

RAE

1. TIPO DE DOCUMENTO: Trabajo de grado para optar por el título de INGENIERO DE SONIDO.

2. TÍTULO: DESARROLLO DE UN SISTEMA DE ADQUISICIÓN DEL NIVEL DE PRESIÓN SONORA Y VARIABLES CLIMÁTICAS UTILIZANDO SISTEMAS EMBEBIDOS.

3. AUTORES: Nicolás Andrés Arias Sabogal.

4. LUGAR: Bogotá, D.C

5. FECHA: Marzo 2014.

6. PALABRAS CLAVE: Estación meteorológica, sonómetro, sistemas embebidos, arduino, Visual Basic, comunicación serial.

7. DESCRIPCIÓN DEL TRABAJO: El objetivo del trabajo es desarrollar un sistema de adquisición y visualización del nivel de presión sonora y variables climáticas, haciendo uso de dispositivos digitales unificados, como los sistemas embebidos.

8. LÍNEAS DE INVESTIGACION: Línea de investigación: análisis y procesamiento de señales, Núcleo Problemático: Acústica y audio.

9. METODOLOGÍA: la metodología propuesta y desarrollada en el proyecto se basa en un enfoque empírico – analítico.

10. CONCLUSIONES: El documento presenta el desarrollo de un sistema de adquisición y visualización del nivel de presión sonora y variables ambientales, implementado en el sistema embebido Mini210s. En el S.E se implementó el sistema operativo Windows CE6, el cual es compatible con la interfaz desarrollada en visual basic, y con el protocolo de comunicación serial.

La transferencia de tecnología y conocimiento es importante en la innovación y creación de nuevos procesos, gracias a las ayudas técnicas que se brindan de manera libre, al momento de realizar desarrollos ingenieriles y tecnológicos que a su vez sirven como guía para nuevos proyectos.

DESARROLLO DE UN SISTEMA DE ADQUISICIÓN DEL NIVEL DE
PRESIÓN SONORA Y VARIABLES CLIMÁTICAS UTILIZANDO
SISTEMAS EMBEBIDOS.

NICOLAS ANDRÉS ARIAS SABOGAL

UNIVERSIDAD DE SAN BUENAVENTURA
FACULTAD DE INGENIERÍA
INGENIERÍA DE SONIDO
BOGOTÁ
2014

DESARROLLO DE UN SISTEMA DE ADQUISICIÓN DEL NIVEL DE
PRESIÓN SONORA Y VARIABLES CLIMÁTICAS UTILIZANDO
SISTEMAS EMBEBIDOS.

NICOLÁS ANDRÉS ARIAS SABOGAL

Trabajo de grado para optar el título de Ingeniero de Sonido

Director

Ing. NELSON FELIPE ROSAS JIMÉNEZ., MSc.

UNIVERSIDAD DE SAN BUENAVENTURA

FACULTAD DE INGENIERÍA

INGENIERÍA DE SONIDO

BOGOTÁ

2014

AGRADECIMIENTOS.

Al ingeniero y director de tesis Nelson Rosas, por la confianza y el apoyo incondicional en la orientación de este proyecto.

A mis compañeros que estuvieron conmigo en todo este proceso de aprendizaje y de amistad.

A mis padres por brindarme esta oportunidad y por su comprensión.

TABLA DE CONTENIDO.

1. PLANTEAMIENTO DEL PROBLEMA.	14
1.1. ANTECEDENTES.	14
1.2. DESCRIPCIÓN DEL PROBLEMA.	21
1.3. JUSTIFICACIÓN.	22
1.4. OBJETIVOS.	23
1.4.1. Objetivo General.	23
1.4.2. Objetivos Específicos.	23
1.5. ALCANCES Y LIMITACIONES.	23
1.5.1. Alcances.	23
1.5.2. Limitaciones.	24
2. METODOLOGIA.	24
2.1. ENFOQUE DE LA INVESTIGACIÓN.	24
2.2. HIPÓTESIS.	24
2.3. VARIABLES	24
2.3.1. Variables Independientes.	24
2.3.2. Variables Dependientes.	25
3. LÍNEA DE INVESTIGACIÓN DE LA UNIVERSIDAD / SUB-LÍNEA DE LA FACULTAD / CAMPO TEMATICO DEL PROGRAMA.	25
4. MARCO TEORICO.	25
4.1. Estaciones de meteorología.	25
4.1.1. Sensor de temperatura.	25
4.1.2. Sensor de humedad relativa.	26
4.1.3. Anemómetro.	27
4.1.4. Veleta de viento.	28
4.2. Sonómetro.	28
4.2.1. Tiempo de integración.	30
4.2.2. Nivel de presión sonora.	30
4.2.3. Nivel continuo equivalente.	31
4.3. Micrófonos.	31
4.3.1. Micrófonos dinámicos.	32
4.3.2. Micrófonos Capacitivos.	32
4.3.3. Pre amplificación.	33
4.3.4. Phantom power.	34

4.4.	Sistemas Embebidos.	34
4.4.1.	Arduino Due.	35
4.4.2.	FliendlyARM mini210s.	37
4.5.	Sistemas de transmisión de datos.	38
4.5.1.	Comunicación Serial.	38
4.6.	Entornos de programación.	40
4.6.1.	Visual Basic.	40
4.6.2.	Arduino.	41
5.	DESARROLLO INGENIERIL.	42
5.1.	Diseño y desarrollo del sonómetro.	43
5.1.1.	Diseño y desarrollo del pre amplificador.	44
5.1.2.	Diseño y desarrollo del software de control para el sonómetro.	51
5.2.	Adaptación del sonómetro y los sensores climáticos, al sistema embebido.	55
5.2.1.	Calibración de los sensores climáticos.	56
5.2.2.	Integración del sonómetro y los sensores climáticos.	58
5.2.3.	Adaptación al sistema embebido.	65
5.3.	Integración del sistema y visualización en tiempo real.	65
5.4.	Medición de la latencia del sistema.	72
6.	ANÁLISIS DE RESULTADOS.	75
6.1.	Comparación de las características de los sonómetros.	75
6.2.	Comparación de los datos del sonómetro con un sonómetro tipo II (Svan-943A).	76
6.3.	Comparación de los datos climáticos con una estación VAISALA.	80
6.4.	Consumo de memoria RAM del sistema.	85
7.	CONCLUSIONES.	86
8.	RECOMENDACIONES.	87
9.	BIBLIOGRAFIA.	89
	ANEXO A – Compilación del sistema operativo Windows CE en el sistema embebido.	
	ANEXO B – Creación del archivo ejecutable para la Implementación en el sistema embebido.	
	ANEXO C – Anexos Digitales.	

ÍNDICE DE FIGURAS

Figura. 1. “Diseño de un sonómetro” Josué Roberto Hernández [1].	14
Figura. 2. “Wireless Multisensor Monitoring [2].	15
Figura. 3. Desarrollo de un protocolo para comunicaciones inalámbricas e implementación en dispositivos de lógica configurable [3].	16
Figura. 4. El ruido en las operaciones mineras El caso de Yanacocha Oeste. [5].	17
Figura. 5. sistema embebido en tiempo real. [6].	18
Figura. 6. Diseño e implementación de un sistema embebido para la adquisición y transmisión de señales biomédicas a través de la red celular [7].	19
Figura. 7. Metodología de diseño Flujo de diseño de un sistema embebido [8].	20
Figura. 8. Phantom Powered Microphone Preamplifier [38]	34
Figura. 9. Sistema embebido Raspberry Pi [34]	35
Figura. 10. Sistema Embebido Arduino Due [33].	36
Figura. 11. Sistema Embebido Arduino Due [33].	36
Figura. 12. sistema embebido Mini210s [32]	37
Figura. 13. sistema embebido Mini210s con pantalla táctil [32]	38
Figura. 14. Asignación de terminales del RS232 [13]	39
Figura. 15. Entorno de programación Visual Basic.	40
Figura. 16. Entorno de programación Arduino. Fuente: propia.	41
Figura. 17. Esquema de desarrollo del sistema. Fuente propia.	42
Figura. 18. Esquema general de las etapas de desarrollo del sistema de adquisición y visualización.	42
Figura. 19. ECM8000 Behringer [38].	45
Figura. 20. Configuración de las resistencias de offset.	47
Figura. 21. Prototipo del pre amplificador desarrollado para realizar pruebas primarias. Fuente propia,	49
Figura. 22. Diseño del impreso del pre amplificador. Fuente: Propia	49
Figura. 23. Etapa final de desarrollo del pre amplificador. Fuente: Propia	50
Figura. 24. calibración del micrófono a 1KHz con 114dB de presión acústica.	50
Figura. 25. Etapas de desarrollo para el software del sonómetro.	51
Figura. 26. Código para activar la velocidad de muestreo máxima de Arduino Due.	52
Figura. 27. Diagrama de flujo para seleccionar el tiempo de integración del sonómetro.	53
Figura. 29. Cálculo del valor RMS del sonómetro.	54
Figura. 30. Código de envío del valor RMS por comunicación serial en el Arduino Due.	55
Figura. 31. Etapas de la adaptación del sonómetro y los sensores climáticos.	55
Figura. 32. Sensor de temperatura y humedad relativa SHT11 [36]	56
Figura. 33. Veleta de viento analógica. Fuente propia	57

Figura. 34. #40R Anemometer [37]	58
Figura. 35. Prototipo de tarjeta de interconexión compatible con el Arduino Due Fuente propia.	59
Figura. 36. Tarjeta de acople de sensores para Arduino DUE Fuente: propia.	59
Figura. 37. Tarjeta de acople para Arduino Due.	60
Figura. 38. Secciones de la integración en software.	61
Figura. 39. Procedimiento para la lectura del sensor SHT11.	62
Figura. 40. Código para obtener el valor de dirección del viento.	62
Figura. 41. Procedimiento para obtener el valor de la velocidad de viento.	63
Figura. 42. Envío de la trama de datos desde el Arduino Due. Fuente: propia	64
Figura. 43. Dispositivo de interconexión serial.	65
Figura. 44. Etapas de la interfaz de usuario.	66
Figura. 45. Ventana principal de la interfaz de usuario. Fuente: propia.	66
Figura. 46. Interfaz del sonómetro. Fuente: propia.	67
Figura. 47. Sección de configuración del sonómetro. Fuente: propia.	68
Figura. 48. Menú desplegable para el tiempo de integración. Fuente: propia.	68
Figura. 49. Proceso interno del botón Comenzar.	69
Figura. 50. Proceso interno de botón “resultados”.	69
Figura. 51. Sección de visualización en la interfaz del sonómetro. Fuente: propia.	70
Figura. 52Fig.52. implementación del código para visualizar los datos en las cajas de texto.	70
Figura. 53. Interfaz de monitoreo. Fuente: propia.	71
Figura. 54. Proceso de lectura y decodificación de la trama de datos.	71
Figura. 55. Ventana de visualización de las gráficas en tiempo real. Fuente: propia.	72
Figura. 56. Etapas de medición de la latencia del sistema.	72
Figura. 57. Tipos de señales, analizadas para la comparación de los sonómetros.	76
Figura. 58. Medición ruido ambiental tiempo de integración Slow.	79
Figura. 59. Medición ruido ambiental tiempo de integración Fast.	79
Figura. 60. Medición ruido ambiental tiempo de integración Impulsive.	80
Figura. 61. Comparación de la señal de temperatura.	82
Figura. 62. Comparación de la señal de humedad relativa.	82
Figura. 63. Comparación de la velocidad del viento.	83
Figura. 64. Comparación de la dirección del viento.	83
Figura. 65. Consumo de memoria RAM por la aplicación “EstaciónMeteorológica”.	85
Figura. 66. Mini210s con Touch Screen Fuente: propia	92
Figura. 67. Memoria de almacenamiento MicroSD 8Gigabytes. Fuente: Propia	93
Figura. 68. SDFlasher formato a la memoria. Fuente: propia.	94
Figura. 69. Relayout en la tarjeta SD. Fuente: propia	95
Figura. 70. Configuración del archivo de inicio en el sistema embebido. Fuente: Propia.	96
Figura. 71. Diagrama de disposición de los archivos dentro de la memoria SD. Fuente: propia.	97

Figura. 72. Boot desde la tarjeta SD. Fuente: propia	97
Figura. 73. Información previa a la instalación del sistema operativo. Fuente: propia	98
Figura. 74. Proceso de instalación de Windows CE finalizado correctamente. Fuente: propia.	98
Figura. 75. Windows CE desde el sistema embebido. Fuente: Propia.	99
Figura. 76. Compilación de la interfaz de usuario. Fuente: propia	100
Figura. 77. Archivos generados por la compilación de la interfaz. Fuente: propia.	101
Figura. 78. Rutas para acceder al archivo ejecutable .exe. Fuente: propia.	101
Figura. 79. Windows Mobile Center, dispositivo conectado. Fuente: propia.	102
Figura. 80. Dispositivos de almacenamiento en el sistema embebido. Fuente: propia.	102
Figura. 81. Archivo ejecutable almacenado en el sistema embebido. Fuente: propia.	103
Figura. 82. Archivo ejecutable almacenado en el sistema embebido. Fuente: propia.	103

ÍNDICE DE TABLAS

Tabla. 1. Especificaciones del sonómetro.	44
Tabla. 2. Características físicas del micrófono ECM8000.	46
Tabla. 3. Tiempos de ejecución de las operaciones internas del arduino.	73
Tabla. 4. Características de hardware de los dos sonómetros.	75
Tabla. 5. Características de software de los dos sonómetros.	75
Tabla. 6. Tabla de configuración para las mediciones.	76
Tabla. 7. Datos comparativos de los sonómetros.	77
Tabla. 8. Error relativo del nivel continuo equivalente.	78
Tabla. 9. Error relativo y absoluto para el nivel máximo y mínimo.	78
Tabla. 10. Tabla de resultados de la medición climática.	81
Tabla. 11. Error relativo y absoluto entre las mediciones climáticas realizadas.	84

ÍNDICE DE ECUACIONES.

