

**DISEÑO DE UN PROTOTIPO AUTONOMO BASADO EN DSP Y UN ASIC, QUE
PERMITA MEDIR LA IMPEDANCIA DE UNA CELDA DUMMY**

**EDWAR FABIAN ANTURI OÑATE
CARLOS FERNANDIO SABALLET RINCON**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIAS FISICOMENCANICAS
ESCUELA DE INGENIERIA ELECTRICA ELECTRONICA Y
TELECOMUNICACIONES
BUCARAMANGA
2007**

**DISEÑO DE UN PROTOTIPO AUTONOMO BASADO EN DSP Y UN ASIC, QUE
PERMITA MEDIR LA IMPEDANCIA DE UNA CELDA DUMMY**

**EDWAR FABIÁN ANTURI OÑATE
CARLOS FERNANDIO SABALLET RINCON**

Trabajo presentado como requisito para optar por el titulo de
INGENIERO ELECTRÓNICO

DIRECTOR:
JAIME GUILLERMO BARRERON PEREZ
Magíster En Potencia Eléctrica (MPE)

CODIRECTOR:
JOSE ALEJANDRO AMAYA PALACIO
Ing. Electricista .MIE(c)

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIAS FISICOMENCANICAS
ESCUELA DE INGENIERIA ELECTRICA ELECTRONICA
Y TELECOMUNICACIONES
BUCARAMANGA
2007**

CONTENIDO

PÁG.

INTRODUCCION	10
1. DESCRIPCION GENERAL	12
1.1. MODELAMIENTO DE UNA CELDA DUMMY	13
1.2. AD5933. DESCRIPCIÓN DEL INTEGRADO	14
1.2.1. DIAGRAMA DE PIN	15
1.2.2. ETAPA DE GENERACION.	17
1.2.3. ETAPA DE DIGITALIZACION.	20
1.2.4. CÁLCULO DE LA IMPEDANCIA.	21
1.2.5. PROTOCOLO I2C	22
1.2.6. ESCRITURA/LECTURA AL AD5933.	24
1.3. DSP MC56F8323.	26
1.3.1. CRITERIO DE SELECCIÓN DE UN DSP.	26
1.3.2. CARACTERÍSTICAS DEL MC56F8323.	27
1.3.3. PROTOCOLO I2C EN EL DSP..	29
1.3.4. ESCRITURA/LECTURA AL DSP 56F8323. .	30
1.4. PANTALLA DE CRISTAL LÍQUIDO	32
2. SOFTWARE.	42
2.1. AD5933.	42
2.1.1. MAPA DE REGISTRO	42
2.1.2. PROGRAMACIÓN GENERAL DEL AD5933	44
2.2. DSP 56F8323.	50
2.2.1. PINES DE PROPÓSITO GENERAL (GPIO).	50
2.3. PANTALLA HYUNDAI.	54
3. PRUEBAS	58
3.1. PANTALLA HYUNDAI	58
3.2. AD5933	60
3.2.1. ANÁLISIS DE RESULTADOS.	65
3.2.1.1. ANÁLISIS DE DATOS PARA CELDA DE MEDICIÓN..	69
CONCLUSIONES	72
RECOMENDACIONES	73
BIBLIOGRAFÍA	74
ANEXOS	75

LISTA DE FIGURAS

	PÁG
FIGURA 1	10
FIGURA 2.	11
FIGURA 3.	13
FIGURA 4.	14
FIGURA 5.	17
FIGURA 6.	18
FIGURA 7	22
FIGURA 8	27
FIGURA 9	27
FIGURA 10	28
FIGURA 11	31
FIGURA 12	31
FIGURA 13	32
FIGURA 14	33
FIGURA15.	33
FIGURA 16	34
FIGURA 17	35
FIGURA 18	35
FIGURA 19	44
FIGURA 20	47
FIGURA 21	50
FIGURA 22	51
FIGURA 23	51
FIGURA 24	52
FIGURA 25	54
FIGURA 26	55
FIGURA 27	56
FIGURA 28	57
FIGURA 29	58
FIGURA 30	58
FIGURA 31	60
FIGURA 32	60
FIGURA 33	61
FIGURA 34	61

FIGURA 35	REGIÓN PARA EL CÁLCULO DE FG	66
FIGURA 36	RESPUESTA DE FASE DEL SISTEMA ANTE LA MEDICIÓN DE UNA IMPEDANCIA	68
FIGURA 37	RESPUESTA SISTEMA MAGNITUD VS. FRECUENCIA	69

LISTA DE TABLAS

	PÁG.	
TABLA 1	AMPLITUD DE TENSIÓN DE SALIDA	13
TABLA 2	ASIGNACIÓN DE PINES	14
TABLA 3	POSICIONES DE MEMORIA DE LA PARTE REAL E IMAGINARIO DE DFT	19
TABLA 4	COMANDOS PARA ESCRITURA Y LECTURA DE DATOS	23
TABLA 5	CARACTERÍSTICAS DE LA LCD HG25504NG-01 DE HYUNDAI	30
TABLA 6	DESCRIPCIÓN DE LOS PINES DE LA LCD	36
TABLA 7	CONFIGURACIÓN DE LOS PINES GPIO PARA EL MANEJO DE LA LCD	37
TABLA 8	DESCRIPCIÓN DE COMANDOS DEL CONTROLADOR GRÁFICO	38
TABLA 9	MAPA DE REGISTRO AD5933/34	41
TABLA 10	COMANDOS DE CONTROL AD5933	42
TABLA 10	RANGOS DE SALIDA Y PGA	46
TABLA 12	PINES GPIO EN EL DSP 56F8323	49
TABLA 13	CONFIGURACIÓN DE LOS PINES GPIO EN EL PROTOTIPO	52
TABLA 14	VALORES DE CALIBRACIÓN	59
TABLA 15	RESULTADOS DE MEDICIONES	62
TABLA 16	FACTOR DE GANANCIA Y RESPUESTA DE LA FASE	64
TABLA 17	CALIBRACIÓN CON UNA RESISTENCIA DE 1K Ω	65
TABLA 18	RESPUESTA DE FASE DEL SISTEMA ANTE LA MEDICIÓN	67
TABLA 19	CAPACITORES SELECCIONADOS PARA PRUEBAS	69
TABLA 20	MAGNITUD DE LOS CAPACITORES	69
TABLA 21	VALOR DE LA RESISTENCIAS EN PARALELO	78

*A Dios que me permitió llegar a este mundo
A mi madre lo mas maravilloso que tengo
A mis hermanos mi fortaleza y apoyo
A Agustín que es como un padre para mí
A Carmen Gracias por estar ahí
Profesor Barrero y Amaya Gracias por su apoyo incondicional*

Carlos Fernando Saballet

*A Dios por darme la vida, por permitirme crecer día a día
A la virgen del Carmen mi eterna protectora
A mi mamá la persona más increíble y hermosa que conozco
A mi papa que siempre me apoyo para convertirme en la persona que soy
A Beto que siempre fue mi ejemplo a seguir y la persona mas correcta que conozco
A Juank, que con su alegría y sencillez me abrió la mente y me enseñó muchas cosas
A Diego, Andrés y Juan Manuel, los amigos son la familia que siempre se quiso tener
Al profesor Jaime Barrero que con simples frases nos daba lecciones de vida
Al profesor José Amaya por su colaboración y su paciencia*

Edwar Fabián Anturi Oñate

RESUMEN

TÍTULO: DISEÑO DE UN PROTOTIPO AUTONOMO BASADO EN DSP Y UN ASIC, QUE PERMITA MEDIR LA IMPEDANCIA DE UNA CELDA DUMMY*

AUTORES: ANTURI OÑATE, EDWAR FABIAN, y, SABALLET RINCON, CARLOS FERNANDO **

PALABRAS CLAVES: AD5933/34, Celda Dummy, Celda electroquímica, DSP, Espectroscopia, Mc56f8323, Metrowerks Codewarrior, Normas ASTM, pantalla de cristal líquido Hyundai.

DESCRIPCIÓN:

En este documento se explica cómo se implementaron los algoritmos de programación para calcular los parámetros de una celda *Dummy*. El desarrollo del código fuente del proyecto se realizó bajo la plataforma *Metrowerks Codewarrior* versión 8.0, a través de una tarjeta de desarrollo del DSP Mc56f8323 de Motorola y consta de tres etapas: comunicación entre el DSP y el ASIC, programación de los parámetros del AD5933 (barrido de frecuencia, Tensión de salida, PGA), procesamiento y despliegue gráfico en la pantalla LCD.

Inicialmente se describe el integrado de propósito específico (AD5933) utilizado en el prototipo, explicando el diagrama de conexiones y los registros que se deben programar para que estas funciones correctamente, también es descrito el DSP y la pantalla gráfica LCD HG25504 utilizados.

En el capítulo 2, se describe el software usado para la puesta en marcha del dispositivo, para esto se describen las funciones y rutinas utilizadas para la programación del Dsp, la pantalla de cristal líquido LCDHG25504 y el integrado de propósito específico AD5933. Finalmente, en el capítulo 3, se describen las pruebas realizadas al prototipo medidor de impedancias para celdas dummy y la forma de calibración del sistema. Se presentan los resultados obtenidos gráficamente, como la respuesta presentada en la pantalla de cristal líquido.

* Trabajo de Grado.

** Facultad de Ingeniería Físico-mecánicas. Ingeniería Electrónica.
Jaime Guillermo Barrero Pérez.

SUMMARY

TITLE: DESIGN OF AN AUTONOMOUS PROTOTYPE BASE DON A DSP AN ASIC THAT LET MEASURE THE IMPEDANCE OF A DUMMY CELL*.

AUTHORS: ANTURI OÑATE, EDWAR FABIAN, and, SABALLET RINCON, CARLOS FERNANDO **

KEY WORDS: AD 5933/34, dummy cell, electrochemical cell DSP, spectroscopy, Mc56f8323, Metrowerks codewarrior, ASTM rules, crystal liquid screen Hyundai.

DESCRIPTION:

In this document it is explained how the programming algorithms were implemented to calculate the parameters of a Dummy cell. The development of the source code of the project was carried out under the platform Metrowerks Codewarrior version 8,0, through a development card of the Dsp Mc56f8323 Motorola and it consists of three stages: communication between the DSP and the ASIC, programming of the parameters of the AD5933 (swept of frequency, Tension of exit, PGA), processing and unfolds graph in LCD screen.

Initially is described the Application Specific Integrated Circuit (AD5933) used for the implementation of the prototype, making the explanation of the diagram and the registries that are due to program so that this functions correctly, also are described to the DSP and used graphical screen LCD HG25504.

Chapter 2, in this is described the software used for the beginning of the device, for that, it is describe the functions and routines used for the for the Dsp programming, the crystal screen LCD HG25504 and Application Specific Integrated Circuit AD5933. Finally, in the chapter 3, there is describing the tests made to the measuring prototype of impedances for dummy cells and the calibration form of the system. The results have been obtained graphically, as of the liquid crystal display.

* Work of Grade.

** Faculty of Engineering Physical-mechanics. Engineering Electronics
Jaime Guillermo Barrero Pérez.

INTRODUCCIÓN

El desarrollo continuo y la construcción de sistemas integrados que ocupan menos espacio y abarcan cada vez más aplicaciones ha permitido realizar procesos de manera más compacta y mucho más eficiente, logrando así mayores prestaciones con un menor consumo de potencia.

El grupo de investigación CEMOS, en asocio con la Escuela de Ingeniería Metalúrgica han desarrollado varios proyectos para el laboratorio de corrosión y últimamente se trazaron como objetivo desarrollar un instrumento que permita analizar el espectro de impedancia eléctrica para determinar la corrosión. Los prototipos desarrollados, generan una señal de tensión para excitación que se aplica a la celda, se mide la corriente resultante muestreándose por separado y se hace el cálculo de impedancia en magnitud y fase de forma separada.

Este proyecto plantea el reto de realizar de forma integrada todos estos pasos en un solo dispositivo; con la ayuda de un ASIC (Circuito integrado de propósito específico) AD5933, mediante su caracterización y puesta en marcha buscando desarrollar un equipo compacto.

Este documento explica el diseño y construcción de un prototipo para medir los valores de impedancias mediante una celda Dummy, cabe destacar que en este trabajo no se intentará plantear ni apoyar ninguna teoría en el campo de la Ingeniería Metalúrgica, es el inicio para el desarrollo de un Medidor de Impedancias para medir corrosión donde la generación y la medición se puedan realizar en un sistema integrado a partir del ASIC AD5933 de Analog Devices.

El documento se encuentra dividido en tres capítulos. En el primer capítulo se presentan las características generales de los elementos del proyecto y de los principales elementos que hacen parte del dispositivo. En el segundo capítulo se presenta el software que se desarrolló para el correcto funcionamiento del equipo, la comunicación entre el DSP y el ASIC, además de la programación de la pantalla. En el último capítulo se muestra como se llevan a cabo las pruebas desarrolladas al prototipo; el despliegue gráfico y las mediciones en el ASIC.

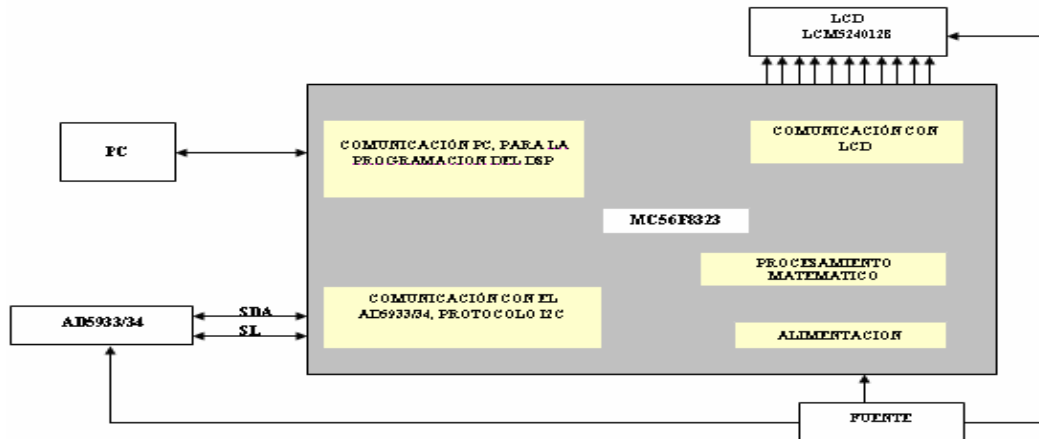
1. DESCRIPCIÓN GENERAL

Este capítulo del texto describe el concepto de celda **Dummy**, con este tipo de celdas se logra modelar el comportamiento de una celda electroquímica, siendo una forma sencilla de constatar y comprobar el correcto funcionamiento del Medidor de Impedancia, ya que a partir de valores conocidos de resistencia y condensadores se obtienen valores reales que sirven como parámetros de comprobación. El dispositivo construido consta de unos elementos que en conjunto permiten medir los parámetros de una celda **Dummy**.

Cuenta con la ayuda del ASIC¹ AD5933 de Analog Devices el cual permite realizar la medición de la impedancia, con este elemento se logra generar una señal de excitación, consigue también medir el efecto de esta excitación en la impedancia desconocida y convierte la respuesta de la medición de forma analógica a digital, después aplica la transformada discreta de Fourier (DFT) para obtener una salida con dos valores uno R real y otro I imaginario.

Las anteriores operaciones son llevadas a cabo con la ayuda de un dispositivo maestro (el AD5933 funciona como esclavo) desde el cual se establece la comunicación utilizando el protocolo I²C para establecer todos los parámetros que garantizan el óptimo funcionamiento del dispositivo. El tipo de maestro utilizado fue el DSP MC56F8323 que sirve como eje central entre la programación y comunicación del ASIC y el despliegue gráfico de la respuesta en la pantalla LCD de Hyundai. En la figura 1 se puede ver un esquema que muestra la disposición de los elementos en el prototipo.

Figura 1 Diagrama de Bloques



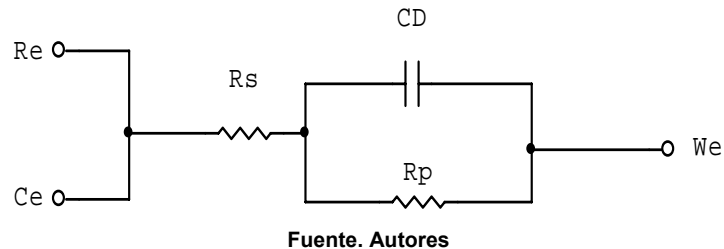
Fuente. Autores

¹ Acrónimo de Application Specific Integrated Circuit.

1.1. MODELAMIENTO DE UNA CELDA DUMMY

Las celdas Dummy son celdas de prueba, las cuales permiten representar una celda electroquímica mediante un arreglo de elementos de circuitos eléctricos, tales como resistencias y condensadores (ver figura 2).

Figura 2 Circuito Equivalente de la Celda Electroquímica



R_S : representa el efecto resistivo de la solución.

R_P : resistencia a la polarización.

C_D : representa el efecto capacitivo de la interfase de unión probeta solución.

Al excitar con una señal de tensión senoidal y variar la frecuencia, la magnitud de la impedancia del condensador cambia de forma inversa, ya que al aumentar la frecuencia, la impedancia disminuye. En cambio el valor de una resistencia se mantiene constante sin importar el valor de frecuencia.

En la celda, para altas frecuencias la impedancia causada por el condensador C_D tiende a ser mucho más pequeña que la impedancia de la resistencia R_P ; puesto que C_D está en paralelo con R_P y la impedancia causada por el condensador se puede modelar como un corto para. Entonces para altas frecuencias la celda es casi enteramente R_S . A bajas frecuencias, el condensador se puede modelar como un circuito abierto y éste puede ser excluido del circuito equivalente.

