. El nivel de voltaje en estas terminales puede tener cualquier valor entre -0.3 [V] y ( $V_{CC}+0.3$  [V]). Dentro de éstos límites, el rango de voltaje diferencial ( $IN^{+} - IN^{-}$ ) va desde  $-0.5V_{REF}$  a  $0.5 V_{REF}$ . Para el caso del RAS-II, con  $V_{REF} = 5[V]$ , el rango del voltaje diferencial de entrada es de -2.5 [V] a 2.5 [V].

*SDI*, *Serial Data Input* (terminal 7). Corresponde a la terminal para la configuración de la velocidad/resolución del convertidor analógico-digital a través de la línea MOSI del Bus SPI. La velocidad/resolución puede ser programada por el usuario al enviar 5 bits correspondientes a la OSR por la línea MOSI desde el microcontrolador hacia el convertidor. De manera alternativa, colocando la terminal SDI a tierra o a  $V_{CC}$ , el convertidor queda configurado con una OSR de 256 o 32768 respectivamente.

$\overline{\text{EXT}}$ , *External* (terminal 10). Corresponde a la selección de la fuente de reloj interna o externa. Cuando esta terminal es puesta a tierra el convertidor realiza el envío de datos hacia el microcontrolador mediante el control de una señal de reloj externa

proporcionada por el mismo controlador u otro dispositivo. Cuando la terminal es puesta a  $V_{cc}$ , la salida de datos se realiza mediante el control de una señal de reloj generada por el mismo convertidor analógico-digital.

$\overline{cs}$ , *Chip Select* (terminal 11). Corresponde a la selección del circuito integrado. Cuando la terminal  $\overline{cs}$  es puesta a un valor lógico bajo, se genera la habilitación de la terminal SDO para la salida de datos. Después de cada proceso de conversión, el ADC automáticamente entra en un estado de reposo y se mantiene en dicho estado en todo momento en que la señal  $\overline{cs}$  cuente con un nivel lógico alto. Cuando se genera una transición de un valor lógico alto a bajo durante un proceso de transferencia de datos hacia el microcontrolador, el proceso es suspendido y se da lugar a una nueva conversión.

*SDO, Serial Data Output* (terminal 12). Es una terminal de tres estados para la salida de datos. Cuando la señal  $\overline{cs}$  se mantiene en un nivel lógico alto, la terminal SDO se mantiene en alta impedancia. Cuando el dato convertido se encuentra listo y la señal  $\overline{cs}$  es llevada a un valor lógico bajo, el dato es transferido desde la terminal SDO del ADC hacia el microcontrolador. El dato convertido es enviado por la línea MISO del Bus SPI.

*SCK* (terminal 13). Corresponde a una terminal bidireccional para la señal de reloj. Cuando la terminal  $\overline{EXT}$  es conectada a tierra, el flujo de salida de datos por la línea SDO se realiza bajo el control de una señal de reloj externa proporcionada por el microcontrolador u otro dispositivo. Cuando la señal  $\overline{EXT}$  es puesta a  $V_{cc}$ , el ADC genera su propia fuente de reloj para el control de salida de datos, además la señal de reloj sale a través de la terminal SCK del ADC.

$f_o$  (terminal 14). Se utiliza para seleccionar el tipo de oscilador de trabajo del convertidor analógico-digital. Cuando la terminal  $f_o$  es puesta a tierra, el ADC utiliza un oscilador interno que opera a una frecuencia de 9 [MHz], por otro lado, la terminal  $f_o$  puede ser manejada por un oscilador externo cuyo rango de frecuencia puede ser de 100 [kHz] a 20 [MHz]. El tipo de oscilador utilizado para la operación del ADC está relacionado directamente con la tasa de conversión. A medida que se provee al ADC de un oscilador de mayor frecuencia, la velocidad de conversión incrementa.

*BUSY* (terminal 15). Se utiliza para conocer el estado del ADC, cuando el convertidor se encuentra en los estados de reposo y salida de datos, el estado de la terminal BUSY es bajo. Cuando el convertidor se encuentra en un proceso de conversión la terminal BUSY se mantiene en un nivel lógico alto.

### *Operación del ADC*

El funcionamiento del convertidor analógico-digital está constituido, a grandes rasgos, de un proceso que consta de tres etapas: conversión, reposo y salida de datos. La figura 3.15 esquematiza el funcionamiento del convertidor mediante un diagrama de flujo.

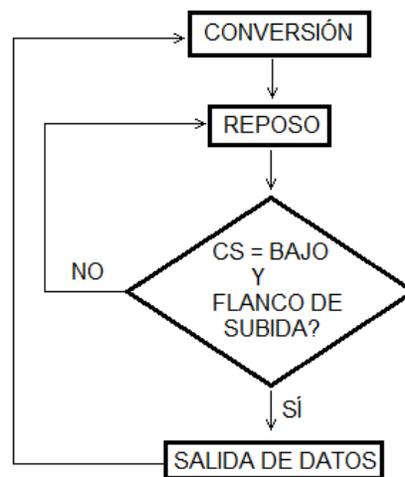


Figura 3.15. Ciclo de operación del ADC.

Inicialmente el convertidor analógico-digital realiza una conversión del voltaje suministrado en sus entradas  $IN^+$  e  $IN^-$ , realizando la diferencia entre ambas y posteriormente sometiendo el voltaje diferencial a la modulación sigma-delta y filtrado digital.

Una vez que el convertidor culmina la conversión y tiene el dato listo, automáticamente entra en un estado de reposo y bajo consumo de corriente independientemente del estado de la terminal  $\overline{CS}$ . El dato convertido permanece indeterminadamente en un registro estático en tanto la señal  $\overline{CS}$  sea mantenida en un nivel lógico alto.

Una vez que la señal  $\overline{CS}$  es llevada a un estado lógico bajo (se activa el circuito integrado) y se genera un flanco de subida en la señal de reloj SCK, comienza la transferencia de datos desde el ADC hacia el microcontrolador. Cada dato es actualizado en el flanco de bajada de modo que el usuario puede leer un bit estable durante el flanco de subida de la señal de reloj. Esto se aprecia mejor observando la figura 3.16 tomada del manual de usuario antes especificado.

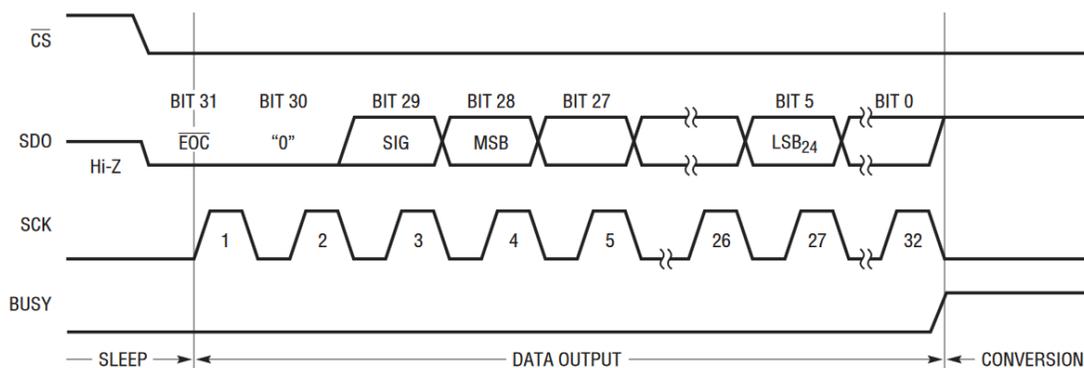


Figura 3.16. Señales manejadas durante la transferencia de datos.

El proceso se repite una vez que la cadena de 32 bits (de los cuáles consta el dato convertido) ha sido transmitida o, una vez que se interrumpa la transferencia de datos al microcontrolador llevando la terminal  $\overline{cs}$  a un estado lógico alto (desactivando el circuito integrado).

#### Formato del dato convertido

La cadena correspondiente al dato convertido consta de 32 bits, de los cuales, los 3 bits más significativos representan información del signo y rango de voltaje de entrada, los siguientes 24 bits conciernen al resultado de la conversión y finalmente, los últimos 5 bits pueden verse como bits menos significativos para aplicaciones de ultra alta resolución. Los bits que conforman la cadena son los siguientes:

**Bit 31.** Corresponde al primer bit de salida y al estado de  $\overline{EOC}$  que indica el estado de la conversión. Se puede leer el valor lógico de este bit (manteniendo la señal  $\overline{cs}$  en bajo) durante el proceso de conversión y estado de reposo. El bit permanece en alto durante la conversión y conmuta a un valor bajo una vez que el dato se encuentra listo.

**Bit 30.** Corresponde a un bit sin utilidad y siempre tiene un valor lógico bajo.

**Bit 29.** Indica el signo del nivel de voltaje convertido. Si el voltaje de entrada es negativo, este bit toma un valor alto, en caso contrario, toma un valor bajo.

**Bit 28.** Corresponde al bit más significativo. En conjunto con el bit 29 proporcionan información acerca del rango de voltaje de entrada convertido. Para el caso del RAS-II, en la tabla 3.3 se muestran los rangos del voltaje de entrada al ADC utilizando  $V_{REF} = 5 [V]$ .

Rango de voltaje	Bit 31 $\overline{EOC}$	Bit 30	Bit 29 Signo	Bit 28 MSB
$V_{IN} \geq 0.5 [V]$	0	0	1	1
$0 [V] \leq V_{IN} \leq 2.5 [V]$	0	0	1	0
$-2.5 [V] \leq V_{IN} \leq 0 [V]$	0	0	0	1
$V_{IN} < -2.5 [V]$	0	0	0	0

Tabla 3.3. Valores de los MSB asociados al voltaje de entrada.

**Bits 28 a 5.** Corresponden al resultado de la conversión con el MSB en la primera posición.

**Bits 4 a 0.** Corresponden a bits menos significativos que pueden ser incluidos opcionalmente en el resultado final de la conversión.

### Modos de operación.

El convertidor analógico-digital cuenta con 4 modos de operación:

1. *Operación de ciclo sencillo y reloj externo.* Para seleccionar este modo de operación la terminal  $\overline{EXT}$  debe ponerse a un valor lógico bajo. La operación inicia con un proceso de conversión, a continuación el ADC entra en un estado de reposo y permanece en él en tanto la señal  $\overline{CS}$  sea mantenida en un nivel alto. Durante el proceso de conversión y estado de reposo el usuario puede leer el valor del bit  $\overline{EOC}$  o BUSY con el fin de saber si el dato se encuentra listo. Una vez que el dato se encuentra listo y el usuario activa el circuito integrado, comienza el proceso de salida de datos del convertidor con ayuda de una señal de reloj externa.
2. *Operación con reloj externo y dos conductores.* Para seleccionar este modo de operación, la terminal  $\overline{EXT}$  y  $\overline{CS}$  deben mantenerse en un nivel lógico bajo. Debido a que la terminal  $\overline{CS}$  siempre tiene un valor lógico bajo, la selección del circuito integrado es permanente. Una vez que la conversión ha terminado, el ADC iniciará con el proceso de salida de datos siempre y cuando se proporcione la señal de reloj externa. Al finalizar con el envío de los 32 bits de los cuales consta la cadena el ADC inicia una nueva conversión.
3. *Operación de ciclo sencillo y reloj interno.* El modo de operación es exactamente igual al primer modo de operación con la excepción de que el ADC genera su propia fuente de reloj para el control del flujo de salida de datos.
4. *Operación con reloj interno y dos conductores.* El modo de operación es exactamente igual al segundo modo de operación a excepción de que el ADC trabaja con su propia fuente de reloj.

Para el caso del RAS-II se eligió el primer modo de operación, utilizando tres convertidores analógico-digitales compartiendo las líneas SCK y SDO del bus SPI, además, se dispuso de tres terminales diferentes del microcontrolador para la activación de cada circuito integrado. La conexión entre el microcontrolador y los convertidores se esquematiza en la figura 3.17.

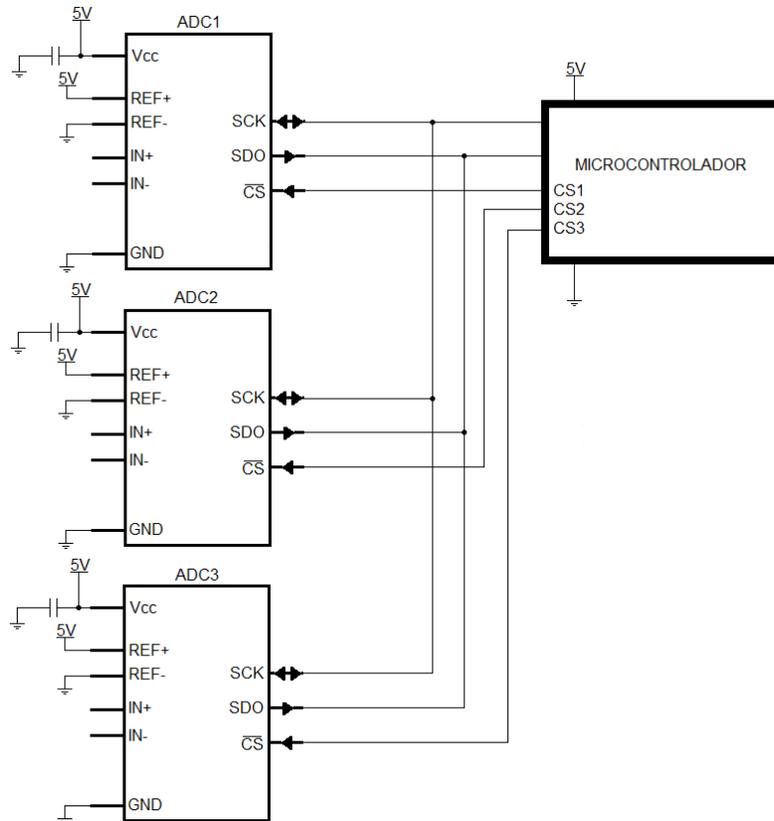


Figura 3.17. Conexión entre convertidores y microcontrolador.

### 3.2.3. Microcontrolador

Una vez que las señales provenientes de los acelerómetros han sido acondicionadas y posteriormente convertidas por el ADC en señales digitales equivalentes, el siguiente paso comprende el procesamiento y manipulación de éstas por parte del microcontrolador; dicha manipulación tiene que ver con el almacenamiento de los niveles de voltaje digitalizados, en variables que el microcontrolador utilizará a lo largo de la ejecución del programa informático grabado en su memoria.

Considerando el tipo de operaciones que el microcontrolador tendría que realizar, del extenso espectro de microcontroladores en el mercado, se eligió uno en particular con base en características como: costo, amplia gama de documentación y herramientas de desarrollo, extendido uso en aplicaciones de electrónica, ancho de palabra adecuado (8 bits) para el tipo de operaciones a realizar, frecuencia de reloj suficiente (16 [MHz]) para la ejecución de las instrucciones, puertos de entrada/salida necesarios para la aplicación (27 puertos), capacidad de la memoria de programa (256 [kB]), interfaces de comunicación soportadas (mediante UART, SPI e I2C), entre otras.

El microcontrolador pertenece a la familia AVR del fabricante estadounidense Atmel. Algunas de las características de la familia AVR son: arquitectura Harvard, modelo

de conjunto de instrucciones RISC avanzado, arquitectura basada en 32 registros de 8 bits, inclusión de memoria flash, EEPROM y SRAM en el mismo circuito integrado, técnica de *pipelining*, en donde la siguiente instrucción máquina es leída mientras la instrucción actual está siendo ejecutada, permitiendo que el tiempo de ejecución para la mayoría de las instrucciones sea de un ciclo de reloj, etc.

Algunas de las especificaciones técnicas del microcontrolador utilizado son las siguientes:

#### *Alimentación*

- *Voltaje de alimentación: 4.5 a 5.5 [V]*
- *Consumo de corriente: 500 [ $\mu$ A]*

#### *Frecuencia de operación*

- *Frecuencia máxima: 16 [MHz]*

#### *Arquitectura*

- *Arquitectura Harvard*
- *CPU: 8 bits*
- *Modelo de conjunto de instrucciones: RISC avanzado*
- *135 instrucciones máquina, ejecución en un ciclo de reloj*
- *32 registros de 8 bits de propósito general*
- *Hasta 16 MIPS (mega instrucciones por segundo) con reloj de 16 [MHz]*

#### *Memoria*

- *256 [kB] de memoria de programa (memoria flash)*
- *4 [kB] de memoria EEPROM*
- *8 [kB] de memoria de datos (memoria SRAM)*
- *Ciclos de lectura/escritura:  $1 \times 10^4$  en memoria flash y  $1 \times 10^5$  en memoria EEPROM*
- *Retención de datos: 20 años a 85 [°C] y 100 años a 25 [°C]*

#### *Periféricos*

- *Dos temporizadores/contadores de 8 bits*
- *Cuatro temporizadores/contadores de 16 bits*
- *12 unidades de PWM*
- *16 canales de convertidor analógico-digital de 10 bits*
- *4 módulos USART (Universal Synchronous/Asynchronous Receiver/Transmitter)*
- *Interfaz SPI*
- *Interfaz TWI*
- *Temporizador Watchdog con oscilador en circuito integrado independiente*

#### *Puertos de entrada/salida*

- *86 puertos programables de entrada/salida*

#### *Características de temperatura*

- *Rango de temperatura de operación: -40 [°C] a 85 [°C]*

Otras características

- Tecnología de fabricación: CMOS
- Tipo de empaquetado: QFP (Quad Flat Package)

La distribución de terminales del microcontrolador se muestra en la figura 3.18, tomada de su manual de usuario.

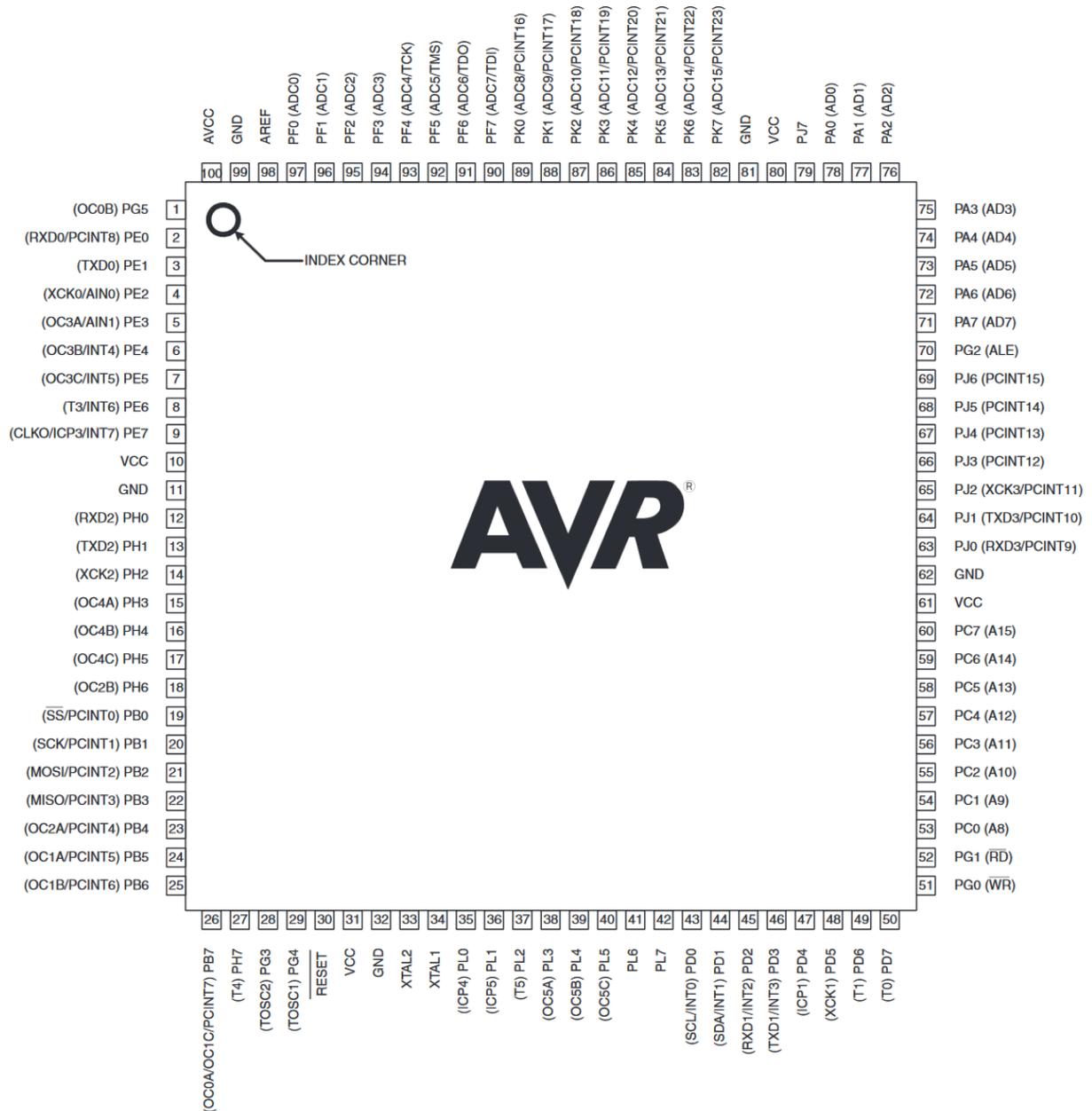


Figura 3.18. Distribución de terminales del microcontrolador.

El diagrama de bloques del microcontrolador se muestra en la figura 3.19, tomada del documento antes especificado. Entre los bloques funcionales se encuentran:

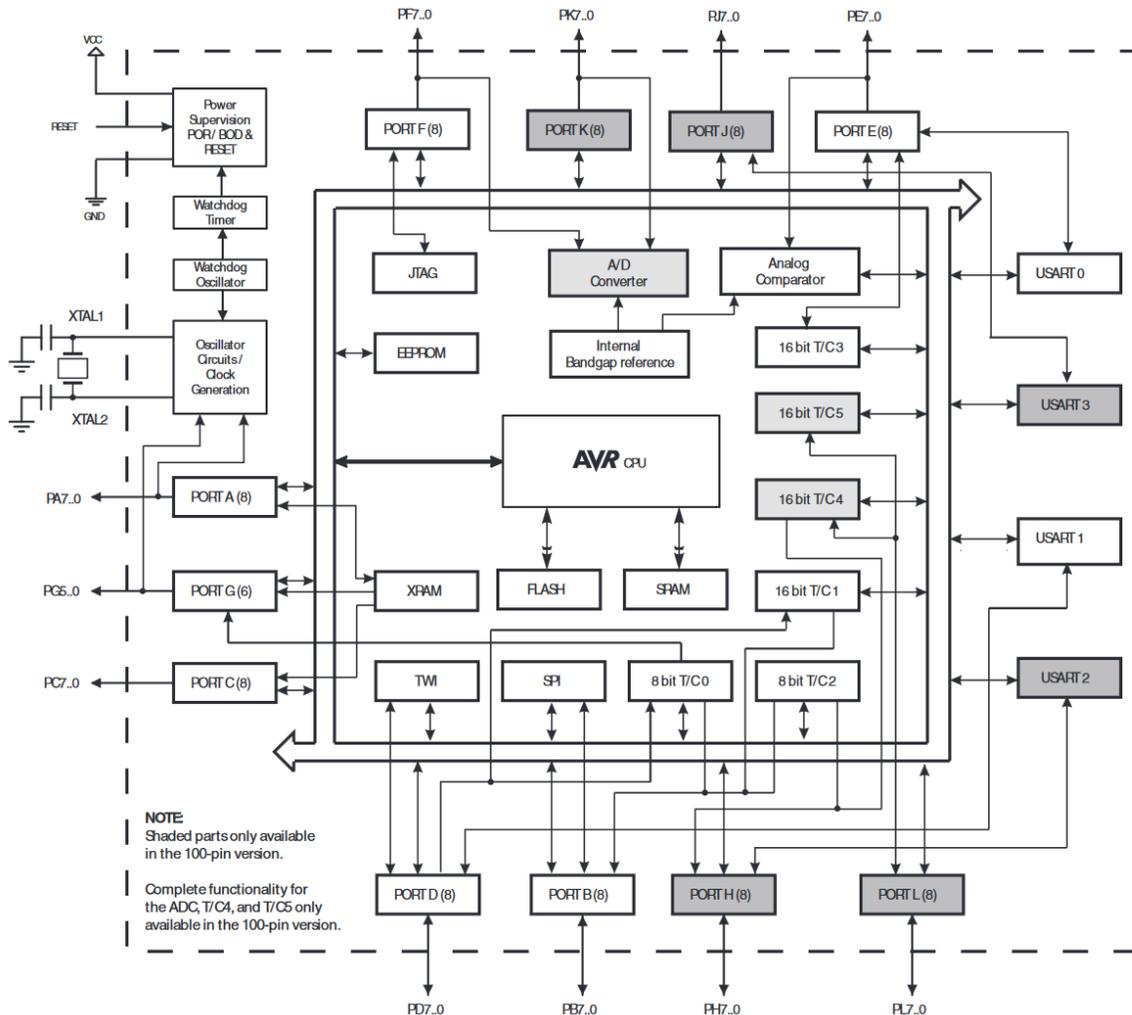


Figura 3.19. Diagrama de bloques del microcontrolador.

- **CPU.** El procesador cuenta con una ALU, un contador de programa, un apuntador de pila, un registro de instrucción, un registro de estado y control y 32 registros de propósito general. El tipo de arquitectura del CPU permite la lectura de una nueva instrucción mientras una instrucción actual se encuentra ejecutándose (*pipelining*).
- **Memorias.** Como se ha mencionado, el tipo de arquitectura implementado en los microcontroladores de la familia AVR, incluye una memoria flash para los datos de programa, una memoria SRAM para las variables manejadas en el programa y una memoria EEPROM para diversos usos, todo esto dentro del mismo circuito integrado. La memoria flash cuenta con una capacidad de 256

[kB] y soporta al menos 10,000 ciclos de lectura/escritura; por su parte, la memoria SRAM cuenta con una capacidad de 8 [kB] y soporta al menos 100,000 ciclos de lectura/escritura. Cada memoria utiliza su propio bus de comunicación con el procesador. La memoria flash contiene dos regiones, una destinada al almacenamiento del programa de aplicación y otra destinada al almacenamiento de un programa de arranque (opcional) que permite la carga de un programa al microcontrolador mediante comunicación serial con una computadora de escritorio. La memoria EEPROM cuenta con una capacidad de almacenamiento de 4 [kB] y soporta al menos 100000 ciclos de lectura/escritura; se encuentra en un espacio totalmente diferente a las otras dos memorias y se puede acceder a ella con ayuda de un par de registros de control.

- *Puertos.* El microcontrolador cuenta con 11 puertos de entrada/salida programables, éstos son: puerto A, puerto B, puerto C, puerto D, puerto E, puerto F, puerto G, puerto H, puerto J, puerto K y puerto L. Todos los puertos cuentan con resistencias de *pull-up* programables, también todos a excepción del puerto G son de 8 bits, por su parte, el puerto G consta de 5 bits. Los puertos tienen funciones alternativas como interfaces de comunicación, entradas analógicas para lectura de voltaje, salidas para las unidades de conteo y comparación, etcétera; los puertos pueden ser programados para lograr tales fines.
- *Sistema de reloj.* El microcontrolador cuenta con una unidad para la generación de la fuente de reloj para la memoria flash y EEPROM, temporizadores, ADCs, CPU, SRAM, *Watchdog* y otros módulos. La frecuencia de reloj por default es obtenida de un oscilador RC interno a 8 [MHz] con el fusible CKDIV8 programado, resultando en una frecuencia de trabajo de 1 [MHz]. Las fuentes de reloj incluyen: un oscilador dentro del circuito integrado que necesita la conexión de un cristal de cuarzo externo, un oscilador RC interno, un oscilador de 128 [kHz] interno y una señal de reloj generada con un dispositivo externo. El sistema de reloj cuenta con un pre-escalador que permite transferir una fracción de la frecuencia original hacia los demás módulos dentro del microcontrolador.
- *Unidad de manejo de potencia.* Permite habilitar y deshabilitar algunos módulos dentro del microcontrolador con el fin de hacer eficiente el uso de potencia. La configuración para la deshabilitación de los módulos no requeridos se puede realizar al configurar un registro destinado a dicho fin.
- *Interfaces de comunicación.* La comunicación mediante UART, SPI e I2C puede realizarse al configurar los puertos del microcontrolador apropiados para tal fin. El microcontrolador cuenta con 4 unidades USART que pueden programarse para establecer una comunicación asíncrona con dispositivos externos que utilicen comunicación mediante UART. Por otro lado, el microcontrolador emplea la designación TWI que es totalmente compatible

con dispositivos externos que utilicen el protocolo I2C para la transferencia de datos, de la misma manera, se cuenta con las terminales programables para la comunicación mediante SPI.

Para el caso del RAS-II se configuraron 27 terminales de entrada/salida: 8 para el bus de datos del LCD, 5 para el bus de control del LCD, una para la activación del led de iluminación del LCD, una para el manejo de un led denominado "SYNC" (para indicar cuando el receptor GPS se ha sincronizado), una para el manejo de un relevador denominado "relay" (que activa la visualización del voltaje de la batería en una *display* de 7 segmentos triple), una para la lectura del estado del led del receptor GPS denominado "directive", 5 para la activación del circuito integrado correspondiente a cada uno de los tres ADCs, la memoria de pre-evento y la memoria principal y, por último, se utilizaron 5 terminales para el manejo de un joystick.

Se programaron dos módulos USART, el primero para la comunicación externa con dispositivos mediante un conector DB9 y el segundo, para la comunicación con el receptor GPS. Se configuraron las terminales necesarias para la comunicación mediante los protocolos SPI e I2C; se utilizó una terminal para la lectura de la señal de *reset* y, finalmente, se conectó un oscilador de cuarzo de 16 [MHz] para la operación del microcontrolador, estableciendo previamente el fusible para la selección del oscilador externo como fuente de reloj. La figura 3.20 muestra de manera esquemática la conexión entre el microcontrolador y los distintos periféricos utilizados.

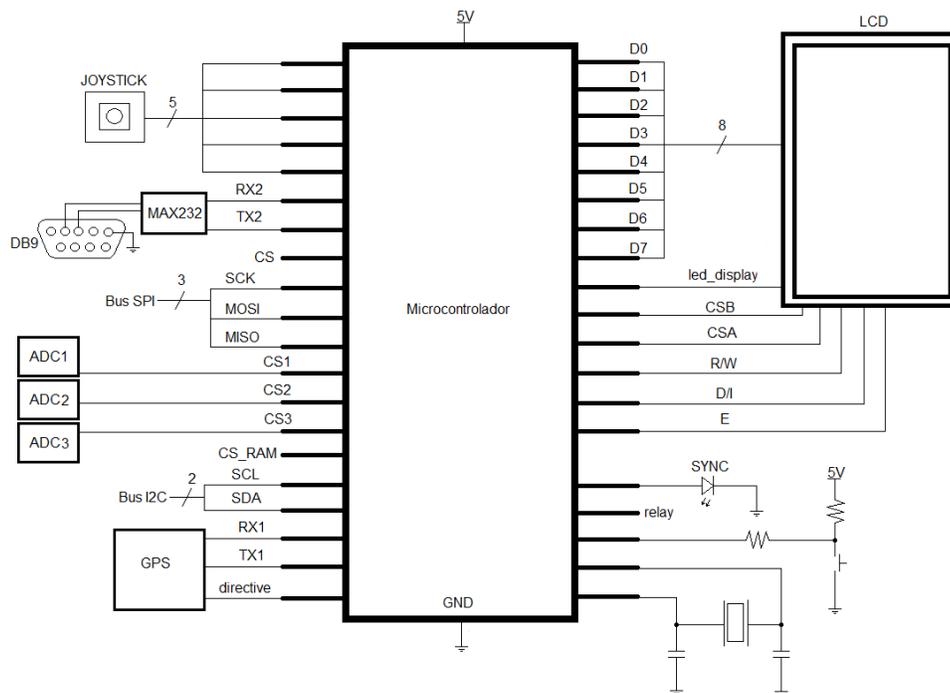


Figura 3.20. Conexión entre microcontrolador y periféricos.

### 3.2.4. Memoria principal

Las señales de los tres canales del acelerómetro son acondicionadas, digitalizadas y posteriormente procesadas, en ésta última etapa el valor de las señales se almacena en variables que el microcontrolador puede manipular y registrar en un medio de almacenamiento de datos como la memoria microSD.

La memoria microSD es un formato de tarjeta de memoria basada en una memoria flash. La memoria microSD es uno de los tamaños del estándar de tarjetas SD, utilizadas popularmente en equipos electrónicos portátiles. La portabilidad de este tipo de tarjeta de memoria es una de las dos razones para que fuera utilizada en el sistema RAS-II como memoria principal para el registro de datos, la otra razón está asociada a la gran capacidad para el almacenamiento de datos.

Los tres macro componentes de la memoria microSD son: un núcleo de memoria flash en donde se almacena toda la información, un controlador para la gestión de operaciones entre un dispositivo maestro y la tarjeta de memoria y, una interfaz de 8 terminales para el intercambio de información. La asignación de las terminales está en función del modo de comunicación utilizado (modo SD o SPI); para el sistema desarrollado se utilizó el modo SPI por razones expuestas en el apartado 2.5.3, de esta manera, la figura 3.21 y la tabla 3.4 muestran la distribución de terminales de la memoria microSD en este modo de operación.

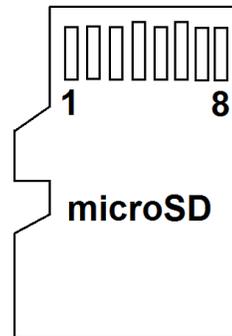


Figura 3.21. Memoria microSD.

Terminal	Nombre	Función
1	X	Reservado
2	CS	Selección del circuito
3	DI	Datos de entrada
4	VDD	Alimentación
5	CLK	Señal de reloj
6	VSS	Tierra
7	DO	Datos de salida
8	X	Reservado

Tabla 3.4. Terminales de la memoria microSD.

Con respecto a los parámetros de consumo de la memoria microSD, ésta requiere 100 [mA] de corriente de alimentación y opera con un nivel de voltaje en el rango de 2.6 a 3.7 [V], por lo tanto, las señales SCK, MOSI y CS generadas por el microcontrolador, que se encuentran en un nivel de voltaje de 5 [V], no pueden ser transferidas directamente a la microSD. Se utilizó un regulador de voltaje de 3.3 [V] y un corredor de nivel cuya corriente de consumo es 50 [mA] para lograr la interfaz entre las señales generadas por el microcontrolador y aquellas que son transferidas directamente a la memoria. Como se puede notar, la cuarta señal del Bus SPI (MISO) no necesita ningún tipo de tratamiento pues el nivel de voltaje generado por la memoria es suficiente para que el microcontrolador lo reconozca como un uno lógico. El diagrama de conexiones entre microcontrolador, corredor de nivel y tarjeta de memoria se muestra en la figura 3.22.

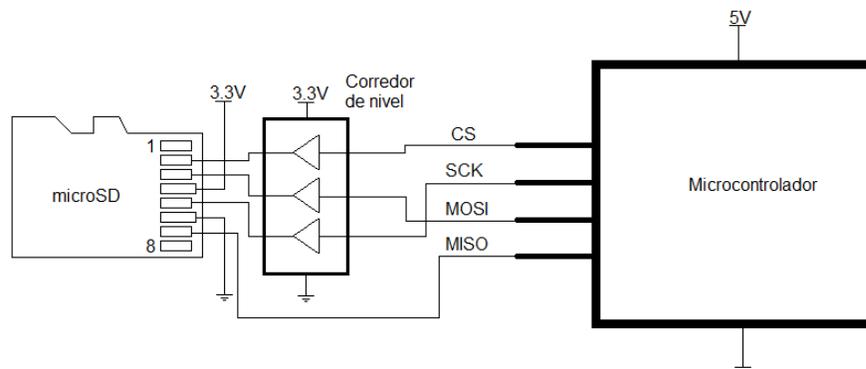


Figura 3.22. Interfaz entre microcontrolador y memoria.

### 3.2.5. Memoria de pre-evento

Las señales provenientes del acelerómetro, después de ser digitalizadas, son almacenadas en la tarjeta de memoria microSD en el instante en que ocurre un fenómeno sísmico, sin embargo, por especificaciones de diseño es necesario contar con el historial de las señales 50 [s] antes de la ocurrencia de dicho fenómeno sísmico. Como punto de partida, la información previa a la perturbación sísmica pudo ser almacenada de manera temporal en la memoria SRAM del microcontrolador para un historial previo de 10 [s], tiempo utilizado con fines prácticos para la implementación general del programa informático. Después de corroborar que el programa cumplía con sus funciones, el tiempo fue ampliado para cubrir los 50 [s] de pre-evento solicitados; en ese momento, el microcontrolador presentó un comportamiento anormal reiniciándose de manera inesperada. La causa de dicho problema fue pretender manejar 90 [kB] de información en una memoria con una capacidad de 8 [kB]. Como medida para solucionar este problema se decidió utilizar una memoria RAM externa que no tuviera las mismas limitaciones para el almacenamiento del pre-evento.

La elección de la memoria RAM se realizó con base en ciertas características de las cuales destacan dos: la primer característica importante, involucra la capacidad de

almacenamiento de la memoria RAM, de manera tal que ésta tuviera la capacidad suficiente para almacenar los 50 [s] de pre-evento. Por otro lado, se tomó en cuenta que las operaciones de escritura y lectura a la memoria RAM fueran lo suficientemente rápidas como para no interferir en la frecuencia de muestreo de 200 [Hz], en otras palabras, los procesos de lectura y escritura debían realizarse en tiempos inferiores a 5 [ms]. De entre las memorias disponibles en el mercado, se eligió una por las razones antes expuestas, esta memoria es fabricada por una empresa estadounidense y algunas de sus características técnicas son las siguientes:

*Rangos máximos absolutos*

- Alimentación ( $V_{cc}$ ): 5.5 [V]
- Voltajes de entrada y salida: -0.3 [V] a ( $V_{cc}+0.3$ ) [V]

*Características de consumo*

- Corriente de operación: máxima = 10 [mA]

*Características eléctricas*

- Voltaje de alimentación ( $V_{cc}$ ): mínimo: 2.5 [V]; máximo: 5.5 [V]
- Voltaje de retención de datos mínimo: 1 [V]

*Características de tiempo*

- Frecuencia de operación: máxima 20 [MHz]

*Interfaz de comunicación*

- SPI

*Características de temperatura*

- Temperatura de almacenamiento: -65 [°C] a 150 [°C]
- Temperatura de operación: -40 [°C] a 85 [°C]

*Otras características*

- Tipo de empaquetado: SOIC (Small Outline Integrated Circuit)
- Tecnología de fabricación: CMOS

La memoria RAM está organizada en 4096 páginas de 32 bytes cada una, contando con 131072 localidades de memoria, a su vez, cada localidad de memoria puede almacenar 8 bits de información. La transferencia de datos se realiza conforme al protocolo SPI, contando para ello con las terminales: SCK para la señal de reloj, SO para la salida de datos de la memoria hacia el microcontrolador, SI para la entrada de datos desde el microcontrolador y CS para la activación del circuito. La figura 3.23, tomada del manual de usuario de la memoria RAM, muestra la distribución de terminales del circuito.

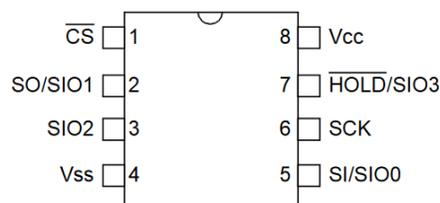


Figura 3.23. Distribución de terminales de la memoria RAM.

La memoria RAM tiene tres interfaces de comunicación, la primera corresponde a la interfaz SPI estándar, utilizando para ello las terminales SO, SI, SCK y  $\overline{CS}$ . La segunda interfaz de comunicación utiliza las terminales SIO0, SIO1, SCK y  $\overline{CS}$ ; esta interfaz recibe el nombre de *Interfaz Serial Doble* o *SDI (Serial Dual Interface)*. La última interfaz denominada como *Interfaz Serial Cuádruple* o *SDQ (Serial Quad Interface)* utiliza las terminales SIO0, SIO1, SIO2, SIO3, SCK y  $\overline{CS}$ . La diferencia entre cada tipo de interfaz es la velocidad durante la transferencia de datos; la interfaz SPI únicamente permite el intercambio de un bit por cada ciclo de reloj, a su vez, las interfaces SDI y SQI permiten el intercambio de 2 y cuatro bits por ciclo de reloj, respectivamente. Para el caso del RAS-II se utilizó la interfaz SPI estándar puesto que SDI y SQI se utilizan para aplicaciones que requieren una velocidad muy alta para la transferencia de datos. La terminal  $\overline{HOLD}$  sólo está disponible para las interfaces SPI y SDI y es utilizada para pausar un proceso de transferencia de datos sin tener que reenviar la instrucción inicial que se explica a continuación.

### Conjunto de instrucciones

El primer paso para comenzar a utilizar la memoria RAM es la activación de la señal  $\overline{CS}$ , a continuación el microcontrolador debe generar el número de instrucción de acuerdo al tipo de operación que se requiera. La memoria RAM reconoce 7 instrucciones diferentes resumidas en la tabla 3.5.

Nombre de la instrucción	Código binario	Código hexadecimal	Descripción
READ	0000 0011	0x03	Lectura de datos en la dirección de memoria especificada
WRITE	0000 0010	0x02	Escritura de datos en la dirección de memoria especificada
EDIO	0011 1011	0x3B	Acceso al modo SDI
EQIO	0011 1000	0x38	Acceso al modo SQI
RSTIO	1111 1111	0xFF	Salir de los modos SDI y SQI
RDMR	0000 0101	0x05	Lectura del registro de modo de operación
WRMR	0000 0001	0x01	Escritura del registro de modo de operación

Tabla 3.5. Conjunto de instrucciones de la memoria RAM.

### Registro de modo de operación

La memoria RAM tiene distintos modos de operación para las secuencias de lectura y escritura de datos; el modo de operación puede especificarse al configurar el registro de modo de operación que se muestra en la figura 3.24.