Ecuación. 1. Expresión del nivel de presión sonora en dB [31].	31
Ecuación. 2. Expresión del nivel de presión sonora continuo equivalente [31].	31
Ecuación. 3. Expresión para calcular la ganancia de un pre amplificador [FALTA]	45
Ecuación. 4. Expresión para calcular la ganancia del amplificador INA217.	46
Ecuación. 5. Ley de Ohm.	47
Ecuación. 6. Cálculo del valor RMS para una señal sinusoidal [1].	51
Ecuación. 7. Expresión para el cálculo del valor RMS. [1].	54

RESUMEN

El presente documento muestra el desarrollo de un sistema de adquisición del nivel de presión sonora y variables climáticas utilizando sistemas embebidos, con el propósito de ser una herramienta fundamental en el análisis y estudio del fenómeno de la contaminación auditiva presente en la actualidad.

Una de las características del dispositivo es permitir el registro y la visualización del comportamiento ambiental de variables climáticas, como la temperatura del aire, la humedad relativa, la velocidad y la dirección del viento en tiempo real, debido a la transmisión de datos por medio del protocolo serial en intervalos pequeños de tiempo (100ms).

Por otro lado el sistema, incorpora un dispositivo desarrollado con base en la norma IEC 61672-1 [18] capaz de digitalizar, almacenar y visualizar los niveles de ruido ambiental representados con el nivel de presión sonora. Este dispositivo es semejante a un sonómetro tipo II [18] al momento de obtener estos niveles sonoros.

Este tipo de herramientas tecnológicas dan pie para analizar y desarrollar nuevas propuestas innovadoras que describan comportamientos de fenómenos sonoros, no solo los que afectan a las personas, sino también, los que favorecen procesos ingenieriles, especialmente en la ingeniería de sonido.

ABSTRACT

The paper presents the development of a system for acquiring the sound pressure level and climatic variables using embedded systems, in order to be a fundamental tool in the analysis and study of the phenomenon of noise pollution present today.

One feature of the device is that it allows the recording and display of the environmental performance of climatic variables such as air temperature, relative humidity, speed and wind direction, in real time due to this data through serial Protocol in small intervals of time (100ms).

On the other hand the total system incorporates a device developed based on the IEC 61672-1 standard [18] able to digitize, store and display the ambient noise levels represented the sound pressure level. This device behaves similarly to a type II [18] the time to get these noise levels.

This type of technological tools give rise to analyze and develop new innovative proposals that describe behaviors presented phenomena, fields of research and engineering development.

INTRODUCCIÓN

En los últimos años, la contaminación auditiva ha incrementado, debido al desarrollo industrial y al alto índice vehicular presente en las ciudades. Estos dos procesos se caracterizan por ser los mayores contaminantes auditivos en la actualidad [2], ya que generan niveles de ruido que sobrepasan el umbral de dolor. Esto ha sido perjudicial para la salud de las personas causando estrés y fatiga auditiva [31].

Los mapas y mediciones de ruido, son herramientas que permiten brindar información real del comportamiento de este fenómeno [31]. La evolución de la tecnología ha permitido que la información sea cada vez más precisa y más confiable, debido a los sistemas de procesamiento que día a día innovan sus características de hardware y de software.

Existen sistemas prácticos y versátiles llamados sistemas embebidos, capaces de realizar una función específica, definida por el diseñador. [8][9] Estos sistemas no solo son funcionales por su capacidad de procesamiento, sino también por su tamaño y por su bajo costo, en comparación con un computador.

Al ser sistemas que permiten modificar su software son útiles para implementar diferentes dispositivos enfocados a la ingeniería de sonido, como dispositivos de comunicación, de registro o de control, lo que los convierte en una herramienta capaz de analizar y estudiar fenómenos sonoros.

A raíz del problema de la contaminación auditiva, se desarrolla un sistema que permita obtener información de este fenómeno, almacenando el comportamiento climático con el fin de realizar a futuro, análisis detallados y estudios relacionados.

1. PLANTEAMIENTO DEL PROBLEMA.

1.1. ANTECEDENTES.

La evolución de la tecnología ha permitido desarrollar nuevos sistemas capaces de analizar y estudiar fenómenos que se presentan en todo el mundo, como los fenómenos climáticos. En el sistema de adquisición y visualización se encuentran tres etapas fundamentales, estas, relacionadas con trabajos ya realizados, entre ellos encontramos los siguientes.

- **JOSUÉ ROBERTO HERNÁNDEZ JUÁREZ. Diseño de un sonómetro. Tesis Profesional / Junio de 2009 [1]**

Este proyecto fue desarrollado en el 2009, en México como proyecto de grado de la Universidad Politécnico Nacional, el cual tuvo como objetivo diseñar un medidor de señales de ruido utilizando el entorno de programación labVIEW, basándose en un esquema representado en la figura 1.

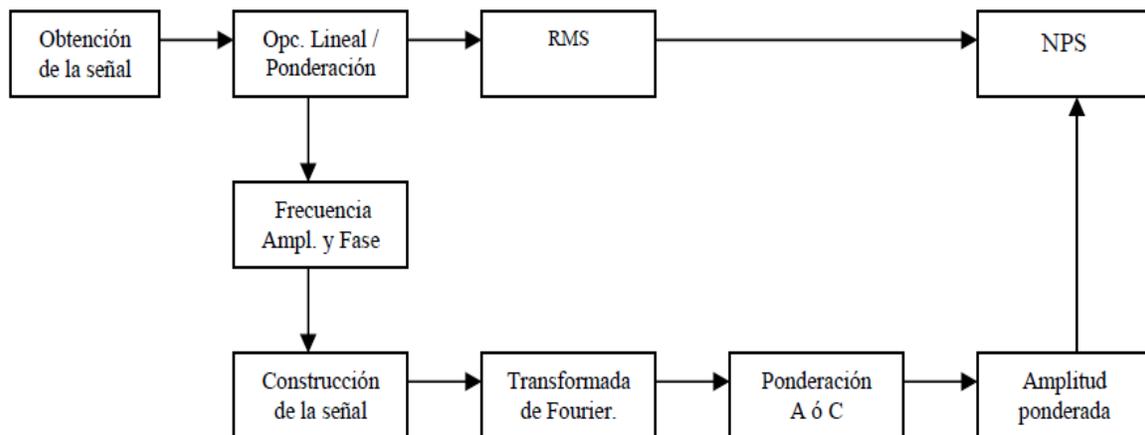


Figura. 1. "Diseño de un sonómetro" Josué Roberto Hernández [1].

- **DICKINSON, DAN; LEIDER, COLBY, MANN, DOUG. Wireless multisensor monitoring of the Florida Everglades: A pilot project. Noviembre de 2010 / Convention Paper 8186 “AES” [2]**

Es un trabajo que describe la necesidad a largo plazo del monitoreo ecológico de los datos ambientales, tales como la temperatura de superficie y la calidad del agua. Los nuevos estudios realizados por los autores Maher y sus colegas motivan, el valor de documentar de manera similar ambientes naturales de sonido en los Everglades de Florida, Estados Unidos (EE.UU), la implementación del sistema se encuentra en la figura 2.

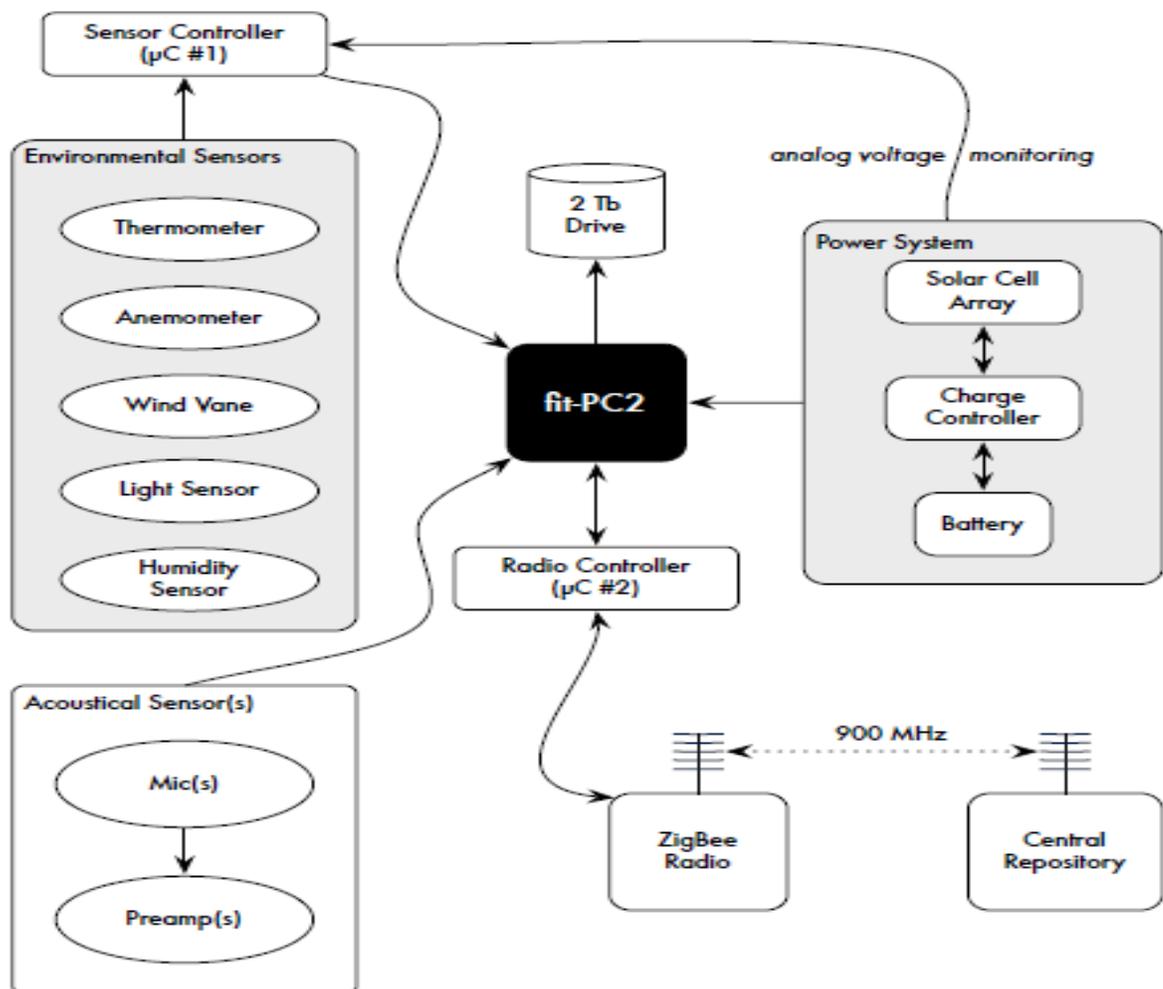


Figura. 2. “Wireless Multisensor Monitoring [2].

- **ÁLVAREZ G, PICO J VICARIO, L AMUCHÁSTEGUI, AYUSO N. BENÍTEZ N. Desarrollo de un protocolo para comunicaciones inalámbricas e implementación en dispositivos de lógica configurable. [3]**

En este trabajo se presenta un sistema de comunicación inalámbrica implementado sobre dispositivos de lógica configurable. Como se muestra en la figura 3 el sistema tiene una estructura modular para permitir una fácil adaptación a los diferentes componentes de comunicación por radio frecuencia (RF) comerciales y posibilita la utilización de diferentes bandas de comunicación libres en función de los requerimientos de velocidad de transmisión, alcance y consumo de las aplicaciones en las que se quiera utilizar.

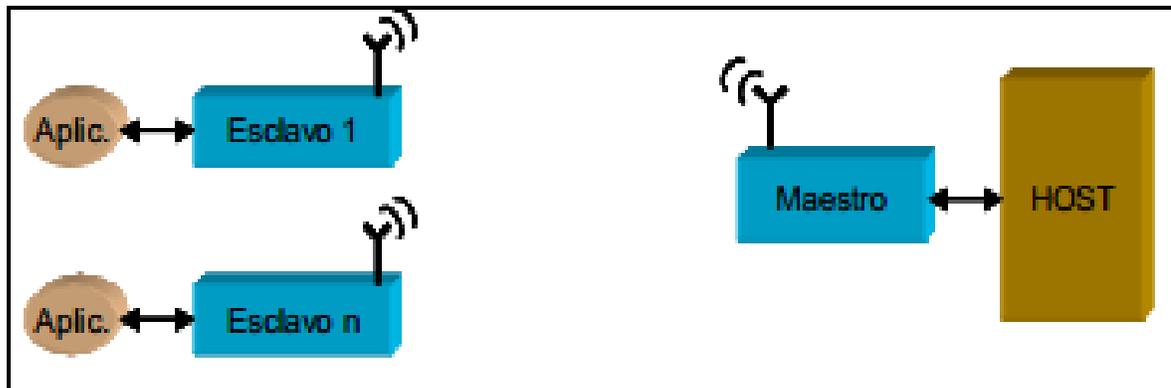


Figura. 3. Desarrollo de un protocolo para comunicaciones inalámbricas e implementación en dispositivos de lógica configurable [3].

- **CARMEN ELENA ZAPATA SÁNCHEZ. Evaluar y monitorear la calidad del aire en el Valle de Aburrá. De abril de 2011 a mayo de 2012. [4]**

La red de monitoreo de calidad del aire, meteorología y ruido del Área Metropolitana Valle de Aburrá cuenta con ocho Estaciones de monitoreo de Ruido Ambiental, siete fijas y una estación móvil, ubicadas a lo largo del Valle de Aburrá, dos de estas vienen operando desde el año 2009 y otras seis fueron adquiridas en el año 2011 y puestas a operar en el mes de abril del mismo año para tener una referencia del comportamiento del ruido en este lugar.

- **EDSON PLASENCIA SÁNCHEZ, CARLOS CABRERA CARRANZA. El ruido en las operaciones mineras; El caso de Yanacocha Oeste. Junio 2009 [5]**

El documento interpreta y analiza técnicamente la propuesta de modificación al Plan de Monitoreo Ambiental suplementario de Yanacocha Oeste presentado en octubre del 2008 ante el Ministerio de Energía y Minas del Perú como un ejemplo de la importancia que tiene la gestión del ruido en las operaciones mineras. Uno de los mapas de ruido que se describen en este estudio se muestra en la figura 4.