La impedancia de la Celda es entonces la combinación de dos resistencias en serie, R_S y R_P . Es decir, en los límites de alta y baja frecuencia, la celda se comporta principalmente como una resistencia. La componente imaginaria de la impedancia es muy pequeña y no cambia con la frecuencia, por lo tanto el ángulo de fase entre la tensión y la corriente en la celda es cercano a 0 grados en los valores de frecuencia mencionados anteriormente.

A frecuencias intermedias aparece un componente imaginario en la medida total de la impedancia, esto debido a que a dichas frecuencias el valor de la parte capacitiva del paralelo va adquiriendo una importancia en el circuito que propicia un desfase entre la señal de tensión y la de corriente. En este rango de frecuencias la impedancia es inversamente proporcional a la frecuencia de excitación de la celda.

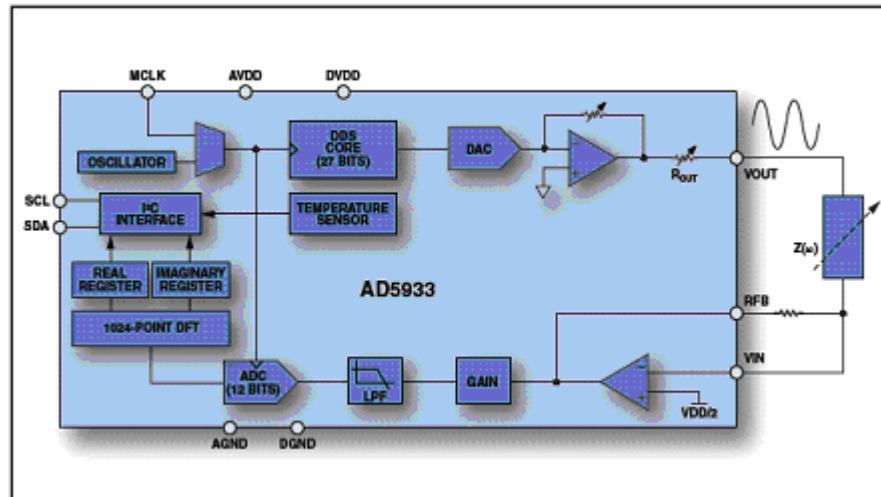
1.2 AD5933. CONVERTOR DE IMPEDANCIA

El AD5933 y AD5934 son los primeros conversores integrados de impedancia del mundo². El AD5933 (ver figura 3) es un sistema medidor de impedancia de alta precisión que combina un generador de frecuencia interno con un conversor analógico digital con una resolución de 12-bit y velocidad de muestreo de 1 MSPS. El generador de frecuencia permite que una impedancia compleja externa sea excitada a una frecuencia conocida. La señal de respuesta de la impedancia es muestreada por el ADC internamente y realiza una transformada discreta de Fourier (**DFT**). El algoritmo de DFT retorna un valor (R) real y otro (I) imaginario en cada frecuencia de la salida el cual es almacenado en diferentes direcciones del ASIC. El rango de impedancias que pueden ser medidas por este elemento va desde 100 Ω a 10 M Ω .

El AD5933 tiene gran aplicación en el área de la supervisión de la corrosión. La corrosión de metales, tales como aluminio y acero, puede dañar las infraestructuras, aviones, barcos y autos. Los daños causados por la corrosión, si no están en supervisión y posterior tratamiento, pueden conducir a una falla prematura que requeriría reparaciones costosas y/o el reemplazo de los elementos. En muchos casos, si el inicio de la corrosión se puede detectar, puede ser contrarrestado o retardado, disminuyendo el costo en las reparaciones o retardando el tiempo de reemplazo.

La inspección visual se emplea actualmente para detectar la corrosión, sin embargo, esto resulta en una operación costosa, que implica gran pérdida de tiempo, además de la dificultad de realizarla en zonas de difícil acceso.

Figura 3 Diagrama de bloques del AD5933



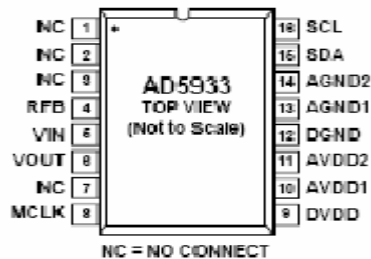
Fuente. Hoja de datos Analog Devices.

Fuente: Analog Devices hoja de datos AD5933/34 2005

El AD5933 permite la programación de sus parámetros internos, como la frecuencia de inicio, los pasos de frecuencia, el ancho de los pasos y la amplitud de la señal senoidal de excitación, además del PGA³ el cual tiene dos valores predefinidos de x 1 o x 5, este valor es determinado en el momento que se definen los parámetros en el código, para el caso específico se utiliza el valor de x1 en el PGA.

1.2.1. Diagrama de pines:

Figura 4 Diagrama de pines AD5933



Fuente: Hoja de Datos de Analog Devices AD5933/34 2005

Tabla 1 asignación de pines

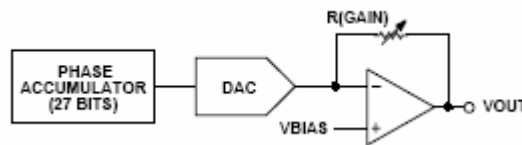
Pin		Descripción
1,2,3,7	NC	No conectado
4	RFB	Realimentación. Pin 4 conectado a pin 5 utilizado para fijar ganancia.
5	VIN	Pin de alimentación de 2.7 a 5.5 V.
6	VOUT	Salida. Señal de excitación
8	MCLK	Reloj Principal del sistema provisto por el usuario
9	DVDD	Tensión de fuente digital
10	AVDD1	Tensión de fuente analógica 1
11	AVDD2	Tensión de fuente analógica 2
12	DGND	Tierra digital
13	AGND1	Tierra analógica 1
14	AGND2	Tierra analógica 2
15	SDA	Entrada I ² C. I/O
16	SCL	Entrada I ² C .I/O reloj de sincronía

Fuente: Hoja de Datos de Analog Devices AD5933/34 2005

³ Acrónimo de programmable gain amplifier-Amplificador de ganancia programable

1.2.2. Etapa de Generación. La figura 5 presenta el esquema de generación de la señal en el AD5933 a partir de una serie de instrucciones dadas al acumulador DDS de fase de 27 bits este provee una salida de excitación a una tensión específica de un determinado rango de tensiones (ver tabla 1), con frecuencia de arranque definida por el usuario; sin embargo los 3 bits (MSB) de los 27 se colocan internamente en bajo (0), por lo tanto el usuario tiene la capacidad de programar solamente los 24 bits más bajos del registro de frecuencia de inicio.

Figura 5 AD5933 etapa de generación.



Fuente: Hoja de Datos de Analog Devices AD5933/34 2005

La resolución de la frecuencia se programa en una palabra 24-bit cargada en serie sobre el interfaz i^2c al registro del incremento de frecuencia. El barrido de la frecuencia es descrito completamente por la programación de tres parámetros: la frecuencia del comienzo, el incremento de la frecuencia, y el número de incrementos.

Tabla 2 Amplitudes de Tensión de salida.

Amplitud de Tensión de salida	Nivel de salida de DC
1.98 Vpp	1.48 V
0.97 Vpp	0.76 V
383 mVpp	0.31 V
198 mVpp	0.173 V

Fuente: Hoja de Datos de Analog Devices AD5933/34 2005

➤ **Frecuencia de Inicio (Start frequency)**

Esta es una palabra de 24 bits que es almacenada en 3 direcciones de 8 bits cada una dentro de la memoria **Ram** del integrado (registro 82h, 83h, 84h) y permite dar inicio al barrido de frecuencia para la medición y la excitación de la señal de salida. Para poder cargar el valor de la frecuencia de inicio, se debe desarrollar un código establecido por la ecuación 1 como se muestra a continuación:

$$S_{frequency} = \left[\frac{R_{OutputSfrequency}}{\frac{MCLK}{4}} \right] \times 2^{27} \quad [1]$$

ROutputSFrecuency = Frecuencia de inicio requerida para la salida.

MCLK= Reloj maestro interno.

SFrecuency= Código de frecuencia de inicio.

➤ **Incremento de frecuencia (Frequency Increment)**

Al igual que la frecuencia de inicio, esta es una palabra de 24 bits que es programada en 3 registros de direcciones (85 h, 86 h, 87 h) y requiere de un código para poder ser cargado en estas direcciones, el cual esta dado por la ecuación 2.

$$FIncrement = \left[\frac{RfrequencyIncrement}{\frac{MCLK}{4}} \right] x 2^{27} \quad [2]$$

FIncrement: código de frecuencia de incremento

RFrequency Increment: Incremento de frecuencia requerido

MCLK: reloj maestro.

➤ **Numero de Incrementos (Number of Increments)**

Esta es una palabra de 9 bits, la cual representa el número de puntos de frecuencia para su barrido y es programada en las direcciones 88h y la 89h, el valor máximo de incrementos es de 511 puntos.

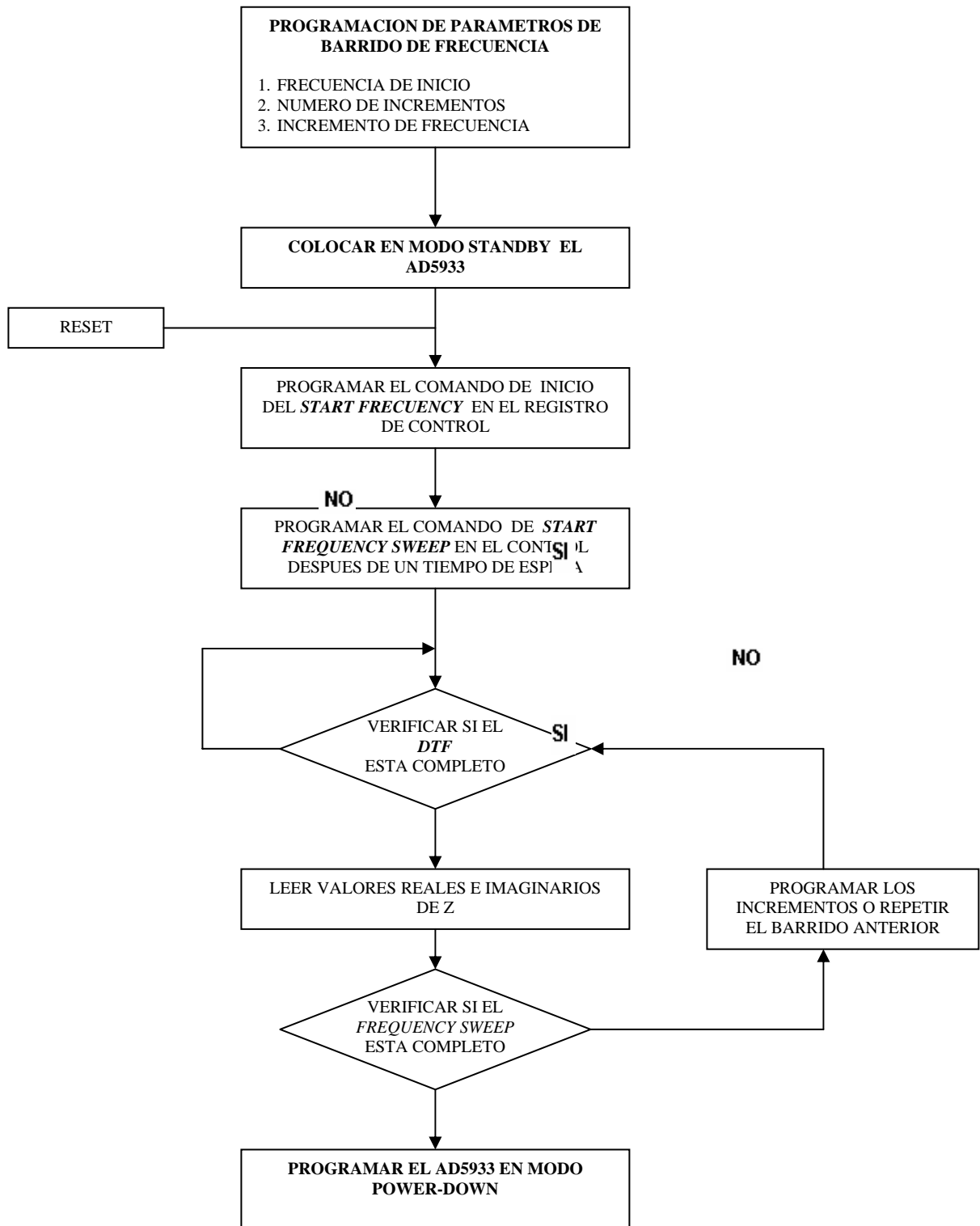
Una vez que se hayan programado los tres valores de parámetro, el barrido es iniciado enviando un comando de inicio barrido de frecuencia al registro de control en la dirección 80 h y la dirección 81 h. 2 bits en el estado de registro indica la finalización de la medición en la frecuencia para cada punto de barrido.

El incremento al punto siguiente del barrido de la frecuencia está bajo control del usuario. El resultado medido se almacena en cuatro registros (94 h, 95 h y 96 h, 97 h) que deben ser leídos antes de enviar un comando de incremento de frecuencia al registro de control para moverse al punto siguiente del barrido.

Existe la facilidad de repetir la medición actual del punto de frecuencia, publicando un comando de repetición de frecuencia al registro de control. Esto tiene la ventaja de permitir que el usuario haga un promedio de lecturas sucesivas logrando mayor confiabilidad en las mediciones.

En la figura 6 se muestra la secuencia de comandos para el barrido de frecuencias.

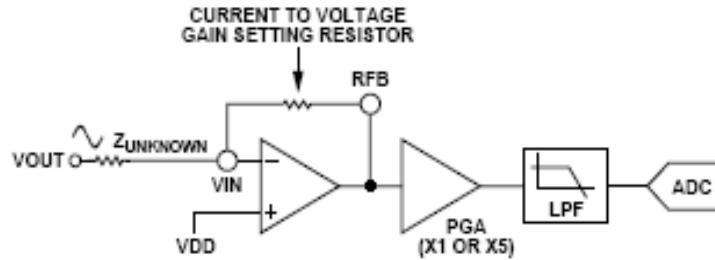
Figura 6 Diagrama de flujo para la programación del AD5933



Fuente: Autores

1.2.3. Etapa de Digitalización. Esta etapa consta de un amplificador de transresistencia, seguido de una PGA (amplificador de ganancia programable), un filtro pasa bajos y un ADC, como se muestra en la figura 7. La impedancia desconocida se conecta entre los pines VOUT y VIN, la señal de corriente que se desarrolla en la impedancia fluye al pin VIN y crea una señal de tensión en el convertidor de corriente a tensión.

Figura 7 AD5933 Etapa de digitalización.



Fuente: Hoja de Datos de Analog Devices AD5933/34 2005

Pasos de programación:

1. La ganancia del amplificador es determinada por el usuario seleccionando el valor de la resistencia de realimentación (RFB) conectada entre el pin 4 y el pin 5, la escogencia de este valor junto con el PGA debe mantener la señal en la región lineal para la entrada a el ADC (0 a VDD volts).
2. La ganancia del PGA es seleccionado por el usuario y puede ser x 1 o x 5 dependiendo del estado de control del bit D8.
3. La señal es enviada a traves de un filtro paso bajo y convertido por el ADC que tiene una resolución de 12 bits.

1.2.4 Cálculo De La Impedancia. El primer paso para el cálculo de la impedancia en cada punto de frecuencia es calcular la magnitud de la **DFT**, para calcular la magnitud dla DFT se utiliza la ecuación 3, en donde se toman valores reales e imaginarios almacenados en diferentes memorias del ASIC.

La magnitud esta dada por $Magnitude = \sqrt{R^2 + I^2}$ [3].

La fase esta dada por $Phase = Tan^{-1}\left(\frac{I}{R}\right)$ [4].

Tabla 3 posiciones de memoria de la parte real e imaginario de DFT.

VALOR REAL			
94H MSB	D15-D8	SOLO LECTURA	DATOS COMPLEMENTARIOS
95H LSB	D7-D0	SOLO LECTURA	
VALOR IMAGINARIO			
96H MSB	D15-D8	SOLO LECTURA	DATOS COMPLEMENTARIOS
97H LSB	D7-D0	SOLO LECTURA	

Fuente: Hoja de Datos Analog Devices AD5933/34 2005

Los valores de R e I utilizados en las ecuaciones 3 y 4, se refieren al valor real e imaginario respectivamente; Los valores Reales R son almacenados en las direcciones de registro 94h y 95h, mientras que el valor imaginario I es almacenado en las direcciones de registro 96h y 97h.

Ejemplo:

$$Real \cdot register = 038B \cdot hex = 907 \cdot decimal$$

$$Imaginary \cdot register = 0204 \cdot hex = 516 \cdot decimal$$

$$Magnitude = \sqrt{907^2 + 516^2} = 1043.506$$

Para convertir esta magnitud de DFT en un valor de impedancia, se deben realizar los siguientes pasos: se debe multiplicar un factor de ganancia por la magnitud dla DFT, al resultado de la anterior operación se le debe sacar el inverso, y así poder hallar el valor de impedancia; el factor de ganancia usado para el cálculo de la impedancia, es hallado durante la calibración del sistema con una impedancia conocida conectada entre los pines VOUT y VIN, una vez el factor de ganancia ha sido calculado, puede usarse este para determinar cualquier impedancia desconocida.

