



Departamento de Electrónica
Universidad de Málaga

Tesis Doctoral


Aportaciones al Hardware para Sensores Táctiles Inteligentes basados en FPGAs

Óscar Oballe Peinado
Málaga, noviembre 2015



Publicaciones y
Divulgación Científica

AUTOR: Óscar Oballe Peinado

 <http://orcid.org/0000-0003-1185-4864>

EDITA: Publicaciones y Divulgación Científica. Universidad de Málaga



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-SinObraDerivada 4.0 Internacional:

Cualquier parte de esta obra se puede reproducir sin autorización pero con el reconocimiento y atribución de los autores.

No se puede hacer uso comercial de la obra y no se puede alterar, transformar o hacer obras derivadas.

<http://creativecommons.org/licenses/by-nc-nd/4.0/legalcode>

Esta Tesis Doctoral está depositada en el Repositorio Institucional de la Universidad de Málaga (RIUMA): riuma.uma.es



Departamento de Electrónica
Universidad de Málaga

Tesis Doctoral

Aportaciones al Hardware para Sensores Táctiles Inteligentes basados en FPGAs

Presentada por: Óscar Oballe Peinado

Dirigida por: Dr. Fernando Vidal Verdú
Profesor Titular de Universidad del
Departamento de Electrónica
Dr. José Antonio Hidalgo López
Profesor Titular de Universidad del
Departamento de Electrónica

Málaga, noviembre 2015



Dr. Fernando Vidal Verdú y Dr. José Antonio Hidalgo López,
Profesores Titulares de Universidad del Departamento de Electrónica
de la Universidad de Málaga

CERTIFICAN: que D. Óscar Oballe Peinado, Ingeniero en Informática por la
Universidad de Málaga ha realizado bajo nuestra dirección la Tesis
Doctoral que tiene por título “Aportaciones al Hardware para Sensores
Táctiles Inteligentes basados en FPGAs”. Y que ha alcanzado los
objetivos de investigación propuestos, estando debidamente cualificada
para su defensa.

Málaga, a 13 de Noviembre de 2015

Fdo.: Fernando Vidal Verdú

Fdo.: José Antonio Hidalgo López

Para ti, Mamá

Agradecimientos

Mi sincera gratitud a mis directores, Dr. Fernando Vidal Verdú y Dr. José Antonio Hidalgo López. Sin vuestra ayuda e interés, esta tesis difícilmente habría llegado a concluirse en los plazos y forma que hoy tiene.

También agradecer a mis amigos y compañeros Jose y Julián su apoyo y comprensión durante estos años en los que hemos compartido el desafío que representa un trabajo de esta índole.

Deseo expresar mi agradecimiento a mis compañeros de laboratorio, Andrés y Patricia, por su colaboración en no pocas tareas para las que les he solicitado su opinión y ayuda. También agradecer a mis compañeros de departamento las facilidades procuradas durante este último año. A CIDETEC por su colaboración como socio tecnológico del grupo de investigación, en concreto a Estíbalitz, Jaime e Iván. Finalmente, a Guti y Javier de Aucore por su inestimable ayuda en el montaje de los circuitos.

Quiero también dar las gracias a la Universidad de Málaga y a las instituciones que han soportado, a través del programa del Plan Nacional y los fondos FEDER bajo los contratos TEC2009-14446 y TEC2012-38653, el trabajo aquí presentado.

A mis amigos por sus ánimos y apoyo, especialmente a Gabi por sus revisiones de este texto.

Por último, gracias a mi familia, por su paciencia y comprensión durante todo el tiempo que les he robado. Álvaro, Papá ya ha terminado su “trabajo”.

Índice

Introducción.....	1
Objetivos.....	4
Estructura de la Tesis.....	5
Capítulo 1. Antecedentes.....	7
1.1. Contexto.....	7
1.2. Circuitos de conexión directa sensor-microcontrolador.....	15
1.3. Conversión tiempo-digital.....	17
1.3.1. Principio de funcionamiento.....	17
1.3.2. Fuentes de incertidumbre.....	18
1.3.3. Análisis de la incertidumbre estándar.....	19
1.3.4. Análisis de la resolución.....	21
1.4. Medida de resistencias.....	22
1.4.1. Calibración de un punto.....	22
1.4.2. Calibración de dos puntos.....	23
1.4.3. Análisis de la exactitud.....	24
Capítulo 2. Circuitos de conexión directa sensor-FPGA.....	29
2.1. Introducción.....	29
2.2. Módulos de captura.....	31
2.2.1. Módulo de captura 1 (MC1) – Circuito síncrono de detección de cambio de nivel de la señal de descarga.....	32
2.2.2. Módulo de captura 2 (MC2) - Circuito síncrono de detección de cambio de nivel de la señal de descarga con generación de pulso único.....	33
2.2.3. Módulo de captura 3 (MC3) - Circuito de detección de cambio de nivel de la señal de descarga con <i>latch</i> anti-rebote.....	35
2.2.4. Módulo de captura 4 (MC4) - Circuito de detección de inicio de tren de pulsos de la señal de descarga.....	36
2.2.5. Módulo de captura 5 (MC5) - Circuito de detección de inicio de tren de pulsos de la señal de descarga por flancos de subida y de bajada, promediando dos valores de cuenta.....	38

2.2.6. Módulo de captura 6 (MC6) - Circuito de detección de inicio y final de tren de pulsos de la señal de descarga por flancos de subida y de bajada, promediando cuatro valores de cuenta	39
2.3. Análisis de la resolución en la conversión tiempo-digital	43
2.3.1. Materiales y método	44
2.3.2. Resultados experimentales	46
2.3.3. Discusión.....	49
2.3.4. Conclusiones	52
2.4. Análisis de la exactitud en la medida de resistencia	53
2.4.1. Calibración de un punto mediante aproximación polinómica.....	53
2.4.2. Materiales y método	54
2.4.3. Resultados experimentales y discusión	56
2.4.4. Conclusiones	64
2.5. Comparativa con otros circuitos de conexión directa	65
2.5.1. Materiales y método	65
2.5.2. Resultados experimentales	66
2.5.3. Discusión.....	70
2.6. Conclusiones	71
Capítulo 3. Circuitos de conexión directa matriz sensores-FPGA	75
3.1. Introducción	75
3.2. Matrices de pequeño o mediano tamaño.....	77
3.2.1. <i>Crosstalk</i> RC	80
3.2.2. Análisis de la incertidumbre en la medida	83
3.2.3. Método de evaluación de la incertidumbre en las medidas.....	86
3.2.4. Materiales y método	88
3.2.5. Resultados experimentales y discusión	89
3.2.6. Conclusiones	91
3.3. Matrices de gran tamaño	91
3.3.1. Circuitos con realimentación resistiva	92
3.3.2. Circuito con realimentación capacitiva	94
3.3.3. Mejoras para los circuitos con realimentación capacitiva.....	97
3.3.4. Cálculo de R_{ij} teniendo en cuenta las tensiones de <i>offset</i> de los OAs	109
3.3.5. Eliminación de los efectos de las intensidades de polarización de los OAs.....	112

3.3.6. Materiales y método	113
3.3.7. Resultados experimentales y discusión	113
3.3.8. Conclusiones	120
Capítulo 4. Propuesta de arquitectura para un sistema táctil	123
4.1. Introducción	123
4.2. Diagrama de bloques	125
4.3. Módulo de control.....	126
4.3.1. Comunicaciones entre el módulo de control y los sensores táctiles	126
4.3.2. Comunicaciones entre el módulo de control y el sistema <i>host</i>	130
4.4. Sensor Táctil	131
4.4.1. Sensor para el dedo	132
4.4.2. Sensor para la palma	139
4.5. Prototipo del sistema táctil.....	140
4.5.1. Módulo de control + sensor de palma	143
4.5.2. Sensor de dedo	151
4.6. Prototipo para una mano de Barrett	156
4.6.1. Módulo de control + sensor de palma	158
4.6.2. Sensor de dedo	162
4.7. Pre-procesado: Calculo de los momentos y la elipse de una imagen	168
4.7.1. Momentos Simples.....	168
4.7.2. Centroide y momentos centrales	171
4.7.3. Elipse de una imagen	175
4.8. Resultados y discusión.....	177
Conclusiones y líneas futuras.....	181
Conclusiones.....	181
Líneas futuras.....	182
Publicaciones que avalan la Tesis.....	184
Anexo A. Módulo de captura.....	187
Anexo B. Esquemáticos	193
Anexo C. Diseño mecánico del dedo para la mano de Barrett	215
Bibliografía.....	219

Índice de Figuras

Figura 1-1. Diagrama funcional de un sistema táctil.	8
Figura 1-2. Circuito basado en un microcontrolador [Vidal-Verdú, Oballe-Peinado, et al. 2011].	11
Figura 1-3. Diafonía originada por caminos resistivos parásitos [Vidal-Verdú, Oballe-Peinado, et al. 2011].	11
Figura 1-4. Circuito basado en un PSoC [Vidal-Verdú et al. 2010].	12
Figura 1-5. Sistema de control de la mano robótica HIRO III [Kawasaki 2015].	14
Figura 1-6 Circuito de interfaz directa sensor-microcontrolador.	15
Figura 1-7. Fases de carga y descarga del condensador.	16
Figura 1-8. Fases de carga y descarga por cada ciclo de medida.	16
Figura 1-9. Circuito de interfaz directa sensor-microcontrolador con una resistencia de calibración.	22
Figura 1-10. Curva de calibración de un punto.	23
Figura 1-11. Circuito de interfaz directa sensor-microcontrolador con dos resistencias de calibración.	23
Figura 1-12. Curva de calibración de dos puntos.	24
Figura 1-13. Circuito de interfaz directa sensor resistivo y μ C cuando se considera la resistencia interna de los puertos del μ C. Fase de carga (a) y fase de descarga (b).	25
Figura 2-1. Transitorio a la salida del <i>buffer</i> de entrada cuando el cambio de nivel es detectado.	31
Figura 2-2. Circuito de detección del cambio de nivel en la señal de descarga (MC1).	32
Figura 2-3. Cronograma para el circuito de detección de cambio de nivel en la señal de descarga (MC1).	32
Figura 2-4. Cronograma de la detección de flanco de bajada con entrada ruidosa.	33
Figura 2-5. Circuito de detección de cambio de nivel con generación de pulso único (MC2).	34
Figura 2-6. Cronograma de la detección de cambio de nivel con generación de pulso único (MC2).	34
Figura 2-7. Circuito de detección, con flanco de subida, del cambio de nivel en la salida del <i>latch</i> (MC3).	35
Figura 2-8. Cronograma de la detección, con flanco de subida, del cambio de nivel en la salida del <i>latch</i> (MC3).	36
Figura 2-9. Cronograma de la detección de un glitch previo al tren de pulsos.	36
Figura 2-10. Circuito modificado de detección, con flanco de subida, del cambio de nivel en la salida del <i>latch</i> (MC4).	37
Figura 2-11. Cronograma de la detección de inicio de tren de pulsos por el módulo de captura MC4.	37
Figura 2-12. Circuito modificado de detección, con flanco de bajada, del cambio de nivel en la salida del <i>latch</i> (MC5).	38
Figura 2-13. Cronograma de la detección de inicio de tren de pulsos por flancos de subida y bajada.	39
Figura 2-14. Circuito de detección de final de tren de pulsos por flanco de subida (MC6).	40
Figura 2-15. Circuito de detección de final de tren de pulsos por flanco de bajada (MC6).	40
Figura 2-16. Cronograma de la detección de final de tren de pulsos por flancos de subida y bajada.	41

Figura 2-17. Captura del circuito de detección de inicio de tren de pulsos (circuitos Figura 2-10 y Figura 2-12).....	42
Figura 2-18. Captura del circuito de detección de final de tren de pulsos (circuitos Figura 2-14 y Figura 2-15).....	42
Figura 2-19. Captura del circuito de detección de inicio y final de tren de pulsos (circuitos Figura 2-10, Figura 2-12, Figura 2-14 y Figura 2-15).	43
Figura 2-20. MC1 - Histograma para 500 muestras sobre una resistencia de 7350 Ω	46
Figura 2-21. MC2 - Histograma para 500 muestras sobre una resistencia de 7350 Ω	47
Figura 2-22. MC3 - Histograma para 500 muestras sobre una resistencia de 7350 Ω	47
Figura 2-23. MC4 - Histograma para 500 muestras sobre una resistencia de 7350 Ω	48
Figura 2-24. MC5 - Histograma para 500 muestras sobre una resistencia de 7350 Ω	48
Figura 2-25. MC6 - Histograma para 500 muestras sobre una resistencia de 7350 Ω	49
Figura 2-26. Desviación estándar experimental $s(X)$ en función de la constante de tiempo para el módulo MC6.	50
Figura 2-27. Incertidumbre estándar relativa en función de la constante de tiempo para el módulo MC6. 51	
Figura 2-28. Circuito con resistencias de calibración.	55
Figura 2-29. Curva de ajuste por mínimos cuadrados.	56
Figura 2-30. Máximo error relativo experimental para la calibración de un punto.	58
Figura 2-31. Máximo error relativo experimental para la calibración de dos puntos.	59
Figura 2-32. Máximo error relativo experimental para la calibración CLChar8.	60
Figura 2-33. Máximo error relativo experimental para la calibración CLChar2.	61
Figura 2-34. Máximo error relativo experimental para la calibración CSQChar8.	62
Figura 2-35. Máximo error absoluto para la calibración de un punto.....	63
Figura 2-36. Máximos errores absolutos para la calibración de dos puntos y las técnicas de calibración propuestas.	63
Figura 2-37. Máximo error relativo para la calibración de un punto.	64
Figura 2-38. Máximos errores relativos para la calibración de dos puntos y las técnicas de calibración propuestas.	64
Figura 2-39. Máximo error relativo experimental para la calibración de un punto.	67
Figura 2-40. Máximo error relativo experimental para la calibración de dos puntos.	68
Figura 2-41. Máximo error relativo experimental para la calibración con el polinomio de grado uno (CLChar2).....	69
Figura 3-1. Matriz de sensores resistivos con conexiones compartidas para filas y columnas.....	76
Figura 3-2. Circuito de interfaz directa – Integrador Pasivo.	78
Figura 3-3. Detalle del circuito de la columna j mostrando las capacidades parásitas.	80
Figura 3-4. Circuito de calibración de dos puntos mostrando las capacidades parásitas.	82
Figura 3-5. Circuito con realimentación resistiva y tierra virtual.	92
Figura 3-6. Circuito con realimentación capacitiva y tierra virtual.	94
Figura 3-7. Ciclos de CARGA y DESCARGA/MEDIDA para la fila i columna j	95
Figura 3-8. Ejemplo de medida de tiempo del circuito con realimentación capacitiva.	96

Figura 3-9. Circuito para considerar las resistencias de <i>buffers</i> y evitar errores de medida en C_j y VT_j	98
Figura 3-10. Circuito equivalente para escanear los valores de las resistencias de la fila i	99
Figura 3-11. Circuito equivalente para la activación de una única fila.....	106
Figura 3-12. Detalle del circuito para el cálculo de las intensidades durante la activación de la fila i	106
Figura 3-13. Modificación propuesta para la evaluar el efecto de las tensiones de <i>offset</i> y las intensidades de polarización de los OAs.	110
Figura 3-14. Errores relativos de R_{ij} calculados según la ecuación (3.30).....	115
Figura 3-15. Máximos errores relativos de R_{ij} calculados según las ecuaciones (3.46) (azul) y (3.66) (rojo) con $RL = 560 \Omega$ $R_C = 750 \Omega$ y $RS = 100 \Omega$	117
Figura 3-16. Máximos errores relativos de R_{ij} calculados según las ecuaciones (3.46) (azul) y (3.67) (rojo) con $RL = 3,3 \text{ k}\Omega$ $R_C = 6,8 \text{ k}\Omega$ y $RS = 100 \Omega$	119
Figura 3-17. Errores sistemáticos de R_{ij} calculados según las ecuaciones (3.46) (azul) y (3.67) (rojo), con $RL = 560 \Omega$ $R_C = 750 \Omega$ y $RS = 100 \Omega$	120
Figura 3-18. Errores sistemáticos de R_{ij} calculados según las ecuaciones (3.46) (azul) y (3.67) (rojo), con $RL = 3,3 \text{ k}\Omega$ $R_C = 6,8 \text{ k}\Omega$ y $RS = 100 \Omega$	120
Figura 4-1. Diagrama de bloques de la arquitectura del sistema.....	126
Figura 4-2. Cronograma del protocolo SPI entre el módulo de control y un sensor.....	130
Figura 4-3. Diagrama de bloques del sensor del dedo.....	132
Figura 4-4. Diagrama de bloques del sensor de la palma.....	140
Figura 4-5. Diagrama de bloques del prototipo del sistema táctil.....	141
Figura 4-6. Prototipo del sistema táctil.....	142
Figura 4-7. Diagrama de bloques del módulo de control + sensor de palma.....	143
Figura 4-8. Módulo de control + sensor de palma.....	144
Figura 4-9. Conector FPC/FFC y matriz de electrodos de la matriz táctil.....	150
Figura 4-10. Diagrama de bloques del sensor de dedo.....	151
Figura 4-11. Sensor de dedo con matriz de resistencias.....	152
Figura 4-12. Sensor de dedo con matriz de electrodos.....	153
Figura 4-13. Detalle de la lámina de material discreto sobre la matriz de electrodos.....	155
Figura 4-14. Mano de Barrett.....	156
Figura 4-15. Prototipo de sistema táctil para la mano de Barrett.....	157
Figura 4-16. Módulo de control + sensor de palma – Mano de Barrett.....	159
Figura 4-17. Cadena JTAG del sistema táctil para la mano de Barrett.....	160
Figura 4-18. Matriz de electrodos para la palma de la mano de Barrett con 10 filas y 16 columnas.....	161
Figura 4-19. Matriz de electrodos para la palma de la mano de Barrett con 5 filas y 8 columnas.....	162
Figura 4-20. Detalle de la placa Rigiflex del sensor de dedo.....	163
Figura 4-21. Componentes del sensor de dedo para la mano de Barrett.....	165
Figura 4-22. Detalles de la matriz de electrodos del sensor de dedo así como del material discreto.....	166
Figura 4-23. Capas PCB del sensor de dedo.....	166
Figura 4-24. Sensor de dedo para la mano de Barrett.....	167
Figura 4-25. Bloque aritmético para el cálculo de los momentos simples.....	169

Figura 4-26. Bloque aritmético para el cálculo del centroide y de los momentos centrales.	172
Figura 4-27. Configuración para el cálculo de la coordenada X del centroide.	173
Figura 4-28. Configuración para el cálculo de la coordenada Y del centroide.	173
Figura 4-29. Configuración para el cálculo del momento central MC_{20}	174
Figura 4-30. Configuración para el cálculo del momento central MC_{11}	174
Figura 4-31. Configuración para el cálculo del momento central MC_{02}	175
Figura 4-32. Bloque aritmético para el cálculo de la raíz cuadrada y el ángulo.....	176
Figura 4-33. Cronograma de la simulación del circuito de pre-procesado.	178
Figura 4-34. Imagen táctil del sistema táctil con las elipses calculadas de forma local.	179
Figura B-1. Mini Integrador Activo – Amplificadores Operacionales (1/4).....	194
Figura B-2. Mini Integrador Activo – Amplificadores Operacionales (2/4).....	195
Figura B-3. Mini Integrador Activo – Amplificadores Operacionales (3/4).....	196
Figura B-4. Mini Integrador Activo – Amplificadores Operacionales (4/4).....	197
Figura B-5. Mini Integrador Activo – FPGA Sensor Banco 0 E/S.....	198
Figura B-6. Mini Integrador Activo – FPGA Sensor Banco 1 E/S.....	199
Figura B-7. Mini Integrador Activo – FPGA Sensor Banco 2 E/S.....	200
Figura B-8. Mini Integrador Activo – FPGA Sensor Banco 3 E/S.....	201
Figura B-9. Mini Integrador Activo – Circuito de Alimentación.....	202
Figura B-10. Mini Integrador Activo – Cadena JTAG.....	203
Figura B-11. Mini Integrador Activo – Controlador USB.....	204
Figura B-12. Mini Integrador Activo – Puertos de Expansión SPI, conector matriz de electrodos y sensor de temperatura.....	205
Figura B-13. Mini Integrador Pasivo – Matriz de electrodos y resistencias de calibración.....	206
Figura B-14. Mini Integrador Pasivo – FPGA Sensor Banco 0 E/S.....	207
Figura B-15. Mini Integrador Pasivo – FPGA Sensor Banco 1 E/S.....	208
Figura B-16. Mini Integrador Pasivo – FPGA Sensor Banco 2 E/S.....	209
Figura B-17. Mini Integrador Pasivo – FPGA Sensor Banco 3 E/S.....	210
Figura B-18. Mini Integrador Pasivo – Pines de alimentación FPGA Sensor.....	211
Figura B-19. Mini Integrador Pasivo – Cadena JTAG y EEPROM.....	212
Figura B-20. Mini Integrador Pasivo – Circuito de Alimentación.....	213
Figura C-1. Diseño mecánico completo del dedo para la mano de Barrett.....	215
Figura C-2. Detalles del interior del dedo para la mano de Barrett.	216
Figura C-3. Piezas internas impresas en ABS.	217
Figura C-4. Comparativa entre el dedo de la propuesta y el dedo original.....	217

Índice de Tablas

Tabla 2-1. Análisis de la incertidumbre de los módulos de captura propuestos ($R_x = 7350 \Omega$).....	50
Tabla 2-2. Recursos hardware empleados por los distintos módulos de captura.	52
Tabla 2-3. Resumen de resultados para la calibración de un punto.	57
Tabla 2-4. Resumen de resultados para la calibración de dos puntos.	58
Tabla 2-5. Resumen de resultados para la calibración con el polinomio de grado uno (CLChar8).....	59
Tabla 2-6. Resumen de resultados para la calibración con el polinomio de grado uno (CLChar2).....	60
Tabla 2-7. Resumen de resultados para la calibración con el polinomio de grado dos (CSQChar8).	61
Tabla 2-8. Error absoluto máximo y bits de resolución de la medida en ohmios.	62
Tabla 2-9. Análisis de la incertidumbre y la resolución para un sensor PT-1000 ($R_x = 2200 \Omega$).	66
Tabla 2-10. Análisis de la exactitud y la precisión para la calibración de un punto para el sensor PT-1000.	67
Tabla 2-11. Análisis de la exactitud y la precisión para la calibración de dos puntos para el sensor PT- 1000.	68
Tabla 2-12. Análisis de la exactitud y la precisión para la calibración con el polinomio de grado uno (CLChar2) para el sensor PT-1000.	69
Tabla 2-13. Error absoluto máximo para las tres calibraciones.	70
Tabla 2-14. Comparativa entre el circuito propuesto y otras implementaciones para el rango de resistencias del sensor de temperatura PT-1000 (* peor caso, ** promediando 500 medidas).....	71
Tabla 3-1. Incertidumbre en la medida de tiempo para las resistencias de test con $C_j = 47 \text{ nF}$ y $Q = 500$. 89	
Tabla 3-2. Degradación en la resolución debida al incremento del número de filas.	90
Tabla 3-3. Datos de exactitud para las distintas técnicas del calibración ($C_j = 47 \text{ nF}$ y $Q = 500$).	90
Tabla 3-4. Exactitud y precisión para el experimento 1.	114
Tabla 3-5. Exactitud y precisión para el experimento 2.	115
Tabla 3-6. Exactitud y precisión para el experimento 3 usando la ecuación (3.46) para la estimación de R_{ij}	116
Tabla 3-7. Exactitud y precisión para el experimento 3 usando la ecuación (3.67) para la estimación de R_{ij}	117
Tabla 3-8. Exactitud y precisión para el experimento 4 usando la ecuación (3.46) para la estimación de R_{ij}	118
Tabla 3-9. Exactitud y precisión para el experimento 4 usando la ecuación (3.67) para la estimación de R_{ij}	119
Tabla 4-1. Comparativa entre distintas soluciones táctiles para manos robóticas.	180

Lista de Acrónimos

ABS	<i>Acrylonitrile Butadiene Styrene</i>
ASIC	<i>Application-Specific Integrated Circuit</i>
BGA	<i>Ball Grid Array</i>
CAN	<i>Controller Area Network</i>
CMOS	<i>Complementary Metal-Oxide-Semiconductor</i>
CPLD	<i>Complex Programmable Logic Device</i>
CPU	<i>Central Processing Unit</i>
DSP	<i>Digital Signal Processor</i>
EEPROM	<i>Electrically Erasable Programmable Read-Only Memory</i>
ENOB	<i>Effective Number of Bits</i>
FIFO	<i>First In First Out</i>
FPC/FFC	<i>Flat Panel Cable/Flexible Flat Cable</i>
FPGA	<i>Field Programmable Gate Array</i>
FR4	<i>Flame Retardant 4</i>
I2C	<i>Inter-Integrated Circuit</i>
IP	<i>Intellectual Property</i>
ISO	<i>International Organization for Standardization</i>
JTAG	<i>Joint Test Action Group</i>
LAN	<i>Local Area Network</i>
LVDS	<i>Low-Voltage Differential Signaling</i>
MISO	<i>Master Input Slave Output</i>
MOSI	<i>Master Output Slave Input</i>
NMOS	<i>Negative-channel Metal-Oxide Semiconductor</i>
OA	<i>Operational Amplifier</i>
PCA	<i>Principal Component Analysis</i>
PCB	<i>Printed Circuit Board</i>
PEDOT	<i>Poli(3,4-etilenodioxitiofeno)</i>
PET	<i>PolyEthylene Terephtalate</i>
PROM	<i>Programmable Read-Only Memory</i>
PSoC	<i>Programable System on Chip</i>
RAM	<i>Random Access Memory</i>
ROS	<i>Robot Operating System</i>
SM-BUS	<i>System Management Bus</i>
SoC	<i>System on Chip</i>
ST	<i>Schmitt Trigger</i>
UART	<i>Universal Asynchronous Receiver-Transmitter</i>
USB	<i>Universal Serial Bus</i>
VHDL	<i>VHSIC Hardware Description Language</i>
VHSIC	<i>Very High Speed Integrated Circuit</i>

Introducción

Los sensores táctiles son básicamente matrices de unidades que miden fuerza llamadas tácteles, y su salida es un mapa de fuerzas o imagen táctil. La creciente madurez de las tecnologías permite abordar la construcción de sistemas complejos capaces de trabajar en entornos no estructurados. Los sensores táctiles se incorporan en estos sistemas en distintas formas, y en particular como pieles artificiales de robots humanoides. El interés en estos robots tiene orígenes diversos, pero una causa notable es el envejecimiento de las sociedades occidentales y la necesidad de buscar tecnologías que ayuden a prolongar la autonomía de las personas. Como muestra de este interés, es de destacar la reciente publicación de varios artículos de revisión relacionados, como por ejemplo [Tiwana et al. 2012], donde se hace una revisión de las tecnologías de sensado táctil y su aplicación en ingeniería biomédica, o [Dahiya et al. 2013], donde se analizan los aspectos necesarios para una aplicación efectiva de la tecnología de sensado táctil. En [Yan et al. 2014] se pone de manifiesto la importancia de esta tecnología en la interacción hombre-robot, necesaria por ejemplo en un robot de asistencia. Otras revisiones recientes se pueden encontrar en [Dahiya et al. 2010] o [Kappassov et al. 2015], en este último caso orientada a manos robóticas.

Las dificultades en la adquisición y procesado de grandes cantidades de datos de muchos sensores táctiles es una de las razones identificadas para la lenta penetración y aplicación efectiva de estos con respecto a otros sensores [Dahiya et al. 2013]. A esto hay que añadir la complejidad del proceso de manipulación por la cantidad y variedad de contactos que se establecen de forma dinámica [Dahiya et al. 2012]. Para un control realimentado estable y suave de robots humanoides se requiere un periodo de muestreo de 2-100ms [Mukai et al. 2008]. Por tanto, una gran cantidad de datos se deben adquirir, procesar y transmitir en poco tiempo. Los requerimientos dinámicos son más estrictos para detección

de eventos como el deslizamiento (*slippage*) [Maldonado-Lopez et al. 2009]. Para la comunicación, los buses serie se suelen usar porque ocupan poco espacio y crean pocas interferencias mecánicas y eléctricas. Son comunes buses como el I2C o el CAN. Sin embargo estos buses no garantizan una respuesta suficientemente rápida, por ejemplo la tasa de muestreo de la piel artificial en el dedo iCub baja de 100 a 25 Hz conforme crece el número de tácteles [Kappassov et al. 2015]. El bus CAN garantiza una latencia máxima y es muy robusto frente a interferencias y fallos, sin embargo no es adecuado para transmitir grandes cantidades de datos.

Los sensores táctiles comerciales para manos como las de Barrett, Schunk o Shadow usualmente implementan solo primitivas de adquisición y acondicionamiento en una electrónica local basada en microcontroladores o PSoCs. Otras propuestas de laboratorios como [Göger et al. 2009] o [Ito et al. 2012] añaden sensores (piezoeléctricos u ópticos) para detectar el deslizamiento y hacen procesamiento local, normalmente para enviar datos críticos a través de bus CAN y datos no procesados (conjunto de imágenes táctiles) por otros buses como USB. Esta aproximación jerárquica incluye nodos a nivel de mano (además de a nivel de dedo) para hacer funciones de control del agarre de forma autónoma.

Por tanto un pre-procesado básico se hace en la electrónica local para permitir transmitir solo la información necesaria, y en todo caso se usa otro bus como LAN o USB para transmitir los datos sin pre-procesar para las tareas de alto nivel de control y organización, donde no se requiere una respuesta tan rápida. Otros buses menos comunes en robots humanoides como EtherCAT sí pueden transmitir grandes cantidades de datos con una latencia de pocos milisegundos, aunque el pre-procesado de los datos próximo al sensor puede reducir significativamente el tráfico de información entre los sensores y los controladores. Esta arquitectura distribuida libera a la unidad central de procesamiento para poder realizar tareas inteligentes de alto nivel y permite incrementar la complejidad del sistema aumentando el número de sensores.

Partiendo de la base de que los sensores táctiles están sometidos a numerosas fuentes de error y deben hacerse labores de calibrado, además de las de adquisición de datos y las primitivas para el algoritmo de control, la electrónica local debe ser capaz de todo ese procesamiento en tiempo real. Esta tesis propone el uso de *Field Programmable Gate Arrays* (FPGAs) como único dispositivo para hacer la adquisición y procesado. Las

FPGAs se han propuesto en combinación con microcontroladores para hacer esta tarea [Göger *et al.* 2006], pero aquí se propone explotar el gran número de puertos de entrada/salida de las FPGAs, junto con su capacidad para implementar tareas que se ejecutan en paralelo para proponer y evaluar hardware para la adquisición en paralelo de la salida del sensor táctil.

Concretamente, la aproximación que se propone, dado que la mayoría de las FPGAs no implementan convertidores A/D, es la conexión directa sensor-FPGA, que se ha estudiado en detalle en el caso de conexiones sensor-microcontrolador [Reverter y Pallàs-Areny 2005]. En el caso más sencillo, cada táctel de un sensor piezorresistivo puede verse como una resistencia variable con la fuerza, y junto con un condensador forma una red RC como único elemento necesario externo a la FPGA. Aunque la necesidad de calibración lleva a añadir al menos otra resistencia, la celda básica es muy sencilla y permite una realización muy compacta. Si a esto se añade la posibilidad de replicar esta celda y el hardware asociado en la FPGA, tenemos una vía sencilla de realización de adquisición en paralelo. Sin embargo, la necesidad de eliminar la diafonía entre tácteles implica el uso de puertos y pistas de la placa de circuito impreso (PCB) independientes para cada táctel, lo que complica el cableado y requiere PCBs de más capas. Una propuesta alternativa sustituye el integrador pasivo formado por la red RC por un integrador activo basado en un amplificador operacional. Esta alternativa explota la realimentación negativa en el integrador de forma idéntica a como se hace en un circuito común de adquisición para sensores táctiles piezorresistivos, que incorpora amplificadores para cancelar las corrientes parásitas y disminuir la diafonía. Esta segunda propuesta permite direccionar el sensor en forma de filas y columnas, lo que disminuye el cableado.

En cuanto al llamado módulo de captura o hardware implementado en la FPGA para cada canal paralelo de adquisición, se hacen varias propuestas orientadas a aumentar la resolución. Por una parte se busca superar la carencia de seguidores de tipo Schmitt Trigger, presentes en la conexión directa sensor-microcontrolador, cuya histéresis reduce la incertidumbre en la medida, y por otra se usa la gran versatilidad del hardware de la FPGA, que permite por ejemplo usar ambos flancos de la señal de reloj, para proponer módulos que reducen aún más la incertidumbre. Otras contribuciones se hacen para reducir las limitaciones y errores en las implementaciones propuestas.

Finalmente, en esta tesis se integran las propuestas anteriores en un conjunto de sensores táctiles para mano de Barrett. Esta propuesta requiere solucionar múltiples problemas concretos relativos a un diseño suficientemente compacto de la electrónica y que permita el conexionado entre los módulos sensores sin dificultar la manipulación. El sistema ejemplo que se presenta es capaz de transmitir el conjunto de imágenes táctiles para tres falanges y la palma, junto con el cálculo de los momentos de la imagen (hasta el de orden 2) y los parámetros de la elipse asociada, a una razón de 200 fps (*frames* por segundo). Esta tasa es competitiva frente a otras comerciales o de prototipos de investigación [Kappassov *et al.* 2015], pero además hay que destacar que su potencial es muy superior. Por una parte se puede hacer un diseño que permita una tasa de adquisición mayor, pero fundamentalmente esta propuesta abre la puerta a implementar un gran número de primitivas en la electrónica local basada en FPGA, como por ejemplo la detección del deslizamiento, el cálculo de los momentos de la imagen táctil o las componentes principales de la imagen (PCA), o las primitivas para detectar el tipo de objeto y las características de la superficie de contacto como textura y elasticidad, de forma similar a como se hace con otros sensores matriciales como los chips de visión [Díaz, J. *et al.* 2008]. Esto permite tener una electrónica muy potente capaz de encarar aplicaciones más exigentes en complejidad y respuesta dinámica.

A continuación se especifican los objetivos concretos de esta tesis y se describe la estructura de la memoria.

Objetivos

Los objetivos concretos de esta tesis que se desprenden de lo expuesto anteriormente son:

- Implementar y evaluar la conexión directa de un sensor táctil con una FPGA basada en un integrador pasivo o red RC, en especial desde el punto de vista de su rango dinámico, la respuesta dinámica y su capacidad de integración en un espacio reducido.
- Implementar y evaluar la conexión directa de un sensor táctil con una FPGA basado en un integrador activo. Además de los criterios de evaluación compartidos con la conexión basada en integrador pasivo, en este caso cobran

importancia los errores de diafonía (*crosstalk*) y las estrategias de disminución de los mismos.

- Proponer módulos de captura que exploten la versatilidad y recursos de la FPGA para reducir la incertidumbre y aumentar el rango dinámico. Entre estas facilidades cabe destacar la posibilidad del uso de ambos flancos de la señal de reloj y de circuitos que introduzcan histéresis.
- Proponer estrategias de calibración que exploten la capacidad de computación en paralelo de la FPGA para reducir el hardware externo a la misma y conseguir una implementación más compacta.
- Integrar las propuestas en un conjunto completo de sensores táctiles para una mano comercial de Barrett al objeto de mostrar su viabilidad y potencial en aplicaciones de manipulación.

Estructura de la Tesis

La presente tesis doctoral está dividida en 4 capítulos y las conclusiones. A continuación se describe de forma breve cada uno de ellos:

- El primer capítulo presenta el estado del arte de los sistemas táctiles desde un punto de vista del hardware con el que han sido desarrollados. Esto incluye los circuitos de acondicionamiento y adquisición empleados, los posibles algoritmos de pre-procesado que implementan y su estructura a nivel de arquitectura (comunicaciones entre sensores y transmisión de datos hacia un sistema *host*). También se describen conceptos básicos relativos a la conversión tiempo-digital y los principios de funcionamiento de las conexiones directas entre sensor y microcontrolador.
- En el segundo capítulo se describe un módulo de captura para una conexión directa sensor-FPGA que reduce la incertidumbre debida al ruido en la conversión tiempo-digital. Un nuevo método de calibración para la estimación del valor de resistencia del sensor es propuesto y evaluado.
- El tercer capítulo trata la conexión directa entre matrices de sensores y FPGAs. Dos aproximaciones distintas son abordadas: la primera, basada en el circuito integrador pasivo, se presenta como una solución recomendada para matrices de

pequeño tamaño. La segunda, toma como base el circuito integrador activo y está indicada en casos donde las matrices táctiles poseen gran cantidad de tácteles. Un estudio sobre los efectos de segundo orden que afectan a la medida de tiempo para ambas propuestas es llevado a cabo, aportando soluciones que los minimizan.

- En el cuarto capítulo se propone una arquitectura para un sistema táctil y se ofrecen dos implementaciones del mismo: una primera para su uso en un entorno de laboratorio que posibilita su caracterización así como la realización de pruebas funcionales relativas a comunicaciones locales y de pre-procesado, y otra versión que puede ser integrada en una mano robótica y empleada en tareas de manipulación.
- Una sección de Conclusiones resume las aportaciones realizadas, destacando los resultados obtenidos y mostrando posibles líneas de trabajo futuras.

También se incluyen varios anexos con información relativa a los sistemas táctiles desarrollados. En el Anexo A se adjunta el código VHDL de uno de los módulos de captura para la conversión tiempo-digital presentados en el capítulo 2, y en el Anexo B se muestran los esquemas de los circuitos del sistema táctil propuesto para la mano de Barrett [*Barrett Technology, Inc 2015*] en el capítulo 4. Por último, el Anexo C presenta el diseño mecánico del dedo realizado para dicha mano.

Capítulo 1. Antecedentes

1.1. Contexto

En este capítulo se introducen el contexto y los conceptos básicos relacionados con los trabajos realizados en esta tesis. Dado el interés creciente en investigación y aplicación en sensores táctiles, se han publicado recientemente numerosos artículos de revisión, [Tiwana et al. 2012; Yogeswaran et al. 2015; Dahiya et al. 2013; Silvera-Tawil et al. 2015; Yan et al. 2014]. Una revisión actual y profunda de los antecedentes y situación de la investigación y aplicación de los sensores táctiles se puede encontrar en los anteriores artículos. Aquí se hará un repaso de los aspectos más importantes que permitan seguir la tesis y situarla en el contexto general y en el particular del desarrollo del trabajo de investigación que presenta. Para hacer una primera contextualización del trabajo, en la Figura 1-1 se muestra un diagrama de bloques de un sistema táctil [Dahiya et al. 2013]. El trabajo de investigación aquí mostrado se centra en la segunda etapa, es decir en acondicionamiento y adquisición de datos, aunque la justificación de la aproximación propuesta, basada en FPGAs, se entiende mejor al ver que el sensor inteligente resultante de la propuesta es idóneo para la implementación de las etapas posteriores de pre-procesado y comunicación en sistemas, como las manos artificiales, que trabajan en tiempo real.

Los sensores táctiles pueden estar basados en distintos principios de transducción (capacitivos, resistivos u ópticos) y son empleados para un gran número de aplicaciones que van desde la robótica industrial hasta la rehabilitación médica [Dahiya et al. 2010].

La realización de la electrónica del sensor depende de ambas cosas, del principio de transducción y de la aplicación a la que va destinado, aunque existen ciertos factores comunes a todos como son las conexiones, la diafonía, o las capacidades parásitas.

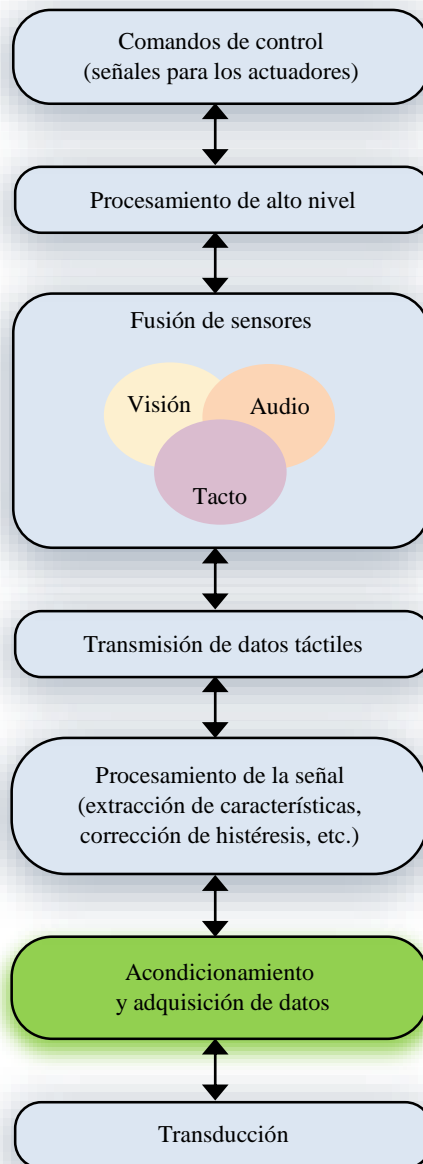


Figura 1-1. Diagrama funcional de un sistema táctil.

Asimismo, el esquema general de un circuito de acondicionamiento incluye una etapa de amplificación, una adaptación de impedancias entre las señales de entrada y salida, el establecimiento de valores de referencia DC (por ejemplo, señal de tierra) y un filtrado paso bajo o paso de banda (filtro anti-aliasing) [Dahiya *et al.* 2011; Vidal-Verdú, Barquero, *et al.* 2011; Cotton *et al.* 2007]. La salida del circuito de acondicionamiento es la entrada al convertidor analógico/digital. El direccionamiento de cada uno de los canales de acondicionamiento puede dar lugar a un conexionado de gran complejidad para sensores de gran área. Algunas soluciones sólo permiten un direccionamiento en serie de

los distintos tácteles, reduciendo así la complejidad a costa de aumentar el tiempo de escaneo [Ohmura et al. 2006]. Una alternativa para el acondicionamiento de señales y la adquisición de datos es la de traducir la señal detectada a un valor en tiempo o frecuencia para que sea posteriormente adquirida por un contador digital. La ventaja de este enfoque es la robustez de la señal de salida del sensor (inmunidad al ruido), aunque el tiempo de adquisición puede ser largo ya que depende de la frecuencia de oscilación y de la resolución del contador. Una aproximación capacitiva con elementos discretos sobre PCB flexible se reporta en [Ulmen et al. 2010], mientras que en [Tee et al. 2015] se implementa un sensor resistivo con osciladores hechos con electrónica orgánica. Sin embargo, la aproximación más común es aquella en la que el sensor táctil se trata como una matriz y se accede a su información organizándolo en filas y columnas. Los distintos canales de acondicionamiento empleados, por ejemplo uno por cada columna de la matriz, son multiplexados en el tiempo a través de un multiplexor analógico a la entrada del convertidor A/D.

La electrónica responsable del sensado y el procesado debe estar situada próxima al sensor para evitar problemas causados por el cableado, y debe emplear el menor número de circuitos integrados y de conexiones para poder ser integrada en manos y pinzas robóticas. Varios sensores táctiles han sido desarrollados usando una tecnología que permite la incorporación de la circuitería sobre el propio sustrato del sensor [Kane et al. 2000; Leineweber et al. 2000; Dahiya et al. 2009; Raibert et al. 1982]. Estas realizaciones incluyen la electrónica de acondicionamiento consiguiendo una alta resolución espacial, que las hacen muy adecuadas en aplicaciones médicas (cirugía mínimamente invasiva), sin dejar de lado las aplicaciones de índole industrial [Leineweber et al. 2000]. La circuitería implementada en el mismo sustrato realiza el acondicionamiento de señal. Esto es frecuente en el caso de sensores capacitivos, porque la influencia de los condensadores parásitos es mayor, y se pueden implementar amplificadores basados en condensadores en conmutación [Leineweber et al. 2000]. Otros circuitos para facilitar la detección de fuerzas normales y tangenciales se realizan también *on-chip*, así como los multiplexores para direccionar la matriz. Procesamientos más complejos, como los basados en convoluciones, se han implementado también en el mismo sustrato [Raibert et al. 1982].

Para aplicaciones en las que se requieren sensores táctiles que cubran áreas grandes, como en robótica, la aproximación más común es la de un sensor compuesto de módulos que contienen circuitos integrados en una placa de circuito impreso flexible. Estos módulos pueden cubrir grandes áreas conformando una red de sensores inteligentes que se comunican con una unidad central de procesamiento a través de un bus serie [Göger *et al.* 2009; Cannata *et al.* 2008; Mittendorfer *et al.* 2011]. Distintos dispositivos pueden ser usados como núcleo de estos módulos.

Una primera opción es el diseño e implementación de circuitos de aplicación específica (ASICs) [Iwashita *et al.* 2004; Maldonado-Lopez *et al.* 2009], o bien el uso de otros circuitos integrados de propósito general [Cannata *et al.* 2008]. Estos circuitos pueden llevar a cabo la conversión analógico/digital, y tareas de reducción de errores y compensación de interferencias, siendo la mejor opción en términos de área y eficiencia energética. Por otra parte, la detección del deslizamiento en tareas de manipulación en manos o pinzas debe realizarse en un intervalo de tiempo muy pequeño (2-4 ms). Esto significa que todo el conjunto de sensores de fuerza del sensor táctil (tácteles) tiene que ser procesado en este tiempo (conversión analógico/digital más el algoritmo de detección), lo que es posible gracias al alto rendimiento dinámico de los ASICs [Maldonado-Lopez *et al.* 2009]. Desafortunadamente, los ASICs son bastante rígidos en lo que a su programación se refiere, por lo que la posibilidad de actualizar su funcionalidad una vez que se fabrican es limitada.

La mayoría de las implementaciones de la circuitería de acondicionamiento de sensores táctiles están basadas en microcontroladores [Choi *et al.* 2005; Göger *et al.* 2006; Cannata *et al.* 2006]. Generalmente, esta estrategia requiere de un mayor número de dispositivos en la placa de circuito impreso, dando lugar a un circuito con una mayor superficie debido al espacio que éstos ocupan, así como por el empleo de un cableado más complejo. Los tácteles se leen y se procesan de forma secuencial, por lo tanto, el tiempo de respuesta es pobre y la detección del deslizamiento para un sensor piezorresistivo con un alto número de tácteles no es factible. Sin embargo, este enfoque permite que el diseño sea actualizable, por lo que las tareas a realizar por el microcontrolador se pueden cambiar mediante una nueva programación. Un esquema común de esta realización se ilustra en la Figura 1-2 [Vidal-Verdú, Oballe-Peinado, *et al.* 2011; Cannata *et al.* 2006]. Los amplificadores operacionales se replican por columna

para evitar la diafonía que origina errores por caminos resistivos parásitos, como se ilustra en Figura 1-3 [D'Alessio 1999].

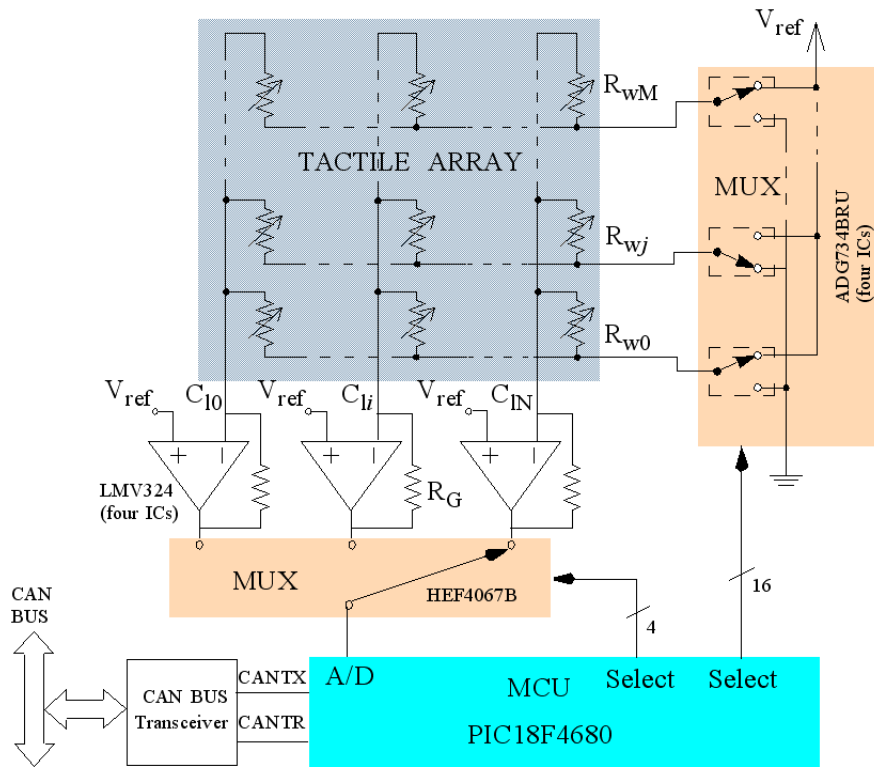


Figura 1-2. Circuito basado en un microcontrolador [Vidal-Verdú, Oballe-Peinado, et al. 2011].

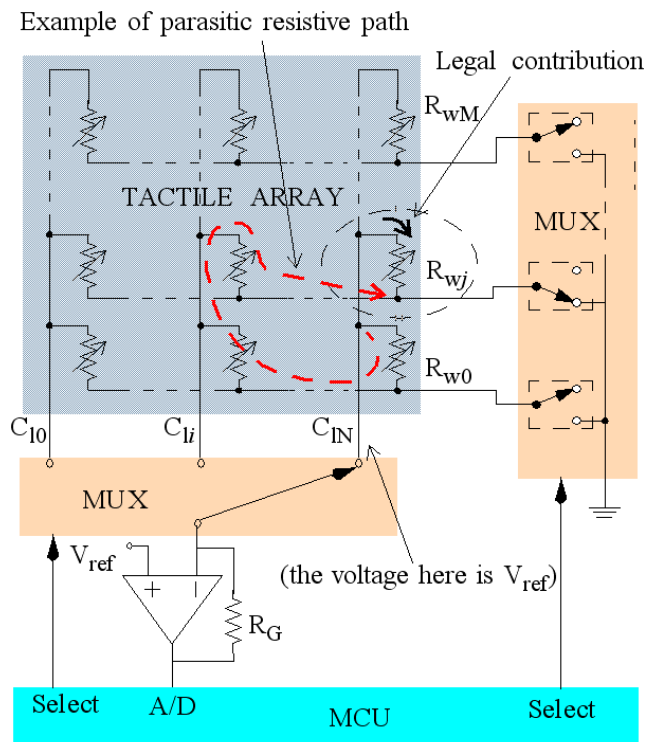


Figura 1-3. Diafonía originada por caminos resistivos parásitos [Vidal-Verdú, Oballe-Peinado, et al. 2011].

Se pueden conseguir mejoras de rendimiento si el hardware se basa en un PSoC (*Programmable System-on-Chip*) [Vidal-Verdú et al. 2010]. Estos dispositivos tienen un conjunto de bloques analógicos y digitales *on-chip* que pueden ser configurados por el usuario, por lo que el hardware se ve reducido si se compara con la solución que emplea microcontroladores estándar (Figura 1-4). Sin embargo, el tamaño de la matriz que puede ser abordado depende de los recursos del PSoC y el número de pines de entrada y de salida que éste posea. Por otra parte, aunque los bloques en el chip permiten cierto grado de paralelismo en el acondicionamiento de la señal, los algoritmos programados se ejecutan de forma secuencial.

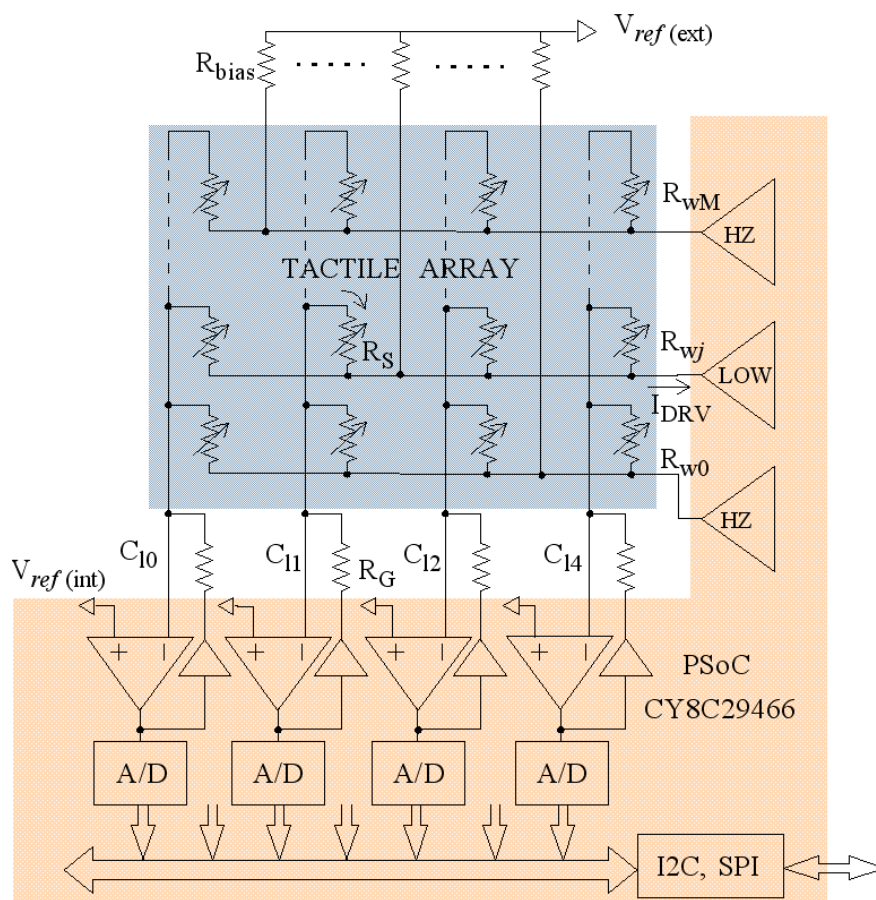


Figura 1-4. Circuito basado en un PSoC [Vidal-Verdú et al. 2010].

Un dispositivo con rendimiento entre un ASIC y un microcontrolador es el *Field Programmable Gate Array* o FPGA. Estos dispositivos son flexibles porque se pueden programar, y al mismo tiempo tienen un rendimiento dinámico alto, como los ASICs, debido a que el hardware interno es configurado mediante la programación y puede trabajar en paralelo [Díaz, Javier et al. 2008]. La principal ventaja de esta estrategia es la

posibilidad de ejecutar algoritmos complejos de pre-procesado en tiempo real. Conforme el sistema se hace más complejo, al añadir sensores táctiles por ejemplo en dedos y palmas de manos artificiales, los abundantes datos registrados deben ser pre-procesados para permitir una ejecución del control en tiempo real. Hay muchos ejemplos de primitivas que se implementan, o podrían implementarse al menos en parte, como pre-procesado local, como el cálculo de los momentos de la imagen táctil, frecuente y apropiado para obtener las características principales del contacto en los sensores de manos artificiales [Gorges *et al.* 2014]. Estos momentos se usan en [Schill *et al.* 2012] como entrada de algoritmos de clasificación y aprendizaje de la estabilidad del agarre, y en [Drimus *et al.* 2011] se parte de series en el tiempo del primer y segundo momentos para conocer la elasticidad del objeto con el que se contacta. La ecualización o binarización de la imagen se hace en [Escalda Navarro *et al.* 2012], además del cálculo de los momentos, para reconocimiento de objetos con manos robóticas. Como se ha mencionado arriba, la detección del deslizamiento (*early slippage*) proporciona la información al controlador de la mano para que ajuste la fuerza de agarre. La implementación embebida puede reducir la carga del robot ya que no tiene que hacer un seguimiento visual para asegurar la estabilidad del agarre. Algunas implementaciones usan un sensor más complejo, con una capa piezorresistiva y otra piezoeléctrica o un sensor de deslizamiento [Göger *et al.* 2009; Ito *et al.* 2012], aunque es posible usar una sola capa piezorresistiva y procesar la lectura [Maldonado-Lopez *et al.* 2009; Schöpfer *et al.* 2010; Schürmann *et al.* 2012]. La reconstrucción de las fuerzas de contacto en la superficie del sensor cubierto con una capa de elastómero a partir de la lectura de la matriz de tácteles de debajo (problema de inversión) se implementa en una FPGA en [Seminara *et al.* 2015]. Otros algoritmos como filtros de partículas o clasificadores se han usado para obtener información a partir de un sensor táctil para un agarre estable y seguro, o la identificación del material con el que se hace contacto y su textura [Dahiya *et al.* 2013]. La capacidad de procesamiento en tiempo real de las FPGAs las hace adecuadas para este pre-procesado y su uso en sistemas de control, y en concreto para manos robóticas. Por ejemplo, en la Figura 1-5 [Kawasaki 2015] la FPGA forma parte de la arquitectura de la electrónica local del brazo y la mano, lo mismo que en otros sistemas similares como el robot humanoide ARMAR III [Göger *et al.* 2006]. Sin embargo en estos sistemas se incluye también otra circuitería de acondicionamiento y adquisición. La razón es que las FPGAs se orientan comúnmente al procesamiento masivo de datos digitales, y no incluyen habitualmente

bloques funcionales para facilitar la interfaz con entradas analógicas. En esta tesis se hacen propuestas dirigidas a obtener una circuitería local basada en FPGAs muy compacta, ya que implementa conexión directa sensor-FPGA, de forma similar a como se hace en las conexiones directas sensor-microcontrolador. En lo que sigue se repasan los conceptos básicos detrás de esta conexión directa. En los distintos capítulos de la tesis se harán también discusiones de los problemas específicos que surgen, por ejemplo en relación con la diafonía entre tácteles, y los antecedentes relacionados con los mismos.

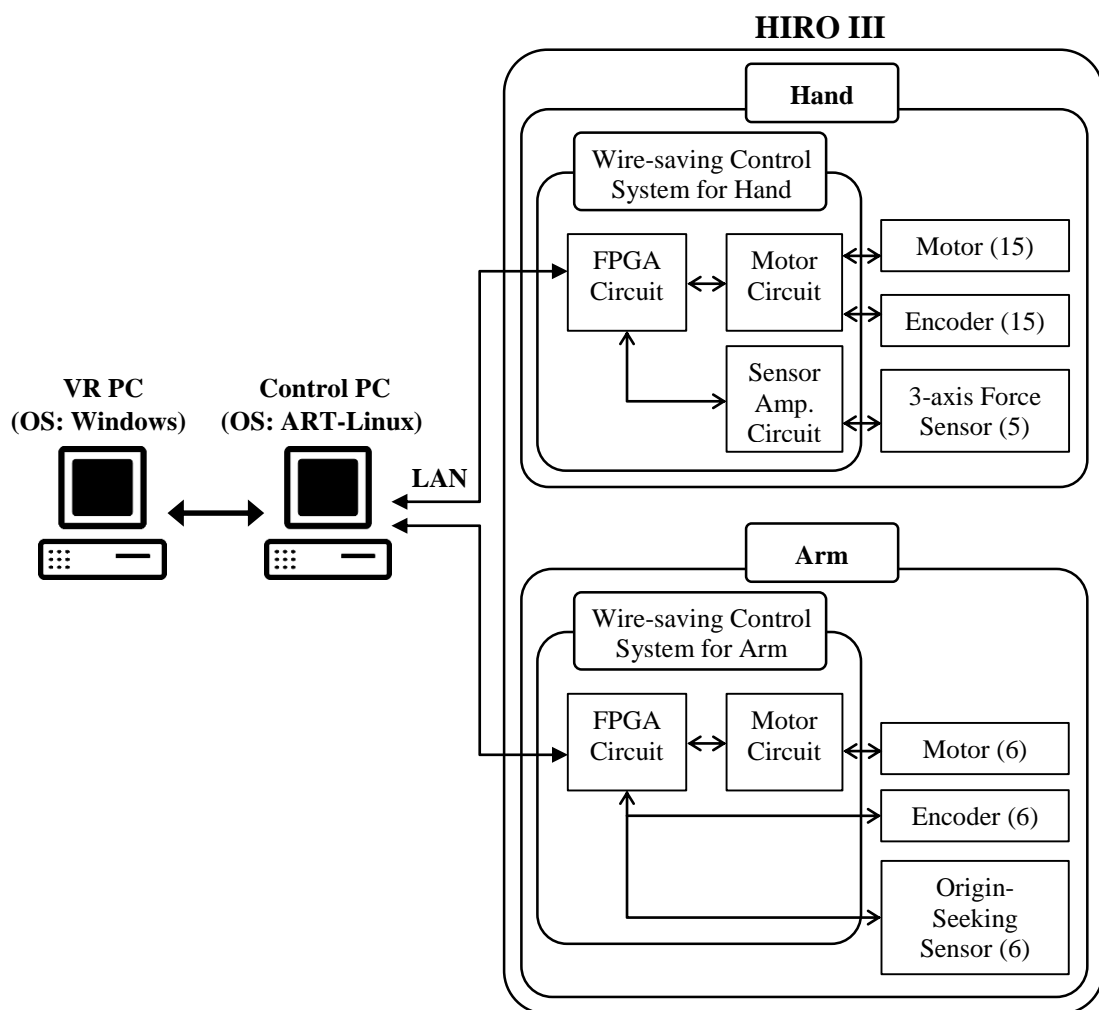


Figura 1-5. Sistema de control de la mano robótica HIRO III [Kawasaki 2015].

1.2. Circuitos de conexión directa sensor-microcontrolador

Tradicionalmente, los circuitos para la adquisición de datos constan de un sensor que convierte una magnitud física en una variable eléctrica, de un circuito de acondicionamiento que realiza tareas de linealización-amplificación, de un conversor analógico-digital y de un microcontrolador/microprocesador que procesa los datos. Para el caso particular de los sensores moduladores resistivos, existen trabajos que reducen la complejidad de la aproximación anterior proponiendo circuitos de interfaz directa sensor-microcontrolador [Webjörn 2004; Bierl 1999; Cox 1997; Sherman 1993; Merritt 1999; Richey 1997], que no hacen uso del circuito de acondicionamiento ni del conversor analógico-digital.

En estos circuitos de interfaz directa, el microcontrolador excita a un circuito RC, que incluye al sensor, para obtener una señal modulada en el tiempo que puede ser medida mediante un temporizador digital del propio microcontrolador. Estas señales reciben el nombre de casi-digitales [Pallàs-Areny et al. 2001], pues la información analógica que representan puede ser medida de forma directa por un sistema digital.

La Figura 1-6 muestra el circuito de interfaz-directa básico para sensores resistivos, donde R_x es el sensor y C es el condensador usado para generar la señal casi-digital.

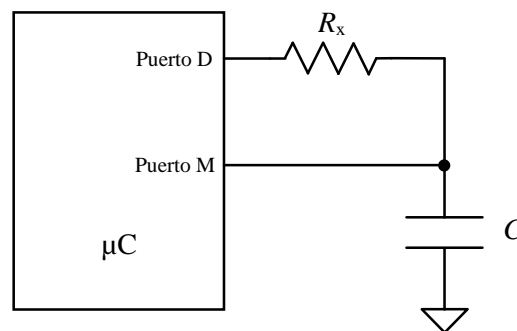


Figura 1-6 Circuito de interfaz directa sensor-microcontrolador.

La medida consta de dos fases: carga del condensador y descarga/medida de tiempo.

Durante la fase de carga (Figura 1-7a) el puerto D (descarga) se configura como entrada estableciendo un estado de alta impedancia (HZ) mientras que el puerto M (medida), fijado como salida, pone un “1” lógico que carga el condensador hacia la alimentación del sistema V_{CC} .

En la fase de descarga (Figura 1-7b) el puerto M pasa a configurarse como entrada (HZ) y el puerto D como salida fijando un “0” a la vez que se inicia la cuenta de un temporizador. De esta forma, el condensador se descarga a través del puerto D mientras que en el puerto M se mide un cambio de “1” a “0” en los niveles lógicos. Este cambio se produce cuando la tensión en el condensador cae por debajo de la tensión umbral (V_{TL}) del búfer de entrada del microcontrolador, deteniendo en ese instante la cuenta del temporizador. Si la velocidad de cambio (*slew rate*) de las señales a medir es lenta con respecto al periodo de la señal de reloj con la que opera el microcontrolador y/o la señal de descarga contiene fuertes componentes de ruido, los puertos de entrada a usar deben emplear un búfer disparador Schmitt (ST, Schmitt trigger).

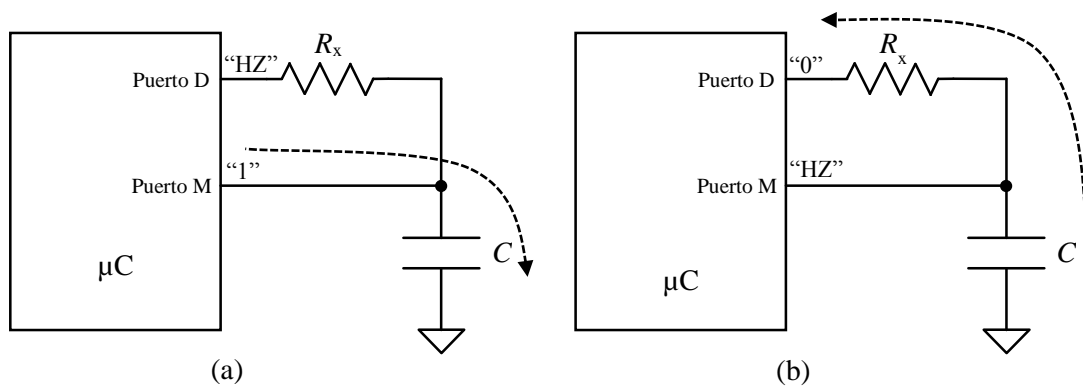


Figura 1-7. Fases de carga y descarga del condensador.

El concepto de señal lenta, en general, depende de la precisión que se quiera tener en la medida.

La forma de onda de la señal del circuito RC que incluye al sensor se muestra en la Figura 1-8.

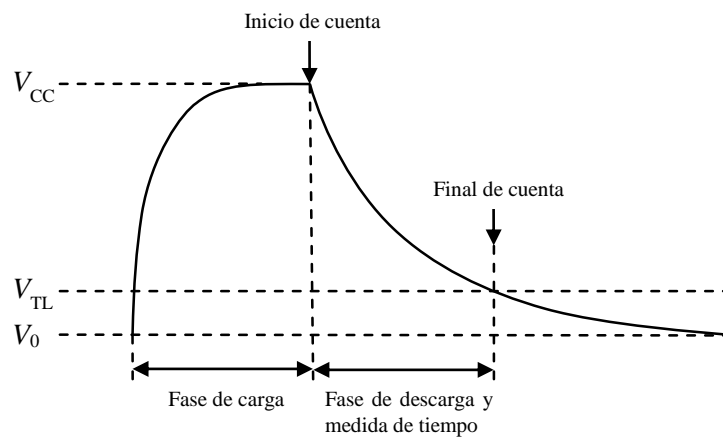


Figura 1-8. Fases de carga y descarga por cada ciclo de medida.

La tensión en el condensador durante la fase de descarga es

$$V_c(t) = V_0 + (V_{CC} - V_0)e^{-t/\tau} \quad (1.1)$$

donde $\tau = R_x C$. De (1.1) se obtiene el tiempo para descargar el condensador desde V_{CC} a V_{TL} ,

$$T_x = R_x C \ln \left(\frac{V_{CC} - V_0}{V_{TL} - V_0} \right) \quad (1.2)$$

El tiempo de descarga T_x se expresa a partir del número digital obtenido por el temporizador durante la fase de descarga y, si V_1 , V_0 , V_{TL} , C y la base de tiempo del temporizador son constantes, N puede expresarse como

$$N = k_R R_x \quad (1.3)$$

donde k_R es una constante que depende de V_1 , V_0 , V_{TL} , C y la base de tiempo del temporizador.

1.3. Conversión tiempo-digital

Determinadas magnitudes físicas pueden ser medidas a través de sensores resistivos. Estos sensores pueden ser excitados de forma directa por un sistema digital, obteniendo una señal con una modulación de un parámetro temporal (frecuencia, periodo) [Richey 1997; Bierl 1996]. La conversión de un intervalo de tiempo en un número digital puede llevarse a cabo mediante temporizadores en un sistema digital. A continuación se analizan sus características, especialmente para señales con una velocidad de cambio lenta.

1.3.1. Principio de funcionamiento

La señal a medir se conecta a una entrada del sistema digital donde un búfer de entrada se encarga de comparar dicha señal con una tensión umbral [Peatman 1997]. La detección de los cruces por esta tensión de referencia hace que la unidad central de proceso inicie o detenga a un contador que incrementa su valor cada T_s , donde T_s es el periodo de la señal de referencia de un oscilador. El número digital resultante representa la información temporal modulada por el sensor.

1.3.2. Fuentes de incertidumbre

En una conversión tiempo-digital existen varias fuentes de incertidumbre que influyen en la medida: inestabilidad de la frecuencia de referencia del oscilador, cuantificación y ruido de disparo.

1.3.2.1. Inestabilidad de la frecuencia de referencia

Si la frecuencia de referencia usada por el contador es inestable, el resultado de la medida también lo es, y por lo tanto la incertidumbre de la medida aumenta. La frecuencia de los osciladores puede verse alterada por varios factores (temperatura, tensión de alimentación, envejecimiento). Concretamente, los osciladores basados en circuitos RC o los resonadores cerámicos pueden tener una inestabilidad muy importante. Sin embargo, los osciladores de cristal de cuarzo poseen una estabilidad igual o mayor a ± 50 ppm, por lo que los efectos de la inestabilidad pueden considerarse despreciables comparados con los de cuantificación y disparo. Por lo tanto, el uso de osciladores de cristal es más que recomendable para que la inestabilidad de la frecuencia del oscilador no influya en la incertidumbre de la medida de tiempo.

1.3.2.2. Cuantificación

La conversión tiempo-digital produce un número digital que representa a un intervalo de tiempo que es múltiplo de la base de tiempo del contador T_s (periodo de la señal del oscilador), produciendo un error de cuantificación en la medida. Si la señal de entrada y el oscilador de referencia son asíncronos, el error de cuantificación está comprendido entre $-T_s$ y T_s [Bouwens 1984] y el resultado de la medida tiene una ambigüedad de ± 1 cuenta. El error de cuantificación posee dos componentes: una asociada al evento de inicio de cuenta e_{q1} y otra al de final de cuenta e_{q2} . Por esto, el error de cuantificación total e_q , será la suma de ambos y es igual a la diferencia entre el intervalo de tiempo medido T_0^* y el intervalo de tiempo a medir T_0 .

$$e_q = e_{q1} + e_{q2} = T_0^* - T_0 \quad (1.4)$$

En el caso de los microcontroladores, el inicio de la descarga y de la cuenta coinciden en el tiempo, por lo que el ruido de cuantificación ya no es de ± 1 cuenta, sino que variará entre 0 y 1.

1.3.2.3. Ruido de disparo

La incertidumbre de disparo depende del ruido de disparo externo ($E_{n,e}$), del ruido de disparo interno ($E_{n,i}$) y de la velocidad de cambio de la señal de entrada. El ruido de disparo externo es la tensión de ruido superpuesta a la señal de entrada, mientras que el ruido de disparo interno tiene al menos dos componentes [Larsson 1999; Hall et al. 2000]: una generada por el propio microcontrolador, debido al ruido térmico o al producido por la propia actividad interna del dispositivo y otra proveniente del ruido de su tensión de alimentación.

1.3.3. Análisis de la incertidumbre estándar

El resultado de una medición es sólo una aproximación o estimación del valor del mesurando, y únicamente se halla completo cuando está acompañado de una declaración acerca de la incertidumbre de dicha estimación [ISO 1995]. Si se considera al tiempo de descarga T_x como el mesurando Y , y al resultado de la cuenta multiplicado por la base de tiempo del contador T_s como la variable de entrada observada X . La relación entre Y y X se modela [Lira et al. 1997] por

$$Y = X + Z \quad (1.5)$$

donde Z es el factor de corrección debido a los efectos de cuantificación de la conversión tiempo-digital.

Los efectos del ruido de disparo en el punto de parada del contador hacen que la variable X sea aleatoria. Para un ruido gaussiano, el valor eficaz de la incertidumbre de disparo es [Hewlett-Packard 1997]

$$u_{\text{trigger}} = \frac{\sqrt{E_{n,i}^2 + E_{n,e}^2}}{\text{SR}} \quad (1.6)$$

donde $E_{n,i}$ es la tensión eficaz de ruido superpuesta a la tensión umbral, $E_{n,e}$ es la tensión eficaz de ruido superpuesta a la señal de descarga y SR es la velocidad de cambio de la señal en el punto de disparo. Estas tensiones eficaces no son conocidas por lo que la ecuación (1.6) no puede ser tomada para analizar cuantitativamente los efectos del ruido en la conversión tiempo-digital.

La incertidumbre de X , por tanto, debe evaluarse cuantitativamente mediante una evaluación de tipo A [ISO 1995], que consiste en realizar un análisis estadístico a partir de una serie de observaciones. Si una observación particular x_k se toma como valor estimado x de la variable X , la incertidumbre estándar de x es

$$u(x) = s(X) \quad (1.7)$$

Si para ofrecer el valor estimado de x se hace uso de m observaciones entonces la incertidumbre estándar queda como

$$u(x) = \frac{s(X)}{\sqrt{m}} \quad (1.8)$$

Teniendo en cuenta lo expresado en el apartado 1.3.2.2, para el ruido de cuantificación, el valor estimado de Z y su incertidumbre estándar son

$$z = -\frac{T_s}{2} \quad (1.9)$$

$$u(z) = \frac{T_s}{\sqrt{12}} \quad (1.10)$$

Siendo X y Z independientes, el valor estimado de Y y su incertidumbre estándar [Lira et al. 1997] son

$$y = x + z \quad (1.11)$$

$$u(y) = \sqrt{u^2(x) + u^2(z)} \quad (1.12)$$

1.3.4. Análisis de la resolución

La resolución en bits de la conversión tiempo-digital teniendo en cuenta únicamente los efectos de la cuantificación es

$$M = \text{lb} \left(\frac{T_{x \max} - T_{x \min}}{T_s} \right) \quad (1.13)$$

donde $T_{x \min}$ y $T_{x \max}$ son los tiempos de descarga mínimo y máximo que se van a medir y T_s es la base de tiempo del contador. Según (1.13), la resolución en ohmios es

$$r = \frac{R_{\max} - R_{\min}}{2^M} \quad (1.14)$$

donde R_{\min} y R_{\max} son los valores mínimo y máximo que producen $T_{x \min}$ y $T_{x \max}$ respectivamente.

Si la conversión tiempo-digital se ve afectada por el ruido superpuesto a la señal de descarga y a la tensión umbral de los buffers, hay que evaluar cuantos de los bits obtenidos por la expresión (1.13) son efectivos. La siguiente ecuación determina el valor de los bits efectivos de la conversión tiempo-digital (ENOB, *effective number of resolution bits*) teniendo en cuenta la desviación estándar como indicador de la incertidumbre debida al ruido de disparo.

$$\text{ENOB} = M - \text{lb} \left(\frac{u_{\max}(y)}{u(z)} \right) \quad (1.15)$$

donde $u_{\max}(y)$ es la desviación estándar de y para $R_x = R_{\max}$. La resolución en ohmios a partir de los bits efectivos queda

$$r = \frac{R_{\max} - R_{\min}}{2^{\text{ENOB}}} \quad (1.16)$$

Para τ pequeñas, $u(y) \approx u(z)$ por lo que (1.15) se reduce a $\text{ENOB} \approx M$. Sin embargo, para τ grandes, $u(y) \approx u(x)$ resultando un ENOB igual a

$$\text{ENOB} \approx \text{lb} \left(\frac{R_{\max} - R_{\min}}{R_{\max}} \frac{(V_{\text{TL}} - V_0) \ln \left(\frac{V_1 - V_0}{V_{\text{TL}} - V_0} \right)}{\sqrt{12} \sqrt{E_{n,i}^2 + E_{n,e}^2}} \right) \quad (1.17)$$

De (1.17) se deduce que el ENOB es independiente del valor del condensador C , por lo que una vez la constante de tiempo alcanza un valor, la resolución no mejora de forma apreciable por el uso de condensadores de mayor capacidad [Reverter et al. 2004].

1.4. Medida de resistencias

El valor de la resistencia a medir R_x se puede deducir a partir de (1.2) conociendo los valores de los parámetros V_1 , V_0 , V_{TL} y C . Este método tiene el inconveniente de ofrecer resultados pobres debido, sobre todo, a la variabilidad de estos valores con el tiempo y la temperatura. Para paliar este problema se recurre a técnicas de calibración, donde el valor de resistencia del sensor se basa en uno o más componentes de referencia.

1.4.1. Calibración de un punto

Esta técnica de calibración incluye una única resistencia de referencia R_{c1} [Cox 1997; Richey 1997; Bierl 1999; Merritt 1999]. La Figura 1-9 ilustra el circuito de conexión directa sensor-microcontrolador para una calibración de un punto. Tal y como se indica en el apartado 1.3.1, el microcontrolador realiza dos medidas: una para la resistencia del sensor, que da lugar a un número digital N_x , y otra para la resistencia de referencia, obteniendo el número N_{c1} .

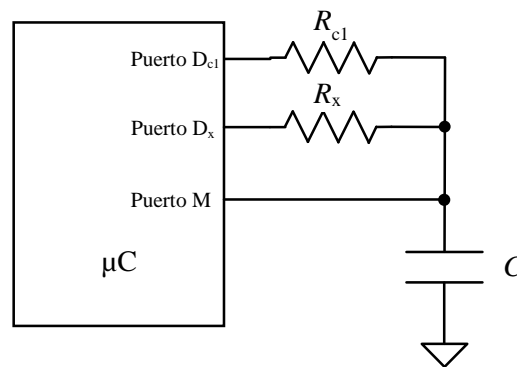


Figura 1-9. Circuito de interfaz directa sensor-microcontrolador con una resistencia de calibración.

En la Figura 1-10 se muestra como la recta de calibración de un punto pasa por los puntos $(0, 0)$ y (R_{c1}, N_{c1}) , por lo que si la característica de transferencia real posee un error de cero, éste solo se compensa para valores de R_x próximos a R_{c1} .

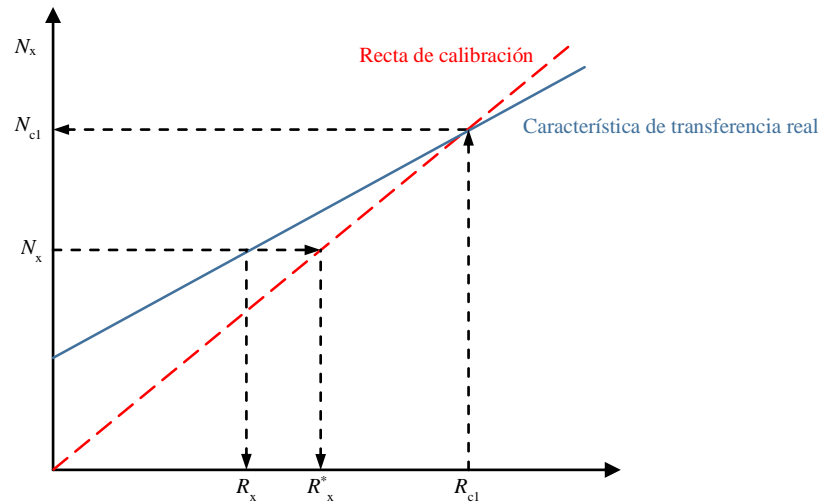


Figura 1-10. Curva de calibración de un punto.