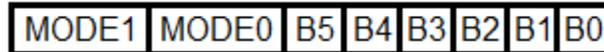


Figura 3.24. Registro de modo de operación.

Los bits B0 a B5 están reservados y no se puede leer ni escribir sobre ellos; por otro lado, los bits MODE1 y MODE0 son de lectura/escritura y pueden ser modificados en el momento que sea necesario. La combinación de los bits MODE1 y MODE0 establece el modo de operación de acuerdo a la tabla 3.6.

MODE1	MODE0	Modo de operación
0	0	Modo byte
1	0	Modo página
0	1	Modo secuencial
1	1	Reservado

Tabla 3.6. Configuración del modo de operación.

### Modos de operación

Los modos de operación en los que puede trabajar la memoria RAM son los siguientes:

- *Modo byte.* Es seleccionado al configurar los bits 7 y 6 del registro de modo de operación con el valor 0b00\*\*\*\*\* (los asteriscos indican que los bits se encuentran reservados). En este modo, las operaciones de lectura y escritura se limitan a un byte. Una vez que se selecciona el circuito integrado y se envía el código para la instrucción de lectura o escritura, se debe enviar la dirección de memoria en la cual se quiere leer o escribir un byte, esta dirección de memoria tiene una longitud de 24 bits. Después de haber enviado la dirección de memoria, en los siguientes 8 ciclos de reloj el byte es enviado por la memoria RAM (en la operación de lectura) o el byte debe ser enviado desde el microcontrolador hacia la memoria RAM (en la operación de escritura). Una vez que el byte es transferido, el proceso culmina con la deshabilitación del circuito integrado.
- *Modo página.* Es seleccionado al configurar los bits 7 y 6 del registro de modo de operación con el valor 0b10\*\*\*\*\*. En este modo, las operaciones de lectura y escritura se limitan a una página. Una vez que se selecciona el circuito integrado y se envía el código para la instrucción de lectura o escritura, se debe enviar la dirección de memoria en la cual se quiere leer o escribir un byte. Después de haber enviado la dirección de memoria, en los siguientes 8 ciclos de reloj el byte es enviado por la memoria RAM (en la operación de lectura) o el byte debe ser enviado desde el microcontrolador hacia la memoria RAM (en la operación de escritura). Después del primer byte, se pueden seguir transfiriendo bytes de información si se continúa

proporcionando la señal de reloj; el usuario ya no necesita generar la dirección de memoria puesto que la memoria RAM cuenta con un contador interno que realiza esta tarea de manera autónoma. La lectura y escritura de datos continúa de manera indefinida mientras se proporcione una señal de reloj; en el momento en que se alcance la última dirección de memoria dentro de la página, el contador interno se reiniciará a la primer localidad de dicha página y el proceso continuará. El término del proceso tiene lugar cuando se deshabilita la selección del circuito integrado.

- *Modo secuencial.* Es seleccionado al configurar los bits 7 y 6 del registro de modo de operación con el valor 0b01\*\*\*\*. Este modo de operación es igual al modo página, con la excepción de los límites de lectura/escritura; los límites están fijados por el número total de localidades de memoria de la RAM (131072).

El modo de operación por default de la memoria RAM es el modo secuencial y fue el utilizado en el sistema RAS-II, de esta manera, proporcionando de manera continua una señal de reloj, no existen límites para la escritura y lectura de datos, tomando en cuenta que para el almacenamiento de 50 [s] de pre-evento no se utilizan todas las localidades de memoria disponibles.

El diagrama de conexiones entre la memoria RAM y el microcontrolador se muestra en la figura 3.25.

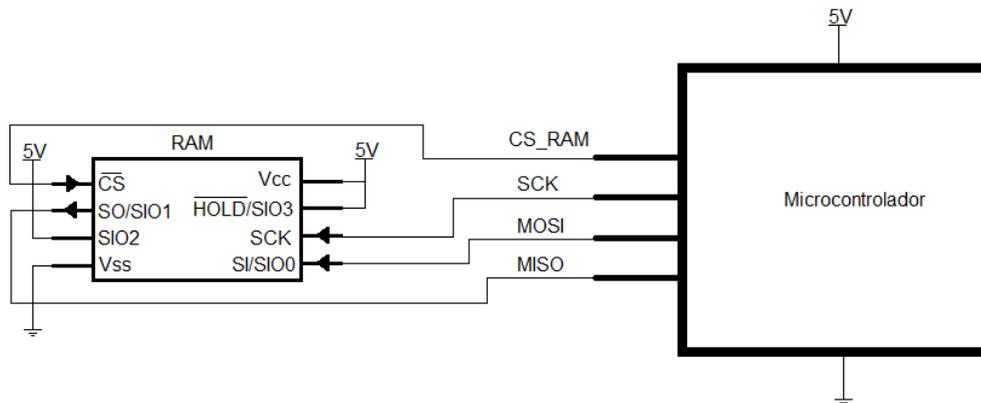


Figura 3.25. Conexión entre memoria RAM y microcontrolador.

### 3.2.6. Receptor GPS

Una de las características que todos los registros sobre fenómenos sísmicos poseen, es la referencia de tiempo en que ocurrió cada sismo. La hora utilizada por estandarización es la hora UTC, de esta manera, es necesario contar con un elemento capaz de proporcionar la hora UTC en todo momento y que además ésta sea confiable. Como se explicó en el apartado 2.6, los receptores GPS obtienen datos de ubicación y hora desde la constelación de satélites NAVSTAR cuya precisión en el manejo de datos es muy alta. En el mercado existe una gran variedad

de receptores GPS, unos con mejor rendimiento que otros, sin embargo, esta ventaja se ve reflejada en el costo de los mismos. Se decidió elegir un receptor GPS cuyo equilibrio entre costo y prestaciones permitieran su uso en el sistema RAS-II, además, se consideraron también sus dimensiones físicas.

Algunas de las características técnicas del receptor GPS utilizado son las siguientes:

#### *Alimentación*

- *Voltaje de alimentación: 4.5 a 6.5 [V]*
- *Consumo de corriente: 70 [mA]*

#### *Características de exactitud*

- *Posición: 10 [m]*
- *Velocidad: 0.1 [m/s]*
- *Tiempo: 1 [ $\mu$ s]*

#### *Características de tiempo de adquisición*

- *Readquisición de datos: 0.1 [s]*
- *Arranque en frío: 35 [s]*
- *Arranque en caliente: 1 [s]*

#### *Condiciones dinámicas*

- *Altitud máxima: 18 [km]*
- *Velocidad máxima: 515 [m/s] (1000 nudos náuticos)*
- *Aceleración máxima: 4g*

#### *Comunicación*

- *Niveles eléctricos: TTL*
- *Tasa de transmisión de datos: 4800 bps (bits por segundo)*
- *Mensajes de salida: NMEA 0183, GGA, GSA, GSV, RMC, VTG, GLL*

Es importante mencionar que para este tipo de receptores basta con suministrar el voltaje de alimentación para poder obtener las cadenas de datos del protocolo NMEA 0183; sin embargo, para poder visualizar los datos provenientes del GPS, como son: fecha, hora, latitud y longitud, es necesario que el microcontrolador realice el procesamiento de la cadena ingresante y obtenga únicamente aquella información que es útil para este proyecto. La comunicación entre el microcontrolador y el receptor GPS se realiza con base en la comunicación serial mediante un módulo UART del mismo microcontrolador. La tasa de transferencia de datos utilizada es de 4800 bits por segundo, ya que si no se configura a esta velocidad se estarán registrando datos totalmente ajenos a los establecidos por el protocolo NMEA 0183.

La distribución de terminales del receptor GPS se muestra en la figura 3.26, tomada del manual de usuario de dicho dispositivo.

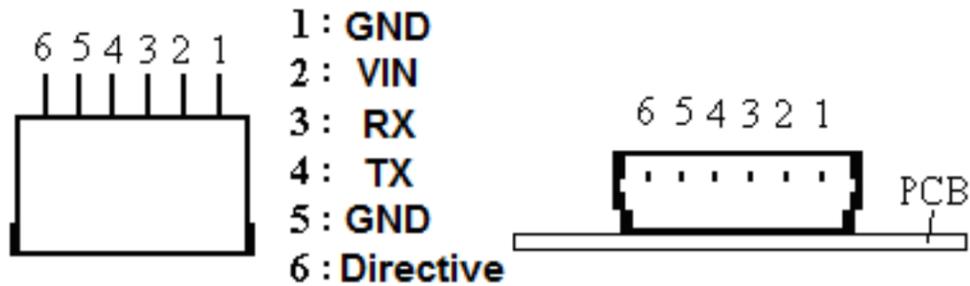


Figura 3.26. Distribución de terminales del receptor GPS.

De acuerdo con la figura anterior, las terminales 1 y 5 corresponden con la conexión a tierra mientras que la terminal 2 corresponde con el voltaje de alimentación. Las terminales RX y TX se utilizan para la transferencia de datos entre el receptor GPS y el microcontrolador y, la terminal 6 (*Directive*) es utilizada para conocer el estado del led del receptor GPS; al alimentar al receptor, el led permanecerá encendido, sin embargo, toma un tiempo para que éste se sincronice con los satélites, una vez que se ha establecido comunicación con los satélites, el led permanecerá en un estado intermitente entre encendido y apagado, en este momento ya se puede obtener la cadena de datos del protocolo NMEA 0183.

La conexión entre el receptor GPS y el microcontrolador se muestra esquemáticamente en la figura 3.27.

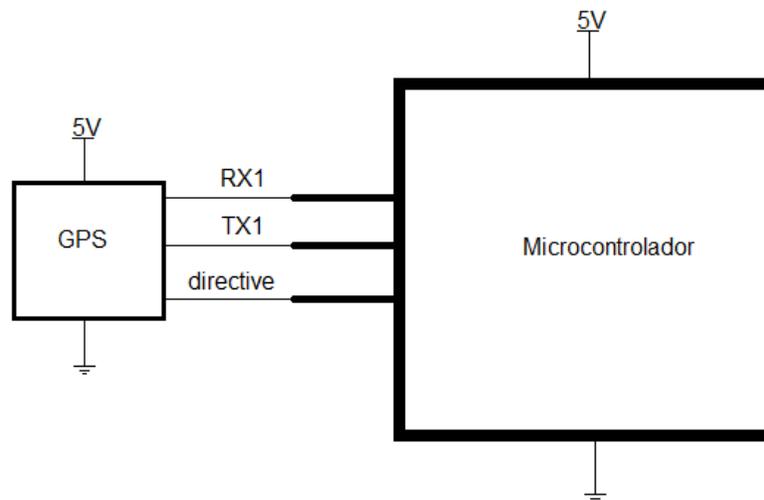


Figura 3.27. Conexión entre receptor GPS y microcontrolador.

### 3.2.7. Reloj de Tiempo Real

A pesar de que los receptores GPS cuentan con una alta precisión y confiabilidad para proporcionar una referencia de tiempo, la obtención de ésta depende en gran medida de la ubicación del receptor GPS. El receptor GPS debe tener como mínimo 4 satélites a la vista para tener una buena sincronización; además, en lugares con un elevado número de obstáculos como árboles y edificios, la sincronización del

receptor GPS con los satélites será más difícil. Ante esta problemática surge la utilización de un elemento capaz de actuar aun cuando el enlace con los satélites no sea posible, este elemento es el reloj de tiempo real; con su uso, se garantiza que el sistema para el monitoreo de fenómenos sísmicos contará con una referencia de tiempo en todo momento.

El reloj de tiempo real utilizado en el sistema RAS-II es un reloj/calendario cuyos datos se almacenan en una memoria SRAM de 64 bytes; la memoria SRAM está respaldada por una batería de tipo botón, de modo que la información se conserva aún ante la ausencia de una fuente de polarización. El reloj/calendario proporciona información de segundos, minutos, horas, nombre del día de la semana, día, mes y año. El reloj/calendario actualiza de manera automática sus datos y proporciona la fecha y hora correctas, incluyendo el ajuste para los años bisiestos.

El RTC opera como dispositivo esclavo conectado al bus I2C, de esta manera, el RTC puede comenzar a trabajar al generar la condición de inicio genérica para el protocolo I2C, en seguida se debe enviar la dirección de 7 bits del RTC, esta dirección es: 0b1101 0000. El ingreso a los registros del dispositivo puede realizarse en todo momento hasta que se genere una condición de parada en el bus. Cuando el voltaje de polarización  $V_{CC}$  cae por debajo de  $1.25V_{BAT}$ , donde  $V_{BAT}$  es el voltaje de la batería de tipo botón, el RTC suspende cualquier acceso en progreso y ninguna entrada de datos es aceptada mientras dicha condición se mantenga, esto tiene la finalidad de evitar la escritura de datos erróneos a los registros del RTC. Por otro lado, cuando el voltaje de polarización cae por debajo de  $V_{BAT}$ , el dispositivo entra en un estado de bajo consumo, operando únicamente con la batería de tipo botón (la actualización de los datos se mantiene y el reloj/calendario continúa operando). Una vez que el voltaje de polarización se restablece a un nivel superior a  $V_{BAT} + 0.2$  [V], el RTC regresa a su operación normal y los accesos a sus registros son reconocidos. Algunas de las características técnicas del RTC son las siguientes:

#### *Rangos máximos absolutos*

- *Voltaje aplicado a cualquier terminal: -0.5 [V] a 7 [V]*

#### *Características de alimentación*

- *Voltaje de alimentación ( $V_{CC}$ ): mínimo = 4.5 [V]; máximo = 5.5 [V]*
- *Voltaje de batería ( $V_{BAT}$ ): mínimo = 2.0 [V]; máximo = 3.5 [V]*

#### *Características de consumo*

- *Corriente de consumo en estado activo: 1.5 [mA]*
- *Corriente en estado de bajo consumo: 200 [ $\mu$ A]*

#### *Características de tiempo*

- *Frecuencia de operación: máxima = 100 [kHz]*

#### *Características de temperatura*

- *Temperatura de operación: 0 a 70 [°C] (rango comercial)*
- *Temperatura de almacenamiento: -55 [°C] a 125 [°C]*

- *Temperatura de soldadura: 260 [°C] por 10 [s]*

La distribución de terminales del dispositivo se muestra en la figura 3.28, tomada del manual de usuario del reloj de tiempo real.

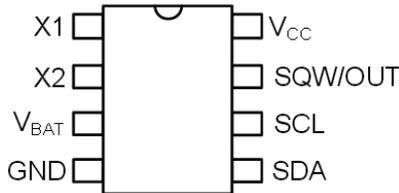


Figura 3.28. Distribución de terminales del RTC.

La función de cada una de las terminales es la siguiente:

*X1, X2 (terminales 1 y 2).* Corresponden a la conexión de un cristal de cuarzo de frecuencia de operación igual a 32.768 [kHz].

*V<sub>BAT</sub> (terminal 3).* Corresponde a la terminal para conectar la batería de respaldo. Si la aplicación no requiere una batería de respaldo, la terminal *V<sub>BAT</sub>* debe conectarse a tierra para prevenir un funcionamiento erróneo. El tiempo de operación del RTC al utilizar la batería de respaldo varía, sin embargo, el fabricante garantiza la operación por hasta 10 años al utilizar una batería de 48 [mAh].

*GND (terminal 4).* Corresponde con la conexión a tierra.

*SDA, Serial Data Input/Output (terminal 5).* Corresponde a la terminal compatible con la señal SDA del bus I2C. Esta terminal es del tipo *drenaje abierto* y requiere de la conexión de una resistencia de *pull-up* externa.

*SCL, Serial Clock Input (terminal 6).* Corresponde a la terminal compatible con la señal SCL del bus I2C. Su uso permite la sincronización durante la transferencia de datos entre dispositivos conectados al bus. Esta terminal es del tipo *drenaje abierto* y requiere de la conexión de una resistencia de *pull-up* externa.

*SQW/OUT, Square Wave/Output (terminal 7).* Cuando se encuentra habilitada, corresponde a la terminal para el manejo de una onda cuadrada con una de cuatro frecuencias seleccionables: 1 [Hz], 4 [kHz], 8 [kHz] y 32 [kHz]. La terminal es del tipo *drenaje abierto* y requiere de la conexión de una resistencia de *pull-up* externa.

*V<sub>CC</sub> (terminal 8).* Corresponde a la terminal para la conexión de la fuente de polarización.

### *Mapa de memoria*

El RTC cuenta con una memoria SRAM de 64 bytes; 8 localidades de memoria almacenan la información de fecha y hora (localidades de memoria 0x00 a 0x07) y

el resto (localidades de memoria 0x08 a 0x3F) se encuentra libre para el uso opcional por parte del usuario. La figura 3.29, tomada del documento antes señalado, muestra el mapa de memoria del reloj de tiempo real.

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	FUNCTION	RANGE
00h	CH	10 Seconds			Seconds				Seconds	00–59
01h	0	10 Minutes			Minutes				Minutes	00–59
02h	0	12	PM/ AM	10 Hour	Hours			Hours	1–12 +AM/PM 00–23	
		24	10 Hour							
03h	0	0	0	0	DAY			Day	01–07	
04h	0	0	10 Date		Date			Date	01–31	
05h	0	0	0	10 Month	Month			Month	01–12	
06h	10 Year				Year			Year	00–99	
07h	OUT	0	0	SQWE	0	0	RS1	RS0	Control	—
08h–3Fh									RAM 56 x 8	00h–FFh

Figura 3.29. Mapa de memoria del RTC.

El mapa de memoria puede ser leído o escrito por parte del microcontrolador en cualquier momento, después de realizar el reconocimiento del RTC a través del bus I2C. Las localidades de memoria 0x00 a 0x07 contienen la información del reloj/calendario, almacenada en formato BCD (*Binary-Coded Decimal*).

En la figura 3.29 se puede apreciar el bit *CH* (*Clock Halt*) que corresponde a la detención del oscilador. Cuando el bit es puesto a uno, se deshabilita el oscilador y la actualización de los datos se detiene, proporcionando un bajo consumo de corriente, por otro lado, cuando el bit es puesto a cero, el oscilador es habilitado y el RTC tiene un funcionamiento normal. En el registro 0x00, los bits 0 a 3 corresponden a las unidades mientras que los bits 4 a 6 corresponden a las decenas, lo mismo ocurre para el caso de los minutos.

El RTC puede operar tanto en un formato de 12 horas como en uno de 24 horas. Para operar en el modo de 12 horas el bit 6 del registro 0x02 debe ser puesto a uno, mientras que si se desea que el RTC opere en el modo de 24 horas, este bit debe ser puesto a cero. En el formato de 12 horas, el bit 5 del registro 0x02 corresponde a PM cuando se encuentra en alto. En este mismo formato de 12 horas, el bit 4 corresponde a las decenas de la hora. En el modo de 24 horas los bits 4 y 5 corresponden a las decenas, mientras que los bits 0 a 3 corresponden a las unidades.

El registro 0x03 corresponde al día de la semana, el código queda a consideración del usuario y como ejemplo, el número 1 puede corresponder al día domingo, el número 2 al día lunes y así sucesivamente, siempre y cuando se mantenga un orden consecutivo.

Los registros 0x04, 0x05 y 0x06 corresponden al día, mes y año, teniendo un rango de 1 a 31 días, 1 a 12 meses y 0 a 99 años, respectivamente. El último registro

(0x07) corresponde a la configuración de la señal SQW/OUT. Para habilitar la señal se debe colocar un uno lógico en el bit 4 de este registro; una vez seleccionado se pueden elegir cuatro frecuencias para la señal al configurar los bits 1 y 0: 1 [Hz] para el valor 0b00, 4 [kHz] para el valor 0b01, 8 [kHz] para el valor 0b10 y 32 [kHz] para el valor 0b11.

Por último, vale la pena mencionar que al alimentar el circuito integrado por primera vez, la fecha y hora quedan configuradas como: 01/01/00 01 00:00:00 con el formato: DD/MM/AA dd hh:mm:ss, donde DD corresponde al día, MM al mes, AA al año, dd al día de la semana, hh a la hora, mm al minuto y ss al segundo. La fecha y hora pueden ser modificados por el usuario al escribir en los registros apropiados la información que se ajuste a la fecha y hora actuales.

La conexión del RTC y el microcontrolador en el sistema RAS-II se muestra en la figura 3.30.

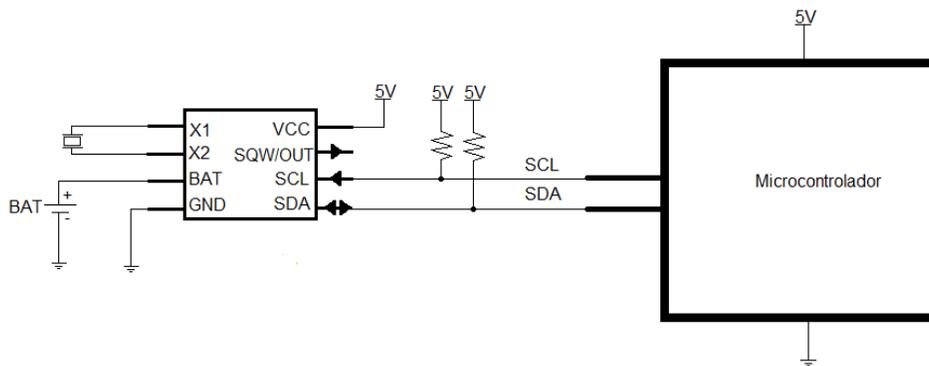


Figura 3.30. Conexión entre RTC y microcontrolador.

### 3.2.8. Joystick y display gráfico

La mayoría de los sistemas electrónicos reciben datos de entrada y proporcionan datos de salida por parte del usuario. Los datos de entrada son utilizados para configurar algunos parámetros que serán utilizados durante la operación del sistema electrónico, por otro lado, durante su operación el sistema proporciona información útil al usuario, representada de distintas maneras: gráficamente, sonoramente, luminosamente, etc. Para el caso del sistema RAS-II los datos de entrada son proporcionados con ayuda de un *joystick*, para indicarle al sistema, entre otras cosas, el umbral de disparo o la fecha y hora con que debe cargarse el RTC. A su vez, el RAS-II proporciona información al usuario mediante un *display* de cristal líquido, de esta manera, el usuario puede, en primer lugar, configurar el equipo y en segundo lugar, obtener información como la hora recibida del GPS, conocer el nivel de voltaje en los canales, observar el estado de la batería o el espacio disponible de la memoria microSD.

## Joystick

Un *joystick* es un dispositivo de entrada digital usado para el control de dos o más posiciones conectadas a un nivel lógico determinado. En su composición más elemental está formado por un conjunto de interruptores (botones o teclas) en cruceta, situada en la base, más un botón de acción. Estos botones son de contacto mecánico que generan un estado lógico ('1' o '0') al cerrar o abrir dicho interruptor. En la figura 3.31 se muestra el joystick utilizado en el sistema RAS-II

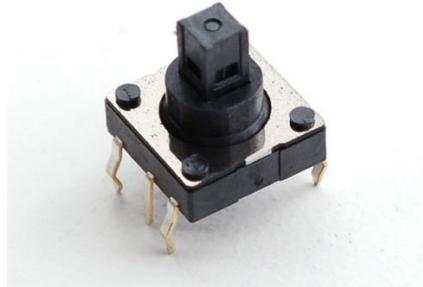


Figura 3.31. Joystick.

La forma de onda que entrega un interruptor en forma ideal es la mostrada en la figura 3.32.

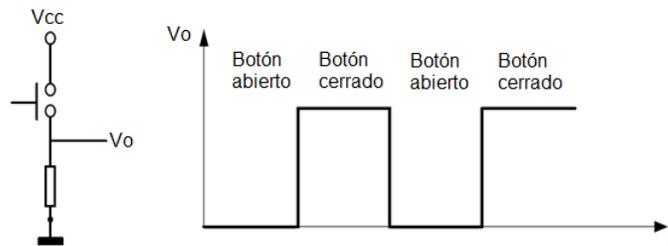


Figura 3.32. Comportamiento ideal de un interruptor.

Sin embargo, debido a las características mecánicas de los elementos que constituyen a un botón, se presenta una vibración (efecto rebote) en la señal cuando se cierra o abre dicho botón. Este efecto se muestra en la figura 3.33.

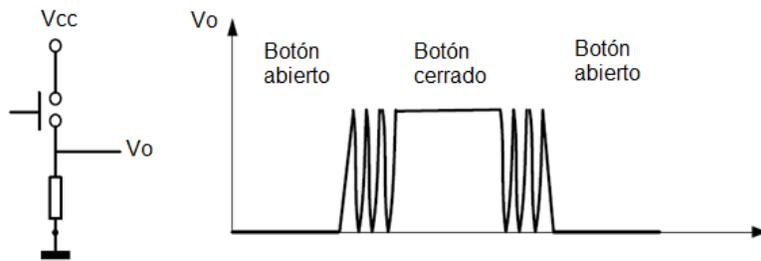


Figura 3.33. Efecto de rebote.

Este efecto rebote es indeseado, ya que puede causar problemas en la lectura del estado lógico del botón. El microcontrolador puede interpretar este efecto como si se pulsara varias veces el interruptor, provocando errores en las secuencias o rutinas de programación cuando se estén leyendo los estados o niveles lógicos de estos botones.

Existen varias formas de eliminar este efecto, popularmente se conoce como *debouncer* (sin rebote) al elemento que elimina el efecto de rebote mecánico. Una forma de atacar el problema es haciendo uso de componentes adicionales conectados al botón, entre los cuales sobresalen, capacitores (filtros) o compuertas lógicas. Una alternativa más es arreglar el problema mediante software, agregando retardos en las rutinas encargadas de leer el estado lógico del botón, esto permite esperar a que se establezca la señal proporcionada por el botón al ser presionado y una vez estable, el estado lógico puede leerse sin que se presente ningún problema. Esta solución se observa en la figura 3.34

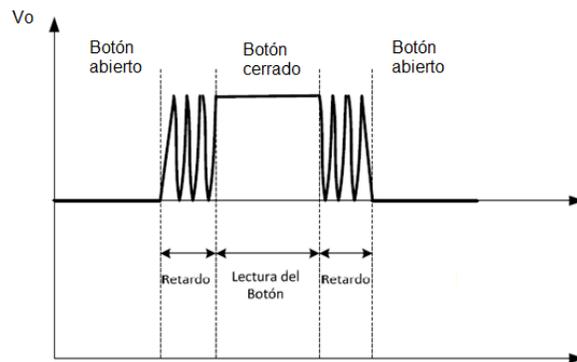


Figura 3.34. *Deboucer* por software.

El *joystick* utilizado en el sistema RAS-II cuenta con un común y cinco posiciones: arriba, abajo, izquierda, derecha y al centro. A cada una de las 5 terminales del *joystick* se conectó una resistencia de *pull-down*, de esta manera, al presionar cualquier tecla, el circuito se cierra con la terminal común, circulando la corriente por la resistencia y generando un uno lógico en la posición presionada. La conexión entre el *joystick* y el microcontrolador se muestra en la figura 3.35.

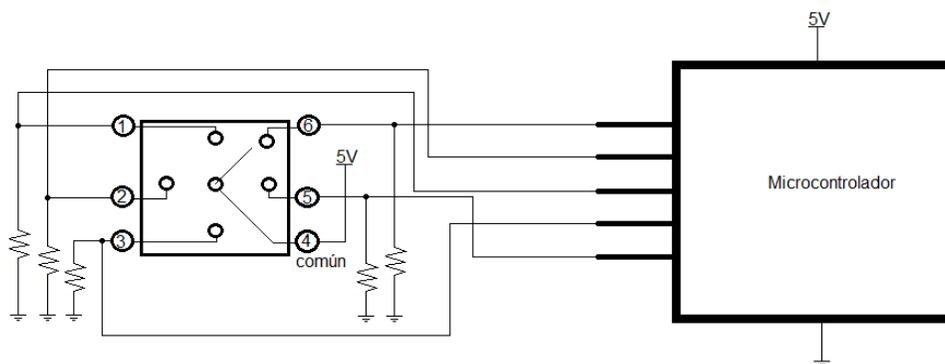


Figura 3.35. Conexión entre *joystick* y microcontrolador.

## Display gráfico

En general, las pantallas de cristal líquido (LCD, *Liquid Crystal Display*) son visualizadores pasivos, con un bajo consumo de energía. Están formadas por un cristal líquido entre dos placas paralelas de vidrio a una cierta separación. Estas placas de vidrio tienen unos electrodos especiales, que definen con su forma los caracteres o símbolos que se visualizarán. La superficie del vidrio que hace contacto con el líquido es tratada de manera que induzca la alineación de los cristales en dirección paralela a las placas. Esta alineación permite el paso de la luz incidente sin ninguna alteración. Cuando se aplica la polarización adecuada entre los electrodos, se forma un campo eléctrico perpendicular entre ellos, el cual causa que las moléculas del líquido se agrupen en sentido paralelo al campo eléctrico y aparezca una zona oscura sobre un fondo claro.

Para el caso de una pantalla grafica GLCD (*Graphic Liquid Crystal Display*), el funcionamiento es exactamente el mismo, con la diferencia que no se tiene un conjunto de caracteres o símbolos, en cambio, se cuenta con una matriz de pixeles que pueden dar forma a caracteres, símbolos, líneas y figuras definidas por el usuario, al encender o apagar un pixel.

El *display* gráfico utilizado en el sistema RAS-II consume 30 [mA] y consta de 128x64 puntos, además está basado en el controlador KS0108, el cual permite tener acceso a cada punto a través del control y datos enviados por el microcontrolador. En la figura 3.36 se muestra el diagrama de bloques del *display*.

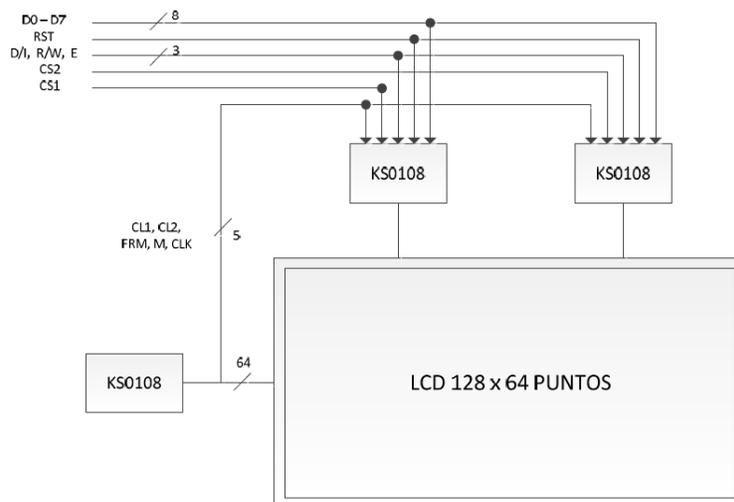


Figura 3.36. *Display*.

El *display* está dividido en dos secciones, cada una de ellas administrada por dos controladores. Cada sección está integrada por un bloque de 64x64 puntos, que en conjunto forman la matriz de 128x64 puntos. El *display* cuenta con 20 terminales y en la tabla 3.7 se muestra la función de cada una de ellas.

Terminal	Nombre	Función
1	CSA	Habilitación de la parte izquierda
2	CSB	Habilitación de la parte derecha
3	VSS	Tierra
4	VDD	Alimentación (5 [V])
5	Vo	Ajuste de contraste
6	D/I	Control para comandos/datos
7	R/W	Lectura/escritura de comandos/datos
8	E	Señal de habilitación del <i>display</i>
9 - 16	A0 – A7	Líneas de datos
17	RST	Señal de <i>reset</i>
18	VEE	Salida interna de 10 [V]
19	LED(+)	Ánodo del led interno del <i>display</i>
20	LED(-)	Cátodo del led interno del <i>display</i>

Tabla 3.7. Terminales del *display*.

La polarización del *display* se realiza a través de las terminales VDD y VSS. El *display* cuenta con un circuito interno que genera un voltaje VEE de 10 volts en la terminal 18. Este voltaje es usado para ajustar el contraste del *display*, con ayuda de un potenciómetro.

La terminal de control D/I permite indicarle al *display* si se le están enviando datos o instrucciones; si la terminal tiene un estado lógico bajo, indica envió de instrucciones, por el contrario, si la terminal tiene un estado lógico alto, indica que se le envían datos.

La terminal de control R/W indica al *display* si se presenta una operación de lectura o escritura de datos. Si en esta terminal se presenta un estado lógico bajo, se trata de una operación de escritura, por el contrario, si en esta terminal se presenta un estado lógico alto, se tiene una operación de lectura.

La figura 3.37 muestra la conexión entre *display* y microcontrolador.

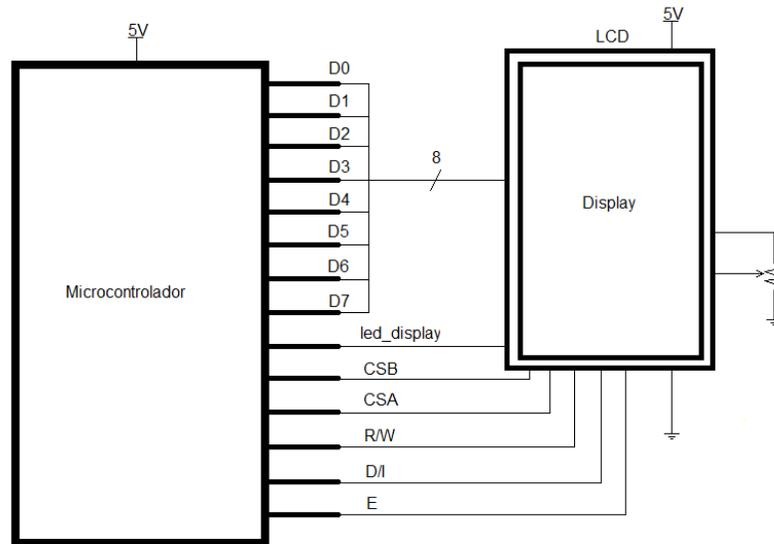


Figura 3.37. Conexión entre *display* y microcontrolador.

### 3.3. Software del sistema

En el apartado anterior se abordó el diseño del RAS-II desde el punto de vista de hardware, es decir, los componentes y módulos electrónicos tangibles cuya interconexión en un circuito electrónico permiten cumplir una tarea específica. Al ser el RAS-II un sistema autónomo, las tareas generales y particulares son gestionadas por un microcontrolador que funge como el cerebro del sistema.

Anteriormente se ha hablado de los tipos de microcontroladores, su funcionamiento y sus características principales, sin embargo, es hasta la programación del microcontrolador cuando realmente se está dotando de inteligencia al sistema electrónico; es en este punto donde se dan las instrucciones específicas al microcontrolador, de manera tal que este responda automática, ágil y eficientemente a las tareas que requieran ser solucionadas en un momento dado.

Para que un microcontrolador pueda gestionar tareas admitiendo y proporcionando señales eléctricas a sus dispositivos periféricos, éste debe ser programado con un lenguaje específico de alto nivel (un lenguaje estandarizado y entendible por todos los individuos que tienen conocimientos en esta área) en cuyo código se establecen las órdenes que regirán el funcionamiento del sistema. Existen un sin número de lenguajes y entornos de programación, sin embargo, todos tienen un funcionamiento similar.

Para la programación del microcontrolador del RAS-II se utilizó el entorno de desarrollo integrado de Arduino, que es una aplicación multiplataforma escrita en Java. Incluye un editor de código con características como el resaltado de sintaxis, llaves “{}” coincidentes, sangría automática y un mecanismo de un solo *click* para la compilación y la carga de programas. El entorno de Arduino soporta los lenguajes

de programación C y C++. La Figura 3.38 muestra el entorno de desarrollo de Arduino.



Figura 3.38. Entorno de desarrollo de Arduino.

### 3.3.1. Estructura general del programa

El Registrador de Aceleraciones Sísmicas del Instituto de Ingeniería, es un sistema electrónico diseñado con el fin de recabar información relacionada con aceleraciones sísmicas. El principal objetivo del RAS-II es, como ya se mencionó, almacenar información proveniente de acelerómetros; sin embargo, para que este propósito se cumpla, tanto en el hardware como en el software se resuelven tareas emergentes, sin las cuales el funcionamiento general del RAS-II sería imposible.

En cuanto a software se refiere, surgen tareas secundarias como la creación de una interfaz con el usuario, la obtención de la fecha y hora, la configuración de los parámetros del sistema, la obtención de información correspondiente al estado del equipo, el despliegue gráfico de las señales muestreadas, entre otras. En este apartado se dará un bosquejo general del programa informático utilizado para la operación del RAS-II.

Al ejecutar el programa informático con el cual ha sido programado el microcontrolador, éste se encarga de realizar las primeras tareas: definir variables, configurar los puertos utilizados, configurar las terminales, velocidad, fase y polaridad en la interfaz SPI, configurar las terminales y velocidad para la comunicación mediante UART y configurar el bus I2C. Una vez completadas estas tareas básicas, el microcontrolador se comunica con el LCD y despliega la primera información disponible; se indica si la memoria microSD ha sido inicializada correctamente, se muestran algunos de sus parámetros: tipo de sistema de archivos, versión, capacidad de almacenamiento total y capacidad de almacenamiento disponible, finalmente, se indica el voltaje de la batería por algunos segundos.

Después de mostrar la información inicial en el LCD, el microcontrolador ejecutará una rutina de repetición infinita, mostrando un menú principal. El usuario interactúa para obtener y dar información al sistema mediante el *joystick* y el LCD. En el momento en que el usuario mueve el *joystick* en cualquier dirección, el LCD muestra la elección correspondiente y el microcontrolador ejecuta la rutina asociada a esta elección. La manera en que se presenta la información en el RAS-II es mediante un menú principal y menús secundarios, cada menú se encarga de una tarea concreta. La estructura general del programa se representa en el diagrama de bloques mostrado en la figura 3.39.

En el menú principal se cuenta con tres menús secundarios: *Configuración*, *Estado del equipo* y *Registro de datos*.

*Configuración.* En el menú *Configuración* se presentan tres opciones. En la primer opción, *Sincronizar reloj*, se obtiene la fecha y la hora provenientes del GPS, el sistema comprobará que el enlace con los satélites se ha realizado, en caso contrario, emitirá un aviso y la referencia de tiempo del sistema no se habrá sincronizado con la del GPS. La segunda opción, *Ajustar reloj*, permite establecer al usuario la hora del reloj auxiliar (Reloj de Tiempo Real), de modo que el sistema cuente con una referencia de tiempo aun cuando el GPS no logre enlazarse con los satélites. En la configuración del RTC, si la fecha y hora establecidas por el usuario son incorrectas, el sistema indicará los errores y la configuración del RTC no se realizará. La tercer opción permite al usuario establecer el umbral de disparo de la señal de los acelerómetros, el umbral tiene un rango que va de 0 a 10 [Gal] con un incremento de 0.5 [Gal]-. Al finalizar la especificación del umbral de disparo, se pedirá al usuario especificar también si el acelerómetro utilizado tiene un rango de  $\pm 1g$  o  $\pm 2g$ .

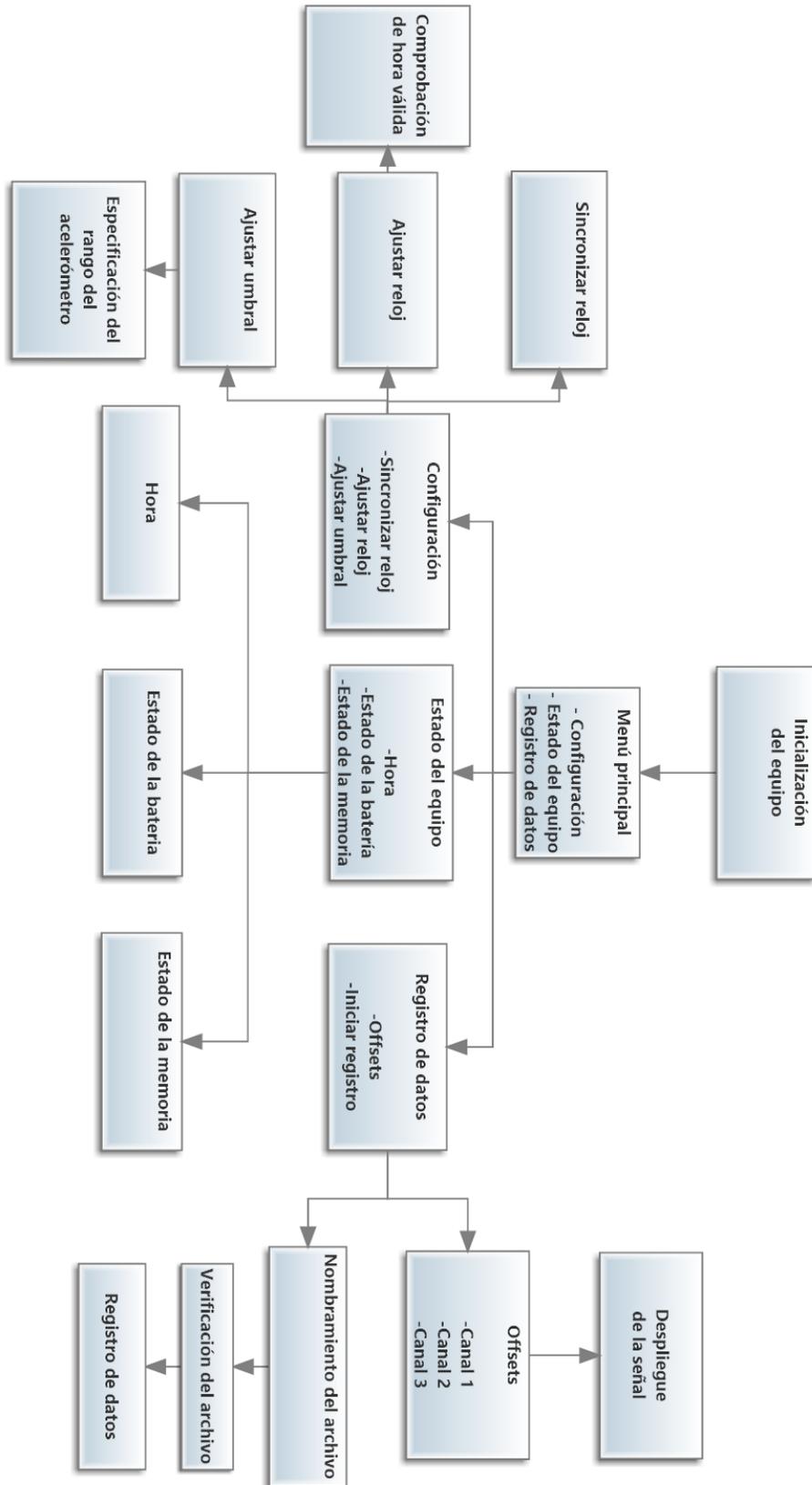


Figura 3.39. Estructura general del programa.