El siguiente ejemplo muestra como se puede determinar el factor de ganancia. Se tienen las siguientes características específicas.

$$V_{out} = 2 \text{ V (p-p)}$$

$$Z_{calibracion} = 200 \text{ k}\Omega.$$

$$Z_{impedancia conocida} = 200 \text{ k}\Omega$$

PGA = x1

La frecuencia de calibración = 30 kHz.

Después de realizar la configuración anterior, se procede a la revisión de los registros de almacenamiento real e imaginario, los datos leídos para esta configuración son los siguientes:

Real · register = F064 · hex = -3996 · decimal

Imaginary · register = 227E · hex = 8830 · decimal

$$\text{Magnitud} = \sqrt{(-3996)^2 + 8830^2} = 9692.106$$

$$\text{Gain} \cdot \text{Factor} = \left(\frac{\text{Admi tan cia}}{\text{codigo}} \right) = \left(\frac{1}{\frac{\text{Im pedancia}}{\text{Magnitud}}} \right) = \left(\frac{1}{\frac{200 \text{ K}\Omega}{9692.106}} \right) = 515.819 \text{ E} - 12 \quad [5].$$

$$\text{IMPEDANCIA} = \left[\frac{1}{\frac{\text{Factor ganancia} * \text{Magnitud DFT}}{1}} \right] = \frac{1}{515.819273 \text{ exp} - 12 * 1043.506} = 200.025 \text{ K} \quad [6].$$

De esta forma se logra determinar la magnitud de la impedancia a partir de los parámetros obtenidos del ASIC.

1.2.5. Protocolo I2C. La programación del AD5933 se lleva a cabo mediante la interfaz serial I2C. Este se conecta a este bus como un dispositivo esclavo bajo el mando de un dispositivo maestro. El AD5933 tiene 7-bits de dirección de esclavo en el bus serial. Cuando el dispositivo se inicia, tiene una dirección del bus predefinida, 0001101 (0D hex.).

El protocolo de I2C general opera como se describe en esta sección. La figura 8 muestra el diagrama general de tiempo para lectura y escritura, operaciones usadas por la interfaz I2C.

El maestro inicia la transferencia de datos estableciendo una condición inicio, definida como la transición de alto a bajo en la línea de datos serial (SDA) mientras la línea del reloj de serie (SCL) permanece en alto. Esto indica que sigue un arreglo de datos. El esclavo responde a la condición de inicio y cambia en los próximos 8 bits, que consisten en una dirección de esclavo de 7-bits (MSB primero) más un bit de R/W que determina la dirección de transferencia de los datos, si los datos serán escritos o leídos del dispositivo esclavo (0 = escriba, 1 = lea).

El esclavo responde enviando los datos durante el período bajo antes del noveno pulso del reloj, conocido como el bit de reconocimiento y sosteniéndolo en bajo durante el período alto de este pulso del reloj. Todos los otros dispositivos en el bus permanecen sin cambios mientras el dispositivo seleccionado espera por

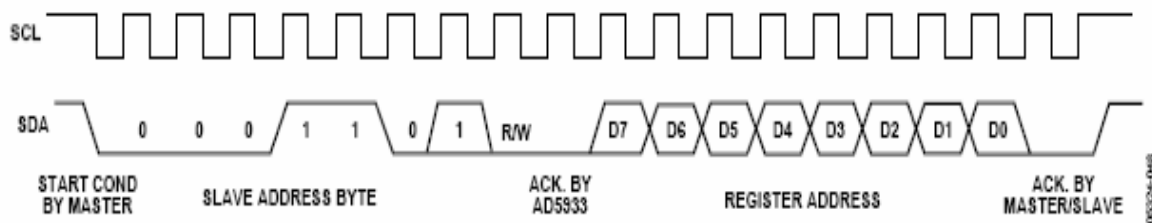
datos para ser leídos de o escritos a este. Si el bit R/W es 0, entonces el maestro escribe a el dispositivo esclavo. Si el bit R/W es 1, el maestro lee del dispositivo esclavo.

Los datos se envían sobre el bus serial en secuencias de nueve pulsos del reloj, 8 bits de datos seguidos por un bit de reconocimiento que puede provenir del dispositivo maestro o del esclavo. La transmisión de datos en la línea de datos debe ocurrir durante el periodo bajo de la señal de reloj por que una transición de bajo a alto cuando el reloj esta en alto puede tomarse como una señal de parada.

Si la operación es una operación de escritura el primer byte de datos después de la dirección del esclavo es un byte comandos. Esto le dice al dispositivo esclavo qué esperar; esto puede ser una instrucción diciéndole que espere un bloque de escritura o este puede ser una dirección de registro que dice al dispositivo esclavo donde deben ser escritos los datos subsecuentes.

Debido a que los datos pueden fluir en solo una dirección según lo definido por el bit R/W, no es posible enviar un comando al dispositivo esclavo durante la operación de lectura. Antes de realizar la operación de lectura, es necesario realizar una operación de escritura, indicándole al esclavo que clase de operación de lectura debe esperar y/o la dirección en la cual los datos deben ser leídos.

Figura 8 Representación de la transmisión de los datos



Fuente: Analog Devices Datasheet AD5933/34 2005

1.2.6. Escritura/Lectura al AD5933. La especificación de la interfaz define diferentes datagramas para las operaciones **read /write**. Esta sección describe los protocolos usados en el AD5933. Las órdenes para detener, confirmar o detener la comunicación son las siguientes:

- S** Start
- P** Stop
- R** Read
- W** Write
- A** Acknowledge
- /A** No acknowledge

A continuación se presentan los diferentes Datagramas, para las diferentes funciones determinadas para escribir o leer del ad5933.

➤ **Write Byte/Command Byte**

S	DIRECCION ESCLAVO	W	A	DIRECCIÓN DE REGISTRO	A	DATOS	A	P
---	----------------------	---	---	--------------------------	---	-------	---	---

➤ **Block Write**

➤ **Receive Byte**

S	DIRECCION ESCLAVO	W	A	BLOCK WRITE	A	NUMERO DE BYTES	A	BYTE O	A	BYTE 1	A	P
		S		DIRECCION ESCLAVO		R	A	DATOS	/A	P		

➤ **Block Read**

S	DIRECCION ESCLAVO	W	A	BLOCK READ	A	NUMERO DE BYTES	A	S
---	----------------------	---	---	---------------	---	--------------------	---	---

DIRECCION ESCLAVO	R	A	BYTE O	A	BYTE 1	/A	P
-------------------	---	---	--------	---	--------	----	---

El orden de codificación se muestra en la tabla 4, estos comandos son usados para el **lectura/escritura** a la interfaz.

Tabla 4 Comandos para escritura y lectura de datos

CODIGO DE COMANDO	NOMBRE DEL CODIGO	DESCRIPCION
1010 0000	BLOCK WRITE	SIRVE PARA ESCRIBIR MULTIPLES BYTES
1010 0001	BLOCK READ	USADO PARA LEER MULTIPLES BYTES
1011 0000	ADDRESS POINTER	USADO PARA ACCEDER A LAS DIRECCIONES DE MEMORIA

Fuente: Analog Devices Datasheet AD5933/34 2005.

1.3. DSP MC56F8323

El DSP MC56F8323 pertenece a la familia Motorola 56800 de los controladores híbridos. Este combina en un chip el poder de procesamiento de un DSP y la funcionalidad de un microcontrolador para crear una solución efectiva y económica. Debido a su bajo costo, flexibilidad de la configuración, y la robustez

de su código de programación, son muy útiles en diversas aplicaciones de ingeniería, como las comunicaciones, el control y la automatización industrial.

1.3.1. Criterio De Selección De Un DSP. Existen diversos fabricantes y tipos de DSP's en la actualidad, por eso a la hora de escoger el más eficiente para la aplicación se deben tener en cuenta diversos parámetros o criterios de selección, dentro de los cuales se tienen:

- Precio.
- Confiabilidad.
- Tamaño.
- Consumo de potencia.
- Facilidad acceso.

La principal razón por la escogencia de este dispositivo radica en que el grupo **CEMOS** contaba con sistemas de desarrollo del MC56F801 y MC56F8323, pero el último contaba con una mayor cantidad de pines de propósito general ya que el MC56F8323 cuenta con 27 GPIO⁴ (pin de salida/entrada de propósito general) y el MC56F801 cuenta con solo 11 GPIO, los cuales eran insuficientes si se deseaba habilitar las aplicaciones usadas como la comunicación con el ASIC, los pulsadores de control y la transmisión paralelo para el despliegue gráfico y además de GPIOs libres para próximos desarrollos.

Para la programación y comunicación del Dsp con el AD5933, se cuenta con la ayuda de los Beans, los cuales son programas ya creados por freescale y pueden ser agregados al proyecto a realizar, los beans permiten emular por medio del software el protocolo I2C, interrupciones, comunicación en bytes, bits, sin necesidad de que el usuario deba realizar el programas de habilitación de puertos.

1.3.2. Características Del MC56F8323. EL DSP posee las siguientes características:

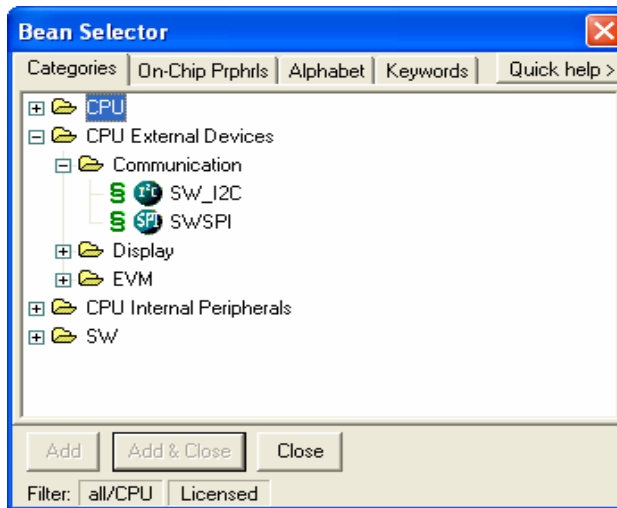
- Un módulo de PWM con seis salidas y tres entradas programables.
- Mas de 60 millones instrucciones por segundo a 60Mhz.
- Cuatro acumuladores de 16 bits.
- Cuatro bus de datos internos.
- Tres bus de direcciones internas.
- Dos interfaces periféricas serie (SPIs).
- Comunicación I2C en modo maestro (emulado).
- Diodo sensor de Temperatura para supervisar la temperatura del chip.
- Regulador de Tensión de 3.3V a 2.6V en el chip.
- Software programable de laso de fase cerrado (PLL).
- Oscilador en chip.

⁴ General Purpose I/O pins

- 27 pines I/O de uso general (GPIO).
- 32 KB de memoria flash.
- **PLL**. El PLL permite programar la frecuencia a la cual va a trabajar el

1.3.3. Protocolo I2C en el DSP. El protocolo I2C en el DSP se debe emular, ya que este no lo utiliza como periférico interno, para esto se debe añadir un **Bean** del procesador experto, como se muestra en la figura 9.

Figura 9 Selector de Bean



Fuente: CodeWarrior Procesador Experto www.processorexpert.com

Al añadir este **Bean** al proyecto se genera un código sobre el cual el DSP se basa para establecer la comunicación con un dispositivo externo, para el cual se deben tener en cuenta diferentes condiciones, como lo es asignación de dos pines del DSP uno para reloj (SCL) y otro para datos (SDA), la dirección de inicialización **Slave Address**, la velocidad de transmisión al bus de datos (**FAST MODE, STANDARD MODE**), y la condición de parada, la cual se puede hacer automática o se puede establecer mediante un código de programación colocando en alto tanto el reloj como la transmisión de datos, estos parámetros se establecen en la ventana de inicialización del I2C, como se muestra en la figura 10.

Figura 10 Inicialización de periféricos

Bean		Items	Visibility	Help	<	>
Properties		Methods	Events	Comment		
✓	Bean name	EI2C1				
>	Data pin	InOutPin	▼	>		
>	Clock pin	InOutPin	▼	>		
✓	Mode selection	MASTER	▼			
✓	Slave address	8	Ⓜ			
✓	Clock timeout	16				
✓	Output buffer size	0				
✓	I2C bus mode	STANDARD MODE	▼			
✓	Automatic stop condition	yes	Ⓜ			
✓	Acknowledge polling trials	2000				
☐	Initialization					
✓	Slave address init	8	Ⓜ			
✓	Enabled in init. code	yes	Ⓜ			
✓	Events enabled in init.	yes				
☐	CPU clock/speed selection					
✓	High speed mode	Enabled	Ⓜ	This bean is enabled		
✓	Low speed mode	Enabled	Ⓜ	The CPU speed mode is disabled		
✓	Slow speed mode	Enabled	Ⓜ	The CPU speed mode is disabled		

Fuente: CodeWarrior Procesador Experto www.processorexpert.com

Cuando todos estos parámetros han sido establecidos, se procede a establecer la selección del esclavo al cual se desea enviar o recibir datos, lo cual es posible debido a las siguientes funciones:

SelectSlave: Esta función permite determinar el esclavo con el cual se desea establecer la comunicación.

SendChar: Permite enviar la orden que se desea transmitir (byte).

RecvChar: Permite recibir los datos generados desde el esclavo.

1.3.4. Escritura/Lectura al DSP 56f8323. Para la transmisión y recepción de datos se usan diferentes Datagramas, ya que se deben establecer distintas condiciones de operación. Las órdenes para detener, confirmar o detener la comunicación son las siguientes.

- S** Start
- P** Stop
- R** Read
- W** Write

A Acknowledge
/A No acknowledge

A continuación se presenta una explicación de los diferentes datagramas para las diferentes condiciones de operación.

➤ **Write Byte/Command Byte (SendChar)**

S	DIRECCION ESCLAVO	W	A	DATOS	/A	P
----------	------------------------------	----------	----------	--------------	-----------	----------

Sin embargo este empaquetado esta incompleto para la comunicación con el ASIC, por consiguiente este no puede entender la comunicación, ya que este no permite la escogencia de una dirección de registro, a la cual se desea enviar datos, por consiguiente mediante programación se le añadió esa parte del protocolo y el DSP transmite de la siguiente forma:

S	DIRECCION ESCLAVO	W	A	DIRECCIÓN DE REGISTRO	A	DATOS	A	P
----------	------------------------------	----------	----------	----------------------------------	----------	--------------	----------	----------

➤ **Receive Byte (RecvChar)**

S	DIRECCION ESCLAVO	R	A	DATOS	/A	P
----------	------------------------------	----------	----------	--------------	-----------	----------

➤ **Block Write**

S	DIRECCION ESCLAVO	W	A	BLOCK WRITE	A	NUMERO DE BYTES	A	BYTE O	A	BYTE 1	A	P
----------	------------------------------	----------	----------	------------------------	----------	----------------------------	----------	-------------------	----------	-------------------	----------	----------

➤ **Block Read**

S	DIRECCION ESCLAVO	W	A	BLOCK READ	A	NUMERO DE BYTES	DE	A	S
----------	------------------------------	----------	----------	-----------------------	----------	--------------------------------	-----------	----------	----------

DIRECCION ESCLAVO	R	A	BYTE O	A	BYTE 1	/A	P
--------------------------	----------	----------	---------------	----------	---------------	-----------	----------

1.4 PANTALLA DE CRISTAL LÍQUIDO

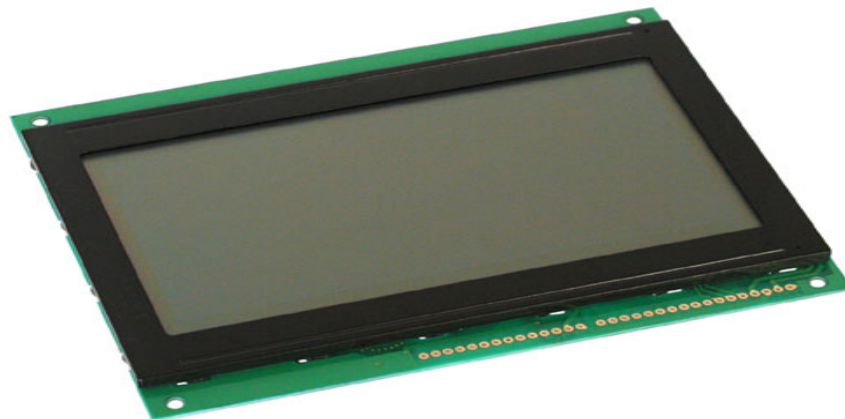
Para la presentación y la visualización de la forma de onda de los diagramas de Bode en magnitud y fase de la Impedancia medida, se utilizó la pantalla gráfica de cristal líquido de referencia HG25504NG-01 de Hyundai de 256 x 128 píxeles de resolución. Se seleccionó este módulo LCD debido a cuatro aspectos importantes: el área de display, la capacidad de combinar texto y gráficos, el generador interno de caracteres y su bajo costo⁵. En la tabla 5 se enumeran las principales características físicas de la pantalla.