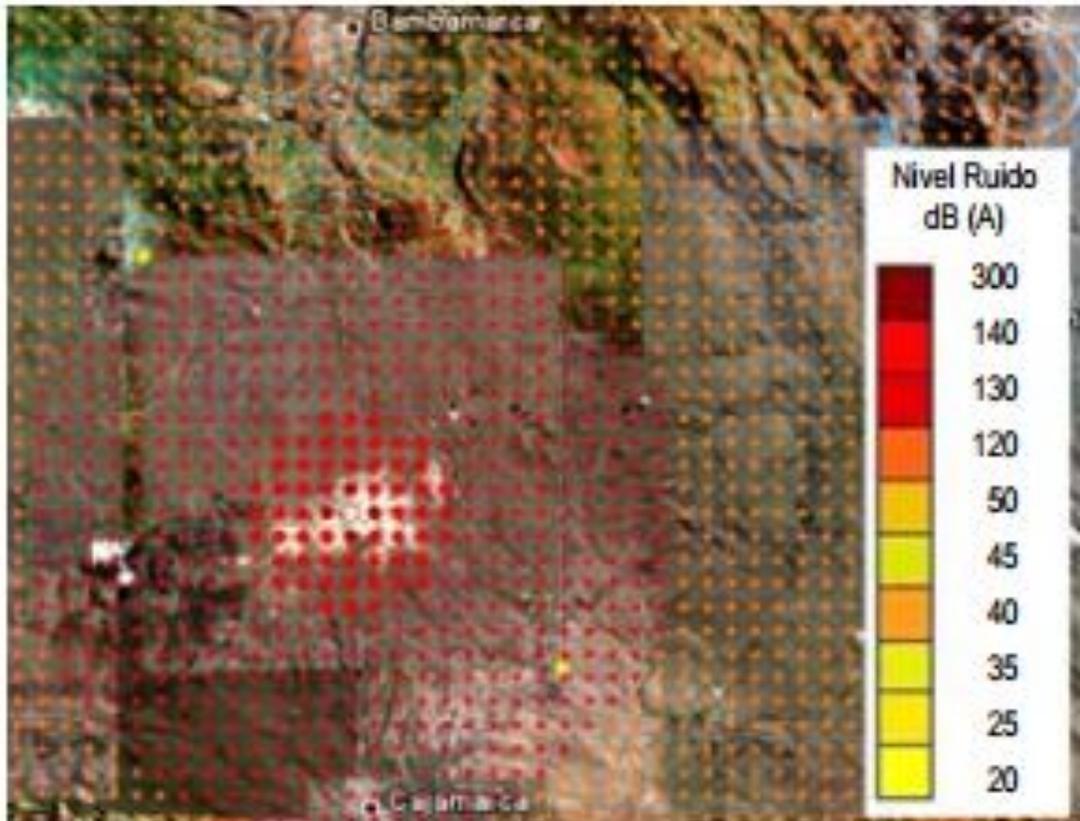


Figura. 4. El ruido en las operaciones mineras El caso de Yanacocha Oeste. [5].

- **JOSÉ LUIS DÍAZ BERNABÉ.** Diseño de un sistema embebido para el control en tiempo real de un convertidor elevador alimentado por una celda combustible. Noviembre 2009.[6]

La celda de combustible es una fuente de energía limpia, porque no genera contaminación ambiental, como otras fuentes de energía convencionales que utilizan combustibles derivados de petróleo. La celda de combustible es una fuente de voltaje sensible a la carga. Es necesario implementar un dispositivo para acondicionar la energía generada a un valor nominal de voltaje y regularlo. En este trabajo se desarrolla el diseño y construcción de un control no-lineal para el control de un convertidor elevador para una celda de combustible tipo PEMFC de 150W para acoplar una carga de 24V, este control se representa en la figura 5.

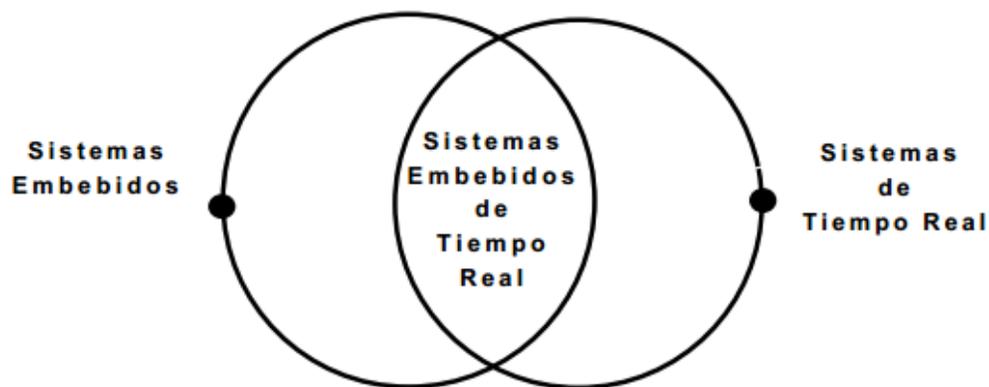


Figura. 5. sistema embebido en tiempo real. [6].

- **NELSON FELIPE ROSAS JIMÉNEZ.** Diseño e implementación de un sistema embebido para la adquisición y transmisión de señales biomédicas a través de la red celular. Bogotá 2011 [7]

Esta tesis presenta el diseño e implementación de un Sistema Embebido como se muestra en la figura 6, capaz de capturar y transmitir por diversos tipos de redes de telecomunicaciones la información de señales biomédicas, correspondientes a señales

de electrocardiografía, frecuencia cardiaca, saturación de oxígeno en la sangre y presión arterial. Esta información es transmitida al sistema clínico "SARURO", en el cual se realiza la visualización y se puede efectuar un análisis detallado por parte de un médico y/o de un especialista, sin la necesidad del contacto directo con el paciente.

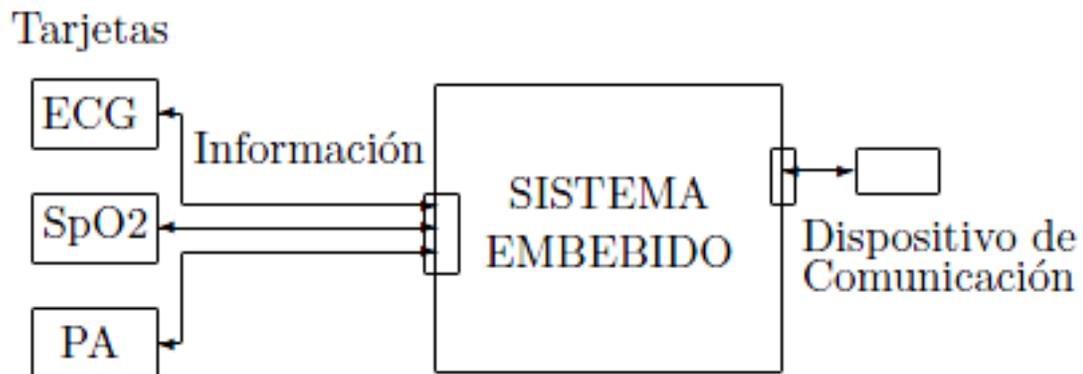


Figura. 6. Diseño e implementación de un sistema embebido para la adquisición y transmisión de señales biomédicas a través de la red celular [7].

- **CARLOS IVAN CAMARGO BAREÑO.** Transferencia tecnológica y de conocimientos en el diseño de sistemas embebidos. Universidad Nacional de Colombia 2011.[8]

Utilizando la capacitación como canal para la transferencia se creó un plan de estudios para la línea de electrónica digital que ayuda a formar en los profesionales las habilidades necesarias para concebir, diseñar, implementar y operar sistemas digitales. Para ayudar en el proceso de enseñanza/aprendizaje y con el fin de facilitar la difusión en la industria, se diseñaron una serie de plataformas hardware que pueden ser utilizadas como diseños

de referencia en la fabricación de nuevos productos, la información necesaria para entender, reproducir u modificar.

Finalmente, tomando como referencia el movimiento de software libre y código abierto se difundirán los resultados de este trabajo a todos los sectores de la sociedad.

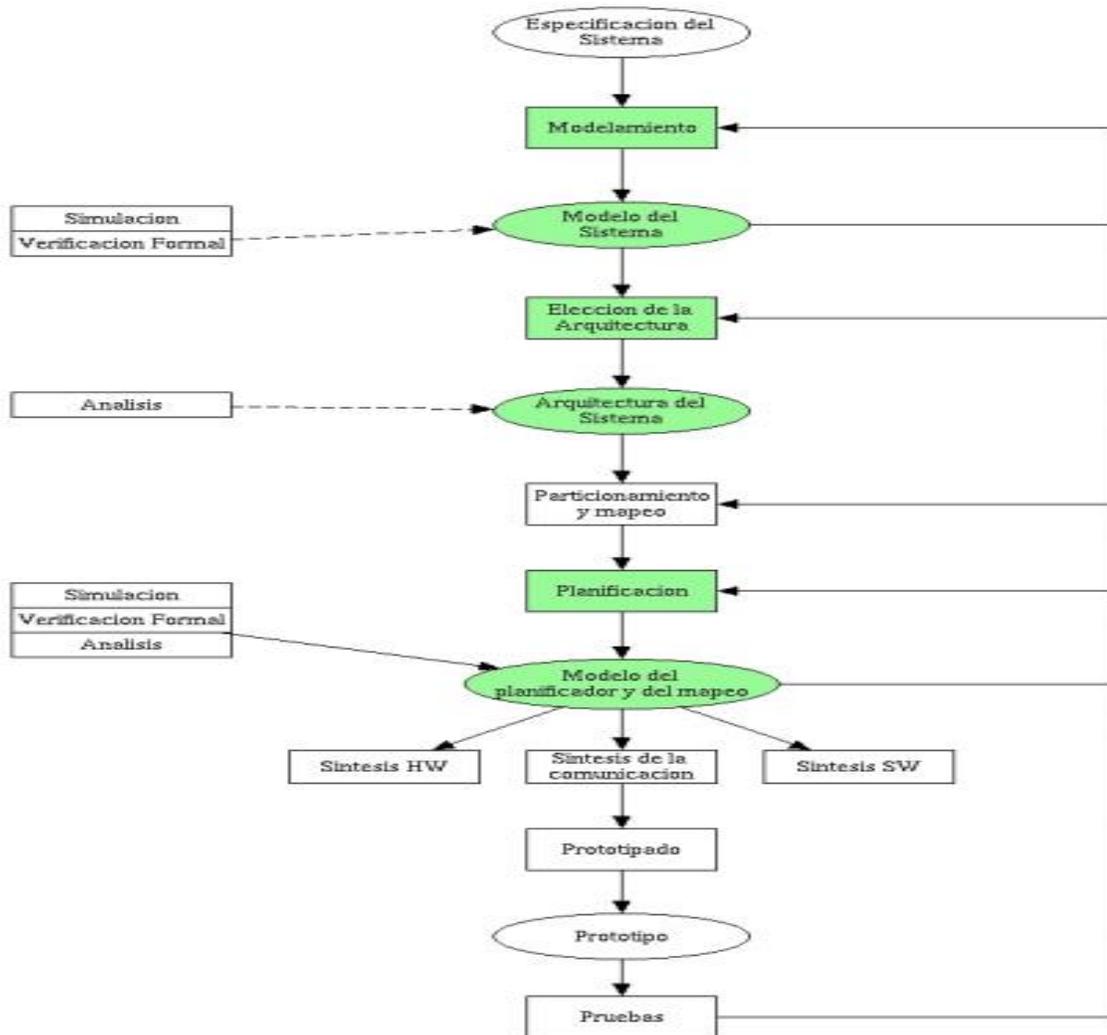


Figura. 7. Metodología de diseño Flujo de diseño de un sistema embebido [8].

1.2. DESCRIPCIÓN DEL PROBLEMA.

La contaminación auditiva es un factor de riesgo sanitario que genera molestias, perturbación del sueño y posibles efectos adversos para la salud; como la hipoacusia o sordera y una intervención perjudicial sobre la calidad de vida [3].

Este tipo de contaminación se debe al incremento del desarrollo urbano, industrial y alto tráfico vehicular. Esta situación, ha llevado a las entidades de salud ambiental a desarrollar estrategias y actividades que permitan el control y la monitorización de este fenómeno, como la resolución 0627 del 2006 [31], al igual que el diseño de sistemas que determinen el daño que se está generando sobre la población. Por lo cual, se proponen medidas que controlen los efectos sobre la salud.

Actualmente, existen sistemas de medición de ruido utilizados para su análisis y estaciones meteorológicas que se emplean para analizar las variables ambientales [1][2]. En consecuencia, surge la necesidad de implementar un dispositivo unificado que permita medir estos parámetros a la vez. Por lo cual, se pretende adaptar las tarjetas de adquisición de datos ambientales con el dispositivo de medición de ruido, teniendo en cuenta las características tanto en hardware como en software de los dos dispositivos.

A partir de lo anterior se formula la siguiente pregunta, ¿cuáles son los parámetros y características de hardware y software, para la adquisición de las señales climáticas y los niveles de presión sonora, en el desarrollo de un sistema de adquisición, utilizando sistemas embebidos?

1.3. JUSTIFICACIÓN.

El ruido ambiental ha sido reconocido como un factor de riesgo sanitario [2], que ha generado la necesidad de analizar el comportamiento de este fenómeno en diferentes establecimientos públicos y residenciales.

Para realizar un análisis adecuado es necesario disponer de información confiable y efectiva, por lo cual realizar un sistema que permita la adquisición y la medición de esos datos se hace evidente, dado que la medición y registro del ruido junto a variables ambientales como la fluctuación de la temperatura, la humedad del aire, la velocidad del viento, son indispensables para realizar un análisis que permita de manera conjunta, elaborar predicciones tanto meteorológicas como de contaminación auditiva con un solo dispositivo.

Otro punto de vista en torno al desarrollo del sistema, es la transferencia de tecnología que juega un papel importante; Países como Colombia dependen de tecnología extranjera haciendo necesario importar nuevas tecnologías, esto genera que Colombia sea un país con bajo desarrollo industrial, el desarrollo de un sistema de adquisición innovaría la industria colombiana dando pie para que a futuro se puedan desarrollar nuevos sistemas y nuevos procesos.