*Estado del equipo.* En este menú se proporciona al usuario información útil con el fin de conocer el estado general del equipo mediante tres opciones disponibles. La primer opción permite conocer la hora del equipo, se despliegan en el LCD la hora obtenida del GPS y la hora obtenida del RTC; la información se refresca cada segundo. La segunda opción permite activar o desactivar el multímetro integrado al equipo, con el fin de observar el voltaje de la batería que alimenta al RAS-II. La última opción ejecuta una rutina en la cual se establece comunicación con la memoria microSD; en el LCD se despliegan algunos de los parámetros, entre ellos: tipo de sistema de archivos, versión, capacidad total de la memoria y capacidad disponible.

*Registro de datos.* En este menú se ejecutan las tareas correspondientes a la digitalización de las señales provenientes de los tres canales. Se cuentan con dos opciones, dentro de la primer opción, *Offsets*, se ofrece la posibilidad de graficar en un plano cartesiano, una señal muestreada durante 640 [ms] (128 pixeles\*5 [ms]) en cualquiera de los tres canales, esto con la finalidad de observar el nivel de offset de las señales. La segunda opción corresponde al almacenamiento de datos, en primer lugar se pedirá al usuario especificar el nombre del archivo, a continuación se verificará que no haya otro archivo con el mismo nombre en la memoria microSD, si el archivo no existe se creará uno nuevo, pero, si otro archivo con el mismo nombre ya ha sido creado, se preguntará al usuario si desea continuar, en cuyo caso, se continuará con la escritura de datos en el archivo existente. Finalmente, se entra a la rutina de adquisición de datos donde el RAS-II se mantendrá operando continuamente a la espera de un fenómeno sísmico.

Para que el RAS-II cumpla con su objetivo final: almacenar en una memoria microSD los datos correspondientes a aceleraciones de la tierra y la hora exacta de ocurrencia de las mismas, el microcontrolador debe solucionar tareas específicas en el momento en que son requeridas. En términos generales, las tareas a solucionar son las siguientes:

- Configuración de la frecuencia de muestreo
- Digitalización y procesamiento de datos
- Almacenamiento de datos en la memoria de pre-evento
- Almacenamiento de datos en la memoria principal
- Transferencia de datos entre la memoria de pre-evento y la memoria principal
- Adquisición de fecha y hora
- Manejo del *joystick* y navegación a través de menús.

Las tareas antes mencionadas son las más críticas, puesto que de ellas depende la correcta digitalización de los datos, el almacenamiento permanente de éstos, la adquisición de la hora y la configuración de los parámetros generales con los que se mantendrá operando el sistema. A continuación se describirá cómo se da solución a cada tarea.

### 3.3.2. Configuración de la frecuencia de muestreo

Una de las primeras tareas que debe resolverse para muestrear la señal analógica, es precisamente establecer la frecuencia de muestreo. Al utilizar un microcontrolador, éste debe tener la capacidad de generar una señal o emitir un aviso en el instante correcto para que la muestra sea tomada. Para lograr esta función, los microcontroladores poseen registros internos capaces de hacer un conteo progresivo; generalmente, los registros van aumentando en una unidad por cada ciclo de reloj del microcontrolador.

Por especificaciones de diseño, la frecuencia de muestreo del RAS-II debe ser de 200 [Hz], una manera de obtener esta frecuencia es con ayuda de un contador. El microcontrolador cuenta con cuatro contadores de 16 bits llamados TCNT (*Timer/Counter*), estos contadores son el TCNT1, TCNT3, TCNT4 y TCNT5. Los contadores pueden trabajar en distintos modos de operación, para ello, los registros de control deben ser configurados. Para establecer la frecuencia de muestreo se utilizó el contador 3 en el modo CTC (*Clear Time on Compare Match*), cabe señalar que se puede utilizar cualquier contador. Trabajando en el modo CTC es posible asignar un valor de cuenta al registro OCR3A (*Output Compare Register 3 A*) de modo que el contador se adecue a la frecuencia de 200 [Hz].

La señal de reloj que llega al contador proviene del oscilador del microcontrolador a 16 [MHz], no obstante, esta frecuencia fue pre-escalada a 64, de manera que la frecuencia del contador es 250 [kHz]. El valor cargado al registro OCR3A se obtiene de la siguiente manera:

$$OCR3A = \frac{T_{muestreo}}{T_{contador}}; \quad \text{con } T = \frac{1}{f} \quad OCR3A = \frac{f_{contador}}{f_{muestreo}} = 1250$$

La figura 3.40 muestra los registros utilizados para establecer la frecuencia de muestreo del RAS-II. Los registros TCCR3A (*Timer/Counter 3 Control Register A*) y TCCR3B (*Timer/Counter 3 Control Register B*) son utilizados para establecer el modo de operación y la fuente de reloj para el contador, conforme a lo establecido en las figuras 3.41 y 3.42 respectivamente. En las figuras 3.41 y 3.42 la “n” se reemplaza por el número del contador utilizado, en este caso por el “3”, por ejemplo, para los bits WGMn3:0 (*Waveform Generation mode 3:0*), se hablaría de los bits WGM33:0.

**TCCR3A – Timer/Counter 3 Control Register A**

Bit	7	6	5	4	3	2	1	0									
(0x90)	<table border="1" style="width:100%; text-align:center;"> <tr> <td>COM3A1</td> <td>COM3A0</td> <td>COM3B1</td> <td>COM3B0</td> <td>COM3C1</td> <td>COM3C0</td> <td>WGM31</td> <td>WGM30</td> </tr> </table>								COM3A1	COM3A0	COM3B1	COM3B0	COM3C1	COM3C0	WGM31	WGM30	TCCR3A
COM3A1	COM3A0	COM3B1	COM3B0	COM3C1	COM3C0	WGM31	WGM30										
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W									
Initial Value	0	0	0	0	0	0	0	0									

**TCCR3B – Timer/Counter 3 Control Register B**

Bit	7	6	5	4	3	2	1	0									
(0x91)	<table border="1" style="width:100%; text-align:center;"> <tr> <td>ICNC3</td> <td>ICES3</td> <td>–</td> <td>WGM33</td> <td>WGM32</td> <td>CS32</td> <td>CS31</td> <td>CS30</td> </tr> </table>								ICNC3	ICES3	–	WGM33	WGM32	CS32	CS31	CS30	TCCR3B
ICNC3	ICES3	–	WGM33	WGM32	CS32	CS31	CS30										
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W									
Initial Value	0	0	0	0	0	0	0	0									

**OCR3AH and OCR3AL – Output Compare Register 3 A**

Bit	7	6	5	4	3	2	1	0									
(0x99)	<table border="1" style="width:100%; text-align:center;"> <tr> <td colspan="8">OCR3A[15:8]</td> </tr> </table>								OCR3A[15:8]								OCR3AH
OCR3A[15:8]																	
(0x98)	<table border="1" style="width:100%; text-align:center;"> <tr> <td colspan="8">OCR3A[7:0]</td> </tr> </table>								OCR3A[7:0]								OCR3AL
OCR3A[7:0]																	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W									
Initial Value	0	0	0	0	0	0	0	0									

**TIFR3 – Timer/Counter3 Interrupt Flag Register**

Bit	7	6	5	4	3	2	1	0									
0x18 (0x38)	<table border="1" style="width:100%; text-align:center;"> <tr> <td>–</td> <td>–</td> <td>ICF3</td> <td>–</td> <td>OCF3C</td> <td>OCF3B</td> <td>OCF3A</td> <td>TOV3</td> </tr> </table>								–	–	ICF3	–	OCF3C	OCF3B	OCF3A	TOV3	TIFR3
–	–	ICF3	–	OCF3C	OCF3B	OCF3A	TOV3										
Read/Write	R	R	R/W	R	R/W	R/W	R/W	R/W									
Initial Value	0	0	0	0	0	0	0	0									

Figura 3.40. Registros del TCNT3.

Waveform Generation Mode Bit Description

Mode	WGMn3	WGMn2 (CTCn)	WGMn1 (PWMn1)	WGMn0 (PWMn0)	Timer/Counter Mode of Operation	TOP	Update of OCRnX at	TOVn Flag Set on
0	0	0	0	0	Normal	0xFFFF	Immediate	MAX
1	0	0	0	1	PWM, Phase Correct, 8-bit	0x00FF	TOP	BOTTOM
2	0	0	1	0	PWM, Phase Correct, 9-bit	0x01FF	TOP	BOTTOM
3	0	0	1	1	PWM, Phase Correct, 10-bit	0x03FF	TOP	BOTTOM
4	0	1	0	0	CTC	OCRnA	Immediate	MAX
5	0	1	0	1	Fast PWM, 8-bit	0x00FF	BOTTOM	TOP
6	0	1	1	0	Fast PWM, 9-bit	0x01FF	BOTTOM	TOP
7	0	1	1	1	Fast PWM, 10-bit	0x03FF	BOTTOM	TOP
8	1	0	0	0	PWM, Phase and Frequency Correct	ICRn	BOTTOM	BOTTOM
9	1	0	0	1	PWM, Phase and Frequency Correct	OCRnA	BOTTOM	BOTTOM
10	1	0	1	0	PWM, Phase Correct	ICRn	TOP	BOTTOM
11	1	0	1	1	PWM, Phase Correct	OCRnA	TOP	BOTTOM
12	1	1	0	0	CTC	ICRn	Immediate	MAX
13	1	1	0	1	(Reserved)	–	–	–
14	1	1	1	0	Fast PWM	ICRn	BOTTOM	TOP
15	1	1	1	1	Fast PWM	OCRnA	BOTTOM	TOP

Figura 3.41. Especificaciones del modo de operación del contador.

Clock Select Bit Description

CSn2	CSn1	CSn0	Description
0	0	0	No clock source. (Timer/Counter stopped)
0	0	1	$clk_{I/O}/1$ (No prescaling)
0	1	0	$clk_{I/O}/8$ (From prescaler)
0	1	1	$clk_{I/O}/64$ (From prescaler)
1	0	0	$clk_{I/O}/256$ (From prescaler)
1	0	1	$clk_{I/O}/1024$ (From prescaler)
1	1	0	External clock source on Tn pin. Clock on falling edge
1	1	1	External clock source on Tn pin. Clock on rising edge

Figura 3.42. Especificaciones de la fuente de reloj para el contador.

Conforme a lo establecido en la Figura 3.41, los bits WGM33, WGM32, WGM31 y WGM30 deben tener un valor en conjunto de 4 o 12 (0100 o 1100 en binario) para operar en el modo CTC. A grandes rasgos, la diferencia entre trabajar con el modo 4 ó 12 reside en el registro al que se cargará el valor de cuenta, para el primer caso, el valor de cuenta se cargará al registro OCR3A, mientras que en el segundo caso se cargará al registro ICR3 (*Input Compare Register 3*), del mismo modo, las banderas levantadas serán OCF3A (*Output Compare Flag 3A*) o ICF3 (*Input Compare Flag 3*), respectivamente, al alcanzar el valor de cuenta cargado. Puesto que no hay diferencia en trabajar con cualquiera de las dos unidades, se eligió trabajar con el registro OCR3A y la bandera OCF3A en el modo 4.

Para pre-escalar la frecuencia de 16 MHz, en la figura 3.42 se indica el valor que los bits CS32:0 (*Clock-Select3 2:0*) deben tomar. Para pre-escalar la frecuencia a 64, los bits mencionados deben tomar un valor de 3 (011 en binario). La sección de código de programación correspondiente a la configuración de la frecuencia de muestreo es la mostrada a continuación:

```
TCCR3A = 0;
TCCR3B = (1<<WGM32) | (1<<CS31) | (1<<CS30);
OCR3A = 1250
```

Las dos primeras líneas permiten la configuración de la fuente de reloj y el modo de operación, escribiendo un uno lógico a los bits necesarios para tal fin; la tercer línea asigna el valor de cuenta del contador. Una vez hecha la configuración del contador, el microcontrolador es capaz de señalar, a una tasa de 200 [Hz], el momento en que debe tomarse la muestra, esta señalización se realiza al monitorear el estado de la bandera OCF3A en el registro TIFR3 (*Timer/Counter3 Interrupt Flag Register*).

### 3.3.3. Digitalización y procesamiento de datos

En el proceso de digitalización de una señal, el convertidor analógico-digital es el principal actor de esta tarea. Habiendo establecido una frecuencia de muestreo, el microcontrolador se comunicará con el convertidor analógico-digital para obtener el correspondiente valor digital de la muestra tomada. El microcontrolador además de recibir el valor digital, también se encarga de proporcionar las señales de reloj y la activación del circuito integrado para que la comunicación se realice de manera sincronizada.

#### *Digitalización*

El convertidor analógico-digital utilizado tiene 4 modos de operación, la diferencia entre uno y otro radica básicamente en la fuente de reloj utilizada por éste y la operación continua al habilitar o inhabilitar permanentemente la terminal Chip Select. El convertidor fue utilizado en el modo “*Operación de ciclo sencillo y reloj externo*”, en este modo de operación el convertidor usa una fuente de reloj externa para sincronizar la salida de datos hacia el microcontrolador y se utiliza la señal CS para llevar el control de la salida de datos.

En el caso del ADC utilizado, el primer paso para lograr la digitalización de una señal es configurar la velocidad de conversión y la resolución o número efectivo de bits (ENOB) asociado a ella. Conforme a las especificaciones indicadas en la figura 3.43, existen 11 opciones para establecer la velocidad/resolución con las que operará el convertidor. Los bits OSR4:0 (*Over Sampling Rate 4:0*) son enviados por la línea MOSI desde el microcontrolador hacia el ADC para configurarlo. De entre las opciones disponibles, debe elegirse aquella que permita un equilibrio entre el número efectivo de bits y la velocidad de conversión; por un lado, entre mayor sea el ENOB, el sistema será más preciso en la lectura de voltajes y, por el otro lado, la frecuencia de conversión debe ser mayor a la frecuencia de muestreo de 200 [Hz].

Para la operación del ADC se decidió utilizar el reloj interno de este circuito integrado. De las opciones mostradas en la figura 3.43, únicamente pueden elegirse aquellas cuya frecuencia de conversión es mayor o igual a 440 [Hz], considerando que el muestreo de las señales de los acelerómetros debe realizarse a una tasa de 200 [Hz]. De las opciones disponibles, la mejor es aquella que proporciona el mayor ENOB, sin embargo, con poco sacrificio de este parámetro, se eligió aquella cuyo ENOB es igual a 21.3. La elección hecha permite la simplificación del diseño de la placa de circuito impreso que se fabricará posteriormente. Tal y como lo indica la figura 3.43, al utilizar la opción con el símbolo “\*\*”, la terminal SDI simplemente se conecta a tierra para la operación del circuito integrado, de esta manera, se logra reducir el ruteo de tres líneas de comunicación (3 ADCs) hacia el microcontrolador.

## SDI Speed/Resolution Programming

OSR4	OSR3	OSR2	OSR1	OSR0	CONVERSION RATE		RMS NOISE	ENOB	OSR
					INTERNAL 9MHz CLOCK	EXTERNAL 10.24MHz CLOCK			
X	0	0	0	1	3.52kHz	4kHz	23 $\mu$ V	17	64
X	0	0	1	0	1.76kHz	2kHz	3.5 $\mu$ V	20	128
0	0	0	0	0	880Hz	1kHz	2 $\mu$ V	21.3	256*
X	0	0	1	1	880Hz	1kHz	2 $\mu$ V	21.3	256
X	0	1	0	0	440Hz	500Hz	1.4 $\mu$ V	21.8	512
X	0	1	0	1	220Hz	250Hz	1 $\mu$ V	22.4	1024
X	0	1	1	0	110Hz	125Hz	750nV	22.9	2048
X	0	1	1	1	55Hz	62.5Hz	510nV	23.4	4096
X	1	0	0	0	27.5Hz	31.25Hz	375nV	24	8192
X	1	0	0	1	13.75Hz	15.625Hz	250nV	24.4	16384
X	1	1	1	1	6.875Hz	7.8125Hz	200nV	24.6	32768**

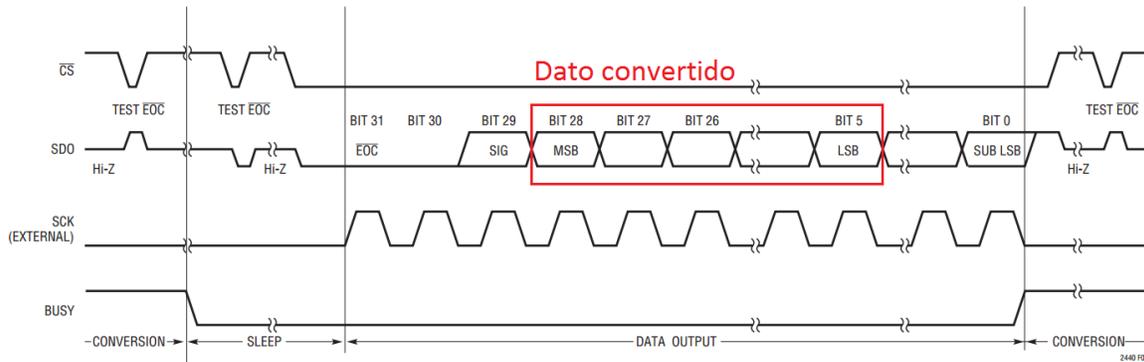
\*\*Address allows tying SDI HIGH \*Additional address to allow tying SDI LOW

Figura 3.43. Opciones del ADC.

Una vez que la velocidad y resolución del convertidor se han fijado, éste se encargará de proporcionar el valor digital equivalente de la muestra tomada; la siguiente tarea es enviar el dato del convertidor hacia el microcontrolador para su posterior manipulación.

El funcionamiento del ADC en el modo de operación antes mencionado se resume esquemáticamente en el diagrama de tiempo de la figura 3.44. La señal  $\overline{CS}$  es utilizada para monitorear el estado de la conversión, o bien, para iniciar la transferencia de datos hacia el microcontrolador. Mientras la señal  $\overline{CS}$  se mantenga en un nivel alto, la terminal SDO permanecerá en alta impedancia. Al poner la señal  $\overline{CS}$  en un valor bajo, se puede monitorear el estado de la conversión; el bit  $\overline{EOC}$  tomará un valor alto si una conversión está en proceso, en caso contrario, tomará un valor bajo indicando que la conversión ha terminado y el dato digital se encuentra listo para ser transferido.

Cuando la conversión se ha terminado, el convertidor entra automáticamente en un estado de reposo independientemente del estado de  $\overline{CS}$ ; el dato es almacenado en un registro interno estático y el dispositivo se mantiene en ese estado hasta que se presente el primer flanco de subida de la fuente de reloj. Manteniendo la señal  $\overline{CS}$  en un nivel bajo y proporcionando una fuente de reloj, los bits correspondientes al dato digitalizado son transferidos en cada flanco de bajada de la fuente de reloj; el último bit es transferido en el 32avo ciclo.



External Serial Clock, Single Cycle Operation

Figura 3.44. Operación del ADC.

En cuanto a la programación del microcontrolador, se utilizó una función nombrada “*SpiRead1*” para obtener el dato proporcionado por el convertidor analógico-digital. La sección de código correspondiente a la recuperación del dato digital se muestra a continuación.

```

long SpiRead1(void)
{
    long result = 0;
    long b;
    digitalWrite(cs1,LOW);
    b = SPI.transfer(0xff);
    result = b<<8;
    b = SPI.transfer(0xff);
    result |= b;
    result = result<<8;
    b = SPI.transfer(0xff);
    result |= b;
    result = result<<8;
    b = SPI.transfer(0xff);
    result |= b;
    digitalWrite(cs1,HIGH);
    long tension = result;
    tension &=0x1FFFFFFF;
    tension = tension>>5;
    return(tension);
}

```

El primer paso es activar la selección del circuito integrado proporcionando un valor bajo en la terminal  $\overline{CS}$ , a continuación, se utiliza la función *SPI.transfer()* implementada en la librería SPI.h de Arduino; con esto se activa la fuente de reloj y los primeros 8 bits del dato digital son proporcionados por el convertidor.

Cabe comentar que se utilizaron tres variables auxiliares de 32 bits (*long*) para obtener la cadena final; en la variable *b* se almacenan los primeros 8 bits, en seguida, el dato es recorrido 8 posiciones a la izquierda y es almacenado en la variable *result*. Los siguientes 8 bits son almacenados nuevamente en la variable *b* y se realiza la operación OR con la variable *result* para de esta manera, tener los primeros 16 bits. Los pasos siguientes son análogos hasta obtener la totalidad de los bits correspondientes al dato digitalizado. Al tener los 32 bits completos se desactiva la selección del circuito integrado proporcionando un valor alto a la terminal  $\overline{CS}$ .

Los bits de interés son los correspondientes a las posiciones 28 a 5 (ver figura 3.44) de la cadena total de 32 bits, por lo tanto, el paso final es eliminar aquellos bits que no corresponden a estas posiciones. La variable *tension* toma el valor de la cadena de 32 bits, en seguida se realiza la operación AND entre ésta y el valor  $0x1FFFFFFF$  eliminando así los bits 31, 30 y 29. Finalmente, se realiza un corrimiento de 5 posiciones a la derecha eliminando los bits menos significativos.

*Procesamiento de datos*

Después de obtener el valor digital, debe realizarse un segundo proceso, en éste, se establece la correspondencia entre el código binario y su valor numérico asociado en volts. La representación de datos con signo utilizada por el convertidor analógico-digital es mediante el complemento a 2. En la figura 3.45 se especifica el formato del dato convertido. Analizando la tabla, se identifican 4 intervalos en los que el dato se encuentra:

Con  $V_{REF} = 5V$

1.  $V_{IN} \geq 2.5V$
2.  $V_{IN} < -2.5V$
3.  $0V \leq V_{IN} < 2.5V$
4.  $-2.5V \leq V_{IN} < 0V$

LTC2440 Output Data Format

Differential Input Voltage $V_{IN}^*$	Bit 31 EOC	Bit 30 DMY	Bit 29 SIG	Bit 28 MSB	Bit 27	Bit 26	Bit 25	...	Bit 0
$V_{IN}^* \geq 0.5 \cdot V_{REF}^{**}$	0	0	1	1	0	0	0	...	0
$0.5 \cdot V_{REF}^{**} - 1LSB$	0	0	1	0	1	1	1	...	1
$0.25 \cdot V_{REF}^{**}$	0	0	1	0	1	0	0	...	0
$0.25 \cdot V_{REF}^{**} - 1LSB$	0	0	1	0	0	1	1	...	1
0	0	0	1	0	0	0	0	...	0
-1LSB	0	0	0	1	1	1	1	...	1
$-0.25 \cdot V_{REF}^{**}$	0	0	0	1	1	0	0	...	0
$-0.25 \cdot V_{REF}^{**} - 1LSB$	0	0	0	1	0	1	1	...	1
$-0.5 \cdot V_{REF}^{**}$	0	0	0	1	0	0	0	...	0
$V_{IN}^* < -0.5 \cdot V_{REF}^{**}$	0	0	0	0	1	1	1	...	1

\*The differential input voltage  $V_{IN} = IN^+ - IN^-$ . \*\*The differential reference voltage  $V_{REF} = REF^+ - REF^-$ .

Figura 3.45. Formato del dato convertido.



Si el dato no se encuentra en ninguno de los rangos anteriores, sólo hay dos opciones: que el dato sea positivo pero inferior a 2.5 [V] o negativo pero superior a -2.5 [V]. En la figura 3.45 se observa que el bit más significativo de todos los datos positivos (pero inferiores a 2.5 [V]) es igual a 0; análogamente, el MSB de todos los valores negativos (pero superiores a -2.5 [V]) es igual a 1. Con ayuda de esta particularidad, el siguiente paso es discernir si se trata de un valor positivo o negativo, para ello se realiza una operación lógica AND entre el dato convertido y el MSB. Si el resultado de la operación es cero, se tratará de un valor positivo, en caso contrario, se hablará de un valor negativo.

Para la obtención en volts de un valor positivo, se realiza la multiplicación del dato convertido con el valor de la resolución del convertidor analógico-digital. La operación es simple pues cada bit tiene un valor posicional, cada bit equivale al valor mínimo que puede manejar el convertidor ( $resolución = \frac{5V}{2^{21}-1}$ ), de modo que al realizar el producto se obtiene una cifra que corresponde al voltaje medido por el convertidor.

Para el caso de los valores negativos, los bits en la cadena no tienen un valor posicional, simplemente es la representación en complemento a 2 de un valor negativo. Existe una ecuación que relaciona el complemento a dos y el complemento a 1 en el sistema binario:

$$C_2^N = C_1^N + 1 \quad (3.12)$$

Donde:

$$\begin{aligned} C_2^N &= \text{complemento a dos} \\ C_1^N &= \text{complemento a uno} \end{aligned}$$

El complemento a uno es a su vez el valor negado del dato negativo convertido, esto es:

$$C_1^N = \bar{N} \quad (3.13)$$

Combinando (3.12) y (3.13) se puede obtener el valor negativo N con la ecuación siguiente:

$$N = \overline{(C_2^N - 1)} \quad (3.14)$$

A manera de ejemplo supóngase un ADC hipotético de 4 bits, con resolución igual a 1 [V] y con representación de datos en complemento a dos. Al convertir un voltaje analógico de -7V, el convertidor entregaría la cadena 1001, la manera de obtener su correspondiente valor en volts es utilizando la ecuación (3.14):

$$N = \overline{(C_2^N - 1)} = \overline{(1001 - 1)} = \overline{1000} = 0111 = 7_{Dec}$$

Finalmente, al multiplicar el resultado por la resolución, se obtiene el correspondiente voltaje negativo de dicho convertidor.

De esta manera, el proceso para obtener el valor negativo es primero restar una unidad al dato convertido, a continuación se realiza una operación NOT del dato convertido, dando como resultado la inversión de unos y ceros. En este paso surge otro inconveniente con el ADC; retomando el ADC hipotético, suponer ahora que aunque el ADC es de 4 bits, el microcontrolador que lo maneja almacena el dato en una variable de 5 bits, en este caso la lectura de -7V se representaría con la cadena 01001, siguiendo el procedimiento anterior:

$$N = \overline{(C_2^N - 1)} = \overline{(01001 - 1)} = \overline{01000} = 10111 \neq 7_{Dec}$$

La manera de contrarrestar la inversión del bit más significativo es realizando una operación AND con el valor 01111:

$$N = 01111 \& 10111 = 0111 = 7_{Dec}$$

Como el dato *ch* de tipo *long* se almacena en una cadena de 32 bits, los 9 bits más significativos que originalmente eran ceros, cambian a unos al realizar la operación NOT, para contrarrestar este efecto, se eliminan estos bits al realizar una operación AND entre el dato y el número 0x001FFFFFF. En el dato obtenido, después de estos procesos, los bits ya cuentan con un valor numérico posicional; al multiplicar por la resolución y por -1, se tendrá el voltaje negativo leído por el convertidor.

### 3.3.4. Manejo de la memoria de pre-evento

La memoria de pre-evento permite almacenar datos temporales a lo largo del tiempo de operación del RAS-II, es en este dispositivo donde las lecturas de voltaje son almacenadas y sobrescritas en el momento en que ya no son útiles. La memoria RAM cuenta con tres modos de operación, a saber: byte, página y secuencial, variando entre uno y otro los límites establecidos en la transferencia de datos; así, en el modo byte las operaciones de lectura/escritura se limitan a un byte; en el modo página a una página de 32 bytes y en el modo secuencial la operación continúa hasta alcanzar el límite de capacidad de la memoria.

El modo de operación de la memoria RAM está determinado por la configuración de su registro de operación mostrado en la figura 3.46. El modo de operación por default es el modo secuencial; trabajando en este modo se permiten lecturas y escrituras a lo largo de todas las localidades de memoria, el contador interno de dirección de memoria es automáticamente incrementado y los límites de página son ignorados. Cuando el contador interno alcanza el fin de localidades de memoria 0x1FFFF, éste regresa al valor inicial 0x00000.

**MODE REGISTER**

7	6	5	4	3	2	1	0
W/R	W/R	-	-	-	-	-	-
MODE	MODE	0	0	0	0	0	0
W/R = writable/readable							

0 0 = Byte mode

1 0 = Page mode

0 1 = Sequential mode (default operation)

1 1 = Reserved

Bits 0 through 5 are reserved and should always be set to '0'.

Figura 3.46. Registro de modo de operación de la memoria RAM.

En el listado que se presenta líneas abajo se muestra la sección del código correspondiente a la escritura de datos a la memoria RAM, englobada en la rutina de adquisición de datos que se tratará posteriormente. Se muestran también las funciones auxiliares utilizadas para realizar la transferencia de datos.

```
long ch1 = SpiRead1(); float valor1 = signo(ch1);
long ch2 = SpiRead2(); float valor2 = signo(ch2);
long ch3 = SpiRead3(); float valor3 = signo(ch3);
```

```
if(saving_data == false)
{
    if(samples<10000)
        samples++;
    index++;
    if(index==10000)
        index=0;
    divide_address(memory_location);
    divide_data(ch1);
    digitalWrite(csram,LOW);
    SPI.transfer(0x02);
    transmit_data(D_2,D_1,D_0);
    transmit_data(B_2,B_1,B_0);
    divide_data(ch2);
    transmit_data(B_2,B_1,B_0);
    divide_data(ch3);
    transmit_data(B_2,B_1,B_0);
    digitalWrite(csram,HIGH);
    memory_location = index*9;
}
```

```
long B_0,B_1,B_2;
long D_0,D_1,D_2;

void divide_address(long addr)
{
    D_1=highByte(addr);
    D_0=lowByte(addr);
    addr=addr>>16;
    D_2=lowByte(addr);
}
void divide_data(long ch)
{
    B_1=highByte(ch);
    B_0=lowByte(ch);
    ch=ch>>16;
    B_2=lowByte(ch);
}

void transmit_data(long A_2, long A_1, long A_0)
{
    SPI.transfer(A_2);
    SPI.transfer(A_1);
    SPI.transfer(A_0);
}
```

Las muestras tomadas en cada canal se almacenan en las variables *valor1*, *valor2* y *valor3*, estas variables son del tipo *float* cuya representación es una cadena de 24 bits.

Por especificaciones de diseño, se estableció que el sistema debía almacenar 50 segundos de pre-evento, de esta manera, la variable auxiliar *index* se utiliza para generar la dirección de memoria a la cual se accederá para almacenar las muestras. El número de la última muestra que será tomada, en función de los 50 segundos de pre-evento, es:  $200 \text{ m.p.s} * 50\text{s} = 10,000$ ; *index* aumentará en una unidad progresivamente hasta llegar a ese valor máximo, cuando sea alcanzado, *index* retornará al valor inicial cero.

El primer paso para utilizar la memoria RAM es activar el circuito integrado con un valor lógico bajo. Conforme a lo establecido en la figura 3.47, para llevar a cabo una secuencia de operación de escritura, se debe enviar el valor 0x02 desde el microcontrolador hacia la memoria RAM mediante el protocolo SPI.

**INSTRUCTION SET**

Instruction Name	Instruction Format	Hex Code	Description
READ	0000 0011	0x03	Read data from memory array beginning at selected address
WRITE	0000 0010	0x02	Write data to memory array beginning at selected address
EDIO	0011 1011	0x3B	Enter Dual I/O access
EQIO	0011 1000	0x38	Enter Quad I/O access
RSTIO	1111 1111	0xFF	Reset Dual and Quad I/O access
RDMR	0000 0101	0x05	Read Mode Register
WRMR	0000 0001	0x01	Write Mode Register

Figura 3.47. Conjunto de instrucciones de la memoria RAM.

Después de activar la selección del circuito integrado, se envía la localidad de memoria en la cual el dato será almacenado, este dato es una cadena de 24 bits. Dado que la función *SPI.transfer* únicamente transmite 8 bits, tanto la localidad de memoria como las muestras tomadas por canal, son seccionadas en tres grupos de 8 bits mediante las funciones *divide\_address()* y *divide\_data()*.

En primer lugar se envía desde el microcontrolador hacia la memoria, la dirección de memoria, a continuación, se envían las muestras tomadas en cada canal. En la memoria utilizada, cada localidad se constituye de 8 bits; cada muestra consta de 3 grupos de 8 bits (24 bits en total), por lo tanto, al tener tres muestras (una por canal) se utilizan en total 9 localidades de memoria para almacenar los datos.

Con el fin de almacenar las muestras consecutivamente a lo largo de las localidades de memoria, la variable *memory\_location* aumenta 9 unidades en cada proceso de muestreo. La última localidad de memoria a la que se accede es la correspondiente a la muestra 10000, que al multiplicar por el factor 9, resulta en la localidad 90000. Finalmente, una señal en alto es enviada desde el microcontrolador y la memoria RAM se inhabilita.

### 3.3.5. Almacenamiento de datos en la memoria principal

En cuanto a software se refiere, el funcionamiento del sistema recae por completo en la rutina de adquisición de datos; en esta rutina se dota al sistema de la capacidad para gestionar el almacenamiento de datos. En términos generales, son tres las tareas que se deben resolver en esta rutina:

1. Almacenar 50 segundos de pre-evento en la memoria secundaria (RAM).
2. Iniciar el almacenamiento del evento en la memoria principal (microSD) al sobrepasar el umbral de disparo establecido por el usuario.
3. Almacenar 60 segundos de post-evento en la memoria principal.

Con el objetivo de lograr un entendimiento pleno de las tareas a resolver es conveniente realizar las definiciones siguientes:

*Pre-evento:* Es la información correspondiente a la señal bajo estudio, obtenida un tiempo antes de sobrepasar el umbral de disparo, la señal es menor que el umbral.

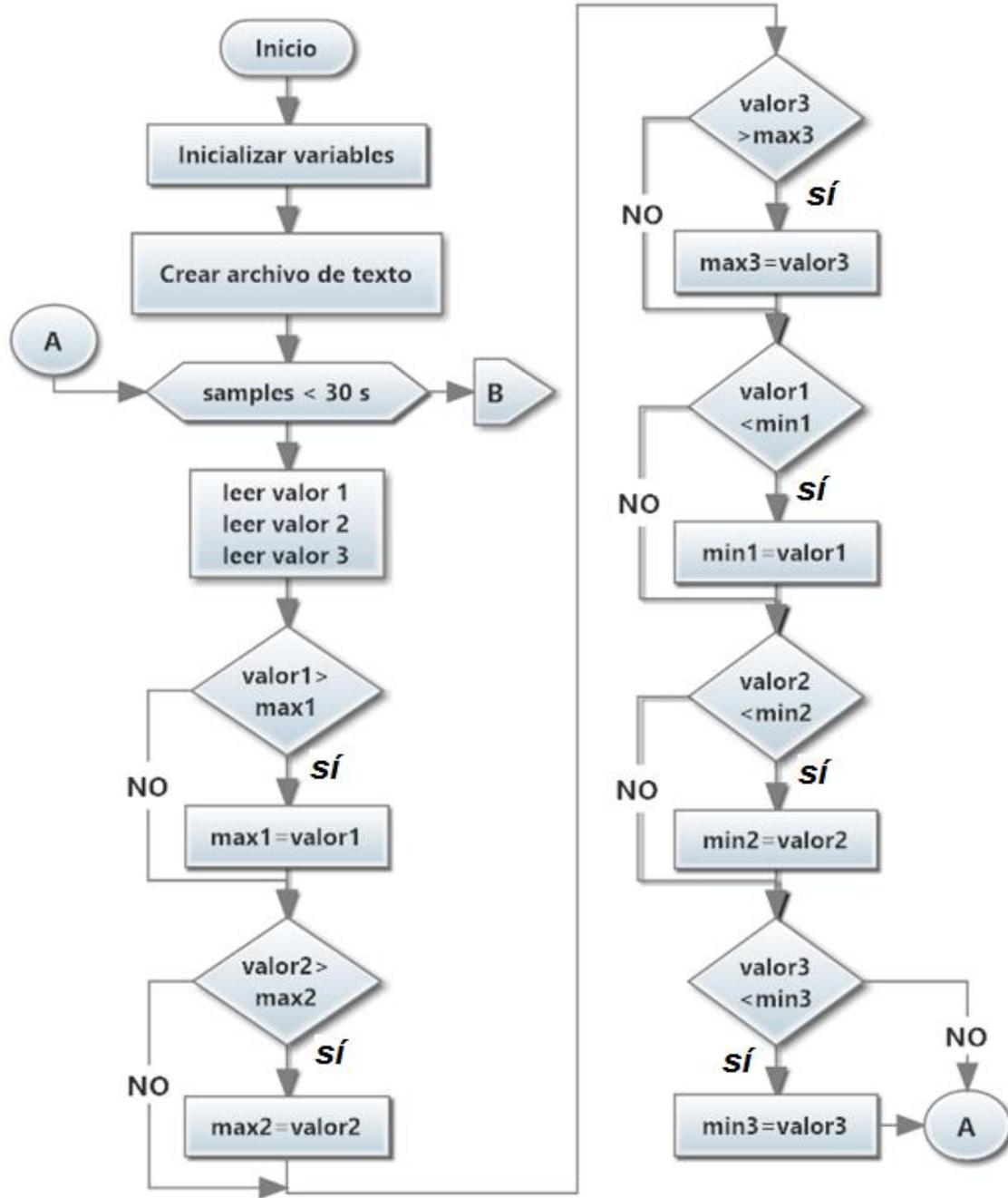
*Evento:* Es la información correspondiente a la señal bajo estudio, obtenida durante todo el tiempo en que la señal sobrepase el umbral de disparo.

*Post-evento:* Es la información correspondiente a la señal bajo estudio, obtenida durante cierto tiempo después de sobrepasar el umbral de disparo, la señal es menor que el umbral.

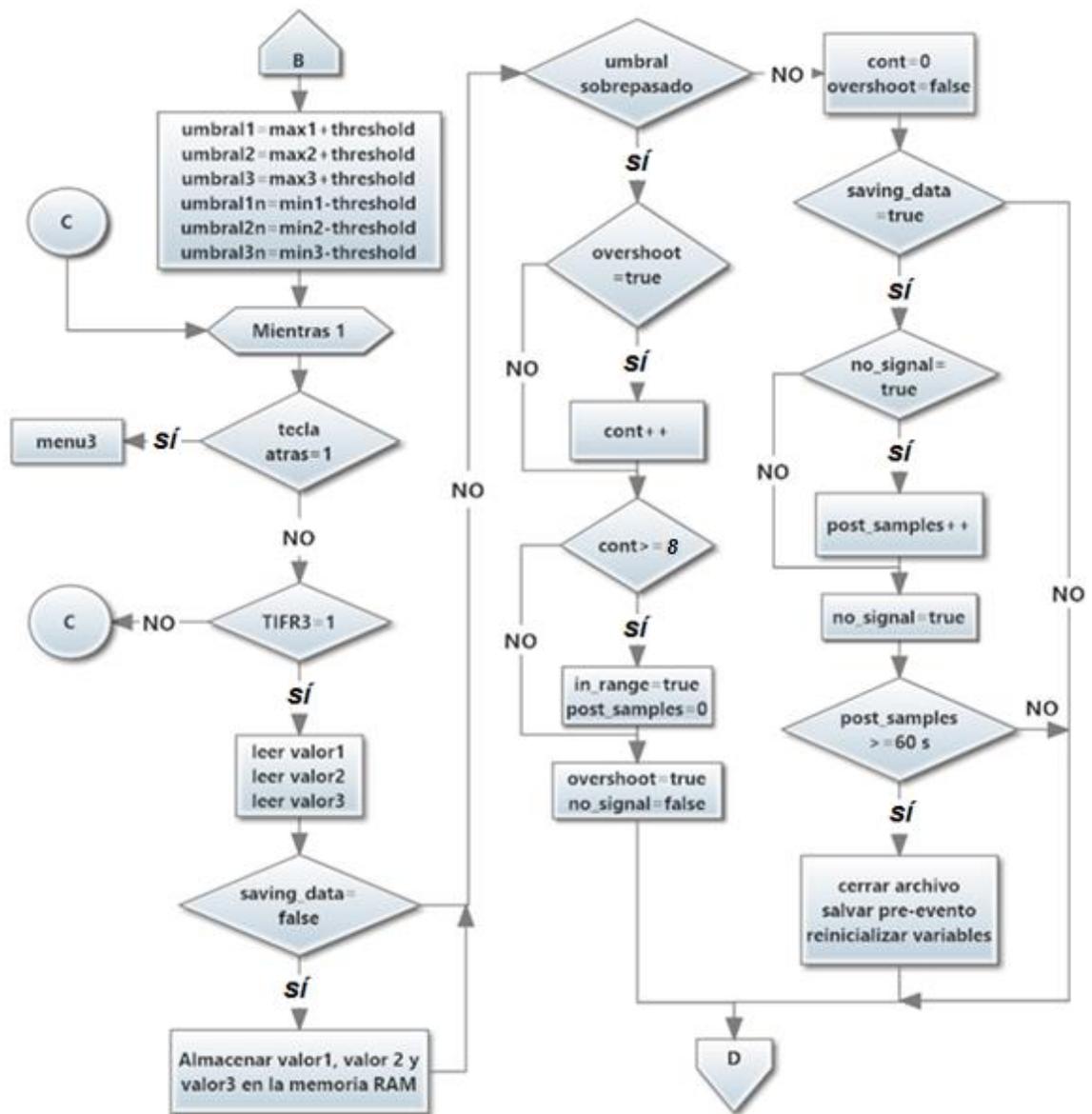
*Umbral de disparo:* Es un valor de voltaje establecido (voltaje de disparo) para definir que una vez superado éste, se tiene un evento sísmico.