El valor estimado de R_x a partir de un ciclo de medida es

$$R_x^* = \frac{N_x}{N_{c1}} R_{c1} \quad (1.18)$$

El valor de la resistencia de calibración de un punto debe estar próximo al valor medio del rango de medida o ser igual al valor más probable del sensor [Reverter, Jordana, et al. 2005].

1.4.2. Calibración de dos puntos

En el caso de una calibración de dos puntos, el circuito lleva a cabo la medida de tres resistencias para poder estimar el valor de R_x . La Figura 1-11 muestra el circuito de conexión directa donde, aparecen dos resistencias de referencia (R_{c1} y R_{c2}).

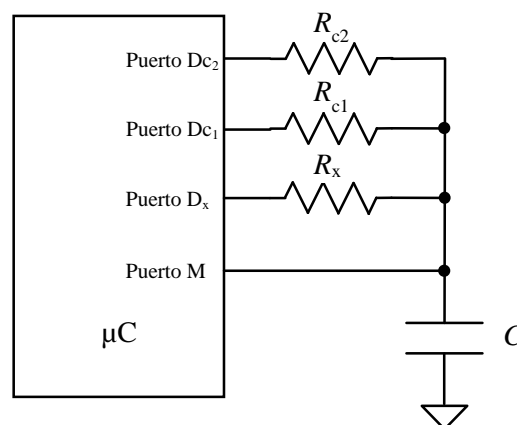


Figura 1-11. Circuito de interfaz directa sensor-microcontrolador con dos resistencias de calibración.

En la Figura 1-12 se observa como la recta de calibración de dos puntos pasa por los puntos (R_{c1}, N_{c1}) y (R_{c2}, N_{c2}) compensando errores de cero, ganancia y no linealidad.

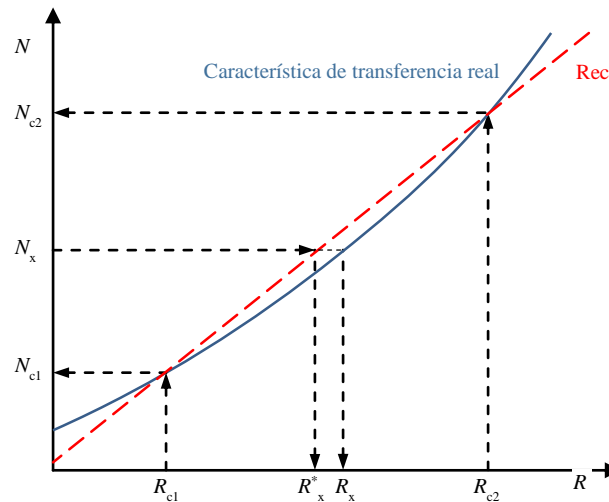


Figura 1-12. Curva de calibración de dos puntos.

Para un valor de cuenta N_x , el valor estimado de R_x es

$$R_x^* = \frac{N_x - N_{c1}}{N_{c2} - N_{c1}}(R_{c2} - R_{c1}) + R_{c1} \quad (1.19)$$

El criterio para la elección de los valores de las resistencias de referencia para esta técnica de calibración de dos puntos persigue disminuir el error de no linealidad [Custodio *et al.* 2001]. Si la característica de transferencia real presenta un comportamiento cuadrático como el de la Figura 1-12, el error máximo es mínimo cuando se escogen resistencias de calibración en el 15% y el 85% del rango de medida [Pallàs-Areny *et al.* 2004].

1.4.3. Análisis de la exactitud

La medida de R_x , haciendo uso de los circuitos propuestos por las técnicas de calibración anteriormente citadas, se ve afectada por las resistencias internas de los puertos del microcontrolador.

La Figura 1-13 muestra el circuito de la Figura 1-6 considerando las resistencias internas de los puertos para la fase de carga (a) y para la fase de descarga (b). Las resistencias R_p y R_n son las resistencias internas de los puertos cuando la salida de los mismos toma los

valores lógicos “1” y “0” respectivamente, y Z_{in} es la impedancia del puerto cuando se configura como entrada.

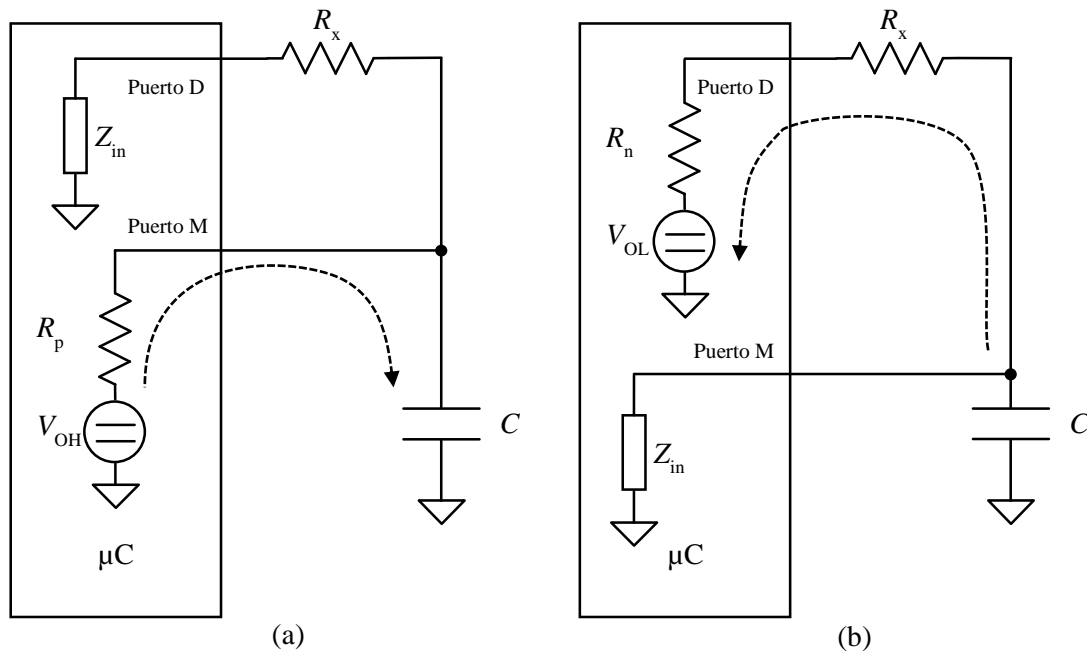


Figura 1-13. Circuito de interfaz directa sensor resistivo y μC cuando se considera la resistencia interna de los puertos del μC . Fase de carga (a) y fase de descarga (b).

El tiempo de carga así como la tensión que se alcanza en el condensador dependen por tanto de R_p , aunque no afectan a la calidad de la medida puesto que si el tiempo de carga es lo suficientemente largo, el condensador C acaba cargándose prácticamente a V_{OH} y, aunque fuese algo inferior a ella, el tiempo de medida no queda afectado si siempre se alcanza la misma tensión.

Por otro lado, durante la fase de descarga el condensador se descarga a través de R_x más R_n con una constante de tiempo $\tau = (R_x + R_n)C$. Es por esto que la resistencia R_n aumenta el tiempo de descarga, alterando por tanto en la medida de R_x , quedando por tanto la ecuación (1.3) de la siguiente forma

$$N = k_R (R_x + R_n) \quad (1.20)$$

En la ecuación (1.20), el valor de R_n se considera constante durante toda la fase de descarga e independiente de la resistencia externa que se quiere medir. En el caso de un microcontrolador con buffers de salida CMOS, R_n modela la resistencia del canal del

transistor NMOS del búfer, pudiéndose considerar constante siempre que el transistor se encuentre en la zona lineal de trabajo.

Además, no se toman en cuenta los efectos de la impedancia finita y la corriente de fuga del puerto de medida configurado como entrada durante la fase de descarga. Estos parámetros, no obstante, producen una relación no lineal entre la resistencia y el tiempo medido [*Custodio et al. 2001*].

A continuación se evalúa los efectos de R_n en la exactitud de las técnicas de calibración.

1.4.3.1. Análisis de la exactitud de la calibración de un punto

Sustituyendo (1.20) en (1.18) se tiene

$$R_x^* = \frac{R_{cl}}{R_{cl} + R_{n,cl}} R_x + \frac{R_{cl} R_{n,x}}{R_{cl} + R_{n,cl}} \quad (1.21)$$

su error relativo es

$$e_r = \frac{|R_x^* - R_x|}{R_x} = \left| \frac{-R_{n,cl}}{R_{cl} + R_{n,cl}} + \frac{1}{R_x} \frac{R_{cl} R_{n,x}}{R_{cl} + R_{n,cl}} \right| \quad (1.22)$$

que es cero cuando

$$R_x = \frac{R_{cl} R_{n,x}}{R_{n,cl}} \quad (1.23)$$

A partir de (1.21), $R_x^* = R_x$ cuando $R_{n,x} = R_{n,cl} = 0$; existiendo errores de cero y de ganancia en cualquier otro caso. Los errores son del mismo orden de magnitud que el valor de R_n (decenas de ohmios). Según (1.23), cuando $R_{n,x} \approx R_{n,cl}$ el error relativo es cero para $R_x \approx R_{cl}$ y aumenta conforme R_x se aleje de R_{cl} . Por esto, el valor de la resistencia de calibración debe ser elegido con un valor próximo al valor medio del rango de resistencias a medir o con el valor del sensor donde se busque una mayor precisión para la aplicación concreta que se esté abordando.

1.4.3.2. Análisis de la exactitud de la calibración de dos puntos

Sustituyendo (1.20) en (1.19) se tiene

$$R_x^* = \frac{R_{c2} - R_{c1}}{R_{c2} - R_{c1} + (R_{n,c2} - R_{n,c1})} R_x + \frac{R_{c1} (R_{n,c2} - R_{n,x}) - R_{c2} (R_{n,c1} - R_{n,x})}{R_{c2} - R_{c1} + (R_{n,c2} - R_{n,c1})} \quad (1.24)$$

El error relativo es

$$e_r = \left| \frac{-(R_{n,c2} - R_{n,c1})}{R_{c2} - R_{c1} + (R_{n,c2} - R_{n,c1})} + \frac{1}{R_x} \frac{R_{c1} (R_{n,c2} - R_{n,x}) - R_{c2} (R_{n,c1} - R_{n,x})}{R_{c2} - R_{c1} + (R_{n,c2} - R_{n,c1})} \right| \quad (1.25)$$

que es cero cuando

$$R_x = \frac{R_{c1} (R_{n,c2} - R_{n,x}) - R_{c2} (R_{n,c1} - R_{n,x})}{R_{n,c2} - R_{n,c1}} \quad (1.26)$$

En este caso, según (1.24), $R_x^* = R_x$ cuando $R_{n,c2} - R_{n,x} = R_{n,c1} - R_{n,x} = R_{n,c2} - R_{n,c1} = 0$. En caso contrario, existen errores de cero y de ganancia pero, a diferencia de la calibración de un punto, la magnitud de los errores es de $R_{n,c} - R_{n,x}$, cuyo valor es menor que R_n . Esto implica que la calibración de dos puntos es más exacta que la de un punto.

En cuanto al criterio para seleccionar las resistencias de calibración, si la característica de transferencia del circuito presenta un comportamiento cuadrático (Figura 1-12), el error máximo es mínimo cuando se seleccionan los dos puntos de calibración en el 15 % y el 85 % del rango de medida [Pallàs-Areny *et al.* 2004]. Si todas las diferencias entre las resistencias internas fuesen iguales a cero y la relación entre R_n y N_x es lineal, cualquier par de resistencias R_{c1} , R_{c2} podrían emplearse para calibrar el circuito.

Capítulo 2. Circuitos de conexión directa sensor-FPGA

2.1. Introducción

El enfoque más común para la adquisición de datos, a partir de sensores analógicos, incluye circuitos de acondicionamiento de señal además de un conversor analógico-digital. En los sistemas actuales, el número de sensores y su complejidad van en aumento, por lo que varios conversores A/D pueden ser necesarios. Algunos microcontroladores poseen un multiplexor que permite seleccionar, de entre un conjunto de señales analógicas de entrada, cuál de ellas es llevada a un único conversor A/D [*Microchip Technology Inc 2012*]. Esto, junto con el hecho de que el procesado en la propia CPU del microcontrolador es realizado de forma secuencial, hace que la respuesta a ciertos eventos pueda ser lenta. Además, en el caso de un microcontrolador con varios canales A/D, el coste y el tamaño del sistema se ven incrementados por la circuitería de acondicionamiento externa, estando finalmente el sistema limitado por el número de canales que el microcontrolador disponga.

Una alternativa al planteamiento anterior consiste en el empleo de una conexión directa entre el sensor y el microcontrolador [*Cox 1997*]. Como se explica en el capítulo 1, esta aproximación consiste, básicamente, en la medida de un intervalo de tiempo determinado por la resistencia o la capacidad de un sensor resistivo o capacitivo, respectivamente. Estas implementaciones requieren de pocos componentes externos y pueden realizarse con entradas/salidas digitales de los propios microcontroladores.

Las limitaciones relacionadas con la adquisición y el procesado secuencial anteriormente mencionadas pueden superarse si se emplea una FPGA en lugar de un microcontrolador, ya que éstas últimas pueden ser configuradas para que una gran cantidad de bloques puedan operar en paralelo. Esta característica ha sido empleada para desarrollar complejos algoritmos de procesado en tiempo real [*Díaz, J. et al. 2008*], pero la misma

estrategia puede aplicarse para la adquisición de datos en paralelo a partir de un gran número de sensores analógicos resistivos o capacitivos.

La ausencia de *buffers* de entrada *Schmitt Trigger* es una desventaja de las FPGAs cuando se las compara con los microcontroladores en el contexto de las conexiones directas con sensores. La histéresis de estos *buffers* reduce la incertidumbre debida al ruido de disparo en la detección del cambio de nivel en señales con una velocidad de cambio lenta. Al ser este evento el que determina el final de la medida de tiempo, se puede afirmar que el ruido de disparo condiciona la precisión de la medida. Por otra parte, la tensión umbral V_{IL} de los *buffers* ST es más robusta frente al ruido superpuesto a la tensión de alimentación [Reverter 2014]. En la propuesta de [Ares et al. 2009] se introduce histéresis en la entrada de una FPGA mediante un bloque compuesto por un *buffer* de entrada, un *buffer* de salida, que proporciona una realimentación positiva, y dos resistencias externas. Esta solución permite controlar la histéresis a través de los valores de las resistencias externas. El circuito es robusto frente al ruido de disparo pero afecta negativamente al rango dinámico de la señal a medir.

En este capítulo se presentan y analizan varios módulos de captura que miden el tiempo de descarga de un circuito RC que incluye al sensor. Los recursos de una FPGA permiten desarrollar circuitos digitales que detecten el primer cambio del valor lógico de la señal de entrada. En lugar de añadir una realimentación analógica externa para introducir histéresis, se emplean circuitos digitales de la propia FPGA para conseguir el efecto memoria del ciclo de histéresis.

Por otra parte, estos módulos de captura pueden emplearse para mejorar el rendimiento y la precisión del proceso de medida. Como ejemplo, algunos de los módulos presentados identifican y son inmunes a *glitches* que puedan producirse en la señal de entrada, afectando con ello a la precisión de la medida. Además, la posibilidad de trabajar con elementos de almacenamiento síncronos disparados por doble flanco, así como la posibilidad de no sólo detectar la primera transición sino también la última, permiten llevar a cabo un promediado de varias medidas, filtrando con ello parte del ruido de disparo sin perder ancho de banda.

En cuanto a la exactitud, está limitada principalmente por la impedancia de los *buffers* de salida de la FPGA [Custodio et al. 2001; Bengtsson 2012]. Técnicas de calibración de un punto o de dos puntos pueden utilizarse para mejorar la exactitud, empleando para ello

medidas de tiempo de una resistencia externa conocida a través de un pin adicional de la FPGA, o de dos resistencias externas más dos pines de la FPGA, respectivamente. En este trabajo se compara el rendimiento obtenido con ambos métodos. Además, se propone el uso de otra estrategia de calibración que, por medio de un proceso de caracterización previo, tiene en cuenta la resistencia interna de los *buffers* de la FPGA. Esta estrategia emplea menos recursos que la calibración de dos puntos alcanzando una precisión similar.

2.2. Módulos de captura

El módulo de captura en la FPGA consta de un circuito de detección del cambio de '1' a '0' de la señal de descarga del condensador, de un contador que está habilitado durante la fase de descarga y de un registro donde se almacena el valor de la cuenta en el momento de la detección.

Como se describe en el apartado 1.3.2, las fuentes de incertidumbre en la conversión tiempo-digital son: la inestabilidad de la frecuencia de referencia del oscilador, el ruido de cuantificación y el ruido de disparo. A su vez, el ruido de disparo posee tres componentes: el ruido superpuesto a la señal de entrada, el ruido superpuesto a la tensión umbral V_{TL} y el ruido en el condensador al finalizar la etapa de carga. Todas estas fuentes de ruido hacen que la salida interna del *buffer* de entrada de la FPGA oscile cuando la señal de descarga alcanza la tensión umbral (Figura 2-1). Además, no aparece un único evento con el que determinar el final de cuenta, y con ello el tiempo de descarga T_x .

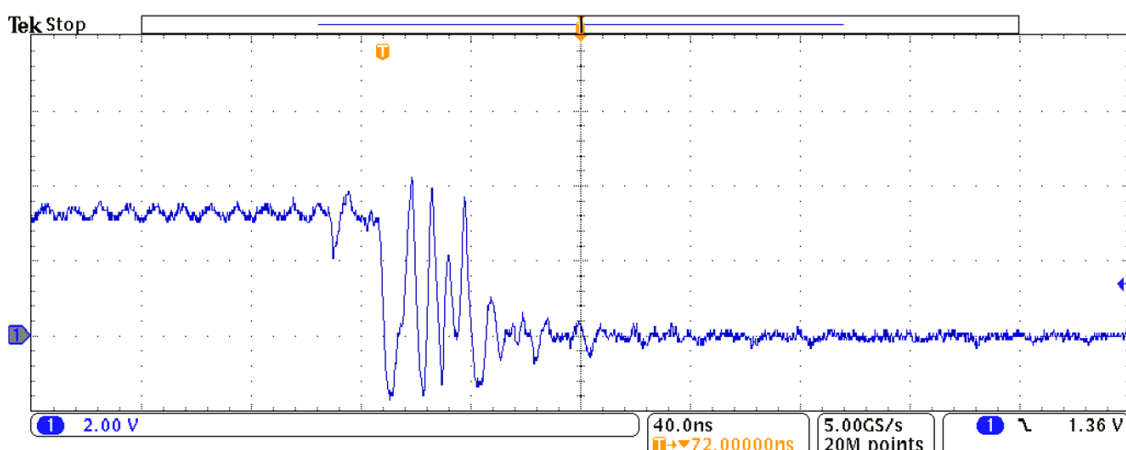


Figura 2-1. Transitorio a la salida del *buffer* de entrada cuando el cambio de nivel es detectado.

A continuación se presentan distintos módulos de captura.

2.2.1. Módulo de captura 1 (MC1) – Circuito síncrono de detección de cambio de nivel de la señal de descarga

El circuito síncrono típico para la detección de un cambio de ‘1’ a ‘0’ en una señal digital consta de dos biestables tipo-D activos por flanco de subida conectados en cascada y una puerta lógica cuya salida es ‘1’ durante un ciclo de reloj cuando el cambio es detectado (Figura 2-2). Ambos biestables son inicializados a ‘1’ durante la fase de carga, por lo que parten desde dicho valor al inicio de la fase de descarga.

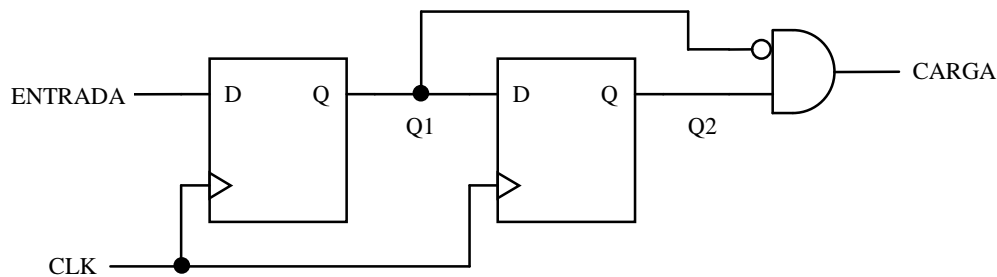


Figura 2-2. Circuito de detección del cambio de nivel en la señal de descarga (MC1).

Esta señal “CARGA” sirve como señal de carga del registro que almacena el valor de cuenta en el momento del cambio. Como se observa en la Figura 2-3, el valor de cuenta almacenado, en el momento de producirse el cambio en la entrada, es igual al valor del contador más uno. Para eliminar la necesidad de usar un restador para obtener el correcto valor de cuenta, se opta por retrasar un ciclo de reloj la puesta en marcha del contador con respecto al inicio de la fase de descarga.

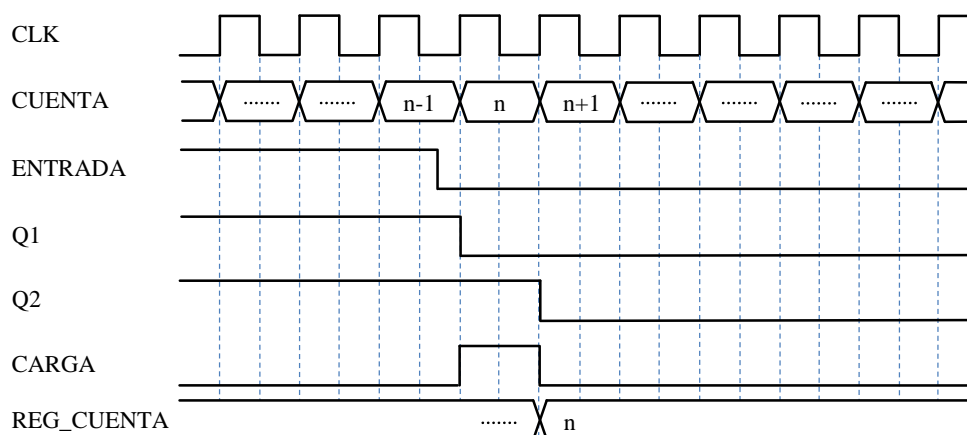


Figura 2-3. Cronograma para el circuito de detección de cambio de nivel en la señal de descarga (MC1).

El circuito anteriormente propuesto necesita, para su correcto funcionamiento, que la señal de entrada no cambie de valor lógico una vez se produce el cambio de ‘1’ a ‘0’. En

el caso de los microcontroladores comerciales, esto se asegura haciendo uso de entradas con *buffers* ST. Estos *buffers* hacen uso de la histéresis en los niveles de referencia para la detección de los valores lógicos ‘1’ y ‘0’ para, de esta forma, prevenir el ruido en la señal de entrada, siendo ideales para la detección de cambios en los valores lógicos de señales con una velocidad de cambio lenta en sus transiciones.

Sin embargo, las FPGAs no disponen de *buffers* ST. Esto provoca un incremento en la incertidumbre de la medida de tiempo debido al disparo, ya que la señal de entrada al primer biestable puede oscilar entre ‘1’ y ‘0’ durante un tiempo hasta que termina estabilizándose en ‘0’. Esto hace que el circuito de la Figura 2-2 pueda generar más de un pulso para la misma medida y el registro de cuenta almacene el valor que coincida con el último pulso generado (Figura 2-4).

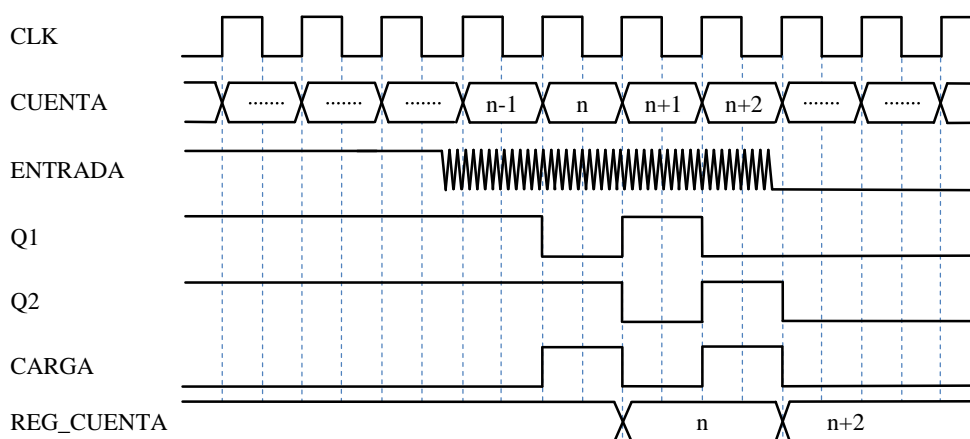


Figura 2-4. Cronograma de la detección de flanco de bajada con entrada ruidosa.

2.2.2. Módulo de captura 2 (MC2) - Circuito síncrono de detección de cambio de nivel de la señal de descarga con generación de pulso único

Para garantizar que este pulso se produzca sólo con el primer cambio de ‘1’ a ‘0’ detectado, se introduce una puerta lógica adicional como se observa en la Figura 2-5. Este circuito modifica ligeramente el anteriormente propuesto añadiendo realimentación.

Con este circuito, una vez que la salida del segundo biestable Q2 toma un valor ‘0’ se mantiene inalterable, por lo que la condición para la generación del pulso que almacena el valor del contador en el registro sólo se produce una sola vez por cada ciclo de descarga (Figura 2-6). Esta realimentación digital introduce memoria al circuito, reemplazando la

memoria analógica presente en los circuitos que emplean *buffers* ST o en el propuesto por [Ares et al. 2009].

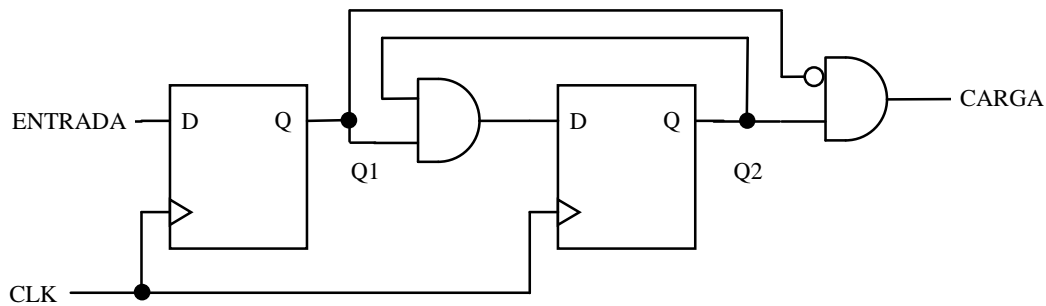


Figura 2-5. Circuito de detección de cambio de nivel con generación de pulso único (MC2).

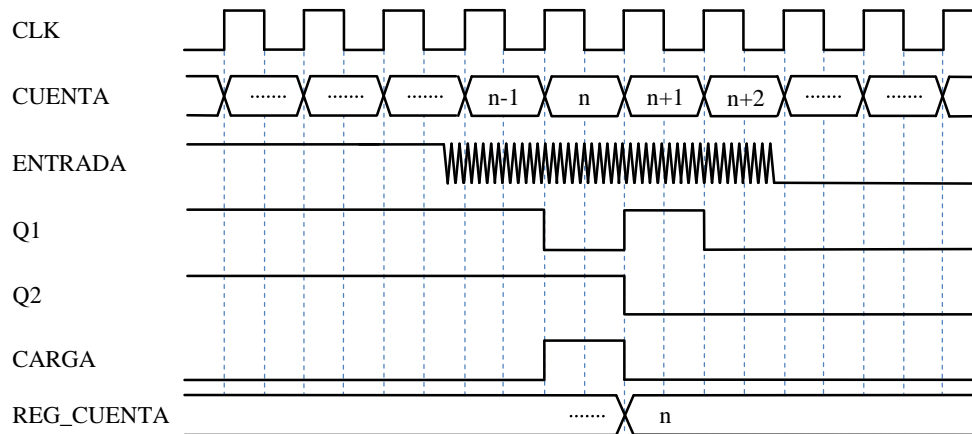


Figura 2-6. Cronograma de la detección de cambio de nivel con generación de pulso único (MC2).

Sin embargo, otro hecho que influye en la incertidumbre de la medida del tiempo, es que el biestable con salida Q1, que registra el cambio de ‘1’ a ‘0’ en la señal de descarga, lo hace en unos instantes de tiempo muy concretos, coincidiendo con el flanco de subida de la señal de reloj, y puede ocurrir que la señal de entrada, que se encuentra oscilando, no sea capturada hasta que termine tomando un valor constante ‘0’. En el cronograma de la Figura 2-6 hay cinco flancos de subida de la señal de reloj desde que la señal de entrada comienza a oscilar hasta que se estabiliza en su valor ‘0’. No se puede precisar cuál de estos flancos será el que coincida con el instante de tiempo donde el biestable registra el primer cambio de ‘1’ a ‘0’.

Para reducir la incertidumbre y evitar la inclusión de elementos externos a la propia FPGA, se proponen técnicas que hacen uso de elementos de memoria asíncronos (*latches*), así como síncronos (*flip-flops*), disparados por flanco de subida y de bajada.

2.2.3. Módulo de captura 3 (MC3) - Circuito de detección de cambio de nivel de la señal de descarga con *latch* anti-rebote

Para introducir la memoria presente en la histéresis de un *buffer* ST y, de esta forma, no tener que ser añadido como elemento externo, se propone el uso de biestables activos por nivel (*latch*). En las FPGAs, los elementos de almacenamiento pueden configurarse como *flip-flops*, disparados por flanco, o como *latches*, activos por nivel. La posibilidad de poder configurar los elementos de almacenamiento como *latches* permite generar una señal libre de *glitches* a partir de la detección de un '0' lógico en la señal de entrada (Figura 2-7). El *latch* se inicializa a '1' durante la fase de carga para que comience con ese valor durante la fase de descarga. La entrada D del *latch* es cero durante toda la fase de descarga y la entrada de G es la señal de entrada negada (el *latch* actualiza su salida cuando encuentra un nivel alto en esta entrada G). Esta señal "ENTRADA" es la salida de un *buffer* de la FPGA que tiene como entrada la señal de descarga del condensador (Figura 2-1). Este es, por tanto, el único uso que se hace dentro de la FPGA de la señal modulada por el sensor, evitando así retrasos innecesarios. Así, una vez que la señal de entrada toma el valor '0' por primera vez, la salida del *latch*, Q0, se actualiza a '0' para no cambiar durante el resto de la fase de descarga. Esta señal Q0 es ahora la entrada libre de *glitches* del circuito síncrono de detección de cambio de '1' a '0' propuesto inicialmente. De esta forma, se asegura un comportamiento equivalente al que se obtiene con un *buffer* ST.

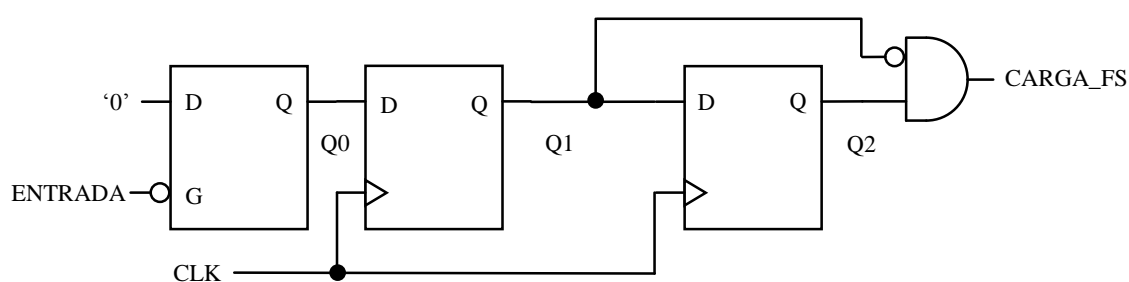


Figura 2-7. Circuito de detección, con flanco de subida, del cambio de nivel en la salida del *latch* (MC3).

En la Figura 2-8 se observa como el *latch* actúa como circuito anti-rebote, generando una señal Q0 libre de ruido.

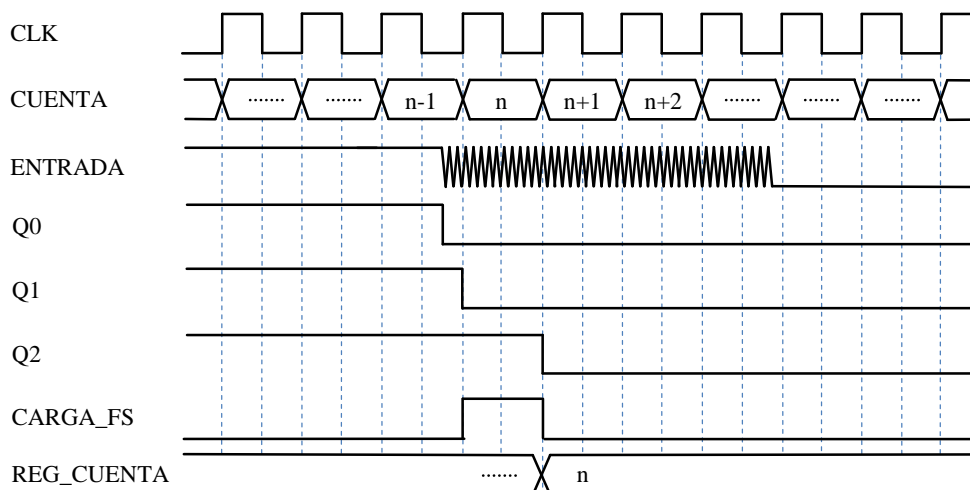


Figura 2-8. Cronograma de la detección, con flanco de subida, del cambio de nivel en la salida del *latch* (MC3).

2.2.4. Módulo de captura 4 (MC4) - Circuito de detección de inicio de tren de pulsos de la señal de descarga

Desafortunadamente, no todas las señales de descarga del condensador generan señales “ENTRADA” como las de la Figura 2-8. Son frecuentes las situaciones donde se detectan cambios de ‘1’ a ‘0’ previos al tren de pulsos producido por la señal de entrada (indicado con un círculo rojo en la Figura 2-9). Considerar estos “*glitches*” como el evento que fija el instante para la medida de tiempo de descarga aumenta la incertidumbre en la conversión tiempo-digital. Por ello, debe modificarse el circuito ligeramente, en lo que al *latch* se refiere, para que estos fenómenos no sean tenidos en cuenta.

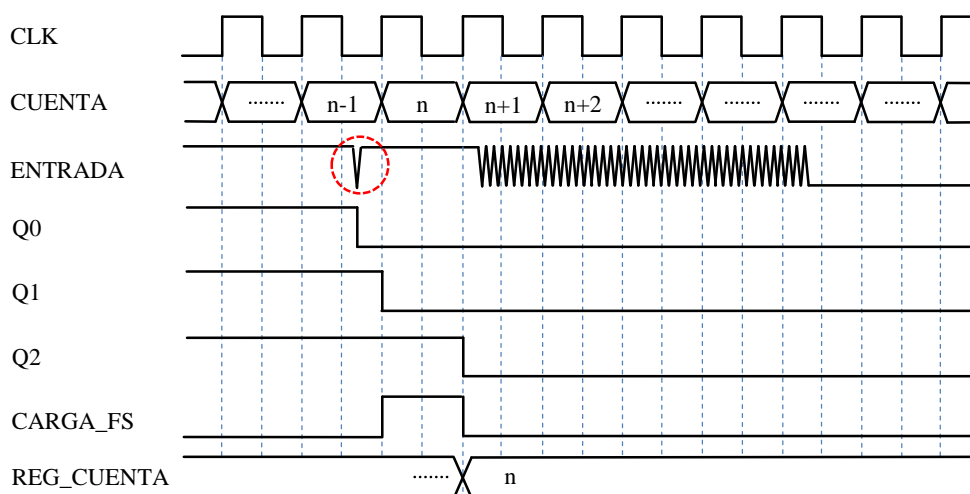


Figura 2-9. Cronograma de la detección de un glitch previo al tren de pulsos.

Se debe detectar no sólo el primer cambio de '1' a '0', sino otros que puedan aparecer posteriormente. El circuito tiene que generar una señal de carga en el registro que guarda la cuenta cada vez que la señal de entrada es '0' durante un semiciclo bajo de la señal de reloj, habiendo tomado el valor '1' durante todo el anterior semiciclo bajo. Así, nuestro criterio para considerar que una bajada en la señal "ENTRADA" es un *glitch* es que en el siguiente semiciclo de reloj bajo, la señal de entrada siempre sea '1'. Parece lógico, en esta situación, entender el *glitch* como una situación debida a un ruido aislado excesivo. El *latch* de la Figura 2-10 realiza ahora una precarga en cada semiciclo alto de CLK, manteniendo el cambio sólo durante el semiciclo previo al flanco con el que se registran los mismos, pudiéndose generar más de un pulso de almacenamiento y quedando el registro actualizado con el valor de cuenta asociado al último cambio detectado.

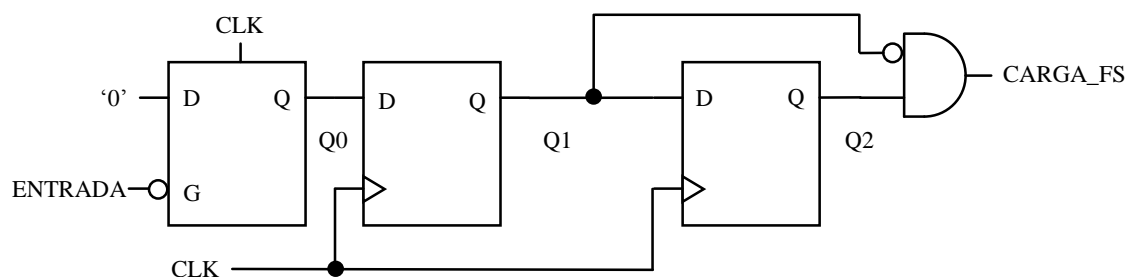


Figura 2-10. Circuito modificado de detección, con flanco de subida, del cambio de nivel en la salida del *latch* (MC4).

Así, se generan tantos pulsos como inicios de tren de pulsos se detecten, sean producidos por *glitches* o no, almacenando el valor de cuenta del último de ellos (Figura 2-11).

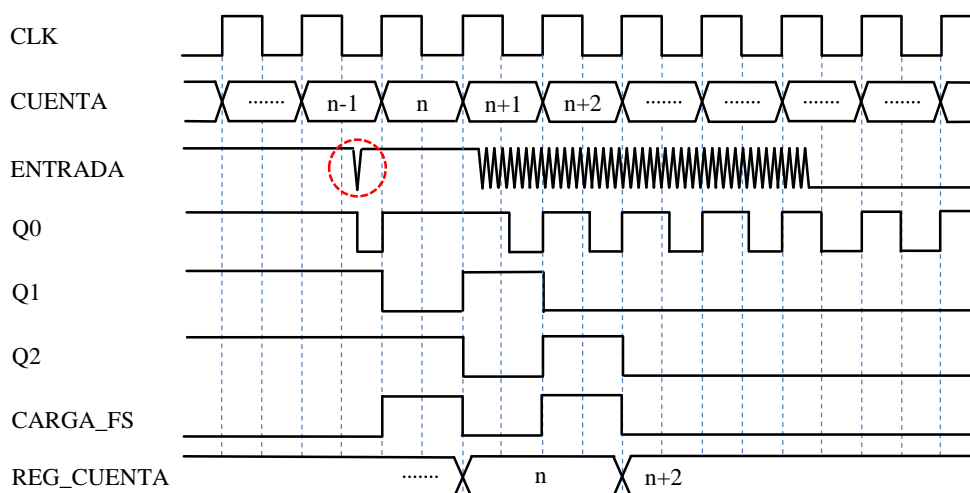


Figura 2-11. Cronograma de la detección de inicio de tren de pulsos por el módulo de captura MC4.

2.2.5. Módulo de captura 5 (MC5) - Circuito de detección de inicio de tren de pulsos de la señal de descarga por flancos de subida y de bajada, promediando dos valores de cuenta

Otra de las ventajas que presentan las FPGAs es la posibilidad de trabajar con *flip-flops* disparados por flancos de subida o por flancos de bajada. Esto permite incrementar la precisión en la medida del tiempo de descarga mediante el uso en paralelo de *flip-flops* disparados por ambos flancos. El circuito equivalente al que aparece en la Figura 2-10 pero usando flanco de bajada se muestra en la Figura 2-12. El valor medio de los dos valores de cuenta obtenidos de forma independiente por ambos circuitos, proporciona un número con un bit más de información, reduciendo de esta forma la incertidumbre debido al ruido de disparo en $1/\sqrt{2}$.

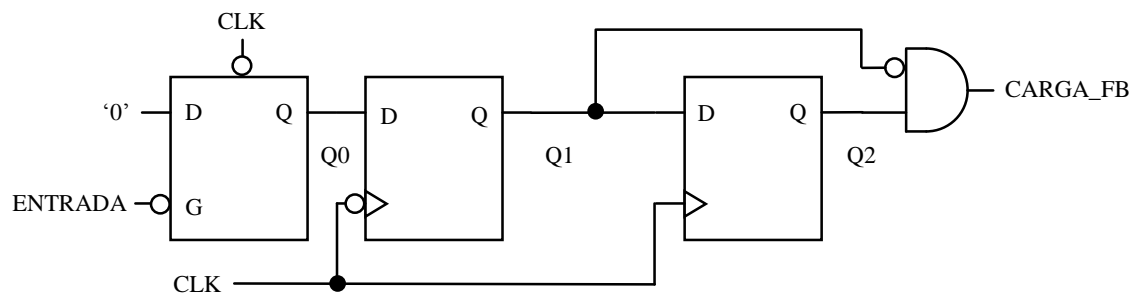


Figura 2-12. Circuito modificado de detección, con flanco de bajada, del cambio de nivel en la salida del *latch* (MC5).

El cronograma de la Figura 2-13 resume el comportamiento de los dos circuitos propuestos para la detección del inicio del tren de pulsos generado a la salida del *buffer* de entrada de la FPGA por la señal de descarga, donde con el sufijo “FS” se hace referencia a las señales del circuito de la Figura 2-10 y con “FB” a las señales del circuito de la Figura 2-12.

Mediante el uso en paralelo de los circuitos de las Figura 2-10 y Figura 2-12 se consigue un funcionamiento similar al que se obtiene con sólo uno de ellos pero doblando la frecuencia de reloj.

La solución propuesta incrementa el hardware utilizado, de una forma muy reducida teniendo en cuenta los recursos de una FPGA, mientras que, al usar una frecuencia de reloj más baja, los ruidos producidos en el sistema por la actividad de la propia FPGA son menores, consiguiendo de esta manera una menor incertidumbre en las medidas como se muestra posteriormente.

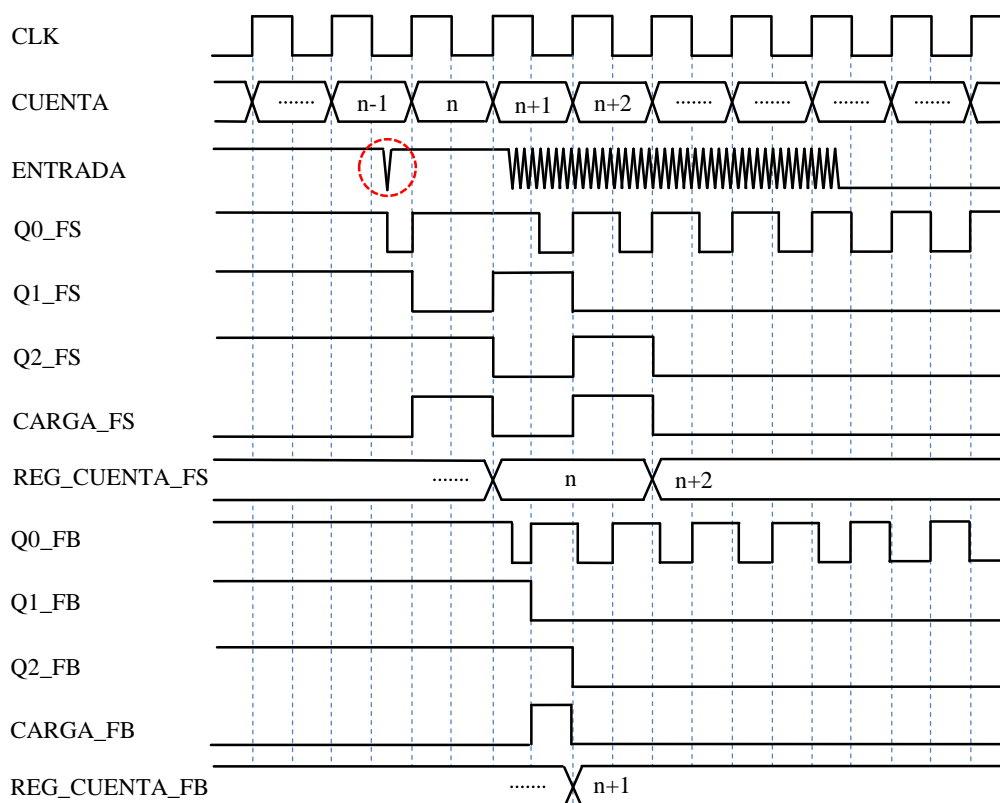


Figura 2-13. Cronograma de la detección de inicio de tren de pulsos por flancos de subida y bajada.

2.2.6. Módulo de captura 6 (MC6) - Circuito de detección de inicio y final de tren de pulsos de la señal de descarga por flancos de subida y de bajada, promediando cuatro valores de cuenta

Por otro lado, no existe constancia en la literatura del uso del final del tren de pulsos como medida del tiempo de descarga a V_{TL} del condensador. Sin embargo, ésta puede ser una medida igual de válida para la detección de la descarga. Es más, combinando ambas medidas se puede ofrecer una mayor información sobre el tiempo de descarga del condensador al tratarse de dos medidas independientes en el tiempo.

Para poder aprovechar este hecho y de forma similar a la empleada para detectar el inicio del tren de pulsos de la señal de entrada, se pueden introducir dos circuitos que detecten el final del mismo, también por flanco de subida y de bajada, con la idea de obtener nuevos valores de forma independiente que midan el tiempo de descarga y consiguiendo, con el promedio de las cuatro cuentas, reducir aún más la incertidumbre de los datos obtenidos.

El circuito de la Figura 2-14 es el encargado de detectar el final del tren de pulsos por flanco de subida. En este caso, la salida del *latch* toma el valor lógico ‘0’ si la entrada es ‘1’ durante el semiciclo bajo previo al flanco de subida de reloj. Por esto, el circuito de detección de flanco debe ser modificado para detectar un cambio de la señal Q0 de ‘0’ a ‘1’. Los biestables con salida Q1 y Q2 son inicializados a ‘0’ durante la fase de carga, para partir desde ese valor al comenzar la fase de descarga.

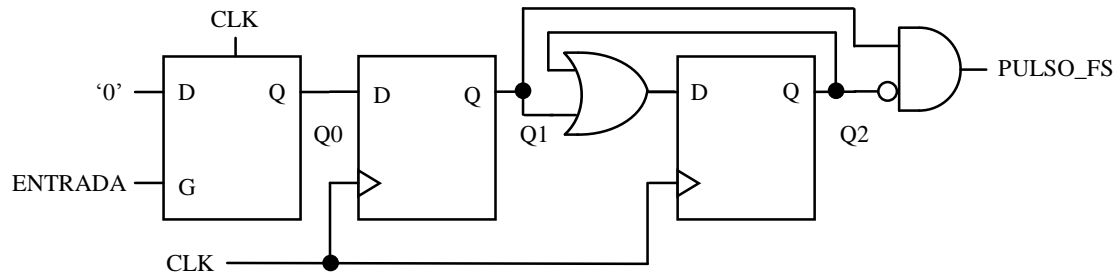


Figura 2-14. Circuito de detección de final de tren de pulsos por flanco de subida (MC6).

La Figura 2-15 presenta el circuito para la detección del final del tren de pulsos por flanco de bajada.

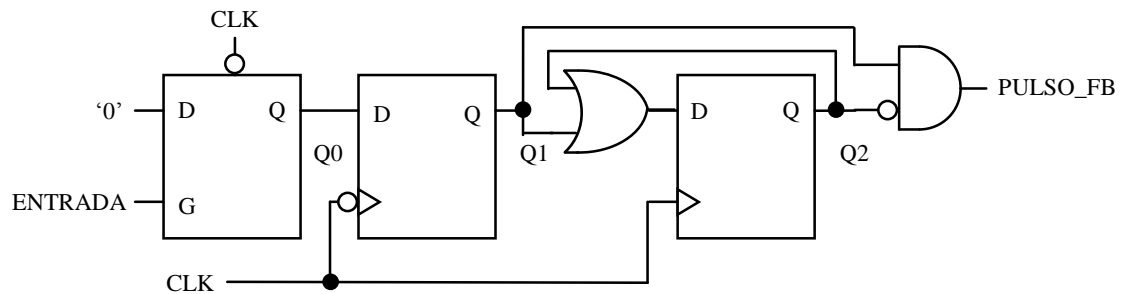


Figura 2-15. Circuito de detección de final de tren de pulsos por flanco de bajada (MC6).

Los circuitos que detectan el inicio del tren de pulsos son inmunes a los *glitches* que se puedan producir anteriormente. En el caso de los circuitos que detectan el final del tren de pulsos, el valor de cuenta no debe actualizarse por *glitches* que se puedan detectar después de dicho final, como el que aparece indicado con un círculo en la señal “ENTRADA” de la Figura 2-16. La inclusión de una puerta OR a la entrada del último *flip-flop* garantiza que la condición de carga se produce una sola vez por cada ciclo de descarga.

El cronograma de la Figura 2-16 muestra el comportamiento del circuito propuesto para la detección del final de tren pulsos. Se puede observar cómo el *glitch* que se produce después del final del tren de pulsos no influye en el valor de cuenta almacenado, en este caso, por parte del circuito que muestrea con flanco de bajada.

El valor de la conversión tiempo-digital para cada ciclo de descarga se estima, por tanto, a partir de cuatro valores obtenidos de forma independiente: inicio de tren de pulsos, por flanco de subida y de bajada, y final de tren de pulsos, igualmente por flanco de subida y de bajada. Con esto, para el caso de señales de descarga con una velocidad de cambio lenta, considerando las cuatro medidas independientes entre sí, la incertidumbre debida al ruido de disparo en la conversión tiempo-digital se reduce a la mitad, si se la compara con la que se obtiene usando un único valor.

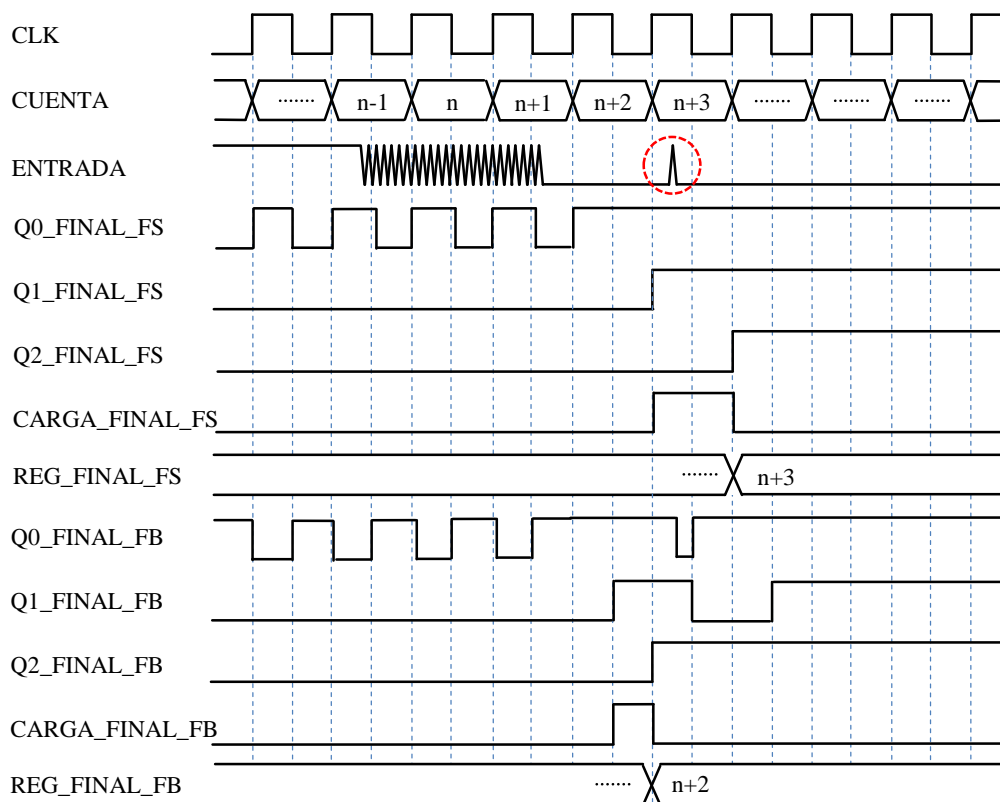


Figura 2-16. Cronograma de la detección de final de tren de pulsos por flancos de subida y bajada.

A continuación (Figura 2-17, Figura 2-18 y Figura 2-19), se muestran algunas capturas de osciloscopio de los circuitos propuestos. La señal analógica que aparece en cada una de las figuras no se corresponde exactamente con la señal interna que llega a la entrada de los *latches* puesto que, para observarla en el exterior de la FPGA, ésta pasa a través de un *buffer* de salida.

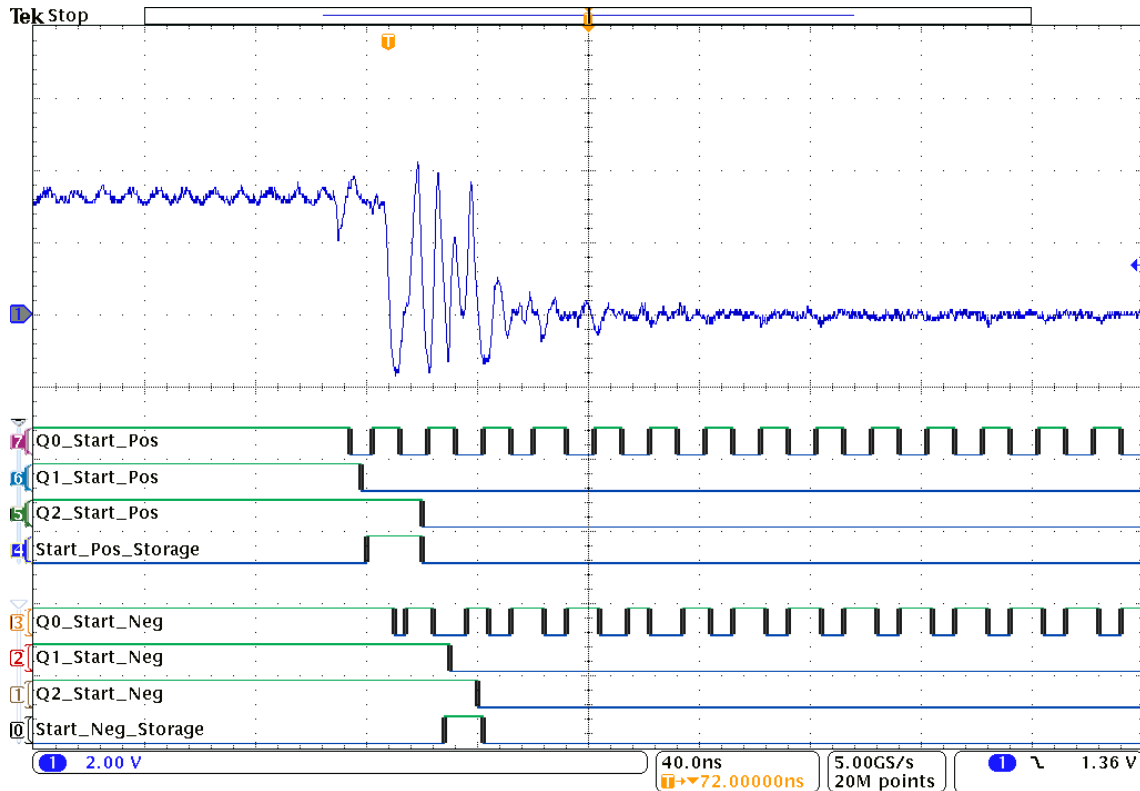


Figura 2-17. Captura del circuito de detección de inicio de tren de pulsos (circuitos Figura 2-10 y Figura 2-12).

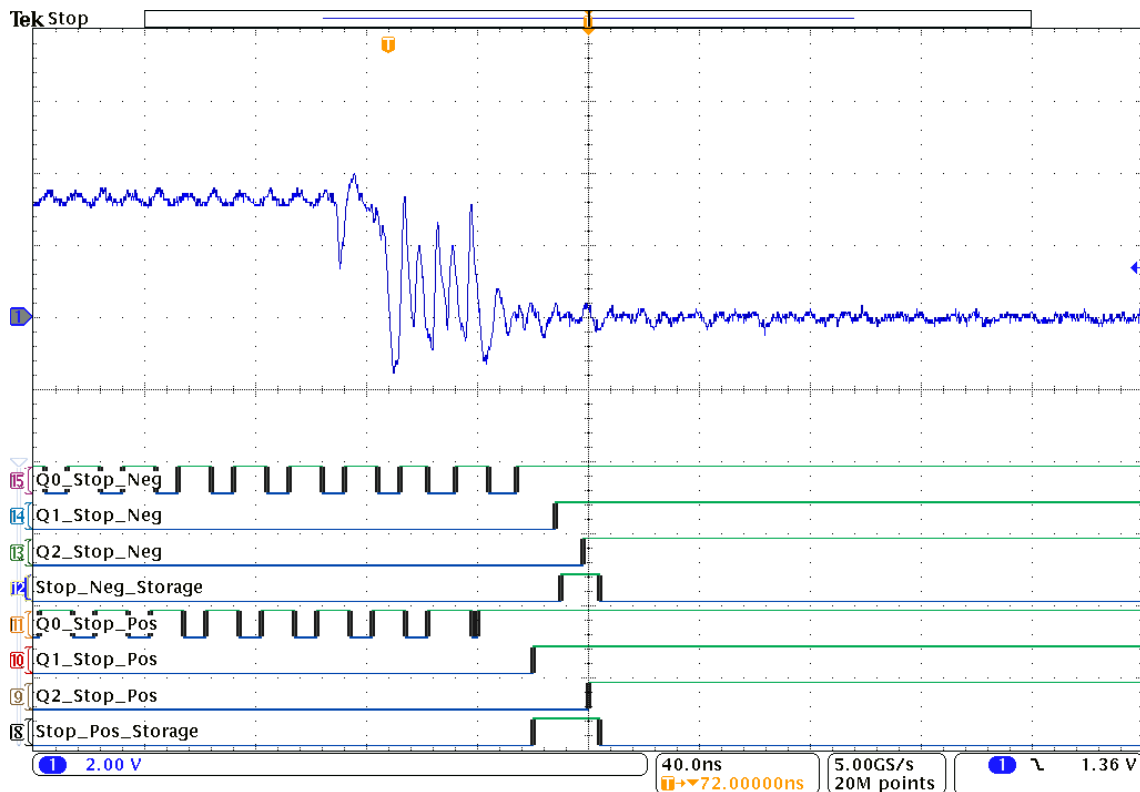


Figura 2-18. Captura del circuito de detección de final de tren de pulsos (circuitos Figura 2-14 y Figura 2-15).

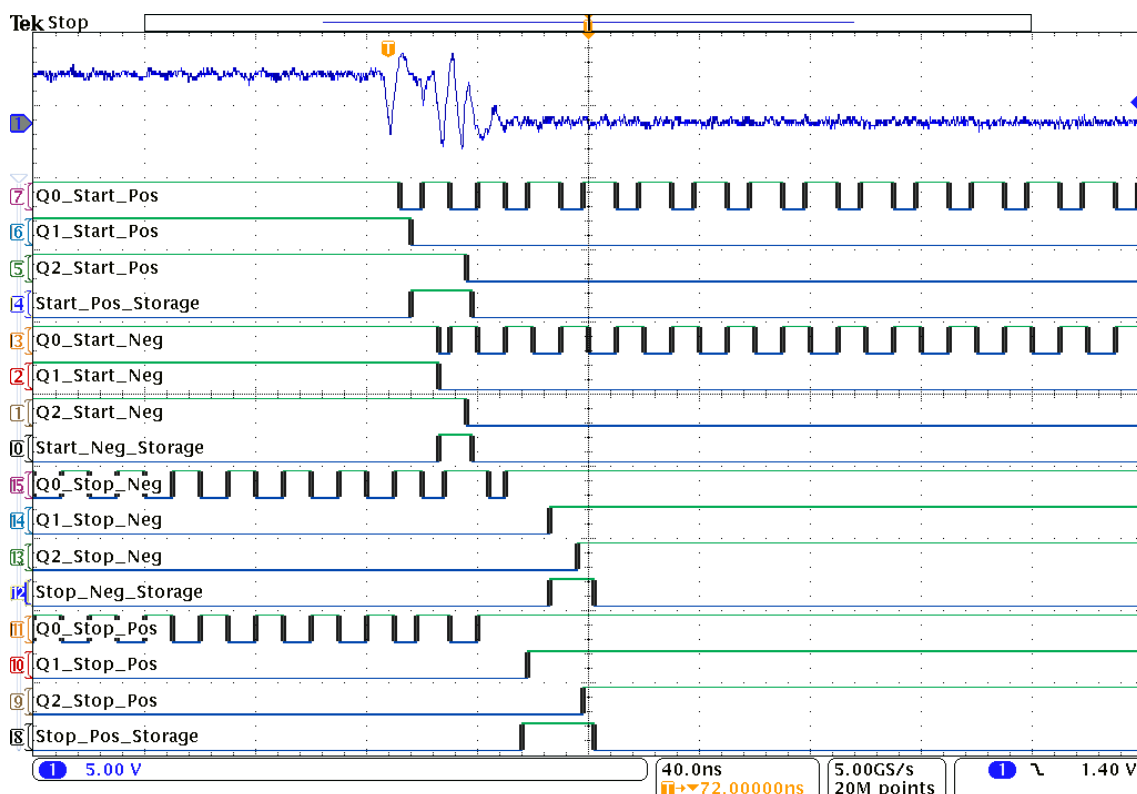


Figura 2-19. Captura del circuito de detección de inicio y final de tren de pulsos (circuitos Figura 2-10, Figura 2-12, Figura 2-14 y Figura 2-15).

El Anexo A incluye el código VHDL del circuito aquí descrito.

2.3. Análisis de la resolución en la conversión tiempo-digital

Como se indica con anterioridad en el apartado 1.2, las fuentes de incertidumbre para la media del tiempo de descarga son: la inestabilidad de la frecuencia del oscilador usado por los contadores, los efectos de cuantificación de la conversión tiempo-digital y el ruido de disparo.

Los efectos de la inestabilidad de la frecuencia del circuito oscilador son despreciables si se usan osciladores de cristal de cuarzo, cuya estabilidad es igual o mayor a ± 50 ppm, y la conversión se realiza sobre señales con una velocidad de cambio lenta.

Por otro lado, la incertidumbre debida a la cuantificación sólo es tomada en cuenta en el punto de parada ya que el inicio de cuenta se produce de forma síncrona con el inicio de la descarga. Esto provoca que no se dé la ambigüedad $-T_s$, $+T_s$ sino $[0, T_s]$. Del mismo modo, los efectos del ruido de disparo sólo afectan al punto de parada del contador.

A continuación se presenta un estudio para evaluar la incertidumbre en la conversión tiempo-digital debida al ruido de disparo a partir de los tiempos obtenidos empleando los distintos módulos de captura descritos anteriormente.

Las diferentes propuestas pretenden mejorar la incertidumbre debida al ruido de disparo, incrementando con ello los bits efectivos (ENOB) de la conversión tiempo-digital.

2.3.1. Materiales y método

El circuitos propuestos se han realizado sobre una FPGA Spartan3AN de Xilinx (XC3S50AN-4TQG144C) [Xilinx, Inc. 2014] haciendo uso de un circuito oscilador basado en un cristal de cuarzo con una frecuencia de trabajo de 50MHz. El contador del módulo de captura es de 14 bits, con una base de tiempo de 20 ns.

Las reglas de diseño recomendadas por el fabricante de la FPGA se aplican de forma rigurosa para que la influencia del ruido que afecta a la tensión de referencia en la detección del cambio de '1' a '0' por parte de los *buffers* de entrada produzca la menor incertidumbre en la medida. Las FPGAs de Xilinx trabajan con tensiones de alimentación independientes para los bloques de entrada/salida y el núcleo, donde el resto de la circuitería reside. Por este motivo, se hace necesario el uso de dos reguladores de tensión independientes para alimentar a la FPGA, reduciendo con ello la influencia de la actividad del núcleo del dispositivo en los *buffers* de entrada/salida. Los reguladores escogidos, TPS79633 [Texas Instruments, Inc. 2015b] para la tensión de los bancos de entrada/salida alimentados a 3,3 V y TPS79912 [Texas Instruments, Inc. 2015c] para la tensión del núcleo a 1,2 V, poseen unos valores extremadamente bajos tanto para la caída de tensión durante su régimen de carga máxima (*dropout voltage*) como para el ruido de la tensión de salida ($40\mu\text{V}_{\text{RMS}}$). Además, necesitan de muy pocos componentes externos para su correcto funcionamiento, lo que los hace ideales para aplicaciones donde el área ocupada por el circuito es importante. Para cada uno de los cuatro bancos de los que consta la FPGA y para el núcleo, se usa una batería de condensadores de desacoplo de diferentes valores en un emplazamiento muy próximo a las entradas de alimentación y conectados a través de dos planos de alimentación dedicados, uno para la tensión de 3,3 V y otro para GND. La placa de circuito impreso sobre la que está montada el circuito ha sido fabricada usando un sustrato de fibra de vidrio de tipo FR-4 y cuatro capas, dejando las capas

internas para los planos de alimentación y las externas para el resto de señales. En el capítulo 4 se describe el circuito con más detalle.

El análisis de la incertidumbre en el valor de cuenta se analiza para cada uno de los circuitos anteriormente propuestos para el módulo de captura:

- Circuito 1 (MC1) – Circuito síncrono de detección de cambio de nivel de la señal de descarga (Figura 2-2).
- Circuito 2 (MC2) – Circuito síncrono de detección de cambio de nivel de la señal de descarga con generación de pulso único (Figura 2-5).
- Circuito 3 (MC3) – Circuito de detección de cambio de nivel de la señal de descarga con *latch* anti-rebote (Figura 2-7).
- Circuito 4 (MC4) – Circuito de detección de inicio de tren de pulsos de la señal de descarga (Figura 2-10).
- Circuito 5 (MC5) – Circuito de detección de inicio de tren de pulsos de la señal de descarga por flancos de subida y de bajada (Figura 2-10 y Figura 2-12), promediando dos valores de cuenta.
- Circuito 6 (MC6) – Circuito de detección de inicio y final de tren de pulsos de la señal de entrada por flancos de subida y de bajada (Figura 2-10, Figura 2-12, Figura 2-14 y Figura 2-15), promediando cuatro valores de cuenta.

El valor de resistencia sobre el que se va a realizar el análisis, coincide con el valor máximo del rango valores ofrecidos por un sensor de presión piezorresistivo desarrollado por CIDETEC dentro del proyecto TEC2012-38653-C02-01 [*Castellanos-Ramos et al. 2009*]. Este sensor disminuye el valor de resistencia a medida que se incrementa la presión ejercida sobre él, y su rango puede variar entre valores entorno a los 7350 Ω para pequeñas presiones de unos pocos kilopascales hasta las decenas de ohmios para presiones elevadas. Por ello, para simular la respuesta del módulo de captura, se elige como valor máximo del sensor una resistencia de 7350 Ω con una tolerancia del 0,1 %. En el caso del condensador, se escoge uno de 47nF con una tolerancia de ± 5 %, que cumple con las reglas de diseño propuestas por [*Reverter et al. 2004*] para la constante de tiempo óptima que minimice la incertidumbre estándar relativa de la medida.

Para cada módulo de captura propuesto, se realiza un análisis estadístico a partir de 500 muestras, obteniendo su histograma, así como su desviación estándar $s(X)$ que, como se

indica en el apartado 1.3.3, sirve para estimar la incertidumbre de la medida debido al ruido de disparo.

2.3.2. Resultados experimentales

Las resistencias de valores bajos generan una señal de descarga con una mayor velocidad de cambio en el punto de disparo que la que producen las resistencias de valores altos, por lo que los resultados obtenidos para este tipo de señales son muy similares para todas las opciones planteadas. Sin embargo, para las resistencias de mayor valor, donde la velocidad de cambio de la señal de descarga del condensador es la más lenta, los resultados son bien distintos. Por esto, las pruebas sólo se realizan sobre el valor de resistencia máxima del sensor.

Según (1.13), la resolución de la conversión tiempo-digital, sin considerar los efectos del ruido de disparo, es de 13,88 bits.

A continuación se presenta cual es la resolución efectiva de la conversión, teniendo en cuenta el ruido de disparo para cada uno de los seis circuitos propuestos.

2.3.2.1. Incertidumbre de disparo del módulo de captura MC1

La Figura 2-20 muestra el histograma para la resistencia de 7350 Ω , obteniendo un valor de desviación estándar de 2,14 cuentas, lo que expresado en tiempo equivale a 42,9 ns. Si se aplica la ecuación (1.15) para la obtención de los bits efectivos de la conversión tiempo-digital, el ENOB resultante es de 10,98 bits, dando lugar a una resolución en ohmios según (1.16) de 3,54 Ω .

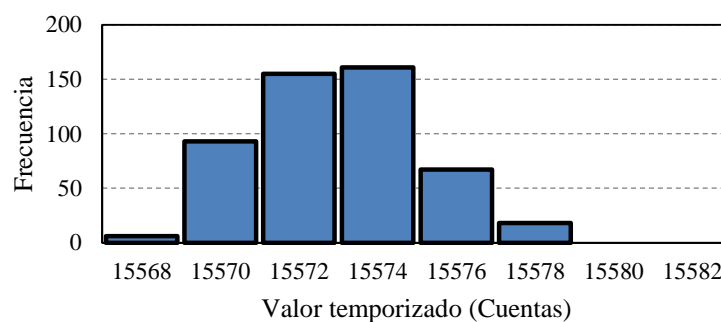


Figura 2-20. MC1 - Histograma para 500 muestras sobre una resistencia de 7350 Ω .

2.3.2.2. Incertidumbre de disparo del módulo de captura MC2

Para el módulo basado en el circuito 2, se obtiene un valor de desviación estándar de 1,83 cuentas, lo que expresado en tiempo equivale a 36,7 ns. Esto hace que los bits de resolución efectiva sean 11,2 bits y la resolución en ohmios resultante sea de 3,04 Ω . La Figura 2-21 muestra el histograma obtenido con este circuito.

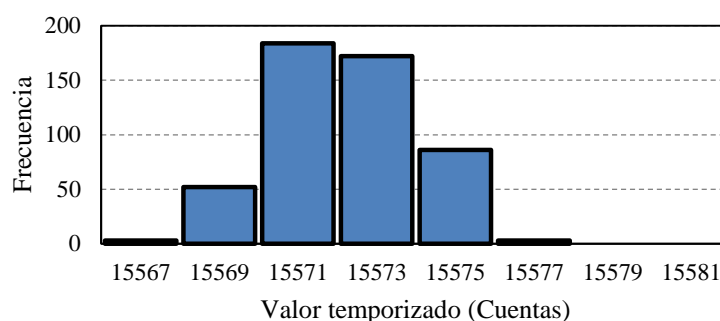


Figura 2-21. MC2 - Histograma para 500 muestras sobre una resistencia de 7350 Ω .

2.3.2.3. Incertidumbre de disparo del módulo de captura MC3

La Figura 2-22 muestra el histograma para módulo de captura basado en el circuito 3, obteniendo un valor de desviación estándar de 2,13 cuentas (42,7 ns). Se observa que los resultados son muy próximos a los obtenidos con el circuito 1, por lo que puede deducirse que el circuito anti-rebote no mejora la incertidumbre debida al disparo. La posible detección de *glitches* puede causar este comportamiento. El ENOB para este circuito queda en 10,98 bits con una resolución en ohmios de 3,53 Ω .

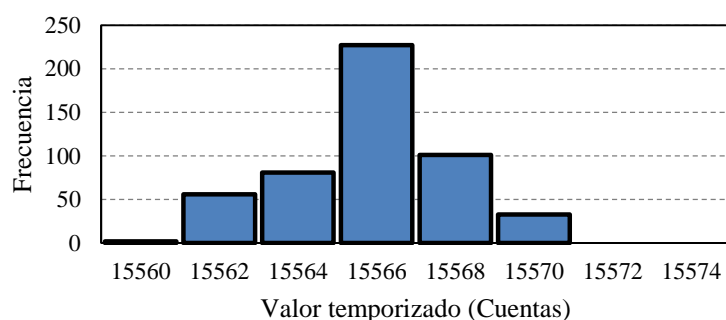


Figura 2-22. MC3 - Histograma para 500 muestras sobre una resistencia de 7350 Ω .

2.3.2.4. Incertidumbre de disparo del módulo de captura MC4

La propuesta llevada a cabo por el módulo de captura basado en el circuito 4, mejora la incertidumbre de la propuesta anterior pero no de forma significativa. La desviación estándar expresada en cuentas es de 1,9, lo que lleva a 38,1 ns expresado en tiempo. Los bits efectivos teniendo en cuenta el ruido de disparo suben a 11,14 bits, que se traduce en una resolución en ohmios de 3,16 Ω . La Figura 2-23 muestra el histograma obtenido por este módulo.

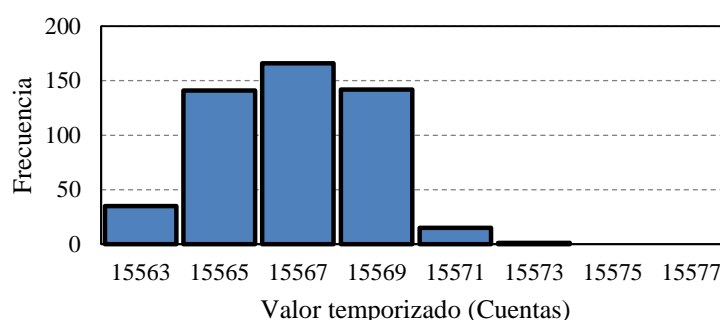


Figura 2-23. MC4 - Histograma para 500 muestras sobre una resistencia de 7350 Ω .

2.3.2.5. Incertidumbre de disparo del módulo de captura MC5

En este módulo, los resultados de la conversión tiempo-digital se obtienen a partir del promedio de dos medidas independientes. La Figura 2-24 muestra el histograma resultante. La desviación estándar es de 1,31 cuentas (26,1 ns expresado en tiempo), y se puede observar cómo es inferior a la obtenida por el módulo basado en el circuito 4 donde un único valor de cuenta es almacenado. Se cumple por tanto lo expuesto en el apartado 2.2.5 y la incertidumbre se reduce, aproximadamente, en un factor de $1/\sqrt{2}$. El ENOB obtenido es de 11,67 bits con una resolución en ohmios de 2,19 Ω .

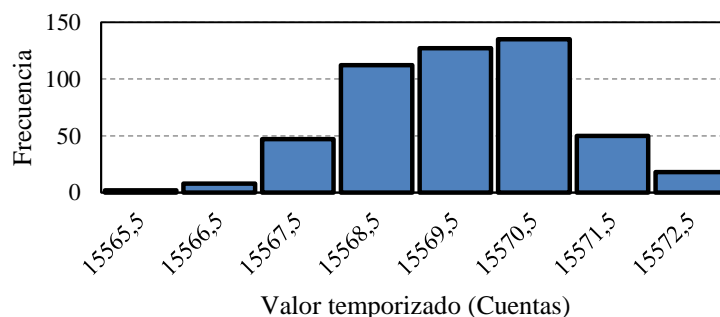


Figura 2-24. MC5 - Histograma para 500 muestras sobre una resistencia de 7350 Ω .

2.3.2.6. Incertidumbre de disparo del módulo de captura MC6

Por último, se analiza el módulo de captura basado en el circuito 6, donde el número digital resultante de la conversión se calcula a partir del promedio de cuatro valores medidos de forma independiente (principio y final del tren de pulsos de la señal de entrada, muestreados por flancos de subida y de bajada). La Figura 2-25 muestra el histograma resultante del experimento, obteniendo un valor de desviación estándar de 1,02 cuentas (20,4 ns).

De nuevo se puede ver cómo la incertidumbre se ve reducida con respecto al módulo que se basa en solo un valor de cuenta, en este caso a la mitad.

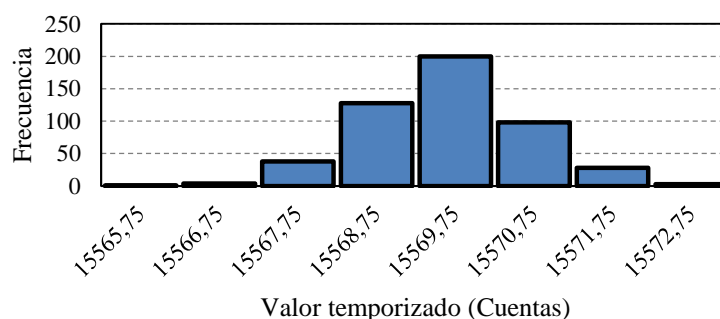


Figura 2-25. MC6 - Histograma para 500 muestras sobre una resistencia de 7350 Ω .

Estos resultados dan lugar a un ENOB de 12,01 bits y a una resolución, expresada en ohmios, de 1,74 Ω .

2.3.3. Discusión

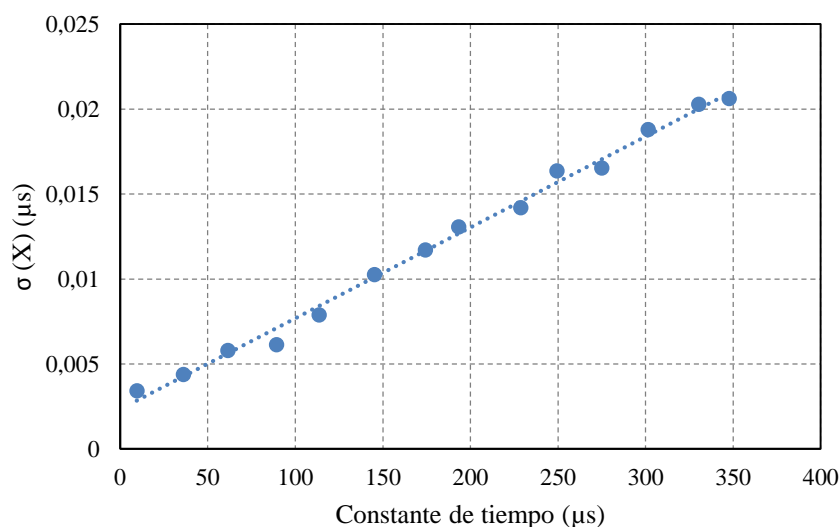
En la Tabla 2-1 se resumen los resultados del análisis de la resolución obtenido por cada uno de los circuitos y se observa cómo el módulo MC2 mejora ligeramente el inicialmente propuesto (MC1). Por otro lado, el primer módulo que emplea un *latch* como elemento de memoria (MC3) no mejora a los anteriormente propuestos. De hecho, obtiene peores resultados que el módulo MC2. Sin embargo, el módulo MC4, que modifica el comportamiento de MC3 filtrando los glitches que pudiesen aparecer a la salida del *buffer* de entrada, obtiene mejores resultados. Por último, los módulos de captura que explotan ambos flancos de la señal de reloj para sincronizar los flip-flops y llevan a cabo el promedio de dos (MC5) y de cuatro valores (MC6), proporcionan los mejores resultados, alcanzando una resolución de 1,70 Ω .