Tabla 5 Características de la LCD HG25504NG-01 de Hyundai

Ítem	Característica	Unidad
Tamaño LCD (W X H X T)	147,0 x 116,0 x 12,0	mm
Área de despliegue(W X H)	127,0 x 70,0	mm
Tamaño de punto (W X H)	0,47 x 0,47	mm

Fuente: Hoja de datos de la pantalla HG25504NG-01 de Hyundai

Figura 11 LCD Gráfica HG25504NG-01 de Hyundai

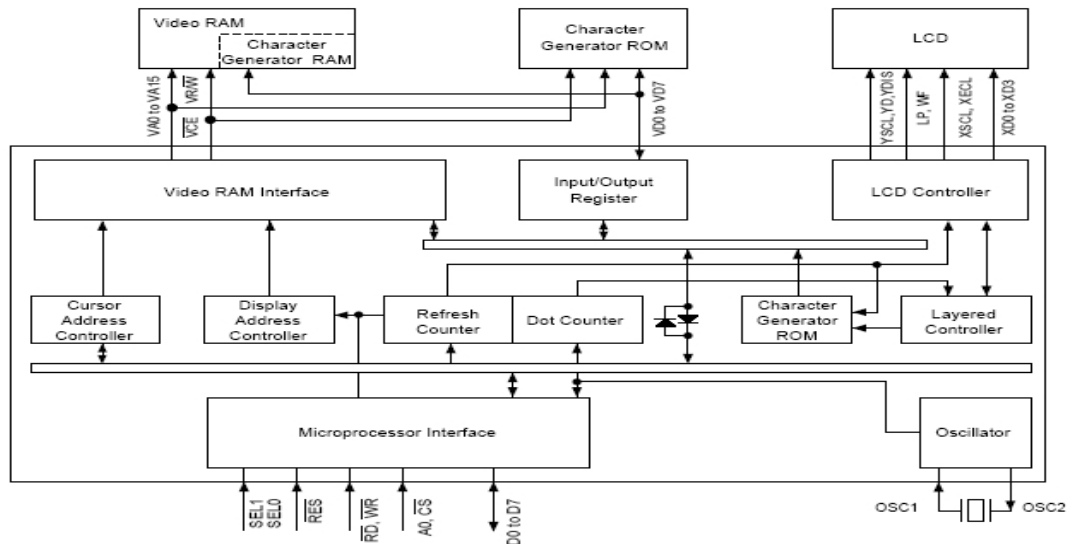


Fuente: All Electronics Corp.: www.allelectronics.com.

Esta pantalla posee un controlador gráfico fabricado por S-MOS Systems, de referencia SED1330F cuyo sistema trabaja con base en matrices de 5x7 puntos y puede combinar tanto texto como gráficos gracias a que posee tres capas de pantalla superpuestas que fueron configuradas la primera para texto y las otras dos para gráficos.

⁵ El costo de la LCD es de US\$18.5 sin tener en cuenta los gastos de impuestos y de importación.

Figura 12 Diagrama de bloques de la LCD



Hoja de Datos SED1330F/1335F/1336F's.

El controlador SED1330F tiene una memoria RAM y ROM con capacidad de 64 y 256 caracteres respectivamente, un bus de datos para la RAM de 8 o 16 bits y posibilidad de interfaz directa con el DSP en paralelo. Si se desea ahorrar pines de propósito general para conectar directamente el DSP al módulo LCD, es posible realizar una interfaz serie/paralelo utilizando un registro de desplazamiento; ya que el DSP elegido brinda la flexibilidad de tener 27 GPIO se utiliza el interfaz en paralelo entre la pantalla y el DSP.

La pantalla de texto puede desplegar 160 caracteres diferentes, generados a partir del código ASCII interno, en 16 filas por 32 columnas según se muestra en las siguientes figuras:

Figura 13 Distribución de la pantalla de texto

32 CARACTERES



Fuente: Hoja de datos lcd-101.

Figura 14 Generador interno de caracteres

		Character code bits 0 to 3															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Character code bits 4 to 7	2		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
	3	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	4	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	
	5	p	q	r	s	t	u	v	w	x	y	z	[\]	^	_
	6	~	ab	cd	ef	gh	ij	kl	mn	op							
	7	qr	st	uv	wx	yz	[\]	^	_	~					
	A		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
	B	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	C	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	
	D	p	q	r	s	t	u	v	w	x	y	z	[\]	^	_
	1	~	ab	cd	ef	gh	ij	kl	mn	op							

Fuente: Hoja de datos SED1330F/1335F/1336F's.

La pantalla gráfica dispone de 128 líneas por 32 columnas distribuidas de la siguiente manera:

Figura15. Distribución de la pantalla gráfica



Fuente: Hoja de datos SED1330F/1335F/1336F's

La memoria para texto en la LCD va en la dirección 0x0000 hasta la dirección 0x0FFF (4095 bytes), y la memoria para gráficos comienza en la dirección 0x1000 y termina en la dirección 0x1FFF (4095 bytes).

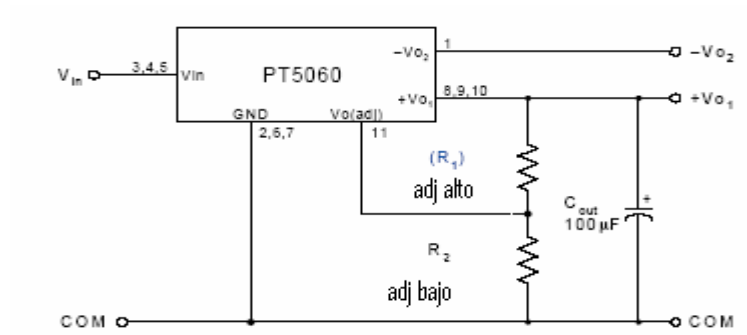
Según las especificaciones dadas por el fabricante la LCD HG25504 se alimenta con una tensión DC de 5V y con un contraste una tensión de entre -10V y -20V.

En esta aplicación se utilizó el circuito integrado PT5062²⁵ fabricado por Texas Instruments. La familia PT5060 (figura 16) son reguladores de tensión conmutados con salida dual de +/- 12 o +/- 15 V. a partir de una simple entrada de 5V. Estos reguladores pueden ser ajustados ya sea para disminuir (back converter) o para aumentar (boost converter) los niveles de Tensión en la salida.

Dependiendo de su aplicación, este arreglo se puede lograr a partir de una simple resistencia de realimentación. Este regulador integrado se utiliza para alimentar el circuito de contraste de la LCD (-12V), los amplificadores operacionales y los switches analógicos utilizados en el prototipo los cuales requieren alimentación dual. Los +5V utilizados para alimentar este circuito provienen del regulador de tensión LM78L05²⁶ (figura17) que se encuentra en la tarjeta de desarrollo del DSP, que en sus pines de entrada tiene conectada la alimentación de la tarjeta (+8V a 600mA). El contraste de la pantalla esta determinado por un potenciómetro de 10K Ω , que además regula la alimentación de la pantalla de tal forma que se cumpla con la condición de alimentación proporcionada por el fabricante ($V_{DD}-V_O=15.1V$ donde $V_{DD}=5V$). La figura 18 ilustra las conexiones realizadas para el

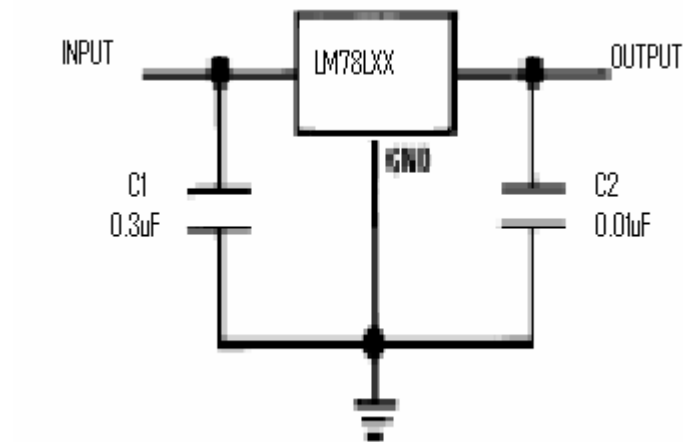
manejo y alimentación del módulo LCD y en la Tabla 7 se especifican los pines que se conectan al DSP para su manejo.

Figura 16 Diagrama de conexión del PT5060 de Texas Instruments



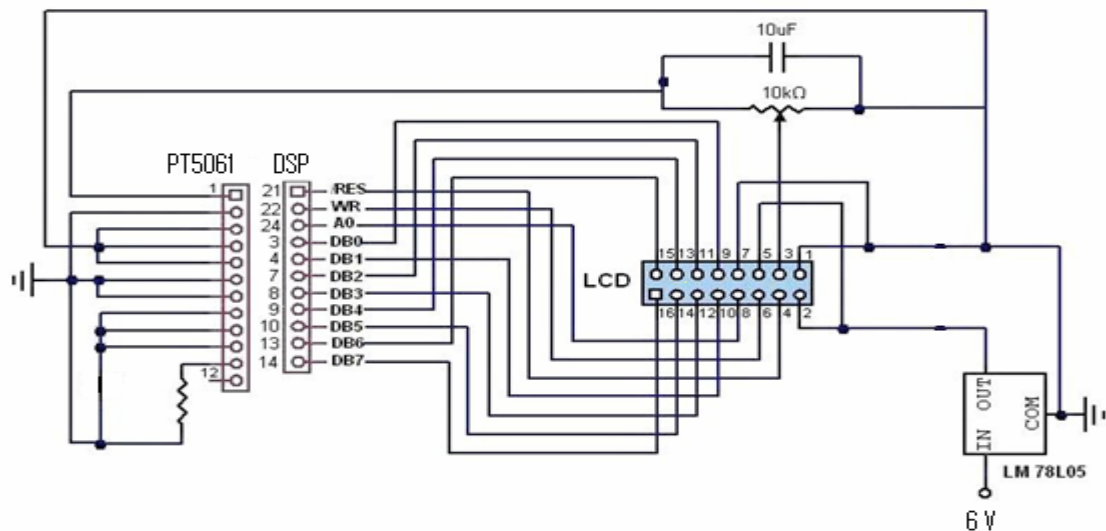
Fuente. Hoja de datos PT5060 series Texas Instruments

Figura 17 Diagrama del LM78L05



Fuente Hoja de datos Nacional Semiconductor.

Figura 18 Diagrama de conexión de la LCD.



Fuente Los Autores

La pantalla de cristal líquido posee un controlador que se encarga de su funcionamiento. Este utiliza 17 pines para su manejo, de los cuales 8 pines son entrada de datos (DB0 a DB7), 5 pines de control y 4 para alimentación. La descripción de estos pines se encuentra en la tabla 6.

Tabla 6 Descripción de los pines de la LCD

Nombre del Pin	Nivel	Descripción
FG	0 V	Tierra de carcasa
Vss (GND)	0 V	Tierra
Vdd (Vcc)	+5 V	Fuente de alimentación para la circuiteria lógica y LCD
-Vo	Potenciómetro	Tensión variable utilizado para el contraste de la LCD
/RES	H/L	Señal de Reset
/RD	H/L	Señal de Lectura
/WR	H/L	Señal de Escritura
/CS	H/L	Señal de elección del chip
A0	H/L	Señal para seleccionar el tipo de datos
DB0	H/L	Dato bit 0
DB1	H/L	Dato bit 1
DB2	H/L	Dato bit 2
DB3	H/L	Dato bit 3
DB4	H/L	Dato bit 4

DB5	H/L	Dato bit 5
DB6	H/L	Dato bit 6
DB7	H/L	Dato bit 7

Hoja de datos de la pantalla HG25504NG-01 de Hyundai.

La interfaz más conveniente para este prototipo es la interfaz paralela por la necesidad de disminuir el “tiempo muerto” del dispositivo. Para el acople de la pantalla con el DSP se dispuso de 11 pines que se configuraron como GPIO como se muestra en la tabla 7.

Tabla 7 Configuración de los pines GPIO para el manejo de la LCD

Nombre del Pin en la LCD	No del Pin en el DSP	Nombre del Pin en el DSP
/WR	7	GPIOB1
A0	9	GPIOB0
DB0	3	GPIOA0
DB1	4	GPIOA1
DB2	7	GPIOA2
DB3	8	GPIOA3
DB4	9	GPIOA4
DB5	10	GPIOA5
DB6	13	GPIOA6
DB7	14	GPIOA7

Hoja de datos de la pantalla HG25504NG-01 de Hyundai.

Para la inicialización de la pantalla, la configuración de las capas, la elección del tamaño del carácter, la configuración del cursor, etc., el controlador dispone de un listado de comandos que se describen a continuación. Adicionalmente en el capítulo 2 se describen los algoritmos implementados para la inicialización y manejo de la pantalla.

Tabla 8 Descripción de comandos del controlador gráfico

Clase	Comando	Valor en Hex	Descripción																
Control del sistema	SYSTEM SET	40	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>DB7</td><td>DB6</td><td>DB5</td><td>DB4</td><td>DB3</td><td>DB2</td><td>DB1</td><td>DB0</td> </tr> <tr> <td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td> </tr> </table> <p>Inicializa los dispositivos, activa el tamaño de ventana y de caracteres. Para esto dispone de ocho instrucciones (P1 a P8) por medio de las cuales se selecciona la ROM generadora de caracteres internos o externos, se escoge la configuración de la memoria para caracteres, se elige la altura de los mapas de bits de carácter, el método del controlador LCD, el tamaño del carácter horizontal y vertical, el número de bytes por longitud de una línea y el rango de direcciones horizontales de la pantalla virtual.</p>	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	0	1	0	0	0	0	0	0
	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0											
0	1	0	0	0	0	0	0												
SLEEP IN	53	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>DB7</td><td>DB6</td><td>DB5</td><td>DB4</td><td>DB3</td><td>DB2</td><td>DB1</td><td>DB0</td> </tr> <tr> <td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td> </tr> </table> <p>Suspende cualquier actividad en la LCD.</p>	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	0	1	0	1	0	0	1	1	
DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0												
0	1	0	1	0	0	1	1												
Control del Display	DISP ON/OFF	58/59	<p>Enciende o apaga la pantalla.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>DB7</td><td>DB6</td><td>DB5</td><td>DB4</td><td>DB3</td><td>DB2</td><td>DB1</td><td>DB0</td> </tr> <tr> <td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>D=0 Display apagado. D=1 Display encendido.</td> </tr> </table> <p>Dispone de una instrucción P1 por la cual se habilita o deshabilita el cursor.</p>	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	0	1	0	1	1	0	1	D=0 Display apagado. D=1 Display encendido.
	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0											
0	1	0	1	1	0	1	D=0 Display apagado. D=1 Display encendido.												
SCROLL	44	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>DB7</td><td>DB6</td><td>DB5</td><td>DB4</td><td>DB3</td><td>DB2</td><td>DB1</td><td>DB0</td> </tr> <tr> <td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td> </tr> </table>	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	0	1	0	0	0	1	0	0	
DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0												
0	1	0	0	0	1	0	0												

Activa el área de despliegue de la pantalla. Posee 10 instrucciones (P1 a P10) mediante las cuales se definen la dirección de inicio del área de despliegue y el número de

		líneas por bloque de despliegue.																															
CSRFORM	5D	<table border="1"> <tr> <th>DB7</th> <th>DB6</th> <th>DB5</th> <th>DB4</th> <th>DB3</th> <th>DB2</th> <th>DB1</th> <th>DB0</th> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>1</td> <td>1</td> <td>1</td> <td>0</td> <td>1</td> </tr> </table> <p>Activa la forma y el tamaño del cursor. Tiene dos instrucciones P1 y P2 por medio de la cual se escoge el tamaño horizontal del cursor y el tamaño vertical del cursor.</p>	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	0	1	0	1	1	1	0	1															
DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0																										
0	1	0	1	1	1	0	1																										
CGRAM ADR	5C	<p>Activa la dirección de comienzo de la RAM del generador de caracteres.</p> <table border="1"> <tr> <th>DB7</th> <th>DB6</th> <th>DB5</th> <th>DB4</th> <th>DB3</th> <th>DB2</th> <th>DB1</th> <th>DB0</th> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>1</td> <td>1</td> <td>1</td> <td>0</td> <td>0</td> </tr> </table>	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	0	1	0	1	1	1	0	0															
DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0																										
0	1	0	1	1	1	0	0																										
CSRDIR	4C a 4F	<p>Activa la dirección del movimiento del cursor.</p> <table border="1"> <tr> <th>DB7</th> <th>DB6</th> <th>DB5</th> <th>DB4</th> <th>DB3</th> <th>DB2</th> <th>DB1</th> <th>DB0</th> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>0</td> <td>1</td> <td>1</td> <td>CD1</td> <td>CD2</td> </tr> </table> <table border="1"> <tr> <th>CD1</th> <th>CD2</th> <th>Dirección del cursor</th> </tr> <tr> <td>0</td> <td>0</td> <td>Derecha</td> </tr> <tr> <td>0</td> <td>1</td> <td>Izquierda</td> </tr> <tr> <td>1</td> <td>0</td> <td>Arriba</td> </tr> <tr> <td>1</td> <td>1</td> <td>Abajo</td> </tr> </table>	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	0	1	0	0	1	1	CD1	CD2	CD1	CD2	Dirección del cursor	0	0	Derecha	0	1	Izquierda	1	0	Arriba	1	1	Abajo
DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0																										
0	1	0	0	1	1	CD1	CD2																										
CD1	CD2	Dirección del cursor																															
0	0	Derecha																															
0	1	Izquierda																															
1	0	Arriba																															
1	1	Abajo																															
HDOT SCR	5A	<p>Ajusta el tamaño del barrido horizontal de la pantalla por píxeles.</p> <table border="1"> <tr> <th>DB7</th> <th>DB6</th> <th>DB5</th> <th>DB4</th> <th>DB3</th> <th>DB2</th> <th>DB1</th> <th>DB0</th> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>1</td> <td>1</td> <td>0</td> <td>1</td> <td>0</td> </tr> </table> <p>Posee una instrucción P1 para elegir el número de píxeles para el área de despliegue.</p>	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	0	1	0	1	1	0	1	0															
DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0																										
0	1	0	1	1	0	1	0																										
OVLAY	5B	<p>Selecciona la composición en capas de la pantalla y el modo de la pantalla texto/gráfica por medio de una instrucción adicional P1.</p> <table border="1"> <tr> <th>DB7</th> <th>DB6</th> <th>DB5</th> <th>DB4</th> <th>DB3</th> <th>DB2</th> <th>DB1</th> <th>DB0</th> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>1</td> <td>1</td> <td>0</td> <td>1</td> <td>0</td> </tr> </table>	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	0	1	0	1	1	0	1	0															
DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0																										
0	1	0	1	1	0	1	0																										

Hoja de datos de la pantalla HG25504NG-01 de Hyundai.