En las figuras 3.48 a, b y c se muestra el diagrama de flujo de la rutina de adquisición de datos. Primero se inicializan todas las variables que serán utilizadas a lo largo de la ejecución de la rutina, a continuación se crea el archivo de texto donde se almacenarán los datos correspondientes al tiempo, lectura del canal 1, lectura del canal 2, lectura del canal 3, además de la fecha obtenida del GPS y el nombre de la prueba establecida por el usuario.

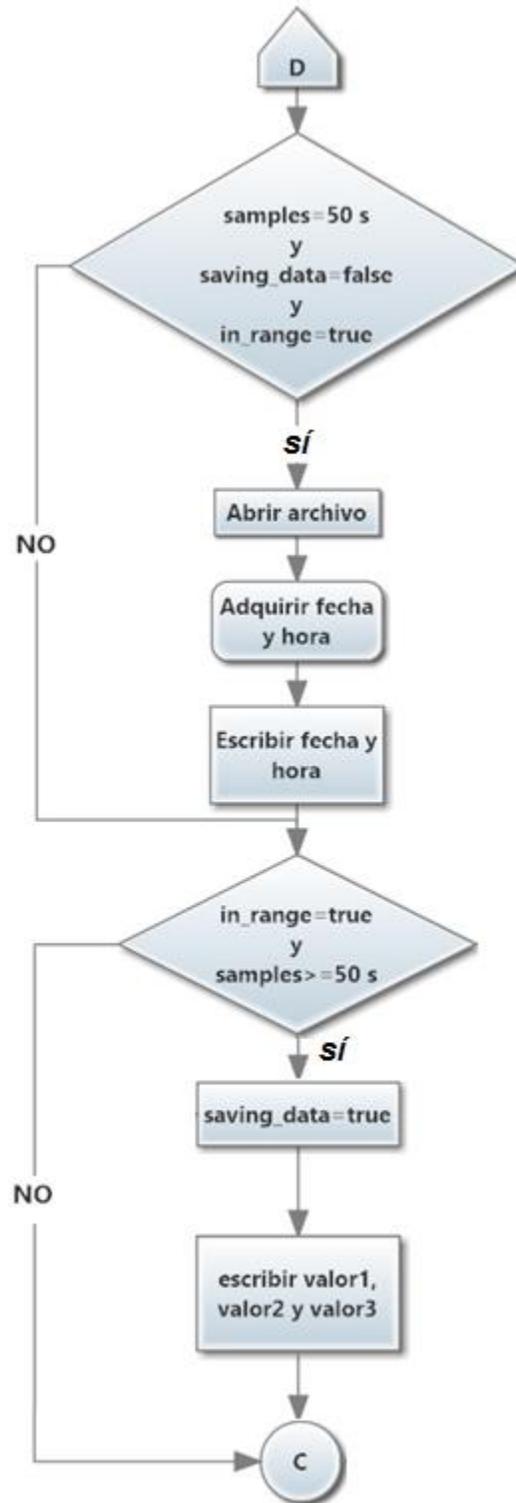
Después de realizar las tareas anteriormente mencionadas, se debe establecer el voltaje de disparo con el que comenzará el registro de un evento. Puesto que el sistema está pensado para ser utilizado con diferentes acelerómetros, la primer tarea a realizar es medir el nivel de offset de éstos. La señal que se obtiene de los acelerómetros en una posición de reposo es un voltaje de directa con pequeñas variaciones alrededor del mismo; la obtención de la variación máxima y mínima permite establecer el voltaje de disparo.



(a) Continúa



(b) Continúa



(c)

Figura 3.48. Diagrama de flujo de la rutina de adquisición de datos.

Los voltajes de disparo se obtienen de la siguiente manera: después de crear el archivo, durante un tiempo de 30 segundos se realizará el monitoreo de las señales en cada uno de los tres canales. En cada lectura se adquieren los valores de voltaje correspondientes a los canales 1, 2 y 3; en cada proceso de lectura se compara el valor del canal con la lectura anterior de dicho canal, si el valor actual es mayor que el dato anterior, entonces se conserva ese valor como máximo, por otro lado, se realiza una segunda comparación del dato actual con el anterior y si el dato actual es menor que el anterior, se conserva el actual como el valor mínimo. Al finalizar el tiempo de monitoreo, se tendrán los valores máximos y mínimos de cada canal.

Continuando con el diagrama de flujo en la figura 3.48 (b) el primer bloque de asignaciones establece los voltajes de disparo en cada canal, en cada uno existen dos niveles de disparo, uno inferior y otro superior; cuando la señal excede cualquiera de ellos, el sistema iniciará el almacenamiento del evento. La forma de obtener estos voltajes de disparo es sumando el umbral establecido por el usuario y el valor máximo del canal correspondiente, obteniendo el voltaje de disparo superior. Para el voltaje de disparo inferior, se resta el umbral establecido por el usuario, del valor mínimo del canal correspondiente.

El umbral elegido por el usuario, que se ha mencionado anteriormente, se encuentra en el rango de 0.5 a 10 [Gal], con incrementos de 0.5 [Gal], donde:

$$1 \text{ Gal} = 0.01 \frac{m}{s^2}$$

El rango de voltaje de los acelerómetros utilizados para estas aplicaciones es típicamente de  $\pm 2.5V$ , no obstante, este valor de voltaje corresponde para algunos acelerómetros a una escala de  $\pm 2g$  y para otros a una escala de  $\pm 1g$ . La equivalencia en volts de 1 [Gal] se obtiene con las ecuaciones (3.15) y (3.16) para acelerómetros de  $\pm 2g$  y  $\pm 1g$ , respectivamente.

$$1 \text{ Gal} = 1 \text{ Gal} * \frac{0.01 \frac{m}{s^2}}{1 \text{ Gal}} * \frac{1 g}{9.81 \frac{m}{s^2}} * \frac{2.5V}{2g} = 1.274 [mV] \quad (3.15)$$

$$1 \text{ Gal} = 1 \text{ Gal} * \frac{0.01 \frac{m}{s^2}}{1 \text{ Gal}} * \frac{1 g}{9.81 \frac{m}{s^2}} * \frac{2.5V}{1g} = 2.548 [mV] \quad (3.16)$$

A manera de ejemplo, supóngase que durante el monitoreo de los *offsets* de los acelerómetros, los voltajes máximo y mínimo obtenidos en el canal 1 fueron  $V_{\text{máx}_{CH1}} = 10 [mV]$  y  $V_{\text{mín}_{CH1}} = 5 [mV]$ , además el umbral establecido por el usuario fue de 1 [Gal] y el acelerómetro es de  $\pm 2g$ , por lo tanto los voltajes de disparo para el canal 1 son:

$$\begin{aligned} V_{\text{superior}} &= V_{\text{máx}_{CH1}} + \text{umbral} = 11.274 \text{ mV} \\ V_{\text{inferior}} &= V_{\text{mín}_{CH1}} - \text{umbral} = 3.726 \text{ mV} \end{aligned}$$

Después de establecer los voltajes de disparo, se entrará en un ciclo de repetición infinita mediante la estructura *while(1)*. En esta estructura se realizará un poleo continuo de la tecla atrás (izquierda); si se ha presionado, se regresará al menú anterior, por otro lado, se realiza el poleo de la bandera de interrupción del contador 3 (OCF3A) en el registro TIFR3, que ha sido configurada previamente para levantarse cada 5 [ms] (o a una frecuencia de 200 [Hz]).

El muestreo de la señal se realiza a una frecuencia de 200 [Hz], si la variable booleana *saving\_data* tiene un valor falso, se almacenará cada valor leído en la memoria RAM, permitiendo de esta manera, almacenar datos en esta memoria únicamente durante el pre-evento. En la memoria RAM se almacenan sólo 50 segundos de pre-evento, en consecuencia, en el tiempo igual a 50.005 [s], comienza la sobre-escritura de los datos en la primer localidad de memoria y así sucesivamente.

El paso siguiente es examinar si se han sobrepasado los voltajes de disparo y realizar un manejo de banderas auxiliares. Como primer ejemplo, si se ha sobrepasado cualquier voltaje de disparo, la variable booleana *overshoot* tomará un valor verdadero y en el siguiente muestreo permitirá conocer si anteriormente se ha sobrepasado el umbral. Si ocho muestras consecutivas que sobrepasen el umbral son obtenidas, la bandera *in\_range* toma un valor verdadero y es en ese momento en que se inicia el almacenamiento del evento. Cabe la pena recordar que una de las especificaciones del usuario fue tener un filtro paso bajas con frecuencia de corte igual a 12 [Hz], considerando una señal senoidal, esto presupone que las señales de interés se mantendrían por arriba del umbral por un tiempo determinado, para el caso de la señal senoidal, el tiempo en elevarse desde el nodo hasta la cresta toma 41.66 [ms], de esta manera, se obtendrían 8 muestras consecutivas por arriba del umbral en el proceso de muestreo. La consideración de que la señal se mantenga por al menos 40 [ms] previene que el RAS-II sea disparado por fenómenos fuera de interés como un golpe puntual sobre la base del equipo.

En la figura 3.48(c), como lo indica la primer estructura condicional, en el momento en el que se cumplan las tres condiciones citadas abajo, se abrirá el archivo, se obtendrá la fecha del GPS y se escribirán en el archivo las lecturas tomadas.

*Condiciones:*

1. La memoria RAM debe contener 50 [s] de pre-evento (*samples=50 [s]*).
2. No se debe haber accedido previamente al almacenamiento de datos (*saving\_data=false*).
3. Debe haberse tenido alguna perturbación con duración mayor a 20 [ms] en cualquier canal (*in\_range=true*).

La segunda estructura condicional en la figura 3.48(c) permite gestionar el almacenamiento del evento una vez que se ha escrito la fecha en el archivo, siempre y cuando una perturbación de interés tenga lugar, y por lo tanto, *in\_range* tome un valor verdadero, el almacenamiento del evento tendrá lugar en la memoria principal.

Regresando a la figura 3.48(b), al preguntar si los voltajes de disparo han sido sobrepasados y la respuesta es negativa, la variable *post\_samples* permite contabilizar cuántas muestras no han sobrepasado el voltaje de disparo o en otras palabras, contabilizar cuántas muestras no corresponden a un evento. En el momento en el que se adquieran 60 segundos de muestras que no correspondan a un evento, se cerrará el archivo de datos, para así lograr que estos sean guardados definitivamente, también se salvará la información correspondiente al pre-evento que hasta ese momento se encuentra almacenada en la memoria RAM, esto se explicará en el siguiente apartado.

### 3.3.6. Transferencia de datos entre la memoria de pre-evento y la memoria principal.

Si durante 60 segundos posteriores a un evento las señales en los 3 canales no sobrepasan los voltajes de disparo, el sistema culminará con el almacenamiento del evento y post-evento cerrando un primer archivo en la memoria microSD (memoria principal). Los datos correspondientes al pre-evento se encuentran almacenados en los registros de la memoria secundaria, memoria de pre-evento o memoria RAM, en este apartado se describe la transferencia de estos datos hacia la memoria principal y la creación de un segundo archivo de datos.

Se presenta un problema al intentar almacenar el pre-evento, evento y post-evento en el mismo documento de texto, este problema tiene que ver con el tiempo para la escritura de caracteres en el archivo mediante el protocolo SPI. El tiempo de escritura de las 3 lecturas en los canales oscila entre 1 y 2 [ms]; considerando que se tienen 30000 datos de pre-evento ( $200\text{mps} * 50\text{s} * 3\text{canales}$ ), el tiempo necesario para la escritura del pre-evento sería aproximadamente de 1 minuto. Al sobrepasar el umbral de disparo, el sistema debe inmediatamente comenzar a almacenar datos en el archivo de texto, por tal motivo, la pérdida de 1 minuto para el almacenamiento del pre-evento es inadmisibles. Por el motivo señalado, se decidió manejar dos archivos de texto, uno correspondiente a la información conjunta de eventos y post-eventos y otro archivo correspondiente a los pre-eventos, de esta manera, el sistema hace un uso eficiente del tiempo y la información se presenta de manera organizada.

En el apartado 3.3.4 se explicó el proceso de almacenamiento de los datos de pre-evento en las localidades de memoria de la RAM, no obstante, se retomará el tema para comprender mejor la rutina de transferencia de datos. En la figura 3.49 se observa la organización de la memoria RAM utilizada de 1Mbit de capacidad, también se indica con un recuadro rojo, azul y verde, las localidades de memoria necesarias para almacenar una muestra del canal 1, canal 2 y canal 3, respectivamente. Para una muestra en un canal se necesitan 3 localidades de memoria, para los tres canales se necesitan 9 localidades de memoria. La memoria RAM cuenta con 131072 localidades de memoria, con una longitud de 8 bits por localidad, sin embargo, sólo se requieren 90009 ( $50\text{s} * 200\text{mps} * 3\text{bytes} * 3\text{canales} + 9\text{ bytes}$  correspondientes a las muestras en  $t = 0$  [s]) de ellas para almacenar 50 segundos de pre-evento. En la sección correspondiente al almacenamiento de

datos en la memoria RAM, la variable que maneja el valor de la localidad de memoria es reiniciada a cero una vez que se llegue a la localidad  $0x15F98$ .

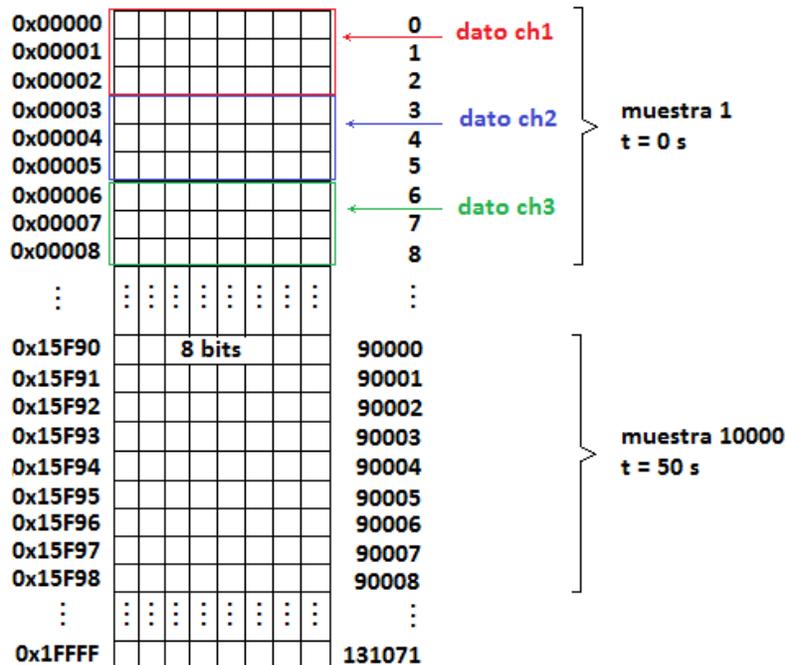


Figura 3.49. Organización de la memoria RAM.

La figura 3.50 ejemplifica el almacenamiento de datos en la memoria RAM al realizar una prueba durante los primeros 20 segundos. A la izquierda se muestra la información almacenada en las localidades de memoria y a la derecha, la variación de las señales correspondientes a estos valores. En la misma figura se muestran con líneas rojas, azules y verdes, los voltajes de disparo correspondientes a los canales 1, 2 y 3, respectivamente; al sobrepasar estos voltajes el sistema debe comenzar a almacenar datos en la memoria microSD. Por último, se debe hacer notar que las localidades de memoria posteriores a la  $0x08C48$  no contienen información aún, en la figura 3.51 por el contrario, las localidades de memoria han sido llenadas, en este momento han pasado 50 segundos de prueba.

La figura 3.52 ejemplifica la misma prueba, esta vez ya han pasado 90 segundos desde el inicio de operación del sistema, puesto que únicamente se requieren 50 segundos de pre-evento, al llegar a 50 [s] en la figura 3.51, los datos obtenidos posteriormente, serán sobrescritos en las localidades de memoria iniciales (en la figura 3.52 estos datos se representan en rojo). Pasados 90 [s] de la prueba, o equivalentemente, 40 segundos después de regresar a la primer localidad de memoria, se toma una lectura de 41.3 [mV] en el canal 1, en este momento el almacenamiento de datos en la memoria RAM concluye. La estructura de datos que toma la memoria RAM es una cola circular, de manera que el primer Byte corresponde al almacenado en la localidad  $11949_{Hex}$  y el último Byte corresponde al almacenado en la localidad  $11948_{Hex}$ .

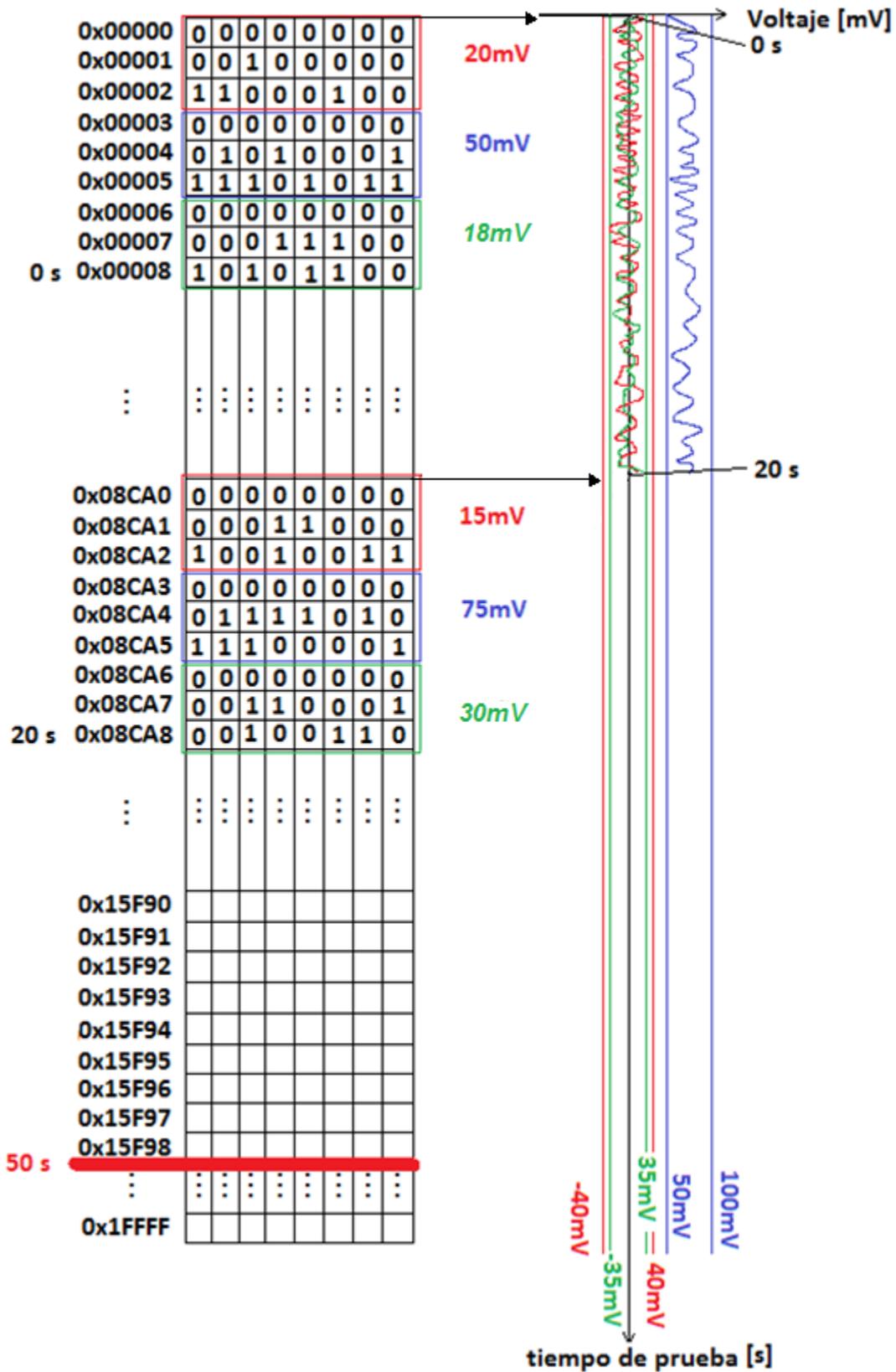


Figura 3.50. Almacenamiento de datos en la RAM. Tiempo de prueba = 20 [s].

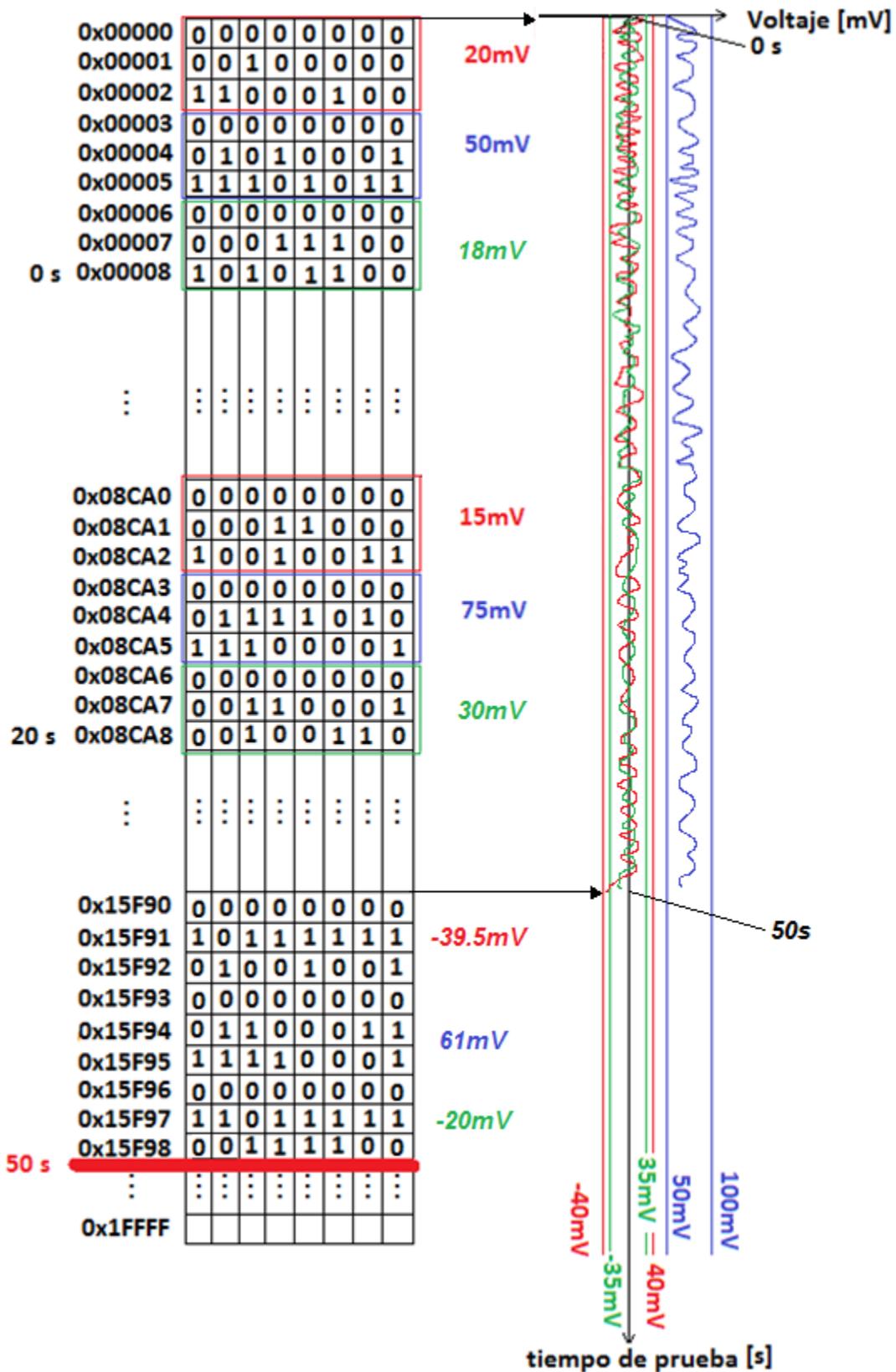


Figura 3.51. Almacenamiento de datos en la RAM. Tiempo de prueba = 50 [s].

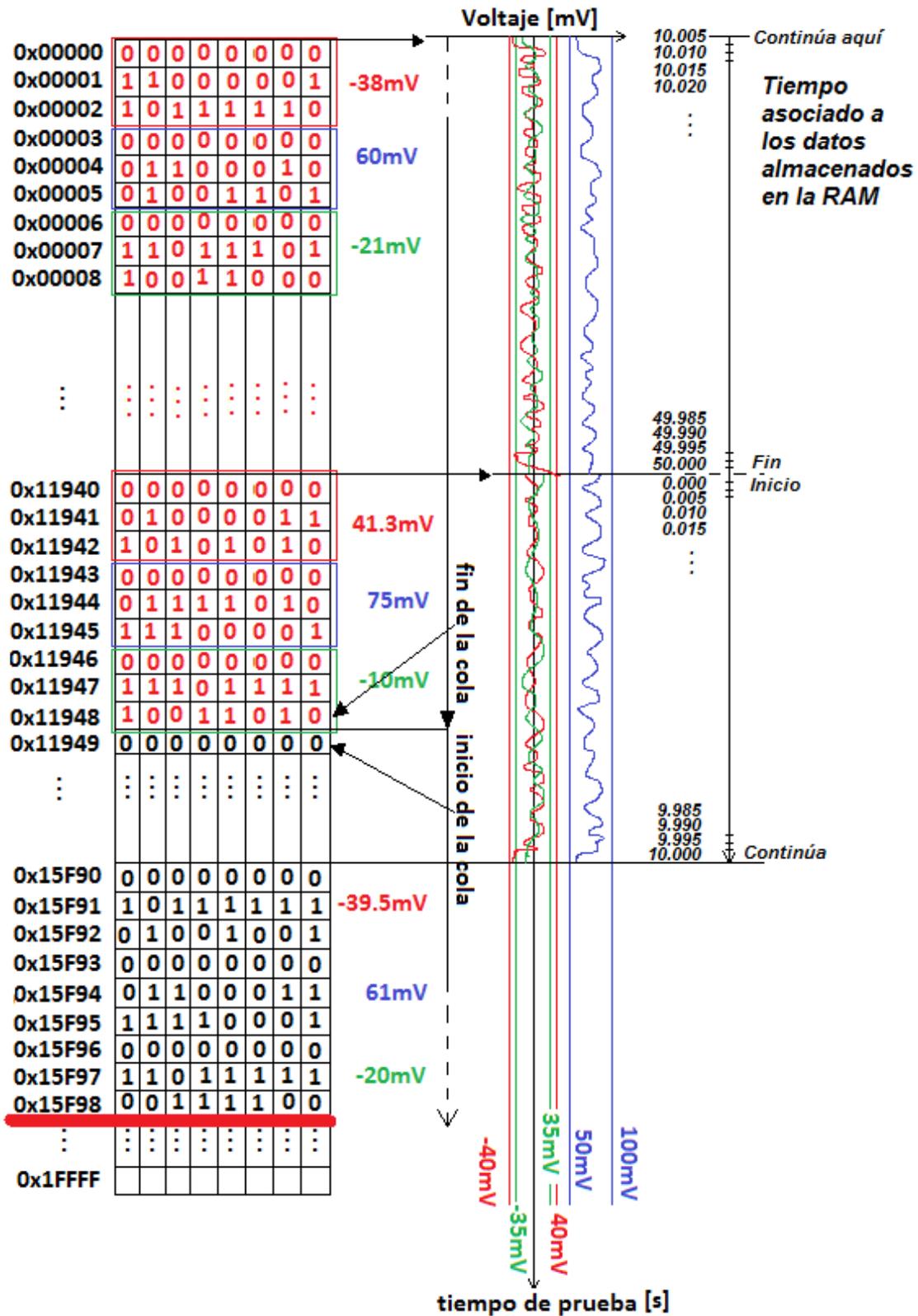


Figura 3.52. Almacenamiento de datos en la RAM. Tiempo de prueba = 90 [s].

Una vez aclarado este tema, el algoritmo utilizado para transferir la información de la memoria RAM a la memoria microSD será retomado. En el diagrama de flujo de la figura 3.53, se muestra el proceso realizado para cumplir con la tarea mencionada; en primer lugar, a las variables locales *addres* y *aux* se les asigna el valor de la variable global *memory\_location* de tipo *long*. La variable global es utilizada para direccionar la memoria durante el almacenamiento de datos en la RAM. En el último acceso a memoria de pre-evento, después de almacenar las tres muestras presentadas en el momento del disparo, la variable adquiere el valor de la localidad de memoria siguiente, el valor *0x11949*, en el ejemplo de la figura 3.52.

Después de realizar las asignaciones a las variable *addres* y *aux*, se crea el segundo archivo de texto para el almacenamiento de pre-eventos. Después de abrir el archivo, la transferencia de los datos se lleva a cabo, esta tarea se divide en dos partes; la primer parte consiste en iniciar en la localidad que indica la variable *addres*, se recorren las localidades de memoria a partir de este valor hasta llegar al valor *0x15F98*. Esto se logra mediante la siguiente secuencia: se habilita la memoria RAM, se envía el comando de modo de lectura de la RAM, se envía el valor de la localidad de memoria, se lee la información contenida en la RAM correspondiente a las lecturas de los tres canales, se desactiva la memoria RAM, se activa la memoria microSD, se escriben los tres valores en el archivo de texto contenido en la memoria microSD, se desactiva la memoria microSD, se incrementa en 9 unidades la variable *memory\_location* y el proceso continua hasta alcanzar el límite de la RAM establecido.

La segunda parte consiste en un proceso que requiere la secuencia antes indicada, la diferencia recae en el recorrido de las localidades de memoria. En este segundo proceso se comienza desde la primera localidad de memoria *0x00*, realizando la secuencia de transferencia de datos de una memoria a la otra, aumentando en 9 unidades la variable *memory\_location* hasta llegar al valor contenido en *aux*; en el ejemplo de la figura 3.52, éste es el valor *0x11948*. Una vez que la RAM se ha recorrido con estos dos procesos, la información correspondiente al pre-evento se encuentra almacenada en la memoria principal, conteniendo 50 segundos de información de la señal bajo estudio antes del sobrepaso de algún voltaje de disparo.

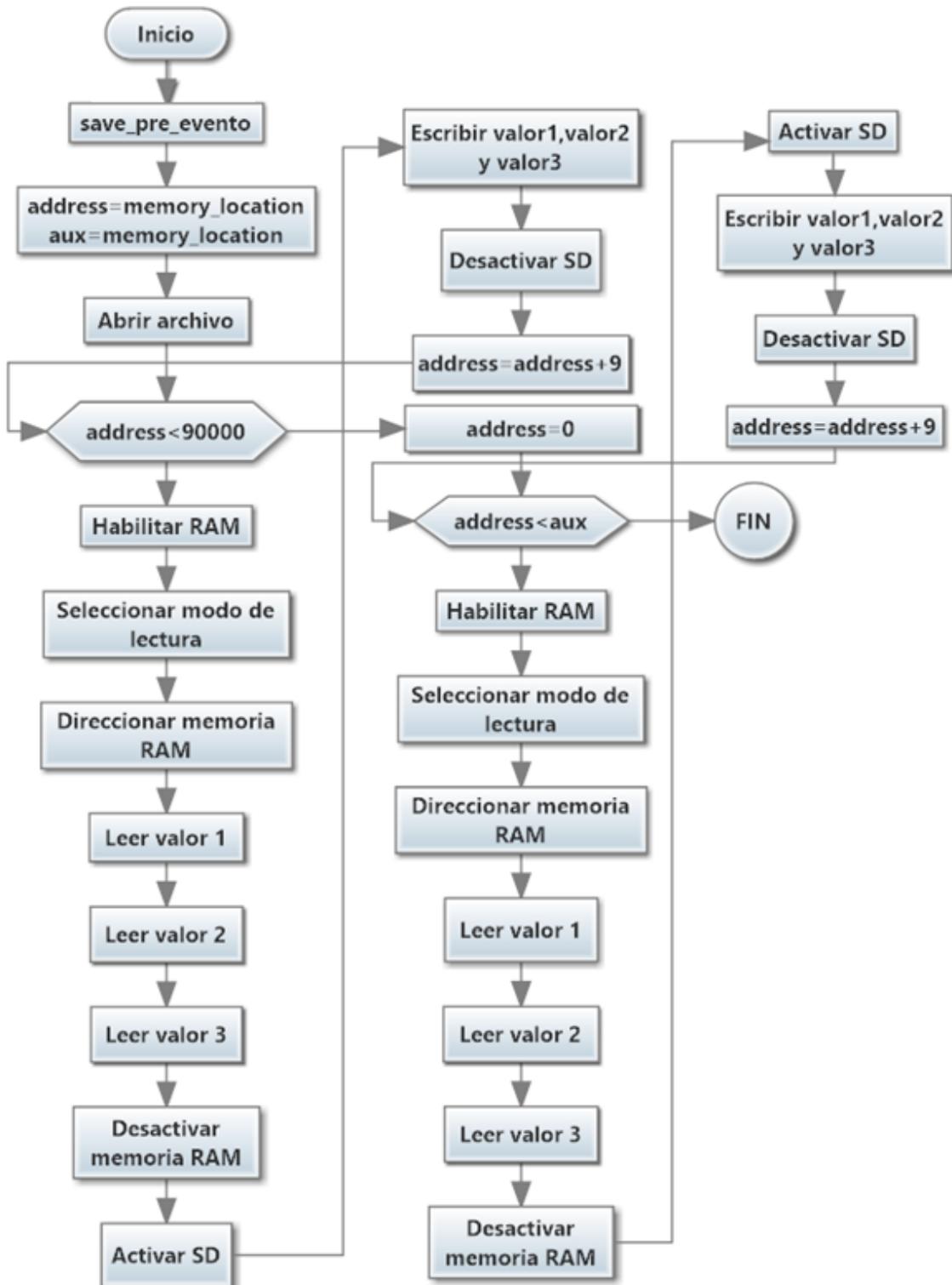


Figura 3.53. Diagrama de flujo de la rutina de transferencia de datos.

### 3.3.7. Adquisición de fecha y hora

Como se ha mencionado en el apartado 3.2.6, el receptor GPS ofrece distintos mensajes de salida correspondientes al protocolo NMEA. Uno de los mensajes de salida es el RMC el cual contiene información referente a la hora UTC, fecha, posición y velocidad. El mensaje RMC cuenta con el siguiente formato:

**\$GPRMC,161229.487,A,3723.2475,N,12158.3416,W,0.13,309.62,120598,,,A\*10**

El mensaje RMC tiene distintos campos separados por comas, el significado de cada uno de ellos se indica en la tabla 3.8.

Campo	Ejemplo	Unidades	Descripción
Identificador del mensaje	\$GPRMC		Cabecera del protocolo RMC
Hora UTC	161229.487		hhmmss.sss
Estado	A		A=dato válido V=dato inválido
Latitud	3723.2475		ddmm.mmmm
Indicador Norte/Sur	N		N=norte S=sur
Longitud	12158.3416		dddmm.mmmm
Indicador Este/Oeste	W		E=este W=oeste
Velocidad sobre la Tierra	0.13	Knots	
Curso sobre la Tierra	309.62	Grados	True
Fecha	120598		ddmmaa
Variación Magnética		Grados	E=este W=oeste
Indicador Este/Oeste	E		E=este
Modo	A		A=Autónomo
Suma de verificación	*10		
Retorno de carro <CR> y salto de línea <LF>			Fin del mensaje

Tabla 3.8. Formato del mensaje RMC.

Los caracteres son enviados desde el receptor GPS al microcontrolador utilizando la comunicación serial mediante UART, de este modo, la cadena consiste de una trama de caracteres que pueden ser almacenados en una variable de tipo *char* para su posterior uso. Para almacenar la fecha y la hora se utilizaron dos arreglos de 6 elementos; el primero, nombrado *time*, almacena los caracteres correspondientes a la hora; el segundo arreglo, nombrado *date*, almacena la fecha.

La figura 3.54 muestra el diagrama de flujo de la rutina de adquisición de la fecha y hora; en primer lugar se inicializan las variables y el arreglo *comandoGPR* se carga con la cadena "GPRMC", inmediatamente, se accede a un ciclo de repetición que se realiza hasta que la variable booleana *finish* tome un valor verdadero. En el ciclo de repetición, se pregunta si existe un caracter disponible en la línea de entrada de datos, si es así, se lee el caracter y se pregunta si se trata del caracter "\$".

Al obtener el caracter "\$", algún mensaje de salida está siendo enviado, para saber si se trata del mensaje RMC, se comprueba si las 5 letras consecutivas coinciden con la cadena almacenada en el arreglo *comandoRMC*, si es así, la variable *bien* aumenta de unidad en unidad hasta llegar al valor 5, en caso contrario, se regresa al inicio de la rutina. Una vez habiendo comprobado que el mensaje es el correspondiente al RMC, se tienen los caracteres correspondientes a la hora; mediante un ciclo de repetición se almacenan los 6 caracteres en el arreglo asociado a la hora.

Después de obtener la hora, el mensaje RMC envía 7 campos más que no son de interés, dado que los campos se encuentran separados por comas, leyendo los caracteres y contabilizando 8 comas ",", se llegará a la posición apropiada para leer los datos correspondientes a la fecha, nuevamente, mediante un ciclo de repetición, los caracteres son almacenados, esta vez en el arreglo asociado a la fecha. Al finalizar el almacenamiento de la fecha, la variable *finish* toma un valor verdadero y se termina con la rutina, por otro lado, si a lo largo de la ejecución de la rutina se pierde la secuencia correcta, el proceso se termina y se reinicializa.

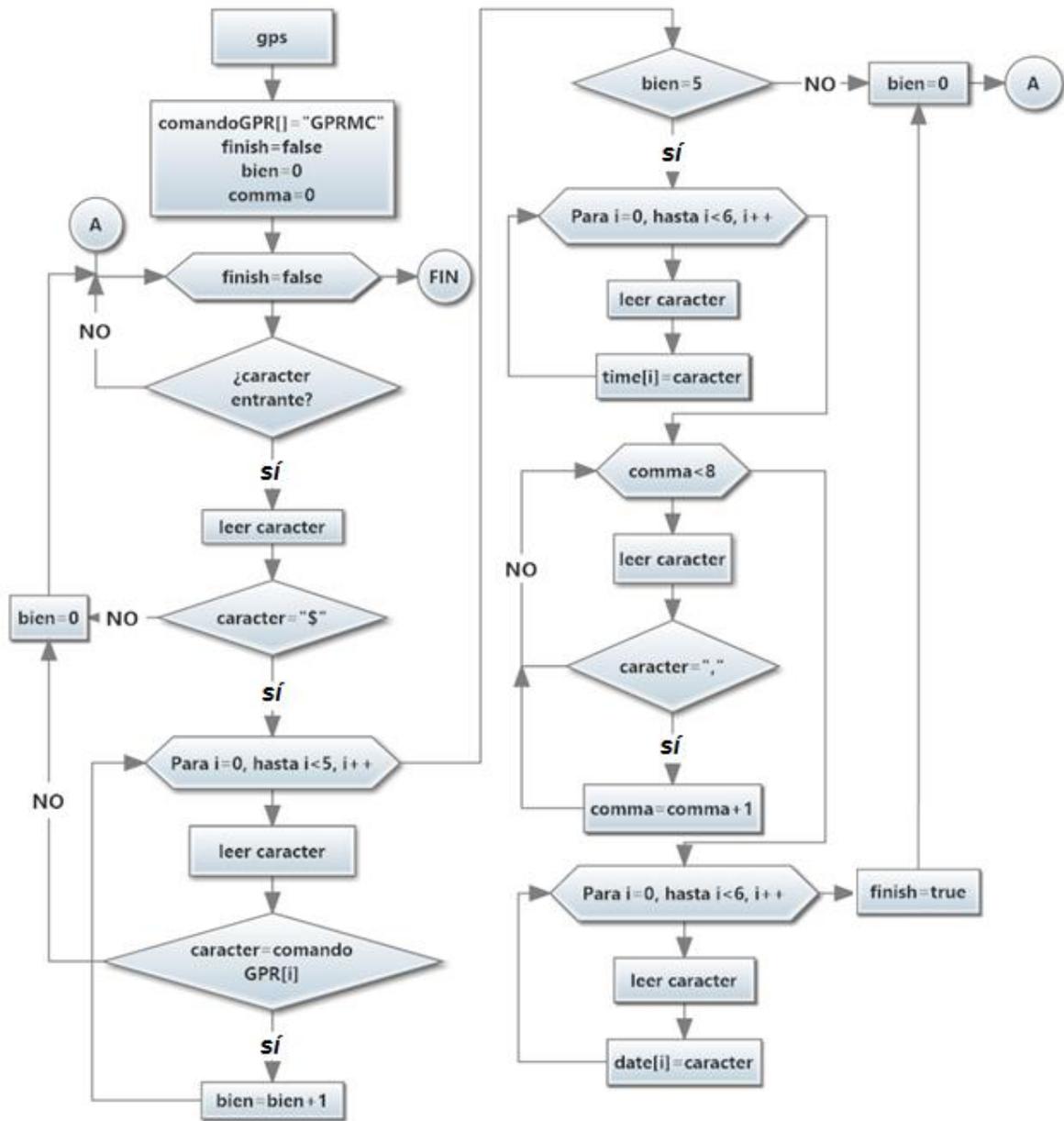


Figura 3.54. Diagrama de flujo de la rutina de adquisición de la fecha.

### 3.3.8. Manejo del joystick y navegación a través de menús

La manera en que un sistema electrónico recibe y proporciona información al usuario, es mediante una interfaz; en el caso del RAS-II, esta interacción se lleva a cabo mediante el *joystick* de mando y el LCD. En cuanto al *joystick*, este dispositivo permite al usuario seleccionar las opciones en los menús mediante 5 interruptores, los interruptores guardan una relación entre la dirección en que son presionados y la dirección en que se mueve el cursor que aparece en el LCD.