1.4. OBJETIVOS.

1.4.1. Objetivo General.

- Desarrollar un sistema de adquisición y visualización de señales de ruido y variables ambientales, utilizando sistemas embebidos.

1.4.2. Objetivos Específicos.

- Desarrollar un sonómetro que permita la adquisición y digitalización del nivel de presión sonora teniendo como base las especificaciones de la norma IEC61672 para la construcción de un sonómetro tipo 2.
- Integrar el sonómetro y adaptar las tarjetas de adquisición de las variables climáticas, como temperatura, humedad relativa y velocidad del viento, al sistema embebido.
- Desarrollar la interfaz de software para la visualización de los datos obtenidos en el sistema embebido.

1.5. ALCANCES Y LIMITACIONES.

1.5.1. Alcances.

- Comparar los datos obtenidos del sonómetro con un sonómetro tipo 2.
- Las tarjetas de adquisición de las variables climáticas se calibrarán con respecto a la estación Davis.
- La visualización de los datos obtenidos se realizará en una interfaz gráfica de usuario permitiendo su almacenamiento.
- El sonómetro permitirá obtener el nivel de presión sonora instantáneo.

1.5.2. Limitaciones.

- El sonómetro no tendrá ningún tipo de validación normativa.
- El sonómetro no podrá realizar ningún tipo de análisis frecuencial para las mediciones.
- Los datos que presente el sonómetro serán en ponderación lineal.
- Los sensores de adquisición de datos a utilizar serán comerciales lo cual impide la modificación de sus características.

2. METODOLOGIA.

El desarrollo del proyecto se enfoca en una metodología propuesta en la tesis de doctorado de Carlos Camargo [8] explicada en la figura 7.

2.1. ENFOQUE DE LA INVESTIGACIÓN.

El enfoque de este proyecto es empírico - analítico, debido a la aplicación de nuevas formas de procesamiento de señales ambientales, a través de procesos de transferencia de tecnología.

Con el desarrollo de este dispositivo pueden comprobarse la efectividad y versatilidad de la aplicación de sistemas embebidos para el procesamiento de señales.

2.2. HIPÓTESIS.

Por medio de la implementación de un sistema embebido se podrá realizar un sistema de adquisición y visualización de niveles de ruido y variables ambientales, que permitirá al usuario estudiar y analizar el fenómeno de la contaminación auditiva.

2.3. VARIABLES

2.3.1. Variables Independientes.

- Función de transferencia de los sensores.

- Protocolo Serial

2.3.2. Variables Dependientes.

- Latencia en la transmisión de datos por protocolo serial.

3. LÍNEA DE INVESTIGACIÓN DE LA UNIVERSIDAD / SUB-LÍNEA DE LA FACULTAD / CAMPO TEMATICO DEL PROGRAMA.

- Línea de investigación: análisis y procesamiento de señales.
- Núcleo Problemático: Acústica y audio.

4. MARCO TEORICO.

En el desarrollo de un sistema de adquisición de datos ambientales, es necesario identificar las siguientes temáticas: Sistemas Embebidos, Estaciones de monitorización, Sonómetros y entornos de programación.

4.1. Estaciones de meteorología.

Una estación meteorológica, es un sistema diseñado para medir y almacenar datos correspondientes al comportamiento climático en un punto específico. Estos datos son utilizados para elaborar predicciones ambientales y análisis atmosférico [28]. Las estaciones meteorológicas se utilizan para realizar estudios de la calidad del aire en diferentes zonas del mundo, teniendo en cuenta los factores que están inmersos en el aire como su temperatura, su humedad, su velocidad y dirección [1][3].

4.1.1. Sensor de temperatura.

Estos elementos se utilizan en todo tipo de proceso que requiera indicación y/o control de la variable de temperatura, básicamente se encuentran dos tipos de sensores; Termocuplas y Termo resistencias (RTD).

- **Termocuplas**

Termocupla, es un par de alambres de distinto material o composición, unidos en un extremo. En donde al aplicar temperatura en la unión se genera una tensión en Mili voltios, tensión que aumenta proporcionalmente con el aumento de la Temperatura [3]. Existen varios tipos de termocupla, puesto que cualquier par de metales conformaría un tipo determinado. Sin embargo, la empírica ha llevado al uso de ciertos tipos estandarizados, a los que se les cita por una letra (las más típicas son las tipo J, K). Cada tipo difiere en el material de los metales A y B. Al diferir los materiales de construcción difieren los rangos de trabajo, el voltaje generado por unidad de grado y la máxima temperatura útil (antes que se funda) [4] [28].

4.1.2. Sensor de humedad relativa.

Un sensor de humedad es un dispositivo que mide la humedad relativa en un área dada. El cual puede ser utilizado tanto en interiores como en exteriores. Los sensores de humedad están disponibles en formas tanto analógicas como digitales [28].

Un sensor analógico de humedad mide la humedad del aire relativo usando un sistema basado en un condensador. El sensor está hecho de una película generalmente de vidrio o de cerámica. El material aislante que absorbe el agua está hecho de un polímero que toma y libera el agua basándose en la humedad relativa de la zona dada. Esto cambia el nivel de carga en el condensador del circuito en el cuadro eléctrico.

Un sensor digital de humedad funciona a través de dos micro-sensores que se calibran a la humedad relativa de la zona dada. Estos se convierten luego en el formato digital a través de un proceso de conversión de analógico a digital que se realiza mediante un chip situado en el mismo circuito. Un sistema basado en una máquina hecha de electrodos con polímeros es lo que constituye la capacitancia del sensor. Esto protege el sensor del panel frontal del usuario [4].

4.1.3. Anemómetro.

El anemómetro es un instrumento que se utiliza para medir la velocidad del viento en un punto específico [31], El más habitual es el anemómetro de cazoletas giratorias (tres o cuatro semiesferas huecas montadas sobre un eje vertical, en su giro cada cazoleta recibe el viento por la parte cóncava reemplazándose en su movimiento por la siguiente) que gira a mayor velocidad cuanto mayor sea la velocidad del viento, cuenta el número de revoluciones y de esta manera calcula velocidad en m/seg o km/h [5].

Existen varios tipos de anemómetros como:

- **Anemómetro de empuje**

Están formados por una esfera hueca y ligera o una pala, cuya posición respecto a un punto de suspensión varía con la fuerza del viento, lo cual se mide en un cuadrante [4].

- **Anemómetro de rotación o de copelas**

Está dotado de cazoletas (Robinson) o hélices unidas a un eje central cuyo giro, proporcional a la velocidad del viento, es registrado convenientemente, en los anemómetros magnéticos, dicho giro activa un diminuto generador eléctrico que facilita una medida precisa[4].

- **Anemómetro de compresión**

Se basa en el tubo de Pitot y está formado por dos pequeños tubos, uno de ellos con orificio frontal (que mide la presión dinámica) y lateral (que mide la presión estática,) y el otro sólo con un orificio lateral .La diferencia entre las presiones medidas permite determinar la velocidad del viento [4].

4.1.4. Veleta de viento.

Una veleta es un dispositivo giratorio que consta de una placa que gira libremente, y un señalador que indica la dirección del viento, al igual que los anemómetros existen varios tipos de veletas, definidas por el tipo de datos que genera la transducción[4].

- **Veletas analógicas o resistivas**

Las veletas analógicas o resistivas poseen un potenciómetro de diseño especial, el cual tiene un recorrido específico definido, lo que permite que al finalizar el recorrido de 365° el potenciómetro vuelva a su resistencia inicial lo que permite dar saltos de 365° a 1°.

Generalmente el potenciómetro tiene una impedancia total de 20K-ohm [3].

- **Veletas Digitales**

Las veletas digitales tienen un sensor de ubicación como una brújula que les permite dar un valor referenciado al norte, estos datos son entregados en 8 bits de serial asincrónico para ser recibidos por micro controladores o dispositivos que permitan la decodificación de este tipo de datos [3] [4].

4.2. Sonómetro.

El sonómetro es un instrumento y herramienta que permite obtener el valor del nivel de presión sonora en un momento y lugar determinado, conformado por un micrófono que se encarga de realizar la transducción de energía acústica en energía eléctrica y un módulo de procesamiento [17].

La norma CEI 60651 y la norma CEI 60804, emitidas por la CEI (Comisión Electrotécnica Internacional), establecen las normas que han de seguir los fabricantes de sonómetros [17].

Se intenta que todas las marcas y modelos ofrezcan una misma medición ante un sonido dado. La CEI también se conoce por sus siglas en inglés: IEC (*International Electrotechnical Commission*), por lo que las normas aducidas también se conocen con

esta nomenclatura: IEC 60651 (1979) y la IEC 60804 (1985). A partir del año 2003, la norma IEC 61.672 unifica estas normas [17][31].

Los sonómetros se clasifican según su precisión y su función de esta forma:

- **Modelo Básico**

Toma únicamente el nivel de presión instantáneo en dB.

- **Modelo Integrador**

Los sonómetros integradores - promediadores podrán emplearse para la medición del Nivel de presión acústica continuo equivalente ponderado A (LAeq, T). En los llamados sonómetros integradores, permite seleccionar la curva de ponderación que va a ser usada:

- **Curva A (dBA).** Mide la respuesta del oído, ante un sonido de intensidad baja. Es la más semejante a la percepción logarítmica del oído humano. Se utiliza para establecer el nivel de contaminación acústica y el riesgo que sufre el hombre al ser expuesto a la misma. Por ello, es la curva que se utiliza a la hora de legislar.
- **Curva B (dBB).** Su función era medir la respuesta del oído ante intensidades para intensidades medias. Como no tiene demasiadas aplicaciones prácticas es una de las menos utilizadas.
- **Curva C (dBC).** Mide la respuesta del oído ante sonidos de gran intensidad. Es tanto, o más empleada que la curva A, a la hora de medir los niveles de contaminación acústica.
- **Curva D (dBD).** Se utiliza, casi exclusivamente, para estudiar el nivel de ruido generado por los aviones [17].

4.2.1. Tiempo de integración.

El tiempo de integración de los sonómetros ajusta la constante de tiempo que se utilizará en las medidas, y con ello determina la velocidad de respuesta del sonómetro frente a las variaciones de presión sonora [31].

Para evitar diferencias de lecturas sobre señales sonoras idénticas, los períodos de promediación o constantes de tiempo están unificados y son los siguientes:

- Lento (slow) promedio sobre 1s. Se usa cuando la señal fluctúa ampliamente.
- Rápido (fast) promedio sobre 125 ms. Se usa para señales estacionarias.
- Impulso (Impulse) promedio de subida 35 ms. promediación de caída 1s. Se usa para medir señales transitorias desde el punto de vista de sensación sonora en el oído humano como señales impulsivas.

En algunos casos es necesario recurrir a constantes de tiempo más rápidas que las antes citadas; como el valor valor Pico (Peak), con tiempos de subida del orden de los 50 microsegundos [17].

4.2.2. Nivel de presión sonora.

La "Presión Sonora" se refiere a la desviación en la presión del aire respecto a la presión atmosférica, causada por una onda sonora [31].

El Nivel de Presión Sonora es la medición logarítmica del valor promedio de la presión sonora, respecto a un nivel de referencia. Generalmente ese nivel de referencia es lo que se considera la presión sonora, del sonido más débil audible: 20 micro Pascales [31].

Mientras que la presión sonora se mide en Pascales, el Nivel de Presión Sonora se mide en decibeles SPL, y se abrevia "dB SPL", por las siglas en inglés de Sound Pressure Level. Cuantos más decibeles SPL tenga un sonido, más fuerte, o con más volumen es percibido. La exposición prolongada a niveles mayores a 85 dB SPL (aproximadamente) causa pérdida temporal de la audición, aunque también puede provocar daño irreversible. Niveles de presión sonora de alrededor de 120 dB SPL causan daño irreversible al oído,

y es en este punto donde se encuentra el umbral de dolor [31][3]. El nivel de presión sonora está definido por la ecuación número 1.

$$SPL = 20 \log\left(\frac{P}{P_0}\right)$$

Ecuación. 1. Expresión del nivel de presión sonora en dB [31].

Donde P es el valor cuadrático medio de la presión sonora y Po la presión de referencia [31].

4.2.3. Nivel continuo equivalente.

Nivel sonoro continuo equivalente, es el nivel en dBA de un ruido constante hipotético correspondiente a la misma cantidad de energía acústica que el ruido real considerado, en un punto determinado durante un período de tiempo T, la ecuación 2 representa la expresión matemática para el cálculo del Leq.

$$Leq(dB) = 10 \log\left(\frac{1}{T} \sum t_i 10^{\frac{L_i}{10}}\right)$$

Ecuación. 2. Expresión del nivel de presión sonora continuo equivalente [31].

Donde N es el número total de intervalos en los que se divide T y $(L_{eq})_i$ es el nivel continuo equivalente en el intervalo i-ésimo [31].

4.3. Micrófonos.

El micrófono es un transductor electro acústico. Su función es traducir las vibraciones debidas a la presión acústica ejercida sobre su cápsula por las ondas sonoras en energía eléctrica, lo que permite grabar sonidos de cualquier lugar o elemento [22][23]. Existen varios tipos de funcionamiento de micrófonos los más comunes y más utilizados son:

4.3.1. Micrófonos dinámicos.

También conocidos como micrófonos de bobina móvil, son los más usados alrededor del mundo. Son muy versátiles al ser resistentes, no necesitando energía externa y buena sensibilidad dependiendo su calidad. Su funcionamiento se atribuye a la vibración del diafragma con las ondas sonoras, lo que hace a su vez vibrar la bobina suspendida en un campo magnético, obteniendo así una pequeña corriente eléctrica causada por la fricción. La fluctuación de esta corriente es similar en forma, amplitud y frecuencia a la onda sonora que la generó [22].

La energía eléctrica generada por un micrófono es muy baja, por lo cual para reproducir la señal eléctrica se requiere la amplificación varias veces. La señal generalmente se designa con mili voltios.

Como ventajas tiene más rango de frecuencia que los de carbón y los piezoeléctricos (cristal y cerámica), son confiables, económicos, robustos, no muy delicados, buen nivel sonoro de salida aunque todo depende claramente de la calidad del micrófono [23].