2. SOFTWARE

Mientras el Hardware es el músculo del Medidor el software se puede considerar como el cerebro del medidor; la programación del AD5933 se lleva a cabo mediante el protocolo I2C, en el que se programa el barrido de frecuencia y medición de los valores reales e imaginarios de la impedancia, además el DSP permite el manejo de la pantalla y el JTAG.

2.1. AD5933.

La programación del ASIC se lleva a cabo a través del DSP en el software Codewarrior. Antes de iniciar la misma se debe tener en cuenta el mapa de registro el cual provee el control de las funciones los pasos de frecuencia, la frecuencia de inicio y otras funciones mas en el ASIC; como se ve en la tabla 9.

2.1.1. Mapa De Registro:

Tabla 9 Mapa de Registro AD5933/34

NOMBRE	DIRECCION	DATO DE REGISTRO	READ/WRITE
CONTROL	80H	D15-D8	READ/WRITE
	81H	D7-D0	READ/WRITE
START FREQUENCY	82H	D23-D16	READ/WRITE
	83H	D15-D8	READ/WRITE
	84H	D7-D0	READ/WRITE
FREQUENCY INCREMENT	85H	D23-D16	READ/WRITE
	86H	D15-D8	READ/WRITE
	87H	D7-D0	READ/WRITE
NUMERO INCREMENTOS	88H	D15-D8	READ/WRITE
	89H	D7-D0	READ/WRITE
NUMERO DE CICLOS TIEMPO	8AH	D15-D8	READ/WRITE
	8BH	D7-D0	READ/WRITE
ESTADO	8FH	D7-D0	READ ONLY
DATOS TEMPERATURA	92H	D15-D8	READ ONLY
	93H	D7-D0	READ ONLY
DATO REAL	94H	D15-D8	READ ONLY
	95H	D7-D0	READ ONLY
DATO IMAGINARIO	96H	D15-D8	READ ONLY
	97H	D7-D0	READ ONLY

Fuente: Hoja de datos Analog Devices AD5933/34 2005

➤ **Registro de control**

En este registro se le dan las órdenes necesarias para cumplir los diferentes modos posibles en el ASIC, estas órdenes son enviadas a las direcciones 80h y 81h, los comandos son los que se muestran en la tabla 10.

Tabla 10 Comandos de control AD5933

BIT	D15	D14	D13	D12	
	0	0	0	0	NO OPERACIÓN
	0	0	0	1	INICIO STAR FRECUENCY
	0	0	1	0	INICIO BARRIDO FECUENCIA
	0	0	1	1	INCREMENTO FRECUENCIA
	0	1	0	0	REPEAT FRECUENCY
	1	0	0	0	NO OPERACIÓN
	1	0	0	1	MEDIDA TEMPERATURA
	1	0	1	0	POWER DOWN MODE
	1	0	1	1	STANDBY MODE
	1	1	0	0	NO OPERACIÓN
	1	1	0	1	NO OPERACIÓN
D11					NO OPERACIÓN
		D10	D9		RANGO DE SALIDA TENSIÓN
		0	0		RANGO 1 (2 VPP)
		0	1		RANGO 4 (200 m VPP)
		1	0		RANGO 3 (400 m VPP)
		1	1		RANGO 2 (1 VPP)
D8					GANANCIA PGA 0=X5, 1=X1
D7					RESERVADO, 0
D6					RESERVADO, 0
D5					RESERVADO, 0
D4					RESET
D3		1			MCLK=1
		0			OSCILADOR INTERNO=0
D2		0			DEBE SER 0
D1					RESERVADO, 0
D0					RESERVADO, 0

Fuente: Hoja de Datos Analog Devices AD5933/34 2005

Es importante tener en cuenta que los 4 bits MSB son los que se programan para cambiar la configuración y el modo del AD.

2.1.2 Programación General del AD5933. El primer paso a seguir es el de definir las direcciones del esclavo (AD5933), en las cuales se va a dar una orden (escribir o leer).

Como segundo paso se aplica el protocolo I2C emulado en el Dsp y se usan las órdenes y comandos definidos en esta para escribir y leer valores, dentro de los cuales tenemos:

- **EI2C1_SendChar ()**: Comando para escribir en el ASIC.
- **EI2C1_RecvChar ()**: Comando para leer datos del ASIC.
- **EI2C1_SelectSlave ()** : Comando para definir la dirección del ASIC.

Después de haber definido los comandos a utilizar, se especifican las funciones del programa que permitan la programación del integrado y recibir los datos del mismo después de realizar la medición de los parámetros de celda dummy.

Funciones Usadas Para la programación del ASIC.

ini_frec (): Usada para programar el start Frequency

Incremento (): Usada para programar el paso de barrido.

Numero (): Usada para programar el número de pasos.

Reset (): reset

Mclk (): Usada para habilitar el reloj interno o externo

Ciclos (): Usada para programar el número de ciclos de la señal de salida.

Standby (): Usada para programar el standby mode.

Inicio (): Usada para programar el barrido.

Datos (): Usada para la recepción de datos de impedancia.

Barrido de Frecuencia. Para desarrollar el barrido en frecuencia, se usan varias funciones declaradas en el programa, dentro de las cuales se programan al ASIC la frecuencia de inicio, la cantidad de pasos, el paso de barrido y la cantidad de ciclos que se desea hacer por frecuencia.

Las funciones utilizadas son:

ini_frec ()

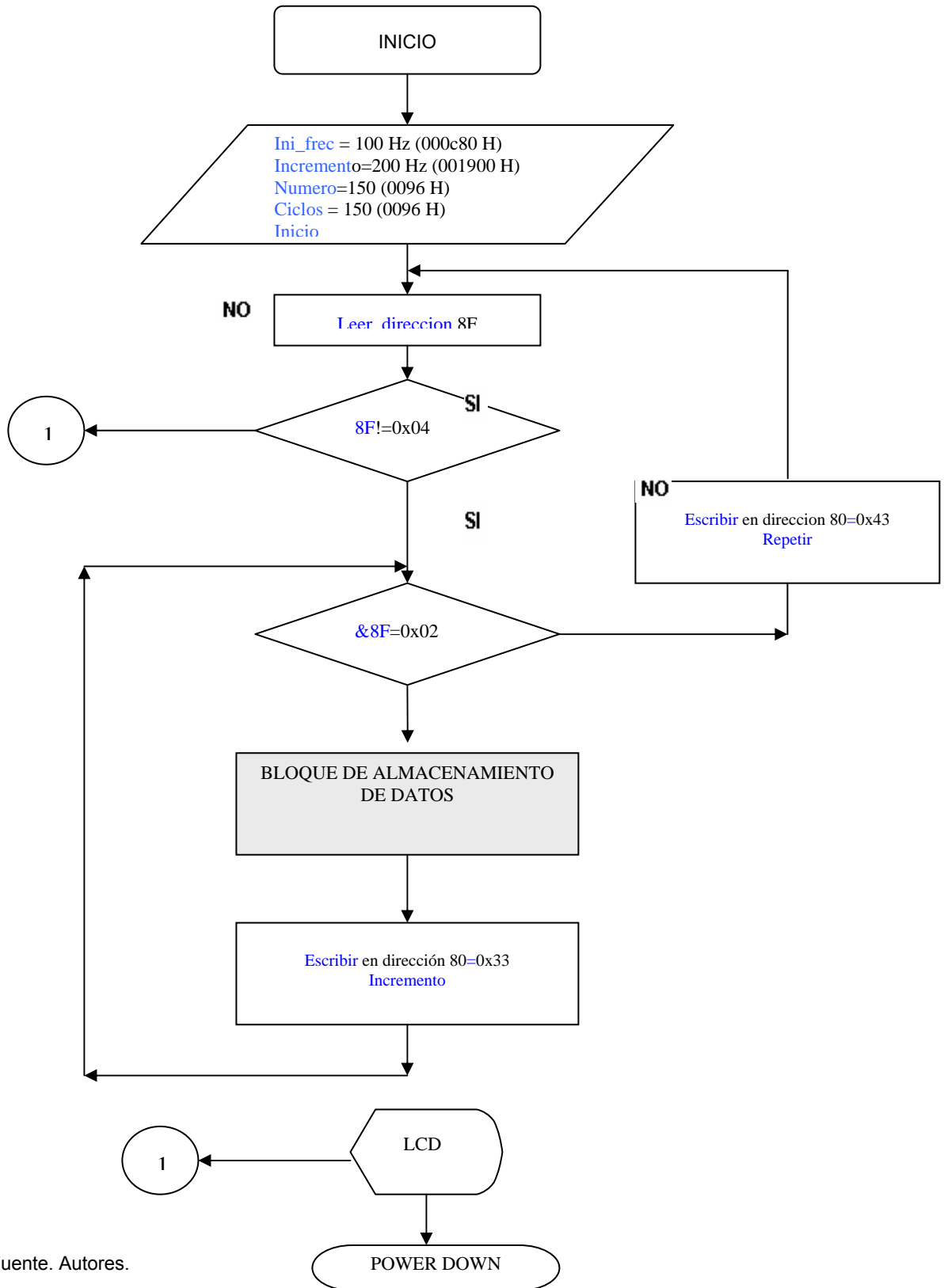
Incremento ()

Numero ()

Ciclos ()

En la figura 19 se presenta diagrama de flujo donde se muestra el orden en que se programación de estas funciones.

Figura 19 Diagrama de Flujo para el barrido de frecuencia



Fuente. Autores.

Tensión de Salida y PGA. En esta parte del Software de manejo del AD5933, se programa la salida de tensión de excitación, la cual puede ser variada dependiendo del código escrito en la dirección 80H del registro de control del integrado.

Para poder programar la Tensión y la ganancia del mismo, se debe tener en cuenta que esos bits de datos deben permanecer siempre fijos dentro de cualquier orden que vaya a la dirección 80H del integrado, teniendo en cuenta los siguientes valores de los bits D10, D9 y D8.

Tabla 11 Rangos de Salida y PGA.

D10	D9	
0	0	Range 1 (2.0 V p-p)
0	1	Range 4 (200 mV p-p)
1	0	Range 3 (400 mV p-p)
1	1	Range 2 (1.0 V p-p)
D8		PGA gain 0 = ×5, 1 = ×1

Fuente. Hoja de Datos Analog Devices AD5933/34 2005

Dentro de las funciones de **inicio ()** y la de **Standby ()** las cuales trabajan sobre la dirección 80H del integrado, se debe dejar fijo los valores de tensión de salida y la ganancia o PGA del integrado.

Impedancia. Para la toma de datos de impedancia, se hace el barrido de frecuencia y se almacena en un vector los valores de magnitud y de fase, después de un algoritmo matemático y una concatenación de las diferentes direcciones de registro y almacenamiento del AD5933.

Las Direcciones de almacenamiento del AD5933 son:

- 94H = MSB parte real.
- 95H =LSB parte real.
- 96H =MSB parte imaginaria.
- 97H =LSB parte imaginaria.

Por consiguiente se debe tener la cuenta que en dos direcciones diferentes se almacena parte de un solo valor y se debe pasar a una sola palabra, la cual se puede desarrollar de la siguiente manera:

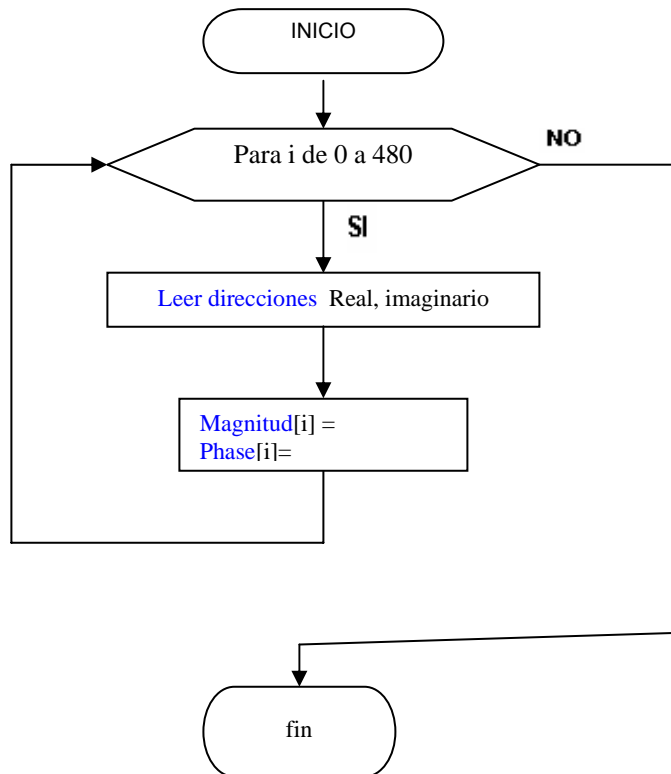
$$\text{reald} = \text{realdo} + (\text{realup} * 256)$$

Donde

Reald es la palabra completa.
Realdo es LSB
Realup es MSB

De igual manera se trabaja la parte imaginaria, a continuación se presenta el diagrama de flujo del almacenamiento de datos, bloque insertado en el diagrama de flujo del barrido de frecuencia.

Figura 20 Diagrama de Flujo para el Almacenamiento de Datos



Fuente. Autores.

2.2 DSP 56F8323

El primer pasó para poder establecer una comunicación de maestro (DSP) con diversos esclavos (LCD y AD5933), se refiere a la configuración de los GPIO para control del ASIC y de la pantalla LCD, ya que no se necesita de otra cosa más que de la programación del protocolo I2C.

2.2.1 Pines de propósito general (GPIO). Cómo se había mencionado en el primer capítulo, este módulo cuenta con 27 pines divididos en 3 puertos que se pueden configurar como GPIO. En la tabla 12 se muestra la configuración de los pines GPIO en el DSP 56F8323.

Tabla 12 Pines GPIO en el DSP 56F8323

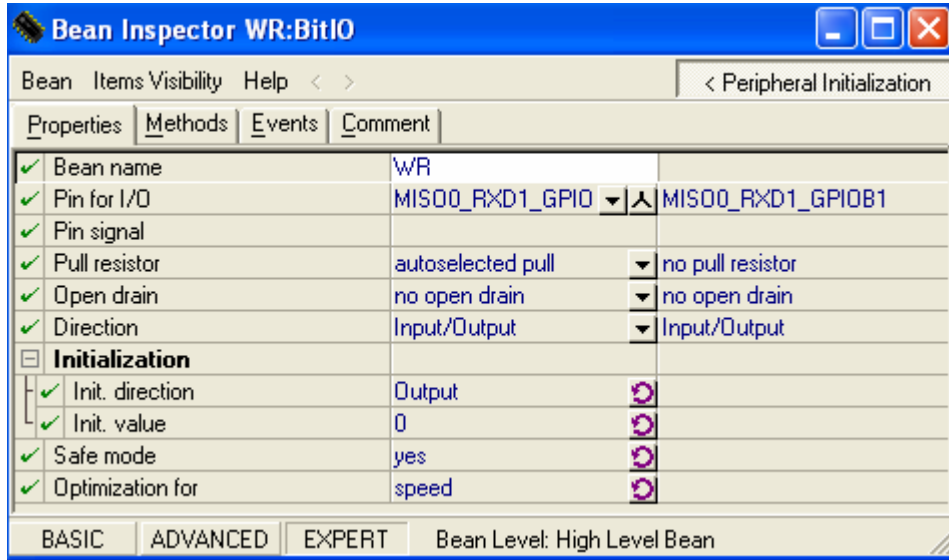
Pin GPIO	Función Periférica	Pin 56F8323
GPIOA0	PWMA0	3
GPIOA1	PWMA1	4
GPIOA2	PWMA2/SSI	7
GPIOA3	PWMA3/MISO1	8
GPIOA4	PWMA4/MOSI1	9
GPIOA5	PWMA5/SCLK1	10
GPIOA6	FaultA0	13
GPIOA7	FaultA1	14
GPIOA8	FaultA2	15
GPIOA9	ISA0	16
GPIOA10	ISA1	18
GPIOA11	ISA2	19
GPIOB0	SS0/TXD1	21
GPIOB1	MISO0/RXD1	22
GPIOB2	MOSI0	24
GPIOB3	SCLK0	25
GPIOB4	HOME0/TA3	49
GPIOB5	INDEX0/TA2	50
GPIOB6	PHASEB0/TA1	51
GPIOB7	PHASEA0/TA0	52
GPIOC0	EXTAL	46
GPIOC1	XTAL	47
GPIOC2	CAN_RX	61
GPIOC3	CAN_TX	62
GPIOC4	TC3	63
GPIOC5	TC1/RXD0	64
GPIOC6	TC0/TXD0	1

- **Configuración de GPIO**

Para configurar los pines de propósito general se utilizaron dos métodos. En primera instancia se configuraron escribiendo directamente a los registros de este módulo expuestos en las secciones anteriores, y para configurar aquellos pines con funciones independientes se utilizó el Bean BitIO, que se encuentra bajo la

categoría CPU Internal Peripherals/Port I/O. En la figura 24 se pueden apreciar las posibilidades de configuración que ofrece este Bean.

Figura 24 Registro Data Register (DR)



Fuente. MC56F8300 Peripheral User Manual, 2004.