Las terminales reservadas para las posiciones del joystick son las siguientes:

- Tecla arriba
- Tecla abajo
- Tecla entrar
- Tecla izquierda
- Tecla derecha

Para detectar si el *joystick* ha sido movido hacia cualquier posición, se realiza una lectura del estado lógico de la tecla en cuestión. Si la tecla en cuestión tiene un valor lógico alto, indicará al microcontrolador que el *joystick* se ha movido en esa posición. Puesto que como todo interruptor, su funcionamiento depende de un desplazamiento mecánico, el sistema contempla un retardo de 20 [ms] para evitar una lectura errónea del estado lógico de la tecla. Cabe mencionar que el retardo mencionado fue obtenido al observar en un osciloscopio, la variación del estado lógico al presionar cualquiera de las teclas.

El manejo del *joystick* se relaciona directamente con los menús desplegados en el LCD. En cada menú se manejan tres variables: *opc*, *lo* y *hi*, estas asignaciones hacen alusión a las palabras opción, *hi* (alto) y *low* (bajo) respectivamente. A cada menú corresponde un número de opción, de la misma manera, cada menú tiene varias opciones; en el despliegue en el LCD la opción superior tiene el valor *hi* y la opción inferior el valor *lo*. A manera de ejemplo, se presenta el menú principal en la figura 3.55.



Figura 3.55. Menú principal.

En este caso, *opc* tiene el valor 1, *hi* tiene el valor 1 y *lo* tiene el valor 3. Si el usuario presiona la tecla *entrar*, el sistema ejecutará el menú 1, si el usuario presiona la tecla *abajo*, la variable *opc* aumentará en una unidad, quedando ahora con el valor 2; si en ese momento se presiona la tecla *entrar* el sistema mostrará el menú 2. Siempre que se presione la tecla en la posición *abajo* la variable *opc* aumentará en una unidad, por el contrario, siempre que se presione la tecla *arriba* disminuirá en una unidad. Las variables *lo* y *hi* sirven como límites para la asignación de estos incrementos, en el momento en que se rebase el valor máximo *hi* (en el ejemplo, el valor 3) *opc* nuevamente tomará el valor 1, en el otro caso, cuando se tenga un valor menor a *lo*, la variable *opc* tomará el valor máximo. Se trata de un conteo circular que garantiza tener únicamente opciones válidas para ejecutar los menús. El diagrama de flujo de la figura 3.56 resume el manejo del *joystick*.

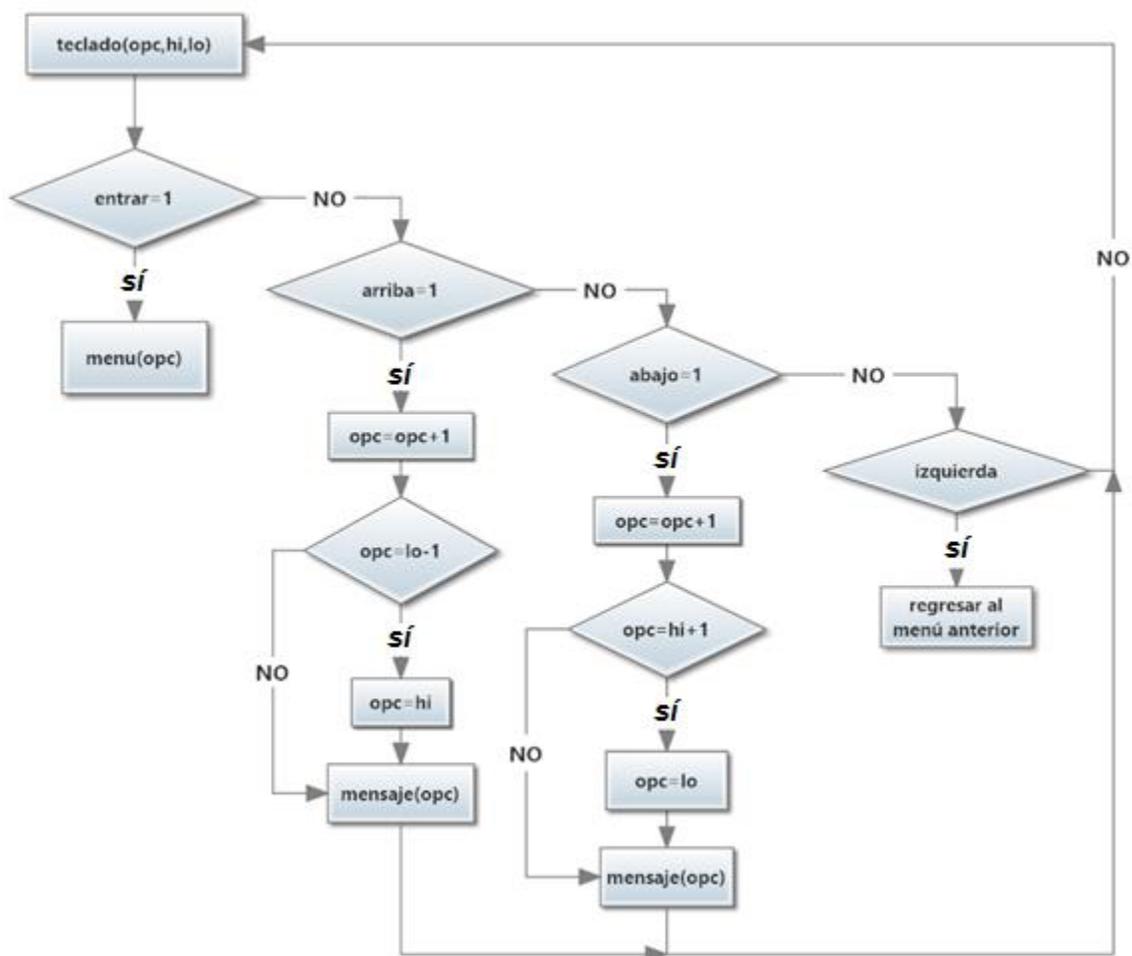


Figura 3.56. Diagrama de flujo del manejo del joystick.

---

# Capítulo 4.

# Integración del sistema

Este capítulo ilustra el punto cumbre del diseño electrónico realizado, se describirá el diseño de la placa de circuito impreso (PCB), la integración de los componentes electrónicos en dicha placa y las pruebas realizadas al circuito electrónico del sistema. El capítulo se divide en dos fases; en la primer fase se muestran las pruebas realizadas al sistema DIGI, este sistema corresponde a los antecedentes tangibles del RAS-II. En la segunda fase se describe la integración del RAS-II y las pruebas realizadas al mismo.

## 4.1. Fase 1

El sistema DIGI, señalado en el apartado 3.1, representa los antecedentes en la elaboración de este proyecto. Como todo proyecto electrónico, el registrador de aceleraciones sísmicas es el resultado de mejoras constantes al diseño original, no obstante, es común partir de una base que conserva su esencia a lo largo de todas las modificaciones realizadas.

En principio, el sistema DIGI no cubría al cien por ciento las necesidades de los usuarios, en consecuencia, se realizaron modificaciones en hardware y en software de modo que esas necesidades fueran cubiertas. Los elementos de hardware más importantes añadidos al sistema original son el reloj de tiempo real y la memoria de pre-evento. En cuanto al software, se re-diseñaron los algoritmos para la operación del equipo.

Precisando el funcionamiento del DIGI, este sistema es capaz de digitalizar señales provenientes de sensores pasivos como los geófonos GS-11D de la compañía Geospace, también es capaz de proporcionar una interfaz con el usuario para modificar la frecuencia de muestreo y graficar las señales en el LCD. La principal diferencia entre el sistema DIGI y el sistema RAS-II, es que el primero realiza un almacenamiento continuo de los datos en la memoria microSD, en donde el tiempo de la prueba es programado por el usuario.

Debido a que la operación se realiza en modo continuo, la información en el sistema DIGI se realiza en binario requiriendo un software diseñado para procesar los datos almacenados en la memoria microSD.

La ausencia de una memoria de pre-evento en el sistema DIGI es otra diferencia entre estos dos equipos. La memoria de pre-evento es indispensable en los registradores de aceleraciones contemporáneos pues juega un papel importante una vez que se analizan los datos recabados durante un fenómeno sísmico; al contar con información previa, se puede estudiar la naturaleza de las ondas y algunos de sus parámetros durante los movimientos telúricos.

En el proceso de desarrollo del sistema RAS-II se trabajó en primer lugar con el sistema DIGI, para ello se realizaron modificaciones al software y adaptaciones de circuitos integrados a la placa de circuito impreso disponible, finalmente, el prototipo proporcionó las características solicitadas por los usuarios y el equipo fue puesto a prueba. En los puntos siguientes se mostrarán las pruebas realizadas al prototipo; esta etapa representa la primera fase del proyecto. El prototipo del sistema RAS-II se muestra en la figura 4.1.



Figura 4.1. Prototipo del sistema RAS-II.

#### 4.1.1. Pruebas al circuito de acondicionamiento de señal

En la sección 3.2.1 se abordó el análisis del circuito de acondicionamiento de señal, como se mencionó, las señales provenientes de los acelerómetros necesitan ser procesadas de tal manera que los niveles de voltaje sean adecuados a la entrada del convertidor analógico-digital. Las señales provenientes de los acelerómetros DCA333 y DSA1 tienen un rango de voltaje de  $\pm 2.5$  [V]. Debido a que el sistema será alimentado únicamente con una sola fuente de polarización, a la señal original

se le suma una componente de directa de 2.5 [V], de modo que la variación de la señal se encuentre en el rango de 0 a 5 [V] con una tierra virtual de 2.5 [V].

El circuito de acondicionamiento de señal para sensores activos se muestra en la figura 4.2 y las expresiones para  $V_A$ ,  $V_B$  y el voltaje diferencial  $V_o = V_A - V_B$ , están dadas por las ecuaciones (4.1), (4.2) y (4.3), asumiendo  $R_1 = R_1'$ ,  $R_2 = R_2'$ ,  $C_1 = C_1'$ ,  $R_3 = R_3'$ ,  $R_4 = R_4'$  y  $C_2 = C_2'$ .

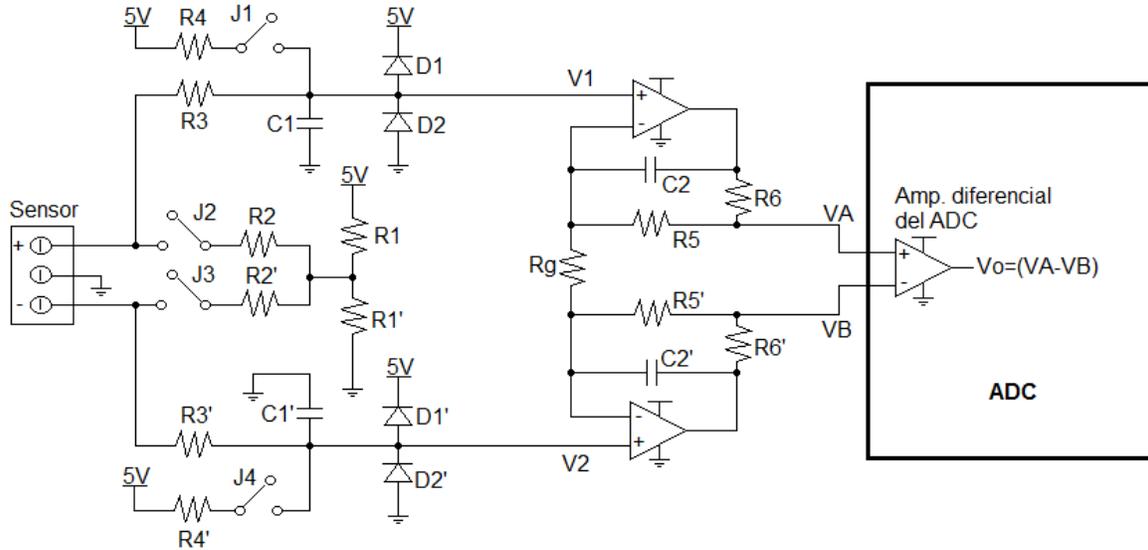


Figura 4.2. Circuito de acondicionamiento de señal para sensores activos.

$$V_A = \left( \frac{R_3 + R_4}{R_3 R_4} \right) \left( \frac{S + \frac{R_3 + R_g}{C_2 R_g (R_3 + R_4)}}{S^2 + S \left( \frac{R_3 + R_4}{R_3 R_4} \right) + \frac{1}{C_2 R_3 R_4}} \right) V_1 - \left( \frac{1}{C_2 R_4 R_g} \right) \left( \frac{1}{S^2 + S \left( \frac{R_3 + R_4}{R_3 R_4} \right) + \frac{1}{C_2 R_3 R_4}} \right) V_2 \quad (4.1)$$

$$V_B = - \left( \frac{1}{C_2 R_4 R_g} \right) \left( \frac{1}{S^2 + S \left( \frac{R_3 + R_4}{R_3 R_4} \right) + \frac{1}{C_2 R_3 R_4}} \right) V_1 + \left( \frac{R_3 + R_4}{R_3 R_4} \right) \left( \frac{S + \frac{R_3 + R_g}{C_2 R_g (R_3 + R_4)}}{S^2 + S \left( \frac{R_3 + R_4}{R_3 R_4} \right) + \frac{1}{C_2 R_3 R_4}} \right) V_2 \quad (4.2)$$

$$V_o = V_A - V_B = \left( \frac{R_3 + R_4}{R_3 R_4} \right) \left( \frac{S + \frac{2R_3 + R_g}{C_2 R_g (R_3 + R_4)}}{S^2 + S \left( \frac{R_3 + R_4}{R_3 R_4} \right) + \frac{1}{C_2 R_3 R_4}} \right) (V_1 - V_2) \quad (4.3)$$

Para obtener las expresiones que proporcionan el valor de ganancia a bajas frecuencias, que de acuerdo al diseño realizado, el comportamiento en frecuencia de los componentes utilizados no tiene ningún efecto en la respuesta en frecuencia, se puede considerar  $S \rightarrow 0$ . De esta manera las expresiones anteriores pueden reescribirse de la siguiente manera:

$$V_A = \left( 1 + \frac{R_3}{R_g} \right) V_1 - \left( \frac{R_3}{R_g} \right) V_2 \quad (4.4)$$

$$V_B = -\left(\frac{R_3}{R_g}\right)V_1 + \left(1 + \frac{R_3}{R_g}\right)V_2 \quad (4.5)$$

$$V_o = V_A - V_B = \left(1 + \frac{2R_3}{R_g}\right)(V_1 - V_2) \quad (4.6)$$

A su vez,  $V_1$  y  $V_2$  están dados por:

$$V_1 = \frac{1}{C_1 R_1} \left( \frac{1}{S + \frac{R_1 + R_2}{R_1 R_2 C_1}} \right) v_{S1} + \frac{R_1}{R_1 + R_2} V_{CC} \quad (4.7)$$

$$V_2 = \frac{1}{C_1 R_1} \left( \frac{1}{S + \frac{R_1 + R_2}{R_1 R_2 C_1}} \right) v_{S2} + \frac{R_1}{R_1 + R_2} V_{CC} \quad (4.8)$$

Del mismo modo, considerando el comportamiento en frecuencia cuando  $S \rightarrow 0$  y se tiene una respuesta plana, las ecuaciones (4.7) y (4.8) se pueden reescribir de la manera siguiente:

$$V_1 = \frac{R_2}{R_1 + R_2} v_{S1} + \frac{R_1}{R_1 + R_2} V_{CC} \quad (4.9)$$

$$V_2 = \frac{R_2}{R_1 + R_2} v_{S2} + \frac{R_1}{R_1 + R_2} V_{CC} \quad (4.10)$$

Considerando  $R_3 = \frac{1}{2}R_g$  y  $R_1 = R_2$ , además, sustituyendo (4.9) y (4.10) en (4.4) a (4.6):

$$V_A = 0.75v_{S1} - 0.25v_{S2} + 0.5V_{CC} \quad (4.11)$$

$$V_B = -0.25v_{S1} + 0.75v_{S2} + 0.5V_{CC} \quad (4.12)$$

$$V_o = V_A - V_B = 2(0.5v_{S1} - 0.5v_{S2}) \quad (4.13)$$

Finalmente, al conectar los acelerómetros, cuya salida es unipolar,  $v_{S1}$  es la salida del canal bajo prueba, mientras que  $v_{S2}$  es conectada a tierra, de esta manera, las ecuaciones (4.11) a (4.13) quedan dadas por:

$$V_A = 0.75v_{S1} + 0.5V_{CC} \quad (4.14)$$

$$V_B = -0.25v_{S1} + 0.5V_{CC} \quad (4.15)$$

$$V_o = V_A - V_B = v_{S1} \quad (4.16)$$

#### *Prueba de ganancia*

En esta prueba se verifica que los voltajes experimentales  $V_A$ ,  $V_B$  y el voltaje diferencial de entrada al ADC  $V_o = V_A - V_B$ , correspondan con los valores de

voltaje teóricos dictados por las ecuaciones (4.14) a (4.16), para ello, se trabajó con una señal de baja frecuencia en donde la respuesta no es alterada (teóricamente) de acuerdo a la respuesta en frecuencia diseñada para el circuito de acondicionamiento de señal.

Utilizando el canal 1 del prototipo RAS-II (figura 4.3), a este canal se le proporcionó una señal senoidal de amplitud igual a 1 [V] pico y una frecuencia de 1 [Hz], esta señal corresponde al voltaje indicado como  $v_{s1}$  en el circuito de la figura 4.2. La terminal correspondiente a la conexión de la señal  $v_{s2}$  fue conectada a tierra ya que se trabajó con una señal sencilla y no diferencial, al realizar esto, se tiene que  $v_{s2} = 0$  [V]. Las señales  $v_{s1}$  y  $v_{s2}$  se muestran en amarillo y azul respectivamente en la figura 4.4.



Figura 4.3. Sistema bajo prueba.

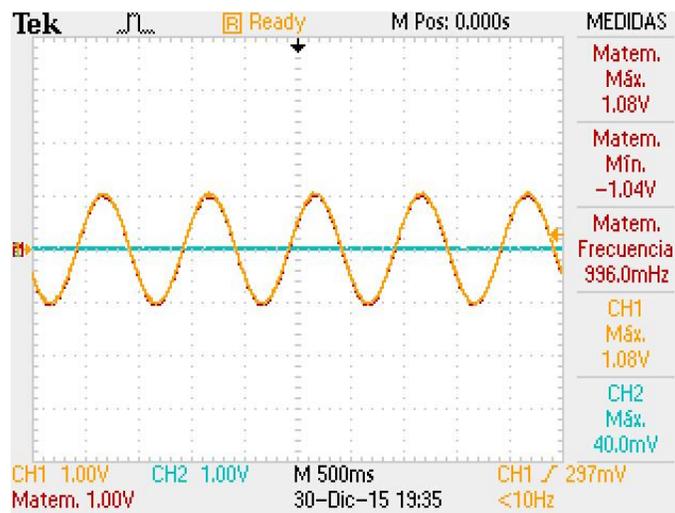


Figura 4.4. Señales  $v_{s1}$  y  $v_{s2}$ .

En la parte derecha de la figura 4.4 se pueden observar distintas mediciones asociadas a las señales graficadas. El primer valor, que es 1.08 [V], corresponde al valor máximo de la señal de entrada  $v_{s1}$ , con dicho valor y utilizando las ecuaciones (4.14) y (4.15) se puede predecir la magnitud teórica de los voltajes  $V_A$  y  $V_B$ . Para el caso de  $V_A$ , considerando que  $V_{CC} = 5$  [V], se tiene:

$$V_{A_{teórico}} = 0.75v_{s1} + 0.5V_{CC} = 0.75(1.08[V]) + 0.5(5 [V]) = 3.31V$$

Para  $V_B$ , el signo negativo en la ecuación (4.15) puede interpretarse como un cambio de fase de  $180^\circ$  con respecto a la señal original, tomando por el momento únicamente el valor absoluto de  $v_{s1}$  para conocer el voltaje máximo de  $V_B$  se tiene:

$$V_{B_{teórico}} = |-0.25v_{s1}| + 0.5V_{CC} = |-0.25(1.08[V])| + 0.5(5[V]) = 2.77V$$

Para el voltaje diferencial  $V_o$ , de acuerdo a la ecuación (4.16), el voltaje debe ser prácticamente el mismo que el de la señal de entrada, es decir, 1.08 [V].

### Resultados

En la figura 4.5 se pueden observar las señales  $V_A$  (en color amarillo) y  $V_B$  (en color azul), por otro lado, debido a que no es posible obtener el voltaje diferencial  $V_o$  en el circuito, dicho voltaje se obtuvo con ayuda del osciloscopio utilizando la función matemática de sustracción del canal 1 menos el canal 2. El voltaje diferencial  $V_o$  se muestra en rojo en la misma figura.

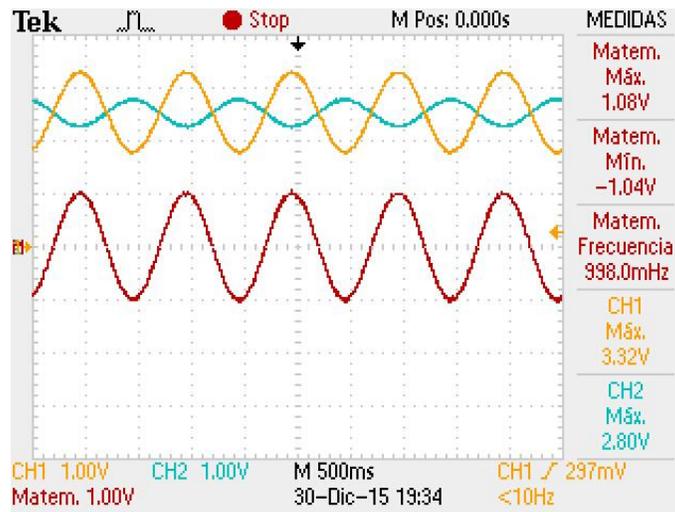


Figura 4.5. Voltajes  $V_A$ ,  $V_B$  y  $V_o$ .

Los valores de voltaje experimentales para  $V_A$  y  $V_B$  se pueden obtener del cuadro de medidas de la figura 4.5, se tiene entonces que  $V_{A_{experimental}} = 3.32$  [V] y  $V_{B_{experimental}} = 2.8$  [V]. Con estos valores y utilizando los voltajes teóricos

calculados previamente se pueden obtener los porcentajes de error para las señales VA y VB.

$$\%e_{VA} = \frac{|VA_{teórico} - VA_{experimental}|}{VA_{teórico}} \times 100 = \frac{|3.31 - 3.32|V}{3.31V} \times 100 = 0.3\%$$

$$\%e_{VB} = \frac{|VB_{teórico} - VB_{experimental}|}{VB_{teórico}} \times 100 = \frac{|2.77 - 2.8|V}{2.77V} \times 100 = 1.08\%$$

Las expresiones (4.14), (4.15) y (4.16) proporcionan información acerca de la ganancia del circuito de acondicionamiento de señal del RAS-II a bajas frecuencias en donde no se afecta la amplitud de la señal de acuerdo al diseño realizado. La prueba realizada al circuito revela que el comportamiento teórico y el comportamiento real del mismo son muy parecidos, considerando los bajos porcentajes de error obtenidos.

En la prueba se observó también que la señal diferencial  $V_o = VA - VB$  es prácticamente la misma que la señal de entrada, cumpliendo con el factor de ganancia unitario bajo el cual fue diseñado el RAS-II.

El circuito de acondicionamiento de señal pasa la prueba de ganancia de voltaje, recordando que el factor de amplificación unitario permite el manejo de sensores cuyo voltaje de salida es igual a  $\pm 2.5$  [V], esta variación se realiza sobre una tierra virtual de 2.5 [V]. Por último, la utilización de la configuración del amplificador de instrumentación permite la adaptación de impedancias al acoplar los acelerómetros al ADC.

#### *Prueba al filtro anti-aliasing*

El filtro anti-aliasing tiene como objetivo limitar en banda las señales provenientes de los acelerómetros dejando pasar las señales de baja frecuencia y bloqueando las señales de frecuencia mayor que no son de interés para el análisis. Por especificaciones de los usuarios, la frecuencia de corte del filtro se estableció en 12 [Hz]. Debido a las prestaciones del convertidor analógico-digital sigma-delta utilizado, el filtro anti-aliasing queda reducido a un filtro pasivo de primer orden como el mostrado en la figura 4.6.

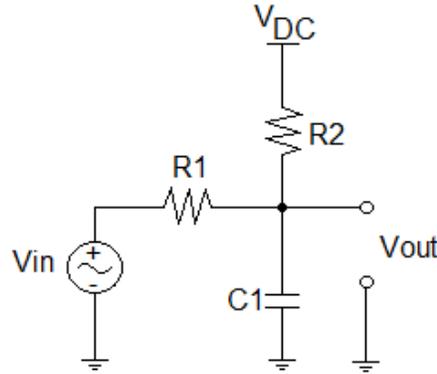


Figura 4.6. Filtro anti-aliasing.

El comportamiento en frecuencia del filtro mostrado puede deducirse de la siguiente expresión:

$$V_{out} = \left( \frac{R_1}{R_1 + R_2} \right) V_{CC} + \frac{1}{C_1 R_1} \left( \frac{1}{s + \frac{R_1 + R_2}{C_1 R_1 R_2}} \right) V_{in} \quad (4.17)$$

El primer término de la ecuación anterior describe la adición de 2.5 [V] de corriente directa, por otro lado, el segundo término corresponde a la respuesta de un filtro paso bajas aplicado a la señal  $V_{in}$ . De acuerdo a las consideraciones realizadas en el apartado 3.2.1, la frecuencia de corte del filtro está dada por:

$$f_{cteórica} = \frac{2}{2\pi C_1 R_1} [Hz] \quad (4.18)$$

Para el caso del prototipo RAS-II los valores de  $R_1$  y  $C_1$  llevan a la frecuencia de corte teórica de 11.76 [Hz].

Para describir el comportamiento teórico de los filtros, se deben observar las señales  $V_1$  y  $V_2$  señaladas en el circuito de la figura 4.2, estos voltajes están dados por las ecuaciones (4.19) y (4.20).

$$V_1 = 0.5V_{cc} + 36.95 \left( \frac{1}{s+73.90} \right) vs_1 = 2.5[V] + 36.95 \left( \frac{1}{s+73.90} \right) vs_1[V] \quad (4.19)$$

$$V_2 = 0.5V_{cc} + 36.95 \left( \frac{1}{s+73.90} \right) vs_2 = 2.5[V] + 0[V] \quad (4.20)$$

La terminal correspondiente a la entrada de señal  $vs_2$  fue conectada a tierra pues se trabajó con una señal sencilla y no diferencial, con esto, el voltaje  $V_2$  únicamente tendrá una componente de directa de 2.5 [V] (recordando que  $V_{cc} = 5$  [V]), teniendo en mente esto, sólo se observó la respuesta en frecuencia de la señal  $V_1$ .

La señal  $V_1$  se compondrá de un voltaje de 2.5 [V] de corriente directa y una respuesta paso bajas para la señal  $vs_1$  con frecuencia de corte teórica igual a 11.76 [Hz]. La ganancia de corriente directa es igual a 0.5 o -6.02 [dB], esta ganancia se

mantiene a bajas frecuencias antes de la frecuencia de corte del filtro. Al sobrepasar dicha frecuencia, el filtro debe tener teóricamente una caída de 20 [dB] por década.

Para observar el comportamiento en frecuencia del filtro anti-aliasing, se utilizó el canal 1 del prototipo RAS-II y un generador de funciones. Se introdujo al canal 1 una señal senoidal  $vs1$  de amplitud igual a 2 [V] pico y una frecuencia inicial de 1 [Hz]. En la figura 4.7 se muestra la señal de entrada  $vs1$  en color amarillo y la señal  $V1$  en color azul.

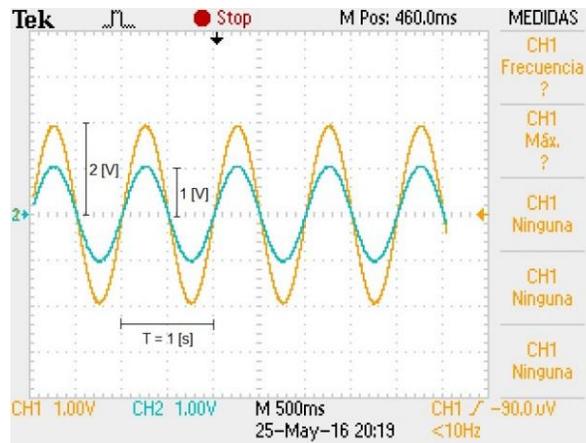


Figura 4.7. Señal  $vs1$  y señal  $V1$ .

La frecuencia de la señal de entrada fue modificada al mismo tiempo que se observó el comportamiento de la señal  $V1$ . En las figuras 4.8, 4.9, 4.10, 4.11, 4.12, 4.13, 4.14 y 4.15 se muestra la señal  $V1$  al variar la frecuencia a 2, 4, 8, 12.4, 16, 20, 40 y 124 [Hz] respectivamente.

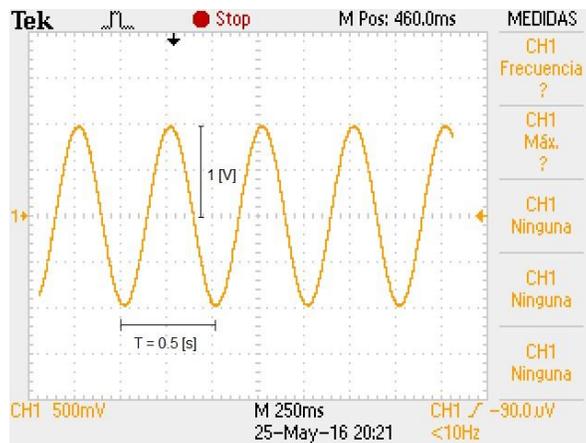


Figura 4.8. Respuesta a 2 [Hz].

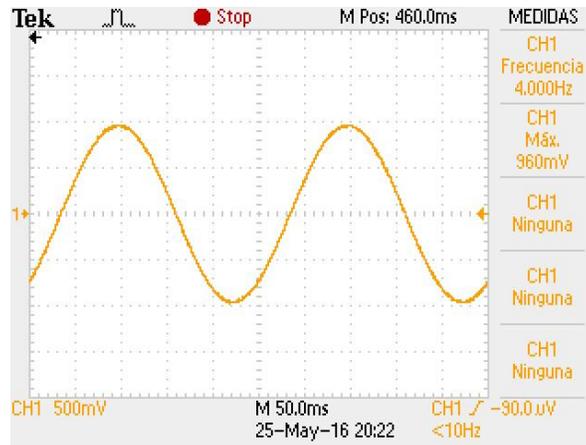


Figura 4.9. Respuesta a 4 [Hz].

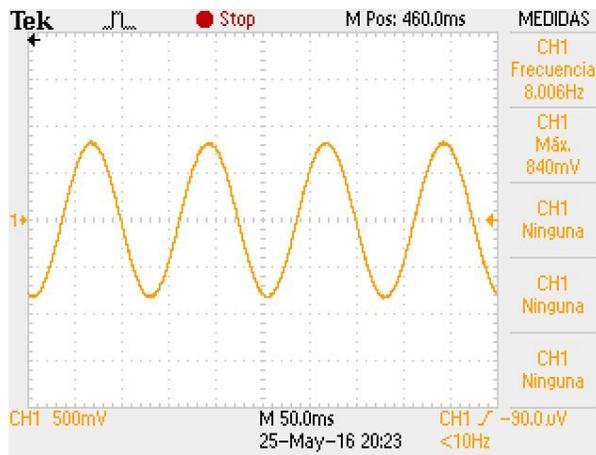


Figura 4.10. Respuesta a 8 [Hz].

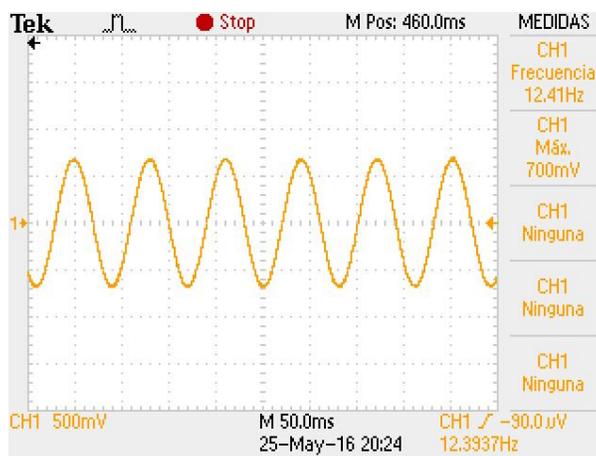


Figura 4.11. Respuesta a 12.4 [Hz].

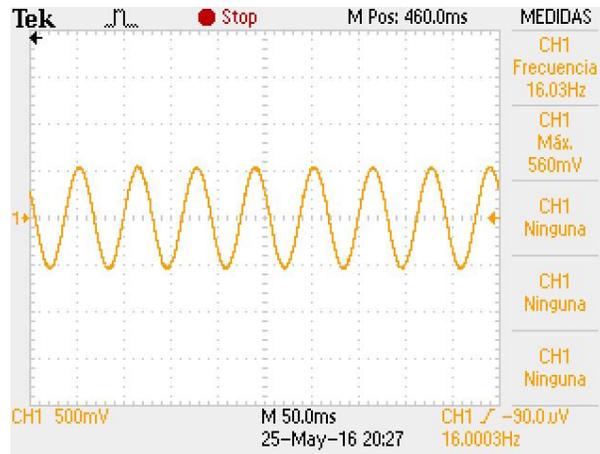


Figura 4.12. Respuesta a 16 [Hz].

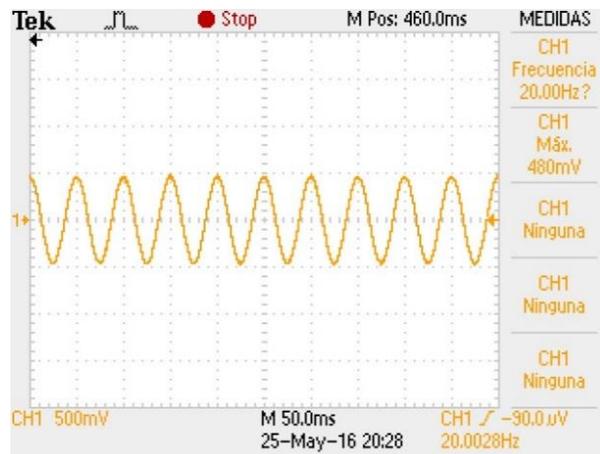


Figura 4.13. Respuesta a 20 [Hz].

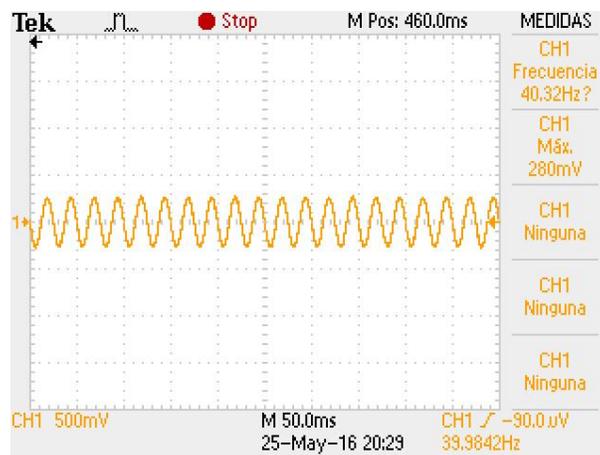


Figura 4.14. Respuesta a 40 [Hz].



frecuencias bajas la ganancia permanece prácticamente inalterada con una ganancia de 0.5 o -6.02 [dB], esto concuerda con la magnitud teórica esperada. También se puede notar que a una frecuencia de 12.4 [Hz] la magnitud de la señal cuenta con el 70% del valor original y al aumentar 10 veces la frecuencia (124 [Hz]), la señal ha caído 20 [dB] respecto a la magnitud inicial, esta atenuación se corresponde con la caída de 20 [dB] por década de un filtro paso bajas de primer orden, por lo tanto, se concluye que el filtro opera en condiciones óptimas.

#### 4.1.2. Reconstrucción de la señal muestreada

La manera de comprobar que el proceso de digitalización se ha realizado de la manera correcta es reconstruyendo la señal a partir de las muestras tomadas y compararla con la señal original. Al graficar en un eje coordenado el valor de las muestras contra el tiempo de referencia, es posible obtener algunos de los parámetros de la señal reconstruida. Se puede decir que la señal reconstruida es una copia fiel de la señal original, si su frecuencia y amplitud son iguales a los de la señal original.

El proceso de reconstrucción de la señal muestreada se realizó de la siguiente manera: empleando un generador de funciones, se introdujo una señal senoidal al canal 2 del prototipo RAS-II. La amplitud de la señal fue de 600 [mV] pico y su frecuencia igual a 12 [Hz], la señal generada se muestra en la figura 4.16.

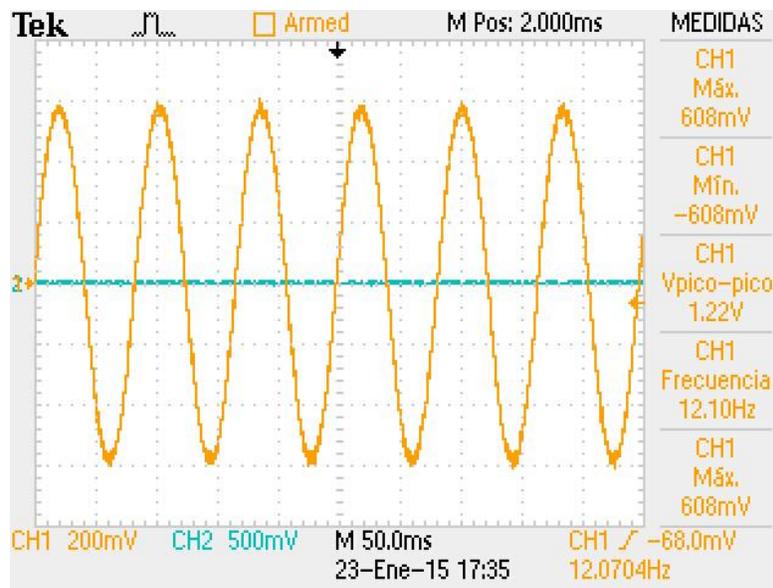


Figura 4.16. Señal original.

La señal fue digitalizada y registrada en la memoria microSD del sistema RAS-II. Una vez que se tuvieron los datos, estos se leyeron con una computadora. Utilizando el programa *Microsoft Excel* se graficaron los pares de datos en un plano cartesiano de voltaje contra tiempo, la gráfica correspondiente a la señal reconstruida a partir de las muestras se observa en la figura 4.17.

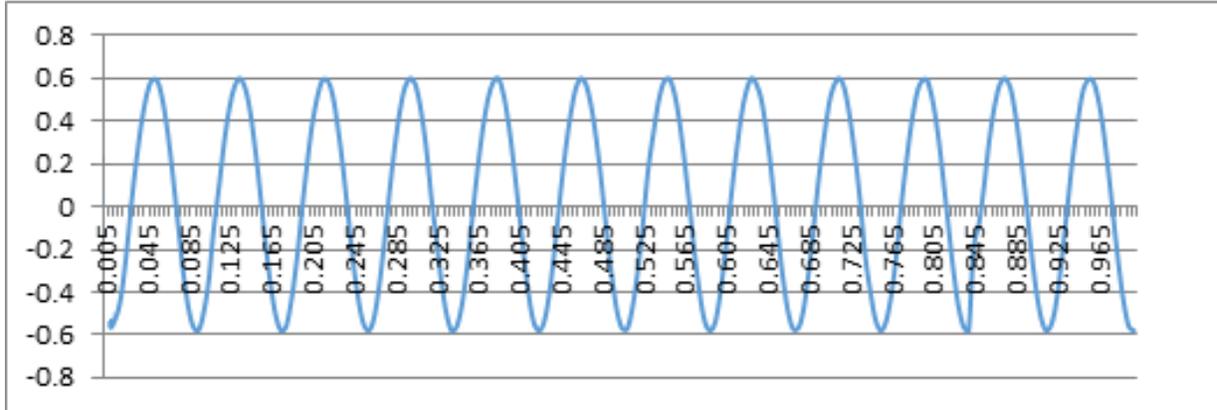


Figura 4.17. Señal reconstruida.

### Resultados

Analizando los datos del archivo de texto, se observaron algunos parámetros de la señal como el voltaje máximo, el voltaje mínimo y la frecuencia de la señal para corroborar las dos condiciones que garantizan la igualdad entre la señal reconstruida a partir de las muestras y la señal original. Estos datos se presentan a continuación.

### Amplitud

En el archivo de texto se identificaron los valores máximo y mínimo con ayuda del programa antes mencionado; estos valores son los siguientes:

$$V_{m\acute{a}x_{exp.}} = 600 [mV]$$

$$V_{m\acute{i}n_{exp.}} = -595 [mV]$$

En el cuadro de medidas de la figura 4.16 se pueden identificar los voltajes máximo y mínimo teóricos, éstos son:

$$V_{m\acute{a}x_{te\acute{o}rico}} = 608 [mV]$$

$$V_{m\acute{i}n_{te\acute{o}rico}} = -608 [mV]$$

Tomando como referencia los voltajes teóricos máximo y mínimo y utilizando los valores de voltaje experimentales, se pueden calcular los porcentajes de error para estos datos:

$$\%e_{V_{m\acute{a}x}} = \frac{|V_{m\acute{a}x_{te\acute{o}rico}} - V_{m\acute{a}x_{exp.}}|}{V_{m\acute{a}x_{te\acute{o}rico}}} \times 100 = \frac{|608 - 600|mV}{608mV} \times 100 = 1.3\%$$

$$\%e_{V_{\min}} = \frac{\left| |V_{\min_{\text{teórico}}}| - |V_{\min_{\text{exp}}}| \right|}{|V_{\min_{\text{teórico}}}|} \times 100 = \frac{\left| |-608mV| - |-595mV| \right|}{|-608mV|} \times 100 = 2.1\%$$

De acuerdo a los errores obtenidos para los voltajes máximo y mínimo de la señal, se puede concluir que el prototipo RAS-II cumple con la condición de amplitud, en donde la amplitud de la señal original y la correspondiente a la señal reconstruida son similares.