4.3.2. Micrófonos Capacitivos.

Los micrófonos de condensador eléctrico poseen muchas ventajas sobre micrófonos dinámicos, desde su incomparable calidad de respuesta como su profesionalidad en todas las características dichas anteriormente.

Constan de un condensador o capacitor que es un dispositivo que almacena energía en forma de campo electrostático. Una de sus desventajas es el uso o la necesidad de una fuente externa de energía, aunque la mayoría de las veces sean pilas pequeñas además que su alta sensibilidad puede mostrar desventaja con sonidos muy fuertes al crear distorsión, por lo cual son más usados en estudios profesionales.

Su funcionamiento se basa en un par de placas metálicas separadas, cuando se conecta a una fuente de energía el condensador se carga eléctricamente, con lo cual se aplica un voltaje a las placas del capacitor. Este voltaje cambia de acuerdo a algunas variables, como el tamaño de las placas, la distancia entre ellas. Generalmente una de las placas

esta fija y la otra es móvil por lo que se puede usar una de estas como diafragma directamente además que se puede variar el voltaje que es el objetivo del micrófono convertir impulsos mecánicos en impulsos eléctricos. Entre más cerca estén las placas mayor será el voltaje y viceversa.

La masa de las placas debe ser muy baja, especialmente la del diafragma, pues esto le brindara alta sensibilidad al micrófono, hasta el punto de a veces ser hecho de una membrana plástica o una película de poliéster cubierto con una delgada capa de oro para hacerlo conductivo.

Lo siguiente es el dispositivo encargado de cargar eléctricamente el condensador conocido como voltaje polarizante y que debe estar entre un rango de 50 y 200 voltios. La variación de este voltaje se da con las vibraciones mecánicas del diafragma causadas a su vez por las vibraciones sonoras [22].

Otro punto de gran importancia, es la impedancia o resistencia en los circuitos y cableados de los micrófonos de capacitor, la cual es muy alta lo que es un problema, pues como se decía anteriormente es preferible una baja impedancia para no distorsionar el sonido a distancias de más de cinco metros. De forma que se usa otro dispositivo llamado conversor de impedancia, el cual reduce la impedancia de 10 millones de ohms (10 megahms) a una escala entre 50Ω y 200Ω [22].

4.3.3. Pre amplificación.

El pre amplificador es un tipo de amplificador electrónico utilizado principalmente para audio. Como todo amplificador su función es aumentar la señal de entrada según una ganancia específica, la ganancia es la relación entre la señal de salida y la señal de entrada [23][19].

Se utiliza para llevar niveles bajos de voltaje del orden de los mili voltios a niveles nominales definidos en +4dBu o niveles de línea. (1.23Voltios).

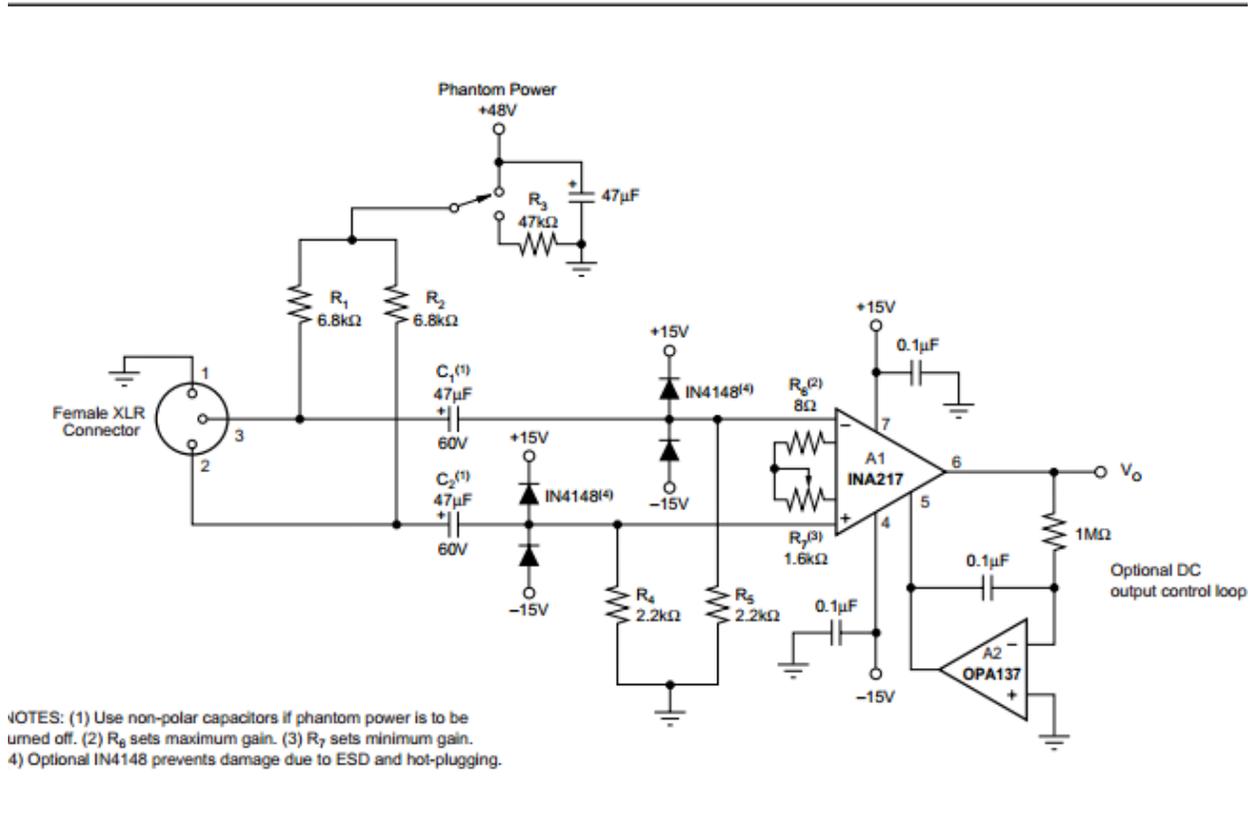


Figura. 8. Phantom Powered Microphone Preamplifier [38]

4.3.4. Phantom power.

La alimentación fantasma o “Phantom power” es un voltaje de corriente directa que se suministra a diferentes dispositivos de audio para polarizarlos o alimentarlos, los más comunes son micrófonos capacitivos, y cajas directas pasivas.

Generalmente el voltaje que se suministra es de +48 voltios aunque en algunos casos se utiliza +18 y +12 voltios [23].

4.4. Sistemas Embebidos.

Son sistemas programables, diseñados para realizar tareas específicas determinadas por el usuario, con el fin de optimizar los procesos para mejorar su desempeño y eficiencia, reduciendo tamaño y costos de producción. Los sistemas embebidos se convierten en

herramientas de desarrollo útiles dentro de diversos ámbitos de la ingeniería de sonido [7].

A diferencia de los computadores convencionales se caracterizan por el bajo consumo de energía, son económicos, poseen periféricos limitados, implementación de diferentes sistemas operativos y algunos son diseñados para realizar tareas específicas.

Tienen tres componentes principales: Procesador, Dispositivos de Almacenamiento y Periféricos. Los cuales varían según los parámetros de diseño requeridos. Para sistemas embebidos pre diseñados o de uso general. En la figura 9 se presenta el sistema embebido más común en el mercado [34].



Figura. 9. Sistema embebido Raspberry Pi [34]

Utilizan arquitecturas de procesador similares, todas son arquitectura ARM, utilizan una memoria SD/MicroSD como dispositivo de almacenamiento y memoria de trabajo. [32][33].

4.4.1. Arduino Due.

El arduino Due es una placa electrónica producida por la compañía arduino, basada en un micro controlador de referencia “Atmel SAM3X8E ARM cortex-M3”, es la primera placa de arduino de núcleo ARM de 32bits. Posee 54 entradas/Salidas digitales, de las cuales 12 pueden ser utilizadas como salidas PWM, 12 entradas analógicas, 4 puertos seriales,

y un reloj de 84MHz, la frecuencia de muestreo del conversor análogo digital oscila entre 500KHz y 1MHz [33].

El arduino Due se diferencia de las otras placas arduino por su eficiencia y capacidad de procesamiento, debido a las características y elementos con los que fue diseñado y ensamblado lo que lo hace idóneo para desarrollos enfocados a audio y comunicación [33]. Las figuras 10 y 11 representan las dos secciones del Arduino Due respectivamente.

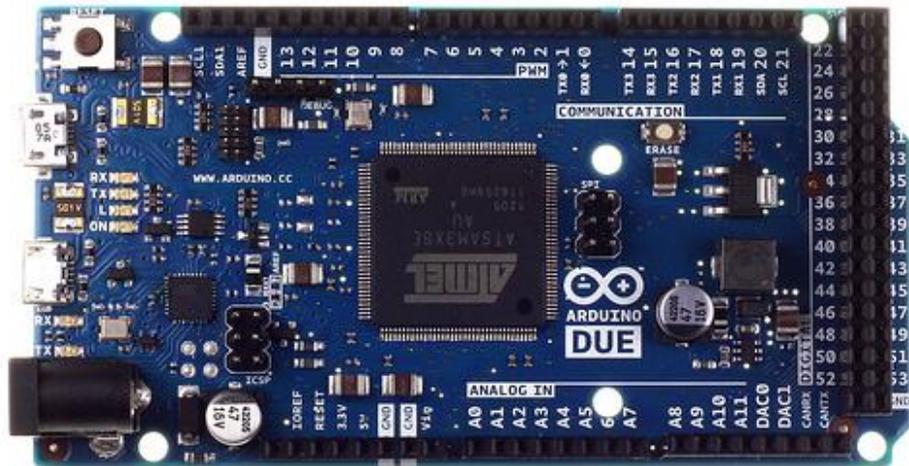


Figura. 10. Sistema Embebido Arduino Due [33].

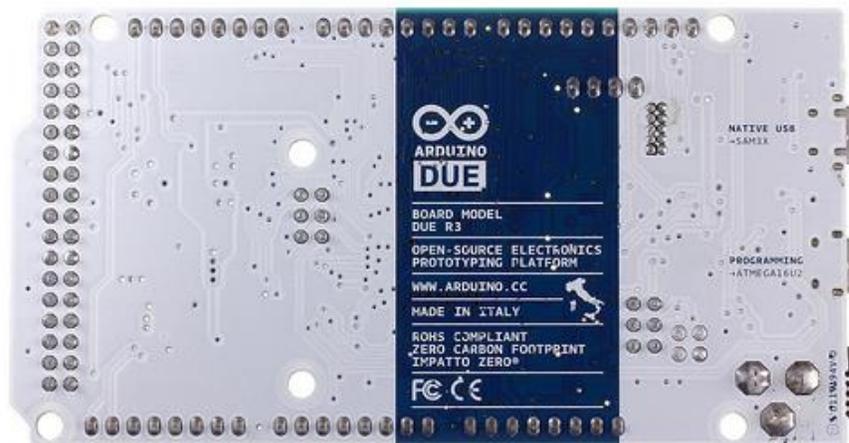


Figura. 11. Sistema Embebido Arduino Due [33].

4.4.2. FriendlyARM mini210s.

Mini210s es un sistema embebido diseñado y desarrollado por la empresa Friendly ARM, basado en el procesador “**Samsung S5PV210 ARM Cortex-A8**” con un reloj de 1GHz, en la figura 12 se describe los componentes internos de este sistema.

En este sistema se permite la implementación de tres diferentes sistemas operativos como Android, Linux y Windows CE, lo que permite que sea un dispositivo multi propósito en cualquier proyecto, además de contar con cuatro puertos de comunicación serial para la interconexión con otros dispositivos.

El sistema embebido posee una pantalla táctil resistiva como se muestra en la figura 13, para visualizar el sistema operativo que se encuentre instalado internamente, esto lo convierte en un sistema útil para implementar aplicaciones graficas que permitan controlarse por un usuario.

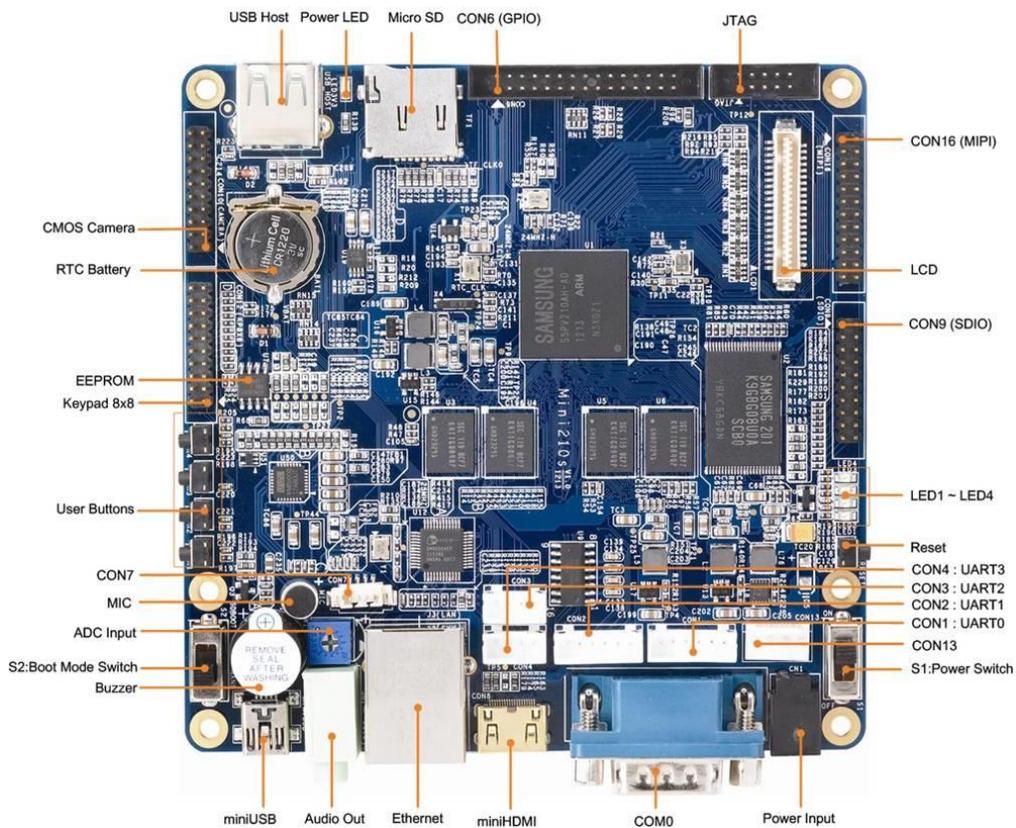


Figura. 12. Sistema embebido Mini210s [32]



Figura. 13. Sistema embebido Mini210s con pantalla táctil [32]

4.5. Sistemas de transmisión de datos.

Son los encargados de interconectar dos o más dispositivos por medio de la transmisión de señales ya sea por vía cableada, o por vía inalámbrica utilizando la modulación de ondas electromagnéticas. Los dispositivos físicos son utilizados como emisor y receptor de señales. [34]

4.5.1. Comunicación Serial.

La comunicación serial es un protocolo para comunicar diferentes tipos de sistemas, La mayoría de las computadoras incluyen dos puertos seriales. La comunicación serial es también un protocolo común utilizado por varios dispositivos para instrumentación; existen varios dispositivos compatibles entre sí que incluyen un puerto RS-232. Además, la comunicación serial puede ser utilizada para adquisición de datos si se usa en conjunto con un dispositivo remoto de muestreo [13].