Tabla 2-1. Análisis de la incertidumbre de los módulos de captura propuestos ($R_x = 7350 \Omega$).

	$\sigma(X)$ (cuentas)	$\sigma(X)$ (ns)	$u(y)$ (ns)	ENOB (bits)	Resolución (Ω)
MC1	2,14	42,8	43,2	10,98	3,54
MC2	1,83	36,7	37,1	11,20	3,04
MC3	2,13	42,7	43,0	10,98	3,53
MC4	1,90	38,1	38,5	11,14	3,16
MC5	1,31	26,1	26,7	11,67	2,19
MC6	1,02	20,4	21,2	12,04	1,70

En cuanto al ancho de banda, especialmente relevante si se adquieren datos de muchos sensores, está limitado por la constante de tiempo $R_x C$, dando lugar a un compromiso entre velocidad y precisión. Si sólo se tiene en cuenta los efectos del ruido de cuantificación, la precisión mejora conforme la constante de tiempo es aumentada.

Por otro lado, en la Figura 2-26 se muestra la desviación estándar experimental para $C = 47 \text{ nF}$ y un conjunto de 14 valores de resistencias distintas dentro del rango del sensor propuesto (200Ω , 762Ω , 1300Ω , 1890Ω , 2400Ω , 3070Ω , 3680Ω , 4100Ω , 4840Ω , 5300Ω , 5820Ω , 6400Ω , 7000Ω y 7350Ω) obtenidas con el módulo de captura MC6. Se observa cómo, para constantes de tiempo grandes, la influencia del ruido de disparo supera al error de cuantificación, aumentando la desviación estándar de las medidas.

Figura 2-26. Desviación estándar experimental $s(X)$ en función de la constante de tiempo para el módulo MC6.

Sin embargo, si se añaden ambas fuentes de error, se puede obtener la incertidumbre estándar relativa de y a partir de (1.12). La Figura 2-27 muestra los resultados obtenidos.

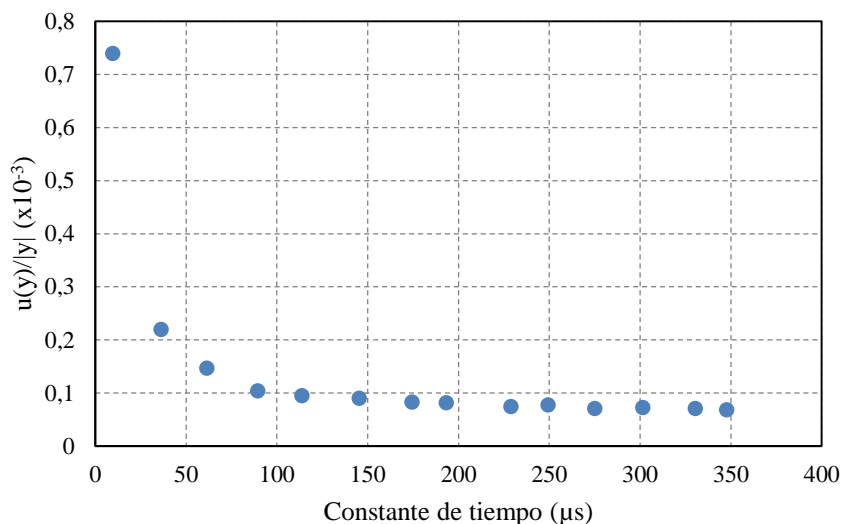


Figura 2-27. Incertidumbre estándar relativa en función de la constante de tiempo para el módulo MC6.

Se puede apreciar como la incertidumbre relativa aumenta para constantes de tiempo pequeñas. Como se comenta anteriormente, esto se debe a que el ruido de cuantificación es la principal causa del error, por lo que si el rango de tiempo es corto, la precisión viene limitada por el periodo de la señal de reloj. Sin embargo, la incertidumbre relativa disminuye para constantes de tiempo mayores, decreciendo muy ligeramente para valores superiores a 100 μs (Figura 2-27). Por lo tanto, para este valor se alcanza un buen compromiso entre precisión y ancho de banda. Debe notarse que el condensador puede ser elegido para establecer la constante de tiempo requerida para un rango de resistencia dado, y puede ser cambiado, para mantener la misma incertidumbre relativa, si otro rango de resistencias debe ser empleado. Esto podría requerir incrementar el número de bits del contador que mide el tiempo de descarga del circuito RC pero no afectaría al rendimiento de los módulos de captura.

En relación con el consumo del circuito, su componente principal está relacionada con la carga del condensador dada por $1/2 CV_{CC}^2 f$, donde f es la frecuencia del ciclo de carga y descarga. Obviamente, a mayor ancho de banda, mayor consumo y una precisión menor dado que la constante de tiempo debe ser reducida.

Con respecto al coste hardware, la Tabla 2-2 resume los recursos empleados por los distintos módulos de captura. Los módulos MC5 y MC6 son los que usan un mayor número de ellos, aunque esto no es una limitación seria debido al amplio catálogo de FPGAs existentes en el mercado. Si la aplicación específica lo requiriese, una FPGA con más recursos podría ser empleada.

Tabla 2-2. Recursos hardware empleados por los distintos módulos de captura.

	MC1	MC2	MC3	MC4	MC5	MC6
Slices	3	3	3	3	24	45
4 inputs Look Up Tables	19	19	19	19	39	99
Flip Flops	35	35	35	35	56	92
Latches	0	0	1	1	2	4

2.3.4. Conclusiones

El uso de FPGAs permite el diseño de circuitos de conversión tiempo-digital basados en la medida de tiempo de descarga sin la necesidad de usar *buffers* ST, lo cual, además de reducir el hardware utilizado, mejora la calidad de la medida.

La principal ventaja de las FPGAs es su gran versatilidad y, hasta para este tipo de aplicaciones donde no trabajan como un circuito digital al uso, se pone de manifiesto. Los elementos de almacenamiento de las FPGAs pueden configurarse como biestables activos por nivel o disparados por flanco. Aprovechando estas características, se proponen distintos módulos de captura que mejoran los bits efectivos en la conversión tiempo-digital y por lo tanto la resolución en ohmios de la medida de resistencia.

El módulo de captura MC6 obtiene unos resultados muy superiores si se compara con los que basan la medida de conversión en la captura del valor de cuenta en un solo evento. La idea detrás de esta aproximación que promedia distintos valores de cuenta es filtrar parte del ruido de disparo sin afectar al ancho de banda. Si se requiriese una alta precisión, por ejemplo, en el caso de sensores no lineales con regiones de diferente sensibilidad, el módulo MC6 logra una precisión de 12 bits efectivos para el ejemplo de un sensor táctil, empleando un tiempo de medición de 348 μ s (teniendo en cuenta la carga y descarga del condensador).

El coste hardware del circuito está justificado puesto que las FPGAs actuales disponen de gran cantidad de elementos lógicos programables que, sumado al control sobre la electrónica a bajo nivel y la posibilidad, incluso, de instanciar bloques IP (*Intellectual Property*) de la complejidad de un microcontrolador o incluso un DSP (*Digital Signal Processor*) los hacen muy interesantes para este tipo de aplicaciones.

2.4. Análisis de la exactitud en la medida de resistencia

Como se indica en el apartado 1.2, según (1.2) y un conjunto de parámetros, se puede deducir el valor de una resistencia desconocida midiendo el tiempo de descarga de un condensador. El problema reside en el hecho de que sólo se conoce el valor nominal de estos parámetros y que, además, sus valores cambian con el tiempo y la temperatura. Por todo ello, se recomienda el uso de técnicas de calibración donde el valor de la resistencia a medir dependa de uno o más componentes de referencia y no de los parámetros anteriormente citados.

Ya se han comentado las técnicas de calibración de un punto y de dos puntos para deducir el valor de R_x a partir de una y de dos resistencias de referencia respectivamente. A continuación se propone otra técnica de calibración que aproxima el valor de la resistencia a partir de polinomios obtenidos por ajuste de curva por mínimos cuadrados, haciendo uso de una única resistencia de calibración.

2.4.1. Calibración de un punto mediante aproximación polinómica

Como se menciona en el apartado 1.4.3, el error en la estimación del valor de R_x se debe, principalmente, a la resistencia interna de los *buffers* de la FPGA con los que se descarga el condensador a través de R_x y de la resistencia de calibración R_{cl} . Si éstas son tenidas en cuenta, a partir de las ecuaciones (1.18) y (1.21) se obtiene

$$R_x^* = \frac{N_x}{N_{cl}} R_{cl} = \frac{R_{cl}}{R_{cl} + R_{n,cl}} R_x + \frac{R_{cl} R_{n,x}}{R_{cl} + R_{n,cl}} \quad (2.1)$$

donde despejando R_x

$$R_x = \left(\frac{N_x}{N_{cl}} R_{cl} - \frac{R_{cl} R_{n,x}}{R_{cl} + R_{n,cl}} \right) \frac{R_{cl} + R_{n,cl}}{R_{cl}} = \frac{N_x}{N_{cl}} (R_{cl} + R_{n,cl}) - R_{n,x} \quad (2.2)$$

Como se puede ver, la ecuación (2.2) se corresponde con una recta cuya pendiente y ordenada en el origen se pueden obtener a partir de los valores de las resistencias internas y de calibración. Por lo tanto, si estos coeficientes son conocidos, la ecuación (2.2) proporciona el valor “real” de R_x .

La caracterización de estas resistencias es un proceso costoso, necesitándose la inclusión de un circuito divisor de tensión continua para su estimación. El puerto por donde se

descarga el condensador, se configura como salida fijando un '0' lógico y es conectado a V_{DD} a través de una resistencia externa conocida. De esta forma, midiendo la tensión de salida se puede inferir la resistencia del *buffer*.

Esta aproximación no tiene en cuenta que R_n varía en función de la corriente que drena el puerto, y ésta puede cambiar por dos causas: los distintos valores que R_x puede tomar dentro del rango de trabajo del sensor, y que la tensión en el condensador no es constante para toda la fase de descarga.

En este apartado se propone un sencillo método para obtener los coeficientes de la recta de la ecuación (2.2), donde los valores de las resistencias de los *buffers* son tenidas en cuenta, evitando la necesidad de caracterizarlas con otro circuito que no sea el propio de la interfaz directa. Se trata de medir dos resistencias conocidas ya que, de esta forma, dos ecuaciones son obtenidas donde la pendiente y el cero son las incógnitas. El procedimiento de calibración basado en esta caracterización será referenciado como “calibración con caracterización lineal de dos puntos” (CLChar2).

Esta técnica puede ser generalizada para contemplar una aproximación lineal más precisa o una dependencia no lineal de R_x sobre el cociente N_x / N_{c1} , debido a un modelo más preciso de la impedancia de entrada de los puertos.

Resumiendo, mediante un modo de pre-calibrado se obtiene el polinomio con el que aproximar las medidas de resistencia en modo de operación normal, donde, por cada uno de los ciclos de medida de resistencia, un valor de cuenta N_x para la resistencia a medir así como otro N_{c1} para la resistencia de calibración son obtenidos. A partir de estas dos medidas se calcula el cociente N_x / N_{c1} (ratio), siendo este valor el que se sustituya en el polinomio de aproximación para obtener el valor estimado de R_x .

2.4.2. Materiales y método

Para llevar a cabo este estudio se realiza una modificación sobre el circuito utilizado en el apartado 2.3.1. En este caso, se han añadido tres resistencias de calibración: una con el valor medio del rango de resistencias del sensor ($R_{c1} = 3500 \Omega$), utilizada por la técnica de calibración de un punto y por nuestra propuesta, y otras dos resistencias ($R_{c2} = 1300 \Omega$ y $R_{c3} = 5800 \Omega$) con valores del 15% y el 85% del rango del sensor, para llevar a cabo la calibración de dos puntos [Pallàs-Areny et al. 2004]. De esta forma, se pueden evaluar

las tres técnicas de calibración de forma simultánea. La Figura 2-28 representa el circuito empleado para la evaluación de la exactitud en la medida de resistencias.

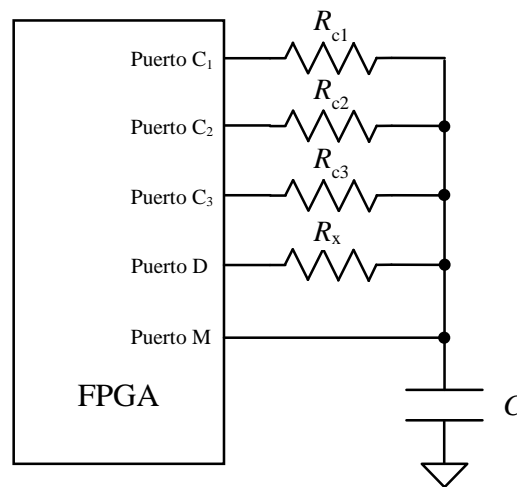


Figura 2-28. Circuito con resistencias de calibración.

El módulo de captura implementado ha sido el propuesto en el apartado 2.2.6, que obtiene el número digital de la conversión promediando cuatro valores (MC6). El rango de resistencias a medir se corresponde, de nuevo, con el que presenta el sensor de presión piezorresistivo desarrollado por CIDETEC (desde 200 Ω a 7350 Ω).

Como se explica en el apartado 2.4.1, la técnica de calibración mediante aproximación polinómica necesita de un modo de pre-calibrado, previo al modo de operación normal, donde se obtienen los coeficientes del polinomio con el que posteriormente se estima el valor de las resistencias del sensor. Así, se escogen un número significativo de resistencias conocidas, en este caso ocho, como valores de resistencias para R_x dentro del rango de trabajo del sensor, con valores nominales de 200 Ω , 1300 Ω , 2400 Ω , 3680 Ω , 4840 Ω , 5820 Ω , 7000 Ω y 7350 Ω con una tolerancia del 1%. Para todas ellas, se mide su valor real con un multímetro digital (Agilent 34401) con una exactitud del 0,011% en el rango tratado. La resistencia de calibración R_{c1} usada, es la misma que se emplea en la técnica de calibración de un punto (3680 Ω). En el caso del condensador, al igual que en el experimento del apartado 2.3.1, se escoge uno de 47nF.

El polinomio se consigue por ajuste por mínimos cuadrados, a partir de los promedios de los ratios de 500 muestras para cada una de las ocho resistencias con valores conocidos indicadas previamente.

Para evaluar la bondad de los resultados de nuestra propuesta de calibración, se realizan pruebas con las mismas resistencias usadas anteriormente, así como para otras distintas que se encuentran en el centro de los intervalos definidos por las resistencias usadas en el modo de configuración (762 Ω , 1890 Ω , 3070 Ω , 4100 Ω , 5300 Ω y 6400 Ω).

2.4.3. Resultados experimentales y discusión

Los polinomios obtenidos en el modo de pre-calibrado son los siguientes:

Para la aproximación a partir de una recta (CLChar8), el polinomio que se obtiene es

$$\frac{N_x}{N_{c1}} = 0,00026992 \cdot R_x + 0,00379844 \quad (2.3)$$

La Figura 2-29 muestra la línea de regresión obtenida por mínimos cuadrados. Despejando R_x en (2.3) se tiene el polinomio que aproxima los valores de R_x a partir de los ratios en el modo de operación normal.

$$R_x = 3704,79 \frac{N_x}{N_{c1}} - 14,07 \quad (2.4)$$

El coeficiente de determinación ($R^2 = 0,999999985$) es muy próximo a uno, por lo que una curva de grado uno aproxima correctamente la relación entre el ratio de cuenta y valor de resistencia.

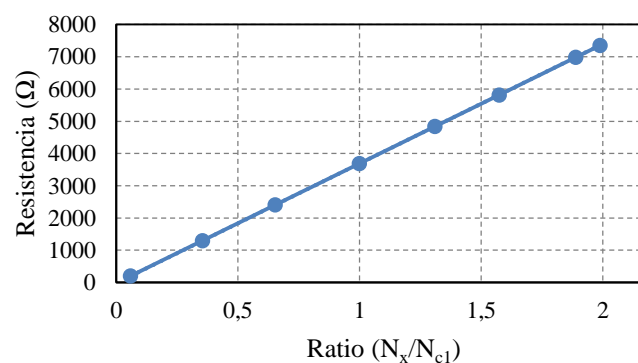


Figura 2-29. Curva de ajuste por mínimos cuadrados.

Dada la linealidad observada en los datos obtenidos, se ha considerado evaluar también la aproximación a partir de una recta obtenida con los datos de sólo dos resistencias (15% y 85% del rango) (CLChar2). La ecuación resultante es

$$R_x = 3704,81 \frac{N_x}{N_{c1}} - 14,22 \quad (2.5)$$

Para el polinomio de grado 2 la ecuación que se obtiene es

$$\frac{N_x}{N_{c1}} = 6,12442E-12 \cdot R_x^2 + 0,000269873 \cdot R_x + 0,00385339 \quad (2.6)$$

El coeficiente de determinación en este caso es 0,999999987.

Despejando R_x queda

$$R_x = -0,31 \left(\frac{N_x}{N_{c1}} \right)^2 + 3705,45 \frac{N_x}{N_{c1}} - 14,28 \quad (2.7)$$

A partir de estos polinomios de grado 1 (2.4), (2.5) y de grado 2 (2.7), y de las expresiones (1.18) y (1.19) que aproximan el valor de R_x para las técnicas de calibración de un punto y dos puntos respectivamente, se obtienen los siguientes resultados en el análisis de la exactitud de la medida del valor de resistencias.

Calibración de un punto:

La Tabla 2-3 resume los errores observados para esta técnica de calibración.

Tabla 2-3. Resumen de resultados para la calibración de un punto.

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
199,96	213,23	0,08	13,271	6,637	13,44	6,720
763,34	773,83	0,12	10,494	1,375	10,80	1,415
1297,32	1305,41	0,16	8,093	0,624	8,50	0,655
1887,55	1892,99	0,19	5,445	0,288	6,00	0,318
2401,95	2404,29	0,24	2,337	0,097	2,96	0,123
3070,25	3070,08	0,31	0,173	0,006	1,08	0,035
3684,25	3681,01	0,36	3,241	0,088	4,40	0,119
4083,85	4078,88	0,38	4,969	0,122	6,13	0,150
4836,05	4827,49	0,46	8,561	0,177	9,99	0,207
5269,05	5259,26	0,49	9,793	0,186	11,10	0,211
5813,45	5800,43	0,58	13,015	0,224	14,68	0,252
6373,15	6357,56	0,60	15,585	0,245	17,25	0,271
6983,15	6964,08	0,64	19,069	0,273	20,91	0,299
7349,15	7329,29	0,68	19,862	0,270	21,90	0,298

La Figura 2-30 muestra los máximos errores relativos para la calibración de un punto, confirmando que el error relativo mínimo se obtiene cuando $R_x \approx R_{c1}$ y aumenta conforme R_x se separa de R_{c1} . El error relativo máximo es aproximadamente igual a 6,7 %. Aunque en la zona plana es de aproximadamente el 0,3 %.

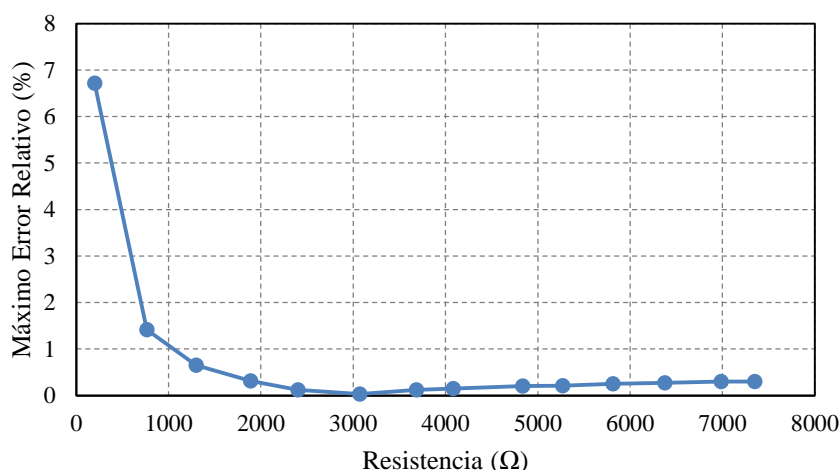


Figura 2-30. Máximo error relativo experimental para la calibración de un punto.

Calibración de dos puntos:

La Tabla 2-4 muestra los errores observados para la calibración de dos puntos, mejorando la técnica de calibración de un punto.

Tabla 2-4. Resumen de resultados para la calibración de dos puntos.

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
199,96	200,04	0,20	0,079	0,040	0,60	0,301
763,34	763,34	0,18	0,004	0,001	0,47	0,062
1297,32	1297,50	0,17	0,184	0,014	0,76	0,059
1887,55	1887,99	0,19	0,444	0,024	0,84	0,044
2401,95	2401,76	0,22	0,188	0,008	0,79	0,033
3070,25	3070,87	0,29	0,615	0,020	1,46	0,047
3684,25	3684,77	0,32	0,518	0,014	1,66	0,045
4083,85	4084,63	0,37	0,785	0,019	1,96	0,048
4836,05	4836,84	0,43	0,786	0,016	2,10	0,043
5269,05	5270,84	0,48	1,791	0,034	3,20	0,061
5813,45	5814,56	0,52	1,113	0,019	2,54	0,044
6373,15	6374,51	0,59	1,356	0,021	3,21	0,050
6983,15	6983,99	0,66	0,842	0,012	3,00	0,043
7349,15	7350,97	0,67	1,824	0,025	3,54	0,048

La Figura 2-31 muestra los errores relativos para la calibración de dos puntos, donde se observa que el error relativo máximo es aproximadamente igual a 0,3 %, consiguiendo por tanto un error relativo 23 veces menor que con la calibración de un punto. En la zona plana el error relativo máximo es del 0,06%.

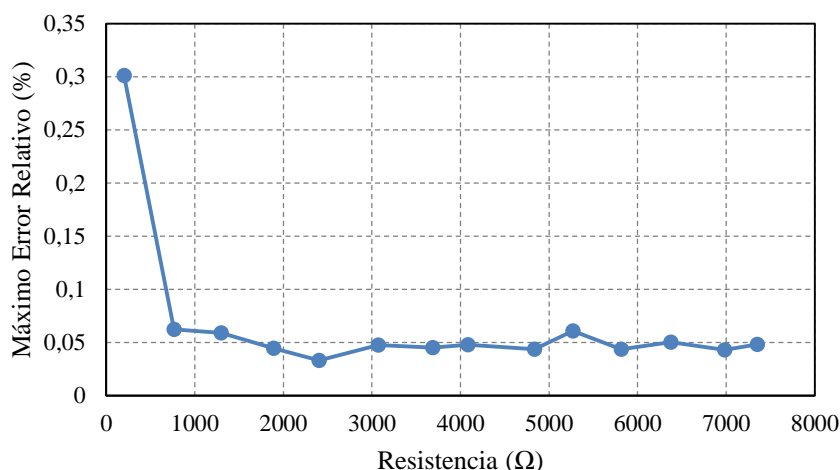


Figura 2-31. Máximo error relativo experimental para la calibración de dos puntos.

Calibración con polinomio de grado uno (CLChar8):

La Tabla 2-5 resume los errores observados para la calibración por aproximación polinómica de grado uno.

Tabla 2-5. Resumen de resultados para la calibración con el polinomio de grado uno (CLChar8).

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
199,96	200,16	0,08	0,199	0,100	0,37	0,183
763,34	763,39	0,12	0,053	0,007	0,36	0,047
1297,32	1297,47	0,16	0,146	0,011	0,55	0,043
1887,55	1887,81	0,19	0,255	0,014	0,81	0,043
2401,95	2401,50	0,24	0,453	0,019	1,20	0,050
3070,25	3070,41	0,31	0,161	0,005	1,19	0,039
3684,25	3684,21	0,36	0,040	0,001	1,21	0,033
4083,85	4083,95	0,38	0,099	0,002	1,24	0,030
4836,05	4836,07	0,46	0,020	0,000	1,41	0,029
5269,05	5269,86	0,49	0,814	0,015	2,16	0,041
5813,45	5813,58	0,58	0,131	0,002	1,86	0,032
6373,15	6373,33	0,61	0,176	0,003	2,19	0,034
6983,15	6982,69	0,65	0,462	0,007	2,31	0,033
7349,15	7349,61	0,69	0,459	0,006	2,42	0,033

La Figura 2-32 muestra los errores relativos para esta técnica de calibración. El error relativo máximo es aproximadamente igual a 0,18 %, consiguiendo un error relativo 37 veces menor que con la calibración de un punto y 1,7 veces menor que la de dos puntos. En la zona plana el error relativo máximo es del 0,04%.

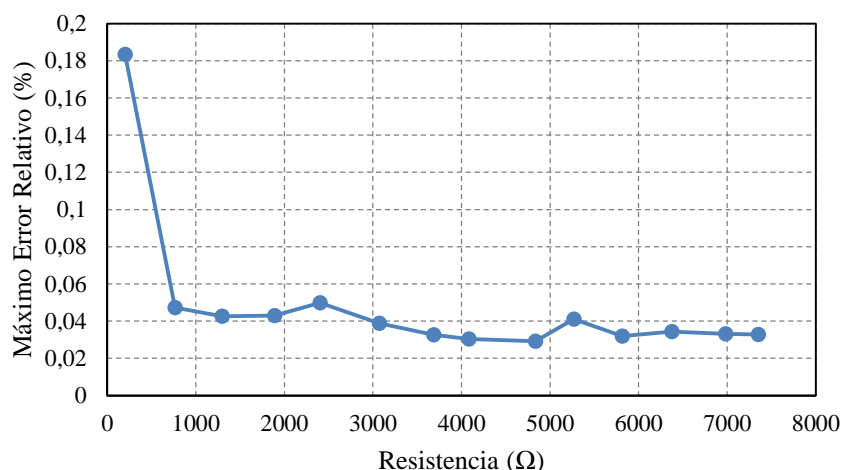


Figura 2-32. Máximo error relativo experimental para la calibración CLChar8.

Calibración con polinomio de grado uno (CLChar2):

La Tabla 2-6 resume los errores observados para la calibración por aproximación polinómica de grado uno (CLChar2).

Tabla 2-6. Resumen de resultados para la calibración con el polinomio de grado uno (CLChar2).

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
199,96	200,01	0,20	0,049	0,025	0,22	0,108
763,34	763,25	0,18	0,095	0,012	0,39	0,051
1297,32	1297,32	0,17	0,000	0,000	0,56	0,043
1887,55	1887,66	0,19	0,111	0,006	0,67	0,035
2401,95	2401,35	0,22	0,596	0,025	1,34	0,056
3070,25	3070,27	0,29	0,021	0,001	1,05	0,034
3684,25	3684,07	0,32	0,179	0,005	1,34	0,036
4083,85	4083,81	0,36	0,038	0,001	1,20	0,029
4836,05	4835,94	0,43	0,115	0,002	1,55	0,032
5269,05	5269,73	0,48	0,681	0,013	2,03	0,039
5813,45	5813,45	0,52	0,000	0,000	1,73	0,030
6373,15	6373,20	0,59	0,046	0,001	2,06	0,032
6983,15	6982,56	0,66	0,589	0,008	2,44	0,035
7349,15	7349,48	0,67	0,333	0,005	2,29	0,031

La Figura 2-33 muestra los errores relativos para esta técnica de calibración. El error relativo máximo es aproximadamente igual a 0,18 %, consiguiendo un error relativo 37 veces menor que con la calibración de un punto y 1,7 veces menor que la de dos puntos. En la zona plana el error relativo máximo es del 0,04%.

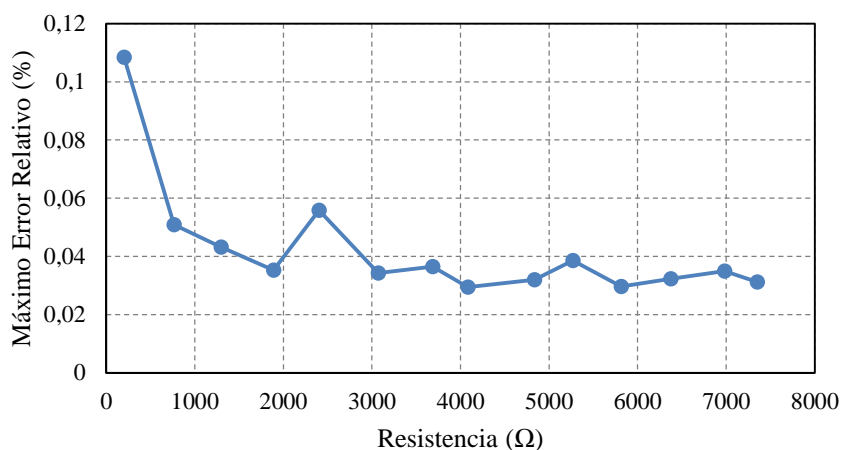


Figura 2-33. Máximo error relativo experimental para la calibración CLChar2.

Calibración con polinomio de grado dos (CSQChar8):

La Tabla 2-7 resume los errores observados para la calibración por aproximación polinómica de grado dos.

Tabla 2-7. Resumen de resultados para la calibración con el polinomio de grado dos (CSQChar8).

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
199,96	199,99	0,08	0,030	0,015	0,20	0,099
763,34	763,31	0,12	0,029	0,004	0,32	0,042
1297,32	1297,45	0,16	0,133	0,010	0,54	0,042
1887,55	1887,85	0,19	0,303	0,016	0,86	0,045
2401,95	2401,59	0,24	0,365	0,015	1,11	0,046
3070,25	3070,53	0,31	0,285	0,009	1,32	0,043
3684,25	3684,35	0,36	0,097	0,003	1,07	0,029
4083,85	4084,09	0,38	0,236	0,006	1,38	0,034
4836,05	4836,19	0,46	0,137	0,003	1,31	0,027
5269,05	5269,96	0,49	0,909	0,017	2,26	0,043
5813,45	5813,63	0,58	0,185	0,003	1,91	0,033
6373,15	6373,32	0,61	0,173	0,003	2,19	0,034
6983,15	6982,61	0,65	0,542	0,008	2,39	0,034
7349,15	7349,47	0,69	0,324	0,004	2,28	0,031

En la Figura 2-34 se representan los errores relativos para esta técnica de calibración. El error relativo máximo es aproximadamente igual a 0,098 %, 68 veces menor que el obtenido con la calibración de un punto y 3 veces menor que el de dos puntos.

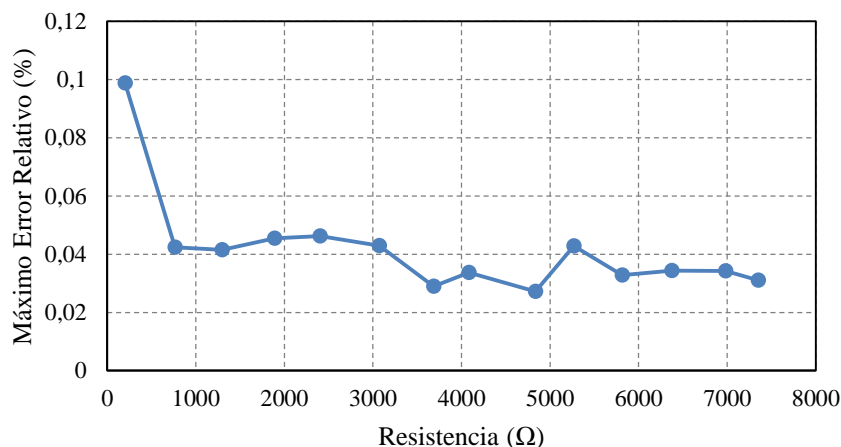


Figura 2-34. Máximo error relativo experimental para la calibración CSQChar8.

Como resumen, la Tabla 2-8 muestra el error absoluto máximo para cada resistencia y técnica de calibración, así como la suma de los errores máximos cometidos por cada una de ellas y el máximo error relativo para todo el rango de resistencias.

Tabla 2-8. Error absoluto máximo y bits de resolución de la medida en ohmios.

Resistencia (Ω)	Máximo Error Absoluto (Ω)				
	Cal. 1 P.	Cal. 2 P.	CLChar2	CLChar8	CSQChar8
199,96	13,44	0,60	0,22	0,37	0,20
763,34	10,80	0,47	0,39	0,36	0,32
1297,32	8,50	0,76	0,56	0,55	0,54
1887,55	6,00	0,84	0,67	0,81	0,86
2401,95	2,96	0,79	1,34	1,20	1,11
3070,25	1,08	1,46	1,05	1,19	1,32
3684,25	4,40	1,66	1,34	1,21	1,07
4083,85	6,13	1,96	1,20	1,24	1,38
4836,05	9,99	2,10	1,55	1,41	1,31
5269,05	11,10	3,20	2,03	2,16	2,26
5813,45	14,68	2,54	1,73	1,86	1,91
6373,15	17,25	3,21	2,06	2,19	2,19
6983,15	20,91	3,00	2,44	2,31	2,39
7349,15	21,90	3,54	2,29	2,42	2,28
Error Acumulado	149,13	26,12	18,87	19,28	19,13
Max Error Relativo (%)	6,72	0,30	0,11	0,18	0,10

Se puede ver cómo la técnica de calibración por aproximación polinómica propuesta consigue unos resultados mejores incluso que los obtenidos por la calibración de dos puntos. Los errores absolutos máximos son: 21,90 Ω para la calibración de un punto, 3,54 Ω para la de dos puntos, 2,42 Ω para la aproximación con un polinomio de grado uno y 2,28 Ω para la aproximación con un polinomio de grado dos. El error absoluto máximo acumulado para todas las resistencias medidas también sirve para comprobar, de una forma general, que los resultados de la calibración propuesta son mejores que los de las técnicas de calibración clásicas.

La Figura 2-35 y la Figura 2-36 representan gráficamente los datos de la Tabla 2-8.

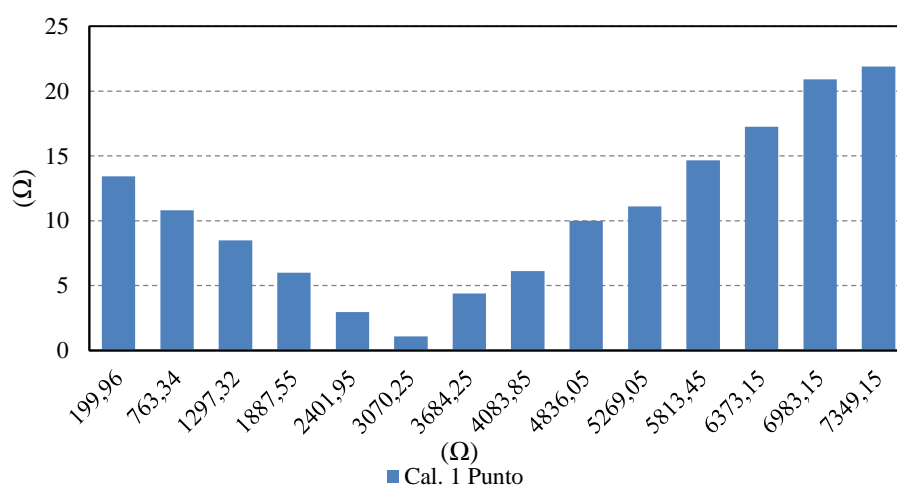


Figura 2-35. Máximo error absoluto para la calibración de un punto.

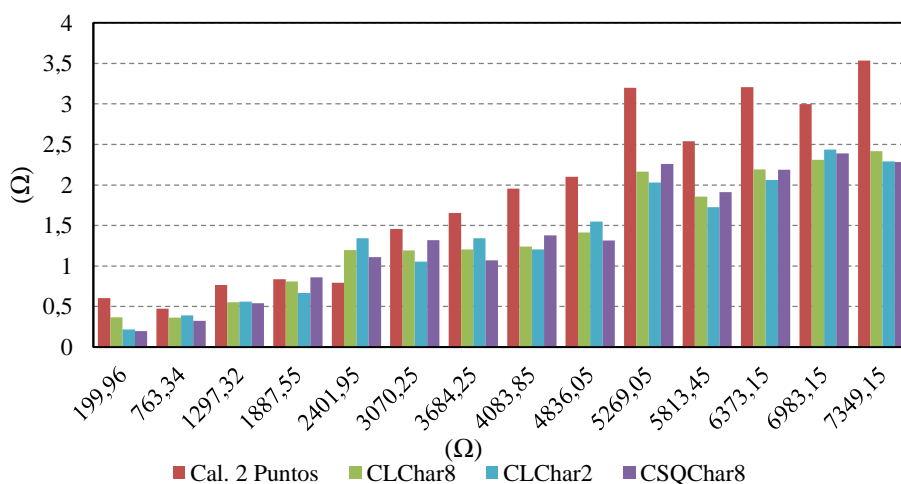


Figura 2-36. Máximos errores absolutos para la calibración de dos puntos y las técnicas de calibración propuestas.

En la Figura 2-37 se observa que el error relativo para la técnica de calibración de un punto (expresado en el eje de la derecha de la gráfica) obtiene los peores resultados,

siendo 23 veces superior al del resto de técnicas. También se distingue que, para todas las técnicas de calibración, los mayores errores relativos aparecen al aproximar el valor de las resistencias más pequeñas.

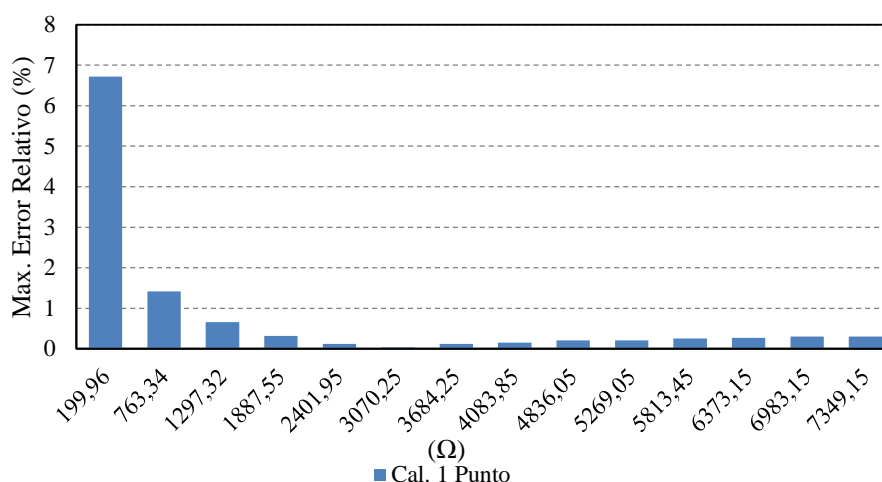


Figura 2-37. Máximo error relativo para la calibración de un punto.

Esta circunstancia no supone un problema en el caso del sensor de presión que está siendo caracterizado, ya que en ese rango de resistencias, que se corresponde con los valores altos de presión, el propio sensor posee una tolerancia mayor que la conseguida en la medida.

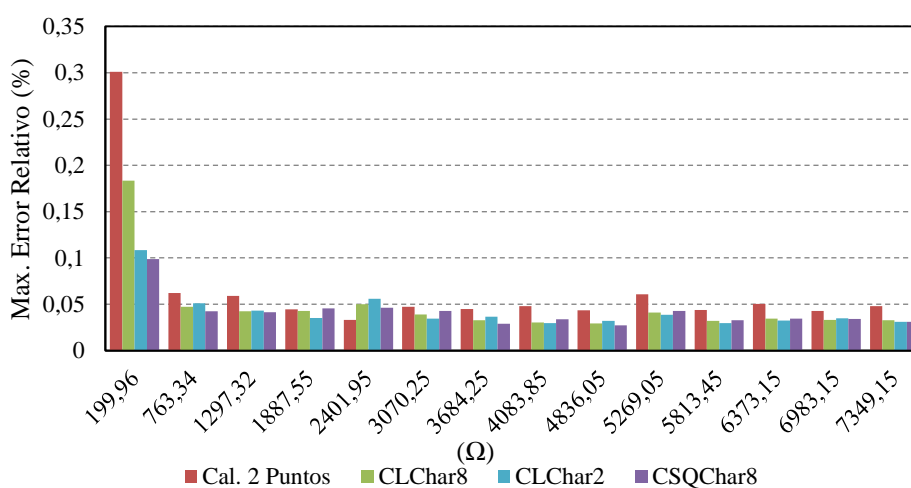


Figura 2-38. Máximos errores relativos para la calibración de dos puntos y las técnicas de calibración propuestas.

2.4.4. Conclusiones

Una nueva técnica de calibración basada en aproximación polinómica es propuesta como alternativa a las calibraciones de un punto y dos puntos. Los resultados de exactitud en la

medida de resistencias mejoran los conseguidos por las técnicas mencionadas, con la ventaja de necesitar un único elemento de referencia. Esto se traduce en una disminución de los elementos pasivos a incluir en el circuito, así como una reducción en el tiempo empleado por cada ciclo de medida, ya que la resistencia se aproxima con dos ciclos de carga y descarga (una para la resistencia del sensor y otro para la resistencia de calibración). Para conseguir el polinomio de aproximación se hace necesario un modo de pre-calibrado previo.

Por otro lado, los resultados experimentales confirman que, mediante la aproximación de un polinomio de grado uno, se consigue una exactitud en la medida del mismo orden que la que resultante del uso de un polinomio de grado dos. Dado que las operaciones aritméticas que nos llevan a la aproximación de R_x son más sencillas en este caso, la aproximación mediante una recta parece la mejor solución. Nótese que la ecuación de aproximación tiene la misma forma que la utilizada en la calibración de un punto, siendo mucho más simple que la empleada en la calibración de dos puntos.

2.5. Comparativa con otros circuitos de conexión directa

Para poder comparar el circuito aquí propuesto (Figura 2-28) con otras implementaciones basadas en una conexión directa entre un dispositivo digital y un sensor resistivo, se han repetido los experimentos realizados anteriormente para el rango de resistencias del sensor de temperatura PT-1000.

2.5.1. Materiales y método

Para emular el comportamiento de este sensor para temperaturas desde -45 °C a $+310\text{ °C}$, se seleccionan 12 resistencias R_x desde $760\ \Omega$ a $2200\ \Omega$ ($760\ \Omega$, $880\ \Omega$, $950\ \Omega$, $1000\ \Omega$, $1100\ \Omega$, $1200\ \Omega$, $1300\ \Omega$, $1400\ \Omega$, $1600\ \Omega$, $1800\ \Omega$, $1900\ \Omega$ y $2200\ \Omega$). Para la calibración de un punto se selecciona $R_{c1} = 1400\ \Omega$. En el caso de las resistencias para la calibración de dos puntos se toman los valores $R_{c2} = 950\ \Omega$ y $R_{c3} = 1900\ \Omega$, encontrándose ambas en el 15% y en el 85% del rango, respectivamente. En el caso del condensador, se escoge uno con una capacidad $C = 170\text{nF}$, con la que se consigue una constante de tiempo próxima a la obtenida en el caso del sensor táctil, asegurando con ello una incertidumbre en la medida de tiempo muy similar.

Para evaluar la exactitud en la estimación de R_x de las distintas técnicas de calibración, se comparan las calibraciones de un punto, dos puntos y CLChar2. Para esta última, el polinomio se consigue por ajuste por mínimos cuadrados, a partir de los promedios de los ratios de 500 muestras para cada una de las dos resistencias con valores en el 15% y el 85% del rango.

2.5.2. Resultados experimentales

La Tabla 2-9 muestra los resultados del análisis de la incertidumbre en la medida de tiempo del circuito propuesto haciendo uso del módulo de captura MC6. Los bits efectivos para la conversión tiempo-digital toman un valor muy cercano al conseguido en el rango de resistencias del sensor táctil (ENOB = 11,48 bits), dando lugar, en este caso, a una resolución en la medida de resistencias de 0,50 Ω .

Tabla 2-9. Análisis de la incertidumbre y la resolución para un sensor PT-1000 ($R_x = 2200 \Omega$).

	$\sigma(X)$ (cuentas)	$\sigma(X)$ (ns)	$u(y)$ (ns)	ENOB (bits)	Resolución (Ω)
MC6	1,03	20,6	21,4	11,48	0,50

En cuanto al análisis de la exactitud en la aproximación de R_x , lo primero es obtener el polinomio de aproximación. En este caso, se va a evaluar sólo a partir de la técnica CLChar2 ya que es la que presenta un coste menor a la hora de su evaluación. El polinomio obtenido es

$$\frac{N_x}{N_{cl}} = 0,000705122 \cdot R_x + 0,009706205 \quad (2.8)$$

Despejando R_x en (2.8) se tiene el polinomio que aproxima los valores de R_x a partir de los ratios en el modo de operación normal.

$$R_x = 1418,19 \frac{N_x}{N_{cl}} - 13,77 \quad (2.9)$$

A continuación se presentan los resultados obtenidos para cada una de las técnicas de calibración empleadas.

Calibración de un punto:

La Tabla 2-10 resume los errores observados para esta técnica de calibración.

Tabla 2-10. Análisis de la exactitud y la precisión para la calibración de un punto para el sensor PT-1000.

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
759,75	763,68	0,08	3,93	0,52	4,13	0,54
876,75	879,20	0,09	2,45	0,28	2,69	0,31
948,55	950,03	0,11	1,48	0,16	1,79	0,19
1016,35	1016,91	0,11	0,56	0,06	0,87	0,09
1106,65	1106,08	0,10	0,57	0,05	0,87	0,08
1196,15	1194,40	0,11	1,75	0,15	2,09	0,18
1296,25	1293,39	0,13	2,86	0,22	3,47	0,27
1398,75	1394,40	0,13	4,35	0,31	4,78	0,34
1598,25	1591,26	0,15	6,99	0,44	7,42	0,46
1798,95	1789,47	0,15	9,48	0,53	10,02	0,56
1891,95	1881,39	0,16	10,56	0,56	11,15	0,59
2193,85	2179,43	0,19	14,42	0,66	15,11	0,69

La Figura 2-39 muestra los máximos errores relativos para la calibración de un punto, confirmando que el error relativo mínimo se obtiene cuando $R_x \approx R_{c1}$ y aumenta conforme R_x se separa de R_{c1} .

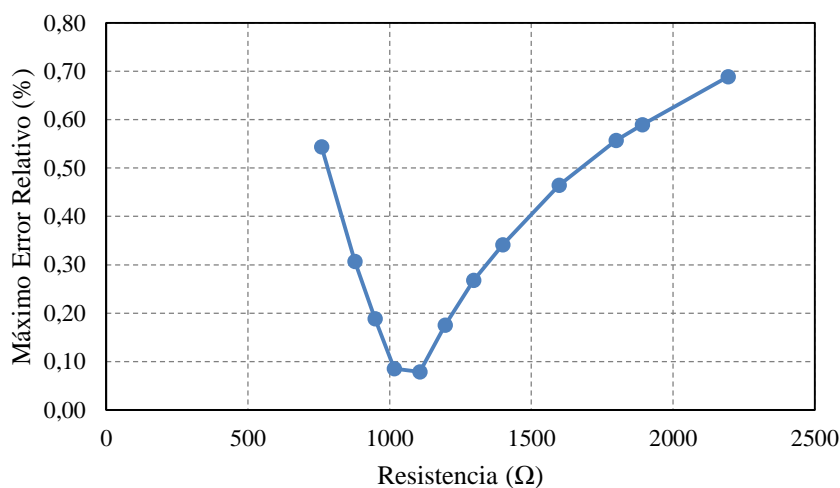


Figura 2-39. Máximo error relativo experimental para la calibración de un punto.

Calibración de dos puntos:

La Tabla 2-11 resume los errores observados para esta técnica de calibración.

Tabla 2-11. Análisis de la exactitud y la precisión para la calibración de dos puntos para el sensor PT-1000.

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
759,75	760,05	0,10	0,30	0,04	0,59	0,08
876,75	877,47	0,09	0,72	0,08	1,04	0,12
948,55	949,47	0,10	0,92	0,10	1,17	0,12
1016,35	1017,47	0,10	1,12	0,11	1,39	0,14
1106,65	1108,11	0,09	1,46	0,13	1,78	0,16
1196,15	1197,89	0,10	1,74	0,15	2,03	0,17
1296,25	1298,51	0,11	2,26	0,17	2,64	0,20
1398,75	1401,20	0,13	2,45	0,18	2,79	0,20
1598,25	1601,34	0,13	3,09	0,19	3,50	0,22
1798,95	1802,83	0,17	3,88	0,22	4,38	0,24
1891,95	1896,24	0,16	4,29	0,23	4,73	0,25
2193,85	2199,24	0,20	5,39	0,25	5,99	0,27

La Figura 2-40 muestra los errores relativos para la calibración de dos puntos, donde se observa que el error relativo máximo es igual a 0,27 %, consiguiendo por tanto un error relativo 2,5 veces menor que con la calibración de un punto.

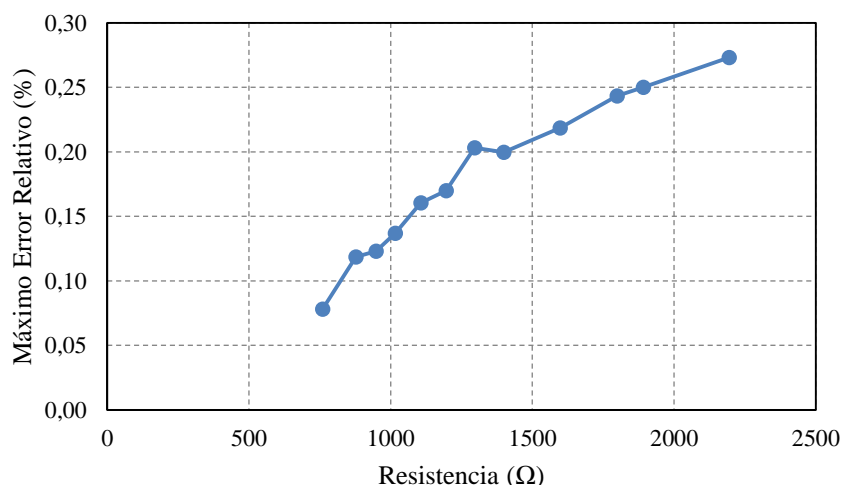


Figura 2-40. Máximo error relativo experimental para la calibración de dos puntos.

Calibración con polinomio de grado uno (CLChar2):

La Tabla 2-12 resume los errores observados para esta técnica de calibración.

Tabla 2-12. Análisis de la exactitud y la precisión para la calibración con el polinomio de grado uno (CLChar2) para el sensor PT-1000.

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
759,75	759,79	0,08	0,04	0,005	0,24	0,032
876,75	876,81	0,09	0,06	0,006	0,30	0,034
948,55	948,55	0,12	0,00	0,000	0,31	0,033
1016,35	1016,30	0,11	0,05	0,005	0,34	0,034
1106,65	1106,62	0,10	0,03	0,003	0,34	0,031
1196,15	1196,07	0,11	0,08	0,006	0,42	0,035
1296,25	1296,35	0,13	0,10	0,008	0,52	0,040
1398,75	1398,67	0,14	0,08	0,006	0,52	0,037
1598,25	1598,07	0,15	0,18	0,011	0,62	0,039
1798,95	1798,84	0,15	0,11	0,006	0,65	0,036
1891,95	1891,95	0,16	0,00	0,000	0,60	0,032
2193,85	2193,84	0,19	0,01	0,000	0,71	0,032

La Figura 2-41 muestra los errores relativos para la calibración CLChar2, donde se observa que el error relativo máximo es igual a 0,04 %, consiguiendo por tanto un error relativo 17 veces menor que con la calibración de un punto y 6,75 veces menor que el de la calibración en dos puntos.

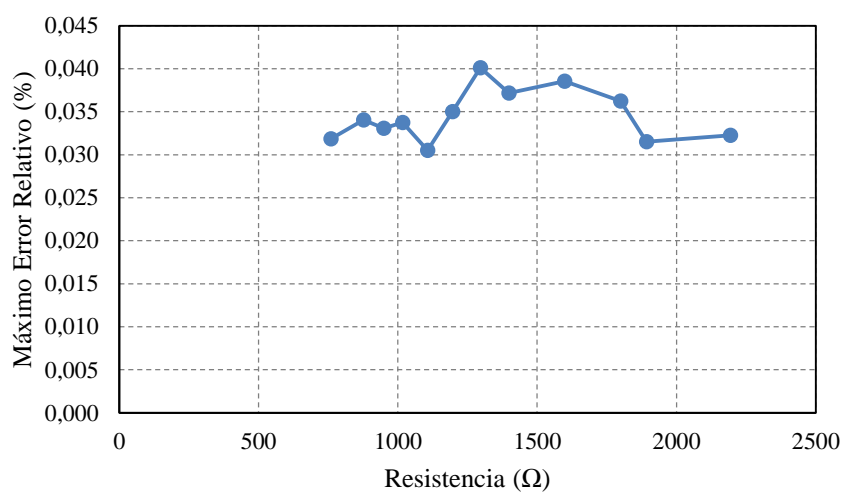


Figura 2-41. Máximo error relativo experimental para la calibración con el polinomio de grado uno (CLChar2).

2.5.3. Discusión

Como resumen, la Tabla 2-13 muestra el error absoluto máximo para cada resistencia y técnica de calibración, así como la suma de los errores máximos cometidos por cada una de ellas y el máximo error relativo para todo el rango de resistencias.

Tabla 2-13. Error absoluto máximo para las tres calibraciones.

Resistencia (Ω)	Máximo Error Absoluto (Ω)		
	Cal. 1 P.	Cal. 2 P.	CLChar2
759,75	4,13	0,59	0,24
876,75	2,69	1,04	0,30
948,55	1,79	1,17	0,31
1016,35	0,87	1,39	0,34
1106,65	0,87	1,78	0,34
1196,15	2,09	2,03	0,42
1296,25	3,47	2,64	0,52
1398,75	4,78	2,79	0,52
1598,25	7,42	3,50	0,62
1798,95	10,02	4,38	0,65
1891,95	11,15	4,73	0,60
2193,85	15,11	5,99	0,71
Error Acumulado	64,40	32,03	5,57
Max Error Relativo (%)	0,69	0,27	0,04

Se puede ver cómo la técnica de calibración por aproximación polinómica propuesta consigue unos resultados mejores incluso que los obtenidos por la calibración de dos puntos. Los errores absolutos máximos son: 15,11 Ω para la calibración de un punto, 5,99 Ω para la de dos puntos y 0,71 Ω para la aproximación con un polinomio de grado uno. El error absoluto máximo acumulado para todas las resistencias medidas también sirve para comprobar, de una forma general, que los resultados de la calibración propuesta son mejores que los de las técnicas de calibración clásicas.

Finalmente, la Tabla 2-14 muestra una comparativa entre los resultados obtenidos por el circuito propuesto en este trabajo, con el módulo de captura MC6 y la técnica de calibración CLChar2, con otros circuitos propuestos en la literatura que hacen uso de una conexión directa entre el sensor PT-1000 y un dispositivo digital (microcontroladores, CPLD y FPGA de otro fabricante) [Reverter, Jordana, et al. 2005; Bengtsson 2012].

Tabla 2-14. Comparativa entre el circuito propuesto y otras implementaciones para el rango de resistencias del sensor de temperatura PT-1000 (* peor caso, ** promediando 500 medidas).

Dispositivo	Captura - Calibración	Rango (Ω)	V _{DD}	Const. RC	Max. Incert. Rel.	Max. Error Abs. (Ω)	Max. Error Rel. (%)
PIC16F87*	Evento Único Cal. 2 Puntos	825 - 1470	5 V	3,23 ms	0,005	0,30	0,02
PIC18F458**	Evento Único Cal. 1 Punto	820 - 2200	5 V	21,94 ms	0,121	8,89	0,73
CPLD EMP3064A**	Evento Único Cal. 1 Punto	820 - 2200	3,3 V	21,94 ms	0,497	3,40	0,21
Cyclone II EP2C20**	Evento Único Cal. 1 Punto	820 - 2200	3,3V	21,94 ms	0,481	26,27	2,57
Spartan 3AN*	MC6 Cal. CLChar2	760 - 2200	3,3 V	0,37 ms	0,010	0,71	0,04

Hay que tener en cuenta que las condiciones no son las mismas para todas las pruebas presentadas en la Tabla 2-14. Así, una mayor tensión de alimentación proporciona una mayor velocidad de cambio en el punto de disparo (*slew rate*) disminuyendo la incertidumbre debida al disparo e incrementando la precisión [Bengtsson 2012]. Por otro lado, para constantes de tiempos altas [Reverter et al. 2004] el ruido de cuantificación se reduce, aunque esto incrementa el tiempo de medida, afectando por tanto a la razón de muestreo. En el caso de la implementación basada en el microcontrolador PIC16F87 [Reverter, Jordana, et al. 2005], los resultados son fruto de una calibración de dos puntos, lo que implica la medida de tres resistencias por cada ciclo de muestreo. Por último, los resultados presentados a partir del circuito aquí propuesto y por el PIC16F87 en [Reverter, Jordana, et al. 2005] son obtenidos a partir del peor caso de entre 500 medidas, mientras que los presentados en [Bengtsson 2012] (PIC18F458, CPLD EMP3064A y FPGA Cyclone II) son los obtenidos a partir del promedio de 500 muestras.

2.6. Conclusiones

En este capítulo se presenta un circuito donde se lleva a cabo una conexión directa entre un sensor y una FPGA. Esto es especialmente interesante en sistemas complejos donde se recogen datos analógicos a partir de una gran cantidad de sensores. Los circuitos propuestos pueden ser empleados tanto en sensores resistivos como capacitivos aunque,

en este caso, dos sensores resistivos han sido estudiados. Tanto la adquisición de la información de los sensores como su posterior procesado pueden ser realizados en paralelo debido a las características propias de las FPGAs.

También se han propuesto varios módulos de captura con el propósito de incrementar la precisión de las medidas. El objetivo ha sido el de emular la histéresis para las tensiones umbrales que algunos microcontroladores pueden aplicar en sus puertos de entrada al disponer de *buffers* ST, mediante el empleo de módulos digitales inteligentes. La flexibilidad de los elementos de almacenamiento de una FPGA ha permitido trabajar con *flip-flops* disparados por flancos (de subida y de bajada) y por nivel, permitiendo desarrollar estrategias que promedian las cuentas almacenadas por dos o cuatro módulos que trabajan en paralelo. La idea es filtrar parte del ruido de disparo sin que el ancho de banda se vea afectado. Para aplicaciones donde una gran precisión es requerida, el módulo MC6 consigue más de 12 bits efectivos en la conversión tiempo-digital en el rango del sensor táctil estudiado, con un tiempo de medida de 348 μ s (teniendo en cuenta el tiempo de carga y de descarga del condensador). Otros módulos, que emplean un menor número de recursos, pueden emplearse en aplicaciones donde no se requieran medidas de gran precisión.

Además de los módulos de captura, una técnica de calibración para la estimación de los valores resistivos de los sensores ha sido propuesta. Esta técnica hace uso de los mismos elementos externos que la técnica de calibración de un punto, que está basada en la medida de una resistencia de calibración conocida. De hecho, alcanza un rendimiento incluso superior al que se consigue con una calibración de dos puntos, mejorando el ancho de banda al necesitar la medida de un único elemento de referencia por cada medida. El inconveniente es que se necesita realizar una caracterización de los puertos de entrada de la FPGA como paso previo a la aplicación de la técnica de calibración. Sin embargo, no es necesario utilizar ningún circuito adicional para caracterizar los puertos, ya que la propia técnica es usada para ese cometido empleando para ello ciertos valores de resistencias conocidas dentro del rango del sensor. Una extensión de esta técnica, donde se implementa una aproximación no lineal, ha sido evaluada para contemplar posibles variaciones de las impedancias de los puertos de entrada dependiendo de las resistencias a ser medidas, aunque no se han observado mejoras significativas para los rangos estudiados. La técnica de calibración propuesta consigue un máximo error absoluto de 2,42 Ω en el rango del sensor táctil piezorresistivo (entre 200 Ω y 7350 Ω).

Por último, se ha realizado una comparación entre los resultados obtenidos por el circuito propuesto y otras implementaciones presentadas por otros autores, para un rango de resistencias correspondiente a un sensor de temperatura PT-1000 (entre 760 Ω y 2200 Ω), consiguiendo un máximo error absoluto de 0,71 Ω .

Capítulo 3. Circuitos de conexión directa matriz sensores-FPGA

3.1. Introducción

Existen un gran número de aplicaciones donde se utilizan matrices de sensores resistivos para obtener información de distinta naturaleza como: sensores de temperatura [Wu *et al.* 2011; Yang *et al.* 2010], detección de gases [Wang *et al.* 2004; Depari *et al.* 2007], sensores táctiles [Tise 1988; Speeter 1990; Shimojo *et al.* 2004; Castellanos-Ramos *et al.* 2010; Kane *et al.* 2000; Dahiya *et al.* 2009; Drimus *et al.* 2014; Xuefeng Zhang *et al.* 2012; Kim *et al.* 2012], etc. La complejidad del sistema electrónico necesario para leer la información de la matriz depende del número de sensores que la componen, del número de conexiones necesarias para extraer la información, de los valores de resistencia de cada sensor, y de la velocidad necesaria para su obtención.

En cuanto a la velocidad de adquisición de los datos de la matriz, existe un compromiso entre ésta y la complejidad del sistema. Así, si se quiere una máxima velocidad de procesado, se precisa acceder de forma paralela a la información de todos los sensores, lo que conlleva un elevado número de conexiones para acceder a esta información, así como un gran número unidades de procesamiento en los circuitos que han de recibir dicha información. De hecho, para circuitos con un máximo paralelismo, si se distribuyen los sensores en una matriz de 2 dimensiones de M filas y N columnas, el número de cables puede alcanzar $2 \cdot M \cdot N$. Normalmente, uno de los terminales de los sensores es compartido por lo que el número final puede reducirse a $M \cdot N + 1$ [Göger *et al.* 2007]. Cada uno de estos cables debe dirigirse a un circuito para obtener la información de un sensor concreto en el que el valor de la resistencia se traduzca en un valor de tensión y posteriormente en

un número digital, por lo que se necesitarían $M \cdot N$ de estos circuitos. El tiempo para procesar la matriz coincide, aproximadamente, con el tiempo en medir un único elemento.

Por otro lado, si la velocidad con la que la información es medida no es un requisito crítico del sistema, se puede reducir el número de conexiones y la complejidad del circuito optando por una organización donde cada sensor esté conectado a una fila y una columna diferentes. Con esta configuración, los sensores comparten conexiones pudiéndose acceder a la información de un único sensor seleccionando la fila y columna adecuada. Como consecuencia de esto, el número de cables se reduce a $M + N$ y el número de circuitos necesarios para escanear la información es de M o N , dependiendo del tipo de lectura de la matriz. Obviamente se necesitan circuitos adicionales para llevar a cabo el direccionamiento de cada elemento de la matriz, y el tiempo necesario para extraer la información es el resultado de multiplicar el tiempo en medir un sensor por el número de columnas N o filas M más el tiempo empleado en la selección.

Esta misma solución es la que hay que emplear si la propia construcción del sensor posee una organización de filas y columnas y la salida del mismo se proporciona por $M + N$ cables (Figura 3-1) [Drimus et al. 2014].

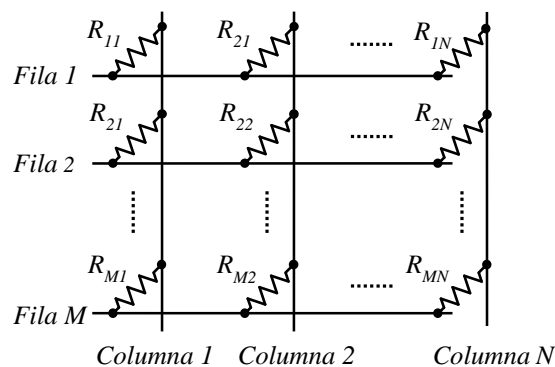


Figura 3-1. Matriz de sensores resistivos con conexiones compartidas para filas y columnas.

Una tercera opción para acceder a la información de la matriz con una mínima circuitería consiste en multiplexar en el tiempo la información que proviene de los $M + N$ cables, de forma que sólo se muestree un único sensor a la vez. Una versión mejorada de este tipo de circuitos es la llamada *improved isolated drive feedback circuit* (IIDFC), propuesta en [Wu et al. 2015]. Esta es la solución más lenta ya que se multiplica por $M \cdot N$ el tiempo de escaneo de un único sensor para obtener la información de toda la matriz, a lo que hay que sumar el tiempo necesario para la multiplexación.

Así, dependiendo del tipo de sustrato sobre el que estén realizados los circuitos y los sensores, del tamaño y arquitectura de la matriz, del tamaño y consumo del circuito del sensor y del tiempo requerido para el escaneo de la matriz, podrá ser empleada una solución u otra. También pueden existir soluciones intermedias entre las tres descritas anteriormente.

En este capítulo se presentan varias propuestas de conexión directa entre una matriz de sensores y una FPGA. La adopción de cada una de ellas depende, fundamentalmente, de cómo son direccionados cada uno de los sensores individuales. Como se indica en el capítulo 2, en las FPGAs se pueden implementar tantos módulos de conversión tiempo-digital como recursos hardware dispongan. Esta característica las hace especialmente útiles para trabajar sobre sensores táctiles piezorresistivos, donde cada sensor de la matriz modula la presión realizada sobre él mediante variaciones del valor de una resistencia.

Los datos obtenidos a partir de matrices de sensores táctiles son tratados de forma similar a como lo son las imágenes provenientes de sensores de imagen. Cada punto de una matriz de sensores táctiles (táctel) es equivalente a un punto de una imagen (pixel). Por lo tanto, las imágenes táctiles estarán formadas por tácteles organizados en filas y columnas.

Estas matrices táctiles contienen gran cantidad de tácteles, por lo que medir de forma simultánea todos ellos tampoco es la opción más recomendable. Una solución híbrida entre las medidas paralelas que se pueden realizar en una FPGA y las medidas secuenciales propuestas con los microcontroladores se adivina como la opción más recomendable para este tipo de sensores.

En este trabajo, las medidas de las imágenes táctiles se realizan de forma paralela para todos los sensores de una fila y de forma secuencial para las diferentes filas. Las soluciones aportadas dependen del tamaño y organización de las matrices.

3.2. Matrices de pequeño o mediano tamaño

Para matrices de pequeño o mediano tamaño, donde la organización del conexionado no venga impuesto por la propia matriz de electrodos, se puede optar por una conexión donde todos los tácteles de una fila son medidos de forma paralela, empleándose para el direccionamiento una conexión por cada uno de los sensores que conforman la matriz. El gran número de puertos de entrada/salida de los que disponen las FPGAs permiten esta

configuración, propiciando un direccionamiento muy sencillo, pues basta con mantener en estado de alta impedancia todas las conexiones con los tácteles, salvo con las de los sensores de la fila que se pretende medir. Con esto se evitan los efectos de *crosstalk* entre sensores al no compartirse conexiones entre tácteles. La Figura 3-2 muestra el esquema de la conexión propuesto en este trabajo. También existe la posibilidad de incluir dos filas de resistencias de calibración como se hizo en el capítulo anterior.

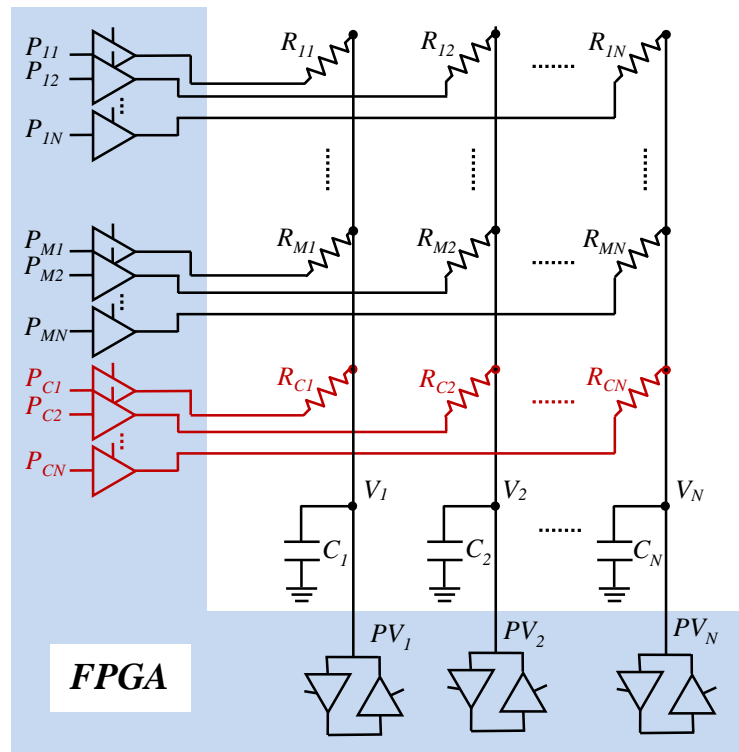


Figura 3-2. Circuito de interfaz directa – Integrador Pasivo.

El circuito integrador pasivo se conecta con el sensor modulador resistivo de igual forma a como se hace en un microcontrolador. En este caso, no se mide un único sensor junto con resistencias de calibración por cada puerto de medida sino que se obtiene la conversión tiempo-digital de todos los sensores de una misma fila. Para ello, se necesitan tantos módulos de captura como columnas posea la imagen táctil.

Como se observa, no existe ningún circuito físico de multiplexación exterior a la FPGA. El número de pines necesarios para direccionar la matriz de sensores es $(M+1) \cdot N$. Cada resistencia de la matriz, R_{ij} , está conectada a un pin de la FPGA, P_{ij} que puede estar configurado como salida (proporcionando tensiones cercanas a 0 V o a V_{DD}) o en alta impedancia (HZ). El cable común a todos los sensores de una columna está conectado a

un condensador, C_j y a un pin de la FPGA que puede configurarse también como entrada/salida, PV_j . Además, cada columna tiene una resistencia extra, R_{cj} (no un sensor) conectada al pin P_{cj} , utilizada para fines de calibración. El máximo tamaño de la matriz de sensores a direccionar vendrá determinado por el número de pines que posea la FPGA, aunque esto no es actualmente una limitación severa, ya que se pueden encontrar FPGAs con rangos de centenares de pines para poder ser utilizados [Xilinx, Inc. 2014].

El proceso de lectura de cada fila de la matriz consta de dos fases. Primero, durante la fase de carga, los condensadores C_j son cargados configurando los pines PV_j como salida con un nivel lógico '1', mientras todos los P_{ij} se mantienen en HZ. Seguidamente, en la fase de descarga, una fila es seleccionada fijando de sus correspondientes pines P_{kj} como salida con un valor lógico '0', es decir, los pines P_{k1}, \dots, P_{kN} son puestos a '0' mientras el resto de pines P_{ij} ($i \neq k$) se mantienen en HZ. Simultáneamente, todos los pines PV_j son configurados como entrada para poder medir la tensión V_j del condensador C_j que está siendo descargado. Un conjunto de N contadores son iniciados coincidiendo con el comienzo de la fase de descarga y sus cuentas son paradas cuando la tensión de detección de '0' lógico, VT_j , es alcanzada en su correspondiente columna. Así, una fila completa es leída en paralelo por la FPGA. El proceso se repite para las M filas de la matriz, más la fila de calibración.

Como se explica en el capítulo 2, el propósito de la resistencia de calibración es evitar tener que conocer los valores de C_j , VT_j y V_{DD} , y compensar las variaciones que puedan presentar en el tiempo debido a cambios en la tensión de alimentación y la temperatura. Con una fila de resistencias de calibración, el método usado para la aproximación del valor de cada R_{ij} , es el que en la literatura se conoce como calibración de un punto [Reverter, Jordana, et al. 2005]. Según este método, si el tiempo de descarga que se obtiene para el condensador C_j , cuando se selecciona la fila i , es Δt_{ij} , y cuando se selecciona la fila de calibración es Δt_{cj} , entonces el valor de R_{ij} viene dado por:

$$R_{ij} = \frac{\Delta t_{ij}}{\Delta t_{cj}} R_{cj} \quad (3.1)$$

donde R_{cj} es un valor conocido que coincide con el 50% del rango que el sensor pueda presentar. Sin embargo, este método de calibración no tiene en cuenta la resistencia de salida de los *buffers* de la FPGA, RB_j , que están unidas en serie a R_{ij} y R_{cj} por lo que

utilizando (3.1) se comete un error en la estimación del valor de la resistencia. Una mayor exactitud se puede obtener si se emplea el método de calibración de dos puntos [Reverter, Jordana, et al. 2005]. Sin embargo, esto origina la necesidad de añadir una fila de calibración adicional, con el consiguiente incremento del hardware del circuito.

En el apartado 2.4.1, se propone una solución de calibrado, que considera las resistencias RB_i , usando una única resistencia de calibración [Oballe-Peinado et al. 2015].

3.2.1. Crosstalk RC

En la Figura 3-3, se muestra el detalle de la columna j del circuito cuando se está midiendo la descarga de C_j a través de la resistencia R_{ij} . Sean Cp_{kj} con $k \neq i$ los condensadores que agrupan todas las capacidades parásitas conectadas a los pines P_{ij} en estado HZ de la FPGA. En la Figura 3-3 se asume que la descarga del condensador Cp_{ij} es muy rápida comparada con la de C_j , ya que, $RB_i \ll R_{ij}$ y $Cp_{ij} \ll C_j$. Analizando la figura, se observa que todas las capacidades parásitas influyen en el proceso de descarga de C_j . Además, esta influencia no siempre es la misma ya que las resistencias que los unen a C_j también varían.

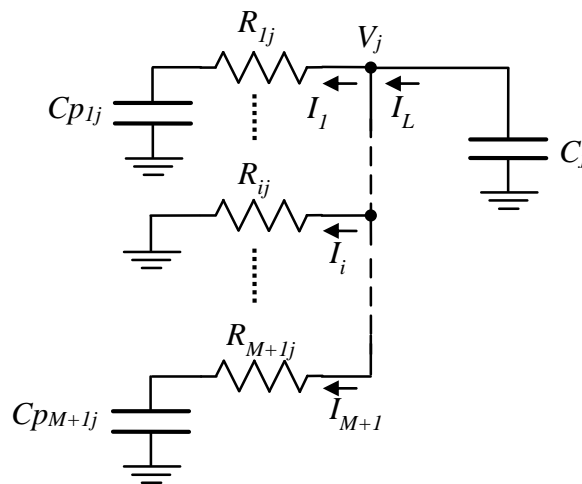


Figura 3-3. Detalle del circuito de la columna j mostrando las capacidades parásitas.

Sea $V_j(s)$ la tensión de C_j en la variable de Laplace durante el proceso de descarga. Su valor viene dado por

$$V_j(s) = \frac{-I_L}{C_j s} + \frac{V_{DD}}{s} \quad (3.2)$$

donde I_L es la intensidad que descarga al condensador y V_{DD} es la tensión inicial de carga de éste. Por otro lado para cada I_k de la Figura 3-3 con $k \neq i$

$$V_j(s) = I_k \left(R_{kj} + \frac{1}{Cp_{kj}} \right) + \frac{V_{DD}}{s} \quad \forall k \in \{1, 2, \dots, n\}, k \neq i \quad (3.3)$$

mientras que

$$I_i = \frac{V_j(s)}{R_{ij}} \quad (3.4)$$

Despejando I_L de (3.2), I_k de (3.3) y teniendo en cuenta que

$$I_L = \sum_{k=1}^{M+1} I_k \quad (3.5)$$

donde $k = M+1$ indica la fila de calibración, se puede escribir

$$-V_j(s)C_j s + C_j V_{DD} = \frac{V_j(s)}{R_{ij}} + \sum_{\substack{k=1 \\ k \neq i}}^{M+1} \frac{1}{R_{kj}} \frac{sV_j(s) - V_{DD}}{s + \frac{1}{R_{kj} Cp_{kj}}} \quad (3.6)$$

despejando $V(s)$

$$V_j(s) = V_{DD} \cdot \frac{C_j + \sum_{\substack{k=1 \\ k \neq i}}^{M+1} \frac{1}{R_{kj}} \frac{1}{s + \frac{1}{R_{kj} Cp_{kj}}}}{C_j s + \frac{1}{R_{ij}} + \sum_{\substack{k=1 \\ k \neq i}}^{M+1} \frac{1}{R_{kj}} \frac{1}{s + \frac{1}{R_{kj} Cp_{kj}}}} \quad (3.7)$$

y finalmente

$$V_j(s) = V_{DD} \cdot \frac{1 + \sum_{\substack{k=1 \\ k \neq i}}^{M+1} \frac{Cp_{kj}}{C_j} \frac{1}{R_{kj} Cp_{kj} s + 1}}{s \left(1 + \sum_{\substack{k=1 \\ k \neq i}}^{M+1} \frac{Cp_{kj}}{C_j} \frac{1}{R_{kj} Cp_{kj} s + 1} \right) + \frac{1}{R_{ij} C_j}} \quad (3.8)$$

A partir de la expresión (3.8) se observa cómo $V_j(s)$ ya no es simplemente la tensión de descarga de un condensador con un único polo, sino que tiene $M + 1$ polos y M ceros. La ecuación (3.8) muestra también la aparición de *crosstalk*, ya que $V_j(s)$ ya no es función únicamente de R_{ij} y C_j , sino que depende del resto de resistencias de la columna, R_{kj} , y de las capacidades parásitas asociadas a cada pin de la columna, Cp_{kj} (de ahí el nombre de *crosstalk RC*).

Si $Cp_{kj} \ll C_j$, (3.8) queda reducido a

$$V_j(s) \approx \frac{V_{DD}}{s + \frac{1}{R_{ij}C_j}} \quad (3.9)$$

que es la ecuación de descarga a través de un único condensador. Por lo tanto, cuanto mayor sea C_j comparado con Cp_{kj} menor será el efecto de *crosstalk*. Reducir Cp_{kj} significa realizar un buen diseño del sensor y del circuito de control, sin embargo, hay términos que componen Cp_{kj} que no pueden ser disminuidos, como la capacidad del pin de la FPGA o las de los contactos de éstos con el cableado de los sensores, una vez seleccionada la tecnología PCB del circuito. También en (3.8) se observa la dependencia con R_{kj} . Pero sobre este término no se puede actuar al tratarse de la resistencia de los propios sensores.

Aunque el *crosstalk RC* se ha analizado en una matriz de sensores, también estará presente si se utiliza cualquiera de las técnicas de calibración indicadas en el capítulo 2, aunque se trate de calibrar un único sensor. En la Figura 3-4 se muestra un circuito para la medida de R_X con 2 resistencias de calibración R_{C1} y R_{C2} . Como ilustra la figura, las resistencias R_{C1} como R_{C2} introducen 2 condensadores parásitos Cp_{C1} y Cp_{C2} .

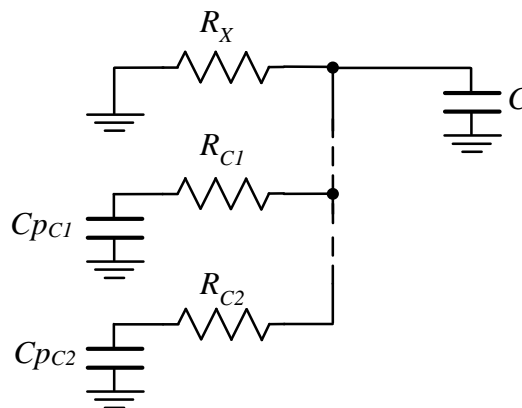


Figura 3-4. Circuito de calibración de dos puntos mostrando las capacidades parásitas.