### *Frecuencia*

Para comparar la frecuencia de ambas señales, del archivo de texto se obtuvieron los siguientes datos:

*Tiempo de la primer muestra: 0 [s]*

*Tiempo de la última muestra: 1.0 [s]*

*Número de ciclos presentes: 12 [Hz]*

Por lo tanto, la frecuencia de la señal reconstruida es igual a 12 [Hz] que es igual a la de la señal original, entonces se concluye que el prototipo RAS-II cumple con la condición de frecuencia y en consecuencia, la señal reconstruida a partir de las muestras es una copia fiel de la señal original.

## **4.2. Fase 2**

Una vez concluida con la fase 1, correspondiente a las pruebas realizadas al prototipo DIGI, se continuó con el diseño de un nuevo sistema que incluyera todos los elementos de hardware agregados al diseño base, este nuevo sistema es el RAS-II.

Desde el punto de vista comercial, el sistema RAS-II constituye una versión actualizada y con prestaciones adicionales al sistema DIGI. RAS-II ofrece dos elementos de hardware importantes que constituyen sistemas de respaldo ante condiciones no previstas por el sistema DIGI, los elementos son el reloj de tiempo real que actúa ante la falta de sincronización por parte del receptor GPS y la memoria RAM que brinda información de interés en los instantes previos a la ocurrencia de un sismo. RAS-II también incluye una actualización en el software que en primer lugar representa la constante mejora al sistema con el afán de evolucionar a un equipo con las más altas prestaciones y competir en el área comercial o científica y, en segundo lugar, facilita la integración de nuevo código al programa al utilizar bibliotecas renovadas.

La fase 2 de este proyecto incluye el diseño de la tarjeta de circuito impreso, la integración del hardware y las pruebas realizadas al equipo electrónico. Estas pruebas se describen con mayor detalle a continuación.

#### 4.2.1. Diseño de la placa de circuito impreso.

Los elementos de hardware como: componentes pasivos, circuitos integrados y módulos utilizados dentro de un sistema electrónico se encuentran interconectados entre sí para proporcionar un funcionamiento de conjunto y solucionar una o varias tareas. Típicamente esta interconexión se realiza con ayuda de varias pistas conductoras trazadas sobre una tarjeta rígida de material no conductor que da soporte a las mismas pistas y a los componentes electrónicos integrados a ella. Esta tarjeta recibe el nombre de placa de circuito impreso (PCB, Printed Circuit Board) y es utilizada en la etapa final para la integración de un sistema electrónico.

La placa de circuito impreso constituye el circuito final del sistema electrónico y por lo tanto, al realizar el trazado de las pistas y la ubicación de cada componente, se deben realizar múltiples consideraciones para asegurar que el diseño realizado no afecte el funcionamiento eléctrico del sistema. Cuando la placa de circuito impreso se realiza sin tomar en cuenta pautas básicas de diseño, se pueden presentar comportamientos anormales al realizar la integración del hardware, algunos de los errores en el diseño de la placa de circuito impreso son: una mala distribución de las líneas de alimentación, poca anchura de las pistas conductoras, falta de capacitores de desacoplo cercanos a los circuitos digitales que manejan señales de alta frecuencia, una ubicación muy próxima entre componentes digitales y analógicos, entre otros.

Para el sistema RAS-II se consideraron algunas pautas de diseño mencionadas en el estándar IPC221 de la *IPC Association Connecting Electronics Industries* (originalmente fundada en 1957 como *Institute for Printed Circuits*) cuyo objetivo es la estandarización de los procesos de producción y ensamblaje en la industria electrónica. El documento IPC2221 proporciona información básica para el diseño de las placas de circuito impreso (las figuras demostrativas manejadas en este apartado fueron tomadas de este documento).

Para el diseño de la placa de circuito impreso del sistema RAS-II se tomaron en cuenta sólo algunas consideraciones básicas puesto que el documento trata temas de aplicación muy específica como: mecanismos de enfriamiento para dispositivos de elevado manejo de temperatura, especificación de adhesivos para componentes integrados a placas cuyo uso se someterá a un gran estrés mecánico, consideraciones eléctricas para el manejo de señales de radio-frecuencia, entre otros, que en el caso del RAS-II no tienen aplicación. El diseño se realizó utilizando el software *Altium Designer* optimizado para el diseño de placas de circuito impreso a nivel profesional.

Entre las pautas seguidas en el diseño de la PCB del RAS-II se encuentran las siguientes:

### Consideraciones de distribución de potencia

- Un factor predominantemente importante que debe ser considerado en el diseño de una placa de circuito impreso es la distribución de potencia. El esquema de un plano de tierra puede ser utilizado como parte de la distribución de potencia en el circuito, éste proporciona no solo un camino de retorno para las señales de DC sino también un plano de referencia para señales de AC de alta frecuencia acopladas al circuito. Con el uso de un plano de tierra los efectos del ruido circundante son disminuidos debido a que las corrientes eléctricas parásitas se conducirán mucho mejor hacia el punto de referencia de alimentación a través de superficies de cobre que muestren la menor impedancia posible. En las figuras 4.18 y 4.19 se muestran los planos de tierra de la PCB de dos capas del sistema RAS-II.

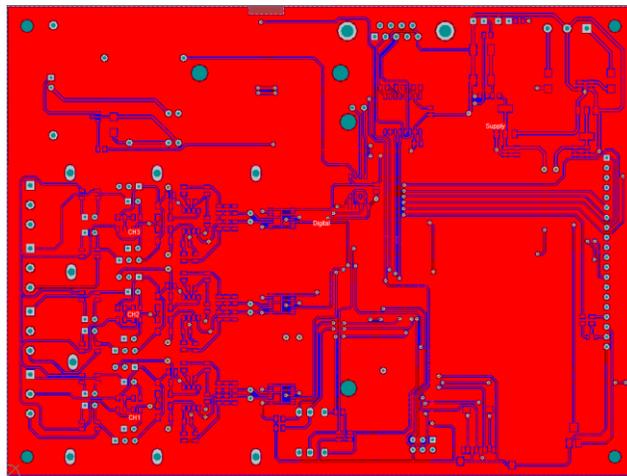


Figura 4.18. Plano de tierra de la capa superior (*top layer*).

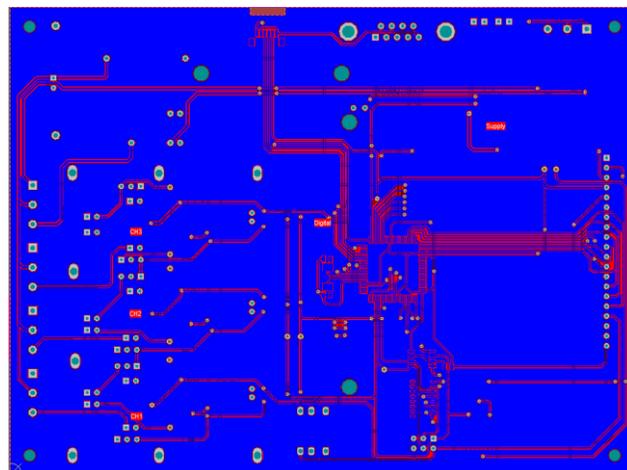


Figura 4.19. Plano de tierra de la capa inferior (*bottom layer*).

- Se debe desacoplar al sistema de la fuente de alimentación, utilizando para ello capacitores de desacoplo ubicados cercanamente al conector de alimentación de la placa. Se deben utilizar tantos capacitores de desacoplo

como sean necesarios, sobre todo para el caso de circuitos integrados que manejan señales digitales de elevada velocidad de conmutación entre estados; los capacitores deben distribuirse uniformemente en todas las áreas de la placa que requieran de su utilización. El uso de los capacitores de desacoplo tiene como objetivo minimizar efectos problemáticos como:

1. Los circuitos digitales conmutan en tiempos muy breves, generando una gran velocidad de cambio en la corriente consumida desde la fuente  $\left(\frac{di}{dt}\right)$ , esto ocasiona una caída de voltaje de acuerdo a la ley de inducción de Faraday (y considerando el principio de Lenz) que establece que la tensión inducida  $\varepsilon$  está dada por:

$$\varepsilon = -\frac{d\Phi}{dt} = -L\frac{di}{dt} \text{ [V]} \quad (4.21)$$

2. Los conductores de alimentación y tierra, y en general, todos los conductores, contienen inductancias parásitas.
3. La mayoría de los circuitos integrados digitales contienen *flip-flops*, elementos básicos de almacenamiento de estados lógicos; éstos pueden cambiar de estado si el voltaje de polarización fluctúa demasiado por la falta de capacitores de desacoplo.

En la práctica se utilizan capacitores de desacoplo en el rango de 10 [pF] a 100 [nF], valor que se puede calcular de acuerdo a la siguiente ecuación:

$$C = \frac{I\Delta t}{\Delta V} \text{ [F]} \quad (4.22)$$

Donde  $I$  corresponde a la corriente necesaria para conmutar la salida del circuito,  $\Delta t$  es el tiempo requerido por el capacitor para suministrar la corriente y  $\Delta V$  es la caída de voltaje en la alimentación que se puede tolerar por parte del circuito digital. El tipo de capacitores utilizados suele ser cerámico puesto que los capacitores electrolíticos contienen inductancias parásitas. Los capacitores de desacoplo también deben ser colocados lo más cercanamente posible a las terminales de alimentación de los circuitos integrados con el objetivo de mantener los niveles de inductancia lo más bajo posibles.

La figura 4.20 muestra la ubicación del capacitor de desacoplo C2 utilizado para la fuente de polarización que se conecta a la bornera B1 del sistema RAS-II. El capacitor se ubica después del diodo rectificador D2 que en conjunto con el capacitor C1 componen un filtro contra el rizado presente en la fuente de polarización. En la práctica se suelen utilizar dos capacitores en paralelo como en la figura mostrada debajo; el de mayor valor actúa a bajas frecuencias mientras que el de menor valor actúa a altas frecuencias.

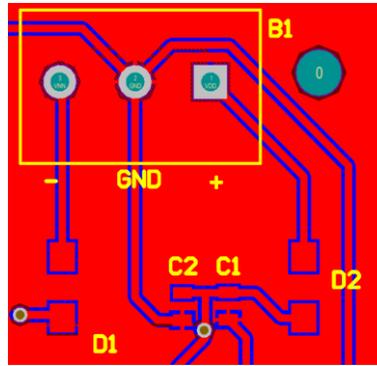


Figura 4.20. Entrada de alimentación de la PCB.

En la PCB del sistema se utilizaron varios capacitores de desacoplo, un ejemplo se muestra en la figura 4.21 en donde los planos de tierra se han suprimido para esclarecer la figura. El capacitor se sitúa en la capa inferior en la proximidad de la fuente de polarización de la memoria de pre-evento.

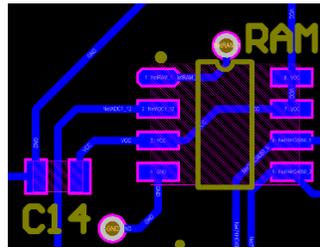


Figura 4.21. Capacitor de desacoplo de la memoria de pre-evento.

- La placa de circuito impreso debe ser dividida en zonas para circuitos de alta, media y baja frecuencia. La figura 4.22 muestra un ejemplo señalado en el documento IPC2221.

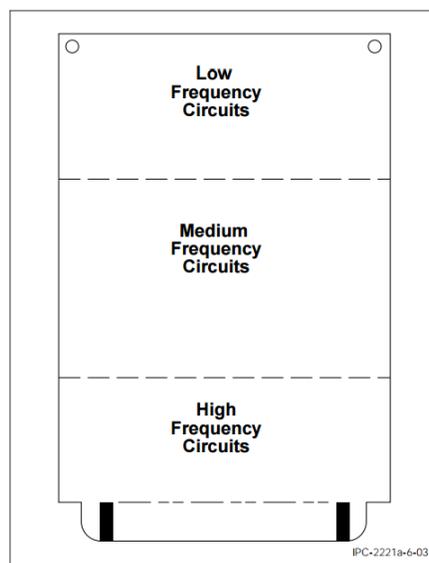


Figure 6-3 Circuit Distribution

Figura 4.22. División de la PCB en zonas de acuerdo a la frecuencia manejada.

En el diseño de la placa de este proyecto se agruparon los distintos componentes en zonas denominadas en *Altium Designer* como *Rooms* (habitaciones). Una *Room* es un área definida dentro de la placa de circuito impreso que reúne componentes de un mismo subsistema con características similares o que cumplen con un fin común y que son agrupados en una zona caracterizada por un nombre. Los nombres asignados a cada *Room* fueron: “CH1” (figura 4.23), “CH2” (figura 4.24), “CH3” (figura 4.25), “Digital” (figura 4.26) y “Supply” (figura 4.27).

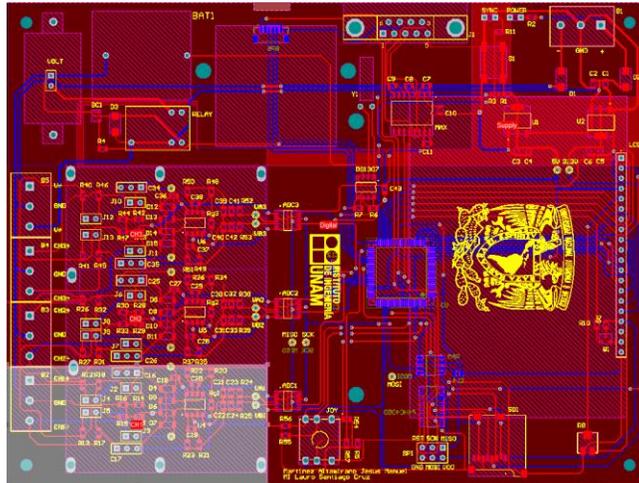


Figura 4.23. Room “CH1”.

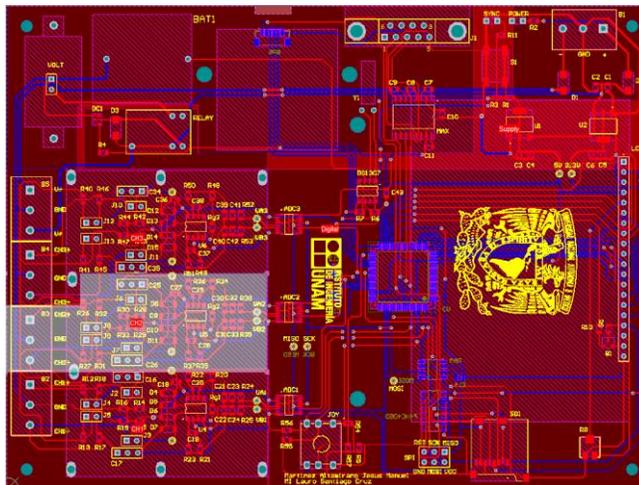


Figura 4.24. Room “CH2”.

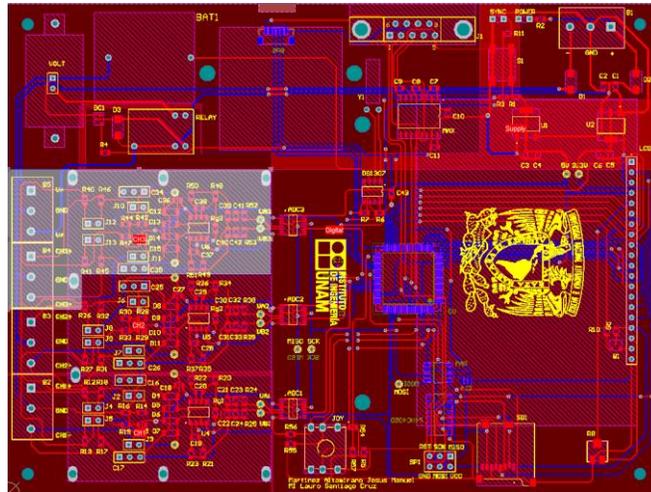


Figura 4.25. Room "CH3".

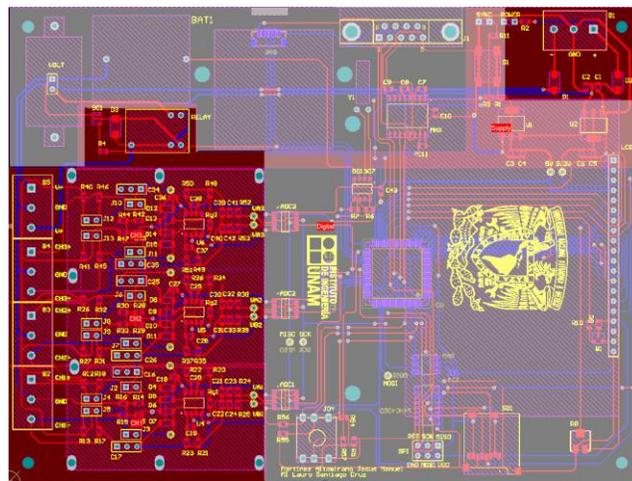


Figura 4.26. Room "Digital".

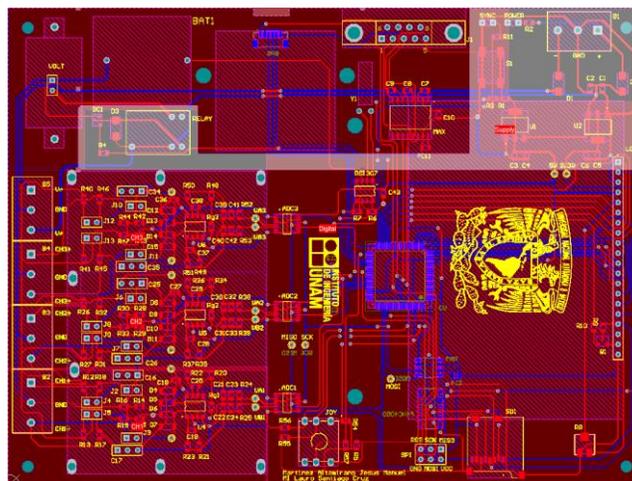


Figura 4.27. Room "Supply".

La zona conformada por las *Rooms* “CH1”, “CH2” y “CH3” corresponde a un área en donde únicamente se realiza el acondicionamiento de las señales de entrada, que por su naturaleza, son de baja frecuencia. En esta zona sólo se pueden encontrar componentes pasivos como resistencias y capacitores, diodos, conectores y amplificadores operacionales que operan a bajas frecuencias. Aislado los componentes que operan a bajas frecuencias de los circuitos digitales, se elimina un posible problema de modulación en las señales que poseen una elevada velocidad de conmutación entre estados lógicos.

En la zona denominada “*Digital*” se colocaron todos los circuitos digitales: los tres convertidores analógico-digitales, el microcontrolador, la memoria de pre-evento, la memoria principal, el reloj de tiempo real, el receptor GPS, el *display* gráfico y un transceptor para la comunicación mediante el puerto DB9. Adicionalmente, el microcontrolador se ubicó en la capa inferior para prevenir un comportamiento errático debido a interferencias entre señales manejadas por circuitos aledaños. En esta zona se manejan señales de alta frecuencia y, siguiendo la pauta marcada por la figura 4.22, la *Room* “*Digital*” se sitúa más próxima a la fuente de alimentación en comparación a la zona de baja frecuencia, puesto que se desea disminuir lo más posible las inductancias parásitas en las pistas conductoras y al mismo tiempo, cubrir las demandas de corriente ante la elevada razón de cambio  $\frac{di}{dt}$ .

La última zona, “*Supply*”, corresponde a los componentes necesarios para proporcionar la interfaz entre la fuente de polarización del sistema y los voltajes entregados a todos los circuitos integrados; se incluye también un relevador mecánico que, por sus características ajenas al acondicionamiento de señal o procesamiento digital, fue incluido en esta zona.

Vale la pena mencionar que cada *Room* maneja un esquemático propio, de manera que el circuito completo se detalla en 5 plantillas diferentes.

#### *Consideraciones para los circuitos integrados*

- En el caso de utilizar un circuito integrado con múltiples terminales, se debe señalar con alguna marca la primera terminal de dicho circuito con la finalidad de evitar el posicionamiento incorrecto del componente durante el proceso de ensamblado.

En la figura 4.28(a) se muestra el *foot print* del reloj de tiempo real de 8 terminales con un punto utilizado para marcar la primer terminal del circuito integrado. Las terminales del circuito se cuentan en el sentido anti horario. En la figura 4.46(b) se muestra una zona de la PCB donde se sitúan un amplificador operacional y el convertidor analógico-digital del canal 3 con sus respectivas marcas para identificar la primera terminal.

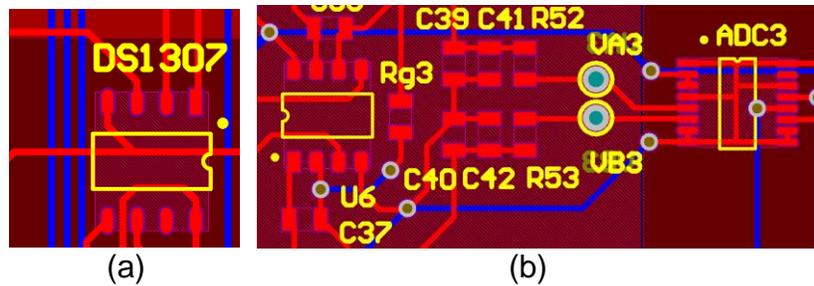


Figura 4.28. Marcas para identificar la posición correcta de los C.I.

- Mantener la longitud de las pistas que conectan circuitos integrados contiguos lo más corta posible. Cuando las pistas conductoras son muy largas se fomenta la aparición del fenómeno denominado *diafonía* (*crosstalk*), en el cual, una señal transmitida por un circuito A, crea un efecto indeseado en un circuito B al acoplarse capacitiva, inductiva o conductivamente. La señal del circuito perturbador (circuito A) aparece acoplada en el circuito perturbado (circuito B) que puede ser reconocida como una señal determinada o como simple ruido pero que en cualquiera de los dos casos representa una interferencia en el circuito perturbado.

Para el caso de la PCB del RAS-II la longitud de los conductores entre dispositivos procuró conservarse lo más corta posible siempre y cuando la ubicación de los componentes a conectar o los elementos localizados en la periferia lo permitieran. Un caso especialmente importante es el correspondiente al cristal de cuarzo utilizado para el circuito oscilador del microcontrolador. Como se observa en la figura 4.29, se recomienda poner especial atención al interconectar los cristales con aquellos circuitos que los requieren, se debe procurar la proximidad entre uno y otro elemento, además, ambos conductores deben tener la misma longitud.

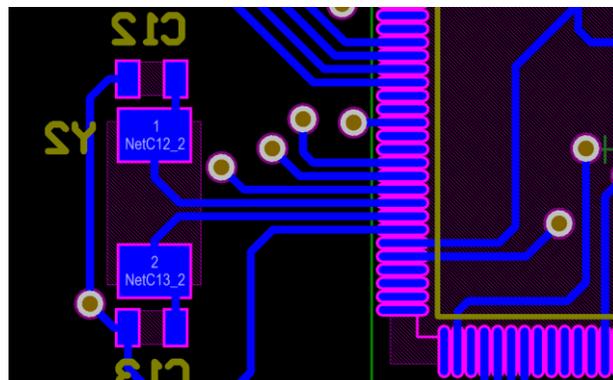


Figura 4.29. Ubicación del cristal.

### Circuitos digitales

- Los circuitos digitales están compuestos por elementos electrónicos que pueden almacenar información en forma de un estado lógico. Los circuitos digitales se manejan en distintas familias lógicas que tienen sus propias características. Entre las familias lógicas se encuentran *TTL* (*Transistor*

*Transistor Logic*), *MOS (Metal Oxide Semiconductor)*, *CMOS (Complementary Metal Oxide Semiconductor)*, entre otras. Las dos familias lógicas más ampliamente utilizadas son:

1. TTL. El rango de voltaje para un valor lógico bajo se encuentra entre 0 [V] y 0.8 [V]. El rango de voltaje para un valor lógico alto se encuentra entre 2.2 [V] a 5 [V]. Sus ventajas son su velocidad de transmisión entre estados lógicos y la mayor resistencia ante descargas por electricidad estática. Sus desventajas son el alto consumo de potencia, requiere una fuente de polarización estable ( $5 \text{ [V]} \pm 0.25 \text{ [V]}$ ) y ofrece un bajo *fan out* (número de compuertas lógicas a las que puede alimentar).
2. CMOS. El rango de voltaje para un valor lógico bajo se encuentra entre 0 [V] y 1 [V]. El rango de voltaje para un valor lógico alto se encuentra entre 3.5 [V] y 5 [V]. Sus ventajas son el bajo consumo de potencia gracias a la alta impedancia de entrada de los MOSFET y a que en estado de reposo un circuito CMOS sólo experimenta corrientes parásitas, inmunidad al ruido, poder auto regenerativo de los niveles lógicos y un mayor *fan out*. Sus desventajas son la menor velocidad de transición entre estados lógicos y alta sensibilidad ante descargas por electricidad estática.

Se recomienda utilizar un solo tipo de familia lógica en la integración del circuito final montado sobre la placa de circuito impreso, esto facilita un único conjunto de reglas de diseño para la longitud y grosor de las pistas.

La familia lógica utilizada en el sistema RAS-II es CMOS con excepción del reloj de tiempo real que pertenece a la familia lógica TTL.

- En el caso de circuitos digitales, se deben identificar aquellos componentes que son más vulnerables a tener comportamientos erráticos debido al tipo de señales que manejan; éstas señales se pueden clasificar como:
  1. *Señales no críticas*. Son aquellas que no requieren de una elevada velocidad de procesamiento y la transición entre estados lógicos puede realizarse con relativa lentitud. Ejemplos de estas señales son aquellas manejadas en buses de datos o de direcciones.
  2. *Señales semi críticas*. Son aquellas en donde se debe tener cuidado en el acoplamiento de señales espurias para evitar falsos disparos. Ejemplos de este tipo de señales son las líneas de reinicio utilizadas por algunos circuitos digitales.
  3. *Señales críticas*. Tienen formas de onda que deben ser monótonas, es decir, mantenerse en un nivel de voltaje específico evitándose su degradación. Una señal de reloj se ubica dentro de esta clasificación ya que un *glitch* (pulso espurio) presente en la forma de onda puede ser procesado como un doble pulso de reloj.
  4. *Señales super críticas*. Son aquellas en las cuales la variabilidad temporal durante el envío de señales digitales puede afectar el resultado total de una

transacción. Esta variabilidad puede ser una ligera desviación de la exactitud en el arribo de una señal (*jitter*), por ejemplo, en el caso de una señal de reloj utilizada para la salida de datos en un convertidor analógico-digital; si el pulso de reloj llega demasiado pronto o demasiado tarde se puede leer un estado lógico incorrecto en la línea de salida de datos del ADC.

Con base en estas directrices, en el diseño de la PCB se identificaron primeramente los componentes que manejarían las señales más críticas, reconociendo a los convertidores analógico-digitales, la memoria RAM, el cambiador de nivel 74HC4050, la memoria microSD y un conector ISP (*In Circuit Programming*) para la programación del microcontrolador, todos estos elementos comparten la línea de reloj SCK generada por el microcontrolador.

En la figura 4.30 se aprecia la zona de la PCB donde se situaron los componentes antes mencionados, se muestra también la línea de reloj resaltada en color blanco. La línea de reloj fue la primera en ser *ruteada* (proceso de interconexión entre dispositivos a través de la generación de pistas) en toda la PCB, procurando ubicar los elementos de la manera más cercana posible. En el primer proceso de *ruteo* la memoria RAM, el cambiador de nivel 74HC4050, el conector ISP y el *socket* para la tarjeta de memoria microSD se ubicaron hacia el extremo izquierdo, proporcionando una mayor cercanía entre dispositivos, sin embargo, pensando en la dificultad para la maniobrabilidad del *joystick* (ver figura 4.31), los elementos fueron reubicados en la posición que se muestra.

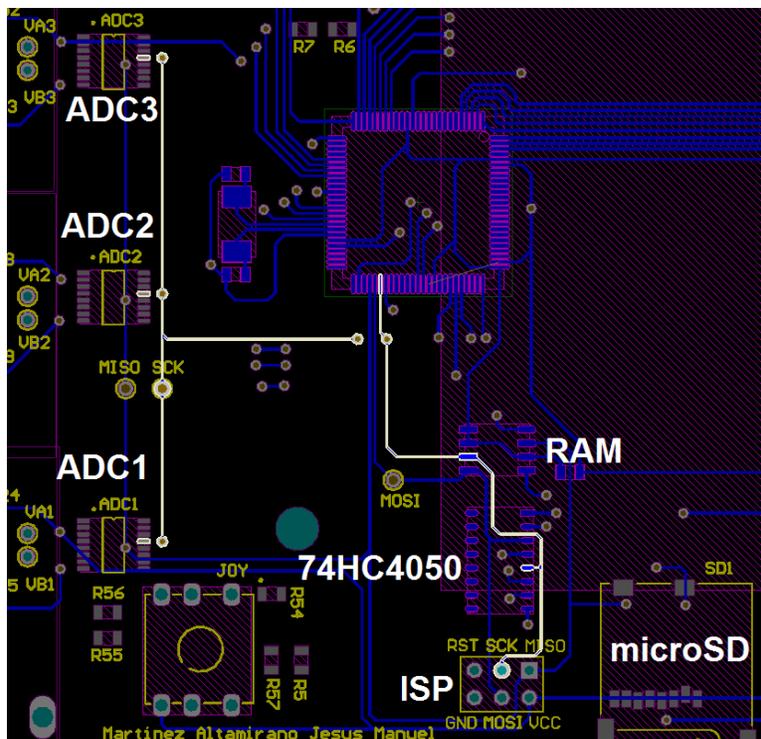


Figura 4.30. Ruteo de los componentes con señales críticas.

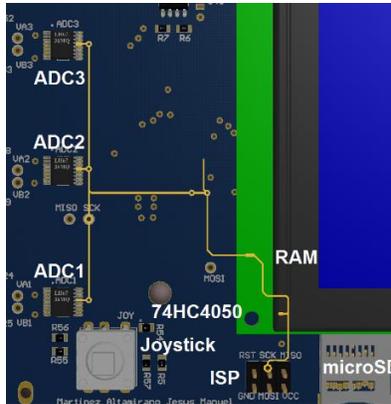


Figura 4.31. Vista 3D de la figura 4.30.

El trazo de la línea de reloj se realizó en una sola capa (capa inferior), excepto en los tres puntos de interconexión con los ADCs (situados en la capa superior) y una zona en donde, por la ubicación del microcontrolador, se utilizó un tramo de 2.7 [mm] en la capa superior para generar la continuidad de la línea. El hecho de realizar el *ruteo* sobre una sola capa disminuye la utilización de *vías* (elementos de interconexión entre capas), que dependiendo de la tecnología de fabricación, puede incrementar el costo de la PCB. Aun cuando las *vías* se realicen mediante electrodeposición, se debe tomar en cuenta que el grosor de la película de cobre en las paredes de las *vías* y *true holes* es de 20 a 25 [ $\mu\text{m}$ ] según este estándar, por lo tanto, la aplicación accidental de una elevada temperatura en las *vías* puede dejar la placa inservible. La segunda y más importante razón de realizar el trazo en una sola capa se debe a que el camino para la corriente eléctrica es más directo en comparación con una ruta que conlleva constantes transiciones entre una capa y otra.

Por último, la ruta de la señal *SCK* es más directa para los convertidores analógico-digitaes cuyo funcionamiento es el más crítico; aunque la separación entre uno y otro ADC es relativamente grande, se consideró que esta era la mejor ubicación posible, puesto que con ello se conservó idéntica la distancia entre cada convertidor analógico-digital y las entradas de señal provenientes de cada canal. En la figura 4.30 las entradas de señal a los ADCs se identifican con las leyendas *VA3* y *VB3* para las entradas de señal diferencial del canal 3, *VA2* y *VB2* para las entradas de señal diferencial del canal 2 y, *VA1* y *VB1* para las entradas de señal diferencial del canal 1. Como se observa, la distancia de dichas señales a su respectivo convertidor analógico-digital es la misma.

#### *Requerimiento de los conductores*

- La anchura y espesor de los conductores en la placa deben ser determinados con base a la capacidad de transporte de corriente y al máximo aumento de temperatura en el conductor. Estos parámetros se relacionan de acuerdo a la siguiente ecuación:

$$I = k\Delta T^{0.44} A^{0.725} \quad (4.23)$$

Dónde  $I$  es la corriente en [A],  $A$  es la sección transversal en milésimas de pulgada cuadrada,  $\Delta T$  es el aumento de temperatura en [°C] que se obtiene de la diferencia entre la temperatura ambiente y la máxima temperatura a la cual se someterá la placa. La constante  $k$  tiene un valor de 0.048 para capas externas y 0.024 para capas internas.

Para capas internas el espesor es la hojuela de cobre a menos que se utilicen vías ciegas o enterradas, en cuyo caso el espesor incluye el chapado de cobre. Para las capas externas el espesor también incluye el cobre chapado depositado durante el proceso del *true hole* pero no incluye el cubrimiento con soldadura o el estañado de un conductor.

La anchura de las pistas se puede obtener a partir de las ecuaciones (4.24) y (4.25) que son obtenidas de las gráficas proporcionadas en el estándar IPC2221 y las cuales son mostradas en la figura 4.32.

$$\text{Sección transversal [mil}^2\text{]} = \left( \frac{I}{k \cdot \Delta T^{0.44}} \right)^{\frac{1}{0.725}} \quad (4.24)$$

$$\text{Anchura [mil]} = \frac{\text{sección transversal}}{\text{espesor} \cdot 1.378} \quad (4.25)$$

En el diseño de la PCB el máximo consumo de corriente es de 100 [mA] asociado a la tarjeta de memoria microSD, sin embargo, se consideró un margen amplio de tolerancia de 100 [mA] con el objetivo de afianzar el funcionamiento del sistema, de esta manera  $I = 200$  [mA].

El componente más crítico en cuanto a la temperatura de operación es el reloj de tiempo real que soporta una temperatura máxima de 70 [°C], por otro lado, considerando una temperatura ambiente de 25 [°C], el aumento de temperatura  $\Delta T$  es igual a 45 [°C].

Con los valores antes señalados se calcula el área de la sección transversal de la pista, a continuación se calcula la anchura, para ello se requiere del espesor del cobre. Los espesores de cobre más comunes son 0.5, 1, 2 y 3 [oz/ft<sup>2</sup>] (onzas por pie cuadrado), los mismos utilizados en la popular FR4, una designación para las placas de fibra de vidrio con resina epóxica resistente a la llama (las siglas FR se deben a las voces inglesas *flame retardant*). Realizando el diseño con base a las peores condiciones para el sistema, se consideró un espesor de 0.5 [oz/ft<sup>2</sup>].

(For use in determining current carrying capacity and sizes of etched copper conductors for various temperature rises above ambient.)

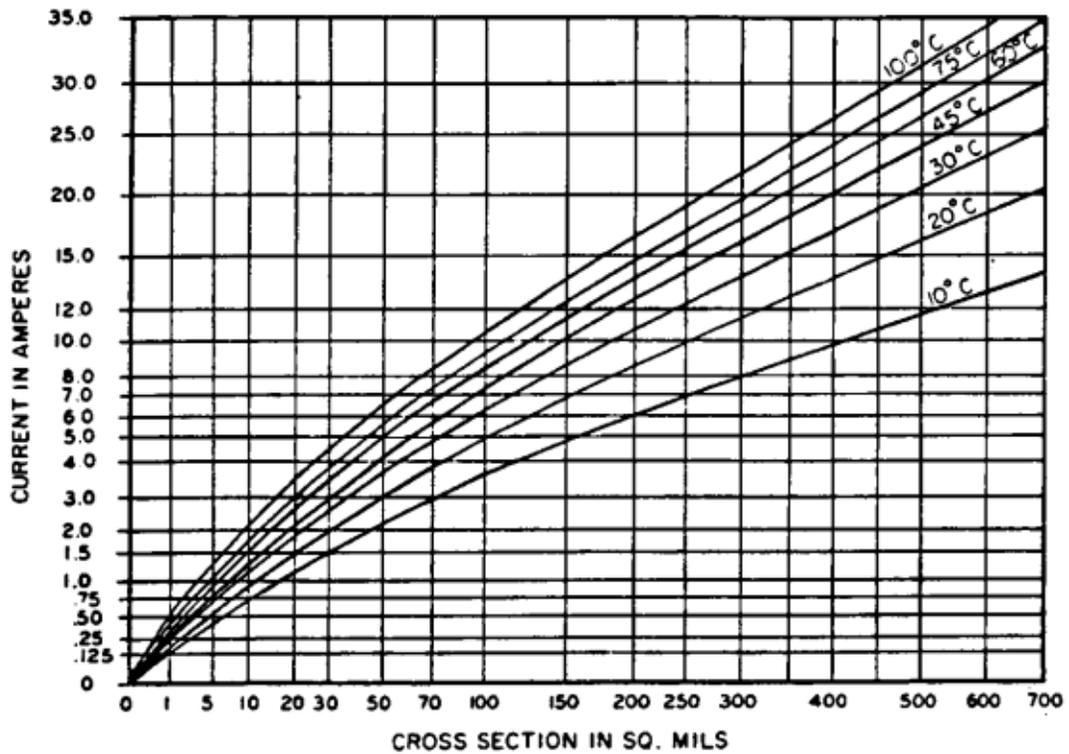


Figure A External Conductors

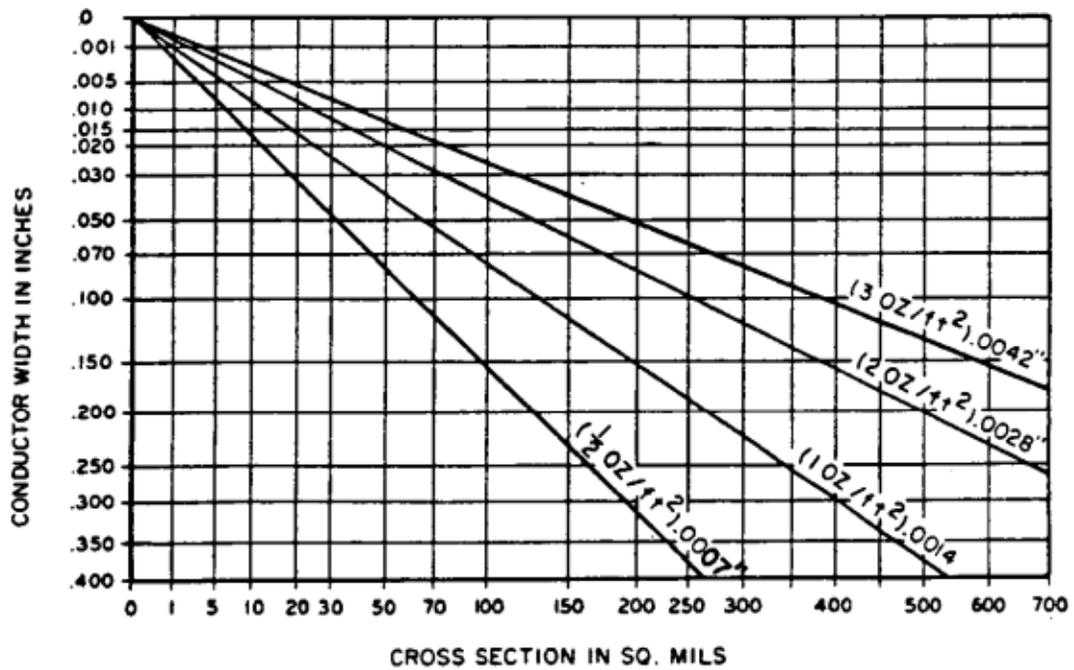


Figure B Conductor width to cross-section relationship

Figura 4.32. Cálculo de la anchura de las pistas.

La anchura de las pistas conductoras con base en los parámetros asignados es: 1.03 [mil] o 0.026 [mm], por lo tanto, si se conserva una anchura de las pistas por lo menos de este valor, se garantizará el funcionamiento correcto de la PCB.

Vale la pena mencionar que aunque el valor teórico de las pistas garantiza el funcionamiento de los circuitos eléctricos sobre la PCB, existen otras limitantes como la resolución mínima que los fabricantes de placas de circuito impreso pueden ofrecer. Cuando el proceso de fabricación se lleva a cabo de manera profesional por una empresa dedicada a esta tarea, no existe dificultad alguna, sin embargo, para el prototipado de subsistemas (como sucedió en el caso del RAS-II), de los cuales se quiere evaluar su operación, las PCBs suelen fabricarse con tecnologías menos avanzadas (para disminuir el costo de diseño del sistema final) donde se presentan limitantes como el tamaño mínimo de las brocas y fresas que trazan las pistas.

Pensando en la fabricación de una PCB previa a la definitiva en las instalaciones del Instituto de Ingeniería, se eligió una anchura de 0.3 [mm] para las pistas, que corresponde a la resolución mínima que puede obtenerse en dichas instalaciones. Como se observa, esta medida es casi 12 veces superior a la teórica, sin embargo, después del análisis realizado, se puede concluir que la importancia de realizar el cálculo de la anchura de las pistas radica en conocer cuál es la medida mínima para garantizar el funcionamiento de la PCB, además, por experiencia se sabe que aunque la resolución de 0.026 [mm] estuviera disponible en las instalaciones del Instituto de Ingeniería, las exposiciones a altas temperaturas durante el proceso de soldado de componentes, traería como consecuencia la asolación de las pistas por carecer de una máscara antisoldante, esto sucede incluso con pistas de una anchura de 0.3 [mm].