Por medio de este Bean se puede seleccionar el pin del DSP que se quiere destinar para propósito general. Éste se puede configurar ya sea como salida o como entrada y además se puede seleccionar su valor inicial, aunque por supuesto éste valor se puede modificar dentro del programa utilizando alguno de los métodos destinados para este propósito, como lo son SetVal, PutVal y ClrVal.

Como se explicó anteriormente, los pines de propósito general también se pueden configurar escribiendo directamente a los registros de éste módulo.

La tabla 13 proporciona información sobre la configuración y disposición de los pines GPIO en el prototipo final, así como su función dentro de la aplicación.

Tabla 13 Configuración de los pines GPIO en el prototipo

GPIO	Pin 56F8323	Función
GPIOA0	3	LCD_D0
GPIOA1	4	LCD_D1
GPIOA2	7	LCD_D2
GPIOA3	8	LCD_D3
GPIOA4	9	LCD_D4
GPIOA5	10	LCD_D5
GPIOA6	13	LCD_D6

GPIOA7	14	LCD_D7
		LCD RES
GPIOB1	22	LCD WR
GPIOB2	24	LCD A0
GPIOB3	25	SCL
GPIOB0	21	SDA
		PULSADOR
		PULSADOR
		PULSADOR

Fuente.Autores.

2.3 PANTALLA HYUNDAI

Para la programación de la pantalla Hyundai se desarrollaron diferentes funciones, al igual que con la programación del AD5933 y dentro de las funciones tenemos:

Escribir_Comando () = Ejecuta un comando (orden).

Escribir_Dato () = Despliega en pantalla un valor o dato.

Inic_Lcd () = Usada para el encendido e inicialización de la pantalla.

Mostrarv () = Sirve para mostrar en pantalla el valor de un vector.

Mostrart () = Permite desplegar en pantalla un texto.

Conv_asc() = Permite convertir números a caracteres ASCII para la impresión.

Csr_Dir () = Usada para definir la dirección de los cursores.

Csr_Pos () = Usada para definir la posición del cursor dentro de la pantalla.

Clr_Texto () = Permite borrar un texto mostrado en pantalla.

Clr_Gráfico () = Usada para borrar gráficos presentados en pantalla.

Portada () = Esta función permite desplegar la presentación del proyecto, logo UIS, nombres, nombre del proyecto, etc.

Marcos () = Permite graficar marcos alrededor del logo UIS y de el borde de la pantalla.

MARCOS _BODE () = Marcos Para los diagramas de Bode.

GRAFICA_EJES () = Función que permite desplegar en pantalla los ejes X y Y.

GRAFICA_EJE_LOG() = Permite desplegar en forma grafica el eje logarítmico.

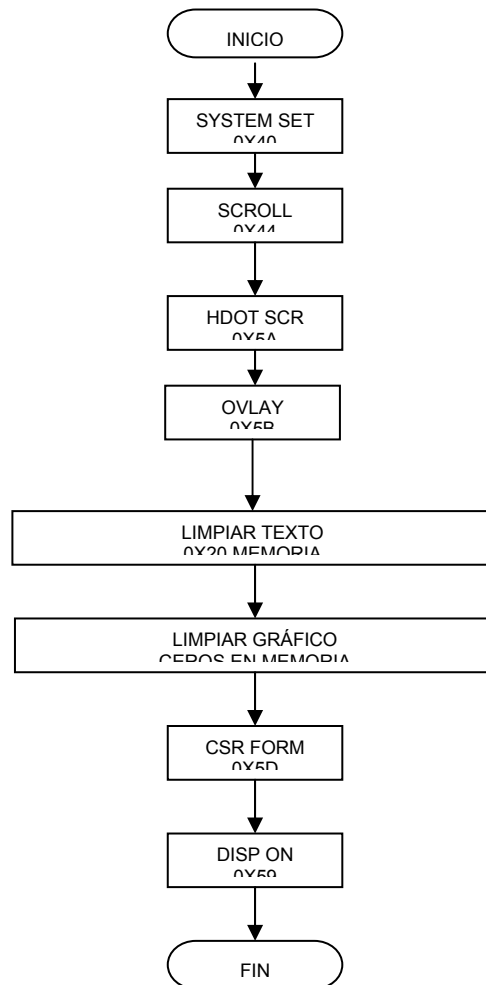
GRAF_BODE_MAG() = Función desarrollada para el despliegue del diagrama de Bode en magnitud.

GRAF_BODE_FAS () = Función que procesa los parámetros de la celda para el despliegue gráfico del diagrama de Bode en Fase

La inicialización de la pantalla gráfica LCD se realiza enviando al controlador de la pantalla los comandos y datos necesarios para configurarla y de esta forma determinar sus características de funcionamiento. En el capítulo 1 se presentaron las instrucciones necesarias para la inicialización de la pantalla así como su configuración final. En la figura 25 se muestra un diagrama de flujo en el que se enumeran los pasos que se deben seguir para inicializar la pantalla correctamente.

Es necesario aclarar que aquí sólo se presentan los comandos, más no los datos que se deben enviar después de cada comando para configurar la pantalla.

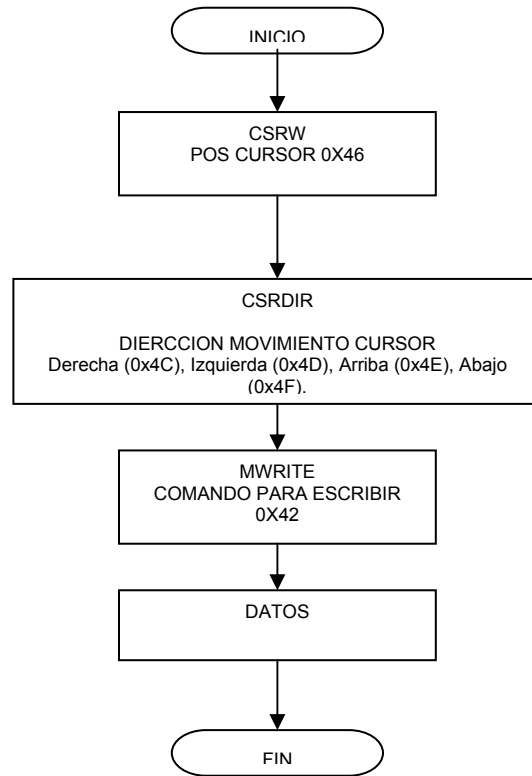
Figura 25 Diagrama de flujo inicialización LCD



Fuente. Autores.

Una vez se ha inicializado la pantalla se puede proceder a enviar datos a ella para su posterior visualización. La escritura de datos en la pantalla se hace mediante una serie de instrucciones que determinan la posición del cursor en la pantalla (capa de texto o gráfica), la dirección de movimiento del cursor (arriba, abajo, izquierda o derecha) y finalmente el comando de escritura seguido del código ASCII de los caracteres (capa de texto) o de los puntos del gráfico (capa de gráficos).

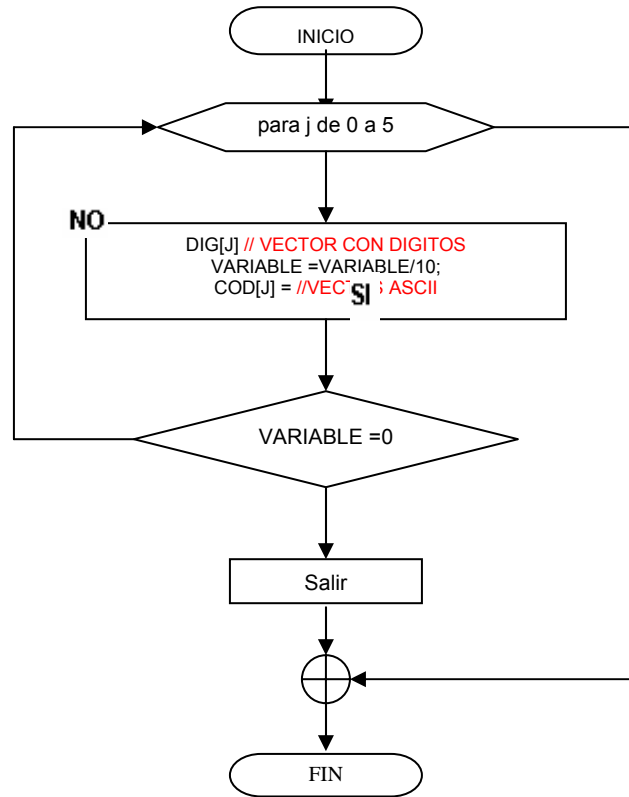
Figura 26 Diagrama de flujo escritura de datos en la LCD



Fuente. Autores.

En esta aplicación es necesario escribir en pantalla los valores de los parámetros calculados. Para este propósito es necesario identificar cada dígito del número y posteriormente enviar el código ASCII de este dígito a la pantalla. En el siguiente diagrama de flujo se puede apreciar el método utilizado para separar la variable en dígitos y posteriormente guardar su código ASCII correspondiente en un vector para su visualización. El método consiste en hacer una división por diez utilizando el operador % (residuo de la división entera). Este operador permite recuperar el último dígito de la variable. Posteriormente se hace una división normal por diez para ir descartando de la variable los dígitos ya almacenados. Para cada dígito del número se debe calcular el código ASCII; en este estándar los números tienen un código que va desde el 0x30 para el cero hasta el 0x39 para el nueve, entonces luego de obtener el dígito se debe sumar 0x30 para obtener su correspondiente código ASCII. La sentencia IF permite identificar si el número ya fue separado completamente en sus dígitos.

Figura 27 Diagrama de flujo código ASCII de variables



Fuente. Autores.

3. PRUEBAS

3.1. PANTALLA HYUNDAI

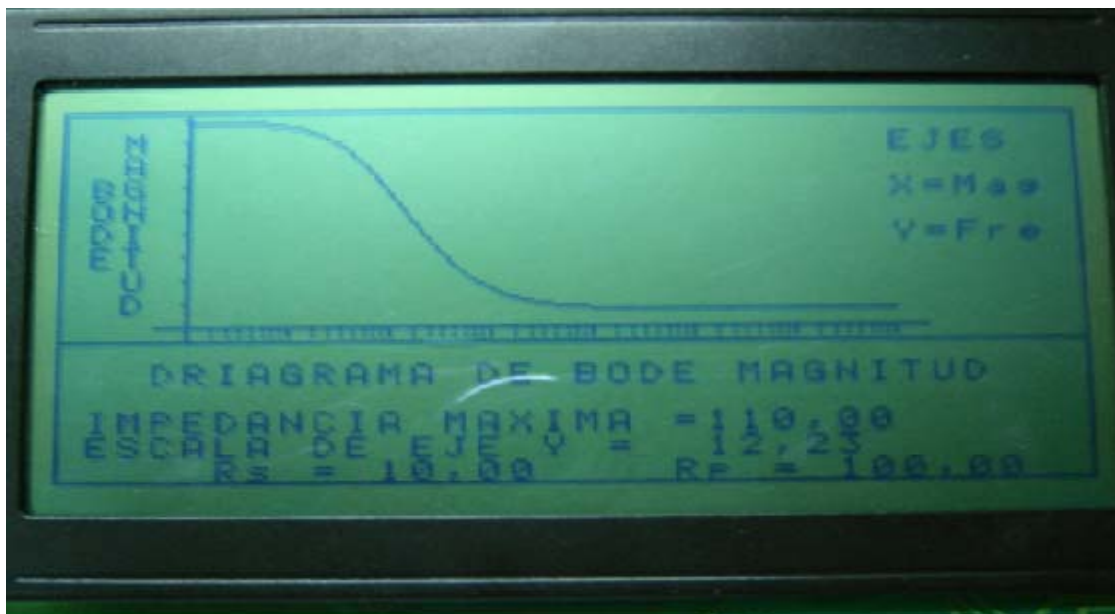
En esta sección se presenta una serie de ilustraciones, en las cuales se puede observar el funcionamiento de la pantalla, estas ilustraciones o fotografías muestran la presentación del proyecto en pantalla, el diagrama de Bode en magnitud de bode y el de fase.

Figura 28 Presentación de proyecto



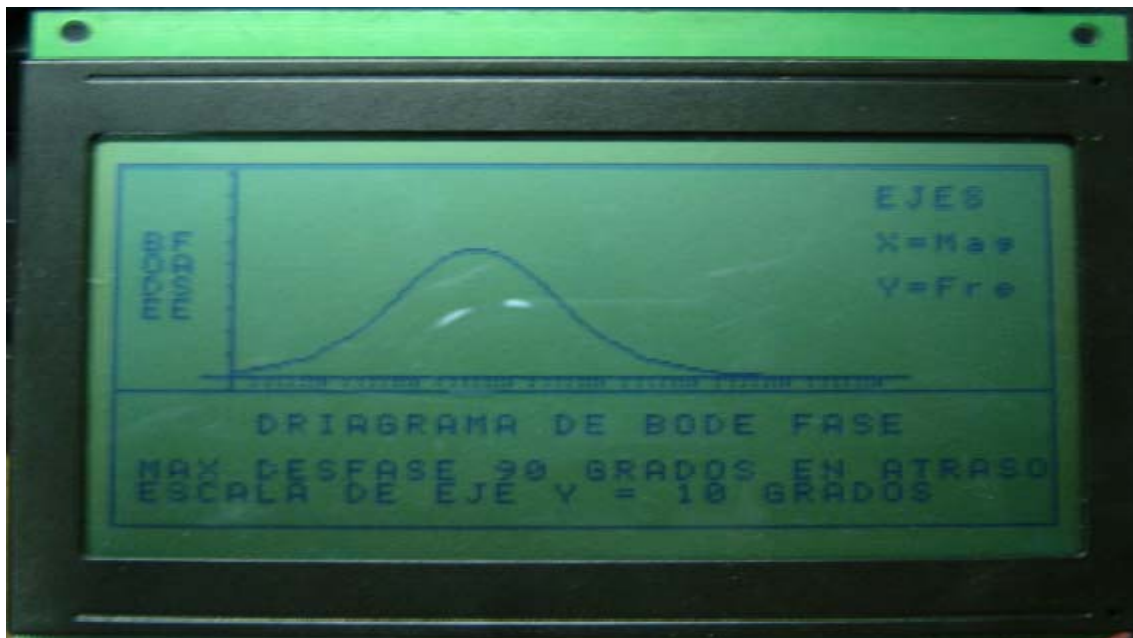
Fuente Los Autores

Figura 29 Diagrama de Bode Magnitud Vs Frecuencia



Fuente Los Autores

Figura 30 Diagrama de Bode Fase Vs Frecuencia



Fuente Los Autores

3.2 AD5933

Para poder realizar las pruebas del integrado, se debe tener en cuenta los puntos de calibración y los valores de programación del inicio de frecuencia, el número de incrementos, el paso, la tensión de salida, el PGA, la ganancia y el número de ciclos que la señal debe hacer para realizar la medición.

A continuación se presenta una tabla con el desarrollo de los valores necesarios para programar los parámetros antes mencionados:

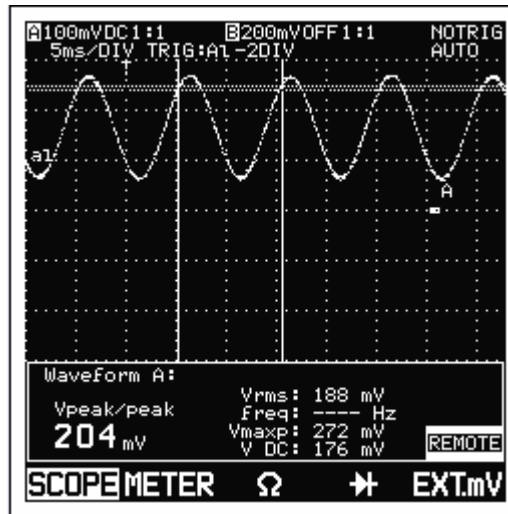
Tabla 14 Valores de calibración

PARAMETROS	ECUACION	VALOR DEC	VALOR HEX
Start frequency 100Hz	$Sfrequency = \left[\frac{100Hz}{\frac{16.776Mhz}{4}} \right] \times 2^{27}$	3200	000C80
Incremento de Frecuencia 200Hz	$FIncrement = \left[\frac{200Hz}{\frac{16.776Mhz}{4}} \right] \times 2^{27}$	6400	001900
Number increments 480		480	01E0
Ciclos 150		150	96
Tensión de salida		200 mVpp	
PGA		X1	
Rfb			

Fuente. Autores.

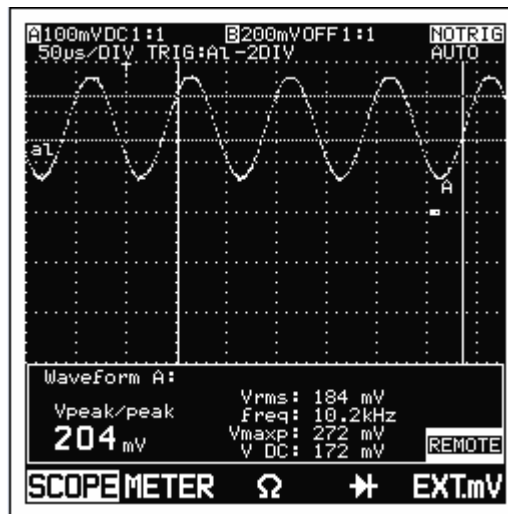
Después de realizar la programación del AD5933, se realiza el barrido en frecuencia, el cual es presentado en las figuras 32 -37.