El concepto de comunicación serial es sencillo. El puerto serial envía y recibe bytes de información un bit a la vez. [13][35], en la figura 14 se muestra el diagrama de conexión de un puerto serial.

- **Velocidad de transmisión (baud rate)**

Indica el número de bits por segundo que se transfieren, y se mide en baudios (*bauds*). Por ejemplo, 300 baudios representan 300 bits por segundo. Cuando se hace referencia a los ciclos de reloj se está hablando de la velocidad de transmisión. Por ejemplo, si el protocolo hace una llamada a 4800 ciclos de reloj, entonces el reloj está corriendo a 4800 Hz, lo que significa que el puerto serial está muestreando las líneas de transmisión a 4800 Hz [35].



Figura. 14. Asignación de terminales del RS232 [13]

4.6. Entornos de programación.

Los sistemas embebidos requieren la instalación de un software adicional para poder controlarlos, en el caso de los dos sistemas que se implementan para el desarrollo de este proyecto es necesario desarrollar el control en los siguientes entornos de programación.

4.6.1. Visual Basic.

Visual Basic es un lenguaje de programación dirigido por eventos, desarrollado por Alan Cooper para Microsoft. Este lenguaje de programación es un dialecto de BASIC, con importantes agregados. Su primera versión fue presentada en 1991, con la intención de simplificar la programación utilizando un ambiente de desarrollo completamente gráfico que facilitara la creación de interfaces gráficas y, en cierta medida, también la programación misma [11][15].

En la figura 15 se presenta el entorno de programación de visual basic.

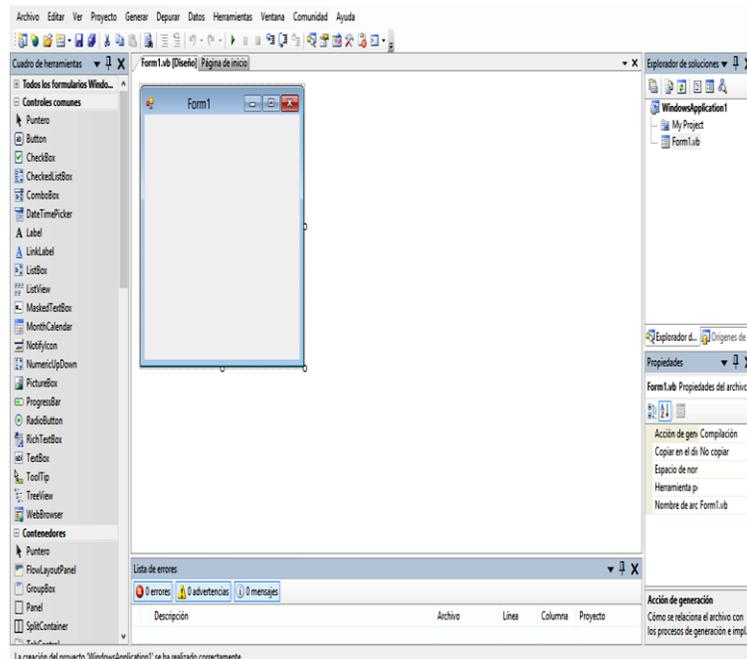


Figura. 15. Entorno de programación Visual Basic.

4.6.2. Arduino.

La plataforma Arduino mostrada en la figura 16, se basa en el lenguaje de programación de alto nivel llamado “*Processing*”. Sin embargo, es posible utilizar otros lenguajes de programación y aplicaciones populares en Arduino, debido a que Arduino usa la transmisión serial de datos soportada por la mayoría de los lenguajes de programación [33].

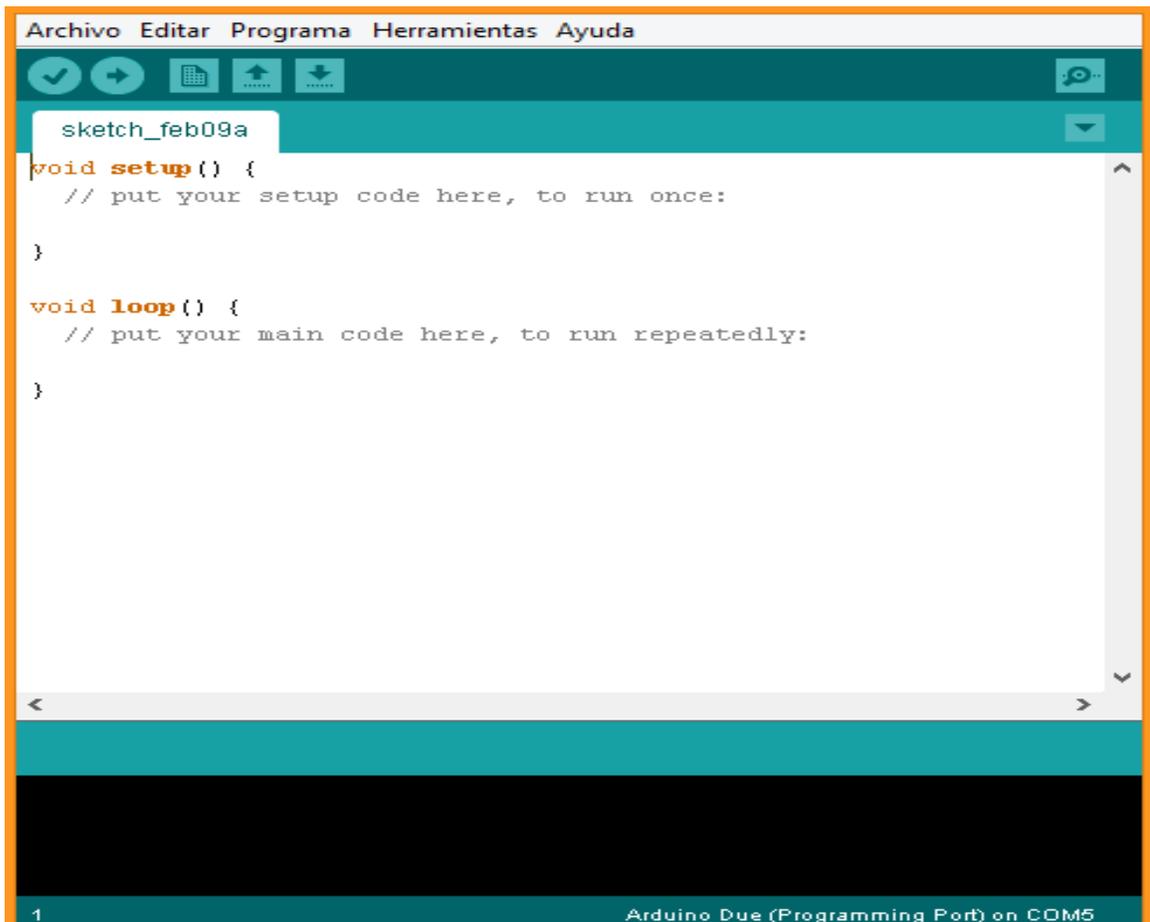


Figura. 16. Entorno de programación Arduino. Fuente: propia.

5. DESARROLLO INGENIERIL.

Para este proyecto de grado se desarrolló un sistema que permite la adquisición y visualización del nivel de presión sonora junto con variables climáticas haciendo uso de sistemas embebidos, por medio de tres etapas fundamentales descritas en la figura 17.

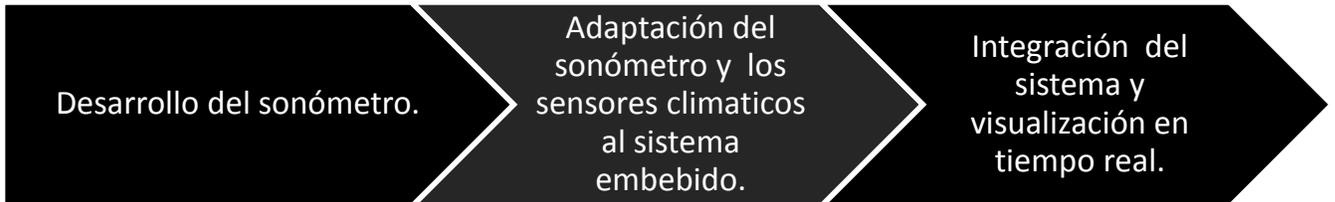


Figura. 17. Esquema de desarrollo del sistema. Fuente propia.

El sistema utiliza el protocolo de comunicación Serial para controlar el hardware/software de los sistemas embebidos y transmitir los datos de los sensores entre ellos. En la figura 18 se muestran las tres etapas del desarrollo, las secciones y los elementos correspondientes a cada una.

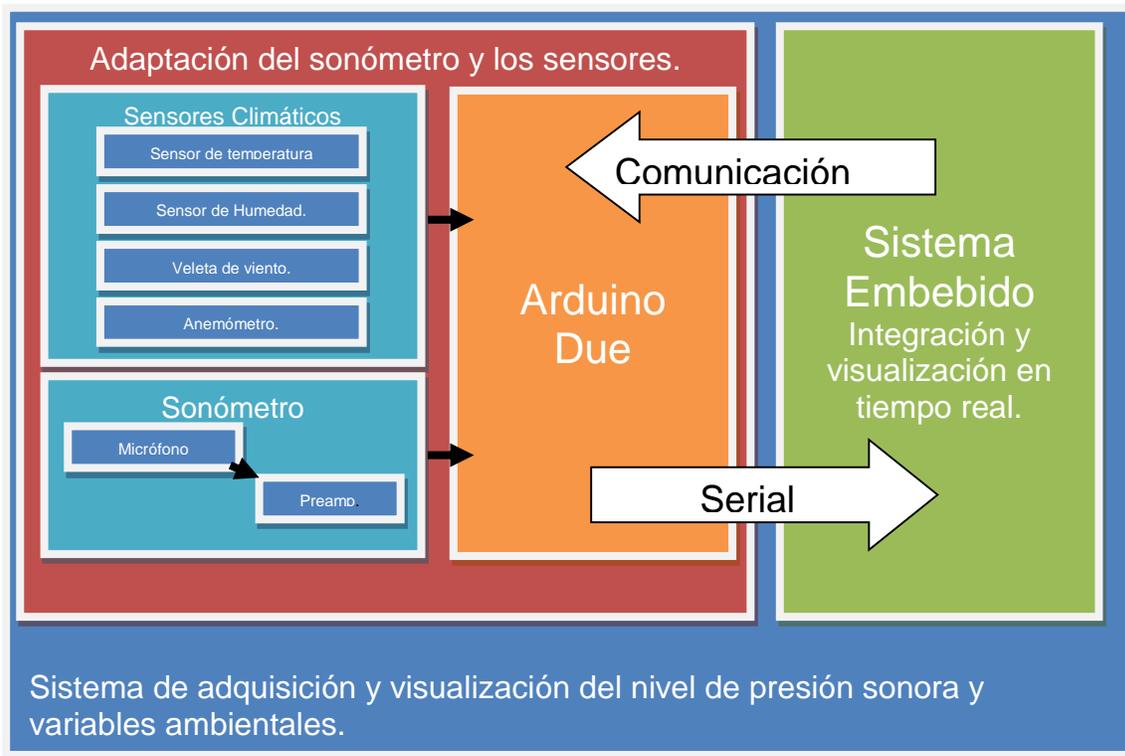


Figura. 18. Esquema general de las etapas de desarrollo del sistema de adquisición y visualización.

5.1. Diseño y desarrollo del sonómetro.

Para el diseño y desarrollo del sonómetro se toma como base las siguientes especificaciones plasmadas en la norma IEC61672-1, al momento de adquirir el nivel de presión sonora:

- El sonómetro pertenecerá al grupo Z, Sonómetros que requieren una o más piezas de equipo que constituyen partes esenciales del sonómetro **(Numeral 5.1.3) [17]**.
- Rango de frecuencia de 16Hz a 16KHz **(Numeral 5.5.7.Nota 1) [17]**.
- Ponderación Temporal. Slow, Fast, Impulsive **(Numeral 5.7.1) [17]**.
- El rango dinámico debe ser al menos de 60 dB a 1KHz **(Numeral 5.5.4) [17]**.
- La presentación de los resultados del nivel de presión sonora deben permitir mediciones con una resolución de 0,1dB en todo el rango de niveles, al menos 60dB **(Numeral 5.15.3) [17]**.
- Si se posee una salida analógica o digital se deben describir las características **(Numeral 5.16.1) [17]**.
- Influencia de las variaciones en las condiciones ambientales **(Numeral 9.2.7) [17]**.

El desarrollo del sonómetro está compuesto por dos etapas, la primera es una etapa de hardware donde se desarrolla el proceso de pre amplificación de las señales de audio y la segunda, una etapa de software donde se digitalizan las señales de audio y se transmiten desde el sistema Arduino Due.

La etapa de hardware se enfoca en el desarrollo de los elementos tangibles que modifican las señales analógicas generadas por el micrófono de medición, este hardware adecua las señales para ser digitalizadas posteriormente en la etapa de software.

Las especificaciones del sonómetro se muestran en la tabla 1.