Otras características tomadas en cuenta durante el proceso de diseño de la placa de circuito impreso se enlistan a continuación:

- Incrementar el espaciado entre componentes ajenos entre sí; esto disminuye el problema de diafonía. Se debe buscar un equilibrio entre el espaciado de los componentes y las dimensiones finales de la PCB.
- Se debe tomar en cuenta el rendimiento eléctrico del sistema, sin embargo, no se deben olvidar factores como la maniobrabilidad de las partes mecánicas dentro de la PCB o las dimensiones físicas de los componentes que podrían interferir con el posicionamiento de algún otro elemento de hardware. Se recomienda utilizar un cuerpo 3D para los componentes para ubicar visualmente el espacio que éstos ocuparían.
- Se recomienda diseñar puntos de prueba dentro de la placa de circuito impreso con el fin de analizar aquellas señales que permiten verificar el funcionamiento del sistema.

- Evitar el trazado de pistas en ángulos rectos, esto dificulta el flujo de electrones al recorrer un circuito.
- Se recomienda que la orientación de los circuitos integrados y las leyendas en la máscara de componentes sea la misma, siempre que sea posible (para el caso de los circuitos integrados).

#### 4.2.2. Integración del hardware

Una vez terminada la etapa de diseño de la PCB, se generaron los archivos *Gerber* que contienen la información necesaria para la fabricación de la placa de circuito impreso. Estos archivos se generan con cualquier software relacionado con el diseño de PCBs, que para este proyecto fue *Altium Designer*. La PCB fue fabricada por una compañía extranjera dedicada a la manufactura y ensamble de placas de circuito impreso. En la figura 4.33(a) se puede observar la cara superior de la PCB diseñada mientras que en la figura 4.33(b) se aprecia la cara inferior de dicha tarjeta.

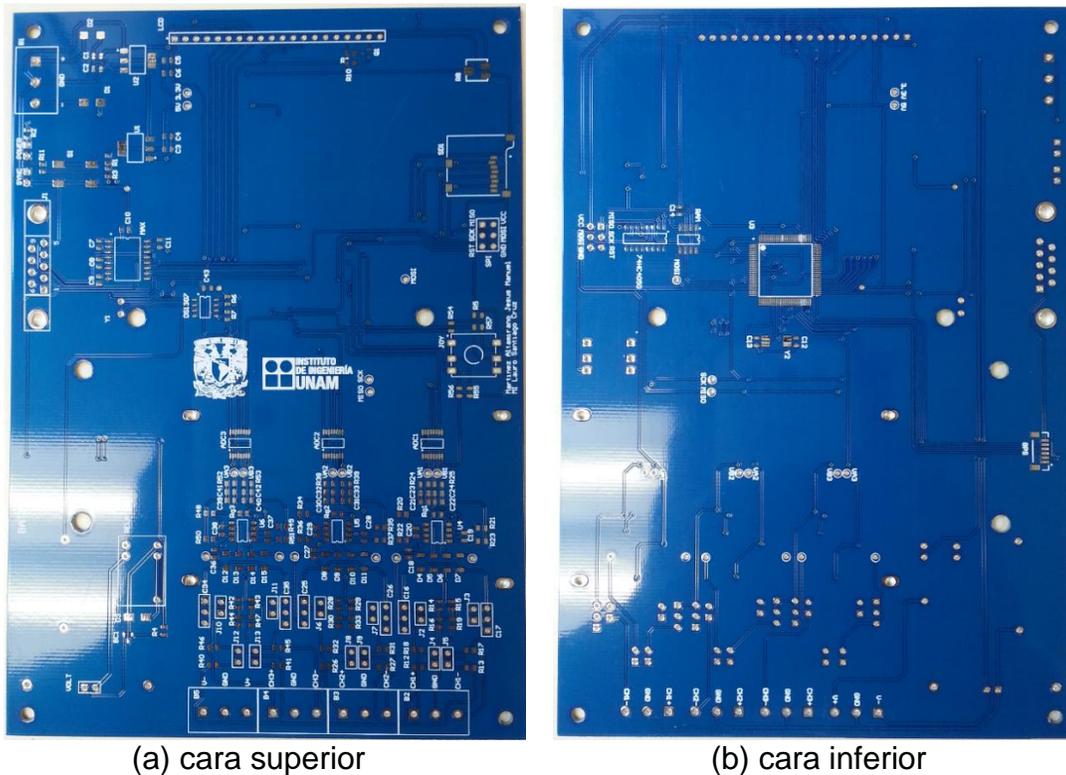


Figura 4.33. PCB del sistema RAS-II.

Los componentes electrónicos fueron incorporados a la placa de circuito impreso iniciando por aquellos cuyas dimensiones no dificultaran la integración de los demás, de esta manera, se soldaron primero las resistencias, capacitores, diodos y transistores seguidos por circuitos integrados, sockets y finalmente, conectores de señal. La figura 4.34 muestra el ensamble final de la PCB.



Figura 4.34. PCB del sistema con todos los elementos integrados.

Los diferentes circuitos dentro de la PCB fueron probados individualmente, una vez que se garantizaba su funcionamiento se proseguía con la integración de otro subsistema.

Para grabar el programa informático en la memoria del microcontrolador se utilizó el programador *AVRISP mkII* fabricado por la compañía *Atmel* diseñado para la actualización de hardware en los microcontroladores de la familia AVR. El programador hace uso de la programación serial en el sistema o ISP (*In-system Programming*). El programador se muestra en la figura 4.35.



Figura 4.35. Programador *AVRISP mkII*.

En la PCB se reservó un conector de 6 terminales para enchufar el programador mencionado a la placa. En la figura 4.36 se muestra la conexión entre el programador y la PCB del sistema. El programador solo puede conectarse en una posición, si se ha conectado de forma incorrecta el led del programador permanece en color rojo. Cuando el programador se conecta de la manera correcta el led permanece en color verde tal y como se aprecia en la figura.

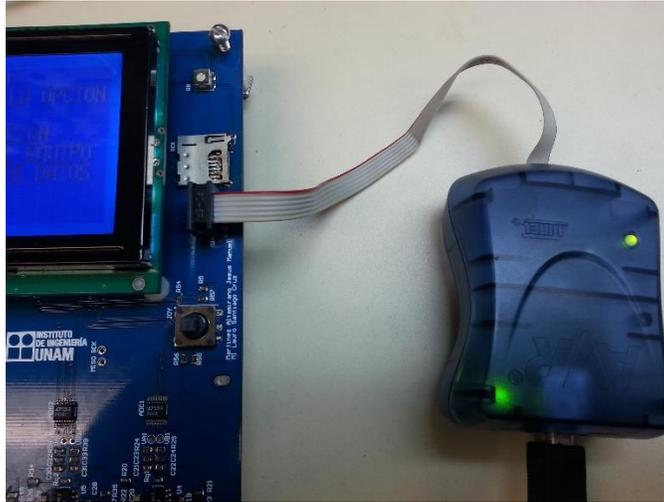


Figura 4.36. Programación del microcontrolador.

Atmel ofrece una plataforma de desarrollo integrado (*Integrated Development Platform, IDP*) de nombre *Atmel Studio* para desarrollar código, depurar y realizar la programación de la familia de microcontroladores AVR. El proceso para cargar el programa a la memoria del microcontrolador es el siguiente: en la plataforma *Atmel Studio* seleccionar la opción *Device Programming* en la pestaña *Tools* (figura 4.37).

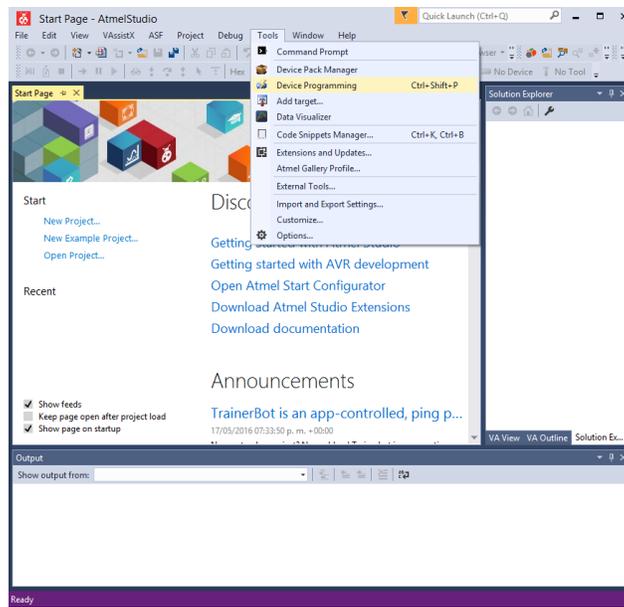


Figura 4.37. Iniciando programación del microcontrolador.

Después de seleccionar la opción *Device Programming* aparece una ventana emergente (figura 4.38), se debe seleccionar la herramienta *AVRISP mkII* en la sección *Tool*; en seguida se debe seleccionar la matrícula del microcontrolador que se desea programar (en este caso *ATmega2560*), a continuación se da *click* sobre el botón *Apply*, inmediatamente se mostrará la pantalla mostrada en la figura 4.39.

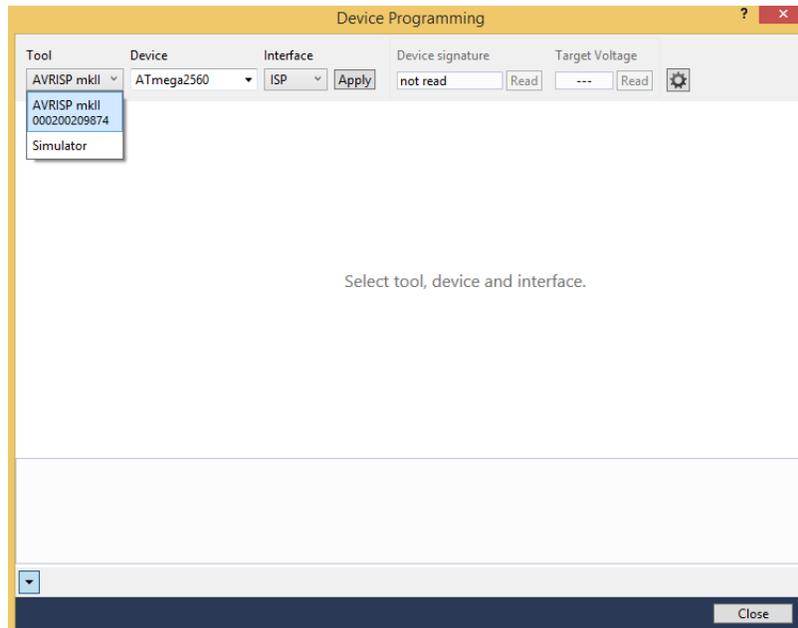


Figura 4.38. Ventana *Device Programming*.

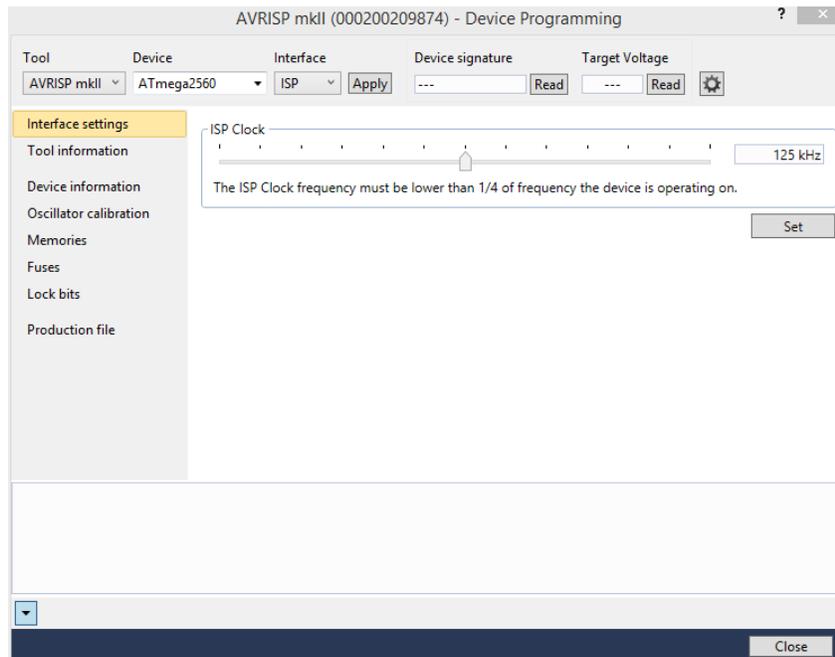
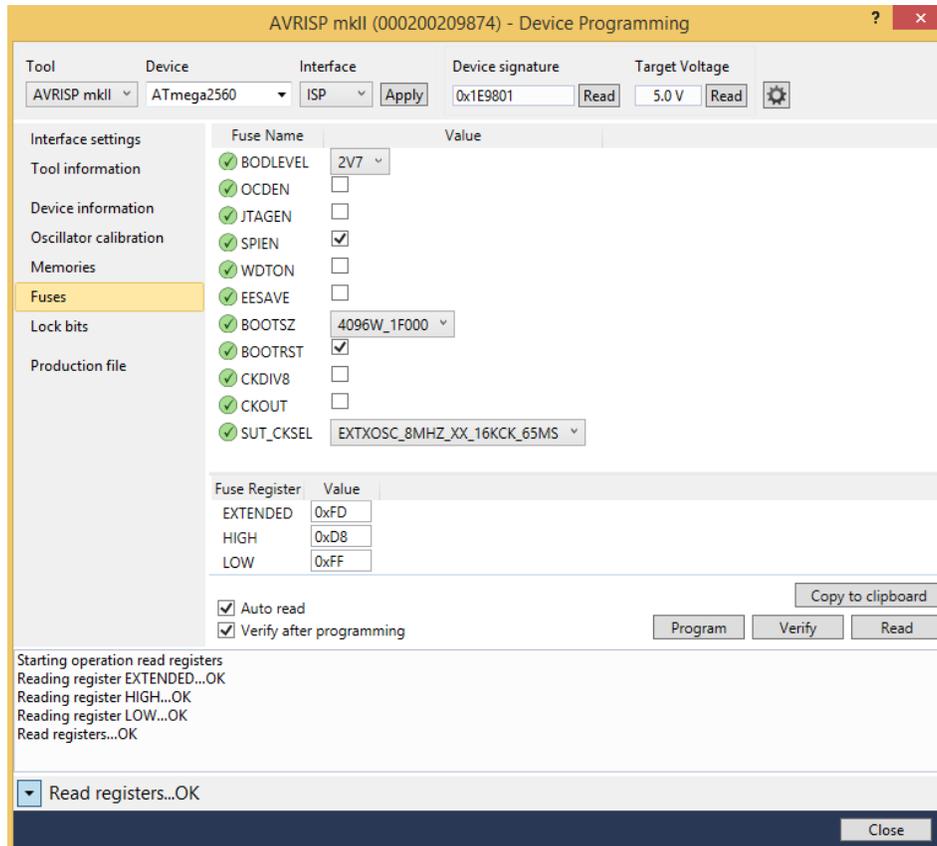


Figura 4.39. Ventana emergente.

Los fusibles del microcontrolador vienen configurados por de fábrica, lo cual puede contraponerse a las necesidades del usuario. Para el caso del microcontrolador del sistema RAS-II, los fusibles fueron establecidos de acuerdo a la figura 4.40. Se estableció un voltaje de 2.7 [V] para la detección de baja tensión en el sistema, se habilitó el bit para la programación de la memoria mediante ISP, se configuró la sección de memoria para la carga de un programa de arranque con una medida de 4096 palabras iniciando en la dirección 0h1F000, se habilitó la función de reinicio y se estableció un oscilador de cristal externo de 16 [MHz] como fuente de reloj.



4.40. Configuración de los fusibles.

Una vez que se configuran los fusibles se debe hacer *click* en el botón *program*, con esto los fusibles del microcontrolador quedan configurados.

Cuando el microcontrolador no es nuevo, estando en la ventana mostrada en la figura 4.39 se debe configurar la frecuencia de reloj para la programación mediante *ISP*, ésta debe ser como máximo la cuarta parte de la frecuencia de reloj del microcontrolador (en este caso hasta 4 [MHz]), se da *click* sobre el botón *Set* para establecer esta frecuencia. A continuación se da *click* sobre el botón *Read*, el microcontrolador responderá con el número identificador 0x1E9801 (para el caso del microcontrolador ATmega 2560), en cualquier otro caso se tendrá un problema durante la programación de la memoria del microcontrolador.

Después de que el microcontrolador ha respondido con el identificador correcto, en la pestaña *Memories* se muestran dos opciones, la primera es para la programación de la memoria de programa (memoria flash) y la segunda es para la programación de la memoria EEPROM (ver figura 4.41). La opción concerniente a la memoria Flash corresponde a la memoria de programa, debe seleccionarse la ruta donde se almacena el archivo fuente (archivo *hex*) para realizar la programación de dicha memoria. Una vez seleccionada la ruta, se da *click* sobre el botón *Program*. Con esto el microcontrolador del sistema queda listo para realizar las funciones programadas.

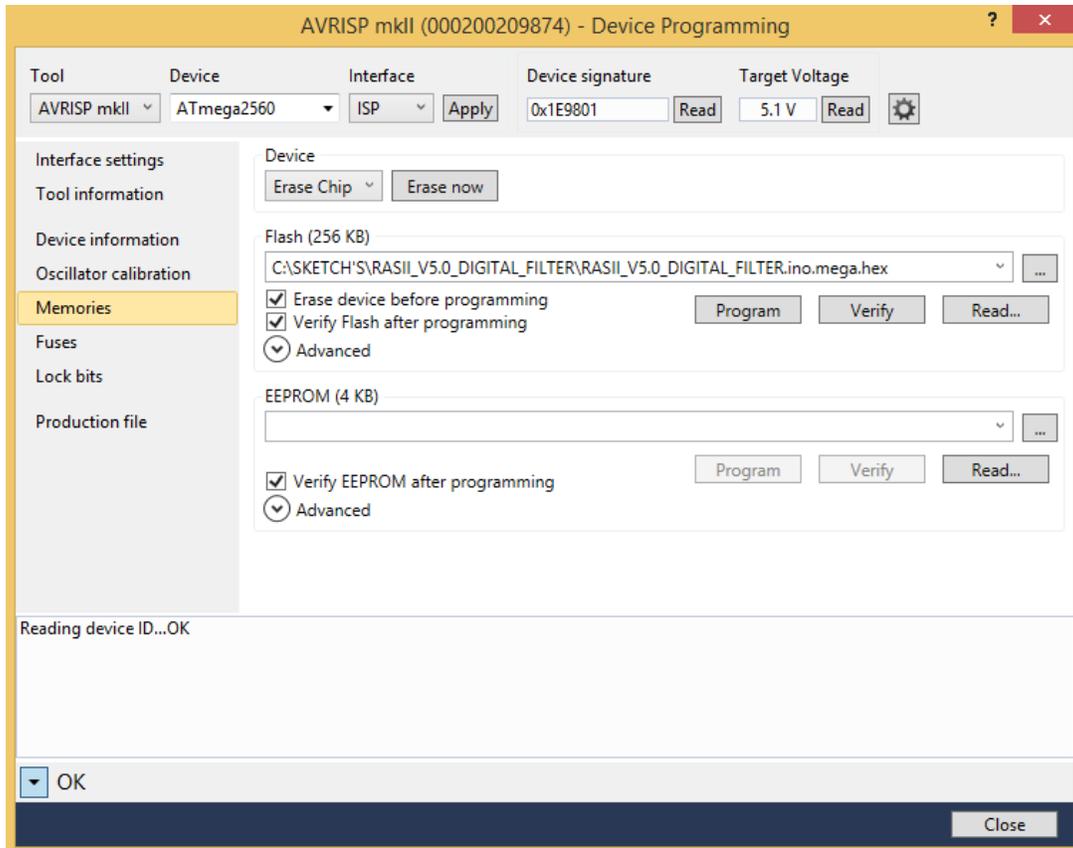


Figura 4.41. Programación de la memoria de programa.

#### 4.2.3. Pruebas al circuito de acondicionamiento de señal

Las pruebas realizadas al circuito de acondicionamiento de señal del sistema RAS-II son las mismas que se realizaron en el apartado 4.1.1 al prototipo del sistema. El circuito de acondicionamiento de señal para sensores activos se repite una vez más en la figura 4.42.

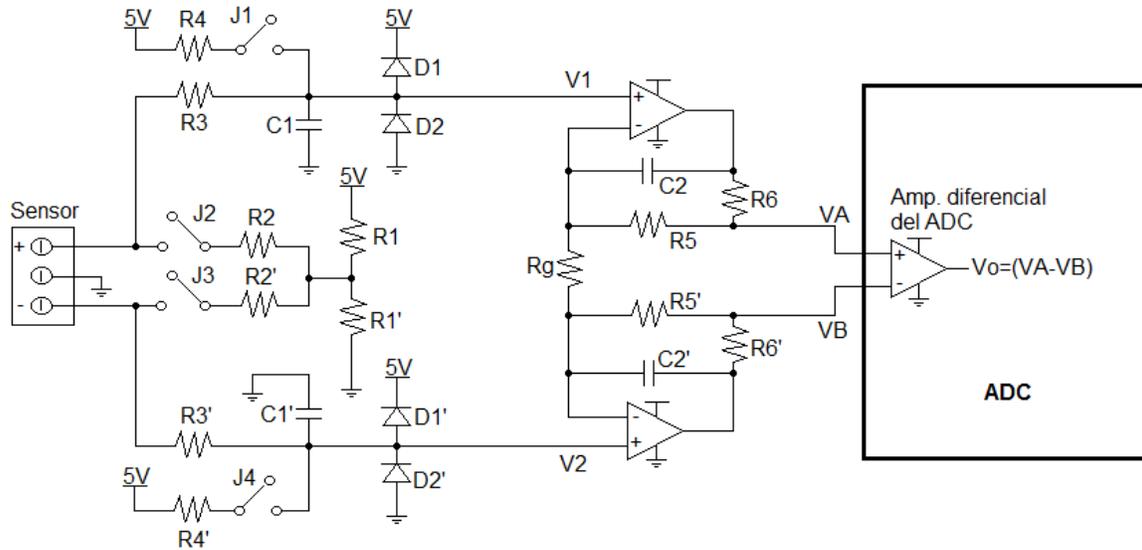


Figura 4.42. Circuito de acondicionamiento de señal para sensores activos.

### Prueba de ganancia

La primer prueba corresponde a la verificación de la magnitud de los voltajes  $V_A$ ,  $V_B$  y el voltaje diferencial  $V_o = V_A - V_B$  a bajas frecuencias, en donde la respuesta en frecuencia de los componentes no afecta a la ganancia de dichas señales. De acuerdo al análisis realizado en el apartado 4.1.1, los voltajes antes mencionados están dados por las siguientes ecuaciones:

$$V_A = 0.75v_{s1} + 0.5V_{CC} \quad (4.26)$$

$$V_B = -0.25v_{s1} + 0.5V_{CC} \quad (4.27)$$

$$V_o = V_A - V_B = v_{s1} \quad (4.28)$$

Trabajando con el canal 3 del sistema RAS-II (figura 4.43), a dicho canal se le proporcionó una señal senoidal de amplitud igual a 1 [V] pico y una frecuencia de 3 [Hz], esta señal corresponde al voltaje indicado como  $v_{s1}$  en la figura 4.42; en cuanto a la terminal correspondiente a la conexión de la señal  $v_{s2}$ , ésta fue conectada a tierra para trabajar con señales sencillas y no diferenciales, por lo tanto, el voltaje  $v_{s2}$  es igual a 0 [V]. La señal senoidal  $v_{s1}$  se muestra en la figura 4.44.

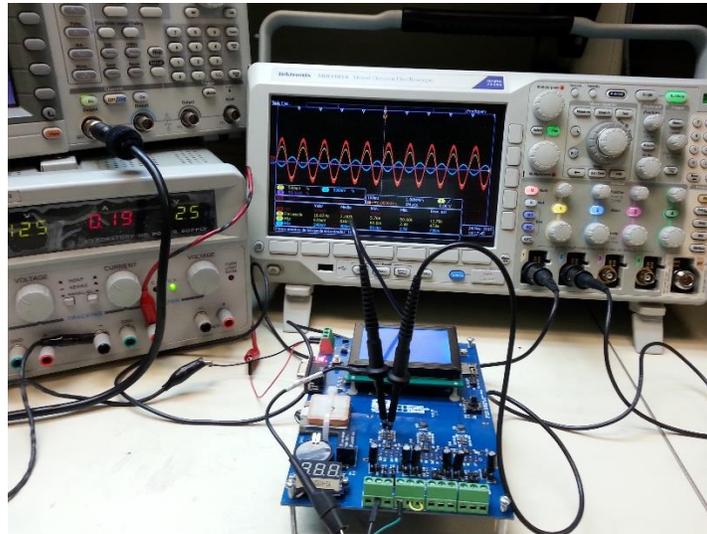


Figura 4.43. Sistema bajo prueba.

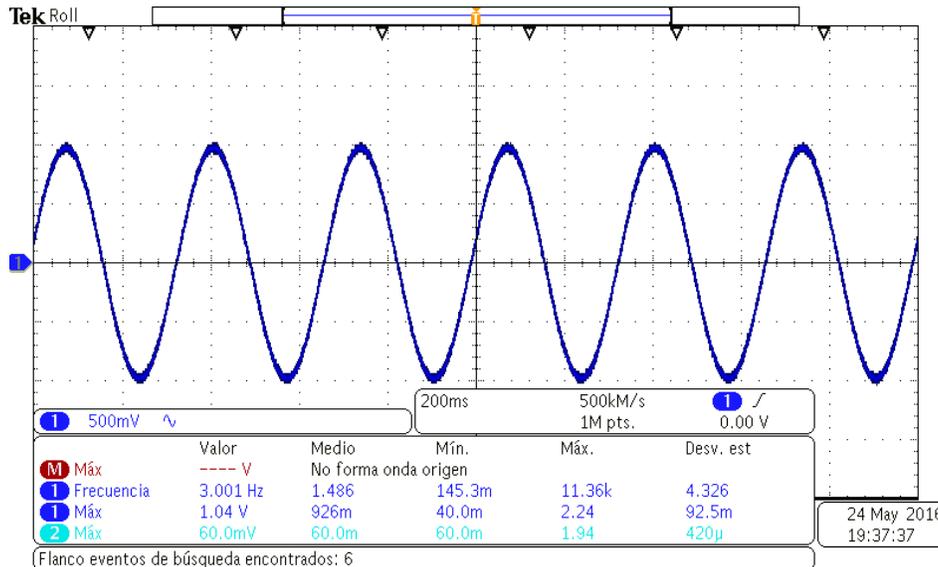


Figura 4.44. Señal senoidal de 1 [V] pico y 3 [Hz].

Para predecir la magnitud de las señales  $V_A$ ,  $V_B$  y  $V_o$ , en la figura superior se puede observar que el voltaje máximo de la señal de entrada es 1.04 [V] (este voltaje se indica en el cuadro de medidas). Tomando este valor como  $v_{s1}$ , además recordando que  $V_{cc} = 5$  [V], con las ecuaciones (4.26), (4.27) y (4.28) se pueden obtener los voltajes teóricos  $V_A$ ,  $V_B$  y  $V_o$ .

$$V_{A_{teórico}} = 0.75v_{s1} + 0.5V_{cc} = 0.75(1.04 \text{ [V]}) + 0.5(5 \text{ [V]}) = 3.28 \text{ [V]}$$

Para el caso de  $V_B$ , el signo negativo en la ecuación (4.27) indica una inversión de fase de  $180^\circ$ , sin embargo, para calcular el valor absoluto máximo se realiza lo siguiente:

$$V_{B_{teórico}} = |-0.25v_{s1}| + 0.5V_{cc} = |-0.25(1.04 \text{ [V]})| + 0.5(5 \text{ [V]}) = 2.76 \text{ [V]}$$

Para el caso del voltaje diferencial  $V_o$ , el valor teórico debe ser exactamente el mismo que el voltaje de entrada  $vs1$ , esto es: 1.04 [V].

### Resultados

En la figura 4.45 se muestra el oscilograma de los voltajes  $V_A$ ,  $V_B$  y  $V_o$ ; la última señal se obtuvo utilizando la función matemática “resta” del osciloscopio puesto que no se tiene acceso a dicha señal en la PCB (se maneja internamente al convertidor analógico-digital). La señal  $V_A$  se grafica en el primer canal y aparece en color azul marino, la señal  $V_B$  se grafica en el segundo canal y se muestra en color azul celeste y la señal en color rojo pertenece a la sustracción de las señales  $V_A - V_B$  que corresponde al voltaje diferencial  $V_o$ .

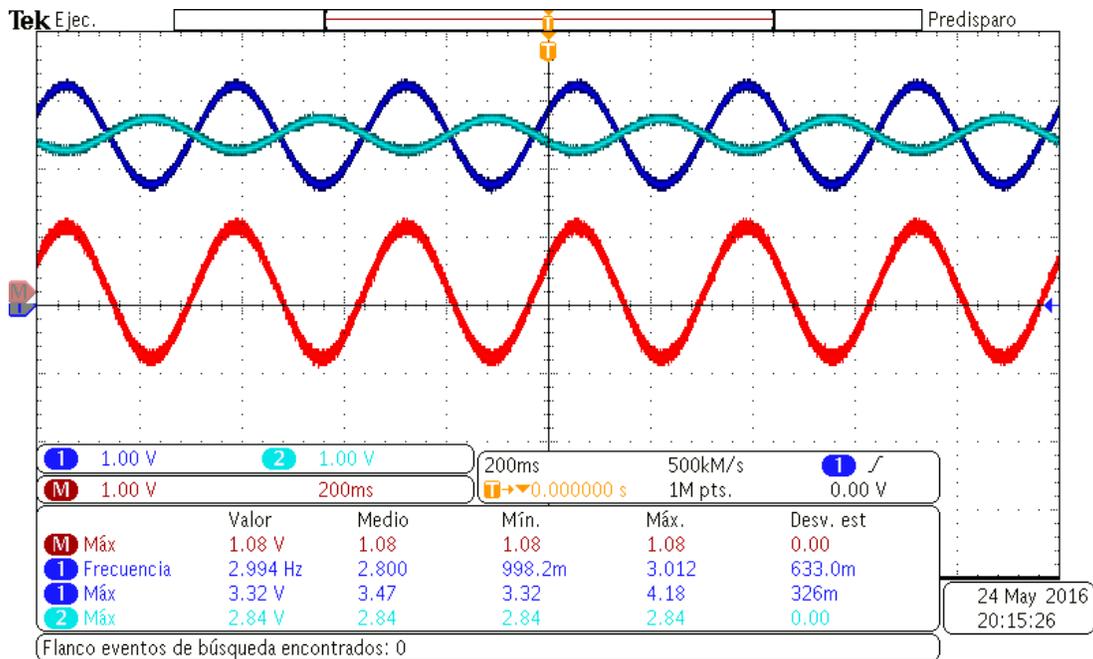


Figura 4.45. Oscilograma de las señales bajo prueba.

En el oscilograma de la figura se pueden identificar los voltajes  $V_A$ ,  $V_B$  y  $V_o$  reales o experimentales. En el cuadro de medidas de esta figura se observa que el voltaje  $V_A$  experimental es  $V_{A_{experimental}} = 3.32$  [V], el voltaje  $V_B$  experimental es  $V_{B_{experimental}} = 2.84$  [V] y el voltaje  $V_o$  experimental es  $V_{o_{experimental}} = 1.08$  [V], además la frecuencia de las señales sigue siendo de 3 [Hz].

Con todos estos datos se puede calcular el porcentaje de error entre los valores teóricos esperados y los valores reales o experimentales, esto se realiza de la manera siguiente:

$$\%e_{V_A} = \frac{|V_{A_{teórico}} - V_{A_{experimental}}|}{V_{A_{teórico}}} \times 100 = \frac{|3.28 - 3.32|V}{3.28V} \times 100 = 1.2\%$$

$$\%e_{VB} = \frac{|VB_{teórico} - VB_{experimental}|}{VB_{teórico}} \times 100 = \frac{|2.76 - 2.84|V}{2.76V} \times 100 = 2.8\%$$

Después de analizar los resultados experimentales se puede concluir que el circuito de acondicionamiento de señal tiene un comportamiento muy aproximado al teórico (considerando los porcentajes de error obtenidos menores al 5%), de esta manera, el circuito cumple con sus tres funciones básicas:

1. Proporcionar una componente de corriente directa de 2.5 [V] para manejar señales positivas y negativas como aquellas provenientes de los acelerómetros DCA333 y DSA1.
2. Proporcionar una ganancia unitaria para manejar los acelerómetros DCA333 y DSA1 cuya salida se encuentra en el rango de  $\pm 2.5$  [V]. A su vez, la ganancia unitaria y la adición de la componente de directa permiten operar al sistema a partir de un regulador cuya salida es igual a 5 [V].
3. Realizar el acoplamiento de impedancias entre el sensor y la entrada del convertidor analógico-digital, evitando la degradación de la señal entrante y un efecto de sobrecarga al ADC.
4. Servir como primera etapa para la obtención de señales diferenciales a la entrada del convertidor analógico-digital.

#### *Prueba al filtro anti-aliasing*

Trabajando con el circuito de acondicionamiento de señal del sistema RAS-II, se observaron las señales a la salida de los filtros para corroborar la atenuación de las señales con frecuencias mayores a 12 [Hz]. Cada filtro tiene la configuración mostrada en la figura 4.46.

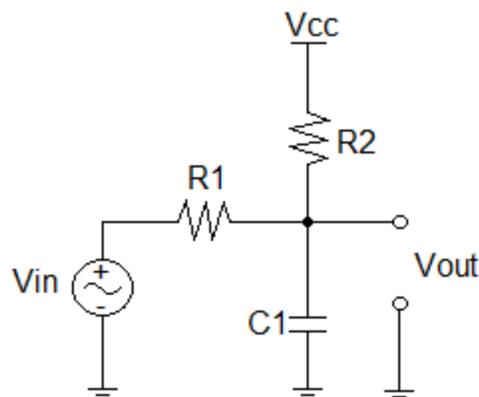


Figura 4.46. Filtro anti-aliasing.

Del análisis al circuito anterior, realizado en el apartado 3.2.1, se llegó a la expresión que describe el comportamiento de tal arreglo, esta expresión es la siguiente:

$$V_{out} = \left(\frac{R_1}{R_1+R_2}\right)V_{CC} + \frac{1}{C_1R_1} \left(\frac{1}{s+\frac{R_1+R_2}{C_1R_1R_2}}\right)V_{in} \quad (4.29)$$

La primer componente de esta expresión indica la adición de 2.5 [V] de corriente directa (considerando que  $V_{CC}$  es igual a 5 [V] para el sistema RAS-II), mientras que el segundo elemento revela un comportamiento paso bajas para la señal de corriente alterna  $V_{in}$ . De acuerdo a las consideraciones establecidas en el diseño, en donde se asignó  $R_1 = R_2$ , la frecuencia de corte teórica del filtro se encuentra en:

$$f_{cteórica} = \frac{2}{2\pi C_1 R_1} [Hz] \quad (4.30)$$

En el sistema RAS-II los valores de los componentes llevan a la frecuencia de corte teórica del filtro de 11.76 [Hz].

Para describir el comportamiento teórico de los filtros, se deben observar las señales V1 y V2 señaladas en el circuito de la figura 4.42, estos voltajes están dados por las ecuaciones (4.31) y (4.32).

$$V1 = 0.5V_{cc} + 36.95 \left(\frac{1}{s+73.90}\right) vs1 = 2.5[V] + 36.95 \left(\frac{1}{s+73.90}\right) vs1[V] \quad (4.31)$$

$$V2 = 0.5V_{cc} + 36.95 \left(\frac{1}{s+73.90}\right) vs2 = 2.5[V] + 0[V] \quad (4.32)$$

La terminal correspondiente a la entrada de señal  $vs2$  fue conectada a tierra pues se trabajó con una señal sencilla y no diferencial, con esto, el voltaje V2 únicamente tendrá una componente de directa de 2.5 [V] (recordando que  $V_{cc} = 5$  [V]), teniendo en mente esto, sólo se observó la respuesta en frecuencia de la señal V1.

La señal V1 se compondrá de un voltaje de 2.5 [V] de corriente directa y una respuesta paso bajas para la señal  $vs1$  con frecuencia de corte igual a 11.76 [Hz]. Para frecuencias menores a 11.76 [Hz] la ganancia del sistema será igual a 0.5 o -6.02 [dB], mientras que al sobrepasar dicha frecuencia, el filtro debe tener teóricamente una caída de 20 [dB] por década.

Trabajando con el canal 3 del RAS-II, a dicho canal se le proporcionó una señal senoidal de amplitud igual a 2 [V] pico, realizando un barrido en frecuencia para observar el comportamiento del filtro.

En la figura 4.47 se puede observar la señal senoidal de entrada en color azul marino, con frecuencia igual a 113 [mHz], la señal V1 en color azul celeste, que es una señal senoidal montada sobre una componente de directa y, la señal V2 en color magenta, que corresponde a una componente de directa.

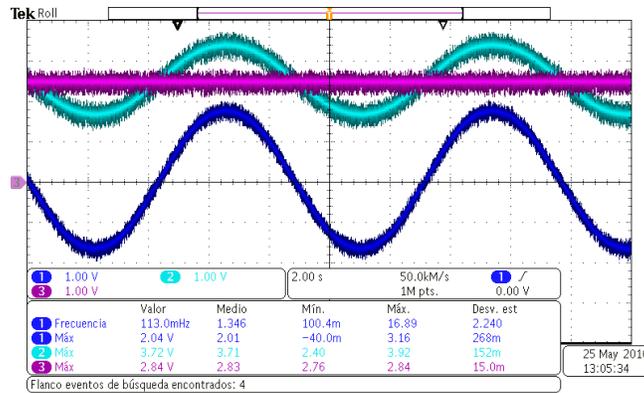


Figura 4.47. Señal de entrada y señales V1 y V2.

### Resultados

Observando el comportamiento en AC (corriente alterna) de la señal V1, se varió la frecuencia de la señal de entrada comenzando en 1 [Hz], posteriormente la frecuencia fue modificada a 2, 4, 8, 12.3, 16, 20, 40 y finalmente, 123 [Hz]. Los oscilogramas a estas frecuencias se muestran respectivamente en las figuras 4.48, 4.49, 4.50, 4.51, 4.52, 4.53, 4.54, 4.55 y 4.56.

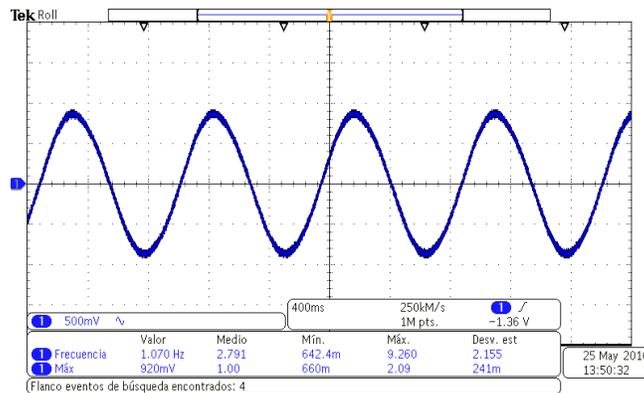


Figura 4.48. Respuesta a 1 [Hz].

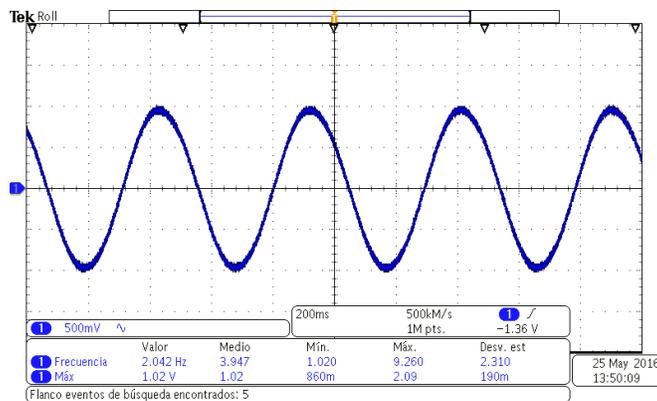


Figura 4.49. Respuesta a 2 [Hz].

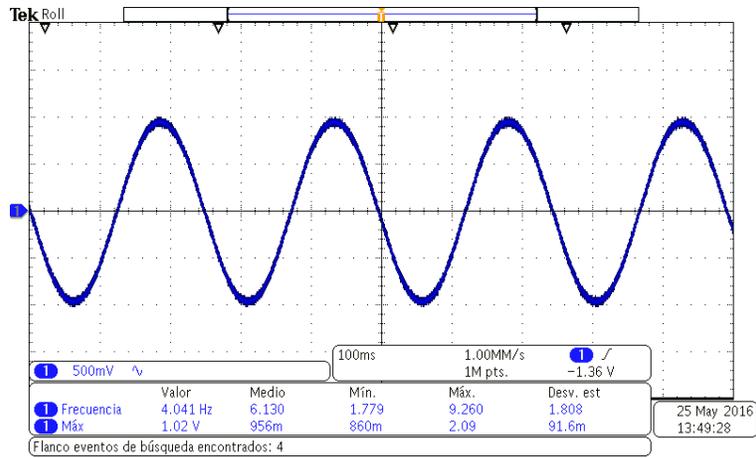


Figura 4.50. Respuesta a 4 [Hz]

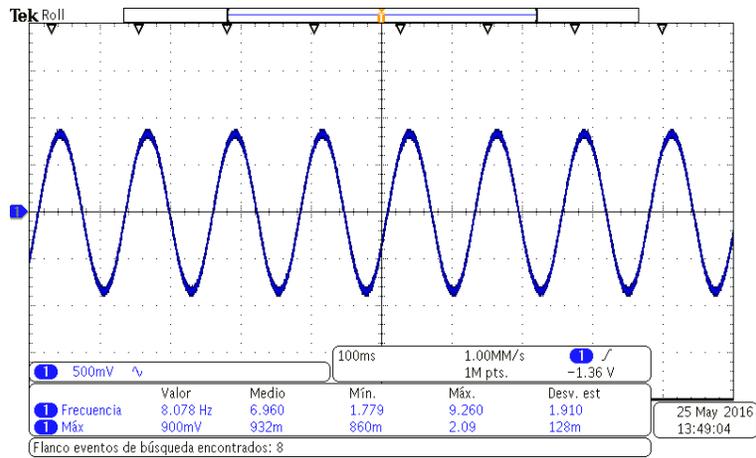


Figura 4.51. Respuesta a 8 [Hz].