3.2.2. Análisis de la incertidumbre en la medida

La ecuación (3.8) también nos ayuda a encontrar el tiempo máximo, $\Delta t_{\max ij}$, y mínimo, $\Delta t_{\min ij}$, que emplea la FPGA en detectar un '0' lógico en el proceso de descarga a través de R_{ij} . Así, si R_{kj} se hace tender a 0, (3.8) queda

$$V_j(s) = \frac{V_{DD}}{s + \frac{1}{R_{ij} C_{eq}}} \quad (3.10)$$

con

$$C_{eq_{ij}} = C_j + \sum_{\substack{k=1 \\ k \neq i}}^{M+1} C_{P_{kj}} \quad (3.11)$$

Por otro, si $R_{kj} \rightarrow \infty$ nuevamente se tiene que la expresión (3.8) se transforma en (3.9). Entonces el tiempo necesario para descargar C_j a una tensión VT_j usando la ecuación (3.8) está limitado por los tiempos de descarga a esa misma tensión de dos circuitos RC con condensadores de valores distintos y la misma resistencia. Es decir:

$$\Delta t_{\max ij} = R_{ij} C_{eq_{ij}} \cdot \ln \left(\frac{VT_j}{V_{DD}} \right) \quad (3.12)$$

$$\Delta t_{\min ij} = R_{ij} C_j \cdot \ln \left(\frac{VT_j}{V_{DD}} \right) \quad (3.13)$$

Así, $\Delta t_{\max ij} < \Delta t_{ij} < \Delta t_{\min ij}$ y su valor exacto dependerá del resto de resistencias. Otra forma de interpretar estos resultados consiste en considerar Δt_{ij} como una variable estadística con su correspondientes función de densidad de probabilidad y su desviación estándar $\sigma(\Delta t_{ij})$. Según el proceso que se ha seguido para hallar (3.12) y (3.13), la función de densidad de probabilidad de Δt_{ij} dependerá del rango de valores de las resistencias R_{kj} , de forma que cuanto más pequeño sea el rango en el que puedan variar éstas, menor será $\sigma(\Delta t_{ij})$. Por otra parte, el valor de la desviación estándar también será menor cuanto menor sea la diferencia entre $C_{eq_{ij}}$ y C_j , o lo que es lo mismo cuanto menor sea la suma de los $C_{P_{kj}}$. Pero, como se ha comentado anteriormente, esto tiene una limitación tecnológica que impone un mínimo a $\sigma(\Delta t_{ij})$. También se puede observar que un incremento en el número de columnas, M , de la matriz incrementa el valor de la suma y por lo tanto de

$\sigma(\Delta t_{ij})$. Si lo que nos interesa es la desviación estándar relativa, $\sigma(\Delta t_{ij})/\Delta t_{ij}$, ésta es, según (3.12) y (3.13), función de Ceq_{ij}/C_j y de la función de densidad de probabilidad de las resistencias. Por lo tanto, una posibilidad para disminuirla es el aumento de C_j , pero esto es contrario a la velocidad de sensado de la matriz. Así, existe un compromiso entre velocidad y exactitud en la medida de las resistencias. Ese mismo efecto aparece en el caso de un sensor con un único elemento, allí también un incremento del condensador a descargar significa un aumento de la precisión, en ese caso, relacionada con la disminución del término de error debido a la cuantificación [Reverter, Jordana, et al. 2005]. Obviamente, este error de cuantificación también aparece en las medidas de las resistencias de la matriz. Por lo tanto, y como resumen, la medida del tiempo Δt_{ij} , correspondiente a una resistencia concreta de la matriz, tiene como fuentes de incertidumbre relacionadas con otros sensores a los valores del resto de resistencias de la matriz y la existencia de los condensadores Cp_{kj} . Esto origina una primera fuente de incertidumbre, incertidumbre de *crosstalk*, $\sigma_{crosstalk}(\Delta t_{ij})$. A esto hay que sumar otras fuentes de incertidumbre que tendría un único sensor que no formase parte de la matriz: el ruido de cuantificación $u(z)$ y los ruidos en V_{DD} , VT_j y C_j , que generan lo que denominaremos incertidumbre debida al ruido de disparo, $\sigma_{trigger}(\Delta t_{ij})$.

De igual forma se puede proceder si lo que se pretende es medir el tiempo en alcanzar el '0' lógico para la resistencia de calibración, Δt_{cj} . Este tiempo será también una variable aleatoria con valores extremos dados por

$$\Delta t \max_{c_j} = R_{c_j} Ceq_{c_j} \cdot \ln \left(\frac{VT_j}{V_{DD}} \right) \quad (3.14)$$

$$\Delta t \min_{c_j} = R_{c_j} C_j \cdot \ln \left(\frac{VT_j}{V_{DD}} \right) \quad (3.15)$$

donde

$$Ceq_{c_j} = C_j + \sum_{\substack{k=1 \\ k \neq i}}^{M+1} Cp_{k_j} \quad (3.16)$$

y por lo tanto, con su propia desviación estándar, $\sigma(\Delta t_{c_j})$. Debe notarse que las diferencias entre los valores de Δt_{ij} y Δt_{c_j} no sólo vienen dadas por las diferentes resistencias, R_{ij} y R_{c_j} , sino también porque en general $Ceq_{c_j} \neq Ceq_{ij}$ con $i \neq c$. Esto se debe a los diferentes

condensadores asociados al cableado que una cada sensor con su correspondiente pin de la FPGA (estas diferencias se pueden hacer muy pequeñas con un cuidadoso diseño y una tecnología adecuada, pero siempre van a existir). Así, si se barajase la posición de las resistencias de una columna (a excepción de la de calibración) se encontrarían variaciones en sus valores. Por lo tanto, como se ha mostrado con R_{cj} , se tiene una tercera fuente de incertidumbre asociada a la diferencia entre las capacidades de los nudos de una columna. Ésta originará una desviación estándar en la medida de tiempos que se denominará $\sigma_{column}(\Delta t_{ij})$.

Podría argumentarse de igual modo que existe una incertidumbre debida a la posición en la misma fila pero en distintas columnas. Sin embargo, asumiendo que la distribución de valores de las resistencias es independiente de las filas y que la variación en los condensadores Cp_{ij} es idéntica en todas las columnas, este ruido sólo se debería a las variaciones en C_j . Esto se compensa a través de la resistencia de calibración de cada columna, por lo que se considera despreciable.

Así, la incertidumbre en la medida Δt_{ij} puede expresarse

$$\sigma(\Delta t_{ij}) = \sqrt{\sigma_{crosstalk}^2(\Delta t_{ij}) + \sigma_{trigger}^2(\Delta t_{ij}) + \sigma_{column}^2(\Delta t_{ij})} \quad (3.17)$$

ya que los 3 términos dentro de la raíz son independientes entre sí.

A esta desviación hay que añadir el ruido debido a la cuantificación, $u(z) = T_s / \sqrt{12}$, [ISO 1995] siendo T_s el periodo del reloj del contador que mide el tiempo de descarga.

Aunque las incertidumbres en los valores de V_{DD} , VT_j y C_j puedan ser compensadas utilizando (2.2), con lo que disminuyen los valores de todos los términos dentro de la raíz) sigue existiendo una incertidumbre en la medida debido al *crosstalk RC*. Existen tres posibilidades según (3.12) y (3.13) para disminuir la influencia relativa del *crosstalk*: la primera está en el incremento de C_j , pero como se ha comentado anteriormente esto disminuye la velocidad de escaneo de la matriz. La segunda posibilidad es la disminución de los términos Cp_{ij} , pero esto (una vez realizado un diseño cuidadoso del sistema) está limitado por la tecnología y el número de filas de la matriz. La tercera posibilidad es la limitación en el rango de los valores de las resistencias de los sensores que conforman la matriz, pero obviamente esto, sólo se podrá conseguir en algunos casos muy concretos. Por lo tanto, existe lo que se ha denominado *crosstalk RC* que ha de ser evaluado.

3.2.3. Método de evaluación de la incertidumbre en las medidas

Un método directo para la evaluación de la incertidumbre en la medida de Δt_{ij} consiste en realizar un conjunto de medidas para cada uno de los sensores de la matriz variando las resistencias del resto de sensores en cada una de las medidas, obteniendo de este modo los rangos máximos de variación. Sin embargo, el número de sensores de una matriz no suele ser pequeño, incluso en nuestro caso utilizando una matriz “reducida” de 7x8 sensores, por lo que usar este método de caracterización es inabordable.

En este apartado se propone la evaluación de la incertidumbre total en la medida de cualquier resistencia de la matriz basándose en la independencia de las incertidumbres debidas al *crosstalk*, la columna y el disparo mediante el uso de (3.17). Para ello, se ha diseñado un circuito que simula mediante resistencias los distintos valores que se pueden leer desde todas las posiciones de una matriz con un direccionamiento como el aquí propuesto. El circuito, obviamente, incluye la FPGA y los condensadores necesarios para la conversión A/D.

El método de evaluación de la incertidumbre en la medida utiliza los siguientes pasos:

- En primer lugar, se selecciona un conjunto suficiente de valores de resistencias que dividan, aproximadamente, el rango del sensor en tramos uniformes (en nuestro caso 8). Cada una de estas resistencias se colocará en sucesivas pruebas en las M filas de una columna de la matriz seleccionada para su caracterización. Las demás resistencias de la columna toman el valor medio en el rango. A continuación, se realizan un conjunto de Q medidas de tiempos Δt_{ij} . El proceso se repite M veces barajando de forma circular la posición de la resistencia de manera que, se tienen $M \cdot Q$ medidas de tiempos para cada resistencia. Este conjunto de tiempos tiene una desviación estándar que es debida tanto a la incertidumbre de disparo como a la incertidumbre de posición en la columna, con varianza $\sigma_{trigger+column}^2$.

Para encontrar el valor de cada una de las varianzas por separado, $\sigma_{trigger}^2$ y σ_{column}^2 (para cada resistencia) se propone el siguiente procedimiento. De las Q medidas hechas para cada fila se obtiene un valor de $\sigma_{trigger}^2$. Este valor debe ser en principio muy parecido para todas las filas. Tomaremos el promedio de las M filas como estimación final de $\sigma_{trigger}^2$. Dado que son fuentes de incertidumbre

independientes, su varianza es $\sigma_{trigger+column}^2 = \sigma_{trigger}^2 + \sigma_{column}^2$, y por lo tanto se puede calcular σ_{column}^2 como

$$\sigma_{column}^2 = \sigma_{trigger+column}^2 - \sigma_{trigger}^2 \quad (3.18)$$

- Para hallar $\sigma_{crosstalk}$ se procede de la siguiente manera: se selecciona una fila en la columna utilizada para el conjunto de medidas anteriores. En esta posición de la matriz se irán colocando cada una de las resistencias seleccionadas dentro del rango a caracterizar. Para cada una se harán dos conjuntos de Q medidas. En las primeras Q medidas, a la resistencia seleccionada se añaden, en el resto de posiciones de la columna, la resistencia mínima del sensor. En las segundas Q medidas se sustituyen las resistencias mínimas por las máximas. A continuación se encuentran los tiempos máximos, Δt_{max} , y mínimos, Δt_{min} , para las $2 \cdot Q$ medidas, y su diferencia $\Delta t_{max} - \Delta t_{min}$, será el estimador usado para encontrar el rango en el que se mueve Δt_{ij} para cada resistencia debido al *crosstalk* RC. Finalmente, asumiendo que los valores de las resistencias del sensor tienen una distribución uniforme (son igualmente probables) y por lo tanto, Δt_{ij} también, para cada resistencia, $\sigma_{crosstalk}$ se puede hallar como

$$\sigma_{trigger+crosstalk} = \frac{\Delta t_{max} - \Delta t_{min}}{\sqrt{12}} \quad (3.19)$$

de donde nuevamente podemos despejar $\sigma_{crosstalk}$

$$\sigma_{crosstalk}^2 = \sigma_{trigger+crosstalk}^2 - \sigma_{trigger}^2 \quad (3.20)$$

Así, usando (3.17) se puede obtener la desviación estándar $\sigma(\Delta t_{ij})$ para cada resistencia. A esto habría que añadir la incertidumbre de cuantificación $u(z)$, con lo que obtendríamos una incertidumbre total, $\sigma_T(\Delta t_{ij})$ dada por

$$\sigma_T(\Delta t_{ij}) = \sqrt{\sigma^2(\Delta t_{ij}) + u^2(z)} \quad (3.21)$$

La incertidumbre en todo el rango será la máxima de las incertidumbres halladas.

3.2.4. Materiales y método

El circuito propuesto en Figura 3-2 se ha realizado usando una FPGA Spartan3AN de Xilinx (XC3S50AN-4TQG144C) y un circuito oscilador basado en un cristal de cuarzo con una frecuencia de trabajo de 50 MHz. La matriz de sensores a medir consta de diez filas por ocho columnas. Las reglas de diseño utilizadas coinciden con las descritas en el apartado 2.3.1. Las distintas columnas son eléctricamente independientes entre sí, por lo que las pruebas experimentales se centran en el estudio de una sola columna y de las diez resistencias que pueden ser medidas a través de un único puerto de entrada. Al igual que en el experimento del apartado 2.3.1, el rango de resistencia a medir se corresponde con los valores ofrecidos por un sensor de presión piezorresistivo desarrollado por CIDETEC con valores que pueden variar entre valores entorno a los 7000 Ω para presiones de unos pocos kilopascales hasta las pocas decenas de ohmios para presiones elevadas. En el caso del condensador, se escoge uno de 47 nF y con una tolerancia de $\pm 5\%$, que cumple con las reglas de diseño propuestas en [Reverter *et al.* 2004] para la constante de tiempo óptima que minimice la incertidumbre estándar relativa de la medida, como se comenta en el capítulo 2. Además, con este valor de condensador, la suma de los tiempos empleados en los ciclos de carga/descarga para las diez filas de la matriz permite alcanzar tasas de muestreo superiores a 200 imágenes táctiles por segundo. Debido a la naturaleza del circuito propuesto, hasta diez resistencias pueden ser medidas por un único módulo de captura. Tres de estas resistencias toman valores fijos puesto que se emplean como resistencias de calibración para aproximar el valor de las resistencias de la matriz. Por ello, las resistencias a medir tendrán que ser evaluadas en cada una de las siete filas restantes.

Para el estudio de la incertidumbre de la conversión tiempo-digital se escogen 8 valores de resistencias dentro del rango con una tolerancia del 0,1% para todas ellas. Los valores nominales para las resistencias usadas son: 200 Ω , 1300 Ω , 2400 Ω , 3680 Ω , 4840 Ω , 5820 Ω , 7000 Ω y 7350 Ω . Para cada valor de resistencia, se realiza un análisis estadístico a partir de 3500 medidas, 500 por cada fila de la columna que se está estudiando, obteniendo su desviación estándar $\sigma(X)$.

Para el estudio de la exactitud en la estimación de la resistencia del sensor, se escogen 14 valores con valores nominales: 200 Ω , 762 Ω , 1300 Ω , 1890 Ω , 2400 Ω , 3070 Ω , 3680 Ω , 4100 Ω , 4840 Ω , 5300 Ω , 5820 Ω , 6400 Ω , 7000 Ω y 7350 Ω . En cuanto a las filas

con las resistencias de calibración, dos de ellas son utilizadas para una calibración de dos puntos, con valores aproximadamente situados en el 15% (1300 Ω) y en el 85% (5820 Ω) del rango a medir [Pallàs-Areny *et al.* 2004]. La tercera fila de calibración monta una resistencia con un valor aproximado del 50% del rango (3680 Ω), que se emplea para la calibración de un punto y para el método de calibración propuesto en el apartado 2.4.1 de este trabajo, usando la ecuación (2.2). Para poder emplear la calibración CLChar hay que evaluar en primer lugar la recta de calibración. Esto se hace colocando las 8 resistencias que se emplean en el estudio de la incertidumbre en la misma posición. Se realizan 2 grupos de pruebas para cada resistencia, 500 medidas con el resto de resistencias de la columna de valor mínimo y otras 500 con las resistencias de valor máximo. A partir de estas 8000 medidas se realiza un ajuste lineal como el descrito en el apartado 2.4.1. De esta forma se obtienen los coeficientes de la ecuación (2.2). Posteriormente, sobre el conjunto de 14 valores de resistencias que dividen el rango posible en tramos aproximadamente iguales, se hacen 500 medidas con el resto de resistencias de valores mínimos, 500 con resistencias medias y 500 con resistencias máximas.

3.2.5. Resultados experimentales y discusión

La Tabla 3-1 muestra las distintas incertidumbres en la medida de tiempo que se describen en el apartado 3.2.3.

Tabla 3-1. Incertidumbre en la medida de tiempo para las resistencias de test con $C_j = 47$ nF y $Q = 500$.

Resistencia (Ω)	$\sigma_{trigger+column}$ (ns)	$\sigma_{trigger}$ (ns)	σ_{column} (ns)	$\sigma_{trigger+crosstalk}$ (ns)	$\sigma_{crosstalk}$ (ns)	$\sigma(\Delta t_{ij})$ (ns)	$\sigma_T(\Delta t_{ij})$ (ns)
199,96	13,58	1,62	13,49	2,89	2,39	13,79	14,95
1297,73	18,06	5,24	17,29	11,55	10,29	20,79	21,58
2401,95	18,58	7,93	16,81	25,98	24,74	30,94	31,48
3687,49	29,83	11,49	27,53	30,31	28,05	40,95	41,35
4836,09	32,65	14,29	29,36	43,30	40,87	52,31	52,63
5810,79	40,13	17,77	35,98	56,29	53,41	66,81	67,06
6966,87	38,56	19,98	32,98	59,18	55,70	67,75	67,99
7348,84	38,46	21,42	31,94	70,73	67,41	77,60	77,82

La resolución en bits de la conversión tiempo-digital teniendo en cuenta únicamente los errores de cuantificación es de 13,89 bits. Si se incluye la mayor σ_T obtenida, los bits

efectivos de la conversión (ENOB) son 10,14. La Tabla 3-2 compara el circuito del capítulo 2 con el aquí analizado. Se observa cómo el incremento del número de filas afecta negativamente en la conversión.

Tabla 3-2. Degradación en la resolución debida al incremento del número de filas.

Filas (M)	$\sigma_T(\Delta t_{ij})$ (ns)	ENOB (bits)	Resolución (Ω)
1	21,40	12,04	1,70
7	77,82	10,14	6,34

En cuanto a la exactitud en la estimación de los valores de las resistencias de test, se van a comparar los resultados obtenidos por los métodos de calibración tradicionales de un punto y de dos puntos con el propuesto en el capítulo 2. La Tabla 3-3 muestra el máximo error absoluto obtenido a partir cada uno de los 3 grupos de 500 medidas. En las dos últimas filas se muestran los errores absolutos y relativos máximos bajo cualquier circunstancia.

Tabla 3-3. Datos de exactitud para las distintas técnicas del calibración ($C_j = 47$ nF y $Q = 500$).

Resistencia (Ω)	Error Absoluto Máximo (Ω)								
	Rmin			Rmed			Rmax		
	1 P.	2 P.	CLChar	1 P.	2 P.	CLChar	1 P.	2 P.	CLChar
199,96	13,66	0,88	0,62	13,74	0,90	0,37	13,60	0,77	0,56
763,34	11,70	0,98	0,59	11,59	0,75	0,69	11,44	0,75	0,82
1297,32	10,03	0,65	0,58	10,11	0,76	0,67	9,79	0,65	0,62
1887,55	8,23	1,18	1,22	8,12	1,20	1,11	7,65	1,09	0,63
2401,95	5,69	0,81	0,84	5,42	1,18	1,08	4,98	0,74	1,44
3070,25	4,24	1,93	2,10	3,83	1,81	1,69	3,34	1,81	1,19
3684,25	1,83	2,21	2,21	1,60	1,94	1,98	1,13	2,07	1,51
4083,85	2,04	2,93	3,02	2,31	2,22	1,91	3,13	1,85	1,12
4836,05	5,73	2,19	2,46	5,68	2,32	1,88	6,67	2,02	1,56
5269,05	6,65	3,31	3,49	6,84	3,31	3,12	7,79	3,23	1,89
5813,45	9,32	2,64	2,72	9,88	2,65	2,37	10,63	2,88	1,68
6373,15	12,09	3,07	3,20	12,61	2,95	2,77	13,34	2,80	1,91
6983,15	16,29	2,29	2,35	15,75	2,37	1,81	17,34	2,69	3,41
7349,15	16,52	3,34	3,16	16,55	3,30	2,54	18,86	3,24	3,42
Error Acumulado	124,03	28,39	28,56	124,03	27,67	23,99	129,69	26,59	21,77
Max. Error Abs.	16,52	3,34	3,49	16,55	3,31	3,12	18,86	3,24	3,42
Max. Error Rel. (%)	0,225	0,045	0,066	0,225	0,063	0,059	0,257	0,044	0,047

Como se observa en la Tabla 3-3, los errores para los métodos de calibración de dos puntos y CLChar reflejan valores muy similares. Por otro lado, los errores para el método de un punto son un orden de magnitud mayor. Como el método CLChar hace uso de una única resistencia de calibración parece una mejor solución, ya que, el hardware es más reducido y el tiempo necesario para el escaneo de toda la matriz también.

3.2.6. Conclusiones

Con respecto al ENOB, se obtiene una menor resolución si se compara con la obtenida en el Capítulo 2, donde sólo un sensor era medido. Esto se debe a la aparición de dos nuevas componentes de incertidumbre.

En cuanto a la exactitud en la estimación del valor de resistencias, de nuevo se observa que el método de aproximación CLChar obtiene resultados equiparables a los obtenidos por la técnica de calibración en dos puntos con la ventaja de emplear una sola fila de resistencias de calibración.

3.3. Matrices de gran tamaño

Para matrices con un gran número de táctiles, donde la propuesta del apartado 3.2 implique el uso de un número excesivo de conexiones, o para aquellas que posean una organización donde se comparten las conexiones por filas y columnas, el uso de técnicas de multiplexación es lo más recomendable.

Sin embargo, cualquier tipo de multiplexación de la información de la matriz conlleva *crosstalk*, es decir, la influencia que el resto de sensores tienen sobre la medida de uno de ellos. Por esto, es necesario que la circuitería empleada trate de evitar la aparición de este fenómeno o en su defecto compensarlo.

Teniendo en cuenta el tipo de matriz a emplear y la aplicación a la que va destinada, existen varios tipos de circuitos propuestos en la literatura para realizar el escaneo de la matriz. Estas propuestas utilizan amplificadores operacionales (OAs) y el principio de tierra virtual para intentar aislar la información del elemento a medir (EBT de aquí en adelante) de la influencia del resto de elementos de la matriz.

3.3.1. Circuitos con realimentación resistiva

La Figura 3-5, muestra una propuesta clásica de circuito con realimentación resistiva y tierra virtual para evitar el *crossstalk*. Cada sensor está modelado por una resistencia y comparte conexiones con los sensores de la misma fila y columna. El OA (*Operational Amplifier*) realimentado negativamente mantiene a tierra virtual todas las columnas.

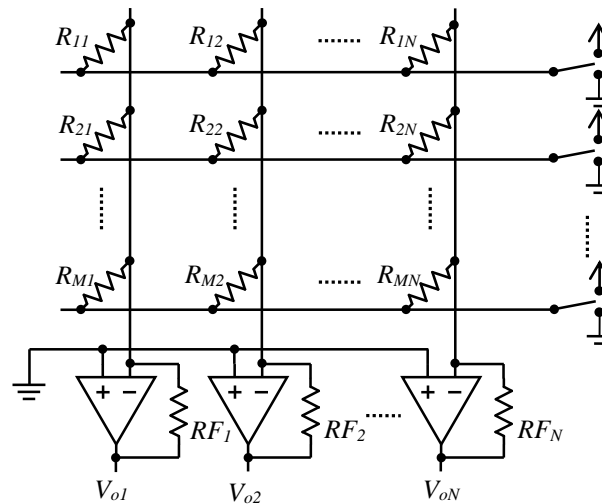


Figura 3-5. Circuito con realimentación resistiva y tierra virtual.

Si se pretende que R_{ij} (la resistencia que modela el táctel de posición fila i , columna j en la matriz) sea el EBT, un circuito de selección debe fijar una tensión de referencia (V_{bias}) en la fila a la que pertenece R_{ij} , manteniendo el resto a 0 V. De esta forma, todas las resistencias que no pertenecen a la fila i tienen sus dos terminales conectados a 0 V, por lo que sólo hay una resistencia en cada columna por la que circula intensidad entre V_{bias} y la entrada inversora del OA. Con esto, los OAs de cada columna j funcionan en una configuración inversora, con una resistencia de realimentación R_F y con la resistencia de entrada R_{ij} . Así, la tensión de salida de cada columna, considerando el OA como ideal es:

$$V_{oj} = -\frac{R_F}{R_{ij}} \cdot V_{bias} \quad (3.22)$$

Es obvio, según la ecuación (3.22), que con esta configuración no existe *crossstalk* entre los distintos sensores de la matriz, ya que V_{oj} sólo depende de R_{ij} y R_F , pero no de la resistencia del resto de sensores.

Las salidas V_{oj} son llevadas posteriormente a un convertor A/D de forma que un procesador digital pueda trabajar con esta información. Este sistema puede ser un

microprocesador con canales A/D, con lo que todo el proceso es realizado sobre un mismo chip (SoC – *System on a Chip*) como se muestra en [Vidal-Verdú, Oballe-Peinado, et al. 2011] siempre que se disponga de tantos canales A/D como columnas posea la matriz.

Sin embargo, esta estructura presenta varios inconvenientes. En primer lugar, debido a que la ganancia es negativa, se necesita que el conversor A/D pueda trabajar con estas tensiones o que V_{bias} sea negativa. En cualquier caso, se precisan valores de tensiones tanto positivos como negativos, con lo que aumenta la circuitería necesaria para generar las tensiones del circuito así como el área ocupada. Una posible solución, que evita las tensiones negativas [Vidal-Verdú, Oballe-Peinado, et al. 2011], es fijar en la entrada no inversora de cada OA la tensión positiva V_{bias} , que es la misma tensión que el circuito de multiplexación pone en las filas no seleccionadas de la matriz, mientras que en la fila seleccionada se colocarán 0 V. Así, el OA de cada columna está ahora en su configuración no inversora y la salida de cada columna será positiva

$$V_{oj} = \left(1 + \frac{R_F}{R_{ij}}\right) \cdot V_{bias} \quad (3.23)$$

Esto presenta un nuevo problema, ya que la ganancia en (3.23) es una cantidad siempre mayor que uno y por lo tanto el rango en el que estarán contenidos los valores de V_{oj} viene dado por

$$\left(1 + \frac{R_F}{RH}\right) \cdot V_{bias} \leq V_{oj} \leq \left(1 + \frac{R_F}{RL}\right) \cdot V_{bias} \quad (3.24)$$

siendo RH y RL la máxima y mínima resistencia del sensor, respectivamente. Así, según (3.24) no todo el rango entre 0 y V_{DD} sería aprovechado por el conversor A/D, con la consiguiente pérdida de resolución. Este problema puede ser reducido, aunque no eliminado, con valores adecuados de V_{bias} y R_F .

Un segundo problema aparece derivado del hecho de que, aunque el microcontrolador pueda tener varios canales de conversión A/D, las operaciones de procesado de esta información digital son realizadas en serie con la consiguiente pérdida del paralelismo obtenido con la lectura de la matriz.

En tercer lugar, la no-idealidad del OA origina, como se indica en [Saxena et al. 2011], la aparición de *crosstalk* y, aunque se analiza el efecto de la ganancia finita del OA (A)

en el funcionamiento del circuito, no se proponen técnicas para su reducción salvo la de disminuir el valor de R_F , con la consiguiente pérdida de resolución en la conversión A/D según (3.24). Tampoco se analizan otros efectos no ideales como la tensión de *offset* “ θ ”, la intensidad de polarización del OA o el efecto de las resistencias internas de los *buffers* de selección de fila.

3.3.2. Circuito con realimentación capacitiva

Para poder implementar una conexión directa, eliminando con ello la circuitería de acondicionamiento y la conversión A/D de la propuesta clásica, este trabajo propone una solución alternativa (Figura 3-6) que emplea una FPGA como procesador digital. Aquí, la resistencia de realimentación es sustituida por un condensador, C_j . El circuito mantiene tierra virtual en todas las columnas de la matriz debido a la realimentación negativa proporcionada por este condensador.

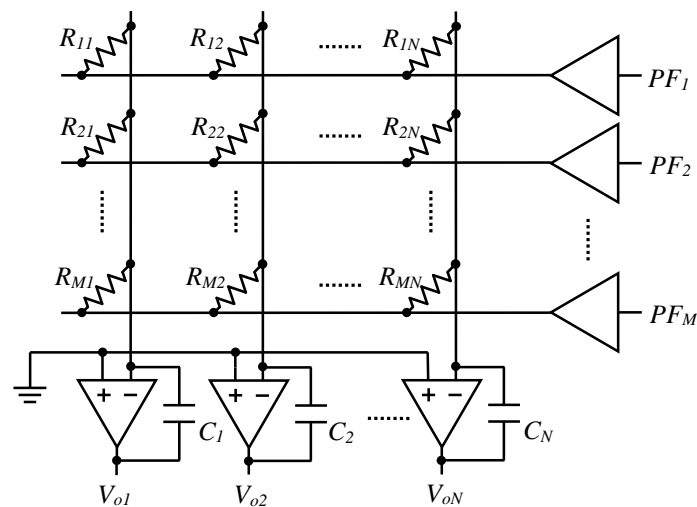


Figura 3-6. Circuito con realimentación capacitiva y tierra virtual.

La forma en la que se lee el valor de resistencia R_{ij} de un EBT es mediante la medición del tiempo de descarga de C_j . Inicialmente el condensador cargado mantiene una tensión, en el nudo de salida del OA, que es interpretada como un ‘1’ lógico en un pin de entrada de un procesador digital. A medida que el condensador se descarga, se alcanza una tensión que hace que el pin de entrada interprete la entrada como un ‘0’ lógico. Así, la medida será la diferencia de tiempo entre que se inicia la descarga del condensador y la detección

de un '0' a través del pin de entrada del procesador digital. Cada una de las entradas puede ser procesada de forma paralela por el hardware programable de la FPGA.

Como se ilustra en la Figura 3-6, las líneas de fila del sensor son las salidas de M pines, configurados como salida en la FPGA. Si se pretende medir la resistencia del sensor con valor R_{ij} , se coloca el pin de la fila i , PF_i , a nivel alto (V_{DD}) y el resto de pines de fila a valor 0 V. De esta forma, sólo circulará corriente a través de las resistencias de la fila seleccionada (una única resistencia en cada columna).

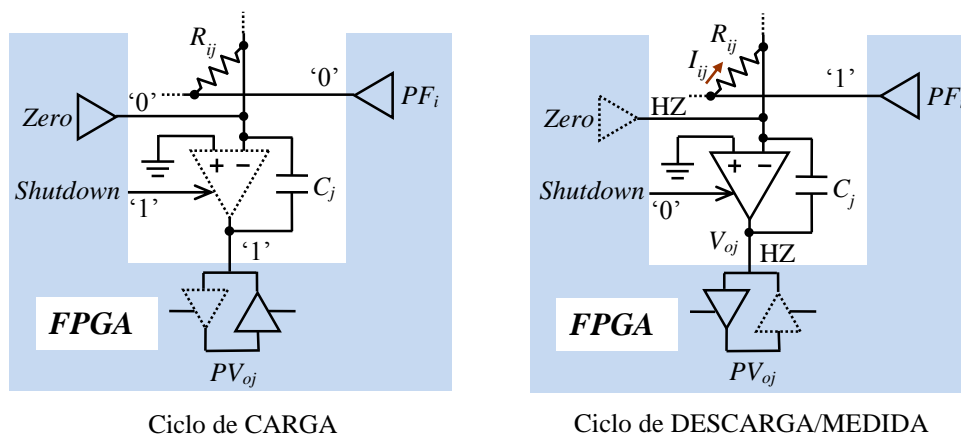


Figura 3-7. Ciclos de CARGA y DESCARGA/MEDIDA para la fila i columna j .

Para poder implementar esta aproximación, en un primer ciclo (CARGA) se han de cargar todos los condensadores C_j con una tensión V_{DD} en el terminal de salida del OA manteniendo 0 V en el terminal inversor. Esto se lleva a cabo mediante dos *buffers* de salida de la FPGA, $Zero$ y PV_{oj} en la Figura 3-7 (observar que los pines que se encargan de la lectura de tensión PV_{oj} están configurados como salida en esta fase). También se mantiene desactivado el OA (a través del pin *shutdown*) y las salidas de la FPGA que controlan la selección de filas, PF_k se encuentran todas a 0 V. A continuación, se entra en un nuevo ciclo (ACTIVACIÓN) donde se activan los OA a través de la señal *shutdown*, a la vez que se configura como entrada (alta impedancia) el pin V_{oj} . Finalmente, se lleva a cabo el ciclo de medida (DESCARGA) donde se produce la descarga de los condensadores. Para esto, se coloca a V_{DD} el *buffer* de salida de la fila que se desea escanear (PF_i), dejando a 0 V todos los demás *buffers* de selección de fila. De esta forma, la tensión de salida de todos los OAs irá disminuyendo a medida que la intensidad que atraviesa las distintas R_{ij} va descargando los condensadores. Operando de esta forma, al comienzo del ciclo DESCARGA, todos los pines PV_{oj} , tienen una entrada de valor V_{DD}

que va disminuyendo, a medida que pasa el tiempo, hasta alcanzar un valor V_{Tj} , la tensión umbral para la cual los *buffers* de entrada de los pines PV_{oj} comienzan a interpretar la entrada como un '0' lógico que es transmitido al interior de la FPGA. El tiempo en el que V_{oj} pasa de V_{DD} a V_{Tj} en el ciclo de DESCARGA será Δt_{ij} (Figura 3-8). Nótese que cuando el amplificador operacional se satura al descargarse por completo el condensador, deja de haber realimentación y por tanto en la entrada inversora no tenemos una tierra virtual. Para evitar la aparición de *crosstalk*, el pin Zero coloca un '0' en dicha entrada. Esto permite también que al final del ciclo de medida del conjunto de la matriz todos los condensadores se encuentren descargados, obteniéndose para el siguiente ciclo de carga una tensión más homogénea entre todas las columnas.

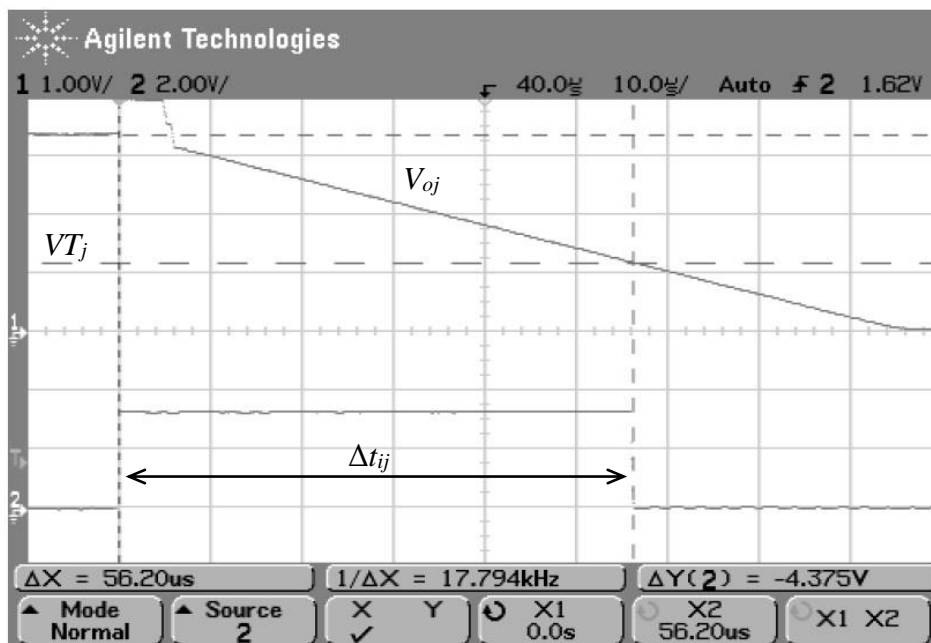


Figura 3-8. Ejemplo de medida de tiempo del circuito con realimentación capacitiva.

Considerando el OA ideal y el valor de la resistencia del sensor constante, un sencillo análisis indica que

$$\Delta t_{ij} = R_{ij} C_j \frac{V_{DD} - V_{Tj}}{V_{DD}} \quad (3.25)$$

donde se observa que existe una relación lineal entre el tiempo medido Δt_{ij} y el valor de la resistencia R_{ij} (nótese la descarga lineal en la Figura 3-8).

Las principales ventajas del circuito aquí propuesto están en la eliminación del conversor A/D y en la posibilidad llevar a cabo un procesado en paralelo de la información procedente del conjunto de sensores medidos simultáneamente por parte de la FPGA, con el consiguiente aumento de la velocidad de procesado, así como de una reducción de área y consumo. Además, se añade una fila adicional (fila de calibración) con el fin de no tener que evaluar C_j y VT_j en (3.25) ya que, estas magnitudes son difíciles de medir al variar su valor con la tensión de alimentación, el tiempo y la temperatura. Este procedimiento es equivalente a lo que se conoce en la literatura como calibración de un punto [Reverter, Jordana, et al. 2005].

Sin embargo, esta calibración no tiene en cuenta las resistencias de los *buffers* de la FPGA RB_k que direccionan cada fila k , originando un error en la estimación de R_{ij} . Si los valores de RB_k fuesen constantes e iguales para todos los *buffers*, su valor podría ser tenido en cuenta con una calibración de dos puntos [Reverter, Jordana, et al. 2005], lo que supondría añadir una segunda fila de calibración al circuito. Además, RB_k aunque es constante durante el ciclo de DESCARGA, no es igual en todos los *buffers* ya que depende de las resistencias de la matriz a las que está conectada, por lo que incluso añadiendo una segunda fila de calibración no se evitan los errores debidos a esta resistencia. Es importante destacar que, no sólo hay que evaluar su valor, sino evitar el efecto de *crosstalk* que origina al estar unida a varias resistencias de la matriz, una resistencia por cada columna, cuestión que no resuelve una calibración de dos puntos. A esto hay que sumar el hecho de que los OAs utilizados no son ideales y por lo tanto presentan efectos de segundo orden, lo que también origina *crosstalk*, como se mostrará más adelante en este capítulo.

3.3.3. Mejoras para los circuitos con realimentación capacitiva

En cada uno de los apartados de esta sección se proponen soluciones a cada uno de los problemas planteados anteriormente.

3.3.3.1. Estimación de R_{ij} considerando el efecto de la resistencia de los *buffers* de selección de fila y variaciones en los valores de C_j y VT_j

Para poder reducir los efectos de la variación en las medidas de C_j y VT_j y, simultáneamente, eliminar el efecto de la resistencia de los *buffers* de selección de fila, se propone el circuito de la Figura 3-9. En él, se han añadido (en rojo) una fila y una columna de calibración. Por esto, el circuito queda con $M + 1$ filas y $N + 1$ columnas, necesitando $N + M + 1$ resistencias adicionales con valores conocidos y un OA extra para la columna de calibración.

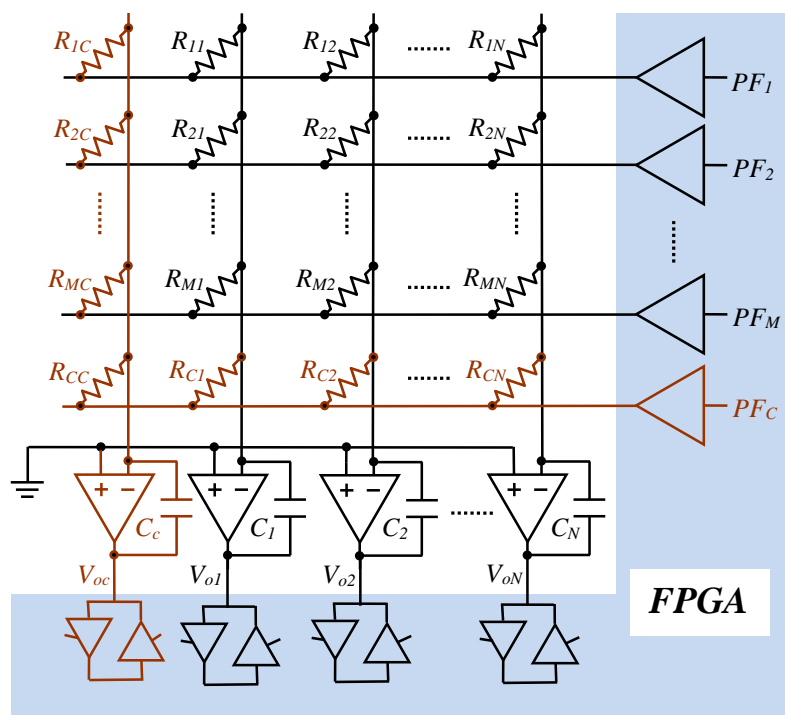


Figura 3-9. Circuito para considerar las resistencias de *buffers* y evitar errores de medida en C_j y VT_j .

El funcionamiento del circuito es exactamente el mismo que se ha comentado en el apartado anterior, teniendo en cuenta que en este caso hay una fila más a medir. Por simplicidad no se muestran los *buffers Zero* de la Figura 3-7, para cada una de las columnas.

En la Figura 3-10 se muestra el circuito equivalente para medir las resistencias de la fila i . Aquí, se ha sustituido el *buffer* de control de filas por su correspondiente resistencia de salida RB_k^p . En estos *buffers*, las resistencias para una salida de nivel alto (RB_k^p) son

diferentes de las que aparecen en una salida de nivel bajo (RB_k^n), aunque ambas suelen ser pequeñas (10-50 Ω) debido a la tecnología CMOS empleada en su fabricación.

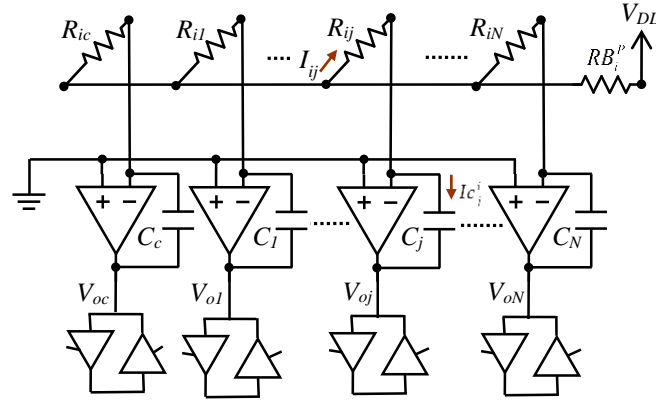


Figura 3-10. Circuito equivalente para escanear los valores de las resistencias de la fila i .

Suponiendo que los OA son ideales, el valor de Δt_{ij} viene dado por la expresión (3.25). Sin embargo, en ésta ecuación no se tienen en cuenta las distintas RB_k . Para ilustrar la influencia de estas resistencias hay que calcular la intensidad I_{ij}^i que entra por la resistencia R_{ij} y descarga al condensador C_j cuando está activada la fila i . De un sencillo análisis del circuito de la Figura 3-10 se obtiene:

$$I_{ij}^i = I_{R_{ij}} = \frac{V_{DD}}{R_{ij}} \cdot \frac{RP_i}{RB_i^p + RP_i} \quad (3.26)$$

con lo que el tiempo para descargar C_j a VT_j viene dado por

$$\Delta t_{ij} = C_j \frac{V_{DD} - VT_j}{I_{ij}} = R_{ij} C_j \frac{V_{DD} - VT_j}{V_{DD}} \cdot \frac{RB_i^p + RP_i}{RP_i} \quad (3.27)$$

donde $RP_i = R_{i1} \parallel R_{i2} \parallel \dots \parallel R_{iN} \parallel R_{ic}$ es el paralelo de todas las resistencias que están siendo medidas, incluyendo la resistencia de calibración.

Cabe destacar que la expresión (3.27) muestra la aparición de *crossstalk* en el circuito, pues Δt_{ij} ya no es función de una única resistencia R_{ij} , sino de todas las resistencias de la misma fila (a través de RP_i). Así, Δt_{ij} se debe calcular mediante un sistema no lineal de $M + 1$ ecuaciones en el que se necesitan conocer de forma exacta los valores de C_j , VT_j , RB_i^p y los tiempos de descarga de todas las columnas cuando la fila i esté seleccionada. Además, hay que tener en cuenta que RB_i^p no es constante y varía a lo largo del proceso

de descarga. A continuación se muestra cómo proceder para evitar estos inconvenientes en el cálculo de R_{ij} haciendo uso de distintos tiempos de medida de la matriz.

Una vez completado el proceso de medida de todas las resistencias de la matriz, se dispone de todos los tiempos almacenados en la FPGA. Así, se puede usar Δt_{ic} para calcular el cociente $\Delta t_{ij} / \Delta t_{ic}$ y expresar R_{ij} a partir de la ecuación (3.27) de la siguiente forma

$$R_{ij} = \left(\frac{C_c \cdot V_{DD} - VT_c}{C_j \cdot V_{DD} - VT_j} \right) \cdot \frac{\Delta t_{ij}}{\Delta t_{ic}} R_{ic} \quad (3.28)$$

En esta expresión, sólo la parte entre paréntesis es desconocida, sin embargo, al disponer también de Δt_{cj} y Δt_{cc} se puede hallar su valor procediendo como se ha hecho para hallar la expresión (3.28)

$$\frac{C_c \cdot V_{DD} - VT_c}{C_j \cdot V_{DD} - VT_j} = \frac{\Delta t_{cc}}{\Delta t_{cj}} \frac{R_{cj}}{R_{cc}} \quad (3.29)$$

sustituyendo (3.29) en (3.28)

$$R_{ij} = R_{ic} \frac{R_{cj}}{R_{cc}} \frac{\Delta t_{ij}}{\Delta t_{ic}} \frac{\Delta t_{cc}}{\Delta t_{cj}} \quad (3.30)$$

Una vez medida la matriz de sensores, todos los términos de (3.30) son conocidos por lo que no se necesita modificar el proceso de escaneo indicado en el apartado 3.3.2 para obtener los valores de las distintas resistencias R_{ij} . Las operaciones necesarias para hallar R_{ij} pueden ser realizadas en la FPGA y sus resultados transmitidos o procesados mientras se procede a la extracción de los datos de un nuevo *frame* de la matriz de sensores.

La ecuación (3.30) se puede expresar también en función del número digital de ciclos (D) con los que la FPGA mide los distintos tiempos de esta ecuación. Así, si $\Delta t = D \cdot T_s$, siendo T_s el periodo del reloj de los contadores, (3.30) queda

$$R_{ij} = R_{ic} \frac{R_{cj}}{R_{cc}} \frac{D_{ij}}{D_{ic}} \frac{D_{cc}}{D_{cj}} \quad (3.31)$$

Cabe destacar que en (3.26), para deducir R_{ij} se han utilizado unas resistencias RB_k que modelan el funcionamiento de cada *buffer*. Sin embargo, si R_{ij} es evaluada mediante (3.30) estos efectos no tienen influencia al no aparecer RB_k en la expresión.

3.3.3.2. Limitaciones en el rango de resistencias a medir

Para poder usar la ecuación (3.30) de forma correcta se ha de determinar el rango de resistencias que se van a medir con el circuito de la Figura 3-9. Dos hechos principales limitan este rango. Por un lado, la máxima intensidad que puede proporcionar la FPGA, que depende de cada modelo concreto, debiendo ser cuidadosamente examinada durante la fase de diseño. No afecta al correcto funcionamiento del circuito el hecho de que la salida del *buffer* proporcione una tensión inferior a V_{DD} en una determinada cantidad, ya que la limitación viene realmente dada por la máxima intensidad que puede proporcionar un *buffer* concreto o el conjunto de *buffers* que seleccionan las filas de la matriz sin afectar al correcto funcionamiento de la FPGA.

En segundo lugar, se debe destacar que la intensidad proporcionada por cada *buffer* depende inversamente de RP_i y aumenta con el número de columnas de la matriz o con la disminución de los valores de las resistencias de la fila seleccionada (Figura 3-10). Esto limita el tamaño de la matriz a procesar y los posibles rangos de resistencias que pueden tomar cada uno de los sensores. Para evitar esta restricción se puede aumentar la resistencia de cada fila a medir añadiendo una resistencia en serie RS_i a la salida de cada *buffer*, que se sumará a RB_k disminuyendo así la intensidad proporcionada por el *buffer*. Aunque se añada esta resistencia, se puede seguir usando la expresión (3.30) para el cálculo de R_{ij} .

No obstante, se sigue manteniendo otra limitación relacionada con el rango de resistencias a medir, ya que cada resistencia de los sensores de una fila emplea un tiempo distinto en descargar el condensador de su columna. Así, si en la misma fila i está la resistencia de valor más bajo posible, RL (en la columna l), y otra del valor más alto posible, RH (en la columna h), puede ocurrir que la primera descargue completamente su condensador a '0'V mientras que la segunda no ha conseguido todavía que su tensión de salida baje por debajo de VT_h . En esta situación, el condensador de la columna l sigue recibiendo intensidad de la resistencia del sensor, pero como V_{ol} ha alcanzado su mínimo valor, el OA entra en un modo de operación no lineal (aunque se trate de un OA *rail-to-tail*) haciendo que la tensión en el nudo de entrada inversora del OA comience a aumentar, dejando de ser ésta una tierra virtual. Si esto ocurre, aparecerá el fenómeno de *crosstalk* a través de las resistencias de las filas no seleccionadas (tal como veremos en la sección

3.3.3.4), afectando a la medida de tiempo de la resistencia RH que todavía no ha sido realizada.

A continuación se define la relación que deben cumplir para que no ocurra la situación de saturación anteriormente mencionada.

Sean I_{RH} e I_{RL} las intensidades que circulan por las resistencias RH y RL respectivamente. La tensión de salida de una columna durante la fase de DESCARGA viene dada por

$$V_o = V_{DD} - \frac{I \cdot t}{C} \quad (3.32)$$

Así, el tiempo necesario para que RL descargue el condensador de su columna a 0V será

$$t_{RL} = \frac{V_{DD} \cdot C_L}{I_{RL}} \quad (3.33)$$

Para que no ocurra el efecto de *crosstalk* por esta causa, la tensión de salida de la columna h ha debido caer por debajo de la tensión umbral VT (habiéndose producido ya la medida de tiempo). Por lo que según (3.32)

$$V_{oh}(t = t_{RL}) = V_{DD} - \frac{I_{RH} \cdot t_{RL}}{C_H} < VT \quad (3.34)$$

sustituyendo (3.33) en (3.34) y suponiendo el mismo valor de capacidad para los condensadores de todas las columnas, se obtiene

$$\frac{I_{RL}}{I_{RH}} < \frac{V_{DD}}{V_{DD} - VT} \quad (3.35)$$

Utilizando la ecuación (3.26) se puede encontrar los valores de I_{RL} e I_{RH} para, sustituyéndolos en (3.35), obtener

$$RH < \frac{V_{DD}}{V_{DD} - VT} \cdot RL \quad (3.36)$$

Así, (3.36) muestra las limitaciones en los posibles valores de las resistencias de los sensores para que el circuito funcione correctamente.

Para valores de RL lo suficientemente grandes, los efectos de incumplir (3.36) son muy pequeños, en este caso bastaría con la activación del pin *Zero* de la FPGA en el instante

en el que el nudo de salida alcance la tensión VT . Sin embargo, para valores pequeños de las resistencias a medir, el efecto del *crosstalk* es cada vez más importante debido a la resistencia del driver del pin Zero y la corriente grande que tiene que drenar.

3.3.3.3. Incremento del rango de resistencias

Para incrementar el rango de resistencias a medir no es necesario modificar el diseño de la Figura 3-9, tan sólo se precisa la lectura de dos filas de forma simultánea: una de la matriz de sensores (i) y otra, la fila de resistencias de calibración (c).

Con este direccionamiento, los condensadores de cada columna se descargan a través de dos resistencias. Suponiendo nuevamente que en la fila que se quiere medir se encuentran las resistencias máxima y mínima de la matriz, RH y RL en las columnas h y l . Simultáneamente, se miden las resistencias de calibración de las mismas columnas: R_c (todas con el mismo valor), así pues la ecuación (3.35) se transforma ahora en

$$\frac{I_{RL} + I_{R_c}}{I_{RH} + I_{R_c}} < \frac{V_{DD}}{V_{DD} - VT} \quad (3.37)$$

Sustituyendo nuevamente los valores de las intensidades obtenidas y usando (3.26) en esta expresión, se puede escribir

$$RH < \frac{V_{DD}}{V_{DD} - VT} \cdot RL + \frac{VT}{V_{DD} - VT} \cdot \frac{RP_c}{RP_i} \cdot \frac{RB_i^p + RP_i}{RB_c^p + RP_c} \cdot \frac{RH \cdot RL}{R_c} \quad (3.38)$$

donde se han vuelto a considerar las tensiones umbrales y los condensadores de todas las filas iguales. Reagrupando en el miembro de la izquierda de (3.38) todos los términos con RH se puede escribir

$$RH \cdot \left[V_{DD} - VT \cdot \left(1 + \frac{RP_c}{RP_i} \cdot \frac{RB_i^p + RP_i}{RB_c^p + RP_c} \cdot \frac{RL}{R_c} \right) \right] < V_{DD} \cdot RL \quad (3.39)$$

Se puede conseguir una expresión más simple con la aproximación $RB^p \ll RP$ (las resistencias de los *buffers* de selección de fila tienen valores muy bajos), por lo que de (3.39) se obtiene un límite superior para RH como

$$RH < \frac{V_{DD} \cdot RL}{V_{DD} - VT \cdot \left(1 + \frac{RL}{R_C}\right)} \quad (3.40)$$

Si se comparan las expresiones (3.36) y (3.40) se observa cómo el rango se ha ampliado. Incluso se pueden eliminar las restricciones sobre RH haciendo el denominador de (3.40) igual a 0. Esto significa que R_C debe valer

$$R_C = \frac{VT}{V_{DD} - VT} \cdot RL \quad (3.41)$$

No obstante, si la restricción $RB^p \ll RP$ no se cumple, ya sea porque el número de columnas es grande, los valores de RL pequeños o se ha introducido una resistencia en serie con el *buffer*, RH seguirá limitada por (3.39), pero, si en esta expresión se hace que el término entre corchetes sea menor o igual a 0, cualquier valor de RH sería posible. Modificando (3.39) se puede escribir esta condición como

$$V_{DD} - VT \cdot \left(1 + \frac{RB_i^p / RP_i + 1}{RB_c^p (N+1) + R_C} \cdot RL\right) \leq 0 \quad (3.42)$$

Para cumplir la restricción, se pueden modificar los valores de R_C y RB_i^p (incluyendo una resistencia en serie con el *buffer*, RS_i). Así, se puede encontrar un límite superior para el valor de R_C y un límite inferior para $RB_i^p + RS_i$. Estos límites serán más restrictivos cuando RP_i sea máxima. RP_{imax} se obtiene de

$$\frac{1}{RP_{imax}} = \frac{1}{RL} + \frac{1}{R_c} + \frac{N-1}{RH} \quad (3.43)$$

Teniendo en cuenta todo lo anterior, se puede despejar R_C en (3.42) obteniendo

$$R_C < \frac{VT}{V_{DD} - VT} \cdot RL \cdot \left(1 + \frac{RB_i^p + RS_i}{RP_{imax}}\right) - RB_c^p (N+1) \quad (3.44)$$

El miembro de la derecha de (3.44) ha de ser mayor que 0, pero esto siempre se puede conseguir gracias a RS_i , aun cuando el número de columnas N sea grande.

Sin embargo, para obtener (3.44) se direccionan simultáneamente dos filas de la matriz y por lo tanto, la expresión (3.30) deja de ser válida ya que, no es una única intensidad la

que descarga al condensador sino la suma de dos: la intensidad que circula por la resistencia R_{ij} y la que circula por la resistencia R_{cj} .

Para encontrar una expresión equivalente a (3.30) y determinar el valor de R_{ij} se necesitan 4 tiempos de descarga: $\Delta t'_{ij}$, $\Delta t'_{ic}$, Δt_{cj} y Δt_{cc} . Los dos primeros son los tiempos de descarga de las columnas j y c cuando simultáneamente están seleccionadas las filas i y la de calibración c . Δt_{cj} y Δt_{cc} son los tiempos de descarga de las columnas j y c cuando sólo se selecciona la fila de calibración. Para calcular Δt_{cj} y Δt_{cc} se puede utilizar la expresión (3.27), mientras que para calcular $\Delta t'_{ij}$, $\Delta t'_{ic}$ se tiene que tener en cuenta que son dos las intensidades que descargan al condensador y por lo tanto sus valores son

$$\begin{aligned}\Delta t'_{ij} &= C_j \frac{V_{DD} - VT_j}{I_{ij} + I_{cj}} \\ \Delta t'_{ic} &= C_c \frac{V_{DD} - VT_c}{I_{ic} + I_{cc}}\end{aligned}\quad (3.45)$$

Teniendo en cuenta estas expresiones y utilizando las obtenidas en (3.26) para las intensidades, se puede expresar R_{ij} como

$$R_{ij} = \frac{R_{cj} R_{ic}}{R_{cc}} \frac{\Delta t'_{ij}}{\Delta t'_{ic}} \frac{\Delta t_{cj}}{\Delta t_{cc}} \left(\frac{\Delta t_{cc} - \Delta t'_{ic}}{\Delta t_{cj} - \Delta t'_{ij}} \right) \quad (3.46)$$

Por lo tanto, modificando el procedimiento de escaneo de la matriz y seleccionando los valores adecuados de R_C se pueden medir resistencias de rangos muy amplios, determinando el valor de R_{ij} a partir de 4 tiempos de descarga.

Hay que destacar que el tiempo empleado en procesar toda la matriz para obtener los datos de la ecuación (3.30) es el mismo que el usado para obtener los datos de la ecuación (3.46). El número de medidas de tiempos a guardar en la memoria interna de la FPGA es el mismo. Por otra parte, los cálculos para obtener R_{ij} por la ecuación (3.46) pueden ser realizados en la FPGA mientras se obtienen los datos para un nuevo *frame*.

3.3.3.4. Crosstalk debido a las tensiones de *offset* de los OA

Los OAs presentan tensiones de *offset* (θ) que, dependiendo de los modelos usados, pueden variar en el rango entre milivoltios y microvoltios. Como las tensiones de *offset* son valores aleatorios determinados por las variaciones en los procesos de fabricación de

los transistores, cada OA, aunque se trate del mismo modelo, puede tener un valor distinto. Además, como en el circuito se fija la tensión de la entrada no inversora del OA a tierra, la tensión de *offset* aparece en el terminal inversor del OA. Por esto, las tensiones de la entrada inversora de todos los OAs pueden ser diferentes entre sí, dando lugar a la aparición de *crosstalk*.

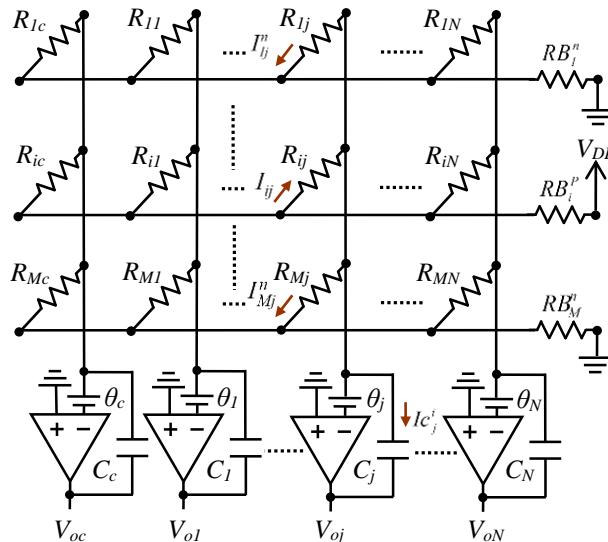


Figura 3-11. Circuito equivalente para la activación de una única fila.

El circuito utilizado para analizar el *crosstalk* debido al *offset* de los OAs, se muestra en la Figura 3-11. En él aparecen las diferentes tensiones de *offset* de los OAs, θ_j ; $j \in \{1, 2, \dots, N, c\}$ y se ilustra la situación en la que se direcciona una única fila de sensores, fila i . Todos los *buffers* de selección de fila de la FPGA han sido sustituidos por sus correspondientes resistencias, RB_i^p , para el caso de filas seleccionadas a V_{DD} o RB_i^n para filas a 0 V.

El valor a encontrar es la intensidad Ic_j^i que entra en el condensador C_j . Esta intensidad tiene varios componentes. En la Figura 3-12 se muestran los componentes de Ic_j^i .

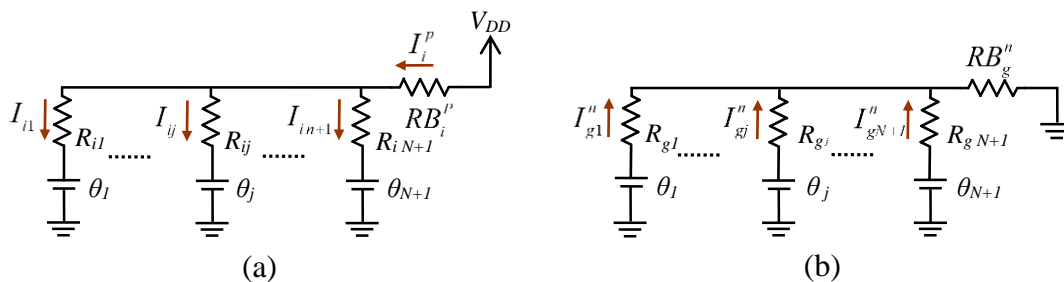


Figura 3-12. Detalle del circuito para el cálculo de las intensidades durante la activación de la fila i .

Así, usando esta ecuación, se puede escribir

$$I_i^p = \sum_{k=1}^{N+1} I_{ik} = I_{ij} \cdot \frac{R_{ij}}{RP_i} + \sum_{k=1}^{N+1} \frac{\theta_{jk}}{R_{ik}} \quad (3.47)$$

donde $\theta_{jk} = \theta_j - \theta_k$, siendo $k = N + 1$ la columna de calibración.

Teniendo en cuenta que V_{DD} se puede expresar como

$$V_{DD} = I_i^p \cdot RB_i^p + I_{ij} \cdot R_{ij} + \theta_j \quad (3.48)$$

se puede despejar I_{ij} usando (3.47)

$$I_{ij} = \left(V_{DD} - \theta_j - RB_i^p \cdot \sum_{k=1}^{N+1} \frac{\theta_{jk}}{R_{ik}} \right) \cdot \frac{RP_i}{RB_i^p + RP_i} \cdot \frac{1}{R_{ij}} \quad (3.49)$$

En segundo lugar, están todas las intensidades que, a través del resto de resistencias de la columna j son drenadas a tierra (Figura 3-12(b)). Tal como aparece en dicha figura, sea I_{gj}^n la intensidad que circula por la resistencia, R_{gj} , con $g \neq i$. Usando el mismo análisis que el realizado para el cálculo de I_{ij} en la Figura 3-12(a), sustituyendo V_{DD} por 0 V, RB_i^p por RB_i^n y tomando los índices adecuados, se llega a

$$I_{gj}^n = \left(\theta_j + RB_g^n \cdot \sum_{k=1}^{N+1} \frac{\theta_{jk}}{R_{gk}} \right) \cdot \frac{RP_g}{RB_g^n + RP_g} \cdot \frac{1}{R_{gj}} \quad (3.50)$$

Como a través de todas las resistencias de la columna j se drena intensidad a tierra excepto por R_{ij} , se puede calcular Ic_j^i como

$$Ic_j^i = I_{ij} - \sum_{g=1}^{M+1} I_{gj}^n + I_{ij}^n \quad (3.51)$$

siendo $g = M + 1$ la fila de calibración. Sustituyendo (3.49) y (3.50) en (3.51) se obtiene

$$\begin{aligned} Ic_j^i = & \frac{V_{DD}}{RB_i^p + RP_i} \cdot \frac{RP_i}{R_{ij}} + \frac{RP_i (RB_i^n - RB_i^p)}{(RB_i^n + RP_i)(RB_i^p + RP_i)} \cdot \frac{RP_i}{R_{ij}} \cdot \sum_{k=1}^{N+1} \frac{\theta_{jk}}{R_{ik}} + \\ & + \theta_j \cdot \frac{RB_i^p - RB_i^n}{(RB_i^n + RP_i)(RB_i^p + RP_i)} \cdot \frac{RP_i}{R_{ij}} - \theta_j \sum_{g=1}^{M+1} \frac{RP_g}{RB_g^n + RP_g} \cdot \frac{1}{R_{gj}} - \\ & - \sum_{g=1}^{M+1} \frac{RB_g^n \cdot RP_g}{RB_g^n + RP_g} \cdot \frac{1}{R_{gj}} \sum_{k=1}^{N+1} \frac{\theta_{jk}}{R_{gk}} \end{aligned} \quad (3.52)$$

donde, separando $\theta_{jk} = \theta_j - \theta_k$, se puede escribir

$$Ic_j^i = \frac{V_{DD}}{RB_i^p + RP_i} \cdot \frac{RP_i}{R_{ij}} + \frac{RP_i (RB_i^p - RB_i^n)}{(RB_i^n + RP_i)(RB_i^p + RP_i)} \cdot \frac{RP_i}{R_{ij}} \cdot \sum_{k=1}^{N+1} \frac{\theta_k}{R_{ik}} + \quad (3.53)$$

$$-\theta_j \sum_{g=1}^{M+1} \frac{1}{R_{gj}} + \sum_{g=1}^{M+1} \frac{RB_g^n}{RB_g^n + RP_g} \cdot \frac{RP_g}{R_{gj}} \sum_{k=1}^{N+1} \frac{\theta_k}{R_{gk}}$$

expresión que caracteriza el *crosstalk* del circuito cuando la entrada inversora de los OAs no tienen una tensión de tierra virtual.

Cabe destacar que esta expresión se puede escribir de una forma abreviada

$$Ic_j^i = \frac{F(i)}{R_{ij}} + G(j) \quad (3.54)$$

donde $F(i)$ es una función que depende del índice i pero no de j ,

$$F(i) = \frac{V_{DD} \cdot RP_i}{RB_i^p + RP_i} + \frac{RP_i^2 (RB_i^p - RB_i^n)}{(RB_i^n + RP_i)(RB_i^p + RP_i)} \cdot \sum_{k=1}^{N+1} \frac{\theta_k}{R_{ik}} \quad (3.55)$$

y $G(j)$ es una función que depende del índice j pero no de i .

$$G(j) = -\theta_j \sum_{g=1}^{M+1} \frac{1}{R_{gj}} + \sum_{g=1}^{M+1} \frac{RB_g^n \cdot RP_g}{RB_g^n + RP_g} \cdot \frac{1}{R_{gj}} \sum_{k=1}^{N+1} \frac{\theta_k}{R_{gk}} \quad (3.56)$$

Además usando (3.50) es fácil comprobar

$$\sum_{g=1}^{M+1} I_{gj}^n = -G(j) \quad (3.57)$$

resultado que será utilizado en el siguiente apartado.

De esta forma, se ha separado la dependencia de Ic_j^i en un término para las filas, otro para las columnas y el valor de R_{ij} . Esto permitirá, como se observa en el apartado siguiente, diseñar una estrategia para encontrar el valor de R_{ij} teniendo en cuenta los efectos de las tensiones de *offset* de los OA. También se puede observar como de la ecuación (3.53), haciendo '0' todas las tensiones de *offset*, se puede derivar la ecuación (3.26). A partir de la ecuación (3.53) se puede analizar el comportamiento del *crosstalk*

en la matriz. Así, si las resistencias de los *buffers* RB^n y RB^p son muy pequeñas, o pequeñas comparadas con RP , (3.53) se simplifica

$$Ic_j = \frac{V_{DD}}{RB_i^p + RP_i} \cdot \frac{RP_i}{R_{ij}} - \theta_j \sum_{g=1}^{M+1} \frac{1}{R_{gj}} \quad (3.58)$$

donde ha desaparecido el efecto de *crosstalk* debido a resistencias de otras columnas.

Así, Ic_j^i sólo queda modificada, con respecto a la ecuación (3.26), por las resistencias de la misma columna (j) y por θ_j .

Este podría no ser el caso si fuese necesaria una extensión en el rango de valores de resistencias ya que a RB_i^p podría tener que sumarse una RS_i si la ecuación (3.44) lo requiriese. No obstante, esto proporciona una guía de diseño ya que, para disminuir el *crosstalk*, RS_i ha de ser lo más pequeña posible. También en (3.53) se puede observar como un incremento en los valores mínimos de las resistencias R_{ij} de la matriz origina una mayor disminución en los términos 2 y 4 del lado derecho de la ecuación con respecto al primero (ya que éstos tienen una dependencia cuadrática con las resistencias de la matriz) por lo que, si las resistencias son lo suficientemente grandes, se podrían eliminar estos términos y (3.53) volvería a quedar reducida a (3.58). Por otro lado, un aumento en el número de filas o columnas debido a los sumatorios que aparecen en (3.53) significa un incremento del *crosstalk* a través de los términos 2, 3 y 4 del miembro de la derecha de esta ecuación. Por tanto, incluso con valores mínimos elevados de resistencias en la matriz, si ésta tiene un gran número de filas y columnas el efecto de *crosstalk* puede ser el factor que más influya en la intensidad que atraviese una resistencia, más incluso que el propio valor de la resistencia.

3.3.4. Cálculo de R_{ij} teniendo en cuenta las tensiones de *offset* de los OAs

En este apartado se presenta un método simple para obtener los valores de R_{ij} , teniendo en cuenta todos los efectos de segundo orden presentados hasta ahora. Para aplicar este método hay que modificar el circuito de la Figura 3-9, añadiendo una segunda fila de calibración (estas filas serán llamadas $c1$ y $c2$), como aparece en la Figura 3-13.

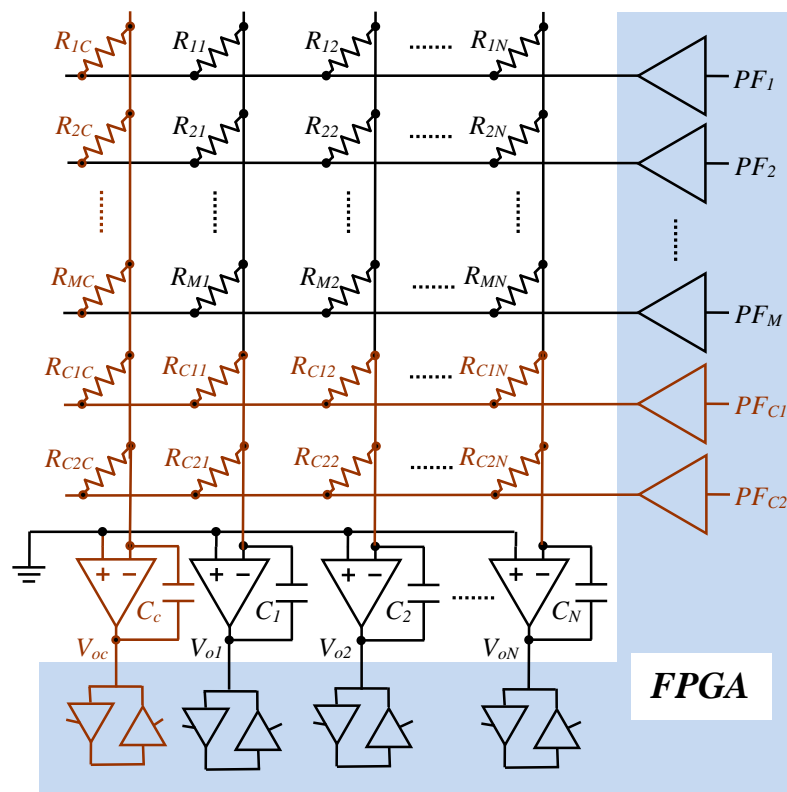


Figura 3-13. Modificación propuesta para la evaluar el efecto de las tensiones de *offset* y las intensidades de polarización de los OAs.

Nuevamente, el método se basa en la modificación del procedimiento de lectura de filas y en el uso de distintos tiempos de descarga con el objetivo de, en primer lugar, eliminar el término $G(j)$ de la ecuación (3.54) y, posteriormente, utilizar simples cocientes para encontrar los valores de las resistencias eliminando $F(i)$.

El proceso requiere los siguientes pasos:

- **Paso 1:** Se activan simultáneamente una fila i y la fila $c1$. El proceso se repite para cada una de las filas de la matriz. Así, se obtienen un conjunto de M tiempos $\Delta t'_{ij}$ empleados en descargar los distintos condensadores de las columnas cuando se activan simultáneamente las filas i y $c1$.
- **Paso 2:** Se activa la fila $c1$, obteniendo los tiempos $\Delta t_{c1,j}$.
- **Paso 3:** Se activan simultáneamente la fila $c1$ y $c2$, obteniéndose los tiempos $\Delta t'_{c2,j}$.

Cabe destacar que los pasos 2 y 3 se realizan una única vez durante el escaneo de todas las filas de la matriz y que el orden de realización de los tres pasos es indiferente. A continuación se muestra el proceso para conseguir R_{ij} a partir de los datos anteriores.

Siguiendo el mismo procedimiento que el utilizado para encontrar (3.51), en el Paso 1 se tiene

$$Ic_j^{i,c1} = I_{ij} + I_{c1j} - \sum_{g=1}^{M+1} I_{gj}^n + I_{ij}^n + I_{c1j}^n = \frac{C_j (V_{DD} - VT_j)}{\Delta t_{ij}'} \quad (3.59)$$

donde $Ic_j^{i,c1}$ indica la intensidad que descarga al condensador C_j activando simultáneamente las filas i y $c1$. Si a esto se le resta la intensidad Ic_j^i , obtenida durante el Paso 2

$$Ic_j^{c1} = I_{c1j} - \sum_{g=1}^{M+1} I_{gj}^n + I_{c1j}^n = \frac{C_j (V_{DD} - VT_j)}{\Delta t_{c1,j}} \quad (3.60)$$

se obtiene

$$Ic_j^{i,c1} - Ic_j^{c1} = I_{ij} + I_{ij}^n = C_j (V_{DD} - VT_j) \left(\frac{1}{\Delta t_{ij}'} - \frac{1}{\Delta t_{c1,j}} \right) \quad (3.61)$$

Teniendo en cuenta (3.51), (3.54) y (3.57) se puede escribir

$$I_{ij} + I_{ij}^n = \frac{F(i)}{R_{ij}} = C_j (V_{DD} - VT_j) \left(\frac{1}{\Delta t_{ij}'} - \frac{1}{\Delta t_{c1,j}} \right) \quad (3.62)$$

de igual modo, para la columna c

$$Ic_c^{i,c1} - Ic_c^{c1} = I_{ic} + I_{ic}^n = \frac{F(i)}{R_{ic}} = C_c (V_{DD} - VT_c) \left(\frac{1}{\Delta t_{ic}'} - \frac{1}{\Delta t_{c1,c}} \right) \quad (3.63)$$

A continuación, se repite el proceso con las intensidades de Paso 3, $Ic_j^{c2,c1}$, y Paso 2, Ic_j^{c1} , obteniendo 2 ecuaciones similares a (3.62) y (3.63)

$$I_{c2j} + I_{c2j}^n = \frac{F(c2)}{R_{c2j}} = C_j (V_{DD} - VT_j) \left(\frac{1}{\Delta t_{c2j}'} - \frac{1}{\Delta t_{c1,j}} \right) \quad (3.64)$$

$$I_{c2c} + I_{c2c}^n = \frac{F(c2)}{R_{c2c}} = C_c (V_{DD} - VT_c) \left(\frac{1}{\Delta t_{c2c}'} - \frac{1}{\Delta t_{c1,c}} \right) \quad (3.65)$$

Ahora, se dividen los miembros de la derecha de (3.62) y (3.64) por los miembros de la derecha de (3.63) y (3.65), procediendo de la misma manera para los miembros de la izquierda. Operando con estos cocientes finalmente

$$R_{ij} = \frac{R_{ic} R_{c2j}}{R_{c2c}} \cdot \frac{\frac{1}{\Delta t'_{ic}} - \frac{1}{\Delta t'_{c1,c}}}{\frac{1}{\Delta t'_{ij}} - \frac{1}{\Delta t'_{c1,j}}} \cdot \frac{\frac{1}{\Delta t'_{c2j}} - \frac{1}{\Delta t'_{c1,j}}}{\frac{1}{\Delta t'_{c2c}} - \frac{1}{\Delta t'_{c1,c}}} \quad (3.66)$$

que también se puede escribir

$$R_{ij} = \frac{R_{ic} R_{c2j}}{R_{c2c}} \cdot \frac{\Delta t'_{ij}}{\Delta t'_{ic}} \cdot \frac{\Delta t'_{c2c}}{\Delta t'_{c2j}} \cdot \frac{\Delta t'_{c1,c} - \Delta t'_{ic}}{\Delta t'_{c1,j} - \Delta t'_{ij}} \cdot \frac{\Delta t'_{c1,j} - \Delta t'_{c2j}}{\Delta t'_{c1,c} - \Delta t'_{c2c}} \quad (3.67)$$

Como las resistencias R_{ic} , R_{c2j} y R_{c2c} son conocidas y también los tiempos, se puede calcular nuevamente el valor de R_{ij} , sin tener en cuenta el valor de los condensadores, VT o de las resistencias de los *buffers*, RB . Se debe notar también que el hecho de activar simultáneamente las filas i y $c1$ permite la extensión de los rangos de resistencias de la matriz como se vio en el apartado 3.3.3.3.

Hay que destacar que, aunque se ha compensado la tensión de *offset* usando la ecuación (3.67), a los términos θ_j sería necesario sumar un término V_{oj} / A debido a la ganancia finita del OA. Sin embargo, como se observa posteriormente en el apartado 3.3.6, este término toma un valor muy inferior a la tensión de *offset*, por lo que no es tenido en cuenta.

3.3.5. Eliminación de los efectos de las intensidades de polarización de los OAs

Las ecuaciones (3.66) y (3.67) también tiene en cuenta los efectos de la intensidad de polarización que entraría al OA a través del terminal inversor, Ib_j . Efectivamente, si Ib_j es tenida en cuenta, (3.59) y (3.60) deben modificarse,

$$Ic_j^{i,c1} = I_{ij} + I_{c1j} - \sum_{g=1}^{M+1} I_{gj}^n + I_{ij}^n + I_{c1j}^n - Ib_j \quad (3.68)$$

$$Ic_j^{c1} = I_{c1j} - \sum_{g=1}^{M+1} I_{gj}^n + I_{c1j}^n - Ib_j \quad (3.69)$$

Como en las dos expresiones aparece de igual forma, la resta $Ic_j^{i,c1} - Ic_j^{c1}$ desaparece e igual sucede con las otras 3 restas de intensidades realizadas para obtener (3.66), por lo que esta ecuación sigue siendo válida, aun considerando las intensidades de polarización.

3.3.6. Materiales y método

Los circuitos propuestos en la Figura 3-9 y en la Figura 3-13 se han realizado sobre una FPGA Spartan3AN de Xilinx (XC3S50AN-4TQG144C) [Xilinx, Inc. 2014] con una frecuencia de trabajo de 50MHz. El contador empleado por los módulos de captura es de 14 bits, con una base de tiempo de 20 ns. Las tensiones de alimentación son de 1,2 V para el núcleo y de 3,3 V para las entradas/salidas.

Los OAs que se han utilizado son del modelo TLV2475N [Texas Instruments, Inc. 2007b] de Texas Instruments. Sus características principales son: tecnología CMOS con Entradas/Salidas Rail-To-Rail, modo *shutdown*, tensión de *offset* de 2400 μ V (máxima) y ganancia de 88dB (mínima). De estos parámetros se puede extraer que, como se comenta en el apartado 3.3.4, la tensión que aparece en el terminal inversor debido a la ganancia finita es, como mucho, de $3,3V/25119 = 0,13$ mV, 20 veces menor que la tensión de *offset*. La matriz de sensores consta de ocho filas y seis columnas. Dos filas y una columna se emplean para medir las resistencias de calibración. Sus valores, así como los de los condensadores de cada una de las columnas de la matriz, se indican en el siguiente apartado para cada uno de los experimentos realizados. Las resistencias *RB* de la FPGA se han caracterizado para establecer la relación entre *RL* y *RH* en los experimentos 2 y 3.