Figura 31 Tensión de Inicial de Salida



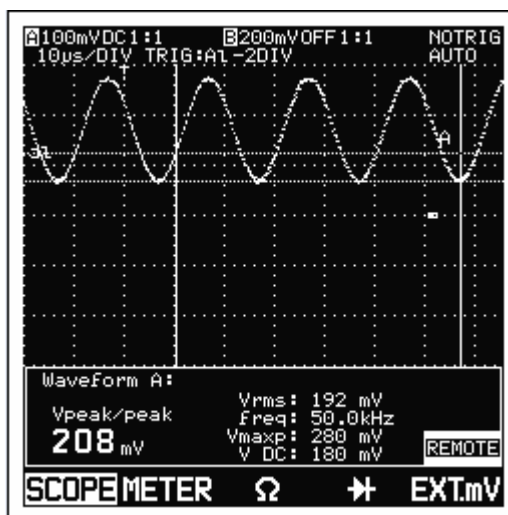
Fuente. Autores.

Figura 32 Tensión de Salida a una frecuencia de 10 KHz



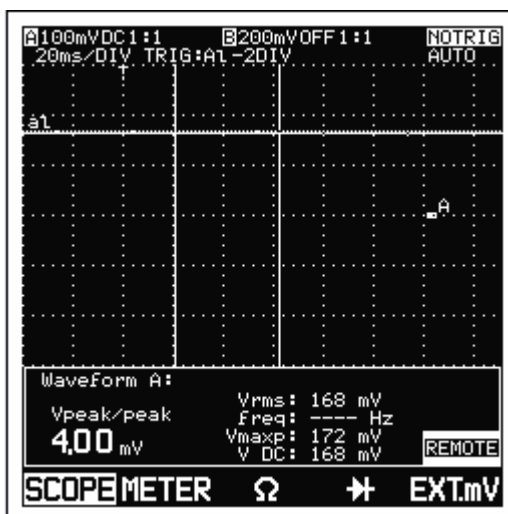
Fuente. Autores

Figura 33 Tensión de Salida a una frecuencia de 50 Khz.



Fuente. Autores.

Figura 34. Reset



Fuente. Autores.

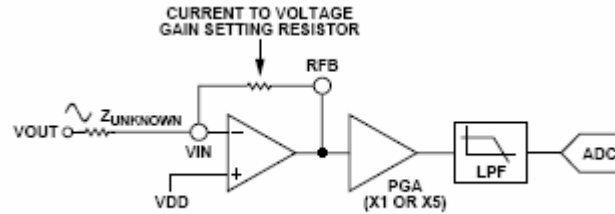
En la tabla 15 se presenta una comparación donde se muestran resultados y análisis de error de medición al tomar datos con el medidor, un multímetro y la tarjeta de evaluación del AD5933.

Para poder realizar la toma de datos, es necesario tener en cuenta el rango de impedancias a medir, debido a que se necesita calibrar el medidor para que este trabaje en una región lineal, para esto es necesario calcular un factor de ganancia,

determinar un valor de PGA y una resistencia de realimentación, lo anterior se resume en la ecuación 8 donde se presenta la relación de estos parámetros.

$$TensionSalida \times \frac{RFB}{Impedancia} \times PGA \quad [8]$$

Figura 7 AD5933 Etapa de recepción



Fuente: Analog Devices Datasheet AD5933/34 2005

Tabla 15 Resultados de mediciones

Impedancia		Medidor		%Error	
Z _{mag}	Z _{fase}	Z _{mag}	Z _{fase}		
1K	0 ^a	1010	0	1	
5.1K	0	4934	0	3.2	
7.5K	0	7614	0	1.5	
2.4K	0	2398	0	0.0833	
10K	0	10898	-0.84	8.98	
1nF	-90	0.9nF	-82.016	10	
2nF	-90	1.9nF	-78.67	5	
CELDAS DE PRUEBA					
Reales			Experimental		
R_p	R_s	C_d	R_p	R_s	C_d
5.1K	1K	1nF	2313.28	3791.23	1.2nF
Celda 1			Rbaja 6104.51		Ralta 2313.28
R_p	R_s	C_d	R_p	R_s	C_d

1K	5.1K	10nF	999.98	5110.71	9nF
Celda 2			Rbaja 6104.59	Ralta 5110.71	
Rp	Rs	Cd	Rp	Rs	Cd
1K	7.5K	10nF	945.42	8116.68	N.C
Celda 3			Rbaja 9056	Ralta 8116,68	
Rp	Rs	Cd	Rp	Rs	Cd
1K	2.4K	10nF	700	2.5	13nF
Celda 4			Rbaja 3152.3	Ralta 2520.2	
Paralelo					
Reales			Experimental		
R	C			R	C
7.5K	1nF			7.15 K	1.09nF
5.1K	1nF			4.628K	1.2nF

Fuente Autores

Para determinar los valores de impedancia, es necesario el cálculo de un factor de ganancia, el cual es hallado para diferentes rangos de impedancia; adicionalmente se debe calibrar la fase del sistema, para esto se debe conectar una resistencia de calibración y observar cual es la respuesta para poder restarlo a las siguientes impedancias medidas, ver tabla 16.

Tabla 16 Factor de Ganancia y respuesta de la Fase

Rango	RFB	Fase sistema	Factor de Ganancia
0.1-1K Ω	100 Ω		
1-10K Ω	1K Ω	116.472	1.1007209 exp-7
10-100K Ω	10K Ω	N.C	1.057200306exp-9
100K Ω -1M Ω	100K Ω	N.C	1.057200306exp-9
1-2M Ω	100K Ω	N.C	5.852274632exp-11
9-10M Ω	9M Ω	N.C	5.852274632exp-11

Fuente Autores

3.2.1 Análisis de Resultados. Se debe tener en cuenta que al calibrar el sistema, la resistencia de realimentación debe ser igual a la resistencia de medición conectada entre Vin y Vout, además el AD5933 posee una resistencia de salida , para el caso en particular, la resistencia de salida es de 200 Ohms y esta debe ser sumada a impedancias diferentes de 1 K Ω , ya que este es el mínimo valor de realimentación y por consiguiente aumenta el error en la medición. Ver tabla 17.

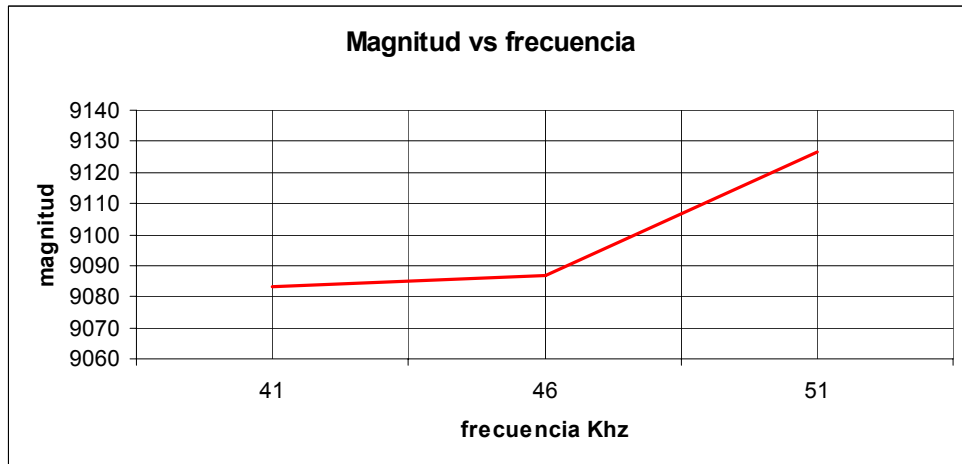
Tabla 17 Calibración con una resistencia de 1K Ω .

Frecuencia	Magnitud
1	21168,7
6	9009,73
11	9053,67
16	9126,38
21	9227,2
26	9108,38
31	9024,67
36	9115,72
41	9083,11
46	9086,8
51	9126,75
56	9054,12
61	8951,47
66	8921,93
71	9087,03
76	8922,02
81	8951,61
86	88844
91	8910,99
96	8862,71
101	8851,56

Fuente. Autores.

Después de hallar la respuesta del sistema, se procede a calcular el valor de Factor de Ganancia, escogiendo la zona mas lineal y al centro de la frecuencia, para esto se escogieron 3 posibles regiones, las cuales son frecuencias a 41KHz, 46KHz y 56KHz, sin embargo se pudo observar que la región comprendida entre 41-46 KHz aunque presentaba la curva mas plana no permitía una buena respuesta al momento de calcular la impedancia. Ver figura 35

Figura 35 Región para el cálculo del Factor de Ganancia FG



Fuente. Autores

El Factor de Ganancia depende de los valores hallados en la magnitud y se calcula con la ecuación 15.

$$FG = \frac{1}{\frac{\text{Im pedancia}}{\text{Magnitud}}} \quad [7]$$

$$FG (41K) = 1.1009445 \exp-7$$

$$FG (46K) = 1.1004974 \exp-7$$

$$FG (51K) = 1.09568028 \exp-7$$

Después de calcular el factor de Ganancia para cada punto se toma un promedio de la región escogida, obteniendo como resultado un nuevo factor de ganancia el cual representa menor error en la medición.

$$FG (41-46) = 1.1007209 \exp-7$$

$$FG (46-51) = 1.09808884 \exp-7 \quad (\text{Factor escogido para los cálculos de impedancia})$$

Para completar la Calibración del sistema, se procede al cálculo de la fase debido a que este responde a una serie de polos internos, si conectamos una impedancia netamente resistiva aparece una fase que varía con el aumento de la frecuencia, la cual debe ser restada en cada punto para que esta sea cero. En la tabla 18 se presenta la medición de esta fase por dos resistores y un capacitor, observando así el desfase de 90 grados entre el resistor y el capacitor.

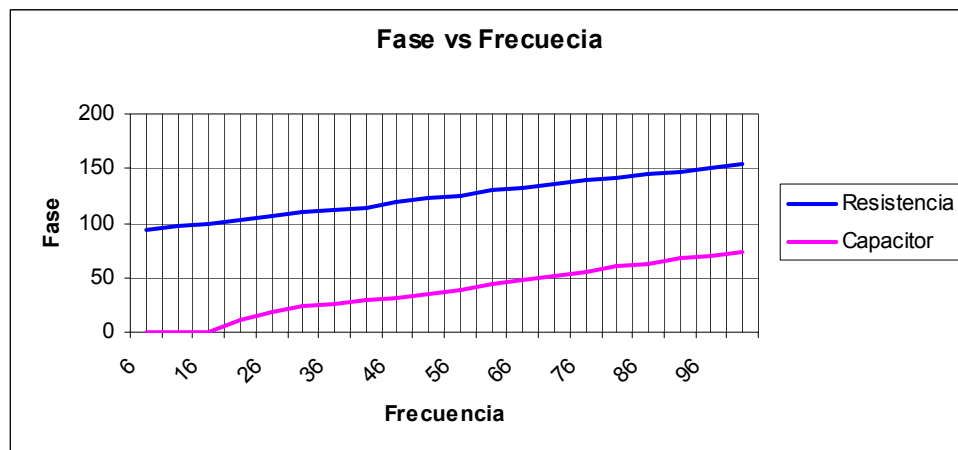
Tabla 18 Respuesta de fase del sistema ante la medición

Frecuencia Khz	Capacitor	Resistencia 1K	Resistencia 10K
6	0	93,2045	90,9996
11	0	96,7526	90,9996
16	0	99,4504	90,9996
21	11,8411	103,206	90,9996
26	18,6909	106,653	104,508
31	23,1981	109,701	104,508
36	26,3784	111,071	104,508
41	28,738	114,583	104,508
46	30,5561	118,361	123,735
51	34,4285	122,236	123,735
56	38,6343	124,658	123,735
61	44,3789	129,558	123,735
66	46,9462	132,002	123,735
71	51,5264	135,626	123,735
76	55,4377	139,173	135,659
81	59,798	141,484	146,862
86	62,6838	145,046	146,862
91	67,3263	147,274	146,862
96	70,3041	150,805	146,862
101	73,4621	154,328	146,862

Fuente Autores

Para observar mejor el desfase de 90 grados entre el resistor y la capacitancia, se puede referir a la figura 36 y analizar el comportamiento de la fase ante la frecuencia.

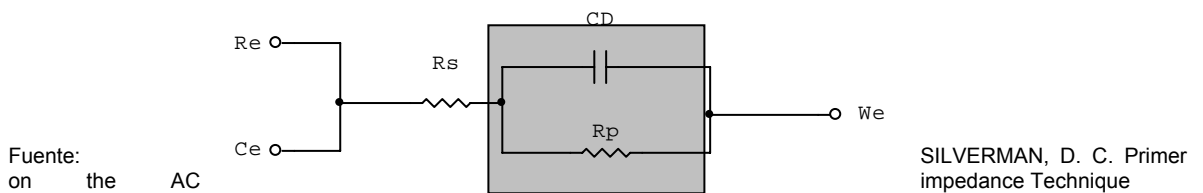
Figura 36 Respuesta de Fase del sistema ante la medición de una impedancia



En la grafica anterior se puede observar que la toma de datos no es iniciada desde ala frecuencia de inicio del barrido, debido a que el primer valor de fase o magnitud no es confiable , ya que el oscilador interno tiende a estabilizarse, por consiguiente ese primer valor debe ser obviado.

3.2.1.1 Análisis de datos para celda de medición. Para realizar mediciones de forma correcta en las celdas, se debe determinar para cuales valores la medición este dentro de los rangos de medición permisibles, para esto es necesario observar el efecto capacitivo de a bajas frecuencias (circuito abierto) y a altas frecuencias (corto circuito), para esto se estudia el paralelo de la celda ya que es aquí donde este efecto se observa, en la figura 2 se puede observar en la parte sombreada al paralelo.

Figura 2 Celda de prueba



El primer paso a seguir es determinar varios valores de capacitores, hallar el valor de su magnitud a diferentes frecuencias (6, 46, 96 Khz.) y compararlo con el efecto de la resistencia en paralelo, para así conocer cual es el valor de resistor que debe estar en paralelo.

Tabla 19 Capacitores seleccionados para pruebas

Capacitor	Valor
C1	1 nF
C2	10 nF
C3	3,3 uF
C4	2,2 uF
C5	100 nF
C6	0,5 nF

Fuente Autores

Tabla 20 Magnitud de los capacitores

Frecuencias	Magnitud					
	C1	C2	C3	C4	C5	C6
6	26525,8	2652,8	8,0381	12,057	265,258	53051,6
46	3459,9	345,98	1,05	1,6	34,59	6919
96	1657,86	165,786	0,5	0,75	16,5786	3315,7

Fuente Autores

Después de calcular los valores de magnitud de los capacitores, se procede a hallar el valor de resistencia que debe ir en paralelo para que el efecto capacitivo se pueda observar a cabalidad.

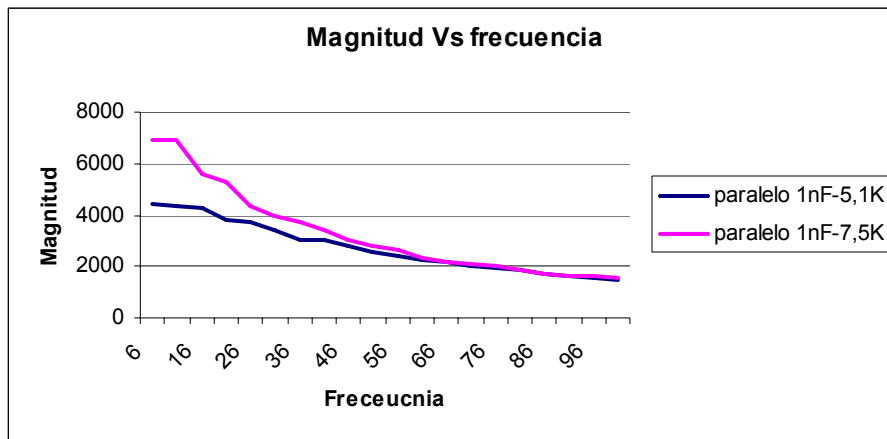
Tabla 21 Valor de la Resistencias en paralelo

Rp C1	Rp C2	Rp C3	Rp C4	Rp C5	Rp C6
5.1K	1K	NC	NC	100	7,5K
7,5K	1,2K	NC	NC		5,1K

Fuente Autores

Después de Calcular los posibles valores de resistencias y capacitancias, se procede a realizar un análisis de magnitud vs. frecuencia y observar el comportamiento de un arreglo, Figura 37.

Figura 37 Respuesta sistema Magnitud vs. Frecuencia



4. CONCLUSIONES

Se logró de manera satisfactoria construir un prototipo que permite realizar la excitación con una señal de tensión senoidal con frecuencia variable, medición de la impedancia y conversión análoga digital de la respuesta, procesamiento de la misma y despliegue gráfico en un solo equipo.

El equipo desarrollado permite a partir de la comunicación entre el AD5933 y el DSP MC56F8323 obtener los parámetros que permitan mediante un procesamiento matemático realizar de manera satisfactoria la medición de los elementos de una celda Dummy, para luego lograr el despliegue gráfico de los mismos en la pantalla Hyundai.

La flexibilidad que brinda el AD5933 de variar los parámetros para la medición garantiza la eficiencia y eficacia, ya que permite cambiar a 4 tipos de niveles de tensión, además la facilidad de establecer la frecuencia de inicio y la cantidad de pasos de frecuencia en la medición, dependiendo de la necesidad se pueden modificar las características de la señal de excitación : el software se desarrollo en la herramienta Codewarrior estableciendo como protocolo de comunicación I2C.

El Processor Expert de codewarrior, permite añadir Beans y así configurar los perifericos para aplicaciones especificas, gracias a su entorno gráfico se logra de manera amable obtener el código para establecer la comunicación con el ASIC mediante el protocolo I2C y la asignación de los pines para el reloj (SCL) y otro para datos (SDA).