Características del sonómetro	Valores
Frecuencia de muestreo.	50Khz
Resolución de conversión A/D.	12Bits
Sensibilidad del micrófono.	11.42 mV/Pa
Protocolo de comunicación.	Serial RS232
Rango de frecuencia.	15Hz-20KHz
Memoria.	8Gbytes
Rango dinámico.	60dB – 120dB
Sound meter level.	SPL, Leq, Max, Min
Ponderación.	Lineal.
Filtros digitales.	No
Filtros temporales.	Slow, Fast, Impulsive.

Tabla. 1.

Especificaciones del sonómetro.

5.1.1. Diseño y desarrollo del pre amplificador.

Los valores máximo y mínimo del rango dinámico se eligieron por ser valores centrales en el rango de audición de las personas, a partir de exposiciones continuas a 70 dB se produce un daño auditivo [31][4][3], por esta razón se deja un ajuste de -10 dB para tener un rango dinámico enfocado a los niveles que producen daños auditivos por altos tiempos de exposición (60dB a 120dB).

Los micrófonos son transductores que convierten energía acústica en energía eléctrica, esta transformación está relacionada por la sensibilidad del micrófono, para este proyecto se utiliza el micrófono de medición ECM8000 de Berinhger mostrado en la figura 19. Para hallar la sensibilidad del micrófono se implementa un calibrador acústico a 94dB de SPL lo que equivale a 1 pascal de presión, el micrófono se conecta a un preamplificador de prueba con ganancia fija y se mide la tensión de salida. El pre amplificador de prueba posee una ganancia de 23.72 V/V por defecto.



Figura. 19. ECM8000 Behringer [38].

El valor de tensión medido a la salida del pre amplificador de prueba al aplicar una señal de 94dB o 1 pascal, es de 0.271 voltios, la ganancia del pre amplificador es una relación entre el voltaje de entrada (V_o) y el voltaje de salida (V_s) como se indica en la ecuación 3.

$$G = \frac{V_s}{V_o}$$

Ecuación. 3. Expresión para calcular la ganancia de un pre amplificador [FALTA]

Se despeja el valor de V_o :

$$V_o = \frac{V_s}{G}$$

$$V_o = \frac{0.271v}{23.72}$$

$$V_o = 11.42mV$$

El voltaje de entrada que recibe el pre amplificador es el voltaje producido por el micrófono cuando se le aplica 1 pascal de presión, por esta razón la sensibilidad del micrófono ECM8000 es de 11.42 mV.

Las características del micrófono ECM8000 se muestran en la tabla 2.

Características del micrófono ECM8000	Valor
Patrón Polar.	Omnidireccional
Sensibilidad.	11.42mV
Respuesta en frecuencia.	15Hz – 20KHz
Impedancia.	600 Ohm

Tabla. 2. Características físicas del micrófono ECM8000.

Los niveles de salida de los micrófonos están en el orden de los mili voltios (mV), los cuales son bajos para poder analizarlos, debido a que el ruido eléctrico y de radio frecuencia que se presenta en los sistemas puede enmascarar las señales propias del micrófono lo cual hace necesario elevar el nivel de voltaje hasta un nivel idóneo para poder realizar análisis sin tener presente el ruido eléctrico. Los amplificadores instrumentales o INA, son amplificadores que se caracterizan por eliminar las señales comunes de ruido que se presentan en sus entradas [38], por esta razón se utiliza e implementa el diseño propuesto en la hoja de datos del amplificador INA128P (Figura 8).

La ganancia del amplificador INA128 está dada por la expresión de la ecuación 4.

$$Gain = 1 + \frac{50000}{Rg}$$

Ecuación. 4. Expresión para calcular la ganancia del amplificador INA217.

En el desarrollo del pre amplificador se ajustó un nivel de salida DC equivalente a 1.65V para poder ingresar al conversor análogo digital del Arduino Due, ya que este permite un rango de voltaje de entrada de 0 a 3.3 voltios. El pin número 5 del INA128P es el pin de Offset [38], el voltaje que se encuentre en este pin se sumará a la salida del pre amplificador. Se utiliza un divisor de voltaje con dos resistencias conectadas en serie a 5

voltios en un extremo y a tierra en el otro. La unión de las dos resistencias se conecta al pin 5.

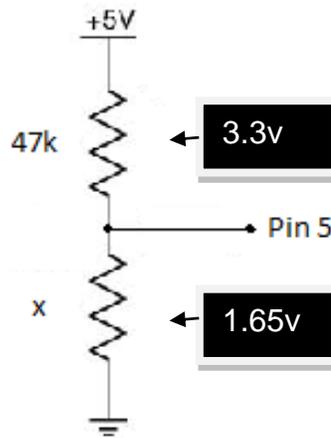


Figura. 20. Configuración de las resistencias de offset.

Para calcular el valor de la resistencia “x” como se muestra en la figura 20 se calcula la corriente que circula por las dos resistencias. Como la configuración es en serie las dos resistencias poseen la misma corriente según la ley de ohm como se muestra en la ecuación 5. El voltaje es igual al producto de la corriente por la resistencia.

$$V = IR$$

Ecuación. 5. Ley de Ohm.

Se realiza el cálculo correspondiente de la corriente que circula por las dos resistencias.

$$I = \frac{V}{R}$$

$$I = \frac{3.35}{1000}$$

$$I = 3.35mA$$

Finalmente, se calcula el valor de la resistencia.

$$Rx = \frac{1.65}{3.3mA}$$

$$Rx = 493 \text{ Ohm}$$

Para diseñar la ganancia del pre amplificador se debe tener en cuenta el rango de voltaje del conversor análogo digital y la sensibilidad del micrófono. El rango de voltaje del ADC es de 0 a 3.3 voltios y la sensibilidad del micrófono de 11.42mV.

El valor máximo de voltaje permitido por el ADC es de 1.65vp o 1.16Vrms y se asocia con el valor máximo del rango dinámico del sonómetro 120dB – 20 pascales. Por la sensibilidad del micrófono 20 pascales equivalen a 0.22 V, si se desea que 20 pascales sea el máximo valor de presión que capte el sonómetro, se amplifica con un factor “G” hasta llegar a 1.16Vrms.

$$1.16v = G \cdot 0.22v$$

$$G = \frac{1.16v}{0.22v}$$

$$G = 5.27v/v$$

La ganancia para el desarrollo del pre amplificador es de 5.27 V/V, seguido a esto se calcula la resistencia equivalente a esta ganancia. Para el diseño propuesto en la figura 8 y Según la ecuación 3, el valor de la resistencia de ganancia “Rg” es:

$$5.27 = 1 + \frac{50000}{Rg}$$

$$4.27 = \frac{50000}{Rg}$$

$$Rg = \frac{50000}{4.27}$$

$$Rg = 11.7KOhms$$

La implementación del pre amplificador se desarrolló en una tarjeta electrónica universal para realizar pruebas y calibrar el sonómetro. (Ver figura 21).

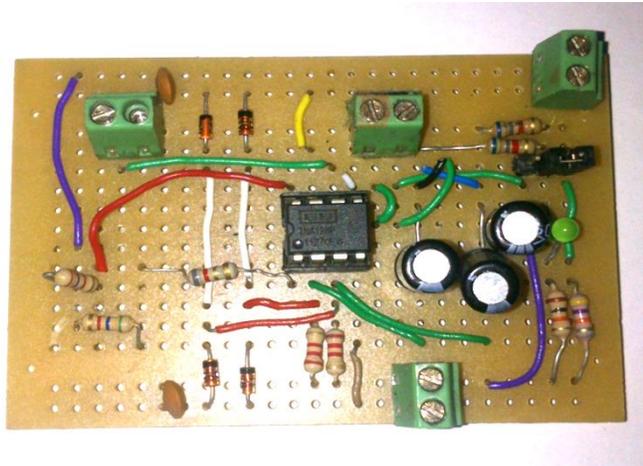


Figura. 21. Prototipo del pre amplificador desarrollado para realizar pruebas primarias. Fuente propia,

La figura 22 representa el diseño del circuito impreso en el software Eagle y en la figura 23 se evidencia el pre amplificador en la etapa final de desarrollo.

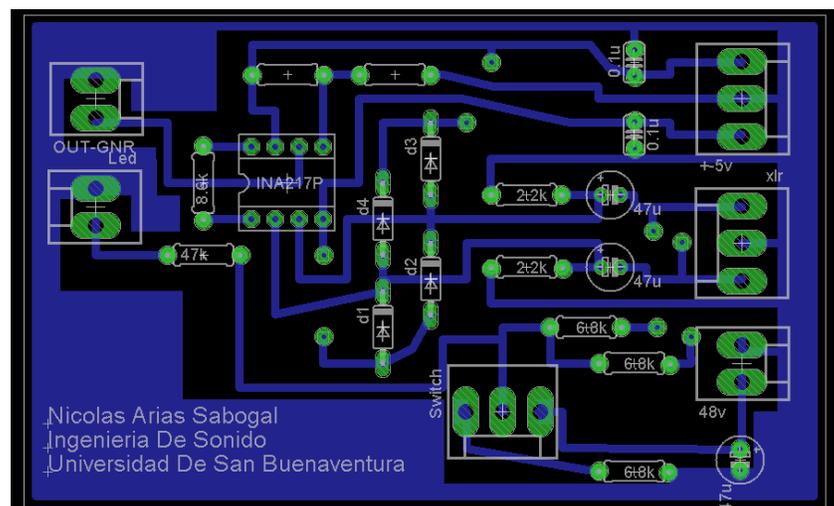


Figura. 22. Diseño del impreso del pre amplificador. Fuente: Propia



Figura. 23. Etapa final de desarrollo del pre amplificador. Fuente: Propia

5.1.1.1. Calibración del sonómetro.

Para calibrar el sonómetro es indispensable utilizar un pistofono (calibrador acústico que genera un tono de 1KHz a 114dB) y medir la tención de salida después de la etapa de pre amplificación, así se puede obtener una relación entre presión y voltaje como se muestra en la figura 24.

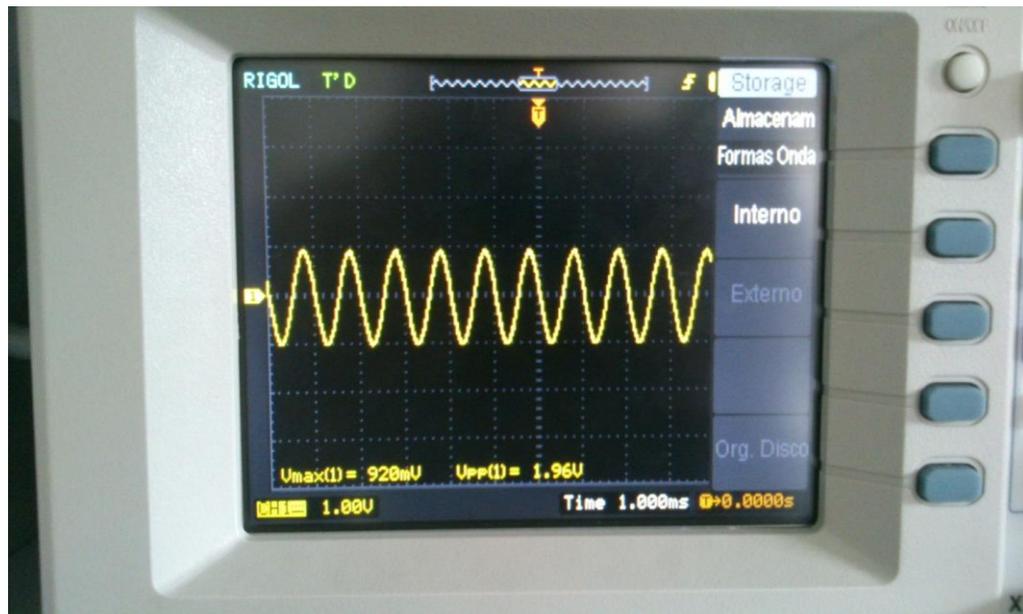


Figura. 24. Calibración del micrófono a 1KHz con 114dB de presión acústica.

Para hallar el valor de referencia se calcula el valor RMS utilizando la ecuación número 6.

$$VRMS = VPICO \times 0.707.$$

Ecuación. 6. Cálculo del valor RMS para una señal sinusoidal [1].

El valor pico de la señal que se visualiza en la figura 36, es la mitad del voltaje pico a pico (vpp) por lo cual para un nivel de presión sonora de 114 dB el voltaje de calibración RMS es equivalente a 0.69 voltios después de la etapa de pre amplificación.

$$VRMS = \frac{1.96}{2} * 0.707$$

$$VRMS = 0.98 * 0.707$$

$$VRMS = 0.692 \text{ voltios}$$

5.1.2. Diseño y desarrollo del software de control para el sonómetro.

- **Desarrollo del software de digitalización y adquisición del nivel de presión sonora en Arduino 1.5.**

La etapa de software tiene como función digitalizar, procesar y transmitir las señales analógicas producidas por el micrófono de medición.

El desarrollo de esta etapa utilizando el entorno de programación Arduino 1.5 se ejemplifica en la figura 25.

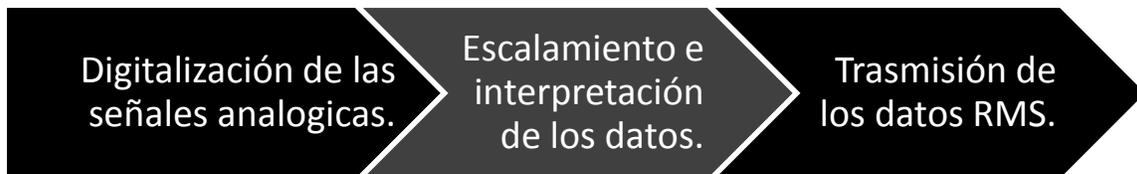


Figura. 25. Etapas de desarrollo para el software del sonómetro.

- **Digitalización de las señales analógicas.**

Para digitalizar las señales analógicas provenientes del micrófono, se utilizó uno de los conversores análogo – digital del Arduino Due, en este caso el conversor número 1.