3.3.7. Resultados experimentales y discusión

Experimento 1:

Este experimento se realiza con el fin de analizar las prestaciones de la ecuación (3.30). Los resultados se muestran en la Tabla 3-4.

Tres resistencias (560 Ω , 5357,51 Ω y 10018,6 Ω) son medidas usando 5350 Ω como valor nominal para las resistencias de calibración. El resto de resistencias de la matriz toman el valor mínimo de 560 Ω , siendo ésta la peor situación en términos de *crosstalk* en la matriz a la hora de evaluar el valor de una resistencia. Los condensadores de cada

columna de la matriz tienen un valor nominal de 47 nF. Los resultados de \bar{R} y σ se han obtenido realizando 500 medidas, mientras que los errores de las columnas 4 y 5 de la Tabla 3-4 muestran el peor caso para las 500 medidas. El mismo procedimiento será empleado para el resto de experimentos de este apartado.

La resistencia máxima RH permitida según la ecuación (3.36) para una RL de 560 Ω es de 870 Ω , condición que no se cumple en ninguna de las dos resistencias distintas de RL . Efectivamente, se comprueba que los resultados para las dos últimas filas de la tabla presentan unos errores absolutos y relativos elevados. Lo mismo se puede observar en la columna que muestra el error sistemático ($|R - \bar{R}|$). Esto sucede incluso haciendo uso del pin Zero de la FPGA, como se indica en la sección 3.3.2.

Tabla 3-4. Exactitud y precisión para el experimento 1.

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
556	571,25	0,26	15,25	2,743	16,24	2,92
5357,51	5157,26	1,41	200,25	3,738	207,20	3,87
10018,6	9196,84	2,48	821,76	8,202	830,23	8,29

Experimento 2:

En este caso, se vuelve a usar la ecuación (3.30) pero para un rango de resistencias (3,3 k Ω – 10 k Ω) cuyo valor mínimo es bastante mayor que en el caso anterior. Para las resistencias de calibración se ha tomado 6,8 k Ω como valor nominal. Los condensadores de cada columna de la matriz siguen siendo los mismos que en el experimento 1. En este caso el valor RH según (3.36) es 5723,94 Ω , por lo que seguimos teniendo resistencias que no forman parte del rango óptimo de medida.

En la Tabla 3-5 se puede observar cómo tanto el error sistemático como los errores máximos se reducen con respecto al experimento 1. Esto confirma la discusión planteada en el apartado 3.3.4 sobre la ecuación (3.52), ya que en este experimento los términos 2 y 4 del miembro derecho de la ecuación se reducen proporcionalmente más que el término 1 al haber aumentado el valor mínimo de la resistencia de la matriz. Además, al ser la resistencia RL mucho mayor que la del experimento 1, el pin Zero de la FPGA consigue poner un valor más próximo a 0 V en la entrada inversora del OA.

Por lo tanto, con estos dos experimentos se muestra que la ecuación (3.30) sólo es aplicable en sensores resistivos donde $RL > 3 \text{ k}\Omega$.

Tabla 3-5. Exactitud y precisión para el experimento 2.

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
3295,6	3294,39	2,02	1,21	0,037	17,33	0,53
3486,3	3485,68	2,00	0,62	0,018	17,81	0,51
3882,6	3882,18	1,63	0,42	0,011	17,28	0,45
4628,4	4628,11	2,43	0,29	0,006	23,27	0,50
5072,8	5073,11	2,99	0,31	0,006	25,26	0,50
5621,1	5621,82	3,68	0,72	0,013	29,76	0,53
6166,1	6166,54	4,40	0,44	0,007	31,63	0,51
6789,1	6789,51	3,84	0,41	0,006	31,08	0,46
7151,2	7152,23	4,97	1,03	0,014	41,49	0,58
7463,7	7465,13	4,87	1,43	0,019	41,99	0,56
8169,2	8171,69	5,22	2,49	0,030	44,53	0,55
9054,6	9057,01	6,62	2,41	0,027	51,64	0,57
9974,7	9975,08	8,45	0,38	0,004	58,48	0,59

La Figura 3-14 muestra los errores relativos en la estimación de la resistencia R_{ij} para los experimentos 1 y 2. En la Figura 3-14(a) la resistencia máxima RH permitida según la ecuación (3.36) para una RL de 560Ω es de 870Ω . Se observa que para resistencias mayores de este valor el error relativo aumenta considerablemente. Por otro lado, la Figura 3-14(b) ilustra los errores para una resistencia RL de $3,3 \text{ k}\Omega$. En este caso, aunque según (3.36) RH es 5742Ω , se observa que para mayores valores de resistencia el error no se incrementa de forma significativa.

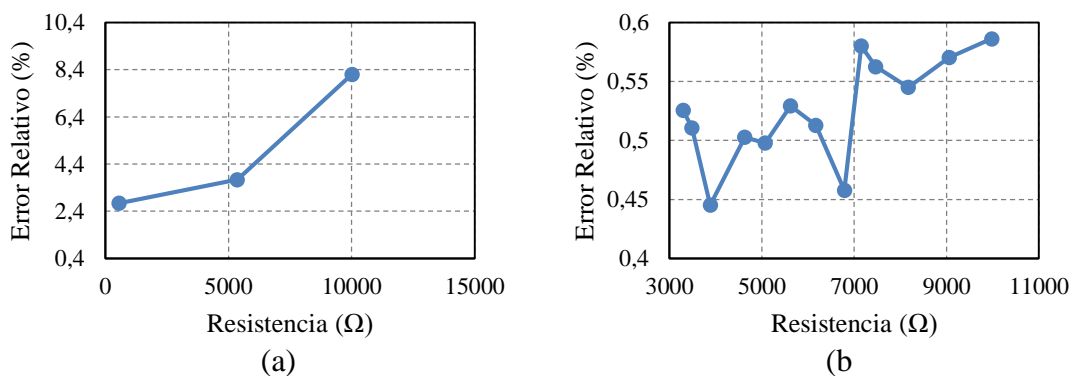


Figura 3-14. Errores relativos de R_{ij} calculados según la ecuación (3.30).

Experimento 3:

Como se observa en el experimento 1, para rangos de resistencia donde RL toma valores bajos los resultados no son los deseados. Por ello, en este experimento se implementa el circuito de la Figura 3-13, que permite el aumento del rango de resistencias modificando el direccionamiento de filas. En este caso son dos los métodos de aproximación que pueden usarse: la ecuación (3.46) que permite el aumento el rango de resistencias permitidas y la ecuación (3.67) que además de incrementar el rango elimina la influencia de la tensión de *offset* en la estimación del EBT.

El rango de resistencias empleado en este caso va de 560Ω a $3,3 \text{ k}\Omega$. Para las resistencias de calibración de la fila $c1$ y de la columna de calibración se ha elegido un valor de 750Ω . Para las resistencias de calibración de la fila $c2$ se ha tomado 990Ω como valor nominal. Los condensadores de cada columna de la matriz tienen un valor de 330 nF .

Como se advierte en la Tabla 3-6, los errores, tanto sistemáticos como relativos, usando la ecuación (3.46) son muy inferiores a los obtenidos en el experimento 1. Por otro lado, la resistencia máxima del rango empleado excede el valor máximo permitido, $RH = 2163 \Omega$, que se obtiene a partir de la ecuación (3.40). En la Tabla 3-6 se observa cómo, en los valores mayores de RH , los errores sistemáticos son muy superiores al resto, aunque inferiores a los obtenidos para las mismas resistencias del experimento 1.

Tabla 3-6. Exactitud y precisión para el experimento 3 usando la ecuación (3.46) para la estimación de R_{ij} .

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
556	555,08	0,38	0,92	0,165	2,52	0,45
678,5	677,81	0,52	0,69	0,102	2,67	0,39
747,2	745,85	0,60	1,35	0,181	3,81	0,51
864,2	863,61	0,84	0,59	0,068	3,91	0,45
1097	1096,46	1,01	0,52	0,047	4,33	0,39
1692,1	1692,95	1,90	0,85	0,050	9,90	0,58
2198	2194,15	3,26	3,85	0,175	18,95	0,86
2615,8	2611,20	4,38	4,40	0,168	28,10	1,07
3158,9	3151,54	5,96	7,36	0,233	25,16	0,80

La Tabla 3-7 utiliza los mismos datos experimentales que los empleados en la Tabla 3-6 pero evaluados según la ecuación (3.67). Se puede apreciar cómo el error sistemático es,

en este caso, menor al haber considerado los efectos de *offset*. Sin embargo, se observa que los máximos errores, aunque disminuyen, no lo hacen tanto como el error sistemático. Esto se debe a que la ecuación (3.67) emplea seis medidas de tiempo independientes, cada una con su error, para llevar a cabo la estimación, mientras que la ecuación (3.46), necesita únicamente cuatro. En este caso, para las resistencias superiores al valor de R_H no se observa un incremento en sus errores sistemáticos.

Tabla 3-7. Exactitud y precisión para el experimento 3 usando la ecuación (3.67) para la estimación de R_{ij} .

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
556	556,52	0,30	0,52	0,094	1,69	0,30
678,5	679,14	0,45	0,64	0,094	3,04	0,45
747,2	748,02	0,51	0,82	0,110	3,80	0,51
864,2	865,10	0,74	0,90	0,104	3,97	0,46
1097	1097,97	0,85	0,97	0,088	4,54	0,41
1692,1	1692,33	1,74	0,23	0,014	9,44	0,56
2198	2199,05	3,05	1,05	0,048	15,93	0,72
2615,8	2615,98	3,95	0,18	0,007	18,41	0,70
3158,9	3159,50	5,46	0,60	0,019	24,45	0,77

En la Figura 3-15 se muestran el máximo error relativo encontrado entre las 500 medidas usando las ecuaciones (3.46) y (3.67) para la estimación de R_{ij} , respectivamente.

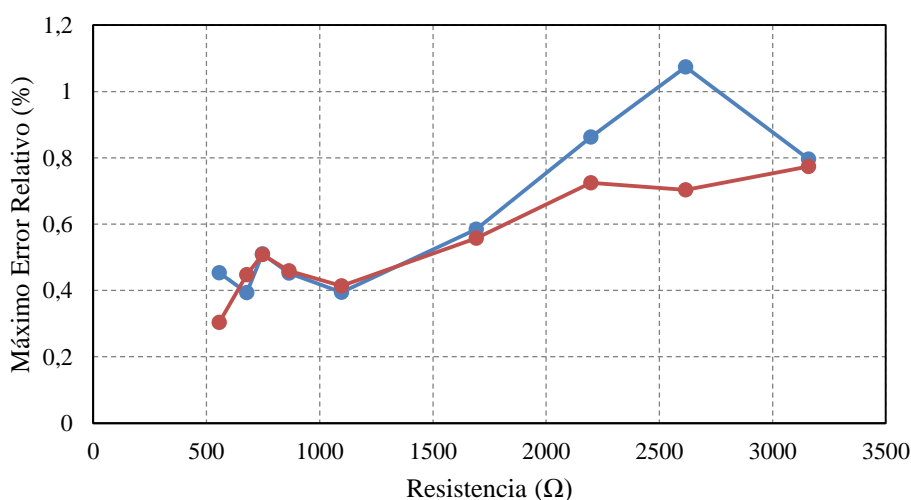


Figura 3-15. Máximos errores relativos de R_{ij} calculados según las ecuaciones (3.46) (azul) y (3.66) (rojo) con $R_L = 560 \Omega$, $R_C = 750 \Omega$ y $R_S = 100 \Omega$.

Experimento 4:

De nuevo, en este experimento se emplea el circuito de la Figura 3-13, y las ecuaciones (3.46) y (3.67) para el cálculo del valor de R_{ij} . El único cambio con respecto al experimento 3 es el rango de resistencias a estudiar, $3,3 \text{ k}\Omega - 10 \text{ k}\Omega$. Para las resistencias de calibración de la fila $c1$ y de la columna de calibración se ha elegido un valor de $10 \text{ k}\Omega$. Para las resistencias de calibración de la fila $c2$ se ha tomado $6,8 \text{ k}\Omega$ como valor nominal. Los condensadores de cada columna de la matriz tienen un valor de 47 nF .

Para este rango, la resistencia máxima, de nuevo, excede el valor máximo permitido, $RH = 7560 \Omega$, obtenido a partir de la ecuación (3.40). Sin embargo, en esta ocasión, usando la ecuación (3.46), como se aprecia en la Tabla 3-8, no se aprecian variaciones muy importantes en los errores sistemáticos y máximos para las resistencias que exceden RH , tal como cabría esperar al aumentar RL y por el uso del pin *Zero*.

Tabla 3-8. Exactitud y precisión para el experimento 4 usando la ecuación (3.46) para la estimación de R_{ij} .

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
3295,6	3278,86	1,89	16,74	0,508	25,88	0,79
3486,3	3468,56	1,97	17,74	0,509	31,41	0,90
3882,6	3863,51	2,07	19,09	0,492	30,52	0,79
4628,4	4605,27	2,59	23,13	0,500	45,03	0,97
5072,8	5046,77	3,47	26,03	0,513	44,81	0,88
5621,1	5592,11	3,44	28,99	0,516	49,21	0,88
6166,1	6132,44	3,92	33,66	0,546	53,75	0,87
6789,1	6751,06	4,34	38,04	0,560	66,99	0,99
7151,2	7114,69	4,71	36,51	0,511	70,85	0,99
7463,7	7427,34	6,22	36,36	0,487	89,98	1,21
8169,2	8125,95	7,29	43,25	0,529	86,61	1,06
9054,6	9010,28	8,88	44,32	0,489	98,50	1,09
9974,7	9924,17	9,80	50,53	0,507	110,99	1,11

Nuevamente, la Tabla 3-9 utiliza los mismos datos experimentales que los empleados en la Tabla 3-8 pero evaluados según (3.67). Se vuelve a observar la reducción significativa de los errores sistemáticos y máximos con respecto a los obtenidos por (3.46).

Tabla 3-9. Exactitud y precisión para el experimento 4 usando la ecuación (3.67) para la estimación de R_{ij} .

Resistencia (Ω)	\bar{R} (Ω)	σ (Ω)	$ R - \bar{R} $ (Ω)	$ R - \bar{R} /R$ (%)	Max. Error Absoluto (Ω)	Max. Error Relativo (%)
3295,6	3294,86	1,87	0,74	0,022	10,42	0,32
3486,3	3486,06	1,94	0,24	0,007	11,09	0,32
3882,6	3882,42	2,13	0,18	0,005	11,31	0,29
4628,4	4627,84	2,74	0,56	0,012	18,87	0,41
5072,8	5071,18	3,12	1,62	0,032	20,74	0,41
5621,1	5619,48	3,73	1,62	0,029	22,20	0,39
6166,1	6163,76	4,06	2,34	0,038	33,82	0,55
6789,1	6783,91	4,46	5,19	0,076	37,00	0,54
7151,2	7148,79	5,70	2,41	0,034	41,22	0,58
7463,7	7460,75	5,35	2,95	0,040	49,56	0,66
8169,2	8165,83	7,02	3,37	0,041	48,51	0,59
9054,6	9049,69	7,93	4,91	0,054	62,20	0,69
9974,7	9967,16	8,92	7,54	0,076	65,68	0,66

En la Figura 3-16 se muestran el máximo error relativo encontrado entre las 500 medidas usando las ecuaciones (3.46) y (3.67) para la estimación de R_{ij} , respectivamente.

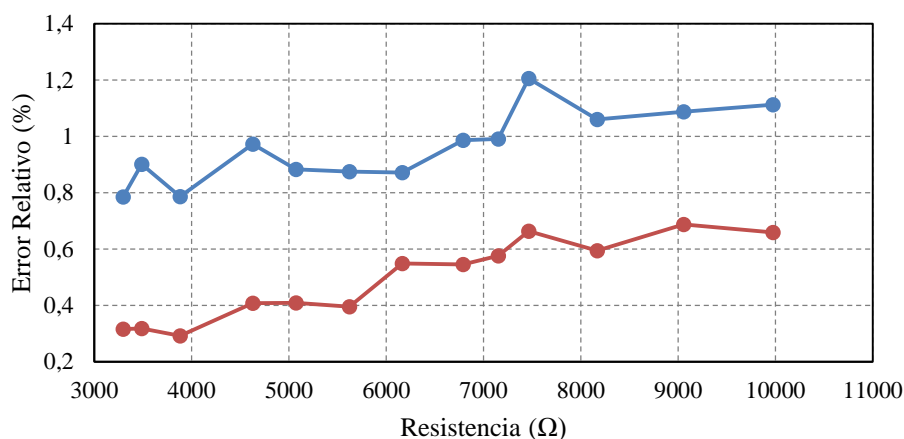


Figura 3-16. Máximos errores relativos de R_{ij} calculados según las ecuaciones (3.46) (azul) y (3.67) (rojo) con $R_L = 3,3 \text{ k}\Omega$, $R_C = 6,8 \text{ k}\Omega$ y $R_S = 100 \Omega$.

Por último, en la Figura 3-17 y en la Figura 3-18 se comparan los resultados para los experimentos 3 y 4. Para el rango de resistencias bajas, los errores son similares para ambos métodos, mientras se estiman valores por debajo de R_H . Sin embargo, en el rango de resistencias altas, la estimación a partir de (3.67) proporciona mejores resultados en cualquier caso.

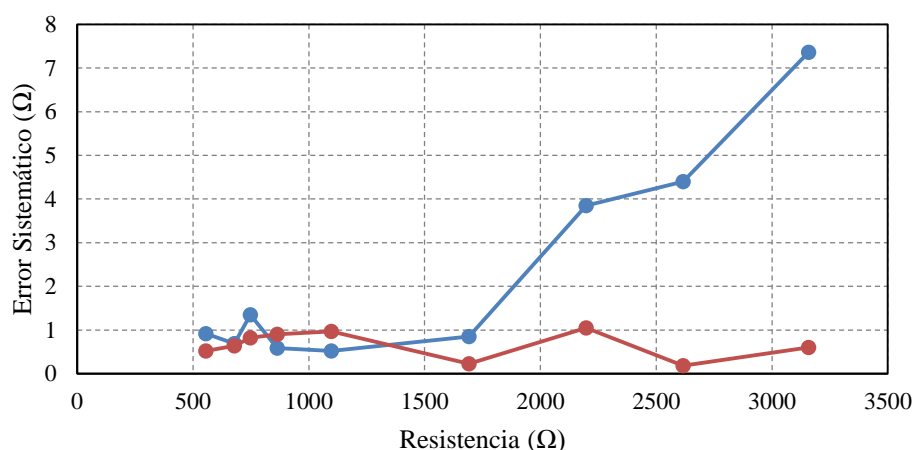


Figura 3-17. Errores sistemáticos de R_{ij} calculados según las ecuaciones (3.46) (azul) y (3.67) (rojo), con $RL = 560 \Omega$ $R_C = 750 \Omega$ y $RS = 100 \Omega$.

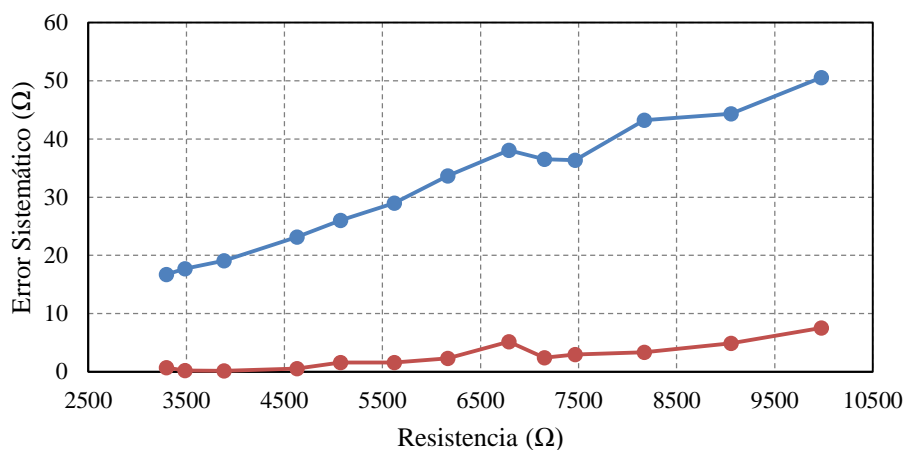


Figura 3-18. Errores sistemáticos de R_{ij} calculados según las ecuaciones (3.46) (azul) y (3.67) (rojo), con $RL = 3,3 \text{ k}\Omega$ $R_C = 6,8 \text{ k}\Omega$ y $RS = 100 \Omega$.

3.3.8. Conclusiones

En esta sección se ha estudiado una matriz de sensores resistivos con una distribución en M filas y N columnas. Esto permite el acceso, con un cierto grado de paralelismo (M lecturas simultáneas) a la información que proporcionan los sensores. También posibilita el uso de circuitos simples de acondicionamiento para la conversión analógica-digital. En este caso, se hace uso de un conjunto de M OAs con realimentación capacitiva.

Sin embargo, este circuito presenta una serie de limitaciones que reducen sus prestaciones. Entre las causas de estas limitaciones están: el valor mínimo de resistencia de los elementos de la matriz (que reduce el rango de resistencias a medir), la variabilidad

y dificultad en la medida de parámetros y elementos importantes del circuito (resistencias de los *buffers* de la FPGA, tensiones umbrales de estos *buffers*, condensadores usados en la realimentación de los OAs) y por último la tensión de *offset* y las corrientes de polarización de los propios OAs.

Para intentar paliar cada una de estas limitaciones se han propuesto distintas soluciones, llegando a un método de lectura de la matriz (sección 3.3.4) y a una ecuación (3.67) para calcular los valores de las resistencias que permiten evitar los errores debidos a las limitaciones expuestas anteriormente. Cabe también destacar que los errores introducidos por las resistencias de los *buffers* de la FPGA son eliminados mediante el uso de (3.30), (3.46) o (3.67).

Para comprobar la efectividad de los distintos circuitos y ecuaciones propuestos para estimar las resistencias de la matriz, se han realizado una serie de experimentos que nos confirman la exactitud de tales ecuaciones y los rangos de resistencias en el que algunas de ellas pueden aplicarse. Así, para una matriz con valores en el rango $[560 \Omega - 3,3 \Omega]$ se ha encontrado un error sistemático relativo máximo del 0,11% y un error relativo máximo del 0,77%. Por otro lado, se ha conseguido un error sistemático relativo máximo de 0,08% y un error relativo máximo del 0,69% para una matriz resistiva con valores en el rango $[3,3 \Omega - 10 \text{ k}\Omega]$.

Para futuros trabajos queda evaluar la influencia que sobre las prestaciones del circuito tiene la ganancia finita del OA. Se ha visto que, en el diseño aquí propuesto, ésta es mucho menor que la influencia debida la tensión de *offset*. Sin embargo, esto puede no ser así en otras implementaciones.

Capítulo 4. Propuesta de arquitectura para un sistema táctil

4.1. Introducción

En este capítulo se hace una propuesta de integración de las estrategias de adquisición de datos descritas en el capítulo 3 en una arquitectura para un conjunto de sensores táctiles de una mano robótica. El sensado táctil implica la gestión de grandes cantidades de datos, como ocurre en visión. Sin embargo, mientras que un robot humanoide comúnmente tiene solo dos cámaras, el sentido del tacto implica un área mayor, de manera que muchos sensores táctiles se distribuyen en manos y otras partes del cuerpo del robot. Por lo tanto, los buses de comunicación en el robot deben ser capaces de portar la gran cantidad de información proporcionada por los sensores táctiles, así como la proveniente de otros sensores y la dirigida a los actuadores. Un aspecto esencial en esta tarea, es que los datos deben ser transmitidos y procesados en un tiempo muy breve. Para un control realimentado estable y suave de robots humanoides se requiere un periodo de muestreo de 2-100 ms [Mukai et al. 2008]. También sabemos que la información registrada por la piel con respecto a la forma del área de contacto, a la fricción y a la dirección de las fuerzas en la yema del dedo, comienza a influir en la fuerza ejercida por la punta del dedo tras 70 ms aproximadamente en la manipulación humana, como se indica en [Flanagan et al. 2006]. Los requerimientos dinámicos son aún más estrictos para detección de eventos como el deslizamiento [Maldonado-Lopez et al. 2009]. Por tanto, una gran cantidad de datos se debe adquirir, procesar y transmitir en poco tiempo.

Para la comunicación en robots, se suelen usar buses serie porque ocupan poco espacio y crean pocas interferencias mecánicas y eléctricas. Por ejemplo, un bus serie I2C se usa en [Kageyama et al. 1999]. También se usa en la piel del iCub [Schmitz et al. 2011], aunque la razón de muestreo de cada táctel disminuye de 100 a 25 Hz conforme crece en número

de tácteles. [Ohmura et al. 2006] propone una red LAN en anillo donde los nodos son esclavos en la red LAN pero maestros en un bus secundario SMBus para direccionar los sensores táctiles. De esta forma se garantiza una latencia máxima y se reporta un tiempo de 51,2 ms para adquirir 1024 tácteles. Un inconveniente de esta arquitectura es que, como todos los nodos son esclavos, ninguno puede enviar una señal de advertencia como por ejemplo la producida por un deslizamiento inesperado. La mayor parte de las implementaciones usan arquitecturas abiertas con CAN bus [Cannata et al. 2006; Göger et al. 2006; Kaynov 2008], que garantiza una latencia máxima y es muy robusto frente a interferencias y fallos. Sin embargo, aunque es apropiado para tiempo real, este bus no es adecuado para transmitir grandes cantidades de datos. En concreto se necesitan 132 μ s para transferir 8 bytes de datos (máxima longitud del mensaje) a 1 Mbps bajo condiciones óptimas (si no se pierde el acceso al bus en favor de otro sensor o actuador). Por tanto, si la lectura de un táctel se codifica en una palabra digital de 8 bits, transmitir 1024 tácteles tomaría 16,9 ms en el mejor de los casos. Además, los datos no se pueden transferir al mismo tiempo, lo que implica cierta asincronía [Bednarski et al. 2008].

Por tanto, un pre-procesado básico se debe hacer en la electrónica local para permitir transmitir solo la información necesaria relativa a eventos críticos en el tiempo que necesiten una latencia garantizada por CANbus [Cannata et al. 2006; Schmid et al. 2008], y en todo caso se usa otro bus como LAN o USB para transmitir los datos sin pre-procesar para las tareas de alto nivel de control y organización, donde no se requiere una respuesta tan rápida [Göger et al. 2006; Kaynov 2008].

Si nos centramos en manos artificiales, aunque hay muchas implementaciones de conjuntos de sensores táctiles reportadas [Kappassov et al. 2015], solo unos pocos se comercializan o se están usando en manos artificiales en investigación sobre manipulación, como los que siguen. El conjunto de sensores de Pressure Profile Systems (PPS) RoboTouch™, para mano de Barrett, que proporciona 162 tácteles (24 palma + 3 x 24 + 3 x 22 falanges) a razón de 30Hz usando bus I2C [Pressure Profile Systems 2015]. El conjunto de sensores para la mano de Schunk, con 462 tácteles (6 matrices de 14 x 6 tácteles) a razón de 40 Hz por RS232 o bus CAN [Schunk GmbH & Co. KG 2015] (aunque cada módulo táctil es capaz de dar 230 fps (*frames per second*) [Weiss Robotics GmbH & Co. KG 2007, 2009]. El conjunto para la Shadow hand que proporciona la información a razón de 200Hz [Kappassov et al. 2015]. La mano Gifu Hand III (no comercial) usa

sensores de Tekscan y lee 640 tácteles (256 palma más 6 x 64 falanges) a razón de 100Hz [Mouri *et al.* 2002]. El robot iCub tiene 108 tácteles en cada mano (48 palma + 5 x 12) y en el peor caso da la información a razón de 25Hz [Schmitz *et al.* 2011].

La arquitectura que se describe en este capítulo consta de cuatro sensores táctiles, uno para cada dedo y otro para la palma de una mano de Barrett. Como se verá, la aproximación es competitiva con respecto a las descritas en términos número de tácteles, resolución espacial, y razón de muestreo. Hay que añadir además que la aproximación de electrónica local basada en FPGAs permite hacer un pre-procesado en paralelo. Esto se ilustra aquí implementando el pre-procesado que proporciona los momentos de la imagen táctil y los parámetros de la elipse que la representa y se deriva de los primeros, dando información del tamaño, forma y orientación del contacto, así como de la fuerza del contacto. Esta información se proporciona en la implementación que se presenta con la misma razón que las imágenes táctiles completas. Finalmente, dada la capacidad de procesamiento paralelo de las FPGAs, esta aproximación mostrará tanto mayor ventaja cuanto mayor sea el pre-procesado, que puede incluir auto-calibrado, linealización o procesados más complejos como detección de formas, determinación de parámetros de agarre, de rigidez o texturas de objetos, de deslizamiento, etc.

4.2. Diagrama de bloques

El sistema táctil propuesto en este trabajo posee una estructura modular, fácilmente escalable, que permite llevar a cabo distintas realizaciones desde el punto de vista del número de sensores a emplear y de la naturaleza de los mismos. Este enfoque permite realizar implementaciones concretas para manos robóticas con distinto número de dedos, falanges, etc., de forma fácil y sin apenas cambios. La Figura 4-1 muestra el diagrama de bloques de la arquitectura propuesta. El núcleo del mismo es un módulo de control que solicita y recibe la información táctil desde un conjunto de sensores inteligentes como paso previo a su envío a un sistema *host*.

Las comunicaciones entre el módulo de control y los sensores táctiles se llevan a cabo por medio de varios puertos, empleando un protocolo serie SPI (*Serial Peripheral Interface*) [Motorola, Inc. 2003], por lo que cada uno de estos puertos puede conectar el módulo controlador con varios módulos sensores.

El envío de la información táctil del conjunto del sistema hacia un sistema *host* se puede realizar a través de distintos protocolos de comunicación (RS-232, CAN-Bus, USB 2.0) [Axelson 2007; Lawrenz 2007; Axelson 2009]. El uso de uno u otro dependerá de los buses disponibles en el sistema *host*, así como de la naturaleza de la información se quiera transmitir. Para imágenes táctiles no procesadas y tasas de muestreo elevadas se hace necesario el empleo de protocolos de alta velocidad, mientras que para sistemas que sólo necesiten de alguna característica extraída por los propios sensores táctiles, protocolos de baja velocidad y baja latencia pueden ser suficientes.

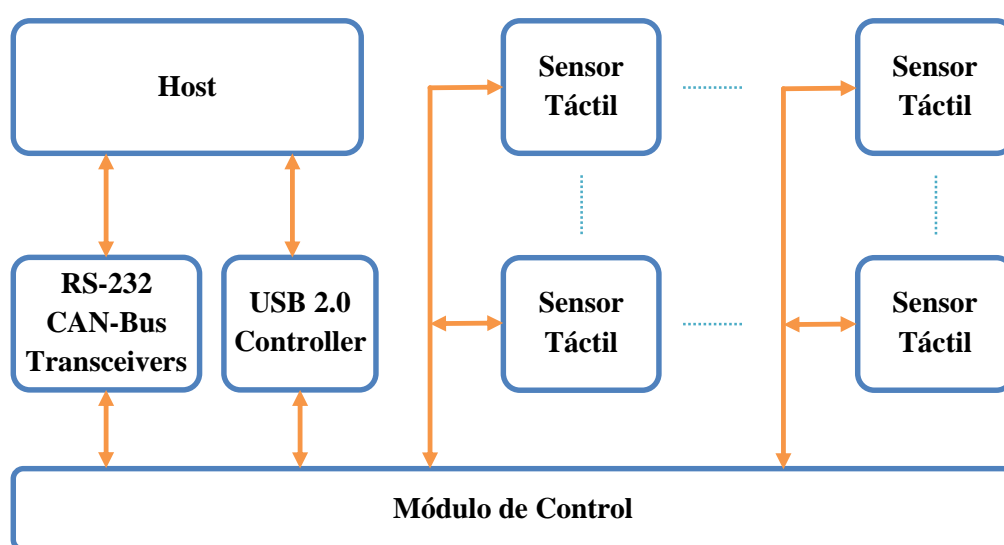


Figura 4-1. Diagrama de bloques de la arquitectura del sistema.

A continuación se describen cada uno de los bloques de la arquitectura del sistema.

4.3. Módulo de control

El módulo de control es el núcleo del sistema táctil y, como principal tarea, coordina las comunicaciones entre los distintos sensores táctiles inteligentes que conforman el sistema y el sistema *host* que recibe dicha información. Sus

4.3.1. Comunicaciones entre el módulo de control y los sensores táctiles

Como se explica con anterioridad, las comunicaciones entre este módulo y los sensores táctiles se llevan a cabo por medio de comunicaciones serie, adoptando para ello un protocolo SPI. Este protocolo permite comunicaciones serie síncronas entre el módulo de

control y varios sensores táctiles con sólo añadir una línea de selección por cada uno de ellos. Es un protocolo Maestro-Esclavo donde, en este caso, el módulo de control hace de maestro mientras cada uno de los sensores conectados al bus son los esclavos. Las señales comunes a todos los dispositivos conectados al bus son:

- La señal de reloj con la que se sincronizan las comunicaciones (SCLK - *Serial Clock*).
- Los datos provenientes del módulo de control (MOSI – *Master Output Slave Input*).
- Los datos provenientes de los sensores (MISO – *Master Input Slave Output*).
- La señal de selección del sensor con el que se establece la conexión (SS – *Slave Select*).

Las ventajas de este protocolo son:

- Comunicaciones *Full-Duplex*: el dispositivo maestro puede enviar datos a la vez que los recibe desde el esclavo y viceversa.
- Implementación hardware extremadamente simple.
- Protocolo flexible: el tamaño de la trama de datos a enviar/recibir, su significado y su propósito no está impuesto por el protocolo.
- Uso de pocas señales para comunicaciones con un número considerable de dispositivos: las señales de datos y reloj son compartidas entre los dispositivos conectados al bus, necesitándose sólo una señal de selección adicional por cada uno de ellos.
- Al tratarse de comunicaciones entre circuitos integrados que se encuentran relativamente cerca unos de otros, no se hace necesario el uso de circuitería externa en forma de *transceivers* o *buffers* para el envío/recepción de datos.

Por otro lado, las principales desventajas que presenta el protocolo SPI son:

- Ancho de banda limitado al tratarse de un protocolo serie.
- Sólo permite comunicaciones con un sensor a la vez.

Como se explica en capítulos anteriores, los sensores táctiles obtienen datos de alta resolución en la medida de fuerza trabajando con frecuencias de muestreo elevadas (por encima de los 200 *frames* por segundo). Esto hace que las tramas de datos de las imágenes

táctiles obtenidas por los distintos sensores contengan gran cantidad de datos y se necesiten unas comunicaciones de alta velocidad entre éstos últimos y el módulo de control.

Por todo lo anterior, el sistema táctil propuesto emplea el protocolo SPI para conectar el módulo de control con varios sensores táctiles, pero hace uso de tantos controladores SPI en paralelo como sean necesarios para alcanzar el ancho de banda requerido.

Tomando como demostrador de un sistema táctil a una mano robótica, una primera aproximación nos lleva a considerar el uso de un controlador SPI por cada dedo-palma de la mano, con distintos sensores por dedo compartiendo bus (un sensor por falange).

En la implementación concreta del protocolo SPI realizada en el sistema, los datos enviados por el módulo de control y los distintos sensores son los siguientes:

- Módulo de control: envía hacia los distintos sensores 3 bytes que son almacenados en otros tantos registros de control. Los datos se definen de la siguiente forma:
 - Registro de operación: habilita/deshabilita el sensor y las distintas rutinas de pre-procesado que pueda desarrollar.
 - Registro de táctel inicio: define cuál es la fila y la columna de la esquina superior izquierda de la matriz táctil.
 - Registro de táctel final: define cuál es la fila y la columna de la esquina inferior derecha de la matriz táctil.

Estos parámetros se envían de forma paralela a la recepción de los datos por parte de los sensores, modificando los registros de control cada vez que se direccionan y posibilitando el cambio del modo de operación del sensor (encendido/apagado del sensor, habilitación/deshabilitación del pre-procesado de imágenes táctiles, cambio del tamaño de la matriz táctil).

- Sensores: envían hacia el módulo de control los datos relativos a la información táctil. En concreto:
 - La marca de tiempo (*timestamp*) de la imagen táctil obtenida: es un byte con el valor de un contador de 8 bits que se incrementa cada vez que el sensor muestrea una imagen táctil. Conociendo el tiempo empleado en el muestreo de la matriz táctil completa y el valor de este contador, se puede

inferir el tiempo transcurrido entre las distintas imágenes táctiles obtenidas.

- La temperatura que registra el sensor cada vez que se toma una muestra: debido a las variaciones que las resistencias pueden tener con la temperatura, cada sensor táctil tiene la posibilidad de incluir un sensor de temperatura. Se reservan dos bytes para enviar dicha información hacia el módulo de control.
- La imagen táctil: se usan cuatro bytes para codificar la información de cada táctel. En los bytes más significativos (bytes 3 y 2) se envía la información expresada en una magnitud distinta del tiempo (fuerza medida, resistencia medida, magnitud adimensional), mientras que los otros dos bytes restantes (bytes 1 y 0) contienen el resultado de la cuenta del módulo de captura que realiza la conversión tiempo-digital. El orden que se sigue para el envío de los distintos tácteles que conforman la imagen es por columnas de izquierda a derecha hasta completar una fila (la de arriba), para continuar con la siguiente fila hasta llegar a la última, la de más abajo). El táctel de la coordenada (0,0) se corresponde con la esquina superior izquierda de la imagen y el (M,N) con la esquina inferior derecha, siendo M el número de filas y N el número de columnas de la matriz.
- Los parámetros que se hayan podido extraer de las imágenes táctiles por medio de las rutinas de pre-procesado: el número de datos enviados dependerá del algoritmo/os de pre-procesado que se implementen.

El tamaño total de la trama de datos, por tanto, depende de la configuración concreta que se haya aplicado al sensor (tamaño de la imagen y número de resultados del algoritmo de pre-procesado empleado).

Dado que el módulo de control es el que hace de maestro en el protocolo SPI, las comunicaciones y, por tanto, el número de imágenes táctiles por segundo que se reciben desde los sensores dependerá de las que sean solicitadas por dicho módulo. Los sensores, siempre que estén habilitados, estarán obteniendo y procesando imágenes táctiles aunque éstas no sean enviadas posteriormente.

La Figura 4-2 muestra el cronograma de las comunicaciones entre el módulo de control y un sensor táctil.

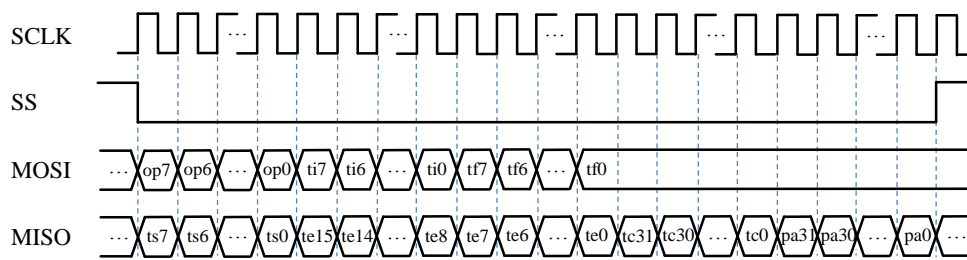


Figura 4-2. Cronograma del protocolo SPI entre el módulo de control y un sensor.

4.3.2. Comunicaciones entre el módulo de control y el sistema *host*

Las comunicaciones entre el módulo de control y el sistema *host* son gobernadas por este último. De la misma forma que los sensores continúan adquiriendo imágenes táctiles aunque el módulo de control no se las solicite, el módulo de control sigue reclamando información táctil a los sensores aunque el sistema *host* no se la requiera. De esta forma, el tiempo de respuesta del sistema ante una petición de una imagen de todo el sistema táctil es mínimo.

Independientemente del protocolo de comunicación empleado (RS-232, CAN-Bus o USB) las comunicaciones están basadas en el envío y recepción de palabras de 8 bits (1 *byte*). Así, el sistema *host* envía un conjunto de *bytes* con la configuración global del sistema cada vez que se solicitan datos (modo de operación y tamaño de la matriz de cada uno de los sensores), y el módulo de control los almacena en unos registros para posteriormente enviarlos a los distintos sensores a través de los puertos SPI.

El uso de un protocolo u otro entre el sistema *host* y el sistema táctil va a estar condicionado por distintos aspectos. Cuando el número de sensores que conformen el sistema sea elevado y los datos requeridos no estén pre-procesados (la información se envía en bruto) los protocolos RS-232 y CAN-Bus no poseen el ancho de banda suficiente para poder ser utilizados. En ese caso, el puerto USB 2.0 sería la opción más indicada. Por otro lado, cuando el número de sensores sea pequeño o no se necesite toda la información sobre la imagen táctil y baste con los parámetros obtenidos por los algoritmos de pre-procesado, cualquiera de los protocolos podría ser empleado.

El mecanismo empleado por el módulo de control para mantener la integridad entre los datos enviados al sistema *host* y los que se reciben de los diferentes sensores es el uso de un doble *buffer* de memoria. Mientras al módulo de control no le llegue una solicitud de

envío por parte del sistema *host*, éste se encuentra recibiendo imágenes táctiles del conjunto de sensores del sistema y almacenándolos de forma alterna en las dos zonas de memoria destinadas a tal efecto. En el momento que el módulo de control recibe una petición por parte del sistema *host*, comienza el envío de la imagen del sistema táctil que se hubiese completado más recientemente. Para poder desacoplar las distintas comunicaciones, el módulo de control no deja de solicitar información de los sensores pero, en este caso, sólo la almacenaría en la zona de memoria que no está siendo enviada al sistema *host*. Este sistema de doble *buffer* también posibilitaría la realización de un pre-procesado a nivel de sistema haciendo uso de las distintas imágenes táctiles o de los datos pre-procesados de forma local en cada uno de los sensores.

Los controladores para los protocolos RS-232 y CAN-Bus forman parte del módulo de control, por lo que sólo se necesitan los *transceivers* de cada uno de los protocolos para completar las comunicaciones con el sistema *host*.

Por otro lado, para las comunicaciones a través del puerto USB se emplea un controlador USB 2.0. El controlador escogido puede ser configurado para trabajar en distintos modos de funcionamiento, pero alcanza un mayor ancho de banda si es configurado en modo “*bridge*” USB/FIFO (*First In First Out*) síncrona, con tasas de transferencia de hasta 40 Mbytes por segundo. El módulo de control recibe datos desde el *host* a través de una FIFO de entrada y envía los datos del sistema táctil por otra FIFO de salida. El ancho de palabra de los datos de las FIFOs es de 8 bits.

4.4. Sensor Táctil

Los sensores táctiles empleados en el sistema están basados en los circuitos descritos en el capítulo 3. Para los sensores que trabajen sobre matrices táctiles de tamaño pequeño o medio, el uso del integrador pasivo (apartado 3.2) parece el más razonable. Las FPGAs poseen una gran cantidad de entradas/salidas, necesarias en el direccionamiento empleado en este enfoque, donde cada táctel emplea una conexión dedicada con la FPGA para la medida de su resistencia asociada. Por ello, los sensores táctiles de los dedos utilizan esta propuesta por poseer un área más reducida comparada con el sensor de la palma.

Por otro lado, para matrices táctiles de tamaño medio-grande, el integrador activo (apartado 3.3.2) aporta la mejor solución debido al menor número de conexiones que se

precisan para el direccionamiento de la matriz. Por ello, este enfoque es el escogido para el sensor de la palma.

4.4.1. Sensor para el dedo

El sensor empleado para el dedo requiere de un diseño mecánico más compacto que el que se emplea para la palma debido a las dimensiones reducidas que presenta. Por ello, el número de componentes y el tamaño de los mismos son críticos en este caso. La arquitectura propuesta para este sensor se muestra en la Figura 4-3, y consta de los siguientes elementos:

- **FPGA:** es la unidad central de proceso del sensor y se encarga del muestreo, pre-procesado y de las comunicaciones con el módulo de control.
- **Circuito de alimentación:** la FPGA necesita dos tensiones de alimentación (3,3 V y 1,2 V) para los bloques de entrada/salida y el núcleo, respectivamente.
- **Circuito de reloj:** es el reloj con el que opera el circuito.
- **EEPROM:** memoria no volátil donde se almacena el firmware a ejecutar por la FPGA.
- **Puertos I/O:** incluyen las señales del protocolo SPI entre el sensor y el módulo de control así como las del protocolo JTAG (*Joint Test Action Group*) para reconfigurar la FPGA o reprogramar la EEPROM.
- **Matriz táctil:** matriz de electrodos sobre la que se coloca el material piezorresistivo.

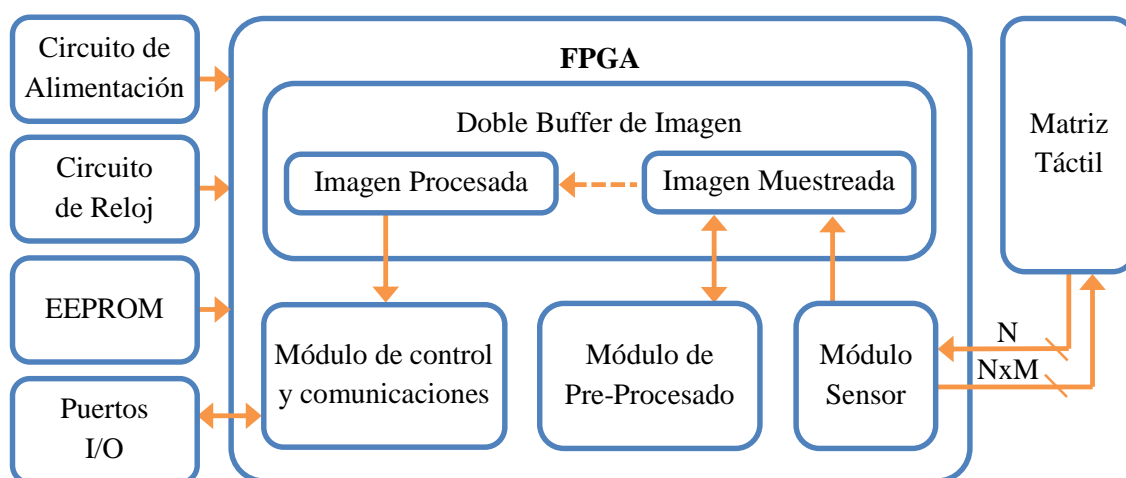


Figura 4-3. Diagrama de bloques del sensor del dedo.

4.4.1.1. FPGA

La mayoría de los sensores táctiles inteligentes actuales están basados en un chip microcontrolador que, junto con un circuito de conversión analógico digital, captura, procesa y envía imágenes táctiles hacia un sistema *host*. Estos circuitos presentan un ancho de banda limitado en la adquisición de datos y poseen limitaciones en lo que al pre-procesado y envío de datos se refiere, debido a la naturaleza secuencial de los algoritmos que pueden ser realizados por el microcontrolador. El trabajo aquí propuesto es el primero en realizar una conexión directa entre una matriz táctil y una FPGA, aprovechando el paralelismo de estos dispositivos tanto para la adquisición de datos como para la realización de distintos algoritmos de pre-procesado.

Las FPGAs son dispositivos muy versátiles donde, cualquier circuito combinacional o secuencial, puede ser implementado siempre que los recursos hardware que posean se lo permitan, llegando incluso a poder instanciar varios microcontroladores en un mismo chip. Además, permiten definir circuitos a bajo nivel, por medio de esquemáticos o algún lenguaje de descripción hardware, para las tareas donde se requiera mucho control, sobre todo desde el punto de vista de las restricciones temporales que se deban cumplir; y por otro lado instanciar microcontroladores a medida, con sólo los recursos que se van a necesitar, para llevar a cabo tareas más complejas de procesado y control.

En nuestro caso, todos los módulos funcionales que se incluyen en la FPGA han sido definidos mediante un lenguaje de descripción hardware, concretamente VHDL (*VHSIC Hardware Description Language*), para conseguir una implementación óptima en lo relativo al uso de recursos, velocidad de proceso y consumo de energía.

Los módulos son los siguientes:

- **Módulo sensor:** es el módulo de captura que lleva a cabo la conversión tiempo-digital a partir de una señal modulada por el sensor de cada táctel. En el capítulo 2 se detallan los circuitos propuestos para conseguir una medida de tiempo con la menor incertidumbre posible sin la inclusión de circuitería externa, propiciando con ello una conexión directa entre sensor y FPGA. En el caso del sensor del dedo, este módulo sigue el esquema del circuito integrador pasivo propuesto en el apartado 3.2, por lo que se necesitan tantas señales como columnas (N) tenga la matriz táctil para medir los tiempos, así como tantas señales como tácteles ($M \cdot N$)

tenga la matriz para poder direccionarlos por filas evitando con esto el *crosstalk* resistivo entre táctels de una misma fila.

Cada vez que finaliza el muestreo en paralelo de todos los sensores de una fila, el módulo sensor escribe los datos obtenidos sobre la zona de memoria “imagen muestreada” del doble *buffer*. La imagen táctil se completa cuando la última fila es medida.

- Doble *buffer* de imagen: de forma análoga a como se hace en el módulo de control del sistema táctil, cada uno de los sensores posee un doble *buffer* de imágenes táctiles. Esto permite mantener la integridad de las imágenes enviadas desde el sensor, desacoplando por tanto los procesos de envío y muestreo. Este *buffer* de memoria está implementado usando memorias RAM de doble puerto que permiten acceso paralelo a los datos con distintas frecuencias de reloj para cada uno de ellos. Esto permite que el módulo de comunicaciones, que implementa un controlador SPI esclavo, haga uso del reloj SCLK del bus tanto para la lógica de control como para el acceso a los datos táctiles almacenados en memoria. Por otro lado, el módulo sensor y el de pre-procesado pueden usar la frecuencia de reloj que precisen para cumplir los requisitos temporales que deban satisfacer (tasa de muestreo, velocidad de pre-procesado, etc.).

El procedimiento que se sigue para el arbitraje del doble *buffer* de memoria es el siguiente: una vez que una imagen táctil completa es escrita en la zona de memoria “imagen muestreada” por el módulo sensor y el módulo de pre-procesado ha concluido los algoritmos que implemente a partir de ella, la imagen táctil muestreada y pre-procesada pasa a ser la “imagen procesada”. El mecanismo es tan simple como cambiar un *flag* que indica cuál de las dos zonas de memoria es la que contiene la información muestreada o procesada. Si en el instante que los módulos sensor y de pre-procesado concluyen con la imagen muestreada en curso se estuviese accediendo a la imagen procesada por parte del módulo de comunicaciones, el *flag* que determina el tipo de imagen que contiene cada *buffer* no cambiaría reescribiéndose la zona de memoria asignada a la imagen muestreada con una nueva imagen táctil, perdiéndose de esta forma una de las imágenes capturadas. Esto se hace para evitar que la imagen enviada al módulo de control del sistema táctil por parte cada sensor pueda contener datos de dos imágenes táctiles distintas, desacoplando con ello las frecuencias de trabajo de los

módulos propios del sensor con el de comunicaciones, controlado por el sistema táctil. El envío de la trama completa de datos desde el sensor al módulo de control no tiene por qué llevar más tiempo que el empleado por todo el proceso de muestreo y pre-procesado de una imagen táctil completa, aunque existan varios sensores compartiendo el mismo bus SPI. Así, es tarea del módulo de control del sistema táctil el requerir la información de cada sensor de forma que no se pierdan imágenes debido al comportamiento anteriormente citado. Conociendo el número de imágenes por segundo que son muestreadas en cada sensor, es fácil calcular cuál debe ser la tasa de transferencia que se precisa en el bus SPI para que no se pierda información táctil alguna.

- Módulo de pre-procesado: este módulo es el que convierte en “inteligente” al sensor puesto que, a partir de unos datos obtenidos en bruto, es capaz de tratar la imagen con distintos algoritmos. Éstos pueden ir desde el compensado de la histéresis del sensor hasta la extracción de determinados parámetros de la imagen como podrían ser sus momentos geométricos y la elipse de la imagen táctil, descargando de este tipo de cálculos al sistema *host*. De hecho, para determinadas aplicaciones, puede que estos parámetros sean suficientes como única respuesta del sistema táctil, pudiéndose utilizar, por tanto, protocolos de baja velocidad entre el sistema completo y el *host*. En un apartado posterior se describe el algoritmo de pre-procesado empleado en este trabajo.
- Módulo de control y comunicaciones: incluye al controlador SPI esclavo y a toda la lógica de control del sensor. Como se ha indicado anteriormente, el controlador SPI trabaja con el reloj del bus SPI y se encarga de enviar el *timestamp*, la imagen táctil procesada y, opcionalmente, la temperatura medida en el sensor. Las comunicaciones se llevan a cabo de forma ininterrumpida, por lo que el sensor responde con un bit por ciclo a las peticiones por parte del dispositivo maestro. La trama de datos, por tanto, se envía completa bit a bit sin ningún tipo de interrupción.

En cuanto a la lógica de control, se encarga de habilitar/deshabilitar módulos, definir el tamaño de la imagen táctil a capturar y de coordinar todos los bloques funcionales entre sí. Contiene unos registros que son actualizados por los tres bytes que envía el módulo de control del sistema cada vez que se solicita un nuevo dato.

4.4.1.2. Circuito de Alimentación

Como se comentó con anterioridad, las FPGAs actuales son capaces de sustituir a sistemas tan complejos como microprocesadores o DSPs (*Digital Signal Processor*) sin perder la capacidad de implementar módulos combinacionales o secuenciales dedicados operando todos en paralelo. Esto hace que los requerimientos energéticos del dispositivo puedan ser importantes a la vez que cambiantes dependiendo del programa que lleve a cabo en cada momento. Por esto, el circuito de alimentación debe ser diseñado de forma que el correcto funcionamiento de la FPGA está asegurado bajo cualquier condición de trabajo.

Normalmente, las FPGAs necesitan de más de una tensión de trabajo. Las alimentaciones de los distintos bancos de entrada/salida pueden tomar distintos voltajes de forma que una única FPGA pueda conectarse con distintos dispositivos externos aunque éstos trabajen con distintos niveles de tensión. Esto las hace unos dispositivos extremadamente versátiles en ese sentido. Por otro lado, el núcleo de la FPGA, donde todas las funciones lógicas se llevan a cabo, emplea una tensión de alimentación baja para minimizar la potencia consumida por el dispositivo. Estas tensiones son cada vez más reducidas, propiciadas por los avances en la tecnología de fabricación (0,72 V para las Virtex Ultrascale+ fabricadas con una tecnología de 16 nm) [Xilinx, Inc. 2015b].

Otro factor a tener en cuenta es el área ocupada por el circuito de alimentación. Al tratarse de sensores táctiles, las dimensiones del conjunto del circuito deben ser lo más contenidas posible y el circuito de alimentación puede ser el causante de un incremento importante en este sentido, sobre todo cuando más de una tensión de trabajo es requerida. Existen soluciones en el mercado basadas en un único circuito integrado con distintas tensiones de salida. Otro aspecto a valorar es el número de componentes discretos externos que necesitan para su funcionamiento, así como el tamaño de los mismos.

En el caso del circuito de alimentación propuesto para los sensores de este trabajo, la alimentación del núcleo de la FPGA opera con una tensión de 1,2 V, mientras que el voltaje para los bancos de entrada/salida es de 3,3 V. Hay que tener en cuenta que, debido a la conexión directa que se realiza entre la FPGA y el sensor, los condensadores que permiten modular la medida de resistencia del sensor piezorresistivo a través del tiempo de descarga son cargados a través de la propia FPGA, por lo que el valor de tensión que

alcance el condensador al final de la fase de carga condiciona la velocidad de cambio de la señal de descarga (*slew rate*) y, por consiguiente, la velocidad de la medida.

El aspecto más importante con respecto al circuito de alimentación en la implementación que aquí se presenta tiene que ver con el ruido de salida de las distintas tensiones generadas. La mayoría de fabricantes de circuitos de alimentación publican notas de aplicación que ayudan en la elección de los circuitos de alimentación a emplear para cada fabricante y modelo de FPGAs pero sólo tiene en cuenta que, en lo relativo al ruido, no se superen unos umbrales que, dada la naturaleza digital del dispositivo, son considerablemente altos. En el caso que nos ocupa, y como se ha demostrado con anterioridad en el capítulo 2, la mayor fuente de incertidumbre en la medida de tiempo, y por tanto de la magnitud física que modula el sensor es el ruido asociado a la alimentación de la FPGA, que afecta a los umbrales de los *buffers* de entrada así como al ruido presente en la señal de descarga a medir. Por esto, los reguladores empleados en este circuito, además de proporcionar la corriente necesaria para el correcto funcionamiento del sensor para cargas de trabajo altas, deben poseer un ruido a su salida lo más bajo posible, ya que debe considerarse como un circuito de tipo mixto. Estos reguladores de bajo ruido (*ultralow noise regulators*) tienen un ruido de salida de unos pocos micro voltios V_{RMS} . Otro parámetro a tener en cuenta es la caída de tensión en la salida para condiciones de carga alta (*dropout voltage*). La caída debe ser lo más pequeña posible para no afectar a los niveles de referencia y de tensión de partida en la descarga de los condensadores, de los cuales depende el resultado de la medida de resistencia.

Después de todo lo expuesto, cabe añadir que el diseño de la PCB (*Printed Circuit Board*) influye de manera decisiva en lo que al ruido de alimentación del sistema se refiere. Un mal diseño de la PCB puede afectar al circuito de alimentación en lo que a ruido se refiere.

4.4.1.3. Circuito de Reloj

El circuito de reloj está formado por un único oscilador que proporciona la frecuencia de trabajo de la FPGA y por tanto del sensor. Este reloj puede ser usado en el interior de la FPGA para obtener otras frecuencias que fuesen necesarias a través de los módulos DCM (*Digital Clock Manager*) que permiten sintetizar cualquier frecuencia a partir de otra de

partida. En este caso, la frecuencia de trabajo elegida para el oscilador es de 50MHz [*FOX Electronics 2008*].

4.4.1.4. EEPROM

Una de las características más potente de las FPGAs es la que permite que puedan ser reconfiguradas en cualquier instante, modificando con ello su comportamiento y adaptándolo a distintas circunstancias. Esto provoca que las funciones que llevan a cabo deban ser almacenadas en una memoria volátil, teniendo que ser configurados cada vez que el sistema se inicia.

Existen distintas formas de configurar una FPGA pero la más habitual, una vez que se ha completado el proceso de depuración del firmware con las funciones a desarrollar, es usar una memoria no volátil donde guardar dicho firmware y que la FPGA se configure a partir de la información almacenada en ella cada vez que el sistema es iniciado. Es posible, incluso, almacenar más de un programa para que la FPGA pueda reconfigurarse a sí misma según convenga con un programa u otro.

Existen distintos tipos de memorias no volátiles desde donde una FPGA es capaz de configurarse (protocolos serie o paralelo) y una característica que deben cumplir es que el tamaño de la memoria tiene que ser suficiente para poder contener, al menos, el archivo de configuración con el programa a ejecutar en la misma. El tamaño de este archivo puede cambiar en función del número de recursos que la FPGA posea, así como de los que sean empleados para una aplicación concreta.

Si el archivo de configuración es de un tamaño inferior a la capacidad de la memoria, las zonas no ocupadas pueden ser usadas por la FPGA para escribir y leer datos que deban mantenerse después de que el sistema sea reiniciado. Como ejemplo, se pueden almacenar datos de configuración, datos de calibración, coeficientes para algoritmos de pre-procesado, etc.

4.4.1.5. Puertos de Entrada/Salida

Los puertos de entrada/salida presentes en el sensor del dedo son dos: el puerto SPI y el JTAG.

- SPI: es el puerto que conecta el sensor del dedo (dispositivo esclavo) con el módulo de control del sistema táctil (dispositivo maestro) e incluye, únicamente, un conector con las señales que se usan para implementar el protocolo SPI (SCLK, MISO, MOSI y tantos SS como dispositivos compartan el bus) ya que no es necesario el uso de *buffers* o *transceivers*. Es, por tanto, una comunicación de FPGA a FPGA.
- JTAG: este protocolo es el empleado para la configuración de la FPGA desde un sistema *host* o para reprogramar la memoria no volátil que contiene el firmware del sensor. Otra de las ventajas de este puerto es la de poder ser usado para la depuración del sistema en tiempo real.

Ambos protocolos comparten conector.

4.4.1.6. Matriz táctil

En el caso del sensor del dedo, el circuito está basado en el integrador pasivo y la matriz de electrodos tiene que estar diseñada de forma que todos los tácteles de cada columna compartan un electrodo común, mientras que el resto de electrodos deben estar aislados eléctricamente entre sí. Por tanto, para una matriz de M filas por N columnas, se emplean N conexiones entre la FPGA y la matriz para poder realizar N medidas en paralelo de los N tácteles de una fila y $M \cdot N$ conexiones independientes para poder direccionar por filas todos los tácteles de la matriz (Figura 3-2).

4.4.2. Sensor para la palma

El sensor empleado para la palma no requiere de un diseño mecánico tan compacto como el del dedo. Su arquitectura es muy similar salvo por cómo se muestrea la matriz táctil (Figura 4-4). En este caso, el direccionamiento de filas se realiza a través de un único cable necesitándose un amplificador operacional por columna para generar una tierra virtual, evitando con ello el *crosstalk* entre los distintos tácteles de una misma fila.

Como se explica en el capítulo 3, el módulo sensor hace uso de dos pines por columna más otro por fila para su conexión directa con la matriz táctil. Además, cada amplificador operacional debe ser apagado/encendido durante cada ciclo de carga/descarga del condensador a través de otra salida (*shutdown*). De todos los pines mencionados, sólo $M+N$ de ellos conectan la matriz táctil con el resto del circuito, por lo que esta conexión se puede llevar a cabo haciendo uso de un conector a tal efecto, pudiendo separar la electrónica de la matriz de electrodos.

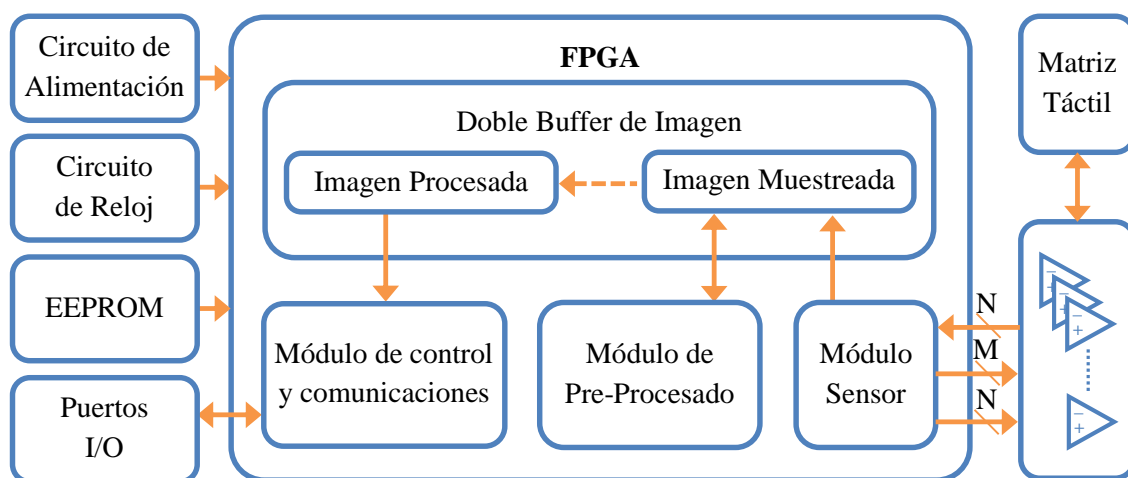


Figura 4-4. Diagrama de bloques del sensor de la palma.

4.5. Prototipo del sistema táctil

Una vez descrita de forma general la arquitectura del sistema táctil, se presentan dos realizaciones de la misma. En esta primera propuesta, se diseña un prototipo, lo más versátil posible, dónde poder llevar a cabo en un entorno de laboratorio todas las pruebas y desarrollos que se precisen para, una vez validados, realizar un diseño a medida para una aplicación concreta.

Este diseño, por tanto, no posee limitaciones en cuanto a especificaciones mecánicas y ha sido concebido como un sistema de desarrollo que facilite la caracterización de los sensores así como la depuración de las comunicaciones y los algoritmos de pre-procesado.

La Figura 4-5 muestra el diagrama de bloques del prototipo del sistema. En este caso, el sistema táctil está formado por tres dedos, de dos falanges cada uno, y una palma. Los tres dedos junto con la palma se comunican con el módulo de control a través de cuatro

buses SPI independientes. En el caso de los dedos, al bus se conectan dos sensores táctiles, mientras que el bus de la palma incluye un único sensor. Además de por razones propiamente estructurales, el número de buses SPI y los sensores conectados a ellos viene marcado por el tamaño de las tramas de datos enviadas por los sensores al módulo de control, mucho mayores en el caso del sensor de la palma si se compara con las de los sensores de los dedos.

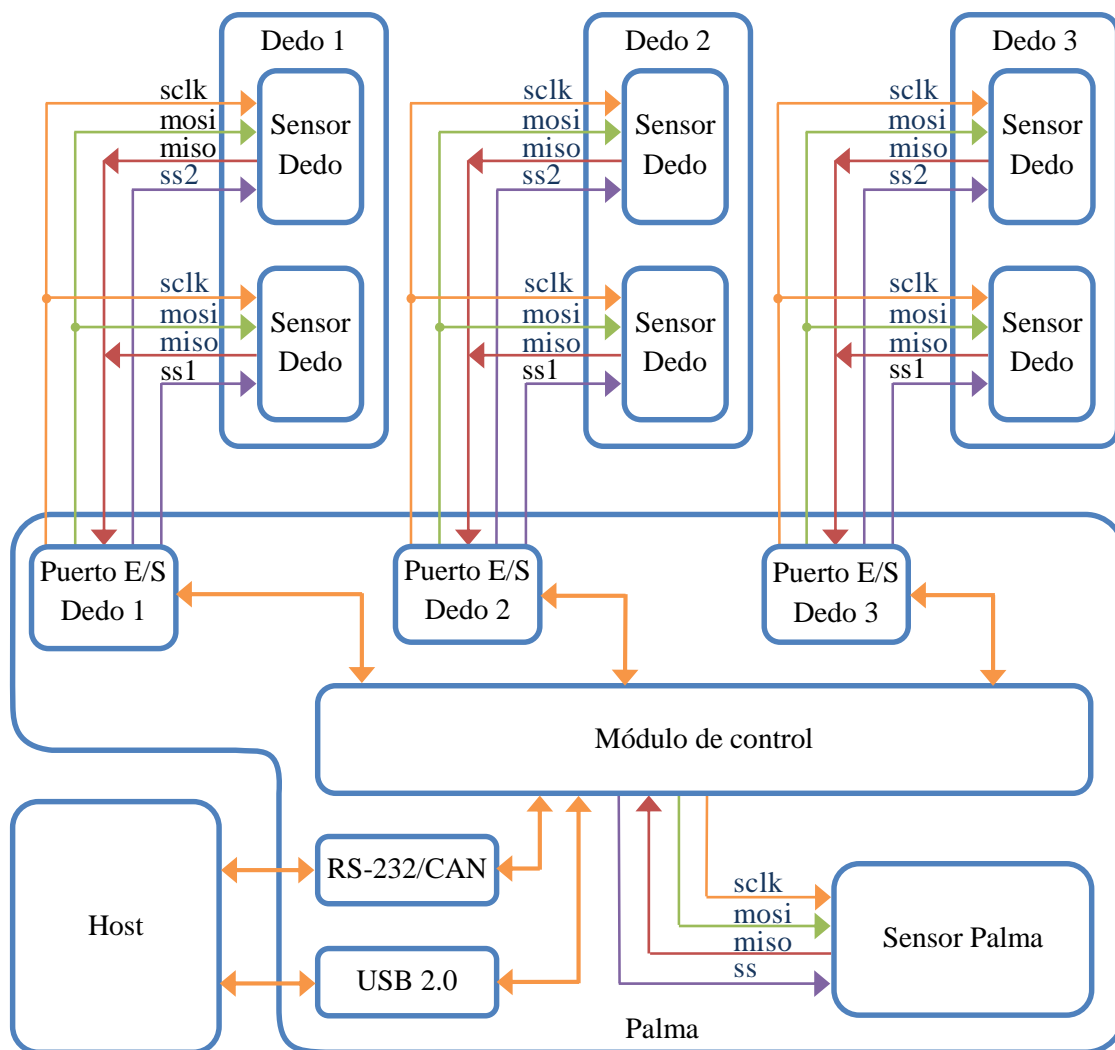


Figura 4-5. Diagrama de bloques del prototipo del sistema táctil.

El sensor de la palma y el módulo de control del sistema se han implementado en una misma placa de circuito impreso, mientras que cada uno de los sensores de dedo se realiza en circuitos separados. La Figura 4-6 muestra el prototipo del sistema táctil, formado por siete circuitos. La disposición de los mismos sigue el esquema de la Figura 4-5, pudiéndose observar cómo los seis sensores de dedo están conectados al circuito de

control a través de tres cables, mientras que el sensor de la palma se ha integrado en el mismo circuito que el módulo de control del sistema.

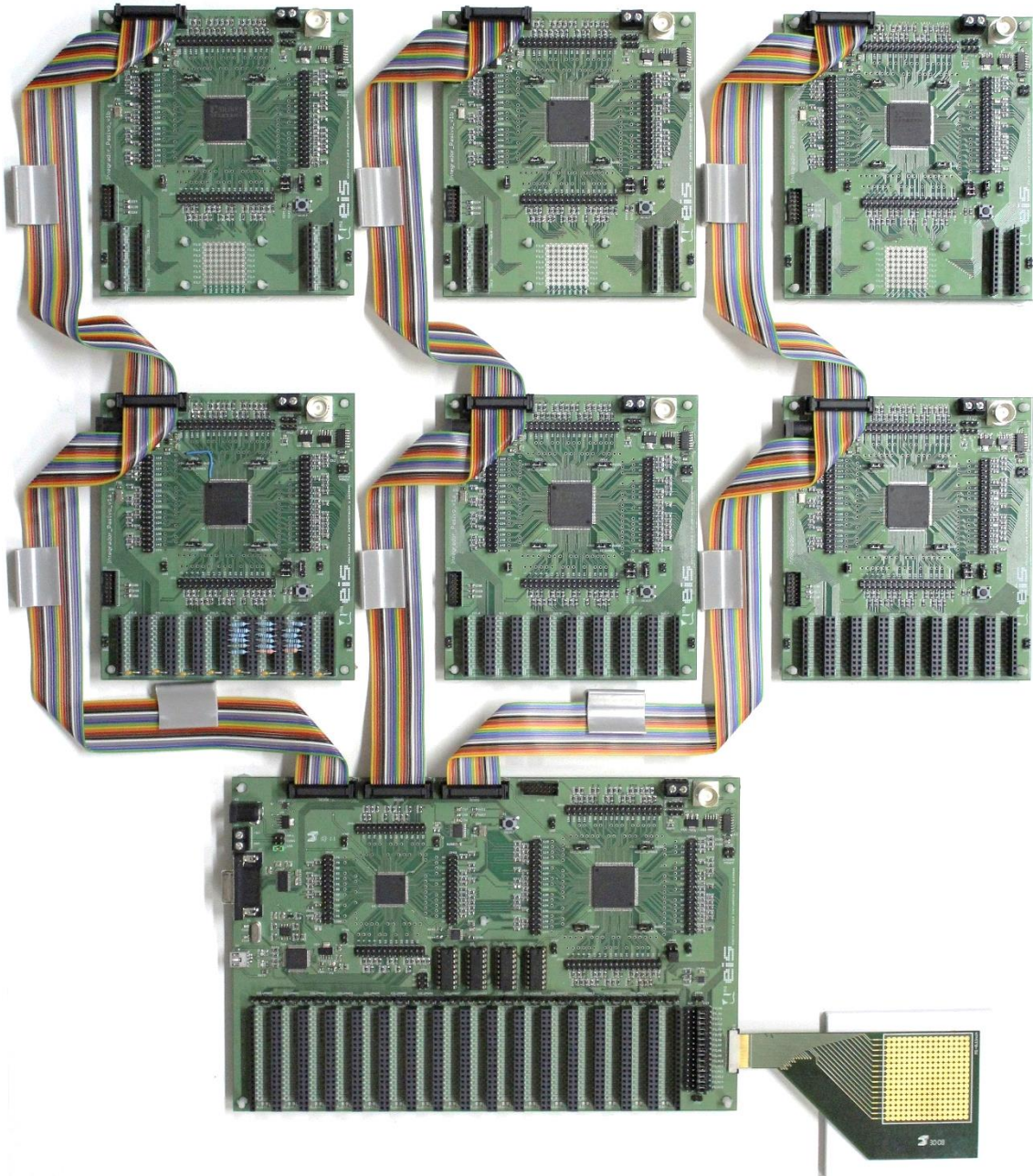


Figura 4-6. Prototipo del sistema táctil.

4.5.1. Módulo de control + sensor de palma

Como se indica anteriormente, el módulo de control y el sensor han sido realizados en un único circuito. La Figura 4-7 muestra su diagrama de bloques.

El circuito se ha fabricado sobre una placa de circuito impreso de cuatro capas con un sustrato de fibra de vidrio FR4 [Underwriters Laboratories 1968]. Las capas externas se han usado para el conexionado de las señales mientras que las dos capas internas se emplean como planos de alimentación, una para GND y la otra para el resto de tensiones de alimentación.

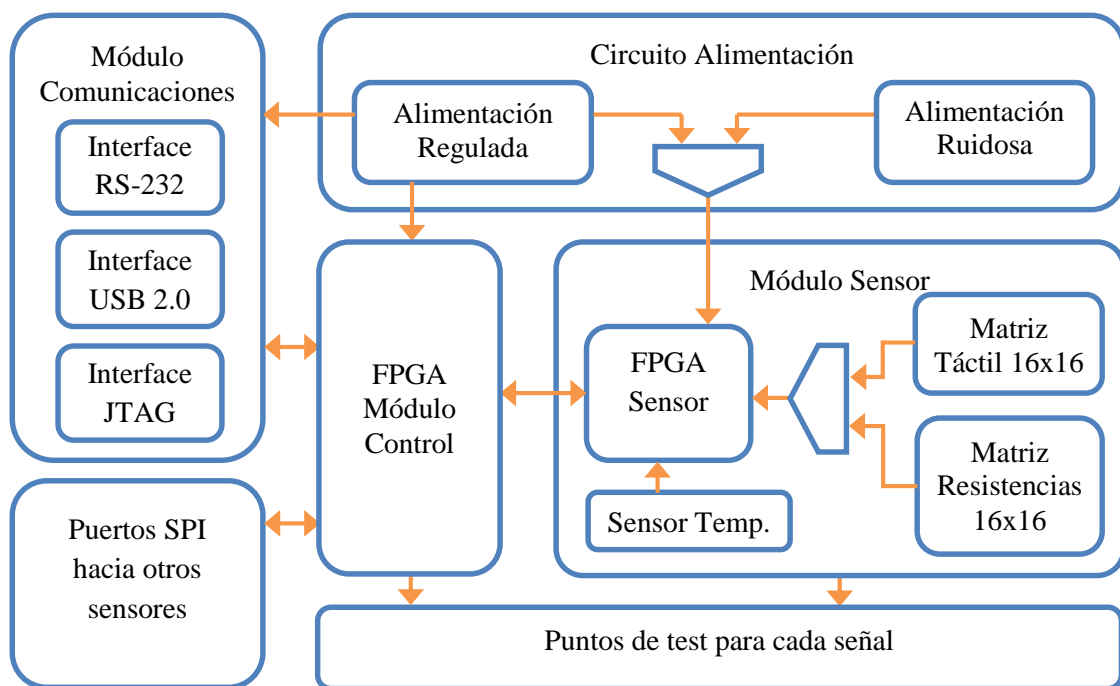
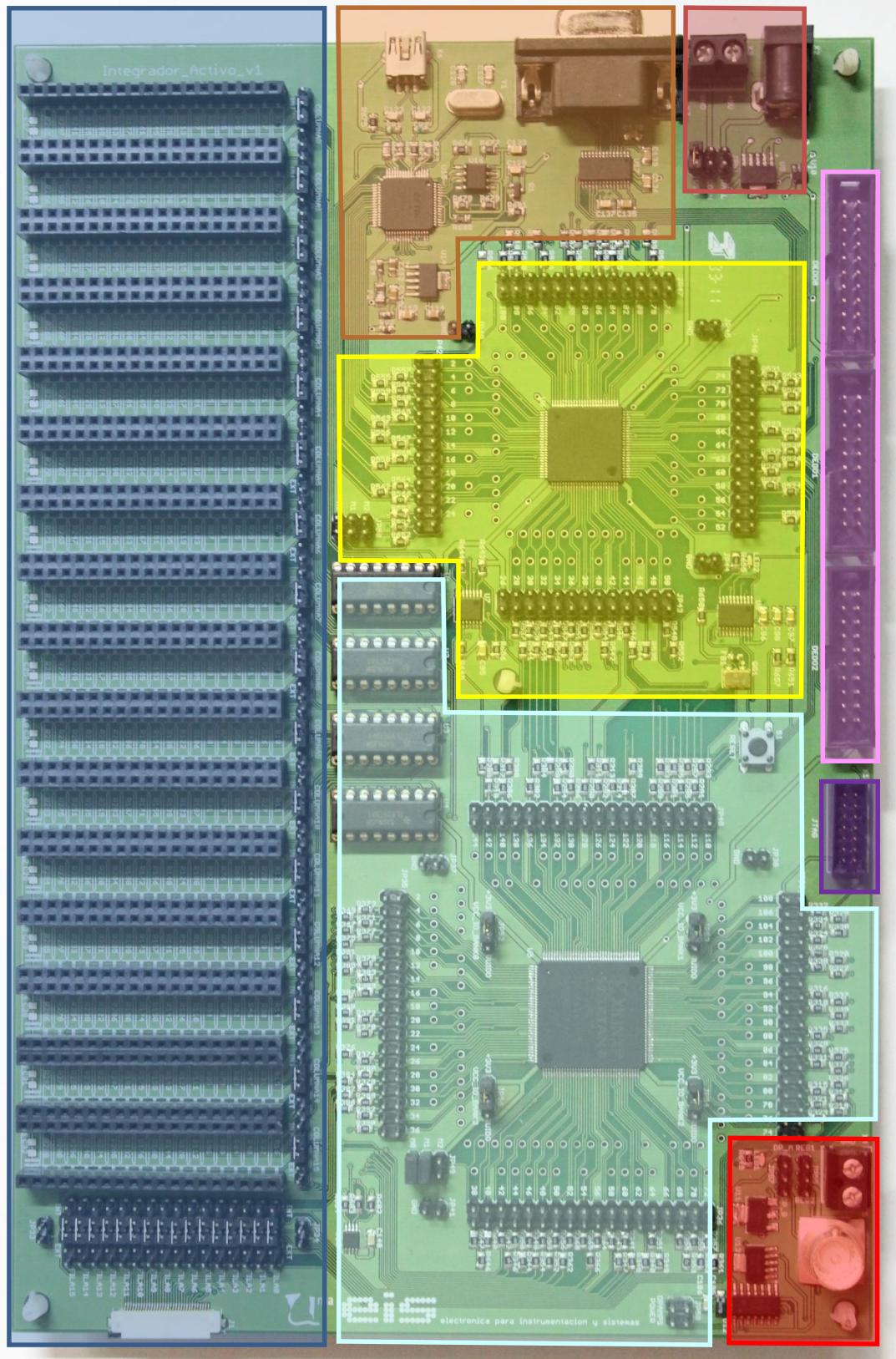


Figura 4-7. Diagrama de bloques del módulo de control + sensor de palma.