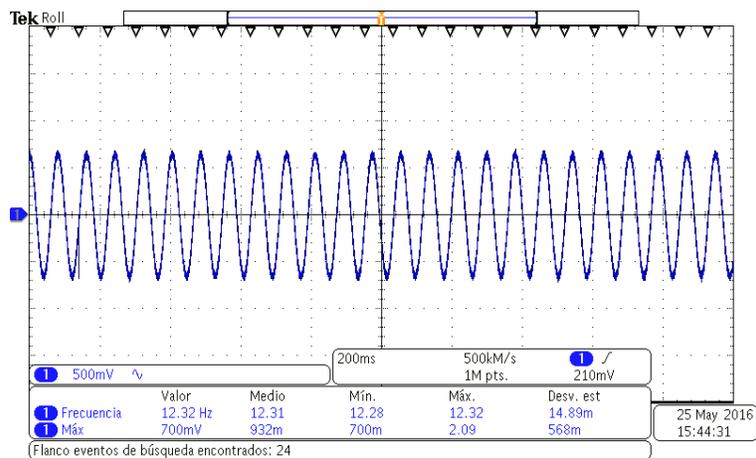


Figura 4.52. Respuesta a 12.3 [Hz].

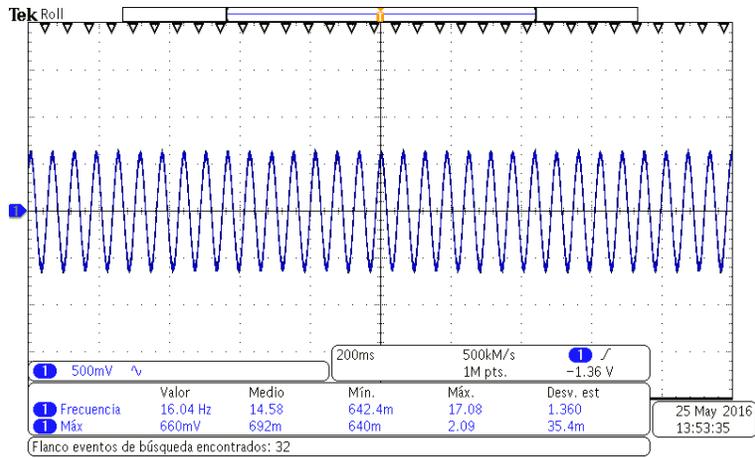


Figura 4.53. Respuesta a 16 [Hz]

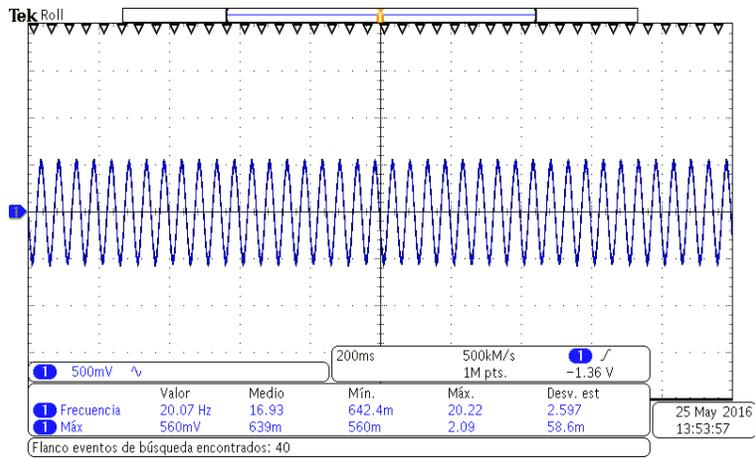


Figura 4.54. Respuesta a 20 [Hz].

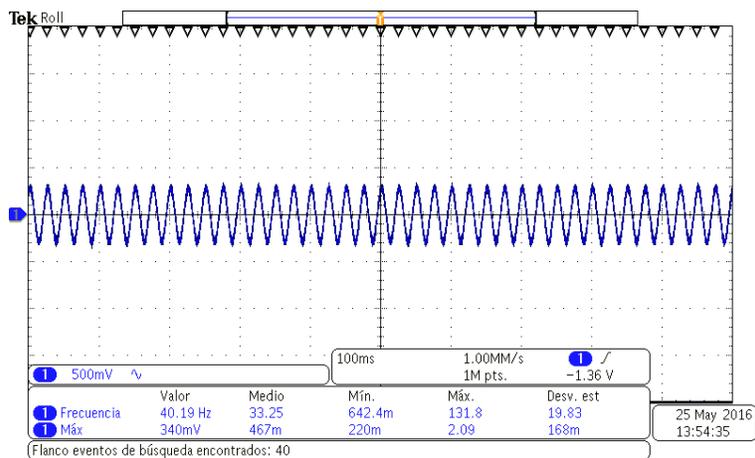


Figura 4.55. Respuesta a 40 [Hz].

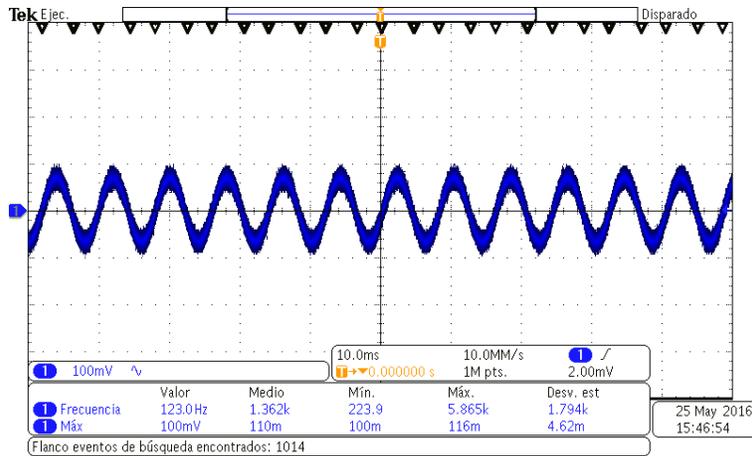


Figura 4.56. Respuesta a 123 [Hz].

Tomando como datos los niveles de tensión máximos mostrados en cada uno de los oscilogramas, en la tabla 4.2 se resume la variación de la ganancia del filtro al modificar la frecuencia de la señal de entrada.

Frecuencia [Hz]	Vin [V]	Vout [V]	Vout/Vin [1]	Ganancia [dB]	Oscilograma
1	2	0.92	0.46	-6.74	Fig. 4.48
2	2	1.02	0.51	-5.84	Fig. 4.49
4	2	1.02	0.51	-5.84	Fig. 4.50
8	2	0.90	0.45	-6.93	Fig. 4.51
12.3	2	0.70	0.35	-9.11	Fig. 4.52
16	2	0.66	0.33	-9.6	Fig. 4.53
20	2	0.56	0.28	-11.05	Fig. 4.54
40	2	0.34	0.17	-15.39	Fig. 4.55
123	2	0.10	0.05	-26.02	Fig. 4.56

Tabla 4.2. Resultados del barrido de frecuencias.

A partir de los oscilogramas mostrados e inspeccionando la tabla 4.2, se observa que para una señal de entrada de 2 [V], a la salida del filtro el nivel de voltaje de la señal V1 es aproximadamente igual a 1 [V], en otras palabras, se tiene una ganancia aproximada a 0.5 o equivalentemente -6 [dB]. La frecuencia de corte se presentó en

$f_{c_{experimental}} = 12.3$  [Hz], en donde la señal V1 tiene el 70% del valor inicial. Al aumentar la frecuencia hasta alcanzar un valor 10 veces superior a la frecuencia de corte experimental, esto es, a una frecuencia de 123 [Hz], se tiene una caída de 20 [dB] con respecto a la ganancia original ( $-6$  [dB]  $- 20$  [dB] =  $-26$  [dB]) que se corresponde con la atenuación de un filtro paso bajas de primer orden.

Después de realizar las pruebas al filtro anti-aliasing se puede llegar a la conclusión de que el filtro cumple con su objetivo:

1. Limitar en banda a las señales provenientes de los acelerómetros, estableciendo un ancho de banda de 12.3 [Hz].

#### 4.2.4. Reconstrucción de la señal muestreada

La prueba de reconstrucción de la señal muestreada permite verificar que la señal reconstruida a partir de las muestras y la señal original son iguales. Para asegurar que ambas señales son iguales, se debe comprobar que tanto la amplitud como la frecuencia de dichas señales es la misma.

Empleando un generador de funciones, se introdujo una señal senoidal de 500 [mV] pico y frecuencia igual a 3 [Hz] (figura 4.57) al canal 2 del RAS-II, a continuación se realizó una operación de almacenamiento continuo.

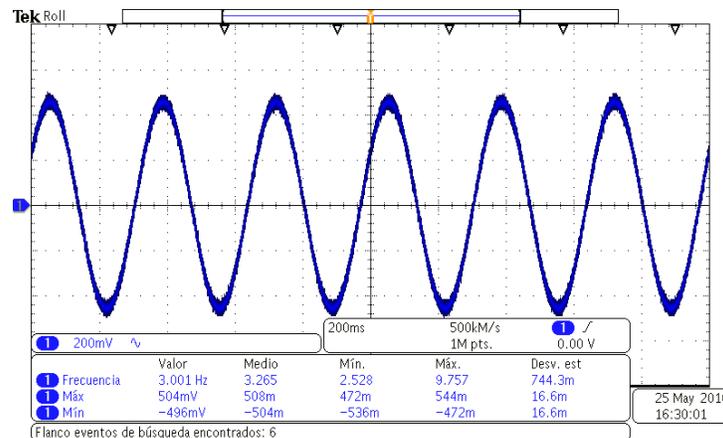


Figura 4.57. Señal a digitalizar.

Se detuvo la prueba después de algunos segundos, se extrajo la memoria microSD del sistema RAS-II y se conectó a una computadora de escritorio; con ayuda de *Microsoft Excel*, se obtuvieron los voltajes máximo y mínimo registrados, también las parejas de datos (tiempo y voltaje) fueron graficados obteniendo la señal mostrada en la figura 4.58.

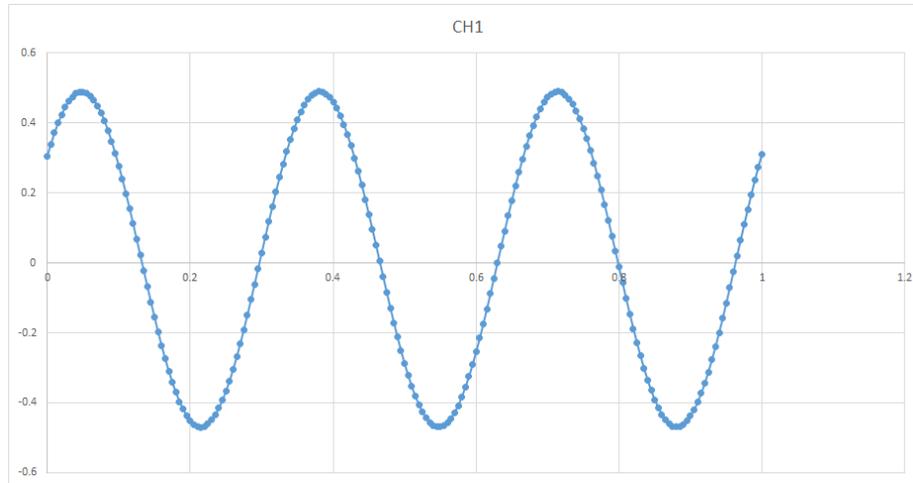


Figura 4.58. Señal reconstruida.

## Resultados

### Amplitud

Los valores máximo y mínimo experimentales son:

$$V_{m\acute{a}x_{exp}} = 497 \text{ [mV]}$$

$$V_{m\acute{i}n_{exp}} = -496 \text{ [mV]}$$

En el oscilograma de la figura 4.57, en el cuadro de medidas se puede observar que el voltaje máximo teórico es  $V_{m\acute{a}x_{te\acute{o}rico}} = 504 \text{ [mV]}$ , mientras que el voltaje mínimo teórico es  $V_{m\acute{i}n_{te\acute{o}rico}} = -496 \text{ [mV]}$ . Para el caso del voltaje mínimo no existe ningún error puesto que los valores teórico y experimental son idénticos, por otro lado, para el caso del voltaje máximo el porcentaje de error es:

$$\%e_{V_{m\acute{a}x}} = \frac{|V_{m\acute{a}x_{te\acute{o}rico}} - V_{m\acute{a}x_{exp}}|}{V_{m\acute{a}x_{te\acute{o}rico}}} \times 100 = \frac{|504 - 497| \text{ mV}}{497 \text{ mV}} \times 100 = 1.4\%$$

A partir del valor de los errores obtenidos en esta prueba se puede concluir que el sistema RAS-II cumple con la condición de amplitud.

### Frecuencia

Para el caso de la frecuencia de la señal, de acuerdo a la figura 4.57, la frecuencia de la señal de entrada es  $f_{te\acute{o}rica} = 3.001 \text{ [Hz]}$ , por otro lado, para determinar la frecuencia de la señal reconstruida a partir de las muestras se tomaron los siguientes datos del archivo de texto:

$$\text{Tiempo de la primer muestra: } 0 \text{ [s]}$$

*Tiempo de la última muestra: 1 [s]*  
*Número de ciclos de la señal: 3*

La frecuencia experimental es  $f_{exp} = 3 [Hz]$ . El porcentaje de error entre la frecuencia de la señal original y la frecuencia de la señal reconstruida a partir de las muestras es:

$$\%e_f = \frac{|f_{teórica} - f_{exp}|}{f_{exp}} \times 100 = \frac{|3.001 - 3|Hz}{3.001Hz} \times 100 = 0.03\%$$

A partir del valor del error obtenido en esta prueba se puede concluir que el sistema RAS-II cumple con la condición de frecuencia y, por lo tanto, cumple con su objetivo de digitalización conservando las propiedades de la señal de entrada.

---

# Capítulo 5.

# Resultados y conclusiones

A lo largo de este trabajo escrito se realizó una descripción paso a paso de las etapas que conformaron el desarrollo de este proyecto, así, al comienzo se abordaron temas relacionados con las ondas sísmicas y la repercusión de éstas en el contexto nacional, a continuación, se describieron conceptos teóricos en el área de la electrónica para consolidar las bases del diseño llevado a cabo posteriormente. Al llegar a la etapa de diseño, se siguieron dos procesos: primero, se experimentó con el sistema DIGI y se incluyeron nuevos componentes para que éste pudiera ofrecer las características solicitadas por los usuarios, una vez logrado esto, se procedió con el diseño de un nuevo equipo electrónico (hardware), que, como un todo, tuviera todas las funcionalidades solicitadas; el nuevo equipo se llamó RAS-II. El segundo proceso, asociado al software del sistema, es la implementación del programa informático necesario para que el equipo solucionara cada una de las tareas emergentes en su trayecto hacia su objetivo final, a saber: realizar el monitoreo de las señales provenientes de los acelerómetros tri-axiales DSA1 y DCA333 y, registrar en una memoria de datos la variación de dichas señales únicamente cuando éstas sobrepasen un valor predeterminado de aceleración. Finalmente, se realizaron distintas pruebas al equipo electrónico para evaluar su desempeño parcial y sacar conclusiones particulares sobre cada uno de los bloques constitutivos de éste.

A continuación se hace mención de los resultados generales del equipo electrónico, así como de las conclusiones finales emanadas de este proyecto.

## 5.1. Resultados

Después de someter el equipo a las pruebas descritas en el capítulo 4, así como al interactuar con éste y evaluar su funcionamiento, se subraya que el equipo satisface las características que solicitaron los usuarios, éstas son:

1. Disparo por umbral en los tres canales. Cuando el equipo se programa para dispararse por umbral, es decir, al elegir un umbral de disparo de 0.5, 1, 1.5, 2, 2.5 y así sucesivamente hasta 10 [Gal], éste es capaz de iniciar el registro de datos en la memoria microSD cuando la señal en cualquier canal sobrepase dicho valor. Una característica adicional no solicitada, es el modo de almacenamiento continuo, en este modo el equipo tiene un funcionamiento similar al de su predecesor DIGI.
2. Filtro anti-aliasing de 12 [Hz]. El equipo cuenta con un filtro pasivo paso bajas con frecuencia de corte experimental de 12.3 [Hz], de este modo, las señales con componentes de frecuencia superior son atenuadas por el equipo. Se permite también la configuración de la frecuencia de corte al intercambiar los capacitores del filtro pasivo.
3. Frecuencia de muestreo = 200 [Hz]. La tasa de muestreo utilizada es 200 [Hz].
4. Datos de pre-evento = 50 [s]. Con la utilización de la memoria RAM, el equipo garantiza el almacenamiento de 50 [s] exactos de información, previa a cada uno de los eventos registrados.
5. Datos de post-evento = 60 [s]. Con la utilización del *Algoritmo de registro de datos*, el equipo garantiza el almacenamiento de 60 [s] exactos de información, posterior a cada uno de los eventos registrados.
6. Capacidad de re-disparo. El equipo es capaz de operar por un tiempo indefinido (siempre y cuando se proporcione una fuente de polarización en todo momento) y realizar el registro de datos en cualquier momento en que la variación en cualquiera de los canales sobrepase el umbral de disparo establecido.
7. Tiempo de la primera muestra en el archivo de datos. Con la utilización del receptor GPS y actuando como respaldo el RTC, el equipo proporciona siempre la fecha y hora a cada archivo de datos.
8. Ajuste automático del reloj por GPS. Con la utilización del receptor GPS el usuario no necesita configurar la hora pues la hora transmitida por los satélites es exacta y ésta simplemente es registrada en el archivo de datos.
9. Ajuste de reloj manual por teclado. El usuario puede interactuar con el equipo para configurar la fecha y hora del RTC que él desee (siempre y cuando sean una fecha y hora válidas) y ésta puede ser utilizada en el archivo de datos.

10. *Display* para despliegue de parámetros. El equipo incluye un *display* de cristal líquido que permite al usuario visualizar la información para la navegación, configuración e inicio de las pruebas en el equipo.
11. Despliegue del reloj en el *display*. El equipo utiliza una rutina para mostrar al usuario tanto la hora obtenida del receptor GPS así como la obtenida del RTC.
12. Teclado para ajuste de parámetros. El equipo incluye un *joystick* que posibilita el ingreso de datos por parte del usuario, de manera que el equipo pueda ser configurado de acuerdo a sus necesidades.
13. Tarjeta de memoria de datos intercambiable. El equipo incluye un *socket* para memorias microSD, con esto, el usuario simplemente tiene que extraer la tarjeta de memoria presente e insertar una nueva para registrar datos en esta última.
14. Visualización del número de eventos sucedidos. Durante la operación del equipo, éste lleva un conteo del número de eventos registrados y los muestra al usuario con ayuda del *display* de cristal líquido.
15. Visualización del voltaje de la batería. El equipo incluye un pequeño voltímetro de manera que el usuario pueda activarlo en el momento que desee conocer el voltaje de alimentación del equipo.
16. Visualización de la cantidad de memoria usada y disponible. El equipo es capaz de proporcionar al usuario la cantidad de memoria utilizada así como la cantidad de memoria disponible a través del *display*.
17. Visualización del offset de los 3 canales. El equipo muestra en un plano cartesiano la variación de la señal en cada uno de los canales, de manera que el usuario pueda cerciorarse que los acelerómetros estén respondiendo correctamente.

Los resultados son favorables en el entendido que cumplen con las peticiones hechas por parte de los usuarios, así mismo, existe una correspondencia entre las señales analógicas que ingresan al equipo con aquellas obtenidas después del proceso de reconstrucción a partir de las muestras, esto último se demostró en el capítulo 4.

## 5.2. Conclusiones

Con base en los resultados puntualizados en el capítulo 4 y las características descritas en el apartado anterior, se puede concluir que el equipo diseñado cumple con el objetivo que motivó a este proyecto, a saber: contar con un equipo electrónico moderno, útil para hacer uso de los acelerómetros tri-axiales DSA1 y DCA333 en el área relacionada con la sismología, capaz de registrar la información proveniente de dichos acelerómetros en una memoria de datos y con la versatilidad de no requerir ningún elemento de hardware para su configuración, ni ningún elemento de software adicional para la extracción y procesamiento de los datos almacenados.

De manera personal, la elaboración de este trabajo me permitió: *reafirmar* los conocimientos adquiridos en el aula de clases, *aprender* los conocimientos que no adquirí a lo largo de mi formación como estudiante, *buscar* soluciones aun cuando no haya una solución evidente, *tolerar* las problemáticas que se enfrentan durante la realización de un proyecto de ingeniería, *preferir* las soluciones correctas, fiables y honestas aun cuando ello demande un sacrificio y, finalmente, *formar* un criterio profesional que me permita desempeñar mi labor como ingeniero de la mejor manera posible, incluyendo además de factores técnicos y presupuestales, otros de igual trascendencia como tiempo y viabilidad con base en objetivos finales. Así mismo, gracias a mi estancia en el Instituto de Ingeniería adquirí conocimientos adyacentes que me obligan a considerar el beneficio para la sociedad más que la factibilidad técnica de realizar cualquier proyecto de ingeniería.

## 5.3. Recomendaciones

Todo equipo electrónico está sujeto a adecuaciones y el sistema RAS-II no es la excepción. A continuación se citan algunas de las adecuaciones sugeridas para que el equipo sea mucho más confiable y su rendimiento general mejor.

1. Implementar un mecanismo para el monitoreo del voltaje de la batería, de manera que una vez que se presente una condición de baja tensión, el equipo sea capaz de cerrar el archivo de datos y conservar la información durante la operación en el modo de registro continuo.
2. Habilitar el *Watchdog* para prevenir el comportamiento errático del sistema, al presentarse un colapso inesperado de éste durante su operación.
3. Mejorar o rediseñar el algoritmo de disparo de manera que se registren únicamente fenómenos sísmicos; posiblemente es conveniente profundizar en el área concerniente al estudio de las ondas sísmicas de manera que en base a sus propiedades, se diseñe un nuevo algoritmo que discrimine totalmente las vibraciones mecánicas medioambientales.

4. Realizar un filtrado digital a las señales. Si el nuevo algoritmo de disparo no mitiga el número de eventos registrados debidos al ruido ambiental, una opción podría ser realizar un filtrado digital a las señales. Así mismo, se sugiere considerar la migración hacia un procesador digital de señales (DSP) con mejores prestaciones y mayor velocidad de procesamiento.
5. La mayoría de los acelerógrafos modernos incorporan la funcionalidad de conexión a internet, esto tiene como finalidad proporcionar de manera remota la información recabada por el acelerógrafo. Se sugiere que el equipo proporcione esta característica para poder competir con los acelerógrafos comerciales.

---

## Apéndice A: Glosario de términos

**ADC.** Analog to Digital Converter. Es un dispositivo utilizado para convertir una señal analógica a su equivalente digital.

**Aliasing.** Es el fenómeno que se presenta cuando se digitaliza una señal analógica al utilizar una frecuencia inferior a la de Nyquist.

**ENOB.** Effective Number of Bits. Es un parámetro que indica cuántos bits del resultado de la conversión analógico-digital se encuentran libres de ruido. Sirve como parámetro para determinar si el ADC cuenta con buenas prestaciones en cuanto a resolución.

**Estándar.** Modelo, norma, patrón o referencia.

**Evento.** Es la ocurrencia de una perturbación ocasionada por un fenómeno sísmico.

**Firmware.** Es un bloque de instrucciones que contiene tareas programadas para el funcionamiento de un sistema electrónico.

**Hardware.** Conjunto de elementos físicos o materiales que constituyen una computadora o un sistema informático.

**LCD.** Liquid Crystal Display. Es una pantalla plana formada por una matriz de pixeles monocromos colocados en una fuente de luz reflectora.

**GPS.** Global Positioning System. Sistema utilizado para la obtención de parámetros como: fecha, hora y posición desde cualquier punto sobre la Tierra.

**I2C.** Inter Integrated Circuit. Es un estándar de comunicación utilizado para la comunicación serial entre distintos circuitos electrónicos dentro de un mismo sistema.

**IDE.** Integrated Development Environment. Es una aplicación de software que proporciona servicios integrales para los programadores para el desarrollo de algún software.

**Interfaz.** Conexión física o lógica, entre el usuario y una computadora, o entre el usuario y un dispositivo electrónico.

**Joystick.** Dispositivo de control utilizado para la navegación en un sistema electrónico.

**microSD.** Es un tamaño y formato de tarjeta de memoria flash.

**NMEA.** Es un protocolo de comunicación que permite comunicar dispositivos de navegación tales como el GPS.

**Post-evento.** Es la información posterior a la ocurrencia de un evento.

**Pre-evento.** Es la información previa a la ocurrencia de un evento.

**Protocolo.** Conjunto de reglas que se establecen en el proceso de comunicación entre dos sistemas.

**RAM.** Random Access Memory. Dispositivo donde se pueden almacenar datos o instrucciones para ser utilizados posteriormente.

**RTC.** Real Time Clock. Es un circuito integrado que permite conocer la hora actual.

**Sensor.** Dispositivo que detecta una determinada magnitud física (temperatura, presión, aceleración, etc.) y proporciona una señal proporcional a dicha magnitud.

**SD.** Secure Digital. Es un formato de tarjeta de memoria flash que incluye los tres tamaños existentes: tarjeta de memoria SD estándar, tarjeta de memoria miniSD y tarjeta de memoria microSD.

**Software.** Conjunto de programas y rutinas que permiten a una computadora realizar determinadas tareas.

**SPI.** Serial Peripheral Interface. Es un estándar de comunicaciones, usado principalmente para la transferencia de información entre circuitos integrados en equipos electrónicos.

**UART.** Universal Asynchronous Receiver Transmitter. Módulo que proporciona al microcontrolador la capacidad de comunicarse con otros dispositivos utilizando el protocolo RS-232.

**UTC.** Universal Time Coordinated. Es un estándar de la referencia de tiempo internacional mediante la cual se calculan todos los husos horarios

---

# Referencia de figuras

## Capítulo 1

Figura 1.1. Cinturón de fuego.

Disponible en:

<[https://es.wikipedia.org/wiki/Cintur%C3%B3n\\_de\\_Fuego\\_del\\_Pac%C3%ADfico#/media/File:Pacific\\_Ring\\_of\\_Fire-es.svg](https://es.wikipedia.org/wiki/Cintur%C3%B3n_de_Fuego_del_Pac%C3%ADfico#/media/File:Pacific_Ring_of_Fire-es.svg)>

Figura 1.2. Placas tectónicas que conforman la República Mexicana.

Disponible en:

<<http://portalweb.sgm.gob.mx/museo/riesgos/tectonica/evolucion-tectonica-mexico>>

Figura 1.3. Zonas sísmicas en México.

Disponible en:

<<http://portalweb.sgm.gob.mx/museo/riesgos/sismos/sismologia-de-mexico>>

Figura 1.5. Onda P.

Figura 1.6. Onda S.

Figura 1.7. Onda Rayleigh.

Figura 1.8. Onda Love.

Disponible en:

<<http://www.smis.org.mx/htm/sm4.htm>>

Figura 1.9. Sismograma.

Disponible en:

<<http://senalesdelostiempos.blogspot.mx/2010/06/un-nuevo-estudio-de-las-ondas-sismicas.html>>

Figura 1.10. Sismógrafo.

Disponible en:

<[https://www.lpi.tel.uva.es/~nacho/docencia/ing\\_ond\\_1/trabajos\\_06\\_07/io3/public\\_html/Sismografo/Sismografo.html](https://www.lpi.tel.uva.es/~nacho/docencia/ing_ond_1/trabajos_06_07/io3/public_html/Sismografo/Sismografo.html)>

Figura 1.11. Sismógrafo moderno.

Disponible en:

<<https://es.wikipedia.org/wiki/Sism%C3%B3metro>>

Figura 1.12. Determinación del epicentro.

Disponible en:

<<http://www2.ssn.unam.mx/website/jsp/Sismo85/sismo85-2.htm>>

Figura 1.13. Acelerograma del terremoto de 1985 en México.

Figura 1.14. Aceleración, velocidad y desplazamiento de la tierra durante el terremoto de 1985.

W. C. Stone et al., Engineering Aspects of the September 19, 1985 México Earthquake (NBS/BSS-165). (29 mayo, 1987). Disponible en:

<[http://www.nist.gov/manuscript-publication-search.cfm?pub\\_id=908821](http://www.nist.gov/manuscript-publication-search.cfm?pub_id=908821)>

## Capítulo 2

Figura 2.2. Variación del factor de amplificación dinámico.

Villaverde R., *Fundamental concepts of earthquake engineering*, primera edición, CRC Press, USA, 2009, p. 156.

Figura 2.9. Respuesta en frecuencia del filtro pasa bajos.

Figura 2.11. Respuesta en frecuencia del filtro pasa altos.

Figura 2.13. Respuesta en frecuencia del filtro pasa banda.

Disponible en:

<[http://www.electronics-tutorials.ws/filter/filter\\_2.html](http://www.electronics-tutorials.ws/filter/filter_2.html)>

Figura 2.16. Respuesta del filtro pasa bajos.

Figura 2.17. Respuesta del filtro pasa altos.

Figura 2.18. Respuesta del filtro pasa banda.

Disponible en:

<<http://cnx.org/contents/pgrh5dEe@15.5:CuW@4/Second-Order-Filters>>

Figura 2.19. Circuito genérico para la síntesis de filtros activos de segundo orden.

Figura 2.20. Topología de realimentación múltiple y ganancia infinita o de Rauch.

Figura 2.21. Filtro pasa bajos con la topología de Rauch.

Figura 2.22. Topología del circuito general VCVS y Sallen-Key.

Figura 2.23. Filtro pasa bajos con topología de Sallen-Key.

Disponible en:

<[http://www2.uca.es/grup-invest/instrument\\_electro/ppjjgdr/Cir\\_An\\_Apl/Cir\\_An\\_Apl\\_arch/temas/T4\\_caa.pdf](http://www2.uca.es/grup-invest/instrument_electro/ppjjgdr/Cir_An_Apl/Cir_An_Apl_arch/temas/T4_caa.pdf)>

Figura 2.28. Error de offset.

Figura 2.29. Error de monotonicidad.

Figura 2.30. Error de linealidad.

Disponible en:

<<http://cnx.org/contents/TZLhERtv@4/Data-Converters>>

Figura 2.31. ADC de rampa.

Figura 2.32. Convertidor de aproximaciones sucesivas.

Figura 2.33. Curva de salida del ADC.

Figura 2.34. Convertidor tipo flash.

Figura 2.35. Convertidor de doble rampa.

Figura 2.37. Convertidor voltaje-frecuencia.

Disponible en:

<[http://quidel.inele.ufro.cl/~jhuircan/PDF\\_CTOSII/ad03.pdf](http://quidel.inele.ufro.cl/~jhuircan/PDF_CTOSII/ad03.pdf)>

Figura 2.41. Modulador Delta.

Figura 2.42. Señales correspondientes a la modulación delta.

Figura 2.43. Convertidor sigma-delta.

Figura 2.44. Señales correspondientes al ADC sigma-delta.

Disponible en:

<[ftp://ftp.ehu.es/cidira/dptos/depjt/Instrumentacion/BK-ANGEL/03\\_ConvertidoresADC\\_DAC/Convertidores\\_sigma\\_delta.PDF](ftp://ftp.ehu.es/cidira/dptos/depjt/Instrumentacion/BK-ANGEL/03_ConvertidoresADC_DAC/Convertidores_sigma_delta.PDF)>

Figura 2.50. Comportamiento de distintos moduladores sigma-delta.

Disponible en:

<<http://www.ti.com/lit/an/slyt423/slyt423.pdf>>

Figura 2.51. Conceptualización de la etapa de diezmado.

Figura 2.52. Elección de la tasa de diezmado.

Disponible en:

<<http://www.ti.com/lit/an/slyt438/slyt438.pdf>>

Figura 2.55. Formato de la cadena de datos.

Disponible en:

<<http://maxembedded.com/2013/09/the-usart-of-the-avr/>>

Figura 2.57. Conexión de módem nulo.

Disponible en:

<[http://www.tervenet.com/itmaz/micros2/PIC32\\_11\\_UART.pdf](http://www.tervenet.com/itmaz/micros2/PIC32_11_UART.pdf)>

Figura 2.61. Configuración de CPOL y CPHA.

Disponible en:

<<http://mindstormsyarduino.blogspot.mx/2011/06/arduino-y-la-comunicacion-spi.html>>

Figura 2.63. Condición de dato válido.

Figura 2.64. Condición de inicio, parada e inicio repetido.

*Atmel, ATmega640/1280/1281/2560/2561 datasheet*, pp. 237,238. Disponible en:

<[http://www.atmel.com/images/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561\\_datasheet.pdf](http://www.atmel.com/images/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf)>

Figura 2.65. Formato de los datos transmitidos en el bus I2C.

Figura 2.66. Secuencia de datos al configurar al esclavo en modo escritura.

Figura 2.67. Secuencia de datos al configurar al esclavo en modo lectura.

*Maxim, DS1307 datasheet*, pp. 11-14. Disponible en:

<<http://datasheets.maximintegrated.com/en/ds/DS1307.pdf>>

Figura 2.69. Celda de memoria DRAM.

Figura 2.70. Celda de memoria SRAM.

Disponible en:

<[https://es.wikipedia.org/wiki/Celda\\_de\\_memoria](https://es.wikipedia.org/wiki/Celda_de_memoria)>

Figura 2.71. Memoria RAM de 4x4 celdas y 6 MOSFETs por celda.

Disponible en:

<[https://moodle.insa-toulouse.fr/pluginfile.php/2632/mod\\_resource/content/0/content/static\\_ram.html](https://moodle.insa-toulouse.fr/pluginfile.php/2632/mod_resource/content/0/content/static_ram.html)>

Figura 2.72. Estructura de una memoria SRAM.

Disponible en:

<[http://www.eetimes.com/document.asp?doc\\_id=1279819](http://www.eetimes.com/document.asp?doc_id=1279819)>

Figura 2.73. Memoria microSD.

Disponible en:

<[http://www.dejazzer.com/ee379/lecture\\_notes/lec12\\_sd\\_card.pdf](http://www.dejazzer.com/ee379/lecture_notes/lec12_sd_card.pdf)>

Figura 2.85. Red NAVSTAR.

Disponible en:

<<http://kobi.nat.uni-magdeburg.de/patrick/pmwiki.php?n=BEng.TheGPSReceiver>>

Figura 2.86. Estructura de la señal GPS.

Figura 2.87. Estaciones del segmento de control.

Disponible en:

<<http://www.caminsdemuntanya.com/publica/docus/GPSBasics.pdf>>

Figura 2.88. Intersección de tres esferas.

Disponible en:

<<http://mason.gmu.edu/~rdonoso/Project2.html>>

Figura 2.90. Diagrama de bloques del microcontrolador MSP430.

*Texas Instruments, MSP430x2xx Family datasheet*, p. 24. Disponible en:

<<http://www.ti.com/lit/ug/slau144j/slau144j.pdf>>

## Capítulo 3

Figura 3.13. Programación de la tasa de sobre muestreo.

Figura 3.14. Distribución de terminales del ADC.

Figura 3.16. Señales manejadas durante la transferencia de datos.

*Linear Technology, LTC2440 datasheet*, pp. 14, 2, 11. Disponible en:

<<http://cds.linear.com/docs/en/datasheet/2440fd.pdf>>

Figura 3.18. Distribución de terminales del microcontrolador.

Figura 3.19. Diagrama de bloques del microcontrolador.

*Atmel, ATmega640/1280/1281/2560/2561 datasheet*, pp. 2, 5.

Figura 3.23. Distribución de terminales de la memoria RAM.

*Microchip, 23LC1024 datasheet*, p. 1. Disponible en:

<<http://ww1.microchip.com/downloads/en/DeviceDoc/25142A.pdf>>

Figura 3.26. Distribución de terminales del receptor GPS.

*GlobalSat Technology Corporation, EM506 datasheet*, p. 3. Disponible en:

<[https://cdn.sparkfun.com/datasheets/GPS/EM506\\_um.pdf](https://cdn.sparkfun.com/datasheets/GPS/EM506_um.pdf)>

Figura 3.28. Distribución de terminales del RTC.

Figura 3.29. Mapa de memoria del RTC.

*Maxim, DS1307 datasheet*, pp. 1, 8.

Figura 3.31. *Joystick*.

Disponible en:

<<https://www.adafruit.com/product/504>>

Figura 3.40. Registros del TCNT3.

Figura 3.41. Especificaciones del modo de operación del contador.

Figura 3.41. Especificaciones de la fuente de reloj para el contador.

*Atmel, ATmega640/1280/1281/2560/2561 datasheet*, pp. 154, 145, 157.

Figura 3.43. Opciones del ADC.

Figura 3.44. Operación del ADC.

Figura 3.45. Formato del dato convertido.

*Linear Technology, LTC2440 datasheet*, pp. 14, 15, 12.

Figura 3.46. Registro de modo de operación de la memoria RAM.

Figura 3.47. Conjunto de instrucciones de la memoria RAM.

*Microchip, 23LC1024 datasheet*, pp. 10, 6.

## Capítulo 4

Figura 4.22. División de la PCB en zonas de acuerdo a la frecuencia manejada.

Figura 4.32. Cálculo de la anchura de las pistas.

*IPC-2221A Standard*, pp. 39, 41. Disponible en:

<[http://sisko.colorado.edu/CRIA/FILES/REFS/Electronics/IPC\\_2221A.pdf](http://sisko.colorado.edu/CRIA/FILES/REFS/Electronics/IPC_2221A.pdf)>

---

# Bibliografía

## Libros

Boylestad R., Nashelsky L., *Electrónica: Teoría de circuitos y dispositivos electrónicos*, octava edición, Prentice Hall, México, 2003.

Dorf R., Svoboda J., *Circuitos eléctricos: Introducción al análisis y diseño*, tercera edición, Alfaomega, México, 2003.

Johnson D., Hilburn J., *Análisis básico de circuitos eléctricos*, cuarta edición, Prentice Hall, USA, 1991.

Manolakis D., Proakis J., *Digital Signal Processing: Principles, algorithms and applications*, tercera edición, Prentice Hall, USA, 2005.

Morton Todd, *Embedded microcontrollers*, primera edición, Prentice Hall, USA, 2001.

Nashelsky L., *Fundamentos de tecnología digital*, primera edición, Limusa, México, 1993.

Newmark N., Rosenblueth E., *Fundamentals of earthquake engineering*, primera edición, Prentice Hall, USA, 1971.

Schilling D., Belove C., *Electronic circuits. Discrete and integrated*, segunda edición, Mc Graw Hill, USA, 2009.

Villaverde R., *Fundamental concepts of earthquake engineering*, primera edición, CRC Press, USA, 2009.

Zerva Aspasia, *Spatial variations of seismic ground motions, modeling and engineering applications*, primera edición, CRC Press, USA, 2009.

## Tesis

Hernández H., *Diseño e implementación de una interfaz de registro de datos para la unidad SR04*, Tesis de licenciatura, UNAM, 2010.

Cuchillo H., Preciado E., *Digitalizador de señales sísmicas sincronizado con un GPS y almacenamiento de datos en una SD card*, Tesis de licenciatura, UNAM, 2013.

## **Recursos electrónicos**

### **ADC sigma-delta**

<http://cnx.org/contents/9OSLzRMI@3.2:TZLhERtv@4/Data-Converters>

### **Protocolo SPI**

<http://www.totalphase.com/support/articles/200349236-SPI-Background>

<http://www.ermicro.com/blog/?p=1050>

<https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi/all.pdf>

[http://www.fmf.uni-lj.si/~ponikvar/STM32F407%20project/SPI\\_Accel.pdf](http://www.fmf.uni-lj.si/~ponikvar/STM32F407%20project/SPI_Accel.pdf)

### **Protocolo I2C**

<https://learn.sparkfun.com/tutorials/i2c/i2c-at-the-hardware-level>

### **Memoria RAM**

[http://www.eetimes.com/document.asp?doc\\_id=1279819](http://www.eetimes.com/document.asp?doc_id=1279819)

[https://moodle.insa-toulouse.fr/pluginfile.php/2632/mod\\_resource/content/0/content/static\\_ram.html](https://moodle.insa-toulouse.fr/pluginfile.php/2632/mod_resource/content/0/content/static_ram.html)

### **SD CARD**

[ftp://ftp.altera.com/up/pub/Altera\\_Material/12.0/Laboratory\\_Exercises/Embedded\\_Systems/DE2/embed\\_lab9.pdf](ftp://ftp.altera.com/up/pub/Altera_Material/12.0/Laboratory_Exercises/Embedded_Systems/DE2/embed_lab9.pdf)

[http://www.dejazzer.com/ee379/lecture\\_notes/lec12\\_sd\\_card.pdf](http://www.dejazzer.com/ee379/lecture_notes/lec12_sd_card.pdf)

[http://embdev.net/attachment/39390/TOSHIBA\\_SD\\_Card\\_Specification.pdf](http://embdev.net/attachment/39390/TOSHIBA_SD_Card_Specification.pdf)

[https://www.sdcard.org/consumers/choices/file\\_system/index.html](https://www.sdcard.org/consumers/choices/file_system/index.html)

### **Sistema GPS**

<https://edis.ifas.ufl.edu/pdffiles/IN/IN65700.pdf>

[http://www.fceia.unr.edu.ar/gps/GGSR/libro\\_gps.pdf](http://www.fceia.unr.edu.ar/gps/GGSR/libro_gps.pdf)

<http://www.caminsdemuntanya.com/publica/docus/GPSBasics.pdf>