La pantalla grafica permite al prototipo ser un dispositivo mas completo ya que no se necesita de ningún dispositivo extra para despliegue gráfico de los resultados, tanto los valores numéricos como los comentarios de los mismos. Además de permitir análisis en frecuencia del comportamiento de la impedancia mediante el diagrama de Bode tanto en magnitud como en fase.

Para diferentes rangos de impedancias se deben utilizar diferentes valores de RFB, para así aumentar la corriente.

5. RECOMENDACIONES

Es recomendable realizar un interfaz con el computador para poder descargar los históricos, desde la memoria del DSP, para poder así realizar informes de manera más eficiente.

Es recomendable colocar un mux para configurar los distintos rangos de frecuencia y las tensiones de salida, esto con el fin de realizar de manera correcta la medición de cualquier impedancia que este en el rango del dispositivo para explotar al máximo el funcionamiento del AD5933, además de comparar los errores en el rango de medición estipulado.

Usar un oscilador externo menor al interno para así realizar un barrido en los rangos más pequeños de frecuencia.

Realizar código que permita procesar los parámetros de la celda para el despliegue el diagrama de Nyquist para un mejor análisis de la respuesta en frecuencia de los resultados.

BIBLIOGRAFÍA

[DSP56F800FM, 03] Motorola. "DSP56800 Family Manual". 2003.

[DSP56F8323, 03] Motorola. "DSP568323 User`s Manual". 2003.

[DSP56F8323/D, 03] Motorola. "DSP56800 Technical Data Sheet". 2003.

[Smith, 99] Smith, Steven. "The Scientist and Engineer`s Guide to Digital Signal Processing". Segunda edición, California Technical Publishing, 1999.

[Sedra & Smith, 98] Sedra, Adel y Smith, Kenneth. "Circuitos Microelectrónicos". Cuarta Edición, Oxford University Press, Inc., U.S.A 1998.

[G3-89] AMERICAN SOCIETY FOR TESTING AND MATERIALS. Standard Practice for Conventions Aplicable to Electrochemical Measurements in Corrosion Testing. ASTM, 1989, 10p.: il (G3-89)

[G15-99b] _____ Standard Terminology Relating to Corrosion and Corrosion Testing. ASTM, 1999, 5p.: (G15-99b)

[G106-89] _____ Standard Practice for Verification of Algorithm and Equipment for Electrochemical Impedance Measurement. ASTM, 1989, 11p.: il (G106-89).

ANALOG DEVICES INC. Datasheet. AD5933 [Online]. 2005. www.analog.com

Anexo A. Entorno Codewarrior

“Metrowerks” y Motorola han dejado disponible una nueva versión especial del entorno de desarrollo para la familia de “DSP56800/E Hybrid Controllers: CodeWarrior Development Studio for Motorola 56F800 v.8.0” es una interfaz tipo Windows que posee menús, barras de herramientas que ayudan a construir, revisar y probar aplicaciones en tiempo real, la cual es gratuita y puede compilar hasta 8kbytes de código C. Con lo cual se cuenta con una máquina virtual que permite simular la CPU, periféricos e interrupciones de toda la familia de DSP56800 actuales de Motorola, lo que facilita el proceso de depuración de las aplicaciones desarrolladas en lenguaje C.

A continuación se procederá a realizar una introducción rápida para crear proyectos en lenguaje C, utilizando el compilador “Metrowerks CodeWarrior V8.0”.

Antes de empezar la creación de proyectos se debe escribir que los parámetros básicos que necesita este software para la realización de un programa adecuado, codewarrior necesita un “DLL” (Dynamic Link Libraries) para iniciar el puerto paralelo, ya que trabaja con el puerto de emulación “JTAG” (Joint Test Action Group), este “DLL” se adquiere de la pagina www.motorola.com y tener conectada las tarjetas del DSP para poder trabajar en codewarrior. Solo si se va a trabajar en modo Simulador no se necesita tener conectada la tarjeta del Puerto JTAG y la tarjeta del DSP MC56F8323 de motorola.

El primer paso es crear el proyecto. Para esto se selecciona **File | New...** Con esto aparecerá la ventana que se muestra en la figura A1. Escoja ‘**Processor expert stationery**’, la carpeta donde va a ubicar su proyecto y el nombre del proyecto.

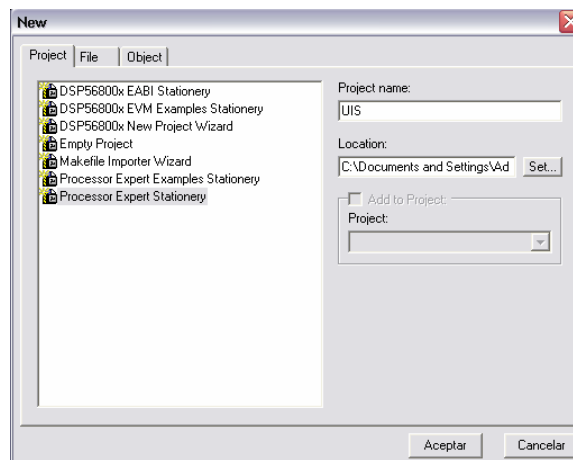


Figura A 1 Crear proyecto nuevo

Cuando presione Aceptar, aparecerá otra ventana para escoger el tipo de la familia DSP, y el DSP correspondiente el cual para este caso es MC56F8323 Tal como aparece en la figura A2.

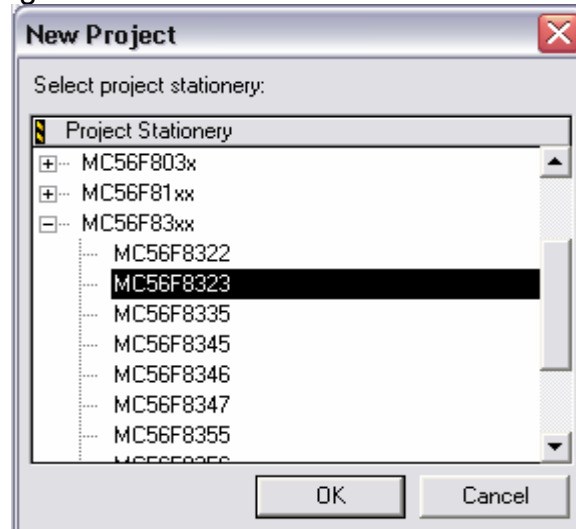


Figura A 2 escoger el DSP

Cuando presione "ok" Codewarrior creará una carpeta con el nombre que le dio a su proyecto y creará un proyecto de ejemplo en el lenguaje que escogió, en este caso lenguaje C. La figura A3 muestra la ventana principal de Metrowers Codewarrior.

En ella se aprecia que aparece una ventana que contiene el nombre del proyecto con la extensión mcp, en la parte superior derecha aparece la ventana Target CPU la cual permite ver el DSP escogido que para este caso es el dsp58f8323, en esta ventana se puede ver la configuración de pines del DSP. Por ultimo esta la ventana de "Bean Selector" que permite a un usuario seleccionar un Bean deseado y agregarlo al proyecto.

Cabe decir que un "Bean" es un componente que puede usarse en el Procesador Experto, el cual encapsula la funcionalidad de elementos básicos incluidos en sistemas (como CPU core, CPU on-chip, componentes periféricos ordenados independientemente, dispositivos virtuales, y los algoritmos del software puros) y proporciona el mando de estos dispositivos vía propiedades, métodos, y eventos.

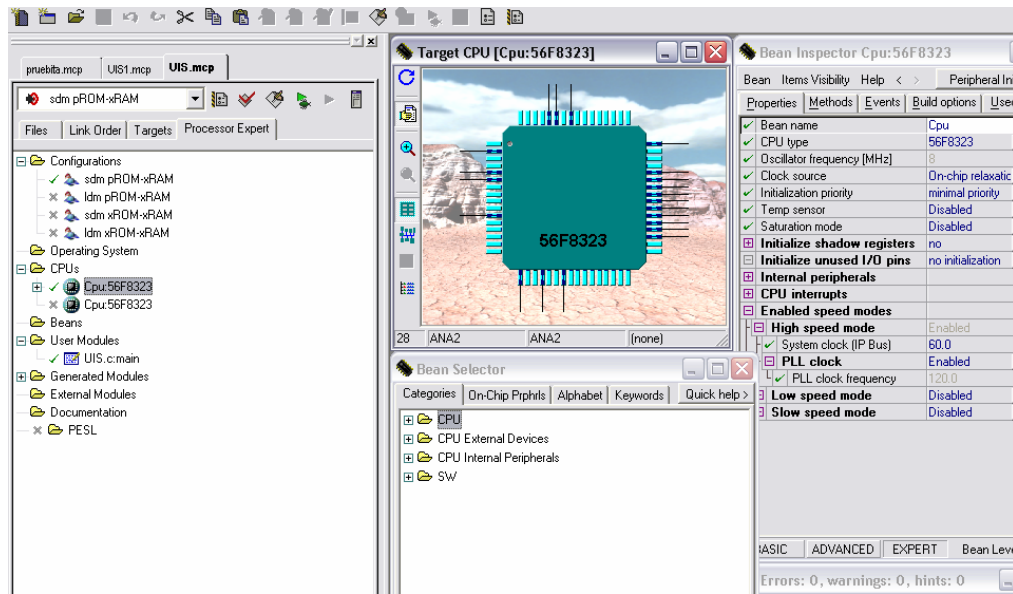



Figura A 3 Ventana principal utilizando Procesador Experto

A Continuación se tiene que ejecutar el comando **Make**  ubicado en la parte superior de la ventana principal de "codewarrior" el cual crea varias carpetas, entre ellas la carpeta donde se creara el proyecto, como se aprecia en la figura A4. "Make" también se encarga de encontrar errores de programación y de posibles advertencias que existen en el programa que se este ejecutando.

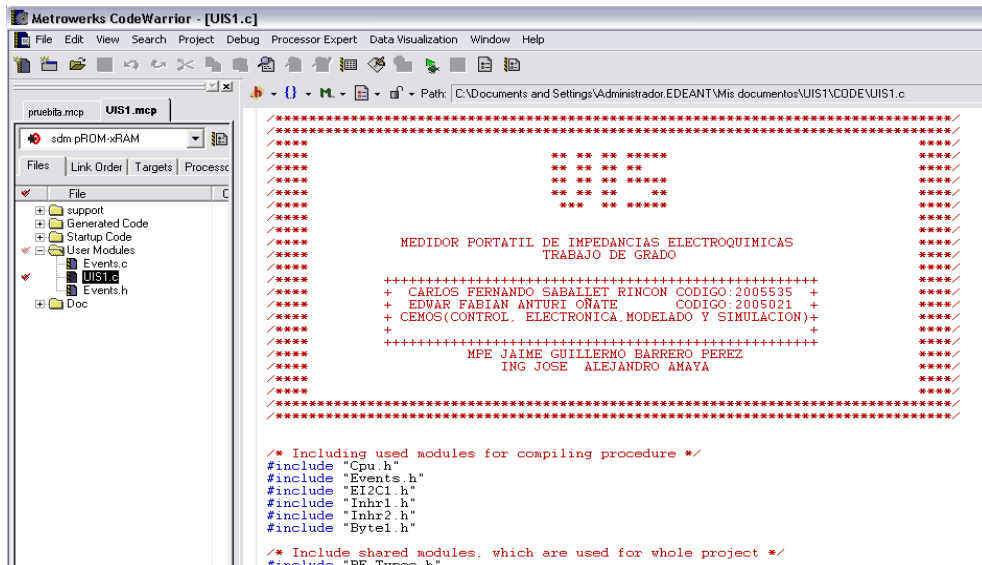


Figura A 4 Archivo generado para empezar el proyecto

En la carpeta **User Modules** se crea por defecto el archivo con el nombre que se asigno al proyecto, donde esta la función principal "main" con un esquema

para editar programas en C, bajo procesador experto. Tal como aparece en la figura A6.

```

/* Including used modules for compiling procedure */
#include "Cpu.h"
#include "Events.h"
#include "E12C1.h"
#include "Inhr1.h"
#include "Inhr2.h"
#include "Byte1.h"


/* Include shared modules, which are used for whole project */
#include "PE_Types.h"
#include "PE_Error.h"
#include "PE_Const.h"
#include "IO_Map.h"
#include <stdio.h>
#include <math.h>
#include <stdlib.h>

//CONSTANTES ALMACENADAS EN RAM
#define Alto 1
#define Bajo 0
#define Derecha 0x4C
#define Izquierda 0x4D
#define Arriba 0x4E
#define Abajo 0x4F
#define P_Grafico 0x1000
#define pi 3.14159265
#define TOLER 0.0001

//CONSTANTES ALMACENADAS EN MEMORIA FLASH
const byte UNIV[]={"UNIVERSIDAD INDUSTRIAL DE SANTANDER"};
const byte ESEE[]={"ESCUELA DE INGENIERIA ELECTRICA, ELECTRONICA Y TELECOMUNICACIONES"};
const byte DISE[]={"DISEÑO DE UN PROTOTIPO AUTONOMO BASADO EN DSP Y UN ASIC."};
const byte DUMM[]={"QUE PERMITA MEDIR LA IMPEDANCIA DE UNA CELDA DUMMY"};
const byte CARB[]={"CARLOS FERNANDO SARATTE" "ETCON"};

```

Figura A 5 Archivo en Lenguaje de programación en C.

Después de haber editado el programa se vuelve a ejecutar "Make" para ver los posibles errores y advertencias .Paso seguido se hace un "Debug" , este comando ubicado esta en la parte superior de la ventana principal de codewarrior, el cual hace que el programa este en modo "Run" y para asi simular el DSP o ya sea ver los resultados en el software de "codewarrior IDE".

"Codewarrior" tiene la posibilidad de mostrar el tamaño en bytes del programa generado, tal como se puede apreciar en la figura A7 la cual indica el tamaño en bytes del programa realizado, esto aparece inmediatamente después de ejecutar el comando debug.

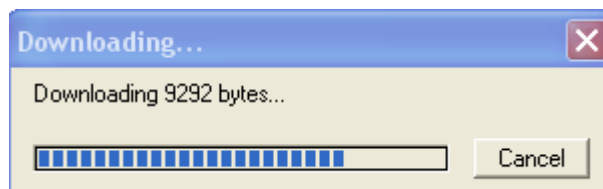


Figura A 6 Tamaño en bytes del Programa

En la figura A7, se muestra la ventana de simulación de CodeWarrior. En ella se pueden observar el contenido de los registros, memoria, variables globales y locales y el código fuente en C y en ensamblador. Finalmente, solo queda grabar el programa en el dsp. Por defecto el puntero queda siempre en la inicialización interna del Procesador experto, la cual no se debe remover. Según se quiere el programa se puede ejecutar en diferentes formas, tal y como se explica a continuación.

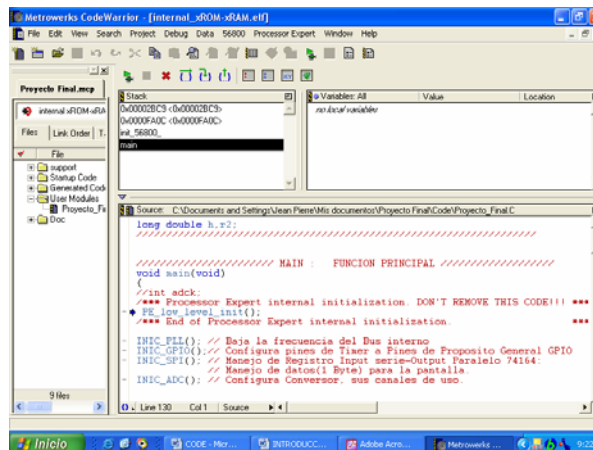






Figura A 7 Panel frontal en Modo Run


Entre los comandos utilizados para el proyecto tenemos:


El comando "RUN"  se encarga de enviar el programa al DSP directamente.

El comando "Kill Tread"  termina la sesión de ejecución del programa completamente, llevándolo al programa original.

El comando "Step Over"  permite literalmente recorrer el código fuente, ejecutando cada declaración hecha en el programa.

El comando "Step into"  realiza un recorrido similar al comando Step Over pero con la diferencia que este entra a la declaración y la realiza paso a paso.

El comando "Step Out"  se encarga de ejecutar el resto de las rutinas en que valla el programa y se detiene la ejecución cuando termina de ejecutar todas las rutinas.

El comando "breakpoints"  es uno de los comandos mas importantes para saber si un programa realiza lo que tiene que hacer el coloca un punto en el que el programa se detiene para permitir detectar y eliminar errores.

Finalmente, la figura de abajo muestra el momento en que el programa se esta ejecutando y simulando al DSP.

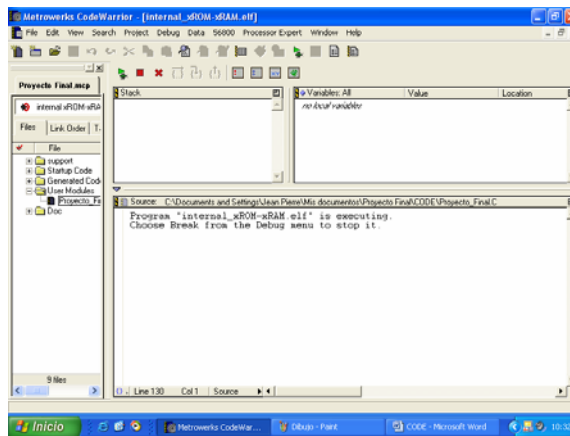


Figura A 8 Ventana de Ejecución