En la Figura 4-8 se puede observar dónde se encuentra localizado cada bloque sobre la placa de circuito impreso.

A continuación se describen los bloques del circuito.



- Matriz Resistencias y conector matriz táctil
 - USB 2.0 y RS-232
 - Circuito Alimentación regulada
 - Puertos de expansión SPI
- Módulo de control
 - Módulo Sensor
 - JTAG
 - Circuito Alimentación ruidosa

Figura 4-8. Módulo de control + sensor de palma.

4.5.1.1. Módulo de control del sistema táctil

Como se explica en el apartado 4.3, el módulo de control constituye el núcleo del sistema táctil. Se encarga de la sincronización entre los distintos sensores que lo conforman y de las comunicaciones con el sistema *host*. También podría emplearse para implementar algoritmos de pre-procesado a nivel de sistema, al disponer de la información táctil del sistema completo.

El dispositivo donde se implementa este módulo es una FPGA de la familia Spartan 3A de Xilinx, concretamente el modelo XC3S200A-4VQG100C [Xilinx, Inc. 2010b]. Esta FPGA dispone de los suficientes recursos hardware (200K puertas lógicas equivalentes) y de memoria RAM (288K Kbits) para llevar a cabo las tareas de sincronización, *buffer* de imágenes táctiles y comunicaciones que se precisan.

Para las comunicaciones a través del protocolo SPI entre el módulo de control y los siete sensores del sistema se ha optado por la utilización de señales diferenciales. Las entradas/salidas de las FPGAs pueden ser configuradas conforme a distintos estándares, entre los que se encuentran varios que hacen uso de señales diferenciales, consiguiendo con ello una mayor inmunidad al ruido para comunicaciones de alta velocidad. Así, de los 68 pines de entrada/salida disponibles en esta FPGA, 30 están conectados a los tres puertos que comunican el módulo de control con los sensores de los dedos y otros 8 lo hacen con el bus que lo conecta con el sensor de la palma. Todos ellos están configurados para trabajar bajo el estándar diferencial LVDS (*Low-Voltage Differential Signaling*).

En cuanto a las conexiones con el módulo de comunicaciones, se necesitan 7 pines de la FPGA para la conexión entre el controlador UART (*Universal Asynchronous Receiver-Transmitter*) implementado en el interior de la FPGA y el transceiver RS-232, así como 15 pines para la conexión con el controlador USB 2.0.

La FPGA se configura al inicio del sistema desde una PROM (*Programmable Read Only Memory*) que almacena el flujo de bits que definen su posterior comportamiento. En este caso, la FPGA actúa como dispositivo maestro generando las señales de control necesarias para la carga del firmware desde la memoria no volátil. La memoria empleada (XCF02SVOG20C) [Xilinx, Inc. 2010a] está fabricada por la propia Xilinx, posee un tamaño de 2 Mbits y ofrece una solución reprogramable de bajo coste y fácil uso.

4.5.1.2. Módulo de comunicaciones

Las comunicaciones entre el sistema *host* y el sistema táctil pueden realizarse a través de dos protocolos: RS-232 y USB 2.0.

El protocolo RS-232 (*Recommended Standard 232*) es un protocolo serie de baja velocidad muy empleado en entornos industriales, capaz de establecer un canal de comunicaciones *full-duplex* entre dos dispositivos usando, de forma opcional, mecanismos de negociación (*handshaking*). El protocolo define unos valores concretos de tensión para las señales que hace necesario el uso de circuitos integrados dedicados (*drivers* de línea) para su realización. En este caso, el *transceiver* empleado es el MAX3243EIPWG4 [*Texas Instruments, Inc. 2005*]. Este chip es capaz de obtener los niveles de tensión que cumplen con el estándar RS-232 a partir de una única tensión de alimentación de 3,3 V, alcanzando velocidades de transferencia de hasta 500 kbits/s.

En cuanto al protocolo USB 2.0, el módulo de comunicaciones incluye un controlador USB fabricado por FTDI (FT2232H) [*Future Technology Devices International Ltd 2012*]. Este circuito integrado ofrece una interfaz simple entre dos canales USB de alta velocidad y distintos protocolos serie/paralelo. La configuración, y por tanto su modo de operación, se almacena en una memoria EEPROM de sólo 1kbit que puede ser reprogramada a través del propio controlador USB usando un software proporcionado por el fabricante. La memoria utilizada en el prototipo propuesto es fabricada por Atmel, con código AT93C46DN-SH-B [*Atmel Corporation 2015*].

Como se indica con anterioridad, el chip USB puede ser configurado para trabajar en distintos modos de operación. Las opciones disponibles van desde protocolos serie asíncronos (UART) y síncronos (JTAG, I²C, SPI) hasta interfaces paralelas tipo FIFO (asíncronas y síncronas). En esta implementación concreta, el primero de los canales USB se ha configurado con una interfaz de FIFO síncrona por ser la que mayor ancho de banda consigue (25 Mbytes/s). Para el segundo de los canales se opta por el protocolo serie JTAG.

La interface de FIFO síncrona entre el controlador USB y la FPGA emplea un bus de datos de 8 bits y señales de control de vacío (*empty*) y lleno (*full*) para las FIFOS de entrada y salida, así como señales de control de lectura, escritura y *output enable*. El controlador genera un reloj de 60 MHz con el que se sincronizan las comunicaciones.

Por otro lado, las señales propias del bus JTAG van a un multiplexor junto con las procedentes del conector presente en la placa. Una señal de detección de presencia en el conector controla el selector, decidiendo que señales son las que finalmente acceden a la cadena JTAG del sistema.

De esta forma, se pueden realizar las tareas de configuración y depuración de las FPGAs así como las de reprogramación de las memorias PROM desde un dispositivo externo o desde el propio puerto USB.

4.5.1.3. Puertos de expansión SPI

Los controladores de estos puertos se implementan en la propia FPGA del módulo de control siendo necesario, únicamente, la inclusión de los conectores en los que colocar los cables que unen este circuito con el resto de sensores.

Los conectores así como el cable usado poseen 20 vías con la siguiente disposición:

- 4 vías con la tensión de alimentación de entrada al sistema (5V), con la que se alimenta a todos los sensores conectados al bus.
- 10 vías para las cinco señales diferenciales propias del protocolo SPI (SS1, SS2, SCLK, MOSI y MISO).
- El resto de vías lleva la tensión de tierra del sistema (GND).

Como se explica en el apartado 4.5.1.1 se usan señales diferenciales para implementar el protocolo SPI. Cada par de señales diferenciales discurren apantalladas por el cable entre tensiones GND.

4.5.1.4. Circuito de alimentación

El prototipo del sistema táctil necesita de una tensión de 5 V de alimentación de entrada. Existen tres opciones a la hora de elegir el origen de la fuente de alimentación del sistema: un conector para un adaptador AC/DC con salida de 5V, un conector para la conexión de una fuente de alimentación de laboratorio y la alimentación proporcionada por el propio puerto USB que conecta el sistema con el *host*.

A partir de estos 5 V, un circuito genera dos tensiones reguladas de 3,3 V y 1,2 V que alimentan a todos los circuitos de esta placa salvo al controlador USB, que posee su propio circuito de alimentación basado en los 5 V del bus. Los reguladores empleados son el TPS79533DCQG4 y el TPS79912DDC para las tensiones de salida de 3,3 V y 1,2 V respectivamente [Texas Instruments, Inc. 2015b, 2015c].

4.5.1.5. Módulo sensor de palma

El sensor de palma se ha implementado en otra FPGA. En este caso, el dispositivo elegido ha sido el modelo XC3S50AN-4TQG144C de la familia de FPGAs de Xilinx Spartan 3AN [Xilinx, Inc. 2014].

La principal diferencia con la FPGA empleada para el módulo de control es que, en este caso, la memoria no volátil donde se almacena el flujo de bits con el que se configura el dispositivo en el arranque está incluida en el propio circuito integrado. Esto permite un diseño más compacto, reduciendo el área ocupada por los componentes del circuito. En lo referente a los recursos hardware, esta FPGA dispone de 50K puertas lógicas equivalentes, 54Kbits de memoria RAM y 108 pines de entrada/salida.

El sensor táctil de la palma sigue el esquema del circuito integrador activo propuesto en el capítulo 3. Esto hace que, para la conexión directa entre la FPGA y la matriz táctil, sea necesario el uso de amplificadores operacionales que eviten la influencia en la media de resistencia entre los tácteles que estén siendo muestreados de forma simultánea.

En el prototipo del sistema táctil aquí descrito, la matriz del sensor de palma posee 16 filas y 16 columnas, junto con 2 filas de resistencias de calibración. El número de amplificadores operacionales depende, del número de tácteles que se pretendan medir de forma paralela. En este caso, las imágenes táctiles se obtienen por filas, de arriba abajo en la matriz, midiendo de forma simultánea todos los tácteles de cada fila. Por tanto, se necesitan tantos amplificadores operacionales como columnas tenga la matriz, 16 en este caso. El amplificador operacional empleado en este circuito es el TLV2475 [Texas Instruments, Inc. 2007b].

El número de pines de la FPGA empleados para cada columna de la matriz es igual a dos, y se usan para llevar a cabo la carga del condensador así como para realizar la conversión tiempo-digital con la que se estima el valor de resistencia. Para poder seleccionar la fila

que se pretende medir es necesario emplear una salida de la FPGA por cada una de ellas. Además, se precisan señales para habilitar y deshabilitar los amplificadores según convenga. Resumiendo, para la conexión directa entre la FPGA y la matriz táctil implementada en este prototipo, se necesitan los siguientes pines:

- 32 (16 x 2) pines para las columnas.
- 18 pines para las filas (16 filas de la matriz táctil + 2 filas de calibración).
- 8 pines para el *shutdown* de los 16 amplificadores operacionales, puesto que se controlan por pares (los amplificadores operacionales elegidos se encuentran encapsulados de cuatro en cuatro con dos señales de habilitación/deshabilitación por chip).

El total de pines de la FPGA empleados en la conexión directa es de 58.

Con respecto a la conexión con el módulo de control, al igual que el resto de sensores, el sensor de palma emplea un bus SPI con señales diferenciales, por lo que 8 entradas/salidas configuradas con un estándar diferencial LVDS son utilizadas.

Por último, el sensor de palma necesita 3 pines adicionales para conectarse con un sensor de temperatura que emplea un subconjunto del protocolo I²C (SM-Bus) [*SBS Implementers Forum 2000*] para sus comunicaciones.

4.5.1.6. Sensor de temperatura

Las resistencias de calibración presentes en el sistema son tenidas en cuenta a la hora de estimar el valor de la resistencia del sensor. Esta estimación será mejor cuanto más se parezcan los coeficientes de temperatura de las resistencias de calibración con las del sensor piezorresistivo.

De todos modos, para poder realizar estudios sobre la influencia que las variaciones de temperatura tienen sobre las resistencias, este prototipo incluye un sensor de este tipo con una precisión de 0,5 °C y una resolución de 0,0625 °C. Concretamente, se trata del modelo TMP275 de Texas Instruments [*Texas Instruments, Inc. 2007a*].

4.5.1.7. Matriz de resistencias y matriz táctil

Al tratarse de un prototipo donde poder caracterizar no sólo el sensor táctil propiamente dicho, sino la propia electrónica y la influencia del ruido del sistema sobre los resultados obtenidos, el sensor de palma puede configurarse, por medio de *jumpers*, para actuar sobre una matriz táctil conectada al prototipo por medio de un conector o para hacerlo sobre una matriz de resistencias discretas conocidas, conectadas por medio de zócalos.

El conector que permite el empleo de un sensor táctil basado en un material piezorresistivo tiene 32 contactos, tantos como filas y columnas posee la matriz. Es del tipo FPC/FFC, fabricado por FCI con un *pitch* de 0,5 mm entre sus terminales y permite la conexión directa de una placa de circuito impreso flexible que posee la matriz de electrodos (Figura 4-9).

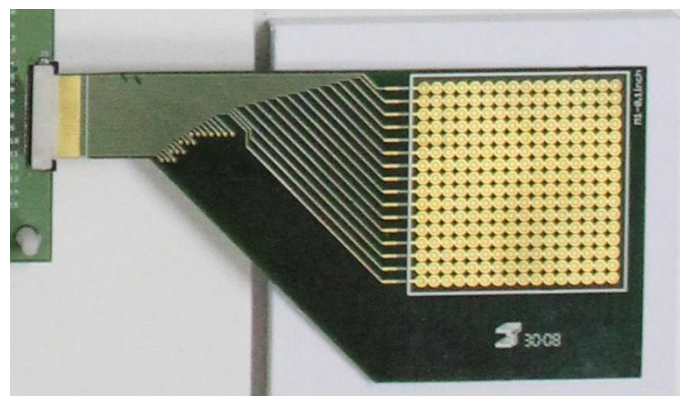


Figura 4-9. Conector FPC/FFC y matriz de electrodos de la matriz táctil.

La placa ha sido fabricada con un sustrato de fibra de vidrio tipo FR4 con un espesor de 0,2 mm, lo que la hace flexible. Sobre la matriz de electrodos se coloca un material que modifica su conductividad en función de la fuerza que se ejerce sobre él. Concretamente, el material empleado se fabrica depositando sobre un sustrato plástico (PET) de 70 micras de espesor una solución de una tinta conductora piezorresistiva (PEDOT) de conductividad 0,082 S/cm [Ochoteco et al. 2007].

4.5.1.8. Puntos de test por cada señal

Este prototipo está ideado como un sistema de pruebas por lo que se han añadido tiras de pines con un terminal macho por cada pin de las dos FPGAs. Esto permite realizar el

estudio del ruido en señales de descarga o en las alimentaciones a través de un osciloscopio y pruebas funcionales en tiempo real usando un analizador lógico.

4.5.2. Sensor de dedo

El sensor táctil para el dedo sigue el esquema del circuito integrador pasivo propuesto en el capítulo 3. En este circuito, la conexión directa entre la FPGA y la matriz de electrodos se realiza a través de un gran número de señales lo que hace poco aconsejable el uso de un conector entre la electrónica y la propia matriz de electrodos. Por ello, se ha optado por integrarla en el propio circuito, dando lugar a que se hayan fabricado dos versiones del sensor de dedo: una con zócalos donde colocar resistencias discretas fijas que simulen la respuesta del sensor piezorresistivo para la caracterización del comportamiento del sensor y otra con la matriz de electrodos integrada.

La estructura del circuito del sensor de dedo sigue un esquema muy similar al del sensor de palma. La Figura 4-10 muestra su diagrama de bloques.

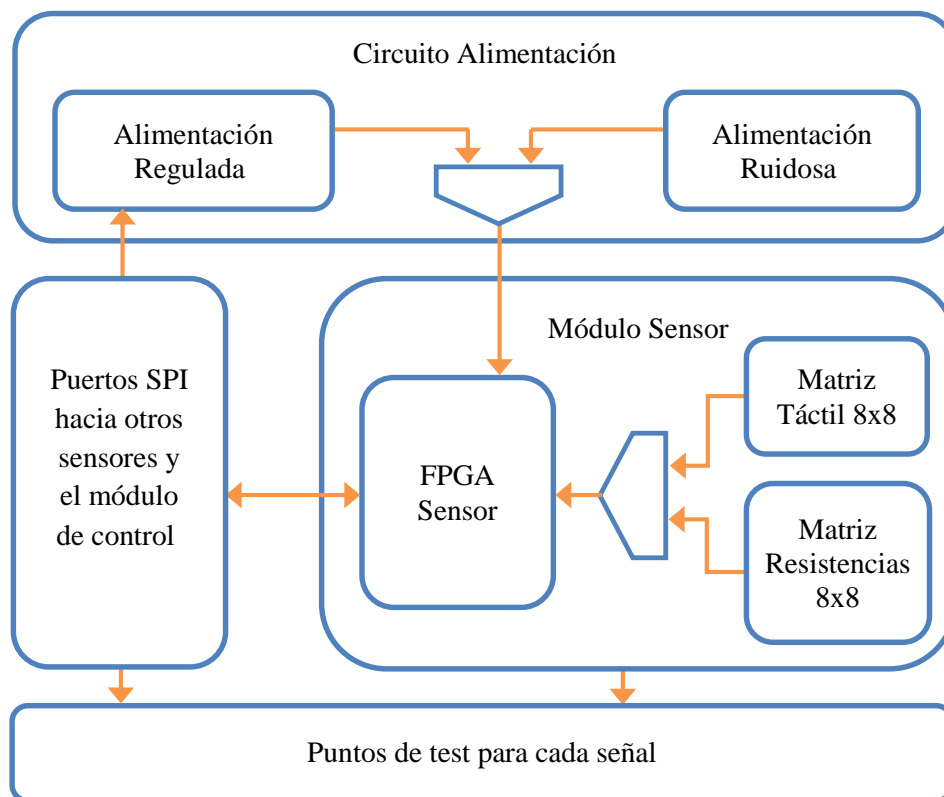
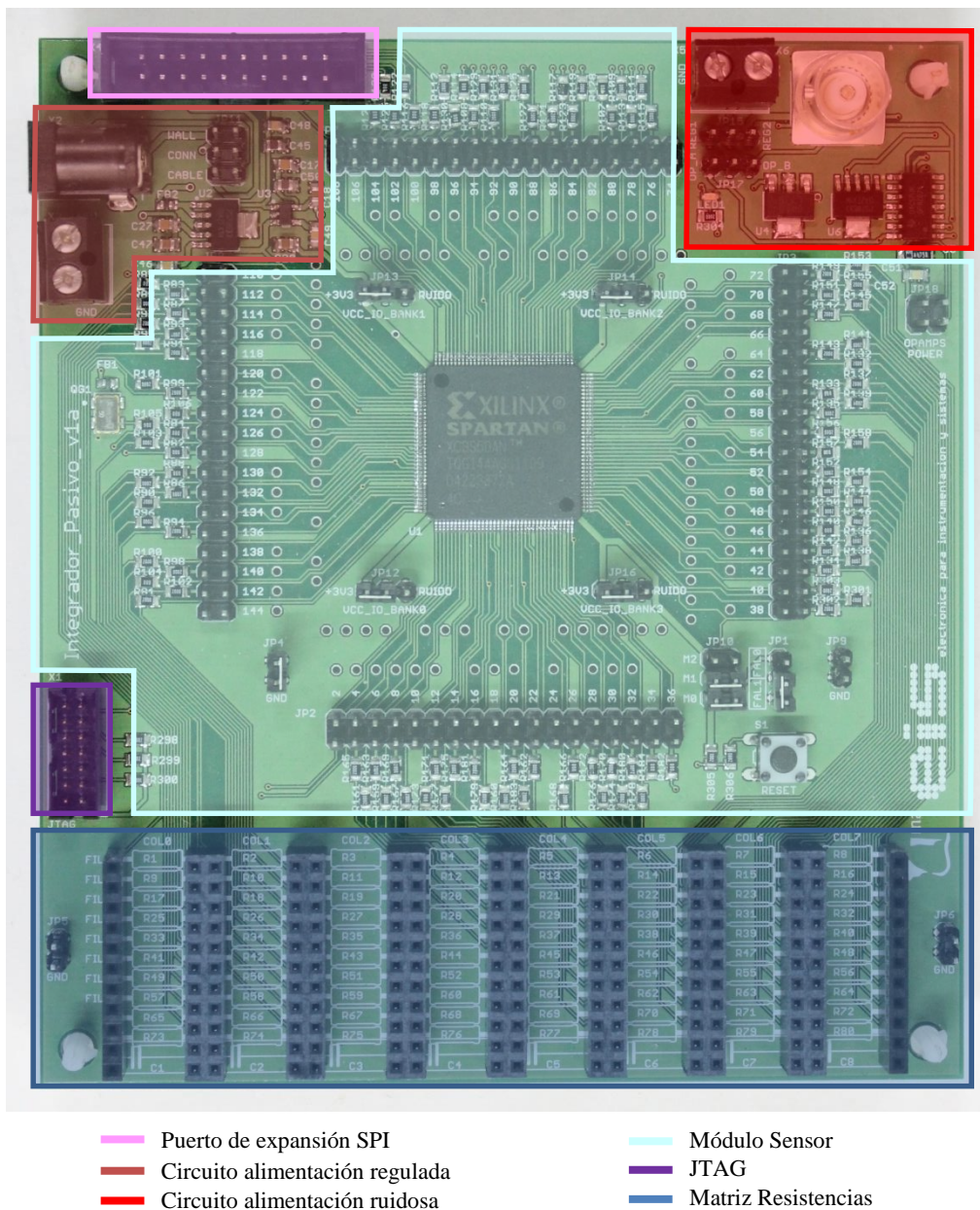


Figura 4-10. Diagrama de bloques del sensor de dedo.

Los circuitos, al igual que en caso del sensor de palma, se ha fabricado en una placa de circuito impreso de cuatro capas con un sustrato de fibra de vidrio FR4. Las capas externas se han usado para el conexionado de las señales mientras que las dos capas internas se emplean como planos de alimentación, una para GND y la otra para el resto de tensiones de alimentación.

La Figura 4-11 presenta el circuito con zócalos para las pruebas de caracterización del sensor, mientras que la Figura 4-12 integra la matriz de electrodos.



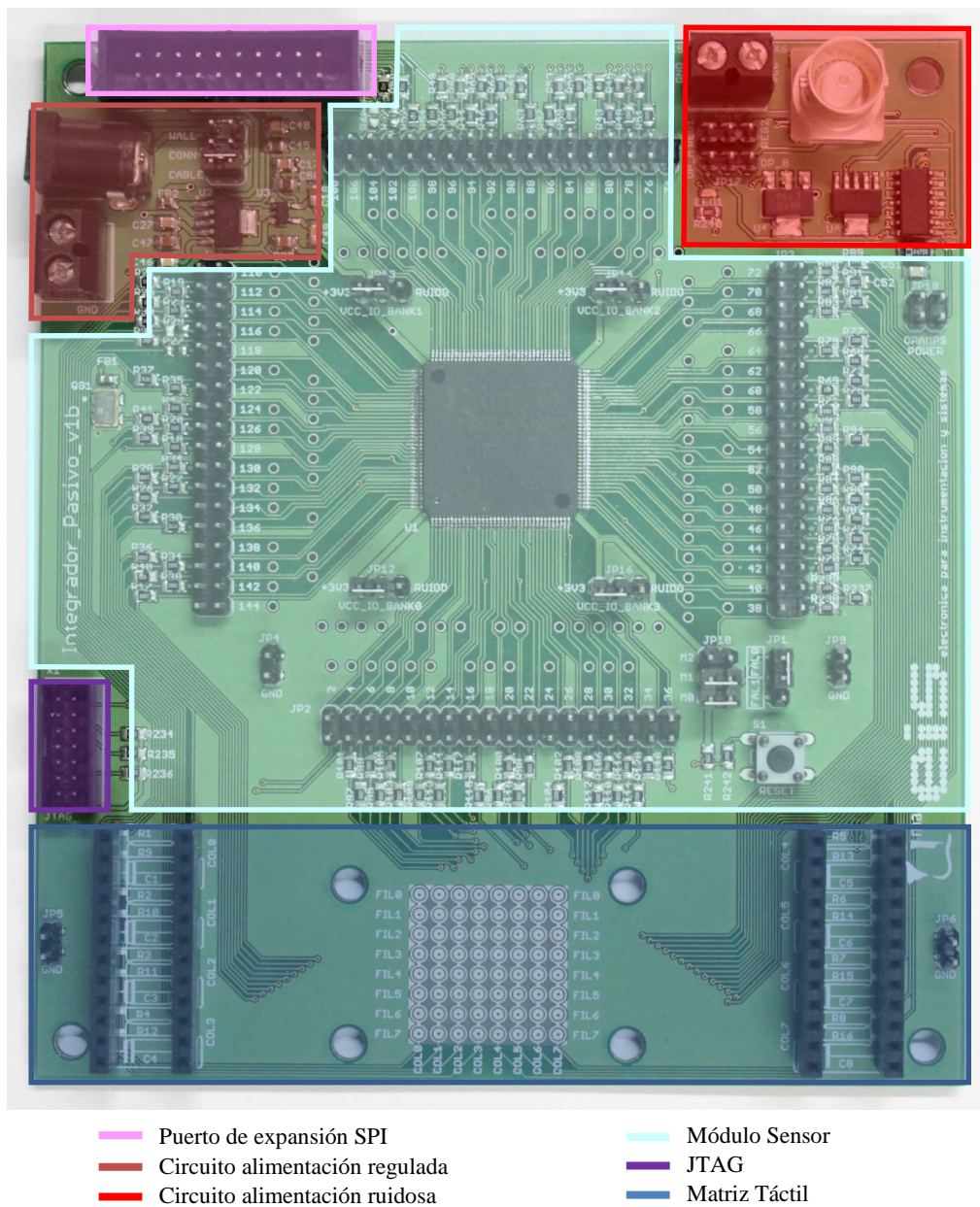


Figura 4-12. Sensor de dedo con matriz de electodos.

4.5.2.1. Circuito de alimentación

Al igual que en el circuito del módulo de control + sensor de palma, existen tres opciones a la hora de elegir el origen de la fuente de alimentación del sensor de dedo: un conector para un adaptador AC/DC con salida de 5 V, un conector para la conexión de una fuente de alimentación de laboratorio y la alimentación proporcionada desde el módulo de control a través del puerto de expansión SPI.

A partir de estos 5 V, un circuito genera dos tensiones reguladas de 3,3 V y 1,2 V que alimentan a la FPGA del sensor.

De igual forma que en el sensor de palma, se ha desarrollado un circuito que permite caracterizar el efecto del ruido en la alimentación de la FPGA en la conversión tiempo-digital. Unos *jumpers* permiten decidir qué alimentación se lleva a cada uno de los cuatro bancos de entrada/salida.

4.5.2.2. Puerto de expansión SPI

En el caso del sensor de dedo, el controlador esclavo del bus SPI se implementa en la propia FPGA del sensor, por lo que sólo es necesario un conector con el que conectarse al bus.

Las señales del conector siguen el mismo esquema que el descrito en el apartado 4.5.1.3, o sea, la tensión de alimentación no regulada de 5 V y las señales diferenciales empleadas por el bus SPI.

4.5.2.3. Módulo sensor de dedo

El sensor de dedo se ha implementado utilizando el mismo modelo de FPGA que el sensor de palma (XC3S50AN-4TQG144C de la familia de FPGAs de Xilinx Spartan 3AN) [Xilinx, Inc. 2014]. Como se comenta en el apartado 4.5.1.5, esta FPGA no necesita de una memoria no volátil para almacenar el programa que define su comportamiento, por lo que se consigue un diseño más compacto.

Para el prototipo del sistema táctil aquí propuesto, la matriz del sensor de dedo posee 8 filas y 8 columnas, junto con 2 filas de resistencias de calibración. Esto hace que, para la conexión directa entre la FPGA y la matriz táctil, se necesiten los siguientes pines:

- 8 pines para las columnas.
- 80 pines para los tácteles (8 filas de la matriz táctil + 2 filas de calibración x 8 columnas).

El total de pines de la FPGA empleados en la conexión directa es de 88.

Con respecto a la conexión con el módulo de control, al igual que el resto de sensores, el sensor de dedo emplea un bus SPI con señales diferenciales, por lo que se emplean 8 entradas/salidas configuradas con un estándar diferencial LVDS.

4.5.2.4. Matriz de resistencias y matriz táctil

Como se indica anteriormente, la Figura 4-11 muestra el prototipo del sensor de dedo que incorpora los zócalos donde colocar la resistencias discretas para las pruebas de caracterización, así como los condensadores que se emplean en la conversión tiempo-digital.

Por otro lado, la Figura 4-12 ilustra el circuito que integra la matriz de electrodos junto con la electrónica del sensor de dedo. Para la matriz, se opta por el diseño redondo con áreas balanceadas por considerarse el que mejor resultados de sensibilidad ofrece [Castellanos-Ramos *et al.* 2010].

El material piezorresistivo que se coloca sobre la matriz de electrodos posee las mismas características que el empleado en el sensor de palma (sustrato plástico de 70 micras de espesor con una solución de una tinta conductora piezorresistiva de conductividad 0,082 S/cm). En este caso, el área de contacto de la lámina con cada par de electrodos correspondientes a un táctel se aísla mediante laser para evitar el *crosstalk* en la medida de resistencia. La Figura 4-13 muestra con detalle como cada zona de la lámina en contacto con cada táctel se encuentra aislada del resto.

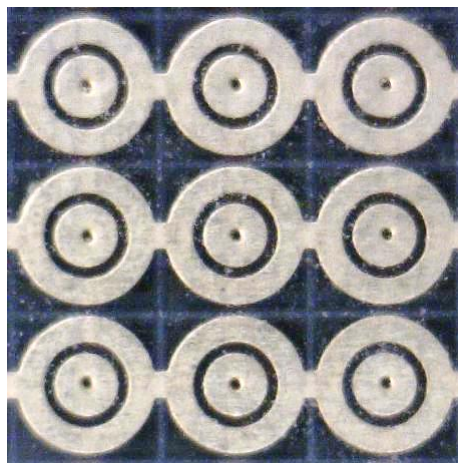


Figura 4-13. Detalle de la lámina de material discreto sobre la matriz de electrodos.

4.5.2.5. Puntos de test por cada señal

Para facilitar las pruebas de caracterización del sensor y la depuración del programa de la FPGA se han añadido tiras de pines con un terminal macho por cada pin del chip.

4.6. Prototipo para una mano de Barrett

Para llevar a cabo un demostrador del prototipo de sistema táctil se ha usado como base a la mano de Barrett [Barrett Technology, Inc 2015] (Figura 4-14). Esta mano posee tres dedos multi-articulados teniendo, dos de ellos, un grado de libertad con 180 grados de movilidad lateral sincronizada. Aunque en la actualidad la mano de Barret puede ser adquirida con soporte táctil como opción, las primeras versiones no la incluían. De esta forma, el prototipo aquí presentado puede ser instalado sobre un modelo de mano antigua (BH8-262 o anterior), proporcionándole capacidad táctil.

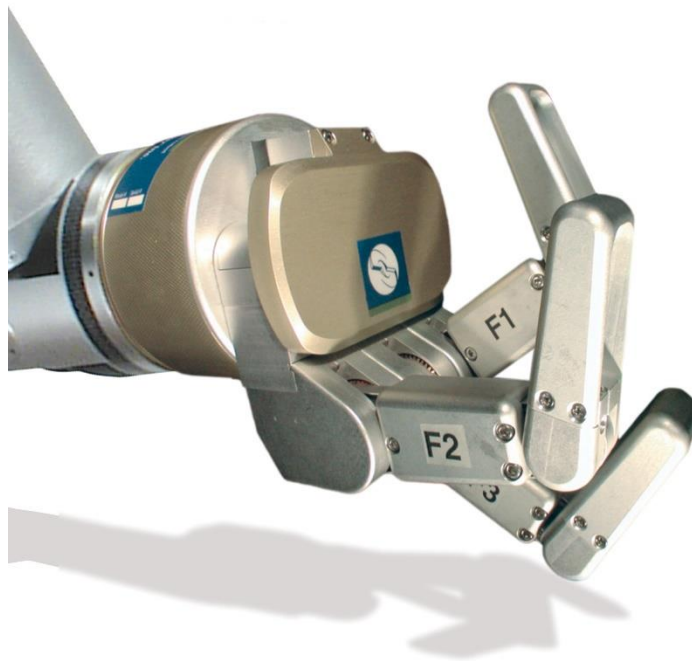


Figura 4-14. Mano de Barrett.

Aunque cada dedo de la mano de Barrett posee dos falanges, este prototipo sólo dota de capacidad táctil a la falange distal de cada uno de ellos. Esta última falange no posee ningún elemento mecánico relacionado con el movimiento por lo que resulta muy fácil

de sustituir por un nuevo dedo que incluya al sensor táctil. Por tanto, el sistema táctil propuesto está formado por tres sensores de dedo y un sensor de palma (Figura 4-15).

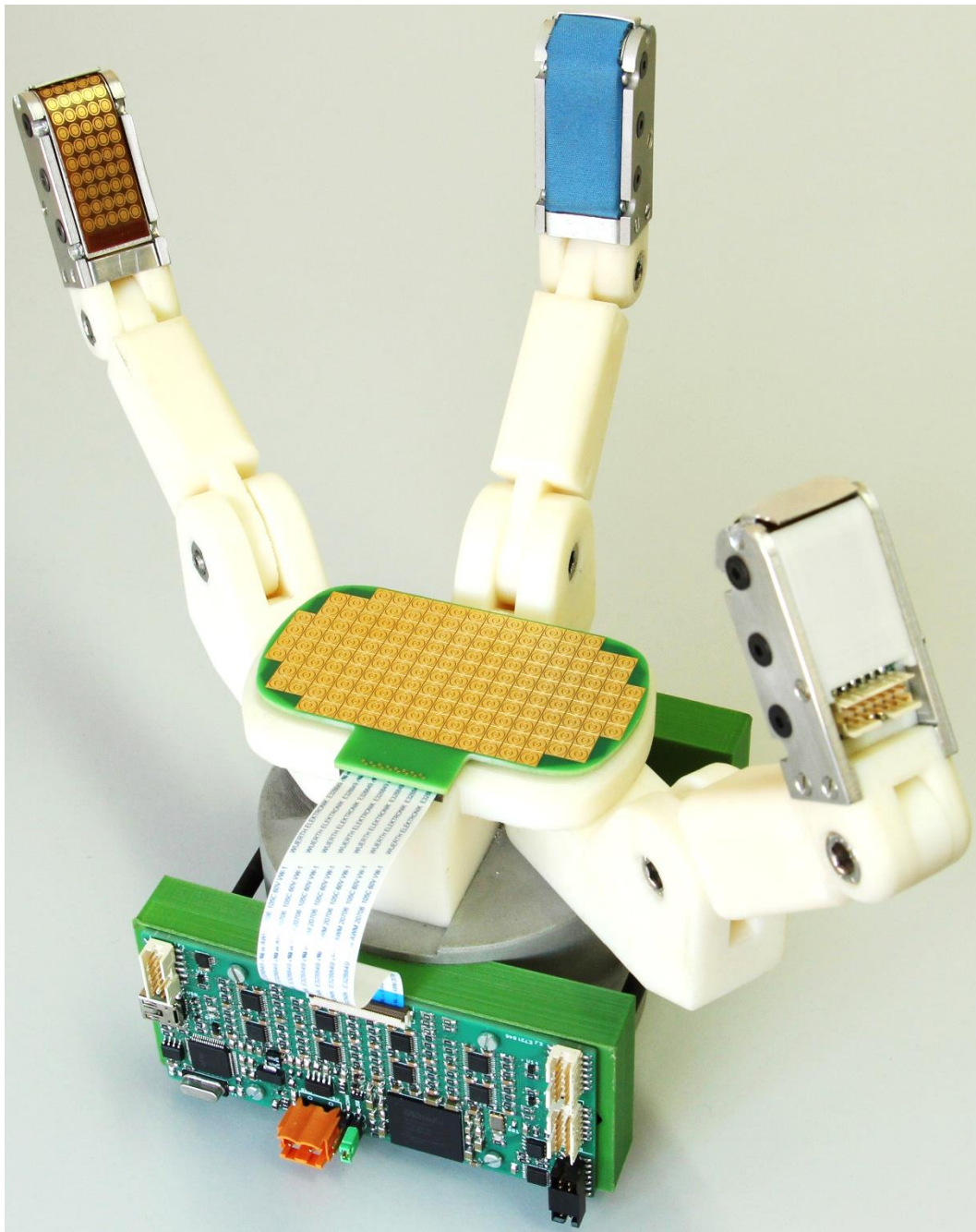


Figura 4-15. Prototipo de sistema táctil para la mano de Barrett.

Los sensores colocados en los dedos han sido integrados en el interior de los mismos y siguen la arquitectura propuesta para el sensor de dedo del prototipo de laboratorio del apartado 4.5.2. La conexión de cada sensor de dedo con el controlador se realiza por medio de un cable, por donde viajan tanto la alimentación como los datos recibidos y enviados a través del bus SPI.

El sensor de palma se ha integrado junto con el controlador del sistema táctil en una única placa de circuito impreso que se instala en la muñeca de la mano robótica. Dos matrices de electrodos diferentes pueden ser conectadas al sensor de palma, pudiéndose evaluar dos resoluciones espaciales distintas en este sensor. Las comunicaciones con el sistema *host* se han reducido a una con respecto al prototipo de laboratorio, optándose por la conexión USB. Este protocolo permite comunicaciones de alta velocidad a la vez que proporciona la tensión de alimentación del sistema, por lo que aporta la solución más compacta de entre las barajadas.

Los esquemáticos de los circuitos aquí descritos pueden consultarse en el Anexo B.

4.6.1. Módulo de control + sensor de palma

Al igual que en el prototipo de laboratorio presentado en este trabajo, el sensor de palma y el controlador del sistema táctil han sido fabricados sobre la misma placa de circuito impreso. Para ello, se ha empleado una PCB de seis capas sobre un sustrato FR4, con la siguiente distribución:

- Capa Top: Componentes y pistas de señales.
- Capa 2: Pistas de señales.
- Capa 3: Tensiones de alimentación (5 V, 3,3 V, 1,2 V).
- Capa 4: GND.
- Capa 5: Pistas de señales.
- Capa Bottom: Pistas de señales.

Los componentes que conforman el sensor son (Figura 4-16):

- FPGA: el modelo de FPGA elegido es el 6SLX25FTG256 de la familia Spartan 6 de Xilinx. Este dispositivo es mucho más potente que el empleado en el prototipo de laboratorio. Dispone de 3758 *slices* (cada *slice* contiene cuatro *Look-Up Tables* de seis entradas y ocho *flip-flops*), 936Kb de bloques de memoria RAM dedicada, 38 bloques DSP48A1 (cada uno con un multiplicador 18x18, un sumador y un acumulador) y 186 entradas/salidas configurables. El circuito utiliza un encapsulado BGA de 17 x 17 mm [Xilinx, Inc. 2011, 2015a]. En este caso, una única FPGA hace de controlador del sistema táctil así como de sensor de palma.

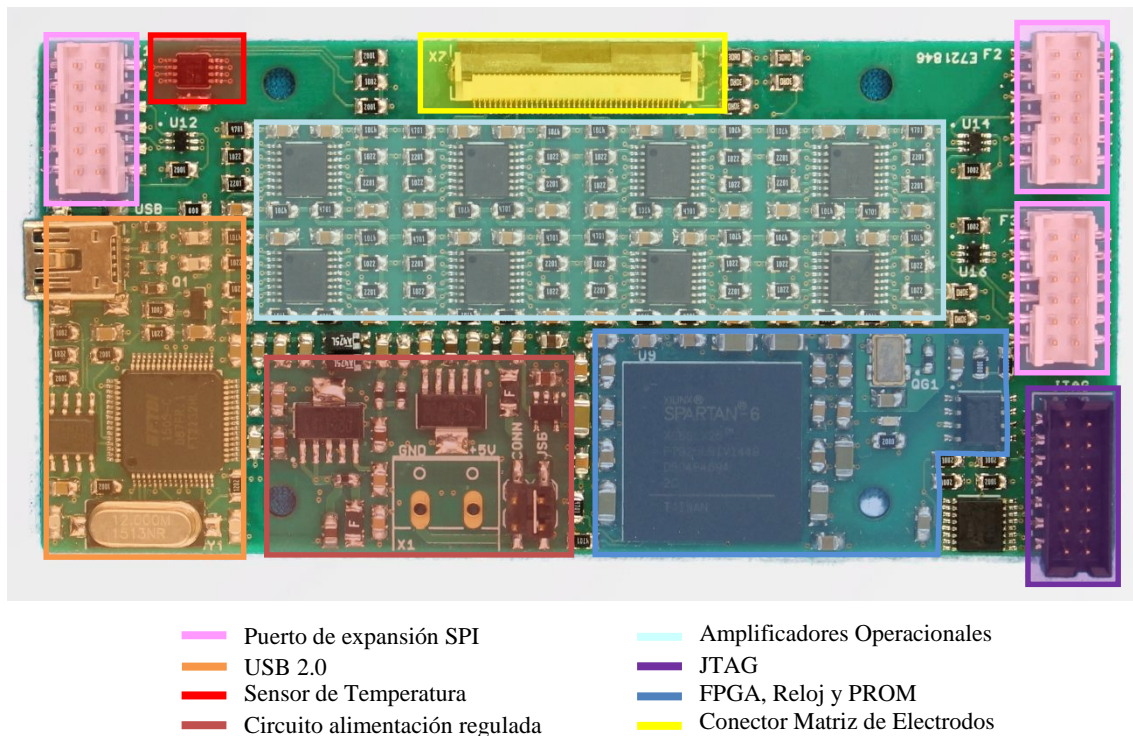


Figura 4-16. Módulo de control + sensor de palma – Mano de Barrett.

- PROM: el firmware con el que se configura la FPGA puede alcanzar un tamaño máximo de 6,4 Mb, por lo que se ha optado por una memoria EEPROM con protocolo serie SPI modelo AT45DB081E-SSU [Adesto Technologies 2015b] de 8Mb de capacidad.
- Circuito de alimentación: el circuito de alimentación coincide con el empleado en el prototipo de laboratorio del apartado 4.5.1.4. Los reguladores empleados son el TPS79533DCQG4 y el TPS79912DDC para las tensiones de salida de 3,3 V y 1,2 V respectivamente [Texas Instruments, Inc. 2015b, 2015c].
- Circuito de reloj: para el circuito de reloj se ha mantenido el mismo resonador que en el prototipo de laboratorio (FXO-HC536R-50 [FOX Electronics 2008] con una frecuencia de trabajo de 50MHz).
- Sensor de temperatura: se usa el mismo que en el prototipo de laboratorio (TMP275 de Texas Instruments [Texas Instruments, Inc. 2007a]).
- Puertos de expansión SPI: Tres conectores son usados como puertos de SPI para cada uno de los tres sensores de dedo. Las señales que proceden y van a los conectores no necesitan de ninguna circuitería para su normal funcionamiento. En este caso, y debido a las restricciones en cuanto a dimensiones del circuito, no se emplean señales diferenciales para implementar el protocolo SPI. Se realizaron

pruebas con el prototipo de laboratorio que indicaban que, para las distancias que los cables iban a requerir, la integridad de las señales estaba asegurada. Además, en esta realización, las señales del protocolo JTAG para la configuración/depuración de las FPGAs y para la programación de las memorias EEPROM, también van por el mismo cable para evitar el uso de un conector a tal efecto en los sensores de dedo. Los conectores poseen 12 vías, 4 para las señales del protocolo SPI (SS, SCLK, MOSI, MISO), otras 4 para el protocolo JTAG (TMS, TCK, TDI, TDO), 1 para la alimentación de los sensores de dedo (5 V), 1 para la salida del circuito de alimentación de los sensores de dedo (3,3 V) y 2 para la tensión de tierra (GND). La alimentación de 3,3 V del sensor se envía al cable para ser usada como detector de presencia por parte del circuito del controlador del sistema táctil. De esta forma, una serie de multiplexores, controlados por estas señales de detección de presencia, se encargan de cerrar la cadena JTAG en lo que a la señal TDO se refiere, independientemente de los sensores de dedo que haya conectados (Figura 4-17).

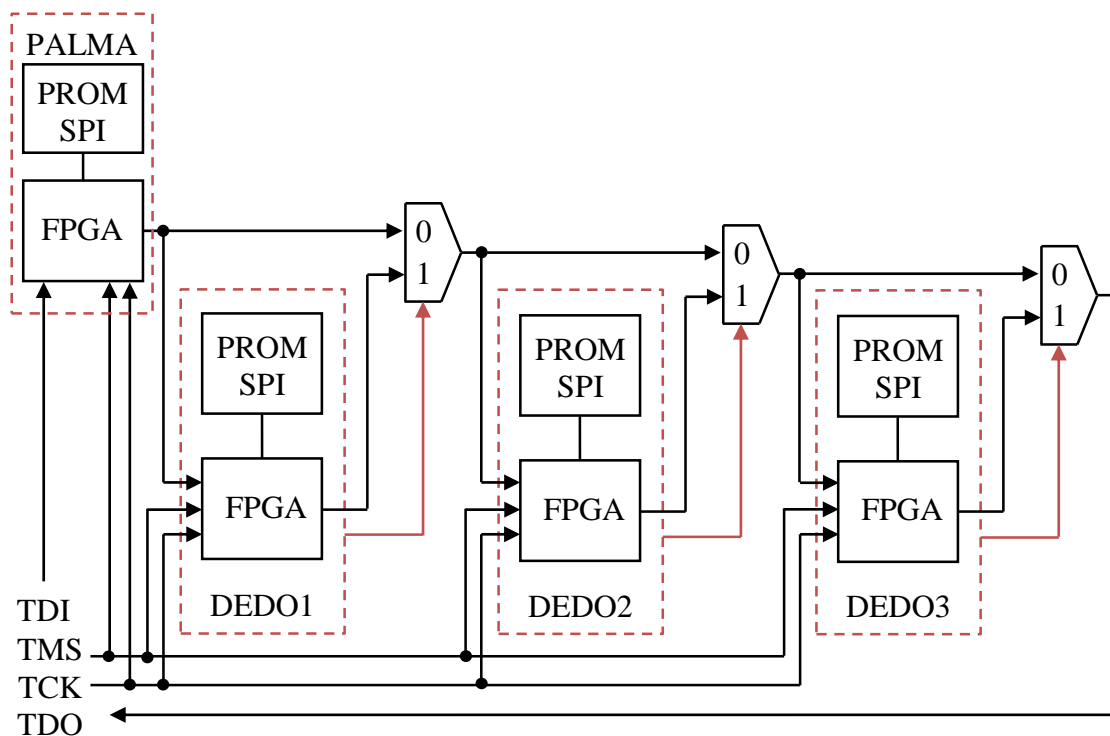


Figura 4-17. Cadena JTAG del sistema táctil para la mano de Barrett.

- Conector JTAG: a este conector llegan las señales propias del protocolo JTAG (TMS, TCK, TDI, TDO) así como la salida de alimentación regulada de 3,3 V y GND.
- Amplificadores operacionales: el amplificador operacional empleado coincide, de nuevo, con el del prototipo de laboratorio del apartado 4.5.1.5 (TLV2475 [Texas Instruments, Inc. 2007b]).
- Módulo de comunicaciones: en este caso se ha optado por implementar únicamente el protocolo de comunicaciones de alta velocidad USB 2.0 por su versatilidad. Este protocolo permite tasas de transferencia altas, donde las imágenes táctiles pueden ser enviadas a un sistema *host* tal cual son leídas, pero también puede emplearse en otras soluciones donde únicamente parámetros extraídos localmente son transmitidos.
- Conector matriz de electrodos: este conector tiene tantas conexiones como filas y columnas compartidas existan en la matriz táctil. El sensor de palma se ha diseñado para poder trabajar con dos matrices distintas:
 - Matriz de 10 filas y 16 columnas (Figura 4-18): aunque la matriz, aparentemente, posee 10 filas y 16 columnas, las conexiones hacia la FPGA hacen que esta última la considere como una matriz de 5 filas y 32 columnas, favoreciendo el tiempo de muestreo del total de la matriz (se lee en la mitad de tiempo). La superficie de sensado de cada táctel es de $20,07 \text{ mm}^2$ ($4,48 \text{ mm} \times 4,48 \text{ mm}$) con electrodos con áreas balanceadas.

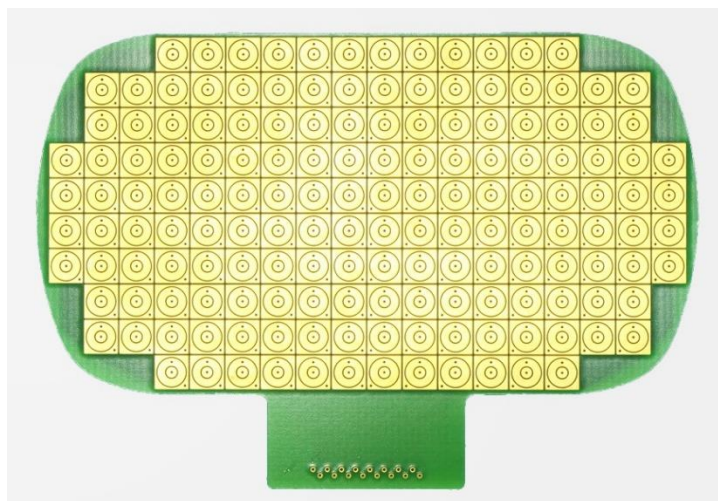


Figura 4-18. Matriz de electrodos para la palma de la mano de Barrett con 10 filas y 16 columnas.

- Matriz de 5 filas y 8 columnas (Figura 4-19): en este caso, la distribución de filas y columnas coinciden con las conexiones hacia la FPGA. En este caso, la superficie de sensado de cada táctel es de $78,3 \text{ mm}^2$ ($8,85 \text{ mm} \times 8,85 \text{ mm}$) con electrodos con áreas balanceadas.

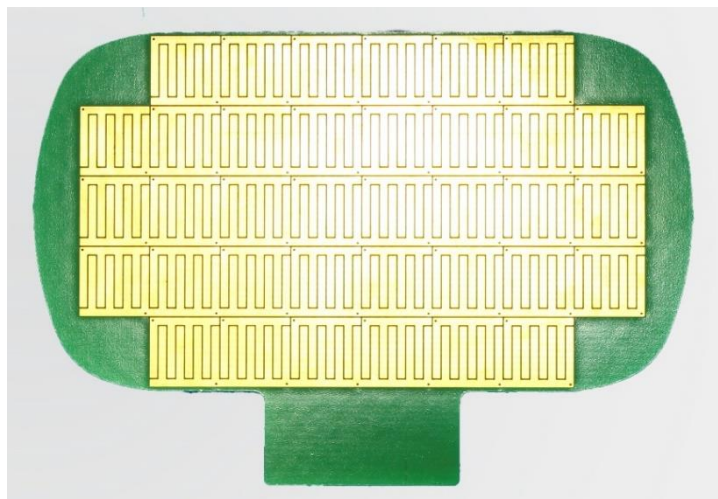


Figura 4-19. Matriz de electrodos para la palma de la mano de Barrett con 5 filas y 8 columnas.

El procedimiento de lectura de la matriz táctil y el tiempo empleado para ello es por tanto el mismo para ambas matrices. La única diferencia es que, en el primer caso, son necesarios 32 módulos de captura trabajando en paralelo, mientras que en el segundo sólo 8 son usados. Los amplificadores operacionales que no estén siendo usados, en este segundo caso, pueden deshabilitarse para no influir en el ruido del sistema ni en el consumo del mismo.

4.6.2. Sensor de dedo

El sensor de dedo es el que mayores restricciones mecánicas posee puesto que debe ser integrado en la propia falange del dedo. La matriz de electrodos debe adaptarse a la forma del dedo [Kim et al. 2013; Choi et al. 2006] por lo que, para la fabricación de la placa de circuito impreso que contiene la electrónica del sensor, se ha optado por una tecnología “Rigiflex” que combina una parte flexible junto con la habitual zona rígida. La construcción de este tipo de circuito posee un núcleo central de poliimida con cobre laminado en ambos lados que constituyen las capas internas. Las capas externas son fabricadas con fibra de vidrio (FR-4). Estos circuitos poseen un coste de fabricación superior al de la tecnología rígida. Como ejemplo, un circuito de cuatro capas “Rigiflex”

es varias veces superior al coste de un circuito de cuatro capas rígido. Esto es debido, a la compleja manipulación de los materiales así como sus elevados costes. La dificultad del proceso se debe a la baja estabilidad dimensional del material, su alta absorción de humedad, a la maquinaria específica necesaria para manipular material tan flexible y a una más compleja metalización de los taladros. La norma de referencia para los circuitos impresos “Rigiflex”, es la IPC-6013 [IPC 2013]. La Figura 4-20 muestra con detalle las pequeñas dimensiones del circuito así como su capacidad flexible.

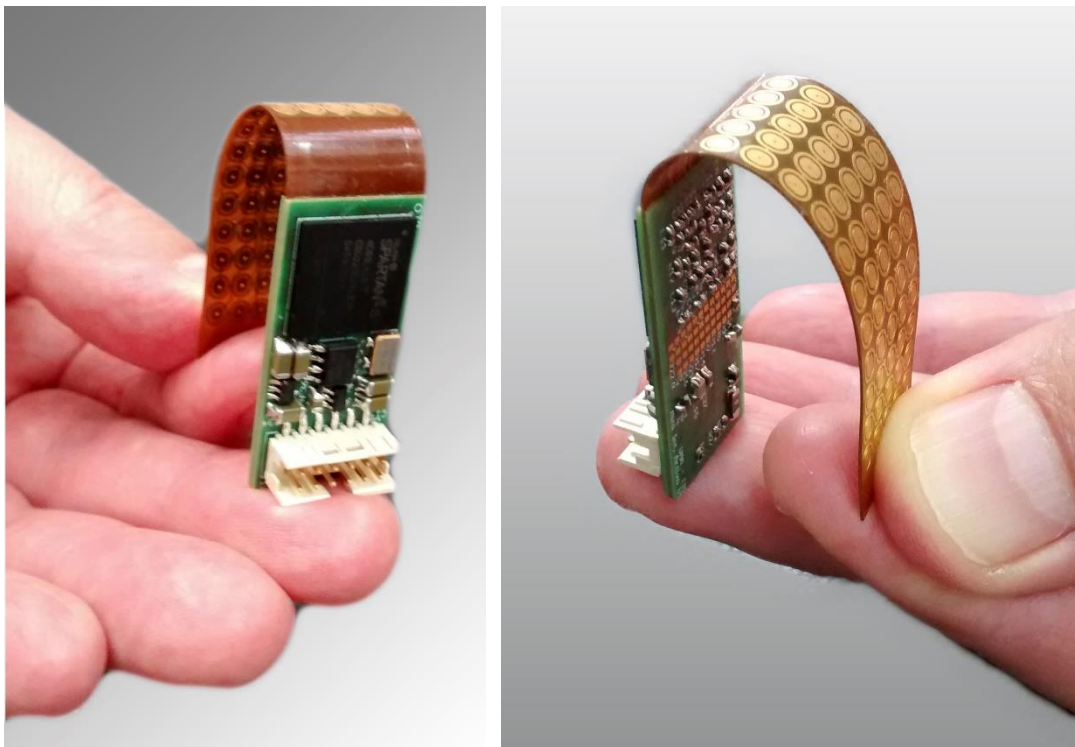


Figura 4-20. Detalle de la placa Rigiflex del sensor de dedo.

La PCB del sensor está fabricada a seis capas. En la zona rígida se monta la electrónica, mientras que en la parte flexible se encuentra la matriz de electrodos. Los componentes que conforman el sensor son (Figura 4-21):

- FPGA: el modelo de FPGA elegido es el 6SLX16CSG225 de la familia Spartan 6 de Xilinx. Este dispositivo es mucho más potente que el empleado en el prototipo de laboratorio. Dispone de 2278 *slices* (cada *slice* contiene cuatro *Look-Up Tables* de seis entradas y ocho *flip-flops*), 576Kb de bloques de memoria RAM dedicada, 32 bloques DSP48A1 (cada uno con un multiplicador 18x18, un sumador y un acumulador) y 160 entradas/salidas configurables. El circuito utiliza un encapsulado BGA de 13 x 13 mm [Xilinx, Inc. 2011, 2015a].

- PROM: en este caso, el flujo de bits para configurar la FPGA puede alcanzar un tamaño máximo de 3,7 Mb, por lo que se ha optado por una memoria EEPROM con protocolo serie SPI modelo AT45DB041D-SSU [Adesto Technologies 2015a] de 4Mb de capacidad.
- Circuito de alimentación: el circuito de alimentación ha sufrido cambios debido, sobre todo, a las restricciones mecánicas que debe cumplir el circuito. Los reguladores empleados para las tensiones de los bloques de entrada/salida (3,3V) y para el núcleo (1,2V) poseen un encapsulado mucho más reducido, manteniendo las características de los utilizados en el prototipo de laboratorio (*low dropout, ultra low noise*). Los modelos empleados son el LP5907MFX-3.3 y el LP5907MFX-1.2 para las tensiones de 3,3V y 1,2V respectivamente [Texas Instruments, Inc. 2015a].
- Circuito de reloj: para el circuito de reloj se ha mantenido el mismo resonador que en el prototipo de laboratorio (FXO-HC536R-50 [FOX Electronics 2008] con una frecuencia de trabajo de 50MHz).
- Puerto de comunicaciones/programación: el puerto como tal lo constituye únicamente un conector, puesto que las señales que proceden y van al mismo no necesitan de ninguna circuitería para su normal funcionamiento. Como se indica anteriormente, en esta realización del sensor de dedo, las señales del protocolo JTAG para la configuración/depuración de la FPGA y para la programación de la memoria EEPROM, también vienen por el mismo cable para evitar el uso de un conector a tal efecto. El conector posee 12 vías, 4 para las señales del protocolo SPI (SS, SCLK, MOSI, MISO), otras 4 para el protocolo JTAG (TMS, TCK, TDI, TDO), 1 para la alimentación del sensor (5 V) proveniente del módulo de control, 1 para la salida del circuito de alimentación del sensor (3,3 V) y 2 para la tensión de tierra (GND). La alimentación de 3,3 V del sensor se envía al cable para ser usada como detectar de presencia del sensor por parte del circuito del controlador del sistema táctil.

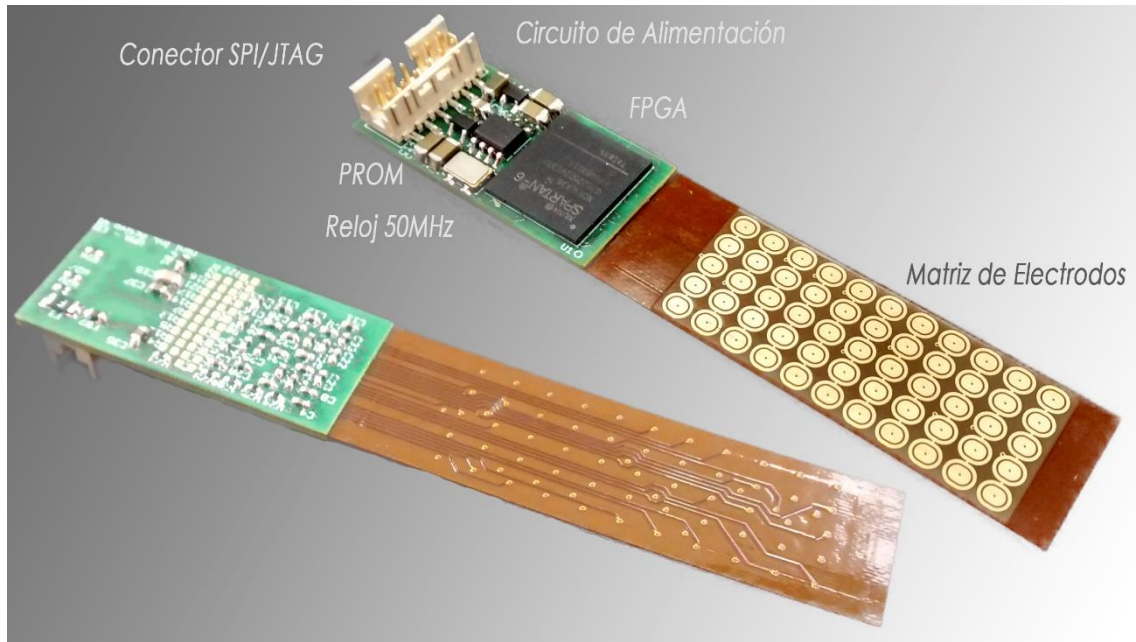


Figura 4-21. Componentes del sensor de dedo para la mano de Barrett.

- Matriz de electrodos: la matriz está formada por 55 electrodos dispuestos en 11 filas y 5 columnas. Esta organización, que sigue la forma del dedo (más alto que ancho), no es la más eficiente para cubrir la superficie a sensar por lo que, internamente, la FPGA la considera como una matriz de 5 filas y 11 columnas, pudiendo efectuar las 11 medidas simultáneas de los tácteles de una fila de forma paralela. La FPGA por tanto sólo debe cambiar filas por columnas a la hora del envío de datos hacia el controlador para que la información llegue correcta al sistema *host*. La Figura 4-22 muestra un detalle de la matriz de electrodos junto con otro de la lámina de material discreto piezorresistivo. Un láser de baja potencia se ha empleado para separar las distintas zonas del material en contacto con cada uno de los electrodos.

Dado que el sensor de dedo implementa el enfoque del Integrador Pasivo para llevar a cabo el direccionamiento de la matriz táctil, se necesitan 66 conexiones entre la matriz y la FPGA. Esto hace que, para la parte flexible del circuito donde se encuentra la matriz de electrodos, haya sido necesario el empleo de cuatro capas (los electrodos apenas dejan huecos en una de las capas externas de la parte flexible). La matriz táctil tiene unas dimensiones de 15 mm de ancho por 40,7 mm de alto. La superficie de sensado de cada táctel es de $11,1 \text{ mm}^2$ ($3 \times 3,7 \text{ mm}$) con un diseño ovalado y electrodos con áreas balanceadas.

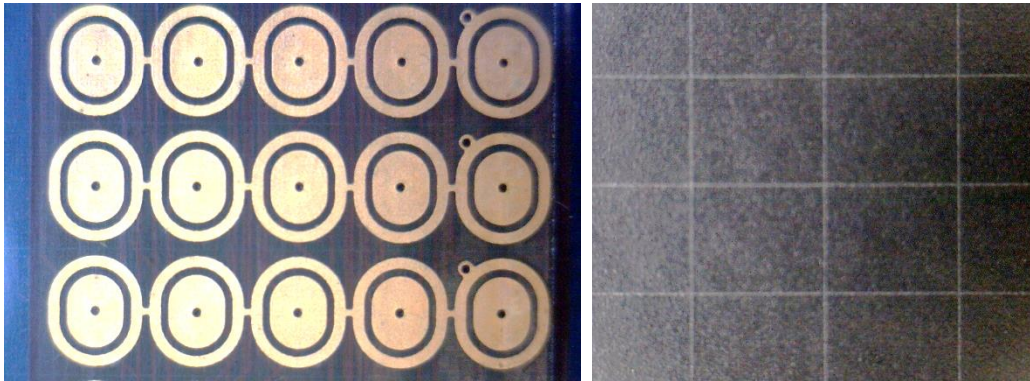


Figura 4-22. Detalles de la matriz de electrodos del sensor de dedo así como del material discreto.

El diseño de la PCB ha requerido del uso de técnicas de diseño y fabricación muy avanzadas. Existen unas reglas muy restrictivas a la hora de diseñar una placa de circuito impreso como la aquí requerida [Fjelstad 2007] que tienen que ver con el tamaño y disposición de las pistas, el diámetro de los taladros, etc. La Figura 4-23 muestra la disposición de las capas de la PCB.

top	50	μ		
FR4	500	μ		
NOFLOW	140	μ	COVERLAY	50
I2	18	μ		18
polyimide	50	μ		50
I3	18	μ		18
NOFLOW	70	μ		
I2	18	μ		18
polyimide	50	μ		50
I3	18	μ		18
NOFLOW	140	μ	COVERLAY	50
FR4	500	μ		
bottom	50	μ		

Polyimide + FR4 tg170	1622	μ	thickness	272
-----------------------	------	---	-----------	-----

Figura 4-23. Capas PCB del sensor de dedo.

Por otro lado, para asegurar una mejor integración del conjunto de la electrónica y la matriz de electrodos con el dedo de la mano de Barrett ha sido necesario realizar un diseño a medida de la pieza del dedo. La Figura 4-24 muestra el sensor integrado en la misma.

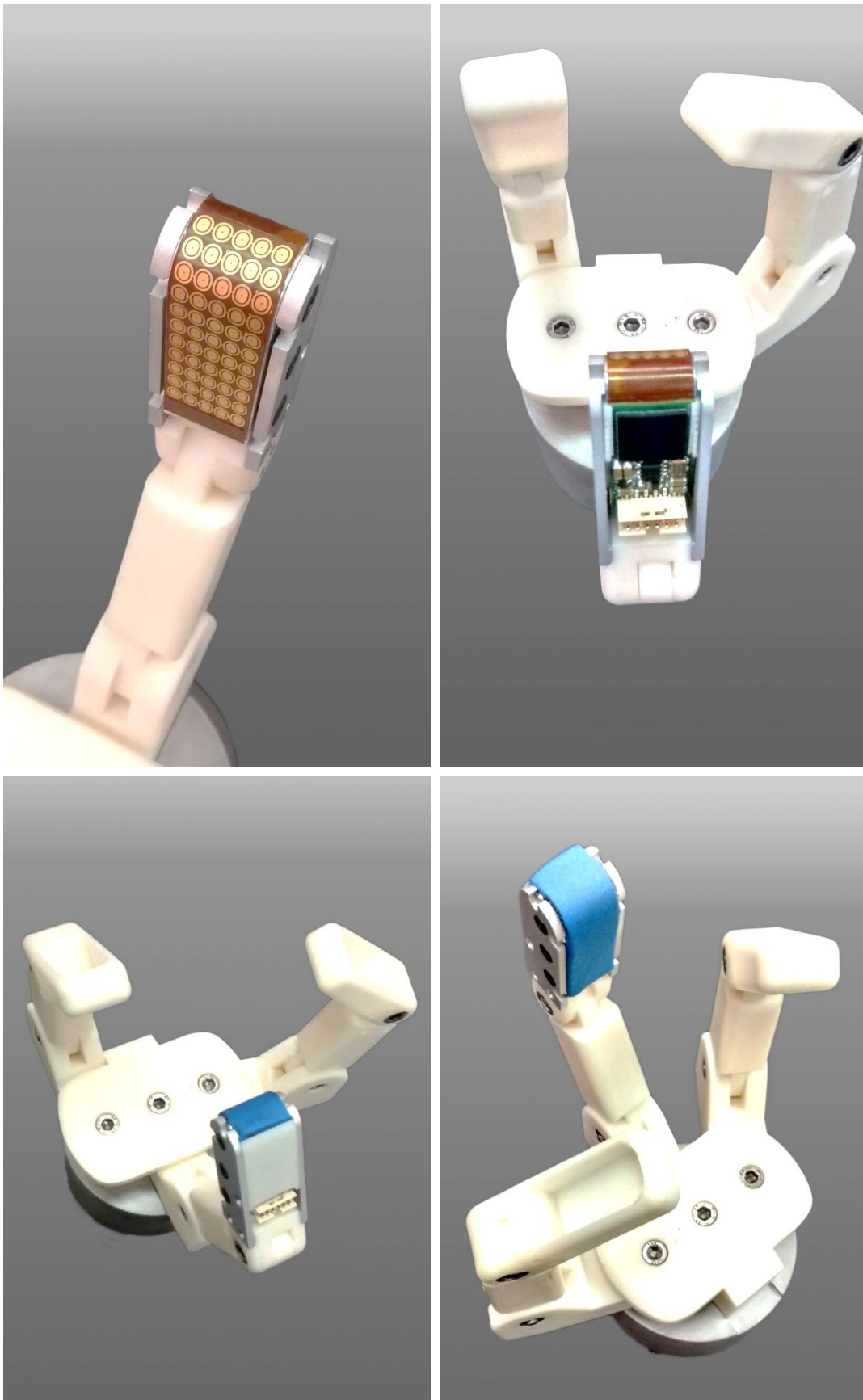


Figura 4-24. Sensor de dedo para la mano de Barrett.

4.7. Pre-procesado: Calculo de los momentos y la elipse de una imagen

La geometría de una región plana se basa en el tamaño, la posición, la orientación y la forma. Todas estas medidas están relacionadas con una familia de parámetros llamada momentos [Mukundan et al. 1998].

Los momentos son propiedades numéricas que se pueden obtener de una determinada imagen. Su uso permite conocer bastante información de la misma y tiene la ventaja de que no solo usa los bordes de la figura, sino que tiene en cuenta todos los tácteles. Se usan principalmente para reconocer una forma en una imagen, por lo que su uso en sensores táctiles tiene infinidad de aplicaciones [Rocha et al. 2004].

4.7.1. Momentos Simples

Se emplean para obtener otros momentos, pero también aportan información por sí mismos. La teoría de los momentos proporciona una interesante y útil alternativa para la representación de formas de objetos. Si se tiene un objeto en una región que viene dada por los puntos en los que $f(x, y) > 0$, definimos el momento de orden $(p+q)$ como:

$$m_{pq} = \iint x^p y^q f(x, y) dx dy \quad (4.1)$$

El teorema de representación de los momentos dice que el conjunto infinito de momentos m_{pq} , con $p, q = 0, 1, \dots$ determinan unívocamente $f(x, y)$ y viceversa. Se parte de una imagen digital definida por la función $f(x, y)$, donde (x, y) son las coordenadas de un táctel y $f(x, y)$, el valor de ese táctel. Al tratarse de imágenes digitales, el momento de orden $(p+q)$ se define ahora como:

$$M_{pq} = \sum_{x=1}^N \sum_{y=1}^M x^p y^q f(x, y) \quad (4.2)$$

donde N es el número de columnas y M el número de filas de la matriz, respectivamente.

Como se deduce de la ecuación (4.2) los momentos se pueden ir calculando a medida que la imagen táctil es obtenida. Cada vez que una fila de la matriz es medida, el resultado parcial de los momentos simples se actualiza sumando al valor anterior el producto de unos coeficientes conocidos por todos los valores de la fila recién obtenida.

La Figura 4-25 muestra el bloque aritmético que se ha implementado para el cálculo de los momentos simples.

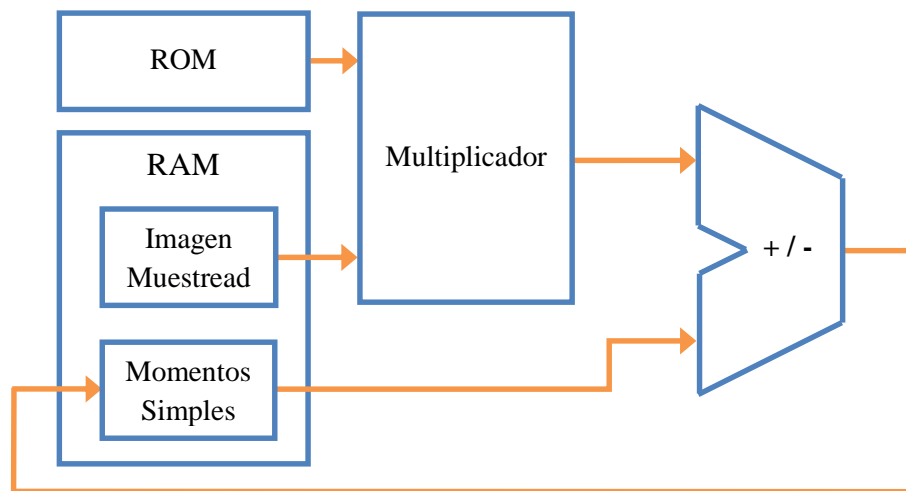


Figura 4-25. Bloque aritmético para el cálculo de los momentos simples.

Como se extrae de (4.2), dos productos son necesarios para el cálculo de los momentos simples. Los términos x^p e y^q no dependen de los valores medidos $f(x,y)$ por lo que el producto de ambos puede ser calculado con anterioridad y guardado como constante en una memoria no volátil. Esto reduce a uno el número de multiplicaciones necesarias para el cálculo de los momentos simples, disminuyendo el hardware (en el caso de una implementación paralela del bloque aritmético) o el tiempo (si los cálculos se realizan en varias etapas) del módulo aritmético. Las FPGAs disponen de bloques de memoria RAM dedicada que pueden ser configurados como de sólo lectura. El contenido de las mismas es definido por el usuario y está incluido en el flujo de bits con el que la FPGA es configurada en su arranque.

En el módulo aritmético aquí propuesto, una memoria ROM almacena el producto de los coeficientes asociados con cada coordenada de la matriz para cada uno de los momentos simples. Así, cada vez que se almacena una nueva fila en la memoria de imágenes, un contador va generando las direcciones de memoria para acceder a cada uno de los táctiles recién medidos, así como a los coeficientes almacenados en la ROM que corresponden con su coordenada. Los datos procedentes de estas dos memorias van a un multiplicador. El resultado de la multiplicación se suma al valor acumulado de su correspondiente momento simple en una zona de memoria dedicada a tal efecto. Cuando se procesa la

última fila, los resultados almacenados en memoria para los momentos simples son los definitivos.

El cálculo de los momentos simples podría realizarse de forma paralela haciendo uso de tantos bloques aritméticos como momentos distintos fuese necesario obtener. En este caso, el cálculo es realizado secuencialmente puesto que se dispone del tiempo empleado por todo un ciclo de medida de una fila de sensores para su procesado.

A continuación se muestran las ecuaciones de los distintos momentos simples:

- Momento simple 00: El coeficiente almacenado en la ROM para todas las coordenadas es 1. El momento M_{00} coincide con el área de la imagen táctil.

$$A = M_{00} = \sum_{x=1}^N \sum_{y=1}^M f(x, y) \quad (4.3)$$

- Momento simple 10: Los coeficientes almacenados en la ROM para cada coordenada (x, y) coinciden con el valor de la columna x .

$$M_{10} = \sum_{x=1}^N \sum_{y=1}^M xf(x, y) \quad (4.4)$$

- Momento simple 01: Los coeficientes almacenados en la ROM para cada coordenada (x, y) coinciden con el valor de la fila y .

$$M_{01} = \sum_{x=1}^N \sum_{y=1}^M yf(x, y) \quad (4.5)$$

- Momento simple 20: Los coeficientes almacenados en la ROM para cada coordenada (x, y) coinciden con el cuadrado del valor de la columna x .

$$M_{20} = \sum_{x=1}^N \sum_{y=1}^M x^2 f(x, y) \quad (4.6)$$

- Momento simple 11: Los coeficientes almacenados en la ROM para cada coordenada (x, y) coinciden con el producto del valor de la columna x por la fila y .

$$M_{11} = \sum_{x=1}^N \sum_{y=1}^M xyf(x, y) \quad (4.7)$$

- Momento simple 02: Los coeficientes almacenados en la ROM para cada coordenada (x, y) coinciden con el cuadrado del valor de la fila y .

$$M_{02} = \sum_{x=1}^N \sum_{y=1}^M y^2 f(x, y) \quad (4.8)$$

4.7.2. Centroide y momentos centrales

El centroide es el centro de masas de una imagen y es utilizado como punto de referencia del objeto. Viene determinado por las coordenadas (X, Y) , de forma que el área de la figura que queda a la derecha e izquierda del punto X es la misma, al igual que el área que queda por encima y por debajo del punto Y . Las coordenadas (X, Y) se pueden expresar en función de los momentos de orden 0 y 1.

$$X = \frac{M_{10}}{M_{00}} \quad (4.9)$$

$$Y = \frac{M_{01}}{M_{00}} \quad (4.10)$$

Por otro lado, los momentos centrales se emplean para el reconocimiento de imágenes independientemente de su situación en un eje de coordenadas. Pueden calcularse con respecto al centroide de la figura:

$$MC_{pq} = \sum_{x=1}^N \sum_{y=1}^M (x-X)^p (y-Y)^q f(x, y) \quad (4.11)$$

Si los momentos simples son conocidos, es menos costoso calcular los momentos centrales a partir de estos últimos que evaluar la ecuación (4.11). Los momentos centrales de orden 0, 1 y 2 se expresan como:

$$MC_{00} = M_{00} \quad (4.12)$$

$$MC_{10} = 0 \quad (4.13)$$

$$MC_{01} = 0 \quad (4.14)$$

$$MC_{20} = M_{20} - \frac{M_{10}^2}{M_{00}} \quad (4.15)$$

$$MC_{11} = M_{11} - \frac{M_{10}M_{01}}{M_{00}} \quad (4.16)$$

$$MC_{02} = M_{02} - \frac{M_{01}^2}{M_{00}} \quad (4.17)$$

Todas las expresiones que definen el centroide y los momentos centrales pueden evaluarse en un único bloque aritmético (Figura 4-26). Un divisor, para calcular el inverso del momento M_{00} , un multiplicador, un sumador/restador y cierta lógica de selección para el encaminamiento de datos, son necesarios.

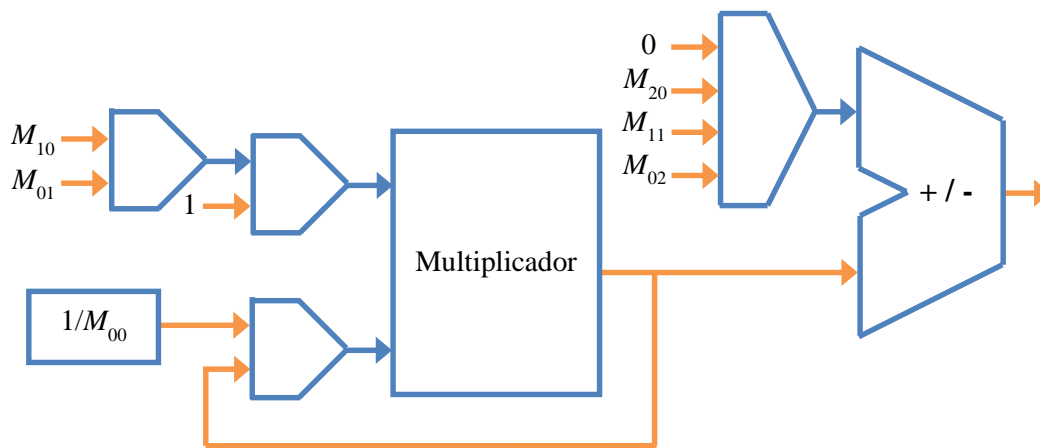


Figura 4-26. Bloque aritmético para el cálculo del centroide y de los momentos centrales.

Los resultados en la salida del sumador/restador son válidos después de dos ciclos de reloj. En una primera etapa se obtiene el producto del inverso del momento M_{00} por uno de los momentos simples de orden 1 (M_{01} o M_{10}). En la segunda, el resultado anterior se multiplica de nuevo por una de entre tres opciones (1, M_{01} o M_{10}) para concluir con una operación de suma o resta, entre 0 o uno de los momentos simples de orden 2 y la salida del multiplicador, según convenga.

A continuación se muestran con mayor detalle las distintas configuraciones que toma el bloque aritmético para el cálculo de las coordenadas del centroide y de los momentos centrales.

- Coordenada X del centroide (Figura 4-27):
 - Etapa 1: $M_{10} * (1/M_{00})$
 - Etapa 2: $0 + (1 * \text{Resultado etapa 1})$

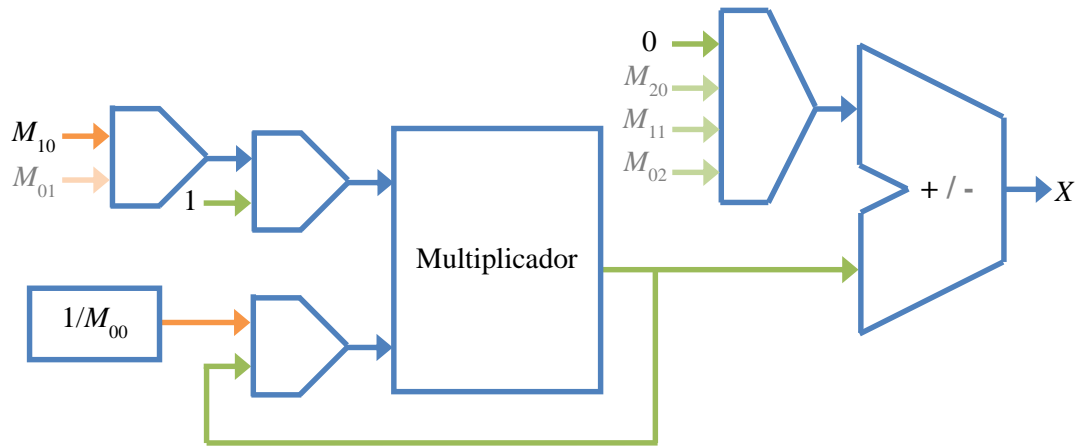


Figura 4-27. Configuración para el cálculo de la coordenada X del centroide.

- Coordenada Y del centroide (Figura 4-28):
 - Etapa 1: $M_{01} * (1/M_{00})$
 - Etapa 2: $0 + (1 * \text{Resultado etapa 1})$

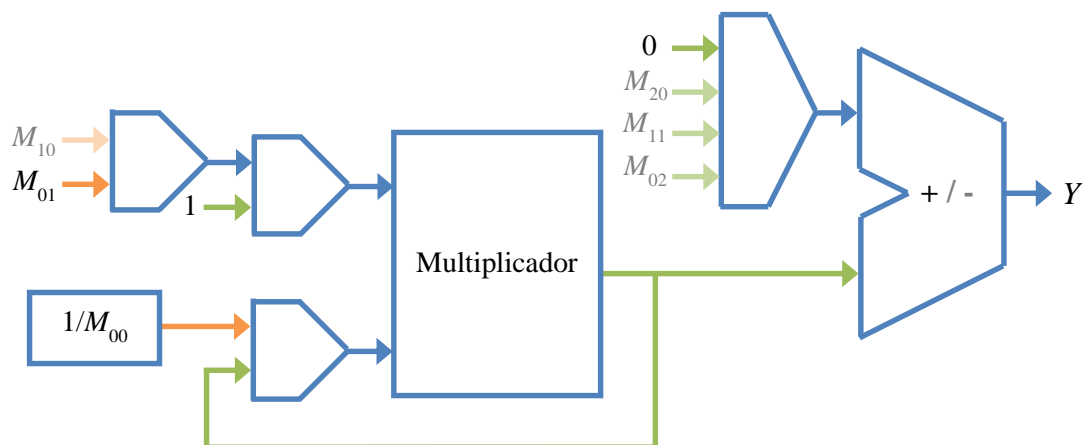


Figura 4-28. Configuración para el cálculo de la coordenada Y del centroide.

- Momento central MC_{20} (Figura 4-29):
 - Etapa 1: $M_{10} * (1/M_{00})$
 - Etapa 2: $M_{20} - (M_{10} * \text{Resultado etapa 1})$

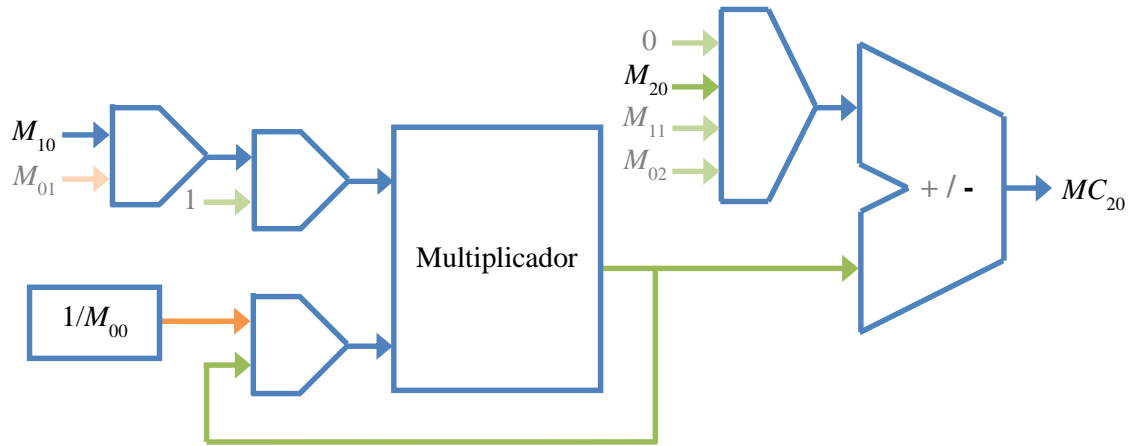


Figura 4-29. Configuración para el cálculo del momento central MC_{20}

- Momento central MC_{11} (Figura 4-30):
 - Etapa 1: $M_{10} * (1/M_{00})$
 - Etapa 2: $M_{11} - (M_{01} * \text{Resultado etapa 1})$

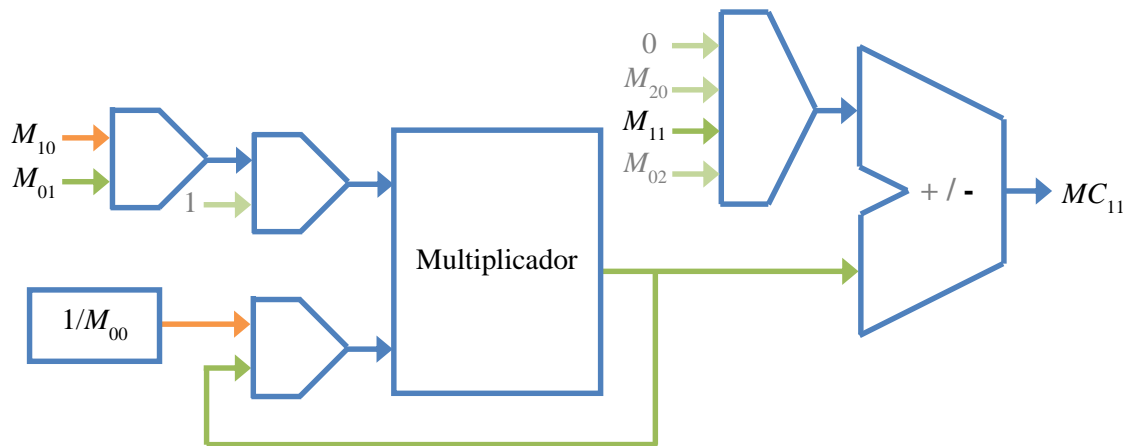


Figura 4-30. Configuración para el cálculo del momento central MC_{11}

- Momento central MC_{02} (Figura 4-31):
 - Etapa 1: $M_{01} * (1/M_{00})$
 - Etapa 2: $M_{02} - (M_{01} * \text{Resultado etapa 1})$

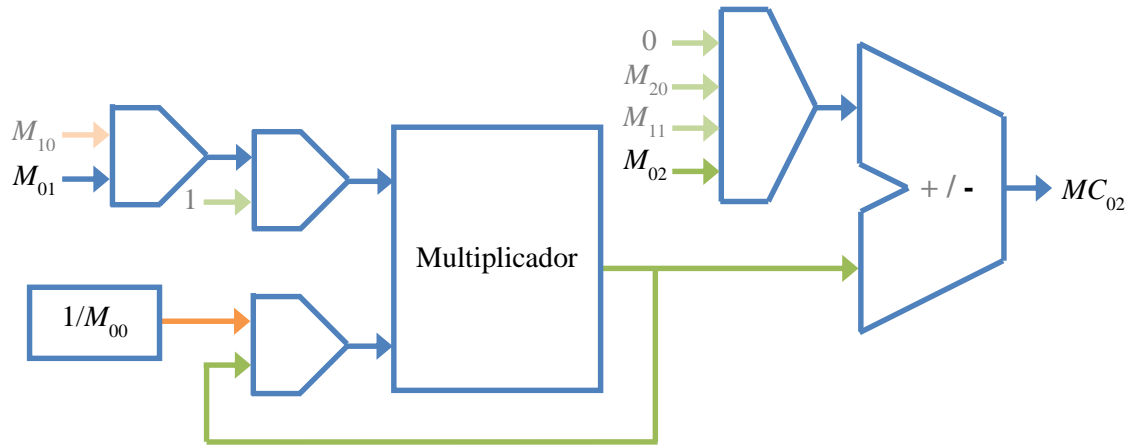


Figura 4-31. Configuración para el cálculo del momento central MC_{02}

4.7.3. Elipse de una imagen

Los momentos de una imagen pueden no ser suficientes para obtener una representación adecuada de la estructura global de un objeto (tamaño, posición y orientación). Para esto se necesita de su elipse de inercia. Esta elipse está centrada en el centroide del objeto y sus ejes son las rectas que pasan por el centroide cuyos momentos centrales de orden 2 son máximos y mínimos respectivamente. Estas rectas son perpendiculares y se corresponden con los vectores propios de la matriz de covarianza del objeto (I_1 , I_2) [Mukundan et al. 1998]. El ángulo θ correspondiente a la orientación del eje mayor y las longitudes (a, b) de los semiejes se pueden obtener de las siguientes expresiones según [Mukundan et al. 1998; Rocha et al. 2004]:

$$a = 2\sqrt{\frac{I_1}{MC_{00}}} \quad (4.18)$$

$$b = 2\sqrt{\frac{I_2}{MC_{00}}} \quad (4.19)$$

$$\theta = \frac{1}{2} \arctan \frac{2MC_{11}}{MC_{20} - MC_{02}} \quad (4.20)$$

donde I_1 e I_2 se definen como:

$$I_1 = \frac{(MC_{20} + MC_{02}) + \sqrt{(MC_{20} - MC_{02})^2 + 4MC_{11}^2}}{2} \quad (4.21)$$

$$I_2 = \frac{(MC_{20} + MC_{02}) - \sqrt{(MC_{20} - MC_{02})^2 + 4MC_{11}^2}}{2} \quad (4.22)$$

A continuación se describe el bloque aritmético que calcula estos parámetros.

Los fabricantes de FPGAs ponen a disposición de los diseñadores, a través de sus herramientas de desarrollo, una serie de bloques IP (*Intellectual Property*) que facilitan y aceleran el proceso de diseño. Estos bloques poseen un gran número de parámetros que permiten un alto grado de personalización dando lugar a implementaciones muy compactas y de alto rendimiento. En este caso, un módulo IP que desarrolla el algoritmo CORDIC (*COordinate Rotation Digital Computer*) es empleado para llevar a cabo, de forma paralela, el cálculo del ángulo de la elipse así como la raíz cuadrada presente en la expresión de los vectores propios. La Figura 4-32 ilustra dicho bloque.

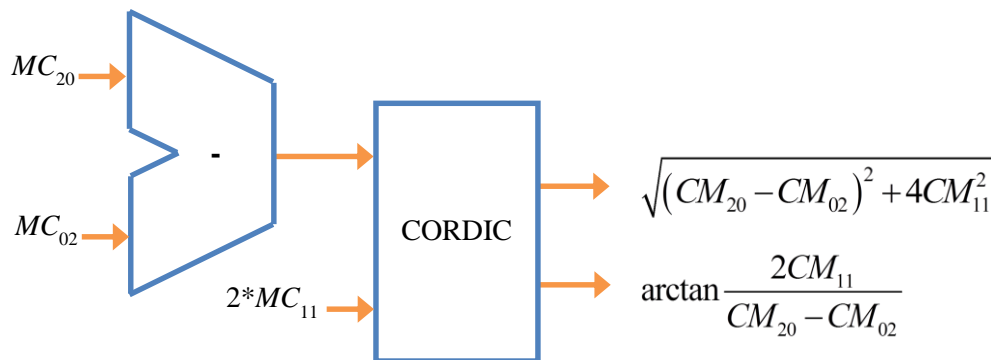


Figura 4-32. Bloque aritmético para el cálculo de la raíz cuadrada y el ángulo

El centroide de la figura así como los momentos centrales y los parámetros de la elipse se calculan cuando la imagen táctil ha sido completada.

4.8. Resultados y discusión

Los resultados de este trabajo demuestran la viabilidad de una arquitectura basada únicamente en FPGAs para la adquisición y pre-procesado a alta velocidad de datos táctiles de un conjunto de sensores montados sobre una mano artificial. El pre-procesado es esencial si las comunicaciones con el sistema host son realizadas a través de buses con un ancho de banda bajo (por ejemplo busCAN) porque sólo unos pocos bytes pueden ser enviados en cada mensaje para garantizar una latencia máxima y trabajar en tiempo real. Otros buses de alta velocidad como USB 2.0 o EtherCAT comienzan a emplearse, consiguiendo un alto rendimiento que los capacitan para trabajar en tiempo real. En este caso, todos los datos en bruto podrían ser enviados a la unidad central, pero el sistema host tendría que hacer frente al procesamiento de todos los datos táctiles, además de otros procedimientos relativos a otros sensores y al envío de comandos a los actuadores. Por esto, la respuesta a cuánto procesamiento tiene que ser realizado a nivel local y cuánto debe hacerse en la unidad central no es trivial.

Por otro lado, los sensores táctiles son muy propensos a errores, por lo que implementar procedimientos de corrección se hace obligatorio. Así, los sensores inteligentes pueden realizar cierto procesamiento de calibración y compensación de errores.

La arquitectura aquí propuesta está compuesta por un conjunto de sensores táctiles inteligentes que se comunican con un módulo de control a través de buses SPI de alta velocidad. Las dos implementaciones basadas en ella son capaces de escanear todas las matrices táctiles, calculando los parámetros de las elipses que se adapten a dichas imágenes táctiles, a una velocidad de más de 200 frames por segundo (más de 380 fps para el prototipo de la mano de Barrett). Esta tasa de adquisición y pre-procesado puede incluso permitir la detección del deslizamiento, tarea que, por otro lado, requiere de los más altos requerimientos dinámicos en aplicaciones de manipulación [*Maldonado-Lopez et al. 2009*].

La Figura 4-33 muestra un cronograma de una simulación del circuito de pre-procesado descrito en el apartado 4.7.

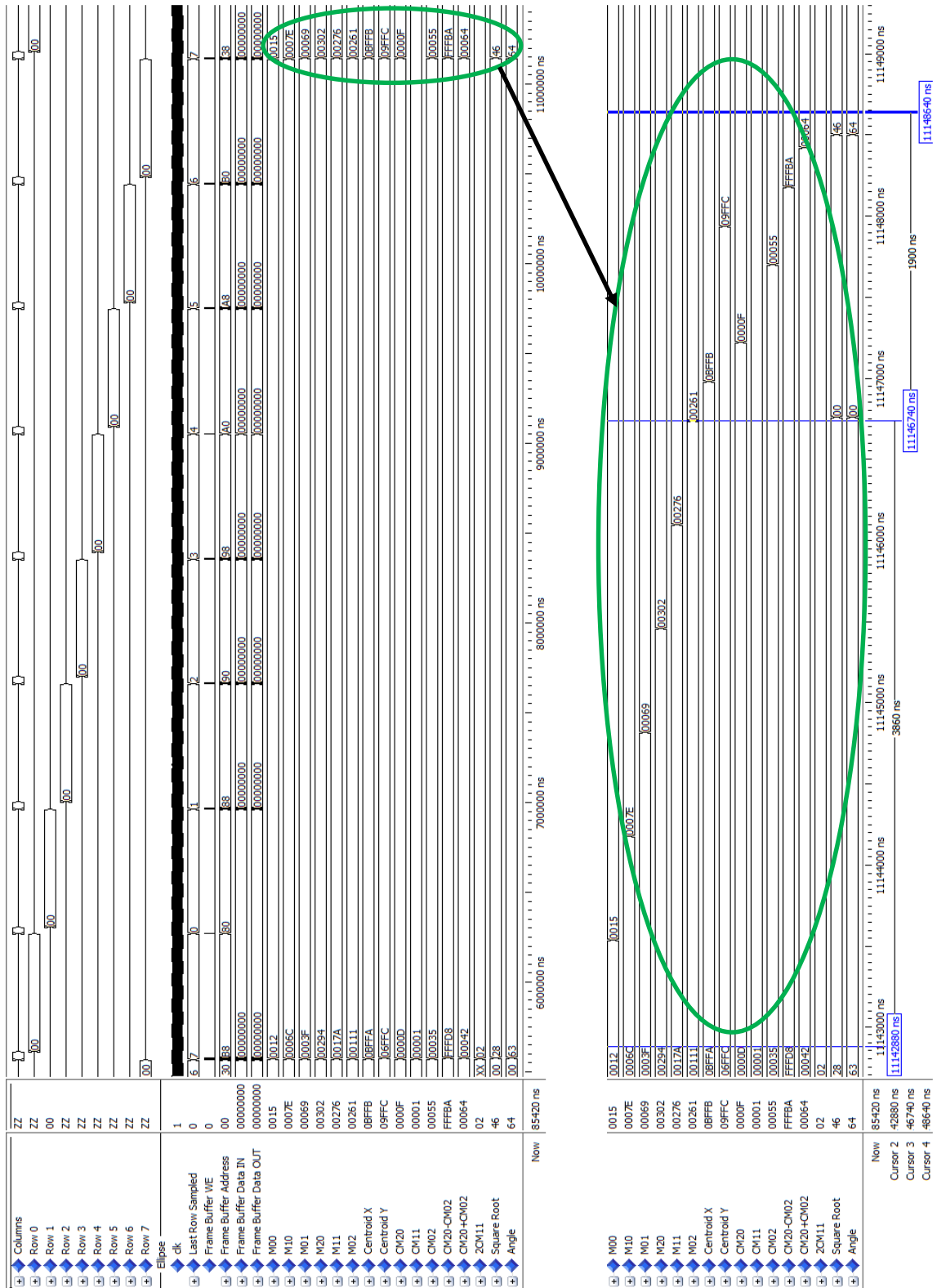


Figura 4-33. Cronograma de la simulación del circuito de pre-procesado.

En ésta se observa cómo el módulo aritmético, que calcula los datos parciales que terminan resultando en los momentos simples tras el escaneo de la última fila, emplea $3,86 \mu\text{s}$ para llevar a cabo estas operaciones. El resto de parámetros de la elipse son obtenidos en $1,9 \mu\text{s}$ cuando el total de la imagen es capturada. Estos tiempos son muy inferiores al tiempo de carga de los condensadores, lo que confirma que toda la actividad debida al pre-procesado se produce antes de que concluya esta etapa, no influyendo, por tanto, en el ruido debido a la actividad de la FPGA durante la fase de descarga.

La Figura 4-34 muestra un ejemplo de la información proporcionada por el sistema táctil.

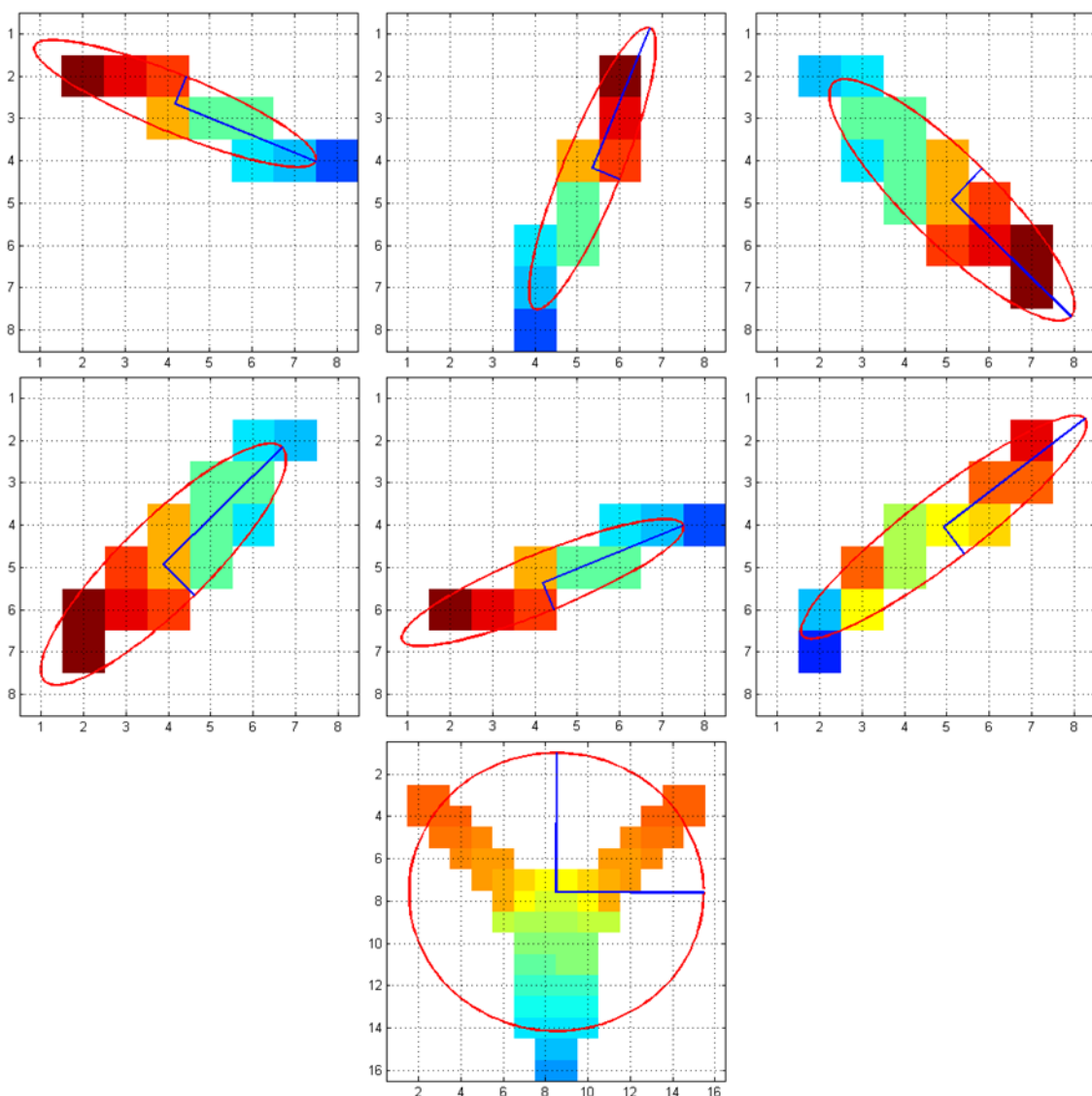


Figura 4-34. Imagen táctil del sistema táctil con las elipses calculadas de forma local.

En cuanto a datos de consumo, cada uno de los sensores de dedo del prototipo de la mano de Barrett consume 275 mW (55 mA @ 5 V) y 0,9 W el conjunto de sensor de palma + controlador (180mA @ 5 V). Así, el sistema táctil completo consume 1,73 W.

Por otro lado, la resolución y la exactitud en la medida de matrices de sensores táctiles piezoresistivos se han evaluado, sobre los circuitos aquí descritos, en el capítulo 3. Para la conexión directa propuesta para los sensores empleados en los dedos se consigue una resolución de 10 bits, con un máximo error relativo del 0,066%. En el caso del sensor de palma, se consiguen 8 bits de resolución y un máximo error relativo del 0,77%.

También cabe destacar que para el uso del sistema táctil en entornos robóticos se ha desarrollado un driver para el framework ROS (Robot Operating System).

Para finalizar y a modo de resumen, la Tabla 4-1 compara soluciones táctiles empleadas por manos robóticas comunes, la mayoría comerciales, con el sistema táctil presentado en este trabajo. Cabe destacar que nuestra propuesta proporciona no sólo la imagen táctil, sino los datos del pre-procesado descrito relativos a la elipse asociada a la imagen, sin penalizar la tasa de muestreo. Este pre-procesado se puede ampliar para incluir auto-calibración y corrección de errores y primitivas que ayuden en tareas de manipulación, por ejemplo, igualmente sin penalizar la tasa de muestreo gracias a la capacidad de adquisición y procesado en paralelo, lo que da un gran potencial a la propuesta.

Tabla 4-1. Comparativa entre distintas soluciones táctiles para manos robóticas.

Mano robótica	Sensor táctil	Nº Tácteles	Resolución Espacial	Razón
Barret Hand	PPS RoboTouch	46 x 3 dedos + 24 palma	6 mm 10 mm	30 Hz
Shunk SDH	Weiss Robotics	(14×6) y (14×7) x 3 dedos	3,5 mm	40 Hz
Shadow Hand	Tekscan	349	4 mm	20 Hz
Gifu Hand III	Gifu Hand sensor	126 x 4 dedos 105 pulgar 313 palma	4 mm	100 Hz
iCub Robot	iCub sensor	12 x 5 dedos + 48 palma	7 mm	25 Hz
Barret Hand	Propuesta	55 x 3 dedos + 160 palma	3,7 mm 4,5 mm	310 Hz

Conclusiones y líneas futuras

Conclusiones

Las principales aportaciones de esta tesis doctoral son:

- El diseño de un módulo de captura para la medida del tiempo de descarga de un condensador que permite inferir el valor de un sensor resistivo a través de una conexión directa entre éste y una FPGA. El módulo propuesto reduce la incertidumbre debida al ruido sin hacer uso de *buffers* tipo Schmitt Trigger externos. Además, implementa estrategias novedosas que mejoran la resolución, básicamente gracias a promedios, que no penalizan el tiempo de ejecución. Para ello se explota la versatilidad de los elementos de almacenamiento de las FPGAs (captura por nivel o por flanco, biestables disparados por doble flanco).
- Un procedimiento para la estimación del valor de resistencia de un sensor a partir de medidas de una única resistencia de calibración, que compensa los errores debidos a las resistencias internas de los *buffers* de la FPGA.
- La propuesta de conexión directa sensor-FPGA para realizar la adquisición de datos de sensores matriciales resistivos, y en concreto de sensores táctiles. En particular, se proponen dos técnicas de conexión directa entre matrices de sensores y FPGAs:
 - La primera, más indicada para matrices de pequeño o mediano tamaño, lleva a cabo en paralelo tantas medidas de resistencia como columnas posea la matriz, utilizando para el direccionamiento de filas tantas conexiones como sensores individuales contenga, evitando así el efecto del *crosstalk* resistivo entre sensores. No se necesitan elementos externos para la conexión directa.

- La segunda, pensada para sensores de gran tamaño o para aquellos donde la organización de la matriz comparta conexiones para las filas y las columnas, realiza las medidas en paralelo de las resistencias de una fila empleando el principio de tierra virtual para reducir el *crosstalk* entre ellas.
- Estudio de los efectos de segundo orden en los circuitos de conexión directa matriz de sensores-FPGA. Tanto en la conexión directa basada en un integrador pasivo, como en la basada en un integrador activo aparecen fenómenos de segundo orden que influyen en la medida de tiempo y que condicionan el rango de resistencias que los sensores a medir pueden tomar, así como la precisión y exactitud de la medida. Se presentan métodos de estimación que tienen en cuenta dichos fenómenos.
- La caracterización de ambas propuestas, obteniendo la resolución y la exactitud en la medida de las resistencias.
- La propuesta e implementación de una arquitectura de un sistema táctil basado en los circuitos anteriormente propuestos. El sistema emplea comunicaciones serie SPI para las comunicaciones locales entre el módulo de control y los distintos sensores. En cada sensor se ha implementado un algoritmo que extrae los momentos geométricos de la imagen táctil capturada, como ejemplo de las capacidades de pre-procesado local. Para las pruebas funcionales y de caracterización se ha fabricado un prototipo con una unidad de control y seis sensores táctiles, con tasas de muestreo que superan las 200 imágenes táctiles por segundo.
- Por último, se ha desarrollado un prototipo de un sistema de sensores táctiles para la mano robótica de Barrett. En este caso, el sistema táctil está compuesto por una unidad de control y cuatro sensores (tres para los dedos y uno para la palma).

Líneas futuras

Algunas líneas futuras de investigación que parten de los trabajos aquí presentados son:

- Nuevos circuitos para la cancelación del *crosstalk* en matrices táctiles de gran área.

- Implementación de algoritmos de pre-procesado para compensación de errores en sensores táctiles (deriva, histéresis, etc).
- Implementación de algoritmos de pre-procesado relacionados con tareas como apoyo a la manipulación, reconocimiento de objetos, de texturas, etc.
- Desarrollo de nuevos prototipos para distintas aplicaciones, y en concreto para las robóticas y en especial las orientadas a aplicaciones biomédicas.

Publicaciones que avalan la Tesis

1. Publicación en Revista:

Título: Smart Capture Modules for Direct Sensor-to-FPGA Interfaces.

Autores: Óscar Oballe-Peinado, Fernando Vidal-Verdú, José A. Sánchez-Durán, Julián Castellanos-Ramos, José A. Hidalgo-López.

Revista: Sensors.

Fecha: 16 de diciembre de 2015.

Indicios de calidad:

Categoría: Q1 en “Instruments & Instrumentation”. Índice de impacto (1 año): 2.245.

2. Publicación en Revista:

Título: Three Realizations and Comparison of Hardware for Piezoresistive Tactile.

Autores: Fernando Vidal-Verdú, Óscar Oballe-Peinado, José A. Sánchez-Durán, Julián Castellanos-Ramos, Rafael Navas-González.

Revista: Sensors, volumen 11, número 3, páginas 3249-3266.

Fecha: 17 de marzo de 2011.

Indicios de calidad:

Categoría: Q1 en “Instruments & Instrumentation”. Índice de impacto (1 año): 2.245.

3. Publicación en Revista:

Título: Tactile sensor hardware based on programmable SoC

Autores: Fernando Vidal-Verdú, David Bravo-Gálvez, Óscar Oballe-Peinado.

Revista: Electronics Letters, volumen 46, número 16, páginas 1117-1118.

Fecha: 5 de agosto de 2010.

Indicios de calidad:

Categoría: Q3 en “Engineering, Electrical & Electronic”. Índice de impacto (1 año): 0,93.

4. Artículo en Conferencia:

Título: Architecture of a tactile sensor suite for artificial hands based on FPGAs.

Autores: Óscar Oballe-Peinado, José A. Hidalgo-López, José A. Sánchez-Durán, Julián Castellanos-Ramos, Fernando Vidal-Verdú.

Conferencia: The Fourth IEEE RAS/EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob).

Lugar: Roma, Italia.

Fecha: 24-27 de junio de 2012.

5. Artículo en Conferencia:

Título: Hardware for Piezoresistive Tactile Sensors.

Autores: Fernando Vidal-Verdú, Óscar Oballe, María José Barquero, David Bravo, José Antonio Sánchez Durán.

Conferencia: Humanoids 09 - The 9th IEEE-RAS International Conference on Humanoid Robots, Workshop on Tactile Sensing in Humanoids.

Lugar: París, Francia.

Fecha: 7 de diciembre de 2009.

6. Artículo en Conferencia:

Título: Direct interfaces for smart skins based on FPGAs.

Autores: Óscar Oballe-Peinado, Julián Castellanos-Ramos, José A. Hidalgo-López, Fernando Vidal-Verdú

Conferencia: Bioengineered and Bioinspired Systems IV.

Lugar: Dresden, Alemania.

Fecha: 4 de mayo de 2009.

7. Artículo en Conferencia:

Título: Interface for tactile sensors based on direct connection to a FPGA.

Autores: Óscar Oballe-Peinado, Julián Castellanos-Ramos, José A. Hidalgo-López,
Fernando Vidal-Verdú, H. Macicior, Estíbaliz Ochoteco

Conferencia: IEEE International Conference on Mechatronics.

Lugar: Málaga, España.

Fecha: 14-17 de abril de 2009.

Anexo A. Módulo de captura

Código VHDL del Módulo de captura 6 (MC6) - Circuito de detección de inicio y final de tren de pulsos de la señal de descarga por flancos de subida y de bajada, promediando cuatro valores de cuenta.

```

-----
-- Company:      Universidad de Malaga
-- Engineer:     Oscar Oballe Peinado
--
-- Create Date:  16:16:45 10/02/2015
-- Design Name:  Sensor Pasivo
-- Module Name:  tactel pasivo - Behavioral
-- Project Name: Sensor_Pasivo
-- Target Devices: xc3s50an-4tqg144
-- Tool versions: ISE 14.7, Modelsim SE 6.5
-- Description:
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
-----

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

library UNISIM;
use UNISIM.VComponents.all;

entity tactel_pasivo is
  port (
    clk                : in std_logic;
    estado_medida      : in std_logic_vector(2 downto 0);
    pin_carga_pos      : inout std_logic;
    rg_med              : out std_logic_vector(7 downto 0);
    rg_low              : out std_logic_vector(7 downto 0);
    end_of_discharge   : out std_logic
  );
end tactel_pasivo;

architecture behavioral of tactel_pasivo is

  signal pin_medida      : std_logic;
  signal descarga        : std_logic;
  signal descarga_ret    : std_logic := '0';

  signal timeout         : std_logic;
  signal contador_descarga : std_logic_vector(15 downto 0) := "0000000000000000";

  signal pin_medida_latch_inicio_neg : std_logic := '1';

```

```

signal pin_medida_latch_inicio_pos      : std_logic := '1';
signal pin_medida_inicio_tren_limpia    : std_logic := '1';
signal pin_medida_inicio_tren_limpia_ret : std_logic := '1';
signal guarda_medida_inicio_tren       : std_logic;
signal registro_inicio_tren             : std_logic_vector(15 downto 0);
signal pin_medida_inicio_glitch_limpia  : std_logic := '1';
signal pin_medida_inicio_glitch_limpia_ret : std_logic := '1';
signal guarda_medida_inicio_glitch     : std_logic;
signal registro_inicio_glitch          : std_logic_vector(15 downto 0);

signal pin_medida_latch_final_neg       : std_logic := '1';
signal pin_medida_latch_final_pos       : std_logic := '1';
signal pin_medida_final_tren_limpia     : std_logic := '0';
signal pin_medida_final_tren_limpia_ret : std_logic := '0';
signal guarda_medida_final_tren       : std_logic;
signal registro_final_tren             : std_logic_vector(15 downto 0);
signal pin_medida_final_glitch_limpia   : std_logic := '0';
signal pin_medida_final_glitch_limpia_ret : std_logic := '0';
signal guarda_medida_final_glitch     : std_logic;
signal registro_final_glitch           : std_logic_vector(15 downto 0);

signal guarda_medida_ret      : std_logic := '0';
signal flag_guardea_medida     : std_logic := '0';

signal suma_inicio  : std_logic_vector(15 downto 0);
signal suma_final   : std_logic_vector(15 downto 0);
signal suma         : std_logic_vector(15 downto 0);

signal pin_medida_not      : std_logic;
signal pin_medida_not_bufg : std_logic;
signal pin_medida_bufg     : std_logic;
signal clk_latch_neg       : std_logic := '0';
signal clk_latch_pos       : std_logic := '0';
signal clk_latch           : std_logic;
signal clk_latch_not       : std_logic;

```

begin

```

-----
-- Triestado pin_carga_pos                               --
-- Durante la carga (estado_medida = "010") pone "1" para cargar el terminal --
-- positivo del condensador                             --
-- Durante el resto de estados se configura como entrada siendo el pin por --
-- donde se mide la descarga del condensador (el cambio de "1" a "0")      --
-----
pin_carga_pos <= '1' when estado_medida = "010" else 'Z';
pin_medida <= pin_carga_pos;

descarga <= '0' when estado_medida = "100" else '1';

pin_medida_not <= not(pin_medida);
BUFG_inst1 : BUFG
port map (
    0 => pin_medida_not_bufg,
    I => pin_medida_not
);

pin_medida_bufg <= not(pin_medida_not_bufg);

LDPE_inst_neg_inicio : LDPE
generic map (
    INIT => '1') -
port map (
    Q => pin_medida_latch_inicio_neg,
    PRE => clk_latch_not,
    D => descarga,
    G => pin_medida_not_bufg,
    GE => '1'
);

LDPE_inst_pos_inicio : LDPE
generic map (
    INIT => '1')
port map (
    Q => pin_medida_latch_inicio_pos,

```

```

PRE => clk_latch,
D => descarga,
G => pin_medida_not_bufg,
GE => '1'
);

LDPE_inst_neg_final : LDPE
generic map (
  INIT => '1')
port map (
  Q => pin_medida_latch_final_neg,
  PRE => clk_latch_not,
  D => descarga,
  G => pin_medida_bufg,
  GE => '1'
);

LDPE_inst_pos_final : LDPE
generic map (
  INIT => '1')
port map (
  Q => pin_medida_latch_final_pos,
  PRE => clk_latch,
  D => descarga,
  G => pin_medida_bufg,
  GE => '1'
);

process (clk)
begin
  if (clk'event and clk = '0') then
    clk_latch_neg <= not(clk_latch_neg);
  end if;
end process;

process (clk)
begin
  if (clk'event and clk = '1') then
    clk_latch_pos <= not(clk_latch_neg);
  end if;
end process;

clk_latch      <= clk_latch_neg xor clk_latch_pos;
clk_latch_not  <= clk_latch_neg xnor clk_latch_pos;

-----
-- Contador de descarga                                     --
-- Cuenta los ciclos durante la fase de descarga (estado medida = "010") --
-- mientras no se detecte un paso de "1" A "0" o un timeout (end of discharge) --
-- Se resetea durante la fase de carga                     --
-----

process (clk)
begin
  if (clk'event and clk = '1') then
    if (descarga_ret = '0') then
      contador_descarga <= (others => '0');
    elsif (descarga_ret = '1' and timeout = '0') then
      contador_descarga <= contador_descarga + 1;
    end if;
  end if;
end process;

process (clk)
begin
  if (clk'event and clk = '1') then
    descarga_ret <= not(descarga);
  end if;
end process;

-----
-- Define el timeout del contador                         --
-----

timeout <= contador_descarga(14);
end_of_discharge <= timeout;

```

```

-----
-- Proceso para detectar el cambio de "1" a "0" en el pin de descarga      --
-- Se registra para evitar que el ruido (glitches) puedan falsear la medida --
-- Solo tiene sentido si no se ha producido un timeout previamente          --
-- Genera un pulso cuando se dan todas estas condiciones                    --
-----

process (clk)
begin
  if (clk'event and clk = '0') then
    if (estado_medida = "010") then
      pin_medida_inicio_glitch_limpia <= '1';
    else
      pin_medida_inicio_glitch_limpia <= pin_medida_latch_inicio_neg;
    end if;
  end if;
end process;

process (clk)
begin
  if (clk'event and clk = '1') then
    if (estado_medida = "010") then
      pin_medida_inicio_glitch_limpia_ret <= '1';
    else
      pin_medida_inicio_glitch_limpia_ret <= pin_medida_inicio_glitch_limpia;
    end if;
  end if;
end process;

-----
-- Define el pulso para el CE del registro de medida                        --
-----
guarda_medida_inicio_glitch <= not(pin_medida_inicio_glitch_limpia) and
pin_medida_inicio_glitch_limpia_ret;

process (clk)
begin
  if (clk'event and clk = '1') then
    if (estado_medida = "010") then
      pin_medida_inicio_tren_limpia <= '1';
    else
      pin_medida_inicio_tren_limpia <= pin_medida_latch_inicio_pos;
    end if;
  end if;
end process;

process (clk)
begin
  if (clk'event and clk = '1') then
    if (estado_medida = "010") then
      pin_medida_inicio_tren_limpia_ret <= '1';
    else
      pin_medida_inicio_tren_limpia_ret <= pin_medida_inicio_tren_limpia;
    end if;
  end if;
end process;

-----
-- Define el pulso para el CE del registro de medida                        --
-----
guarda_medida_inicio_tren <= not(pin_medida_inicio_tren_limpia) and
pin_medida_inicio_tren_limpia_ret;

-----
-- Proceso para detectar el cambio de "1" a "0" en el pin de descarga      --
-- Se registra para evitar que el ruido (glitches) puedan falsear la medida --
-- Solo tiene sentido si no se ha producido un timeout previamente          --
-- Genera un pulso cuando se dan todas estas condiciones                    --
-----

process (clk)
begin
  if (clk'event and clk = '0') then
    if (estado_medida = "010" or descarga_ret = '0') then
      pin_medida_final_tren_limpia <= '0';
    else
      pin_medida_final_tren_limpia <= pin_medida_latch_final_neg;
    end if;
  end if;
end process;

```

```

        end if;
    end if;
end process;

process (clk)
begin
    if (clk'event and clk = '1') then
        if (estado_medida = "010" or descarga_ret = '0') then
            pin_medida_final_tren_limpia_ret <= '0';
        else
            pin_medida_final_tren_limpia_ret <= pin_medida_final_tren_limpia or
pin_medida_final_tren_limpia_ret;
        end if;
    end if;
end process;

-----
-- Define el pulso para el CE del registro de medida --
-----
guarda_medida_final_tren <= pin_medida_final_tren_limpia and
not(pin_medida_final_tren_limpia_ret);

process (clk)
begin
    if (clk'event and clk = '1') then
        if (estado_medida = "010" or descarga_ret = '0') then
            pin_medida_final_glitch_limpia <= '0';
        else
            pin_medida_final_glitch_limpia <= pin_medida_latch_final_pos;
        end if;
    end if;
end process;

process (clk)
begin
    if (clk'event and clk = '1') then
        if (estado_medida = "010" or descarga_ret = '0') then
            pin_medida_final_glitch_limpia_ret <= '0';
        else
            pin_medida_final_glitch_limpia_ret <= pin_medida_final_glitch_limpia or
pin_medida_final_glitch_limpia_ret;
        end if;
    end if;
end process;

-----
-- Define el pulso para el CE del registro de medida --
-----
guarda_medida_final_glitch <= pin_medida_final_glitch_limpia and
not(pin_medida_final_glitch_limpia_ret);

process (clk)
begin
    if (clk'event and clk = '1') then
        if (descarga = '0' and descarga_ret = '0') then
            flag_guarda_medida <= '0';
        elsif (estado_medida = "100" and flag_guarda_medida = '0') then
            flag_guarda_medida <= guarda_medida_ret;
        end if;
    end if;
end process;

process (clk)
begin
    if (clk'event and clk = '1') then
        guarda_medida_ret <= guarda_medida_inicio_tren or
guarda_medida_inicio_glitch or guarda_medida_final_tren or guarda_medida_final_glitch;
    end if;
end process;

process (flag_guarda_medida, suma)
begin
    if (flag_guarda_medida = '0') then
        rg_med <= x"FF";
        rg_low <= (others => '1');
    end if;
end process;

```

```

    else
        rg_med  <= suma(15 downto 8);
        rg_low  <= suma(7  downto 0);
    end if;
end process;

-----
-- Registros inicio
-----

process (clk)
begin
    if (clk'event and clk = '1') then
        if guarda_medida_inicio_glitch = '1' then
            registro_inicio_glitch <= contador_descarga;
        end if;
    end if;
end process;

process (clk)
begin
    if (clk'event and clk = '1') then
        if guarda_medida_inicio_tren = '1' then
            registro_inicio_tren <= contador_descarga;
        end if;
    end if;
end process;

-----
-- Registros final
-----

process (clk)
begin
    if (clk'event and clk = '1') then
        if guarda_medida_final_tren = '1' then
            registro_final_tren <= contador_descarga;
        end if;
    end if;
end process;

process (clk)
begin
    if (clk'event and clk = '1') then
        if guarda_medida_final_glitch = '1' then
            registro_final_glitch <= contador_descarga;
        end if;
    end if;
end process;

suma_inicio <= registro_inicio_glitch + registro_inicio_tren;
suma_final  <= registro_final_tren + registro_final_glitch;
suma        <= suma_inicio + suma_final;

end behavioral;

```

Anexo B. Esquemáticos

A continuación se presentan los esquemáticos de los circuitos para el prototipo de la mano de Barrett.

El circuito del sensor de la palma y el módulo de control está descrito desde la Figura B-1 a la Figura B-12. En el caso del sensor del dedo, los esquemas están representados desde la Figura B-13 a la Figura B-20.

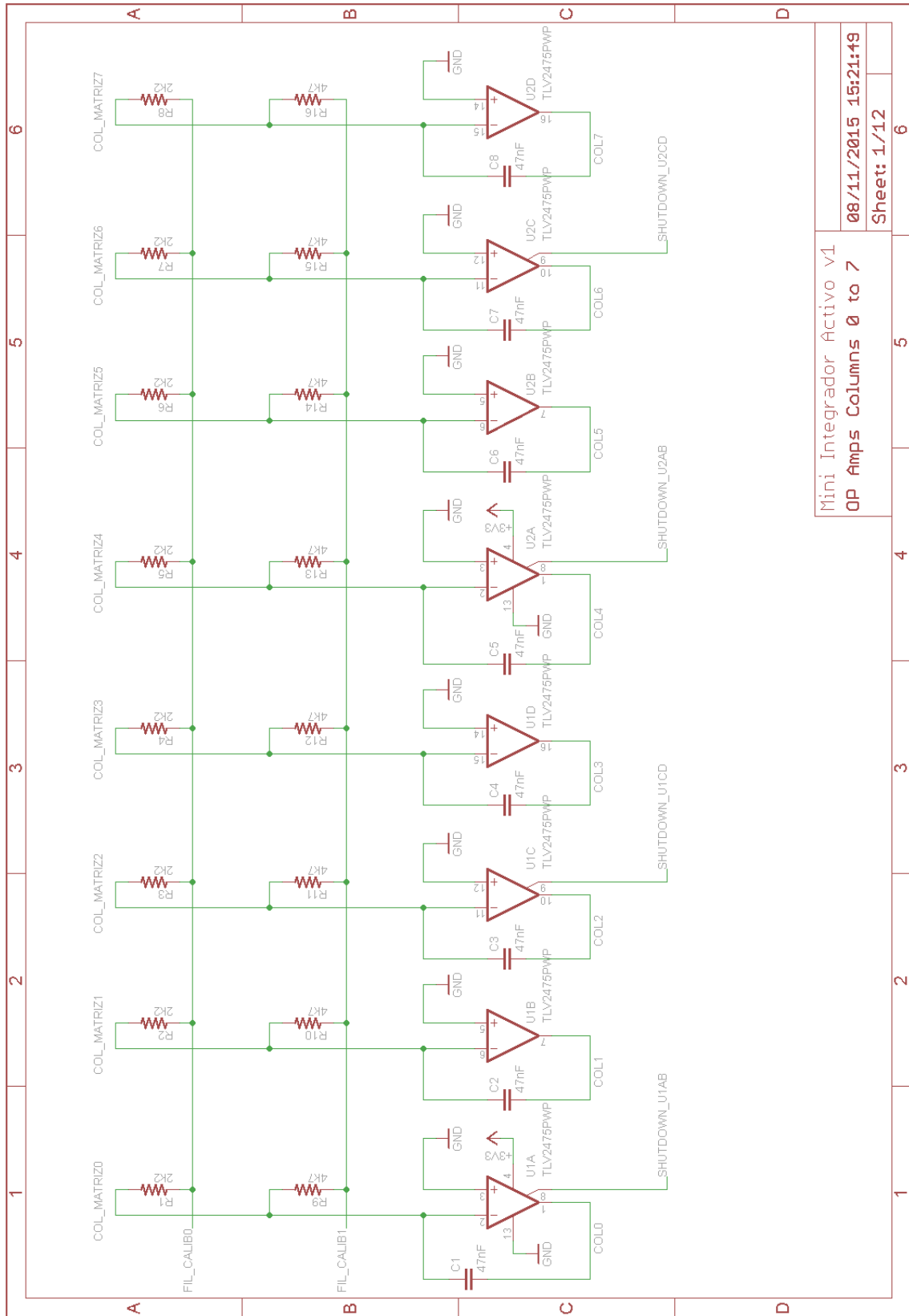


Figura B-1. Mini Integrador Activo – Amplificadores Operacionales (1/4)

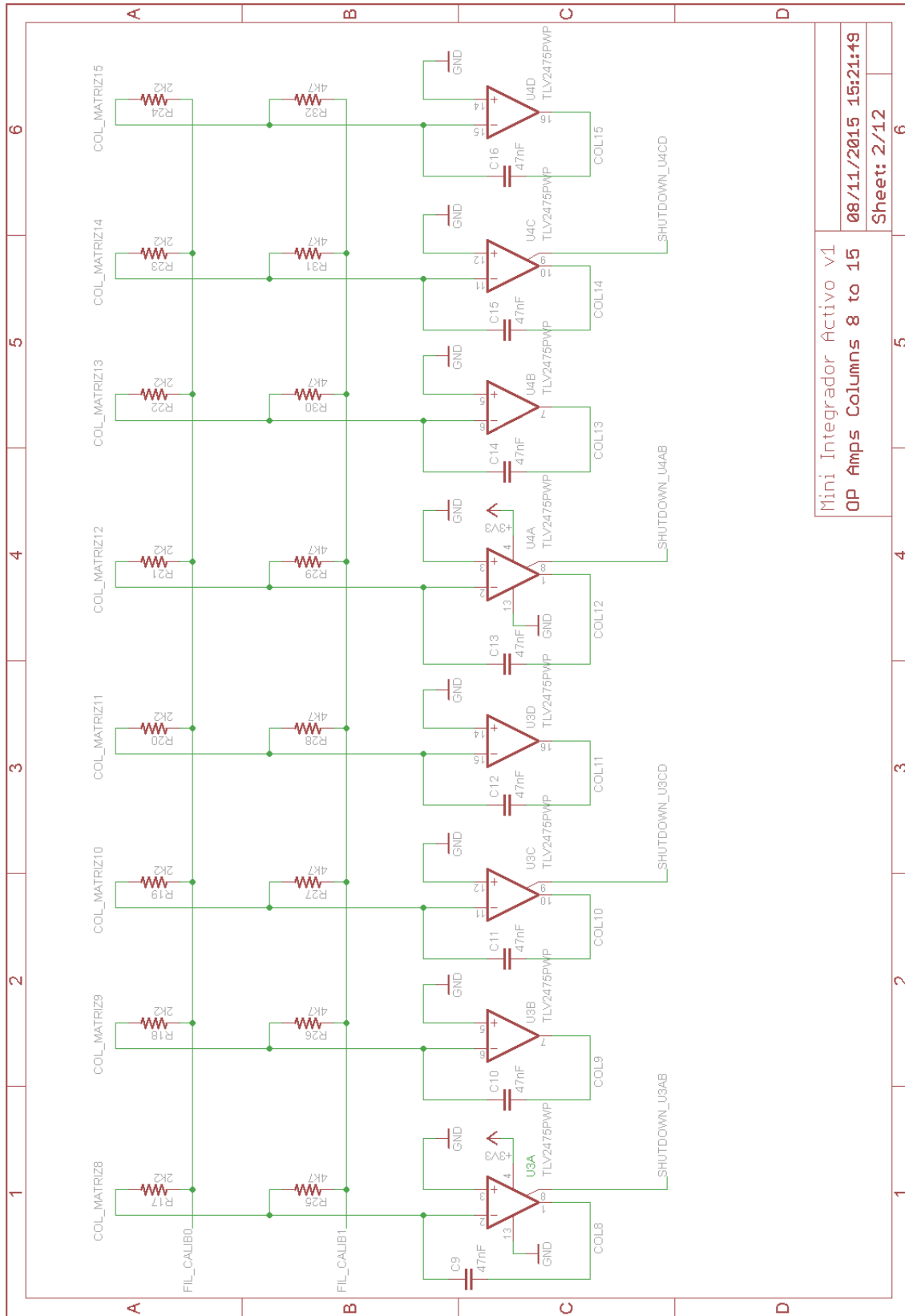
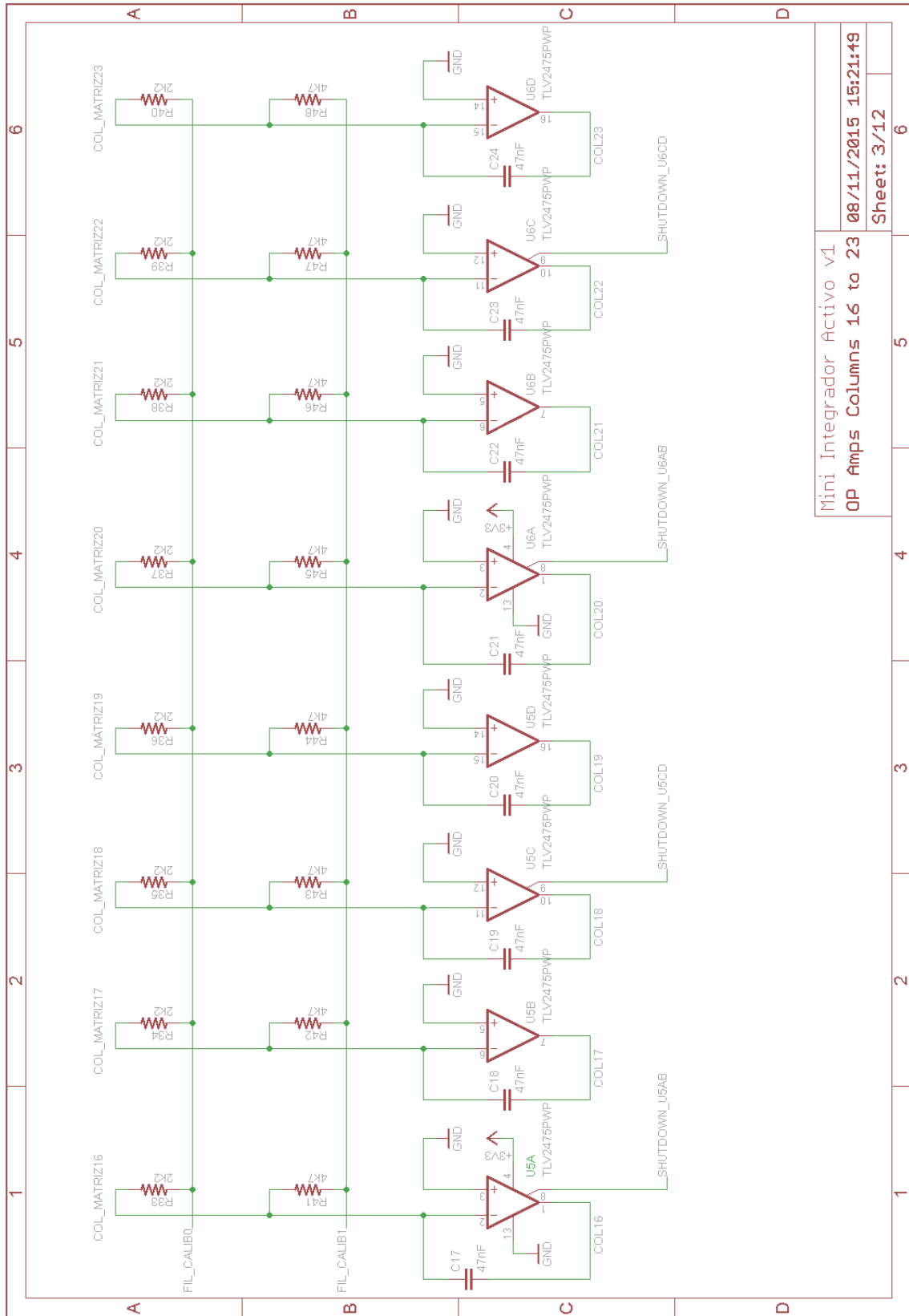
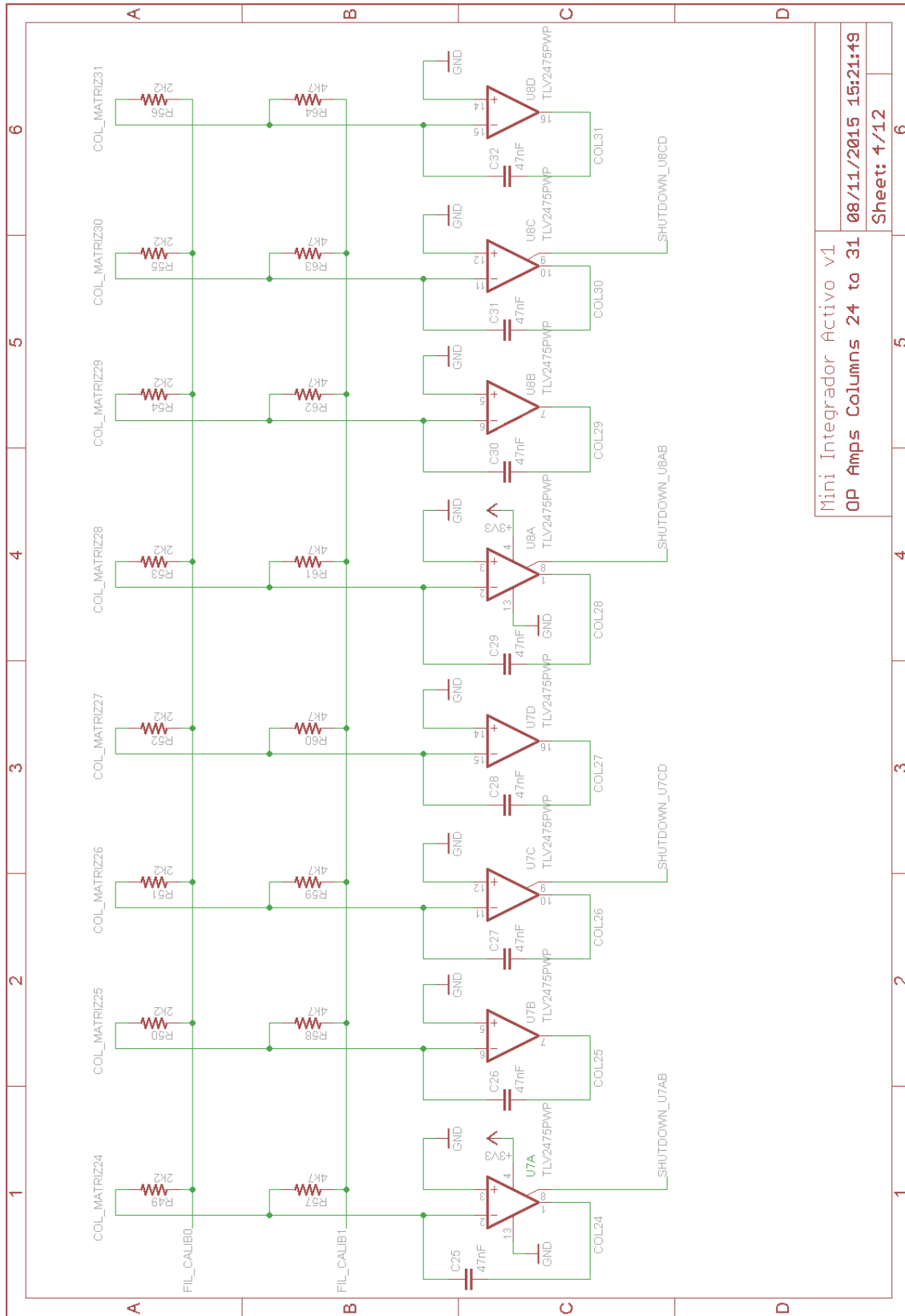


Figura B-2. Mini Integrador Activo – Amplificadores Operacionales (2/4)



Mini Integrador Activo v1		08/11/2015 15:21:49
OP Amps Columns 16 to 23		Sheet: 3/12

Figura B-3. Mini Integrador Activo – Amplificadores Operacionales (3/4)



Mini Integrador Activo v1		08/11/2015 15:21:49	
OP Amps Columns 24 to 31		Sheet: 4/12	

Figura B-4. Mini Integrador Activo – Amplificadores Operacionales (4/4)

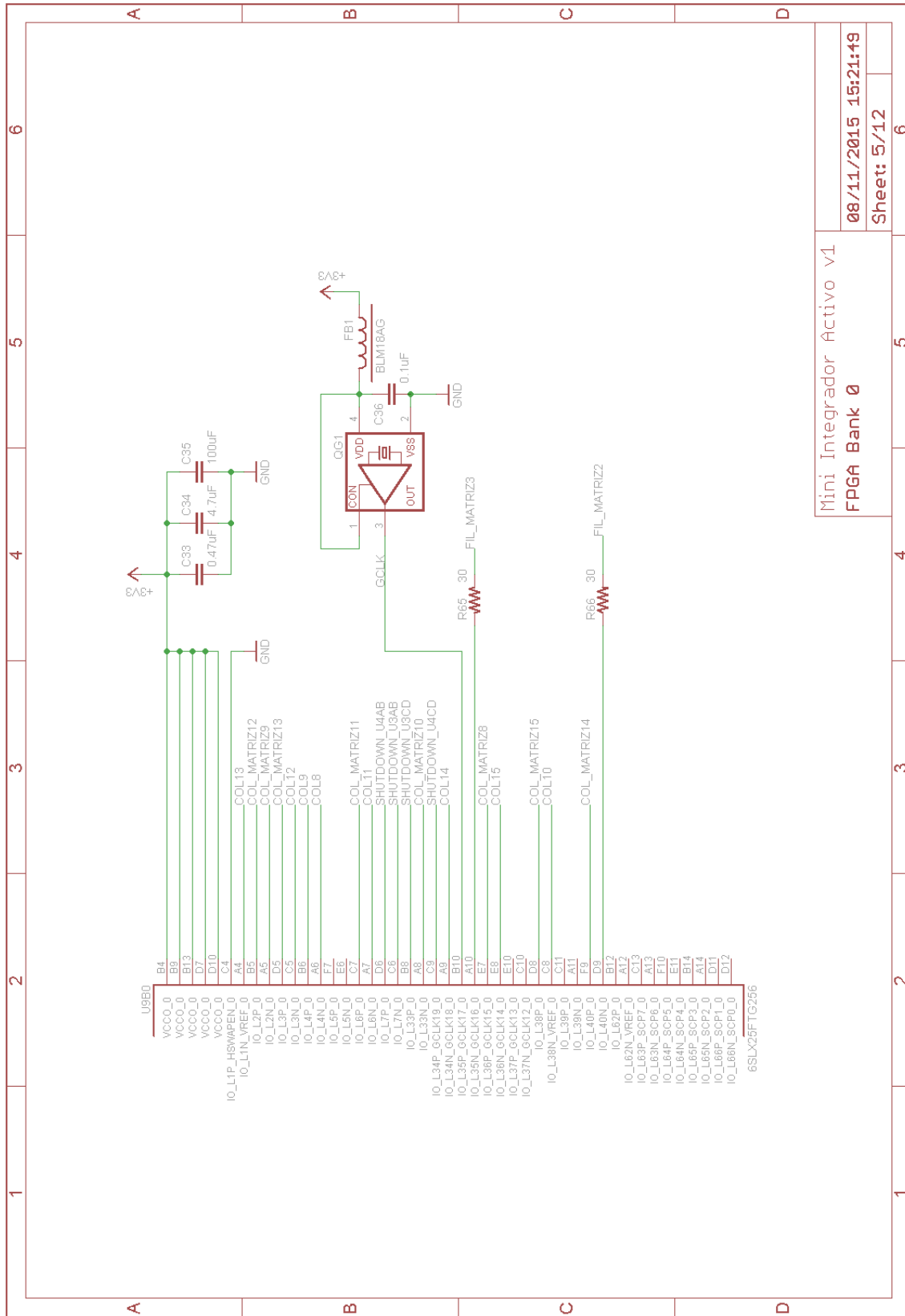


Figura B-5. Mini Integrador Activo – FPGA Sensor Banco 0 E/S

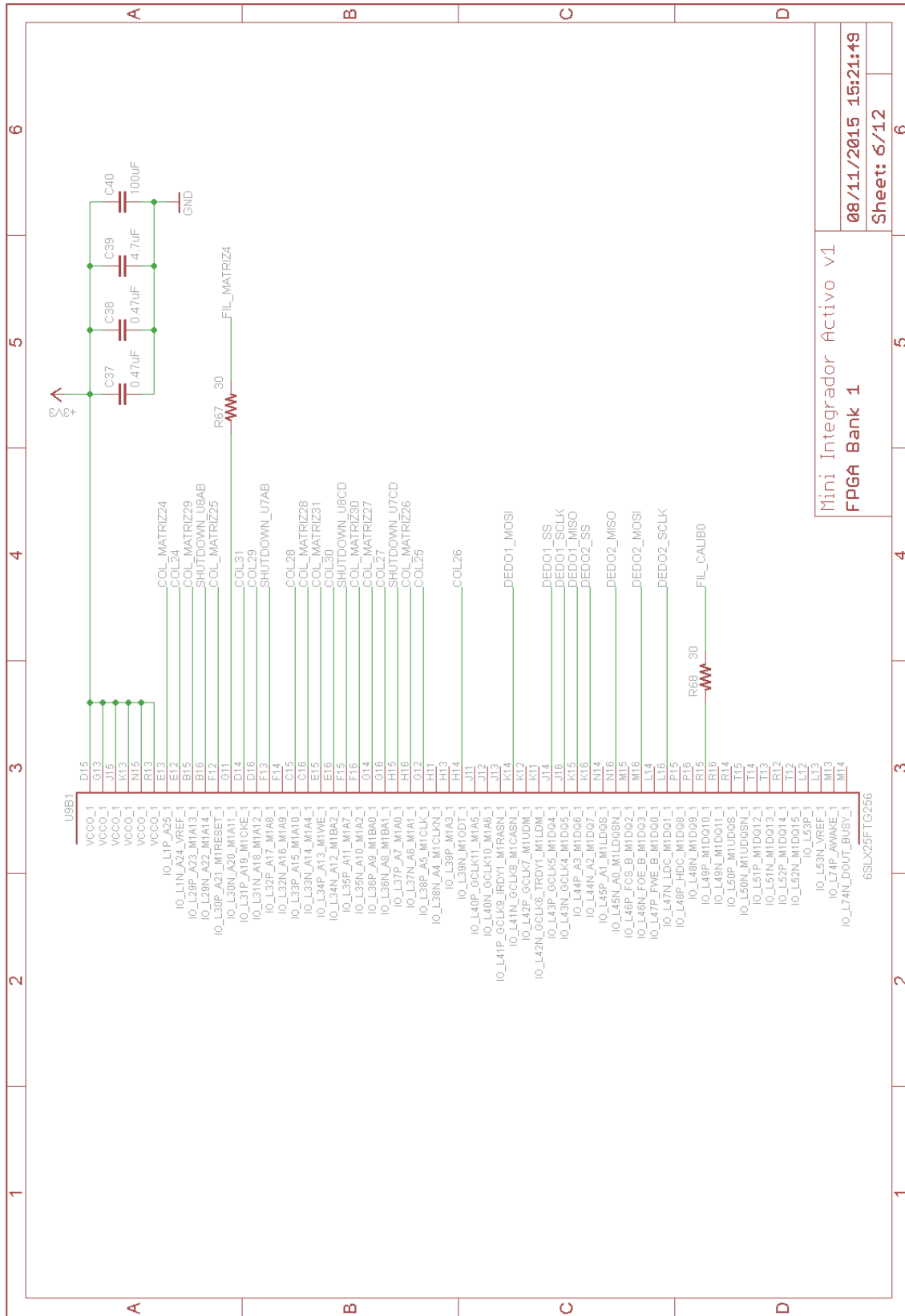
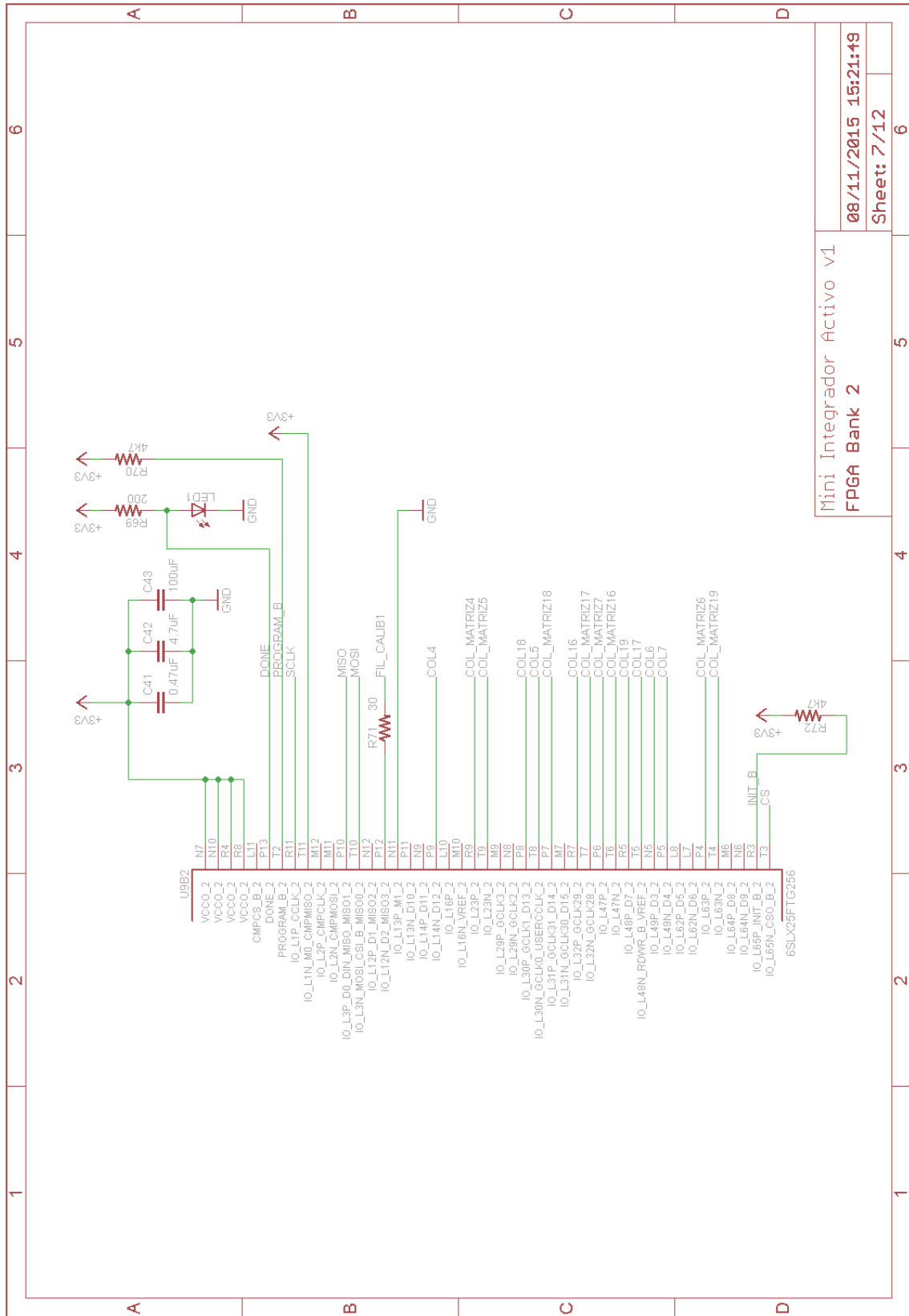


Figura B-6. Mini Integrador Activo – FPGA Sensor Banco 1 E/S



Mini Integrador Activo v1
 FPGA Bank 2
 08/11/2015 15:21:49
 Sheet: 7/12

Figura B-7. Mini Integrador Activo – FPGA Sensor Banco 2 E/S

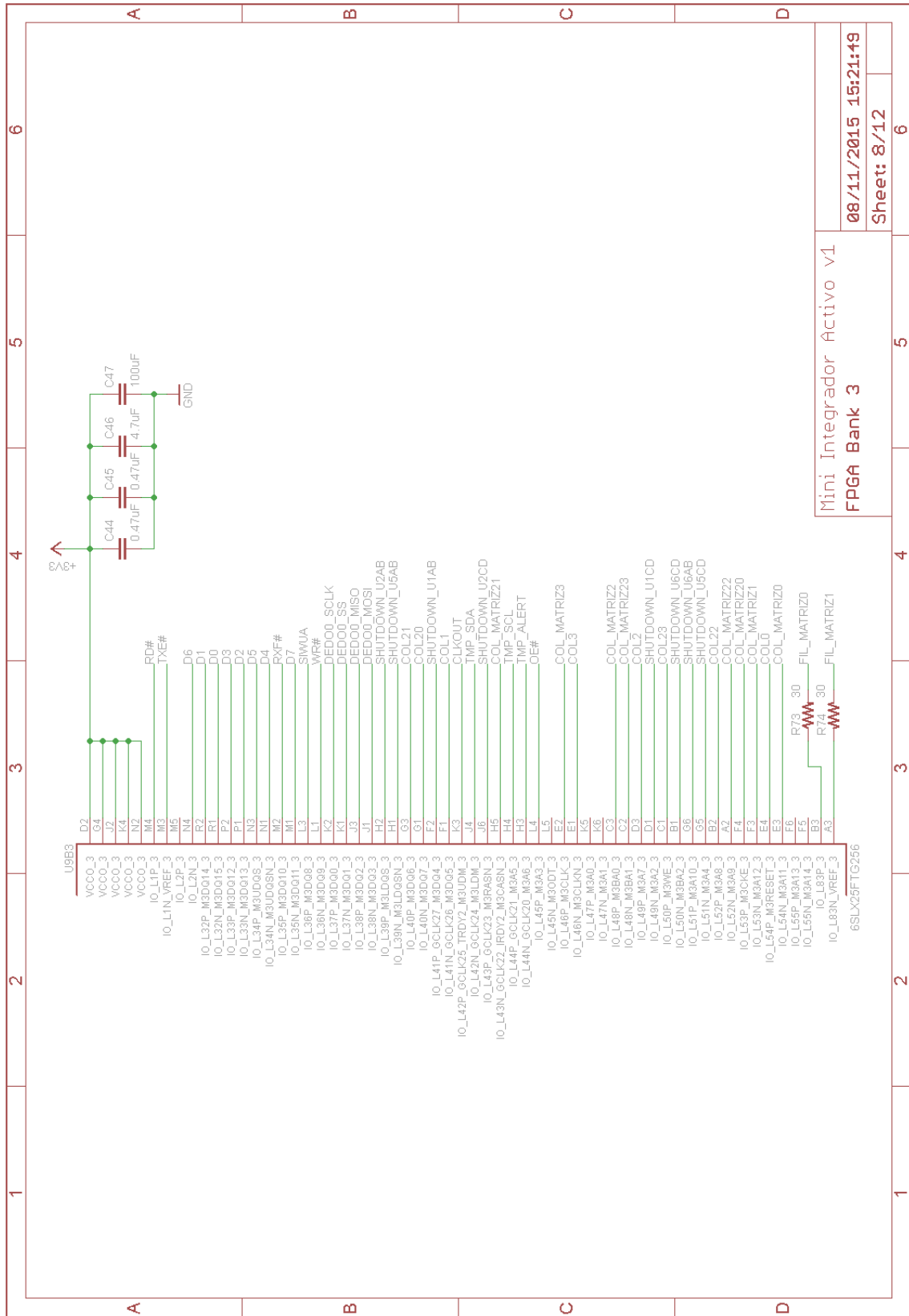


Figura B-8. Mini Integrador Activo – FPGA Sensor Banco 3 E/S

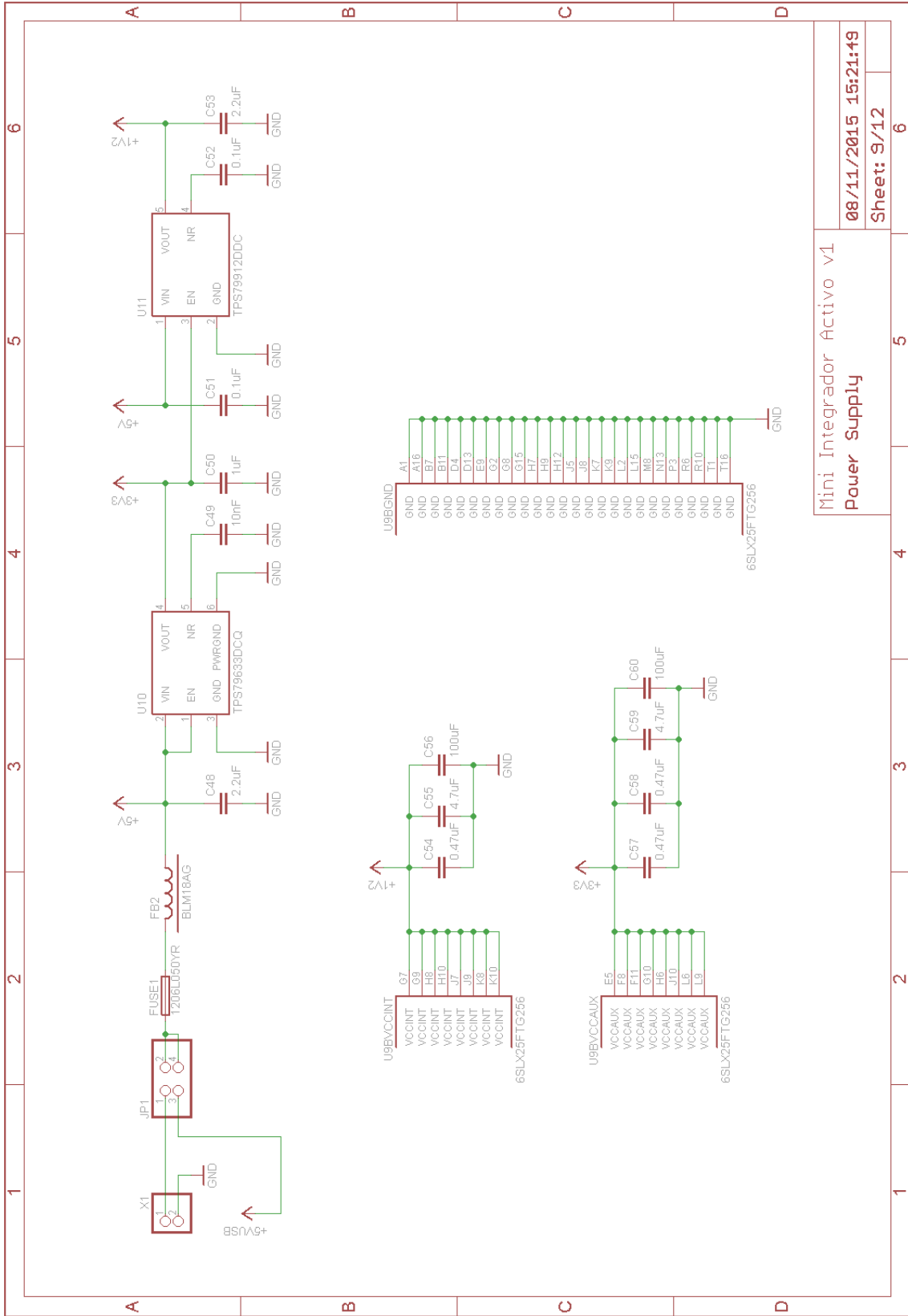
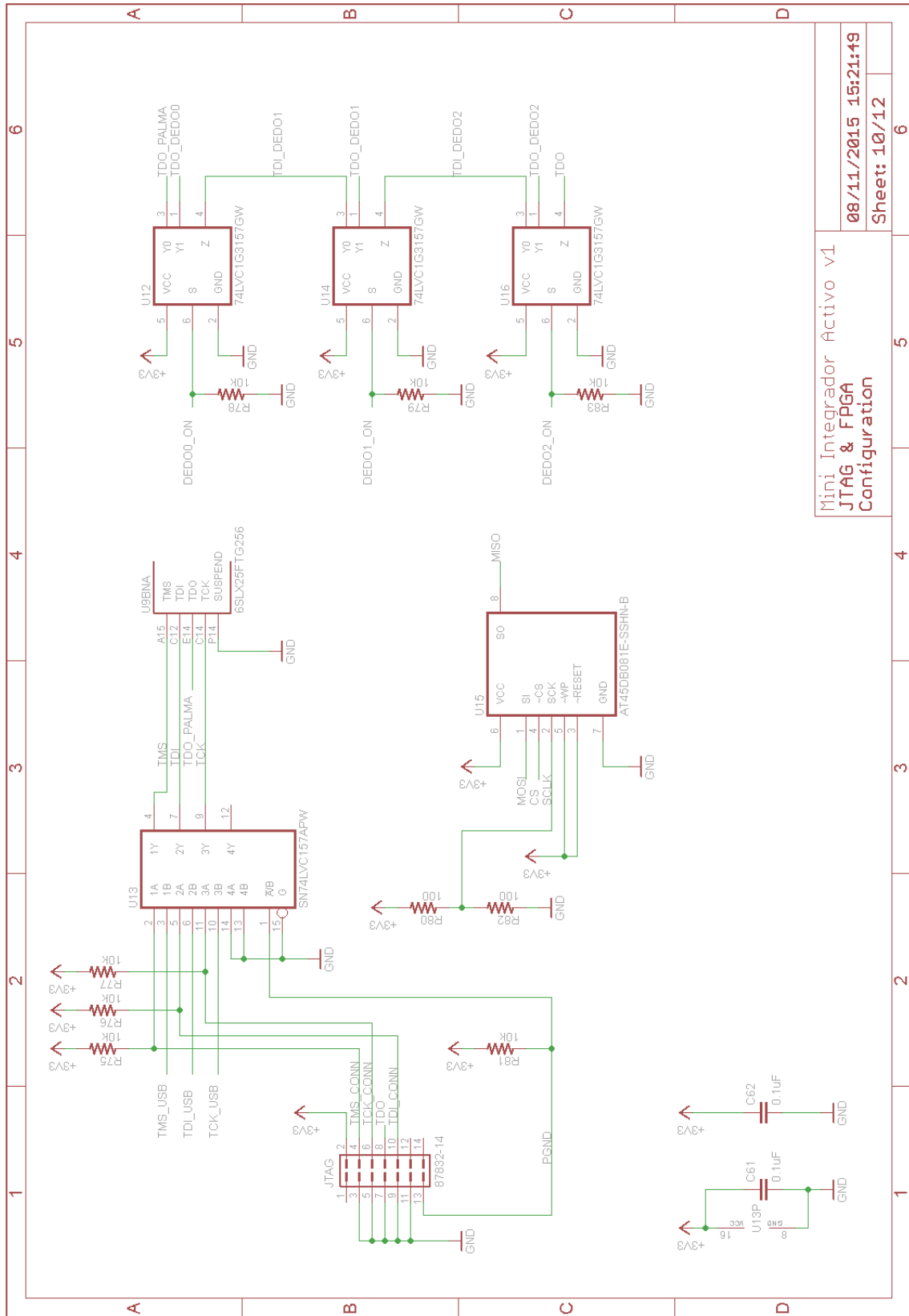
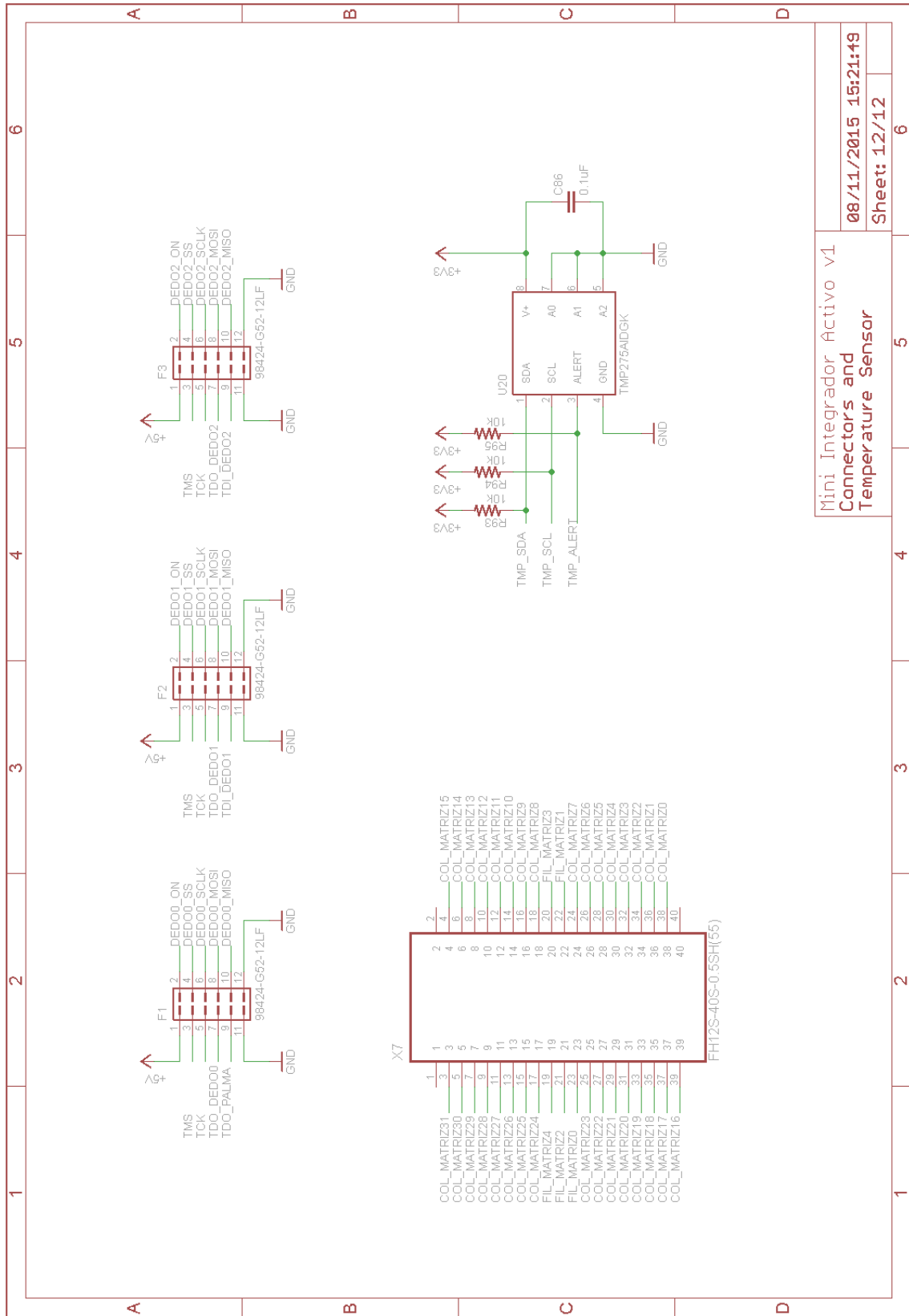


Figura B-9. Mini Integrador Activo – Circuito de Alimentación



Mini Integrador Activo v1
 JTAG & FPGA
 Configuration
 08/11/2015 15:21:49
 Sheet: 10/12

Figura B-10. Mini Integrador Activo – Cadena JTAG



Mini Integrador Activo v1
Connectors and
Temperature Sensor
08/11/2015 15:21:49
Sheet: 12/12

Figura B-12. Mini Integrador Activo – Puertos de Expansión SPI, conector matriz de electrodos y sensor de temperatura

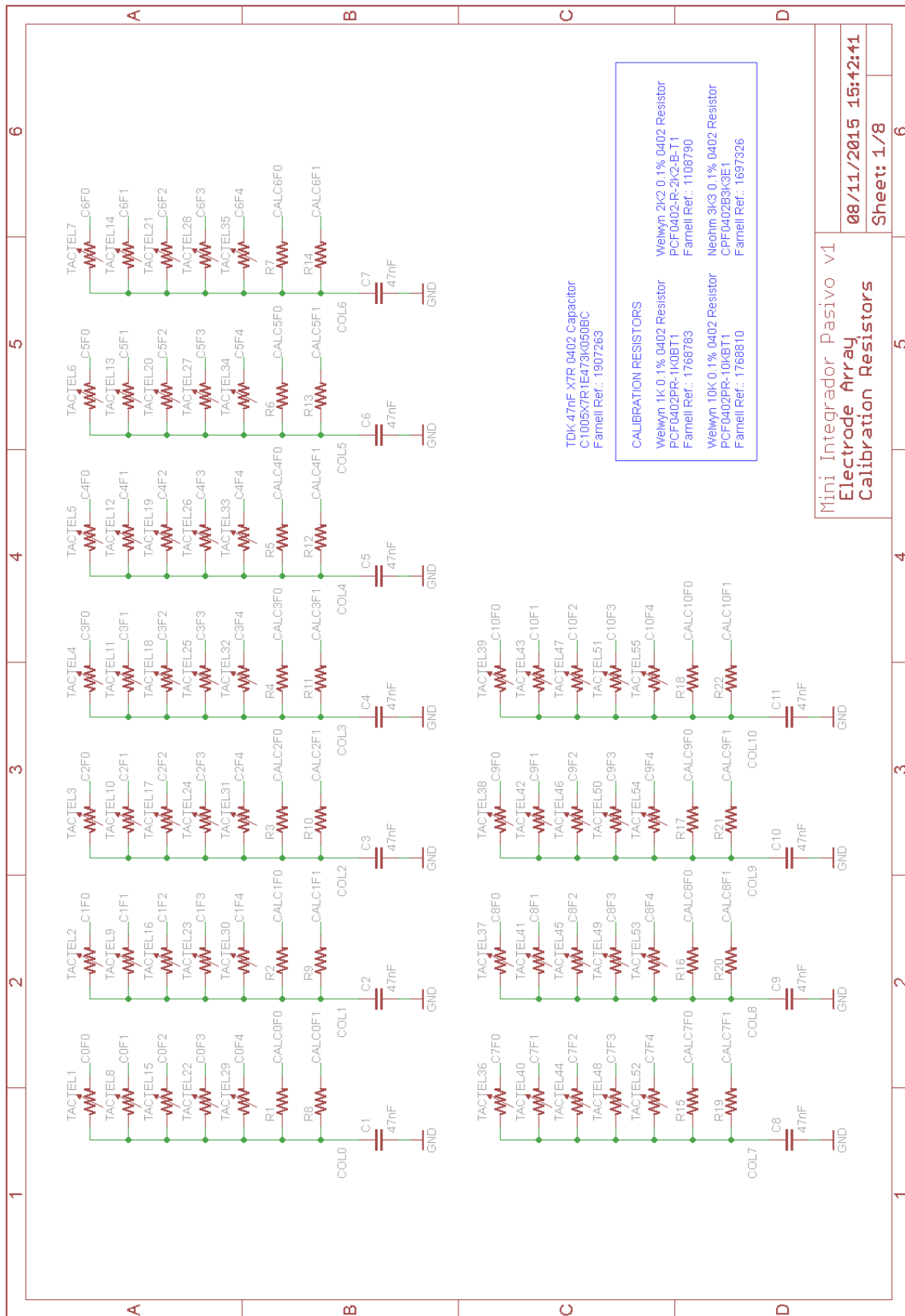


Figura B-13. Mini Integrador Pasivo – Matriz de electrodos y resistencias de calibración

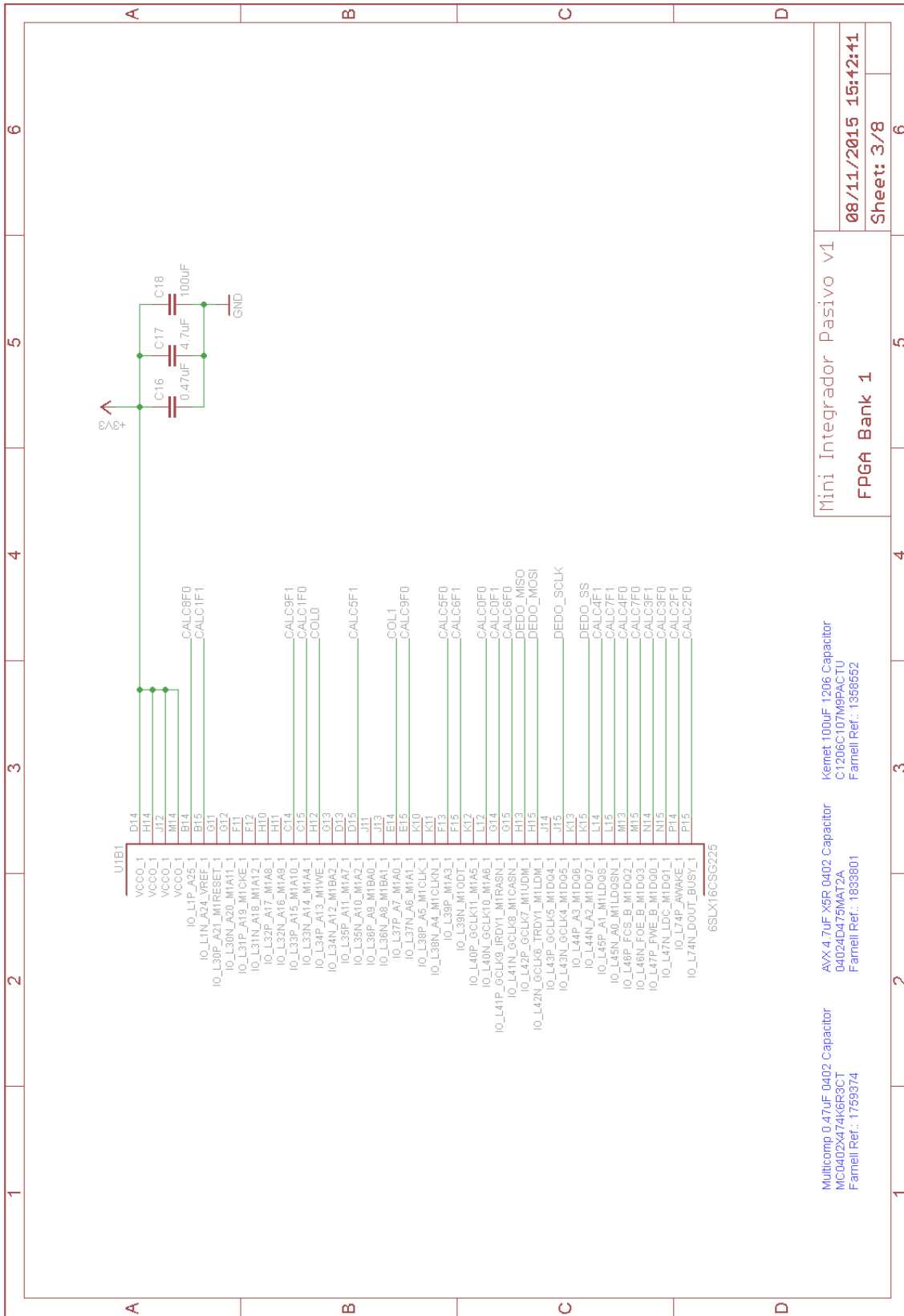
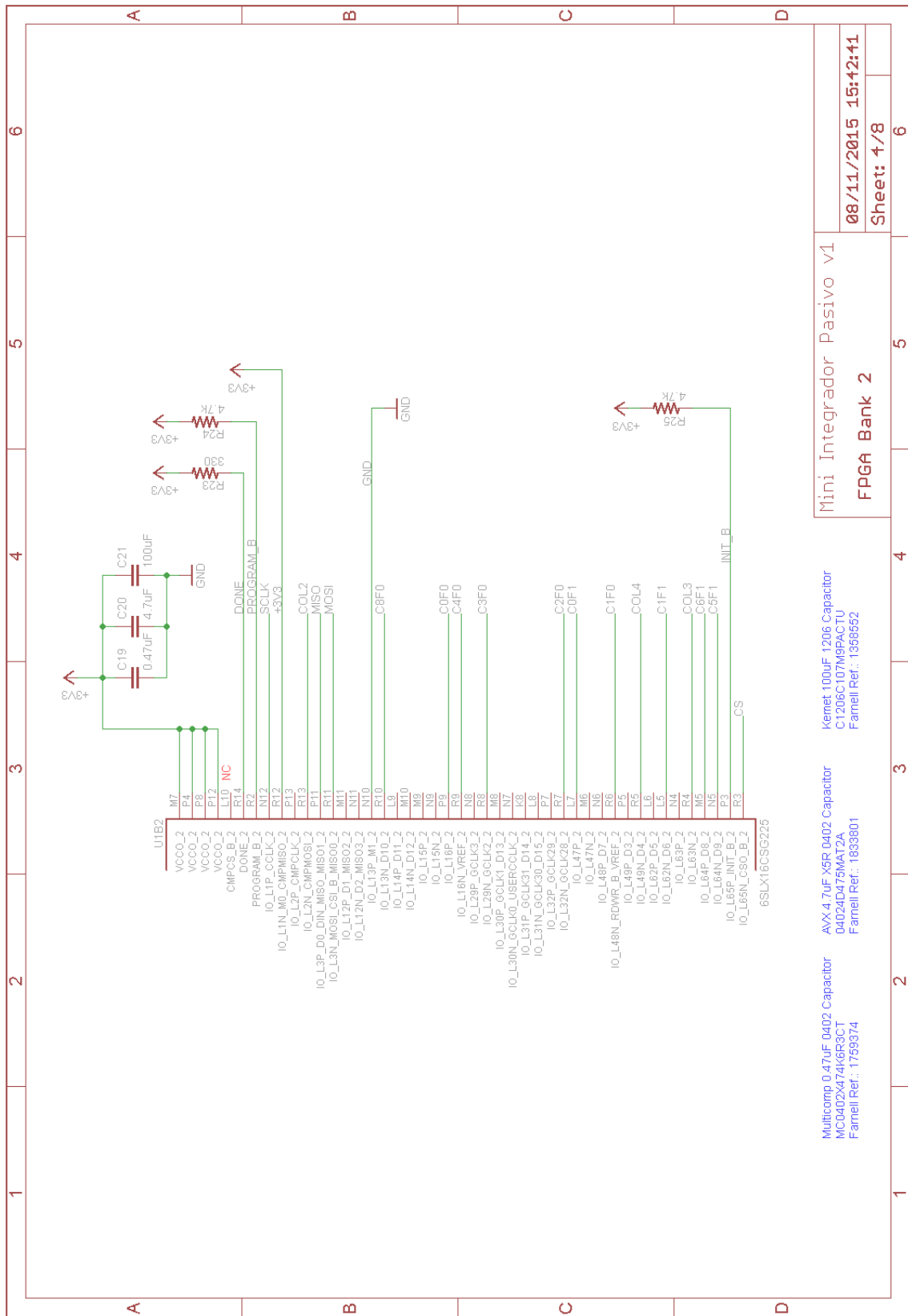


Figura B-15. Mini Integrador Pasivo – FPGA Sensor Banco 1 E/S



Mini Integrador Pasivo v1	
08/11/2015	15:42:41
FPGA Bank 2	
Sheet: 4/8	

Multicomp 0.47µF 0402 Capacitor MIC0402X474K5R3CT Farnell Ref.: 1759374	AXV 4.7µF X5R 0402 Capacitor 04024D475MAT2A Farnell Ref.: 1833801	Kemet 100µF 1206 Capacitor C1206C107M95AC7U Farnell Ref.: 1356552
---	---	---

Figura B-16. Mini Integrador Pasivo – FPGA Sensor Banco 2 E/S

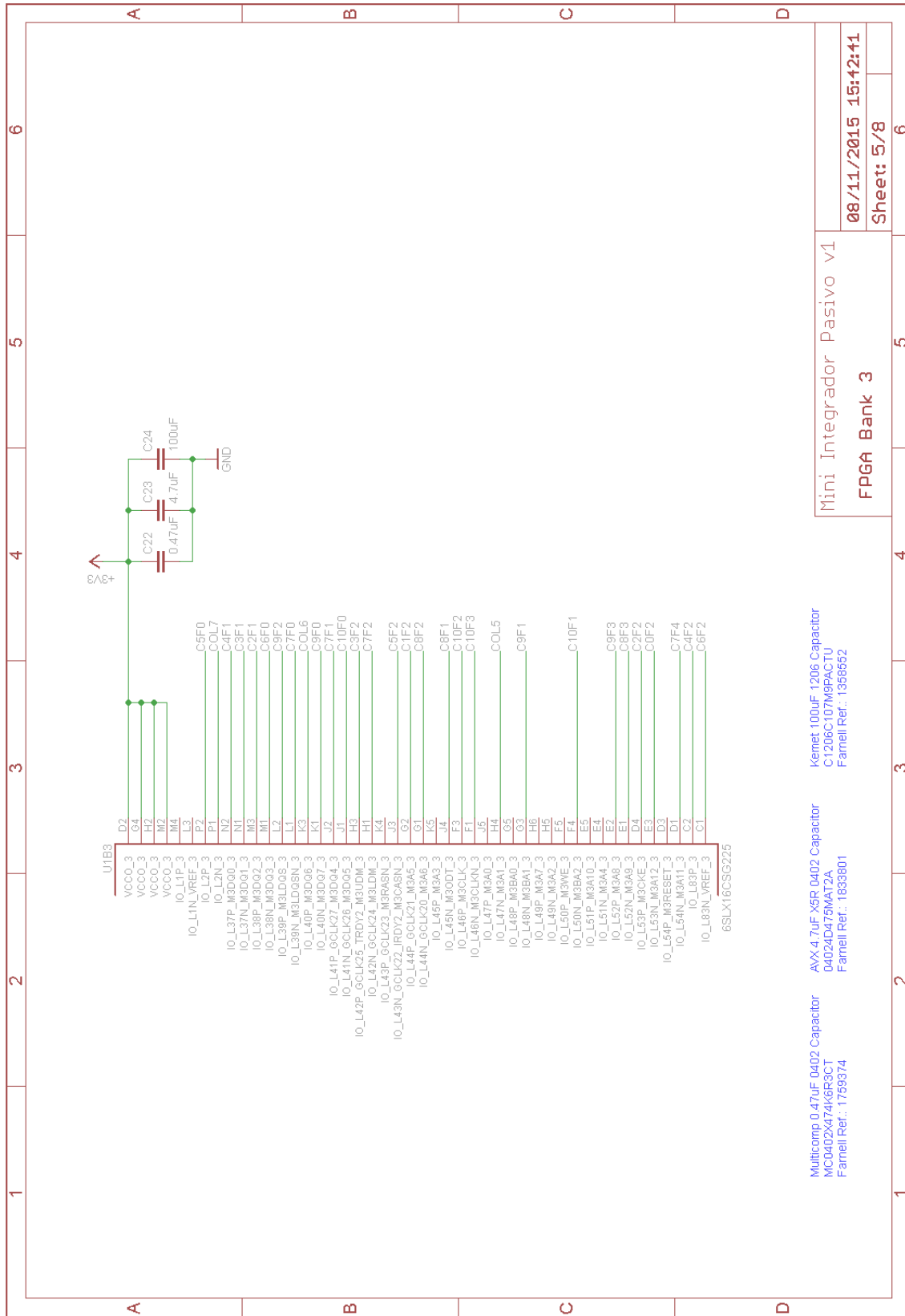


Figura B-17. Mini Integrador Pasivo – FPGA Sensor Banco 3 E/S

Mini Integrador Pasivo v1
FPGA Bank 3
08/11/2015 15:42:41
Sheet: 5/8

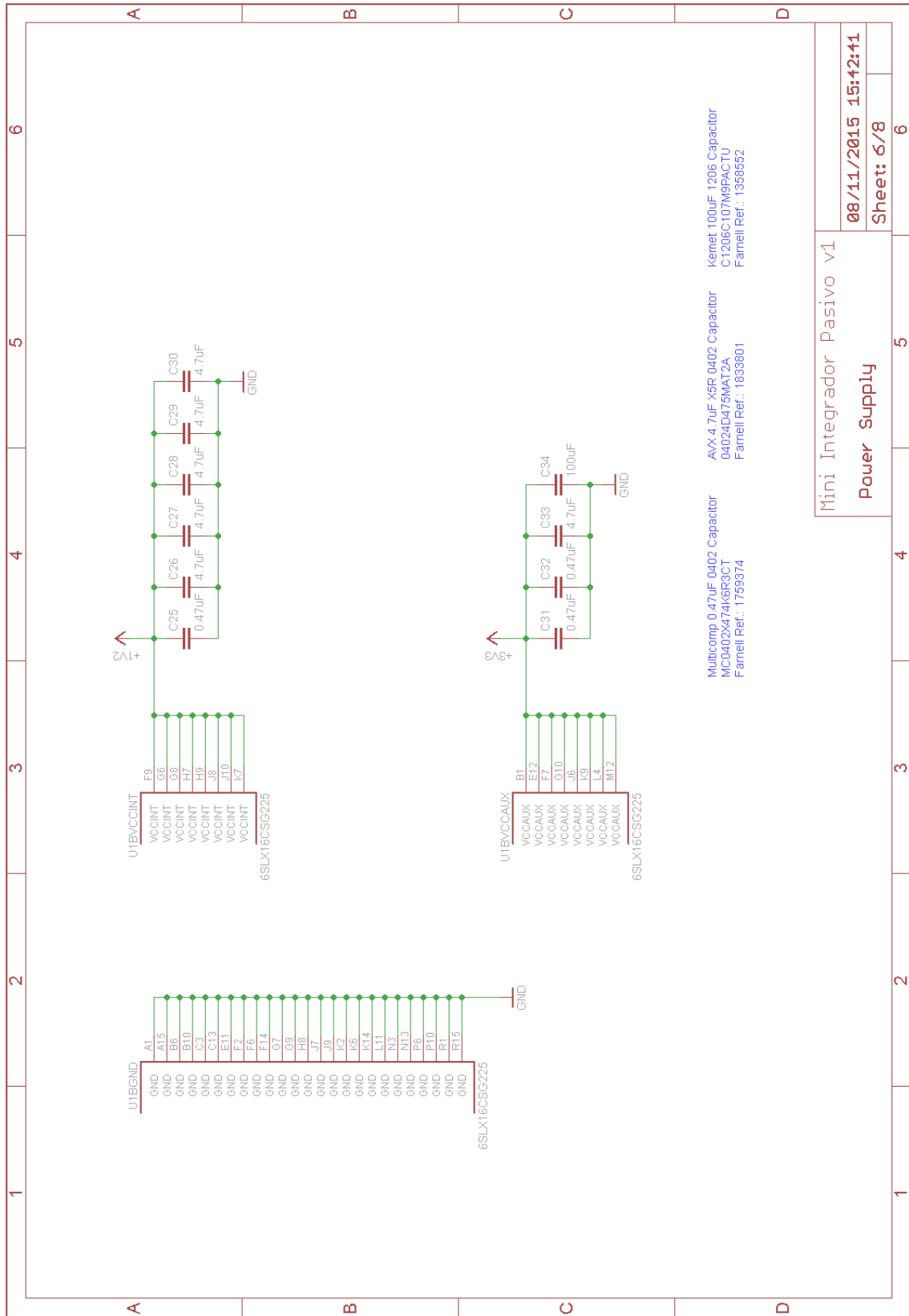
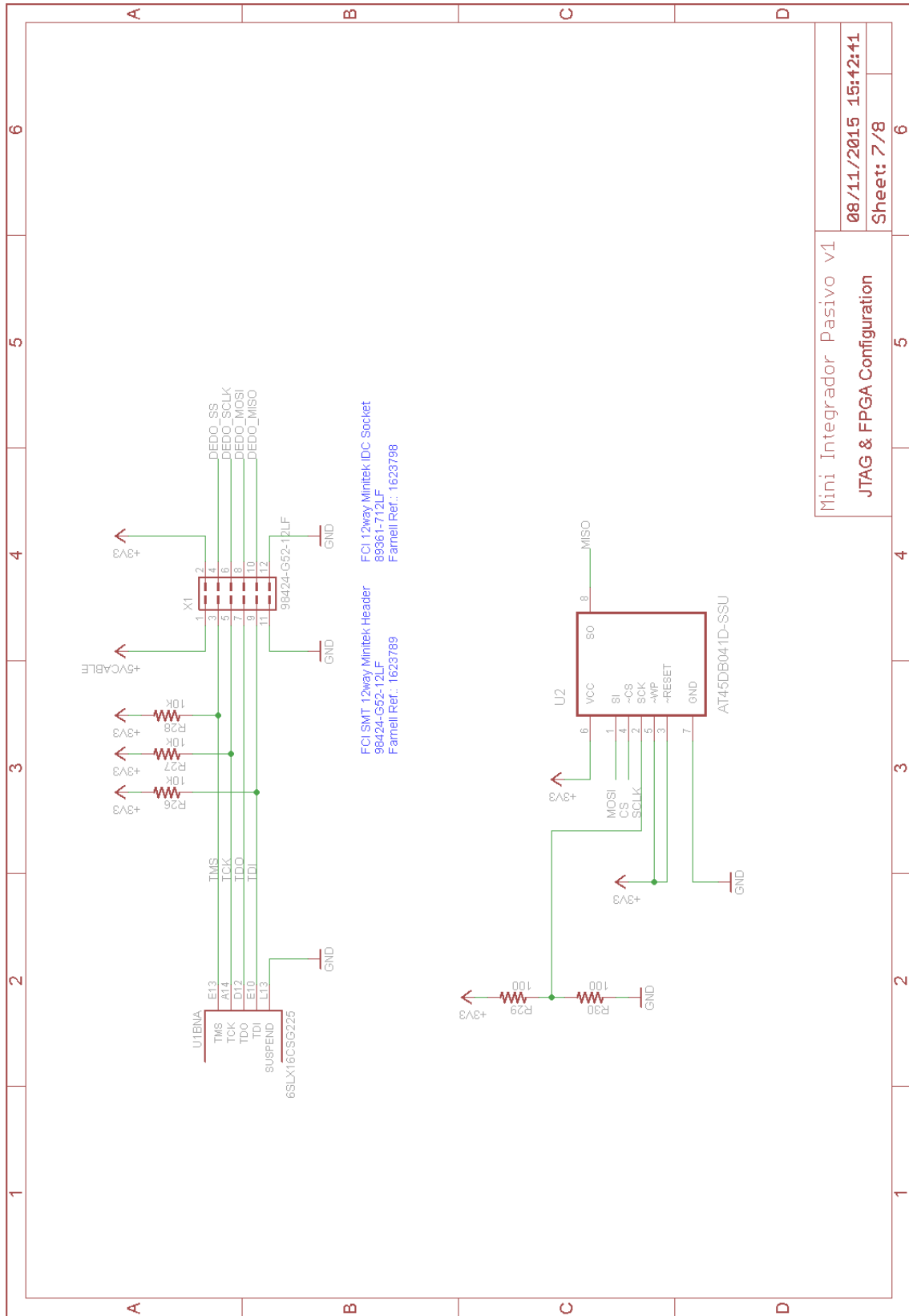
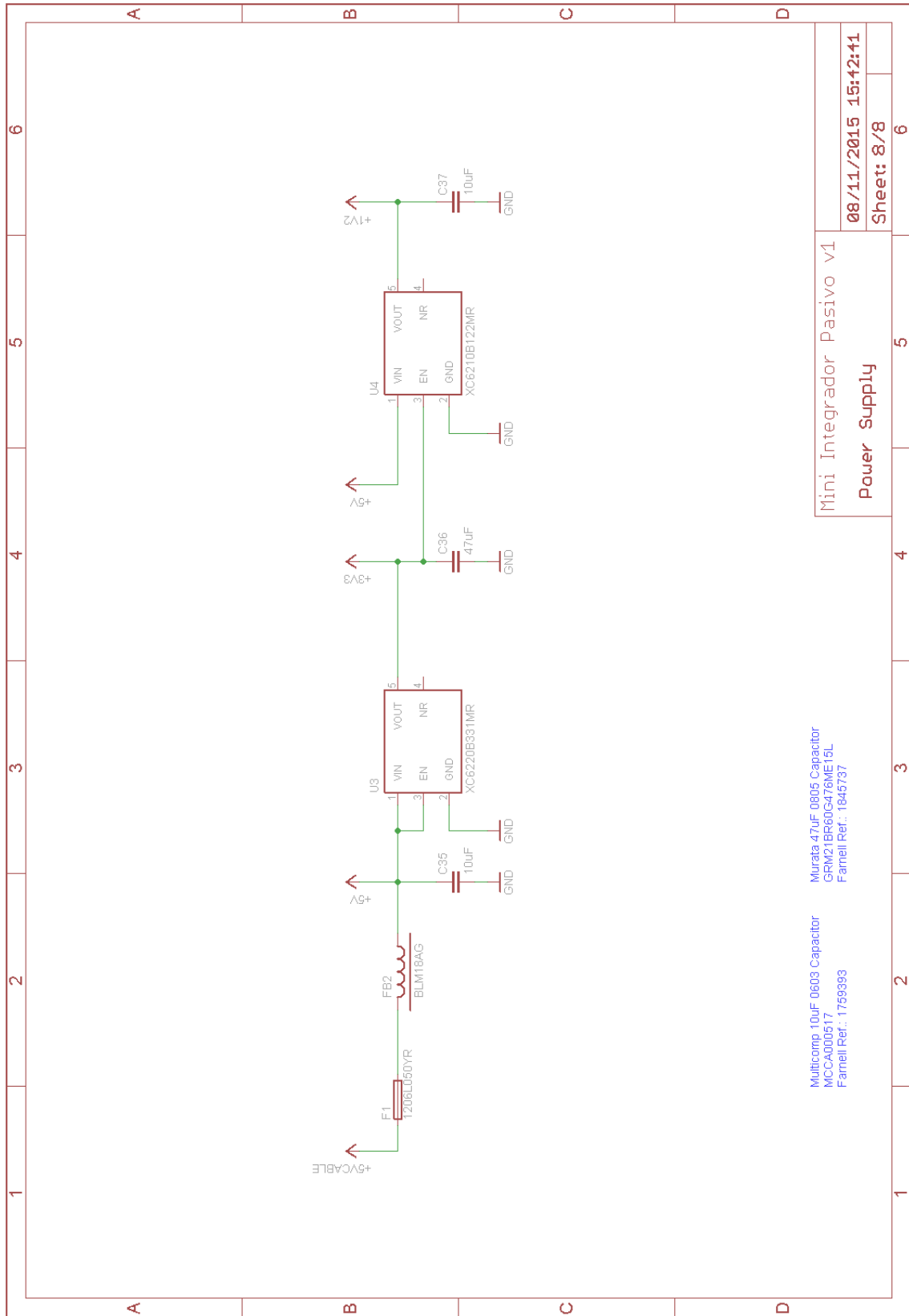


Figura B-18. Mini Integrador Pasivo – Pines de alimentación FPGA Sensor



Mini Integrador Pasivo v1	
08/11/2015	15:42:41
Sheet: 7/8	

Figura B-19. Mini Integrador Pasivo – Cadena JTAG y EEPROM



Mini Integrador Pasivo v1
Power Supply

08/11/2015 15:42:41
Sheet: 8/8

Murata 47µF 0805 Capacitor
GRM21BR60G47ME15L
Fairchild Ref.: 1845737

Multicomp 10µF 0603 Capacitor
MCCA000517
Fairchild Ref.: 1759393

Figura B-20. Mini Integrador Pasivo – Circuito de Alimentación

Anexo C. Diseño mecánico del dedo para la mano de Barrett

En este anexo se aportan figuras del diseño mecánico de la falange distal propuesta para los dedos de la mano de Barrett.

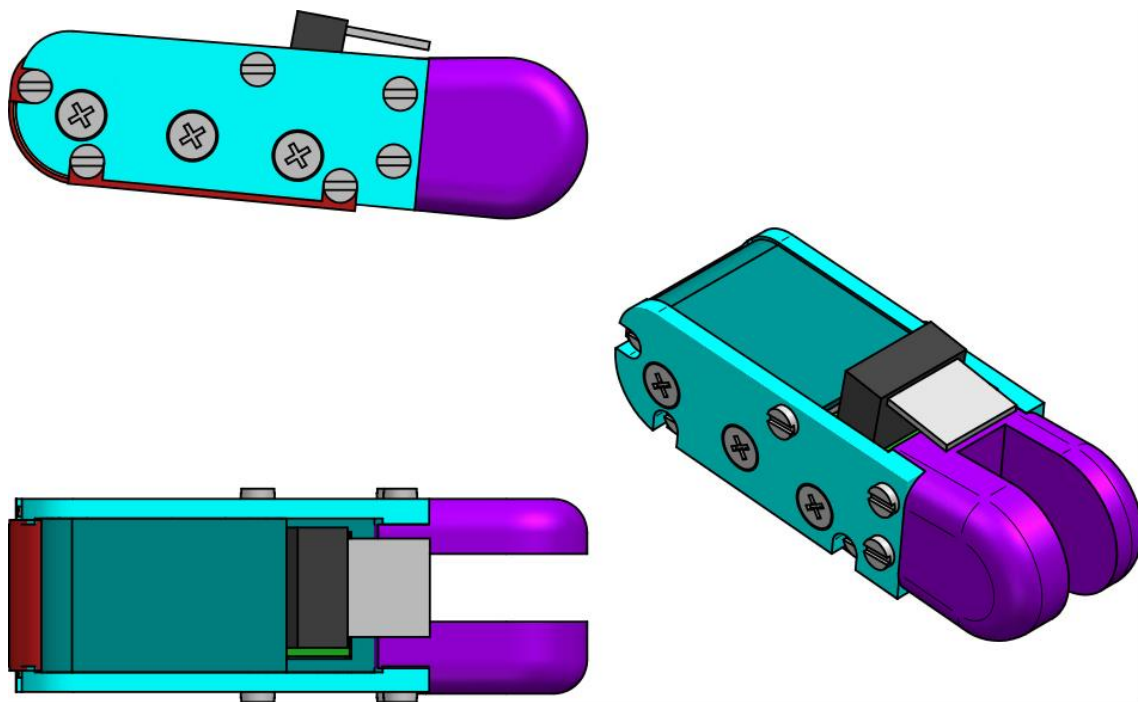


Figura C-1. Diseño mecánico completo del dedo para la mano de Barrett.

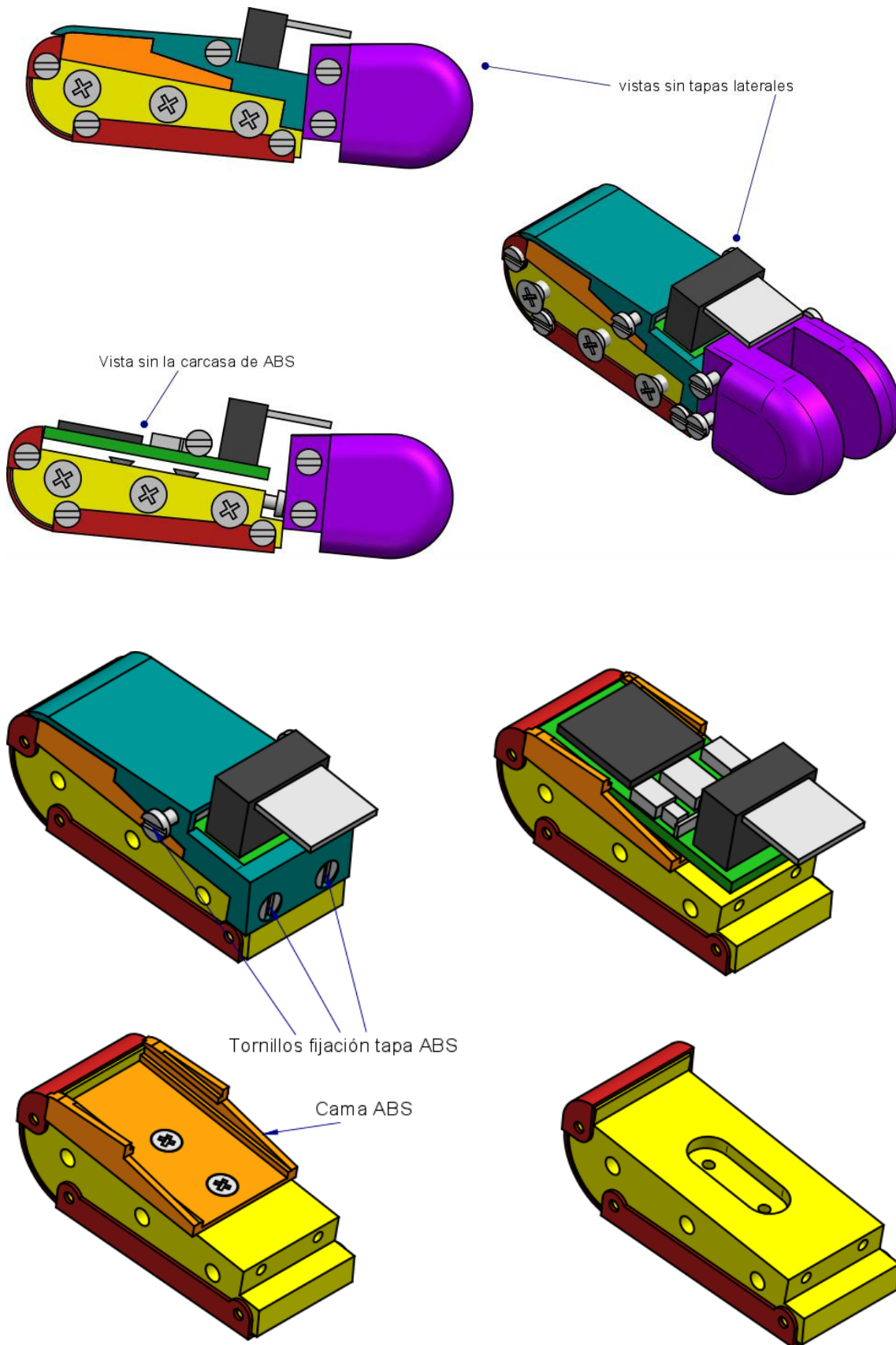


Figura C-2. Detalles del interior del dedo para la mano de Barrett.

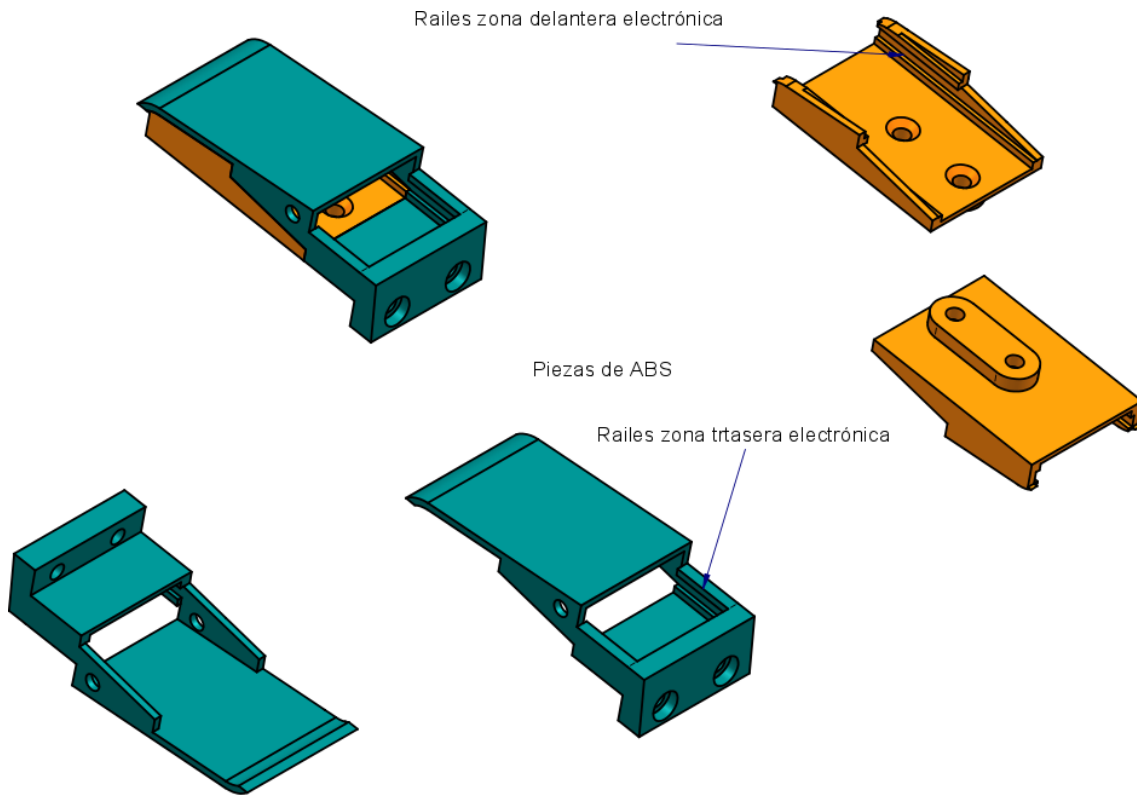


Figura C-3. Piezas internas impresas en ABS.

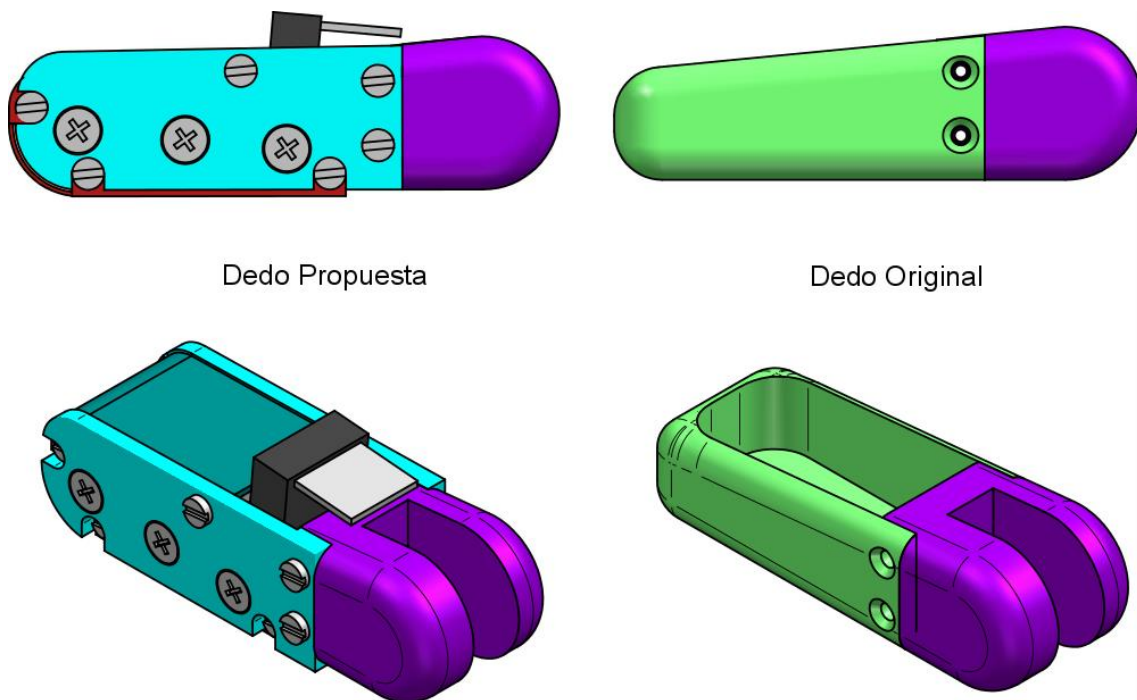


Figura C-4. Comparativa entre el dedo de la propuesta y el dedo original.

Bibliografía

Adesto Technologies 2015a. AT45DB041E 4-Mbit DataFlash (with Extra 128-Kbits), 1.65V Minimum SPI Serial Flash Memory. (En línea). [Consultado: 8 noviembre 2015]. Disponible en: <http://www.adestotech.com/wp-content/uploads/doc8783.pdf>.

Adesto Technologies 2015b. AT45DB081E 8-Mbit DataFlash (with Extra 256-Kbits), 1.7V Minimum SPI Serial Flash Memory. (En línea). [Consultado: 20 diciembre 2015]. Disponible en: http://www.adestotech.com/wp-content/uploads/DS-45DB081E_028.pdf.

Ares, L., Rodríguez-Andina, J.J. y Farina, J. 2009. FPGA-based direct resistance and capacitance measurements. *35th Annual Conference of IEEE Industrial Electronics, 2009. IECON '09*. S.l.: s.n., pp. 2837-2841.

Atmel Corporation 2015. AT93C46D 3-wire Serial EEPROM 1K (128 x 8 or 64 x 16) Datasheet. (En línea). [Consultado: 6 noviembre 2015]. Disponible en: <http://www.atmel.com/images/atmel-5193-seeeprom-at93c46d-datasheet.pdf>.

Axelson, J. 2007. *Serial Port Complete: COM Ports, USB Virtual COM Ports, and Ports for Embedded Systems*. 2nd edition. Madison, WI: Lakeview Research. ISBN 978-1-931448-06-2.

Axelson, J. 2009. *USB Complete: The Developer's Guide*. S.l.: Lakeview Research. ISBN 978-1-931448-08-6.

Barrett Technology, Inc 2015. BarrettHand. (En línea). [Consultado: 8 noviembre 2015]. Disponible en: <http://www.barrett.com/products-hand.htm>.

Bednarski, M., Szlagowski, D.D. and J., Bednarski, M. y Szlagowski, D.D. and J. 2008. Can-Bus Application System In Close Loop Control Systems For Heavy Machines. *ISARC Proceedings*, vol. 2008 Proceedings of the 25th ISARC, Vilnius, Lituania, pp. 47-51.

Bengtsson, L.E. 2012. Analysis of direct sensor-to-embedded systems interfacing: A comparison of targets' performance. *International Journal of Intelligent Mechatronics and Robotics*, vol. 2, no. 1, pp. 41-46. ISSN 2156-1664. DOI 10.4018/ijimr.2012010103. Scopus

Bierl, L., 1996. *Precise Measurements with the MSP430*. 1996. S.l.: s.n.

Bierl, L. 1999. Economic Measurement Techniques with the Comparator_A Module. (En línea). [Consultado: 14 junio 2015]. Disponible en: <http://www.ti.com/lit/an/slaa071/slaa071.pdf>.

Bouwens, A.J. 1984. *Digital Instrumentation*. New York: Mcgraw-Hill. ISBN 978-0-07-006712-7.

Cannata, G. y Maggiali, M. 2006. Processing of tactile/force measurements for a fully embedded sensor. *Multisensor Fusion and Integration for Intelligent Systems, 2006 IEEE International Conference on* (En línea). S.l.: IEEE, pp. 160–166. [Consultado: 18 enero 2015]. Disponible en: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4042034.

Cannata, G., Maggiali, M., Metta, G. y Sandini, G. 2008. An embedded artificial skin for humanoid robots. *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, 2008. MFI 2008*. S.l.: s.n., pp. 434-438.

Castellanos-Ramos, J., Navas-González, R., Macicior, H., Sikora, T., Ochoteco, E. y Vidal-Verdú, F. 2009. Tactile sensors based on conductive polymers. *Microsystem Technologies*, vol. 16, no. 5, pp. 765-776. ISSN 0946-7076, 1432-1858. DOI 10.1007/s00542-009-0958-3.

Castellanos-Ramos, J., Navas-González, R., Macicior, H., Sikora, T., Ochoteco, E. y Vidal-Verdú, F. 2010. Tactile sensors based on conductive polymers. *Microsystem Technologies*, vol. 16, no. 5, pp. 765-776. ISSN 0946-7076, 1432-1858. DOI 10.1007/s00542-009-0958-3.

Choi, B., Choi, H.R. y Kang, S. 2005. Development of tactile sensor for detecting contact force and slip. *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on* (En línea). S.l.: IEEE, pp. 2638–2643. [Consultado: 22 enero 2015]. Disponible en: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1545267.

Choi, B., Lee, S., Choi, H.R. y Kang, S. 2006. Development of Anthropomorphic Robot Hand with Tactile Sensor : SKKU Hand II. *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. S.l.: s.n., pp. 3779-3784.

Cotton, D.P.J., Chappell, P.H., Cranny, A., White, N.M. y Beeby, S.P. 2007. A Novel Thick-Film Piezoelectric Slip Sensor for a Prosthetic Hand. *IEEE Sensors Journal*, vol. 7, no. 5, pp. 752-761. ISSN 1530-437X. DOI 10.1109/JSEN.2007.894912.

Cox, D. 1997. Implementing Ohmmeter/Temperature Sensor. (En línea). [Consultado: 14 junio 2015]. Disponible en: <http://ww1.microchip.com/downloads/en/AppNotes/00512e.pdf>.

Custodio, A., Pallas-Areny, R. y Bragos, R. 2001. Error analysis and reduction for a simple sensor-microcontroller interface. *IEEE Transactions on Instrumentation and Measurement*, vol. 50, no. 6, pp. 1644-1647. ISSN 0018-9456. DOI 10.1109/19.982960.

Dahiya, R.S., Cattin, D., Adami, A., Collini, C., Barboni, L., Valle, M., Lorenzelli, L., Oboe, R., Metta, G. y Brunetti, F. 2011. Towards Tactile Sensing System on Chip for Robotic Applications. *IEEE Sensors Journal*, vol. 11, no. 12, pp. 3216-3226. ISSN 1530-437X. DOI 10.1109/JSEN.2011.2159835.

- Dahiya, R.S., Metta, G., Valle, M., Adami, A. y Lorenzelli, L. 2009. Piezoelectric oxide semiconductor field effect transistor touch sensing devices. *Applied Physics Letters*, vol. 95, no. 3, pp. 034105. ISSN 0003-6951, 1077-3118. DOI 10.1063/1.3184579.
- Dahiya, R.S., Metta, G., Valle, M. y Sandini, G. 2010. Tactile sensing—from humans to humanoids. *Robotics, IEEE Transactions on*, vol. 26, no. 1, pp. 1–20.
- Dahiya, R.S., Mittendorfer, P., Valle, M., Cheng, G. y Lumelsky, V.J. 2013. Directions Toward Effective Utilization of Tactile Skin: A Review. *IEEE Sensors Journal*, vol. 13, no. 11, pp. 4121-4138. ISSN 1530-437X, 1558-1748. DOI 10.1109/JSEN.2013.2279056.
- Dahiya, R.S. y Valle, M. 2012. *Robotic Tactile Sensing: Technologies and System*. Edición: 2013. S.l.: Springer.
- D'Alessio, T. 1999. Measurement errors in the scanning of piezoresistive sensors arrays. *Sensors and Actuators A: Physical*, vol. 72, no. 1, pp. 71–76.
- Depari, A., Falasconi, M., Flammini, A., Marioli, D., Rosa, S., Sberveglieri, G. y Taroni, A. 2007. A new low-cost electronic system to manage resistive sensors for gas detection. *IEEE Sensors Journal*, vol. 7, no. 7, pp. 1073-1077. ISSN 1530-437X. DOI 10.1109/JSEN.2007.895965. Scopus
- Díaz, J., Ros, E., Agís, R. y Bernier, J.L. 2008. Superpipelined High-performance Optical-flow Computation Architecture. *Comput. Vis. Image Underst.*, vol. 112, no. 3, pp. 262–273. ISSN 1077-3142. DOI 10.1016/j.cviu.2008.05.006.
- Díaz, J., Ros, E., Mota, S. y Carrillo, R. 2008. Local image phase, energy and orientation extraction using FPGAs. *International Journal of Electronics*, vol. 95, no. 7, pp. 743-760. ISSN 0020-7217. DOI 10.1080/00207210801941200.
- Drimus, A., Kootstra, G., Bilberg, A. y Kragic, D. 2011. Classification of rigid and deformable objects using a novel tactile sensor. *2011 15th International Conference on Advanced Robotics (ICAR)*. S.l.: s.n., pp. 427-434.
- Drimus, A., Kootstra, G., Bilberg, A. y Kragic, D. 2014. Design of a flexible tactile sensor for classification of rigid and deformable objects. *Robotics and Autonomous Systems*, vol. 62, no. 1, pp. 3-15. ISSN 0921-8890. DOI 10.1016/j.robot.2012.07.021.
- Escalda Navarro, S., Gorges, N., Worn, H., Schill, J., Asfour, T. y Dillmann, R. 2012. Haptic object recognition for multi-fingered robot hands. *2012 IEEE Haptics Symposium (HAPTICS)*. S.l.: s.n., pp. 497-502.
- Fjelstad, J. 2007. *Flexible Circuit Technology, Third Edition*. S.l.: BR Publishing, Inc.
- Flanagan, J.R., Bowman, M.C. y Johansson, R.S. 2006. Control strategies in object manipulation tasks. En: PMID: 17084619, *Current Opinion in Neurobiology*, vol. 16, no. 6, pp. 650-659. ISSN 0959-4388. DOI 10.1016/j.conb.2006.10.005.
- FOX Electronics 2008. HCMOS 5 x 3.2mm 3.3V Oscillator. (En línea). [Consultado: 8 noviembre 2015]. Disponible en: http://www.foxonline.com/pdfs/FXO_HC53.pdf.

Future Technology Devices International Ltd 2012. FT2232H Dual High Speed USB to Multipurpose UART/FIFO IC. (En línea). [Consultado: 6 noviembre 2015]. Disponible en: http://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT2232H.pdf.

Göger, D., Gorges, N. y Worn, H. 2009. Tactile sensing for an anthropomorphic robotic hand: Hardware and signal processing. *IEEE International Conference on Robotics and Automation, 2009. ICRA '09*. S.l.: s.n., pp. 895-901.

Göger, D., Weiss, K., Burghart, C. y Wörn, H. 2006. Sensitive skin for a humanoid robot. *Proceedings of the 2006 International Conference on Human-Centered Robotic Systems* (En línea). S.l.: s.n., [Consultado: 4 septiembre 2015]. Disponible en: http://www.sfb588.uni-karlsruhe.de/old/publikationen/2006/R4_Goeger_HCRS06.pdf.

Göger, D. y Worn, H. 2007. A highly versatile and robust tactile sensing system. *2007 IEEE Sensors*. S.l.: s.n., pp. 1056-1059.

Gorges, N., Navarro, S.E. y Worn, H. 2014. Analysis of tactile imprints for multi-fingered robot hands. *2014 Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM)*. S.l.: s.n., pp. 779-784.

Hall, S.H., Hall, G.W. y McCall, J.A. 2000. *High speed digital system design: a handbook of interconnect theory and design practices*. S.l.: Wiley. ISBN 978-0-471-36090-2.

Hewlett-Packard, 1997. *Understanding Frequency Counter Specifications*. 1997. S.l.: s.n.

IPC 2013. IPC-6013 Specification for Printed Wiring, Flexible and Rigid-Flex. (En línea). [Consultado: 8 noviembre 2015]. Disponible en: <http://www.ipc.org/>.

ISO 1995. Guide to the expression of uncertainty in measurement (GUM:1995). (En línea). [Consultado: 13 septiembre 2015]. Disponible en: http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=50461.

Ito, K., Saen, M. y Osada, K. 2012. Scalable robotic-hand control system based on a hierarchical multi-processor architecture adopting a large number of tactile sensors. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. S.l.: s.n., pp. 14-19.

Iwashita, A. y Shimojo, M. 2004. Development of a mixed signal LSI for tactile data processing. *Systems, Man and Cybernetics, 2004 IEEE International Conference on* (En línea). S.l.: IEEE, pp. 4408-4413. [Consultado: 22 enero 2015]. Disponible en: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1401225.

Kageyama, R., Kagami, S., Inaba, M. y Inoue, H. 1999. Development of soft and distributed tactile sensors and the application to a humanoid robot. *1999 IEEE International Conference on Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings*. S.l.: s.n., pp. 981-986 vol.2.

Kane, B.J., Cutkosky, M.R. y Kovacs, G.T.A. 2000. A traction stress sensor array for use in high-resolution robotic tactile imaging. *Microelectromechanical Systems, Journal of*, vol. 9, no. 4, pp. 425 -434. ISSN 1057-7157. DOI 10.1109/84.896763.

Kappassov, Z., Corrales, J.-A. y Perdereau, V. 2015. Tactile sensing in dexterous robot hands — Review. *Robotics and Autonomous Systems*, vol. 74, Part A, pp. 195-220. ISSN 0921-8890. DOI 10.1016/j.robot.2015.07.015.

Kawasaki, H. 2015. *Robot Hands and Multi-Fingered Haptic Interfaces: Fundamentals and Applications*. S.l.: WSPC.

Kaynov, D. 2008. Open motion control architecture for humanoid robots. (En línea), [Consultado: 7 noviembre 2015]. Disponible en: <http://e-archivo.uc3m.es/handle/10016/5573>.

Kim, D.-K., Kim, J.-H., Kim, Y.-T., Kim, M.-S., Park, Y.-K. y Kwon, Y.-H. 2013. Robot fingertip tactile sensing module with a 3D-curved shape using molding technique. *Sensors and Actuators A: Physical*, vol. 203, pp. 421-429. ISSN 0924-4247. DOI 10.1016/j.sna.2013.09.013.

Kim, M.-S., Shin, H.-J. y Park, Y.-K. 2012. Design concept of high-performance flexible tactile sensors with a robust structure. *International Journal of Precision Engineering and Manufacturing*, vol. 13, no. 11, pp. 1941-1947. ISSN 2234-7593, 2005-4602. DOI 10.1007/s12541-012-0256-3.

Larsson, P. 1999. Power supply noise in future IC's: a crystal ball reading. *Custom Integrated Circuits, 1999. Proceedings of the IEEE 1999*. S.l.: s.n., pp. 467-474.

Lawrenz, W. 2007. *CAN System Engineering: From Theory to Practical Applications*. Edición: 1st ed. 1997. Corr. 2nd printing 2007. New York: Springer. ISBN 978-0-387-94939-0.

Leineweber, M., Pelz, G., Schmidt, M., Kappert, H. y Zimmer, G. 2000. New tactile sensor chip with silicone rubber cover. *Sensors and Actuators A: Physical*, vol. 84, no. 3, pp. 236-245. ISSN 0924-4247. DOI 10.1016/S0924-4247(00)00310-1.

Lira, I.H. y Wöger, W. 1997. The evaluation of standard uncertainty in the presence of limited resolution of indicating devices. *Measurement Science and Technology*, vol. 8, no. 4, pp. 441. ISSN 0957-0233. DOI 10.1088/0957-0233/8/4/012.

Maldonado-Lopez, R., Vidal-Verdu, F., Linan, G. y Rodriguez-Vazquez, A. 2009. Integrated Circuitry to Detect Slippage Inspired by Human Skin and Artificial Retinas. *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, no. 8, pp. 1554-1565. ISSN 1549-8328. DOI 10.1109/TCSI.2008.2008290.

Merritt, B. 1999. MSP430 Based Digital Thermometer. (En línea). [Consultado: 14 junio 2015]. Disponible en: <http://www.ti.com/lit/an/slaa038/slaa038.pdf>.

Microchip Technology Inc 2012. PIC18(L)F2X/45K50 Data Sheet - 30684A. (En línea). [Consultado: 9 octubre 2015]. Disponible en: <http://ww1.microchip.com/downloads/en/DeviceDoc/30684A.pdf>.

Mittendorfer, P. y Cheng, G. 2011. Humanoid Multimodal Tactile-Sensing Modules. *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 401-410. ISSN 1552-3098. DOI 10.1109/TRO.2011.2106330.

Motorola, Inc., 2003. *SPI Block Guide* (En línea). 4 febrero 2003. S.l.: s.n. Disponible en: <http://www.ee.nmt.edu/~teare/ee308l/datasheets/S12SPIV3.pdf>.

Mouri, T., Kawasaki, H., Yoshikawa, K., Takai, J. y Ito, S. 2002. Anthropomorphic Robot Hand: Gifu Hand III. *Proceedings of ICCAS2002* (En línea). Muiu Resort. Jeonbuk. Korea: s.n., pp. 1288-1293. [Consultado: 10 enero 2016]. Disponible en: http://www.mindtrans.narod.ru/pdfs/Gifu_Hand_III.pdf.

Mukai, T., Onishi, M., Odashima, T., Hirano, S. y Luo, Z. 2008. Development of the Tactile Sensor System of a Human-Interactive Robot «RIMA». *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 505-512. ISSN 1552-3098. DOI 10.1109/TRO.2008.917006.

Mukundan, R. y Ramakrishan, K.R. 1998. *Moment Functions in Image Analysis: Theory and Applications*. Singapore; River Edge, NJ: World Scientific Pub Co Inc. ISBN 978-981-02-3524-6.

Oballe-Peinado, Ó., Vidal-Verdú, F., Sánchez-Durán, J.A., Castellanos-Ramos, J. y Hidalgo-López, J.A. 2015. Smart Capture Modules for Direct Sensor-to-FPGA Interfaces. *Sensors*, vol. 15, no. 12, pp. 31762-31780. DOI 10.3390/s151229878.

Ochoteco, E., Pomposo, J.A., Sikora, T., Vidal, F., Martinez, F., Obieta, G. y Grande, H. 2007. All-plastic distributed pressure sensors: taylor-made performance by electroactive materials design. *Microsystem Technologies*, vol. 14, no. 8, pp. 1089-1097. ISSN 0946-7076, 1432-1858. DOI 10.1007/s00542-007-0453-7.

Ohmura, Y., Kuniyoshi, Y. y Nagakubo, A. 2006. Conformable and scalable tactile sensor skin for curved surfaces. *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. S.l.: s.n., pp. 1348-1353.

Pallàs-Areny, R., Jordana, J. y Casas, Ó. 2004. Optimal two-point static calibration of measurement systems with quadratic response. *Review of Scientific Instruments*, vol. 75, no. 12, pp. 5106-5111. ISSN 0034-6748, 1089-7623. DOI 10.1063/1.1818531.

Pallàs-Areny, R. y Webster, J.G. 2001. *Sensors and Signal Conditioning*. 2ª Edición. New York: John Wiley & Sons Inc. ISBN 978-0-471-33232-9.

Peatman, J.B. 1997. *Design with PIC Microcontrollers*. 1 edition. Upper Saddle River, N.J.: Prentice Hall. ISBN 978-0-13-759259-3.

Pressure Profile Systems 2015. RoboTouch - Tactile Sensors with Digital Output for the Barrett Hand. (En línea). [Consultado: 10 enero 2016]. Disponible en: http://static1.squarespace.com/static/53836bf1e4b011aa8ac0ffb9/t/53c85b3ae4b052122aacf107/1405639482642/PPS+_RoboTouch_SpecSheet.pdf.

Raibert, M.H. y Tanner, J.E. 1982. Design and Implementation of a VLSI Tactile Sensing Computer. *The International Journal of Robotics Research*, vol. 1, no. 3, pp. 3-18. ISSN 0278-3649, 1741-3176. DOI 10.1177/027836498200100301.

Reverter, F. 2014. Direct interface circuits for sensors. En: S. Nihtianov y A. Luque (eds.), *Smart Sensors and MEMS: Intelligent Devices and Microsystems for Industrial Applications* (En línea). S.l.: Woodhead Publishing, pp. 27-62. [Consultado: 9 octubre

2015]. ISBN 978-0-85709-502-2. Disponible en: <http://www.sciencedirect.com/science/article/pii/B9780857095022500028>.

Reverter, F., Jordana, J., Gasulla, M. y Pallàs-Areny, R. 2005. Accuracy and resolution of direct resistive sensor-to-microcontroller interfaces. *Sensors and Actuators A: Physical*, vol. 121, no. 1, pp. 78-87. ISSN 0924-4247. DOI 10.1016/j.sna.2005.01.010.

Reverter, F. y Pallàs-Areny, R. 2004. Effective number of resolution bits in direct sensor-to-microcontroller interfaces. *Measurement Science and Technology*, vol. 15, no. 10, pp. 2157. ISSN 0957-0233. DOI 10.1088/0957-0233/15/10/028.

Reverter, F. y Pallàs-Areny, R. 2005. *Direct Sensor To Microcontroller Interface Circuits*. Edición: 1. S.l.: Marcombo. ISBN 978-84-267-1380-3.

Richey, R. 1997. Resistance and Capacitance Meter Using a PIC16C622. (En línea). [Consultado: 14 junio 2015]. Disponible en: <http://ww1.microchip.com/downloads/en/AppNotes/00611b.pdf>.

Rocha, L., Velho, L. y Carvalho, P.C.P. 2004. Motion reconstruction using moments analysis. *17th Brazilian Symposium on Computer Graphics and Image Processing, 2004. Proceedings*. S.l.: s.n., pp. 354-361.

Saxena, R.S., Bhan, R.K., Saini, N.K. y Muralidharan, R. 2011. Virtual Ground Technique for Crosstalk Suppression in Networked Resistive Sensors. *IEEE Sensors Journal*, vol. 11, no. 2, pp. 432-433. ISSN 1530-437X. DOI 10.1109/JSEN.2010.2060186.

SBS Implementers Forum 2000. System Management Bus (SMBus) Specification. (En línea). [Consultado: 6 noviembre 2015]. Disponible en: <http://smbus.org/specs/smbus20.pdf>.

Schill, J., Laaksonen, J., Przybylski, M., Kyrki, V., Asfour, T. y Dillmann, R. 2012. Learning continuous grasp stability for a humanoid robot hand based on tactile sensing. *2012 4th IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*. S.l.: s.n., pp. 1901-1906.

Schmid, A.J., Gorges, N., Goger, D. y Wörn, H. 2008. Opening a door with a humanoid robot using multi-sensory tactile feedback. *IEEE International Conference on Robotics and Automation, 2008. ICRA 2008*. S.l.: s.n., pp. 285-291.

Schmitz, A., Maiolino, P., Maggiali, M., Natale, L., Cannata, G. y Metta, G. 2011. Methods and Technologies for the Implementation of Large-Scale Robot Tactile Sensors. *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 389-400. ISSN 1552-3098. DOI 10.1109/TRO.2011.2132930.

Schöpfer, M., Schuermann, C., Pardowitz, M. y Ritter, H. 2010. Using a Piezo-Resistive Tactile Sensor for Detection of Incipient Slippage. *Robotics (ISR), 2010 41st International Symposium on and 2010 6th German Conference on Robotics (ROBOTIK)*. S.l.: s.n., pp. 1-7.

Schunk GmbH & Co. KG 2015. SDH Servo-electric 3-Finger Gripping Hand. (En línea). [Consultado: 10 enero 2016]. Disponible en: <http://mobile.schunk-microsite.com/en/produkte/products/sdh-servo-electric-3-finger-gripping-hand.html>.

Schürmann, C., Schöpfer, M., Haschke, R. y Ritter, H. 2012. A High-Speed Tactile Sensor for Slip Detection. En: E. Prassler, M. Zöllner, R. Bischoff, W. Burgard, R. Haschke, M. Hägele, G. Lawitzky, B. Nebel, P. Plöger y U. Reiser (eds.), *Towards Service Robots for Everyday Environments* (En línea). S.l.: Springer Berlin Heidelberg, Springer Tracts in Advanced Robotics, 76, pp. 403-415. [Consultado: 7 enero 2016]. ISBN 978-3-642-25115-3. Disponible en: http://link.springer.com/chapter/10.1007/978-3-642-25116-0_27.

Seminara, L., Capurro, M. y Valle, M. 2015. Tactile data processing method for the reconstruction of contact force distributions. *Mechatronics*, vol. 27, pp. 28-37. ISSN 0957-4158. DOI 10.1016/j.mechatronics.2015.02.001.

Sherman, D., 1993. *Measure resistance and capacitance without an A/D*. 1993. S.l.: s.n.

Shimojo, M., Namiki, A., Ishikawa, M., Makino, R. y Mabuchi, K. 2004. A tactile sensor sheet using pressure conductive rubber with electrical-wires stitched method. *IEEE Sensors Journal*, vol. 4, no. 5, pp. 589-596. ISSN 1530-437X. DOI 10.1109/JSEN.2004.833152.

Silvera-Tawil, D., Rye, D. y Velonaki, M. 2015. Artificial skin and tactile sensing for socially interactive robots: A review. *Robotics and Autonomous Systems*, vol. 63, Part 3, pp. 230-243. ISSN 0921-8890. DOI 10.1016/j.robot.2014.09.008.

Speeter, T.H. 1990. A Tactile Sensing System for Robotic Manipulation. *The International Journal of Robotics Research*, vol. 9, no. 6, pp. 25-36. ISSN 0278-3649, 1741-3176. DOI 10.1177/0278364990000900603.

Tee, B.C.-K., Chortos, A., Berndt, A., Nguyen, A.K., Tom, A., McGuire, A., Lin, Z.C., Tien, K., Bae, W.-G., Wang, H., Mei, P., Chou, H.-H., Cui, B., Deisseroth, K., Ng, T.N. y Bao, Z. 2015. A skin-inspired organic digital mechanoreceptor. En: PMID: 26472906, *Science*, vol. 350, no. 6258, pp. 313-316. ISSN 0036-8075, 1095-9203. DOI 10.1126/science.aaa9306.

Texas Instruments, Inc. 2005. MAX3243E 3-V to 5.5-V Multichannel RS-232 Line Driver/Receiver (Rev. D). (En línea). [Consultado: 6 noviembre 2015]. Disponible en: <http://www.ti.com/lit/ds/symlink/max3243e.pdf>.

Texas Instruments, Inc. 2007a. 0.5°C Digital Out Temperature Sensor (Rev. D). (En línea). [Consultado: 6 noviembre 2015]. Disponible en: <http://www.ti.com/lit/ds/symlink/tmp275.pdf>.

Texas Instruments, Inc. 2007b. 600uA/Ch 2.8MHz Rail-to-Rail I/O High-Drive Op Amps with Shutdown. (En línea). [Consultado: 12 noviembre 2015]. Disponible en: <http://www.ti.com/lit/ds/symlink/tlv2475.pdf>.

Texas Instruments, Inc. 2015a. LP5907 Ultra Low-Noise 250-mA Linear Regulator for RF and Analog Circuits (Rev. I). (En línea). [Consultado: 8 noviembre 2015]. Disponible en: <http://www.ti.com/lit/ds/symlink/lp5907.pdf>.

Texas Instruments, Inc. 2015b. TPS795 Ultralow-Noise, High-PSRR, Fast, RF, 500-mA Low-Dropout Linear Regulators Datasheet. (En línea). [Consultado: 6 noviembre 2015]. Disponible en: <http://www.ti.com/lit/ds/symlink/tps795.pdf>.

Texas Instruments, Inc. 2015c. TPS799 200-mA, Low-Quiescent Current, Ultralow Noise, High-PSRR Low-Dropout Linear Regulator Datasheet. (En línea). [Consultado: 6 noviembre 2015]. Disponible en: <http://www.ti.com/lit/ds/symlink/tps799.pdf>.

Tise, B. 1988. A compact high resolution piezoresistive digital tactile sensor. , *1988 IEEE International Conference on Robotics and Automation, 1988. Proceedings*. S.l.: s.n., pp. 760-764 vol.2.

Tiwana, M.I., Redmond, S.J. y Lovell, N.H. 2012. A review of tactile sensing technologies with applications in biomedical engineering. *Sensors and Actuators A: Physical*, vol. 179, pp. 17-31. ISSN 09244247. DOI 10.1016/j.sna.2012.02.051.

Ulmen, J. y Cutkosky, M. 2010. A robust, low-cost and low-noise artificial skin for human-friendly robots. *2010 IEEE International Conference on Robotics and Automation (ICRA)*. S.l.: s.n., pp. 4836-4841.

Underwriters Laboratories 1968. Standard 94 - Standard for Tests for Flammability of Plastic Materials for Parts in Devices and Appliances. (En línea). [Consultado: 6 noviembre 2015]. Disponible en: http://ulstandards.ul.com/standard/?id=94_6.

Vidal-Verdú, F., Barquero, M.J., Castellanos-Ramos, J., Navas-González, R., Sánchez, J.A., Serón, J. y García-Cerezo, A. 2011. A Large Area Tactile Sensor Patch Based on Commercial Force Sensors. *Sensors*, vol. 11, no. 5, pp. 5489-5507. DOI 10.3390/s110505489.

Vidal-Verdú, F., Bravo-Gálvez, D. y Oballe-Peinado, O. 2010. Tactile sensor hardware based on programmable SoC. *Electronics Letters*, vol. 46, no. 16, pp. 1117-1118. ISSN 0013-5194. DOI 10.1049/el.2010.0823.

Vidal-Verdú, F., Oballe-Peinado, Ó., Sánchez-Durán, J.A., Castellanos-Ramos, J. y Navas-González, R. 2011. Three Realizations and Comparison of Hardware for Piezoresistive Tactile. *Sensors*, vol. 11, no. 3, pp. 3249-3266. DOI 10.3390/s110303249.

Wang, J., Chan, S., Carlson, R.R., Luo, Y., Ge, G., Ries, R.S., Heath, J.R. y Tseng, H.-R. 2004. Electrochemically fabricated polyaniline nanoframework electrode junctions that function as resistive sensors. *Nano Letters*, vol. 4, no. 9, pp. 1693-1697. ISSN 1530-6984. DOI 10.1021/nl049114p. Scopus

Webjörn, Å. 2004. Simple A/D for MCUs without built-in A/D converters. (En línea). [Consultado: 14 junio 2015]. Disponible en: http://cache.nxp.com/files/microcontrollers/doc/app_note/AN477.pdf?fsrch=1&sr=2&pageNum=1.

Weiss Robotics GmbH & Co. KG 2007. DSA 9205 Tactile Transducer optimized for Handling Tasks. (En línea). [Consultado: 11 enero 2016]. Disponible en: <http://www.weiss-robotics.de/en/download.html?cid=129&fid=37&id=205>.

Weiss Robotics GmbH & Co. KG 2009. DSA 9210 Tactile sensor with curved sensing surface. (En línea). [Consultado: 11 enero 2016]. Disponible en: <http://www.weiss-robotics.de/en/download.html?cid=28&fid=37&id=204>.

Wu, J., Wang, L. y Li, J. 2015. Design and Crosstalk Error Analysis of the Circuit for the 2-D Networked Resistive Sensor Array. *IEEE Sensors Journal*, vol. 15, no. 2, pp. 1020-1026. ISSN 1530-437X. DOI 10.1109/JSEN.2014.2359967.

Wu, J., Wang, L., Li, J. y Yu, Z. 2011. A small size device using temperature sensor array. *Chinese Journal of Sensors and Actuators*, vol. 24, no. 11, pp. 1649-1652. ISSN 1004-1699. DOI 10.3969/j.issn.1004-1699.2011.11.026. Scopus

Xilinx, Inc. 2010a. Platform Flash In-System Programmable Configuration PROMs Data Sheet. (En línea). [Consultado: 6 noviembre 2015]. Disponible en: http://www.xilinx.com/support/documentation/data_sheets/ds123.pdf.

Xilinx, Inc. 2010b. Spartan-3A FPGA Family Data Sheet. (En línea). [Consultado: 6 noviembre 2015]. Disponible en: http://www.xilinx.com/support/documentation/data_sheets/ds529.pdf.

Xilinx, Inc. 2011. Xilinx DS160 Spartan-6 Family Overview. (En línea). [Consultado: 8 noviembre 2015]. Disponible en: http://www.xilinx.com/support/documentation/data_sheets/ds160.pdf.

Xilinx, Inc. 2014. Spartan-3AN FPGA Family Data Sheet. (En línea). [Consultado: 6 noviembre 2015]. Disponible en: http://www.xilinx.com/support/documentation/data_sheets/ds557.pdf.

Xilinx, Inc. 2015a. Spartan-6 FPGA Data Sheet: DC and Switching Characteristics. (En línea). [Consultado: 8 noviembre 2015]. Disponible en: http://www.xilinx.com/support/documentation/data_sheets/ds162.pdf.

Xilinx, Inc. 2015b. UltraScale Architecture and Product Overview. (En línea). [Consultado: 6 noviembre 2015]. Disponible en: http://www.xilinx.com/support/documentation/data_sheets/ds890-ultrascale-overview.pdf.

Xuefeng Zhang, Yulong Zhao y Xuelei Zhang 2012. Design and fabrication of a thin and soft tactile force sensor array based on conductive rubber. *Sensor Review*, vol. 32, no. 4, pp. 273-279. ISSN 0260-2288. DOI 10.1108/02602281211257506.

Yang, Y.-J., Cheng, M.-Y., Shih, S.-C., Huang, X.-H., Tsao, C.-M., Chang, F.-Y. y Fan, K.-C. 2010. A 32×32 temperature and tactile sensing array using PI-copper films. *International Journal of Advanced Manufacturing Technology*, vol. 46, no. 9-12, pp. 945-956. ISSN 0268-3768. DOI 10.1007/s00170-009-1940-z. Scopus

Yan, H., Ang, M.H. y Poo, A.N. 2014. A Survey on Perception Methods for Human–Robot Interaction in Social Robots. *International Journal of Social Robotics*, vol. 6, no. 1, pp. 85-119. ISSN 1875-4791, 1875-4805. DOI 10.1007/s12369-013-0199-6.

Yogeswaran, N., Dang, W., Navaraj, W.T., Shakthivel, D., Khan, S., Polat, E.O., Gupta, S., Heidari, H., Kaboli, M., Lorenzelli, L., Cheng, G. y Dahiya, R. 2015. New materials and advances in making electronic skin for interactive robots. *Advanced Robotics*, vol. 29, no. 21, pp. 1359-1373. ISSN 0169-1864. DOI 10.1080/01691864.2015.1095653.