Este conversor por defecto tiene una velocidad de muestreo de 15KHz [33], esta velocidad no es la indicada para realizar la digitalización según la norma IEC 61672-1 [17], por esta razón se modifican los registros internos del conversor utilizando el código brindado por el fabricante que se muestra en la figura 26.

```
//Configuramos el conversor analogo digital 0 y 1 para utilizar la velocidad maxima de muestreo
ADC->ADC_MR |= 0x80;
ADC->ADC_CR=2;
ADC->ADC_CHER=0xC0;
```

Figura. 26. Código para activar la velocidad de muestreo máxima de Arduino Due.

Estos registros activan la velocidad máxima de muestreo a 50KHz, velocidad indicada para realizar este proceso de digitalización.

Una vez que se establece la velocidad máxima de muestreo se realizan las lecturas en el conversor. Estas lecturas dependen del tiempo de integración seleccionado por el usuario. Según la norma IEC 61672-1 los tiempos de integración para un sonómetro clase II deben ser Slow, Fast e Impulsive [17], por lo cual se configuran los 3 intervalos de medición con respecto a los 3 tiempos de integración según la elección del usuario. (Ver figura 27).

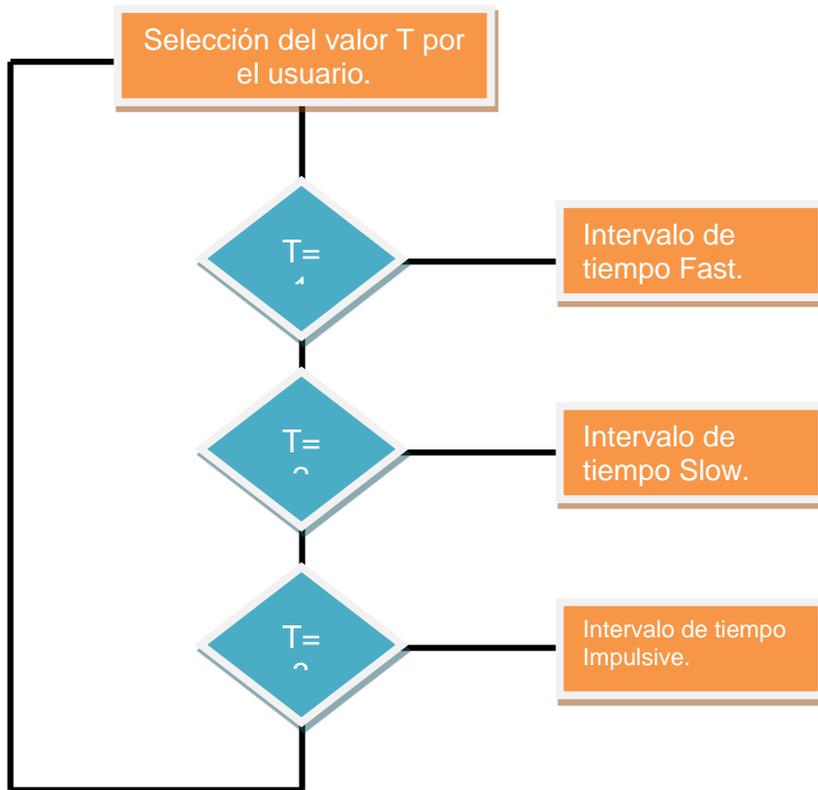


Figura. 27. Diagrama de flujo para seleccionar el tiempo de integración del sonómetro.

- **Escalamiento e interpretación de los datos.**

Como se mencionó anteriormente el rango de funcionamiento del conversor es de 0 a 3.3 voltios, lo que equivale a 4095 valores diferentes en una resolución de 12 Bits. Para representar los valores digitalizados en valores de voltaje se utiliza una relación de conversión.

$$\text{Valor representado en voltios} = (3.3 * \text{valor de medición}) / 4095.$$

Para obtener la representación original de las señales que provienen del micrófono es necesario discriminar el valor del nivel DC en cada una de las mediciones, por esta razón restamos 1.65 al valor de la medición.

$$\text{Valor representado en voltios} = ((3.3 * \text{valor de medición}) / 4095) - 1.65.$$

Después de obtener los valores representados en voltaje, se procede a calcular el nivel representativo de la señal.

$$x_{\text{RMS}} = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} = \sqrt{\frac{x_1^2 + x_2^2 + \dots + x_N^2}{N}}$$

Ecuación. 7. Expresión para el cálculo del valor RMS. [1].

En la ecuación 7, los valores de x representan los valores escalados de las mediciones y N indica el número de mediciones realizadas en el intervalo de tiempo seleccionado por el usuario, la figura 28 ejemplifica el procedimiento para calcular el valor RMS de la medición.

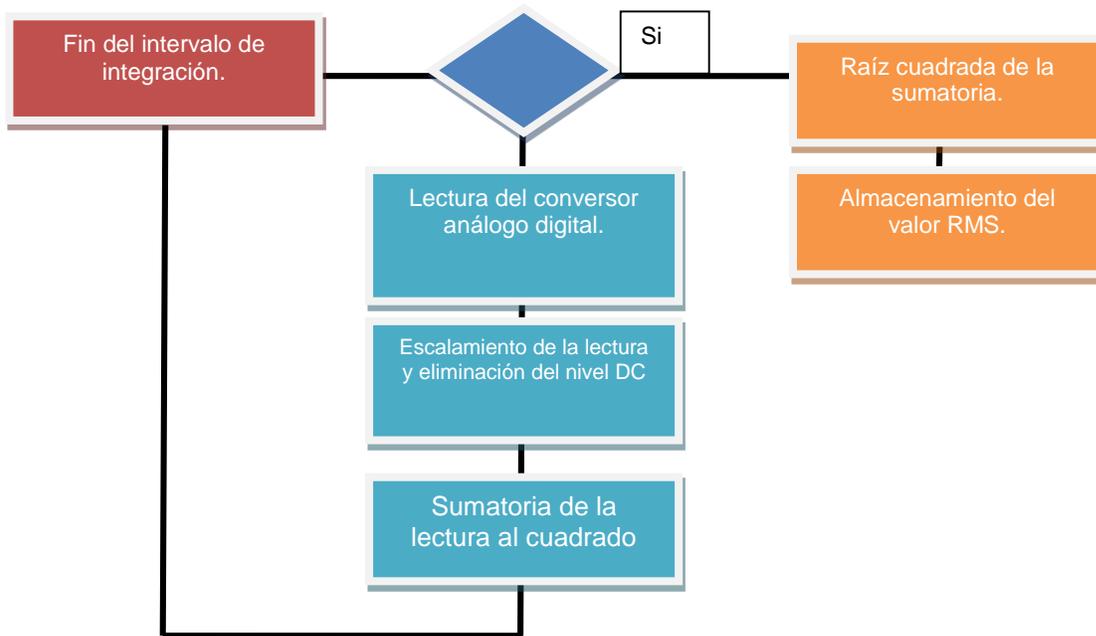


Figura. 28. Cálculo del valor RMS del sonómetro.

- **Transmisión de los datos.**

La transmisión de los datos se realiza por medio del protocolo serial a una velocidad de 115200 baudios (como muestra la figura 29), desde el Arduino Due hacia el sistema embebido mini210s, los datos que se transmiten son los valores RMS de las mediciones, estos datos son enviados periódicamente según el tiempo de integración seleccionado por el usuario.

```
//Definimos la velocidad de transmision Serial  
Serial.begin(115200);  
Serial.println(spl,3);
```

Figura. 29. Código de envío del valor RMS por comunicación serial en el Arduino Due.

5.2. Adaptación del sonómetro y los sensores climáticos, al sistema embebido.

Para realizar la adaptación del sonómetro y los sensores es indispensable unificar estos elementos tanto en hardware como en software, las figura 30 muestra las etapas de este proceso.

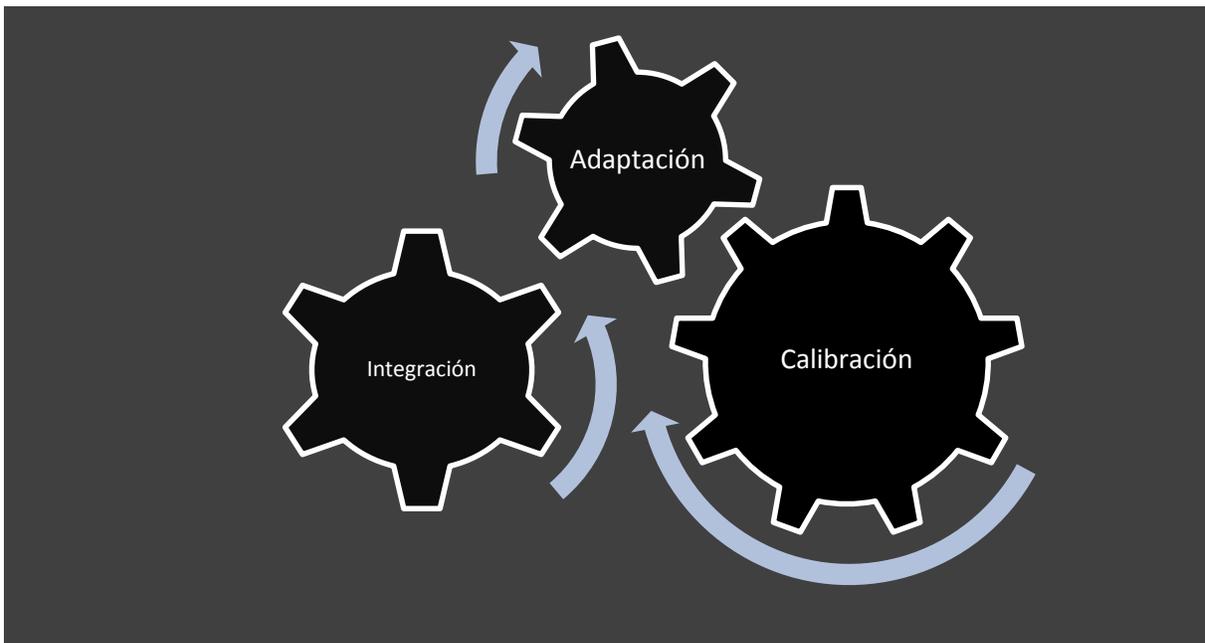


Figura. 30. Etapas de la adaptación del sonómetro y los sensores climáticos.

5.2.1. Calibración de los sensores climáticos.

Para analizar las variables climáticas relacionadas con la contaminación auditiva, son necesarios los sensores correspondientes a cada una de estas, y es necesario que los sensores estén calibrados para que los datos que entreguen sean acertados y sin errores para que no se afecte la medición al igual que la estación VAISALA [39] los sensores climáticos poseen funciones de transferencia y sentencias propias para realizar la calibración.

- **Sensor de Temperatura y humedad relativa**

En la figura 31 se muestra el sensor digital unificado de temperatura y humedad relativa “SHT11” utilizado en este proyecto, el cual cumple con las siguientes características [36]:

- Salida de datos digital (i2c – Protocolo de comunicación consta de un canal de datos y otro canal de sincronismo).
- Bajo consumo de potencia.
- Excelente estabilidad térmica.

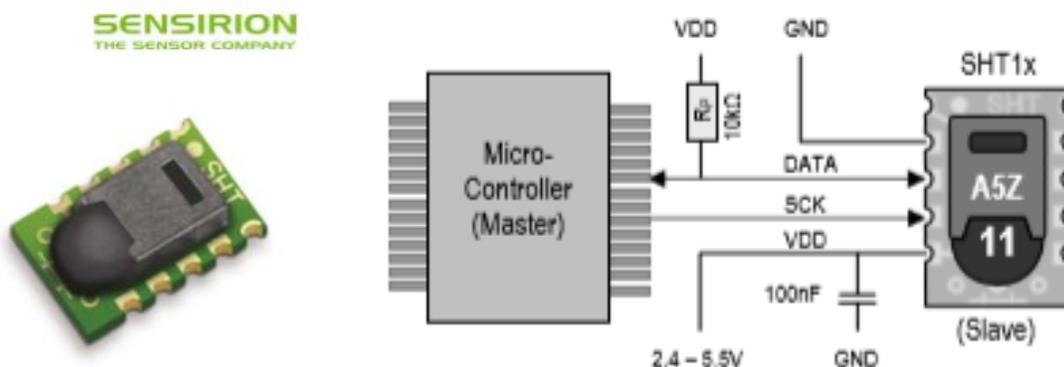


Figura. 31. Sensor de temperatura y humedad relativa SHT11 [36]

Para calibrar el sensor se debe tener en cuenta la función de transferencia brindada por el fabricante al momento de obtener los datos del sensor, estas ecuaciones son empleadas

en la librería “TEMPHUME” la cual viene incorporada con el entorno de programación Arduino.

- **Veleta y anemómetro**

Como se muestra en la figura 32, la veleta de viento implementada en el proyecto es una veleta analógica, que funciona como una resistencia variable con respecto a la posición que presente, lo cual indica que los datos de salida deben ingresar a un conversor análogo digital del Arduino Due.

La calibración de esta veleta debe hacerse al momento de implementar el sistema, de manera que la punta se dirija hacia el norte.



Figura. 32. Veleta de viento analógica. Fuente propia

El anemómetro que se muestra en la figura 33 es el “#40R de ARPS World”, el cual es un dispositivo que genera pulsos a una frecuencia directamente proporcional a la velocidad del viento, al igual que el sensor de temperatura y humedad. El fabricante de

este anemómetro provee una función de transferencia que se debe tener en cuenta al momento de interpretar los datos.



Figura. 33. #40R Anemometer [37]

5.2.2. Integración del sonómetro y los sensores climáticos.

Al momento de unificar los sensores climáticos y el sonómetro es necesario hacer un acople de hardware por medio de una tarjeta de unificación y un acople de software por medio del entorno de programación Arduino.

- **Integración en Hardware.**

Para realizar una conexión cómoda de los sensores y el micrófono al sistema Arduino, fue necesario diseñar una tarjeta de acople, que cumpliera la función de interconectar los diferentes tipos de sensores, de tal manera que la manipulación e interacción fuera de una manera funcional. Se realizó un prototipo de esta tarjeta para realizar pruebas del sistema y analizar el comportamiento que tenía al momento de hacer las conexiones. Esta tarjeta se puede visualizar en la figura 34.

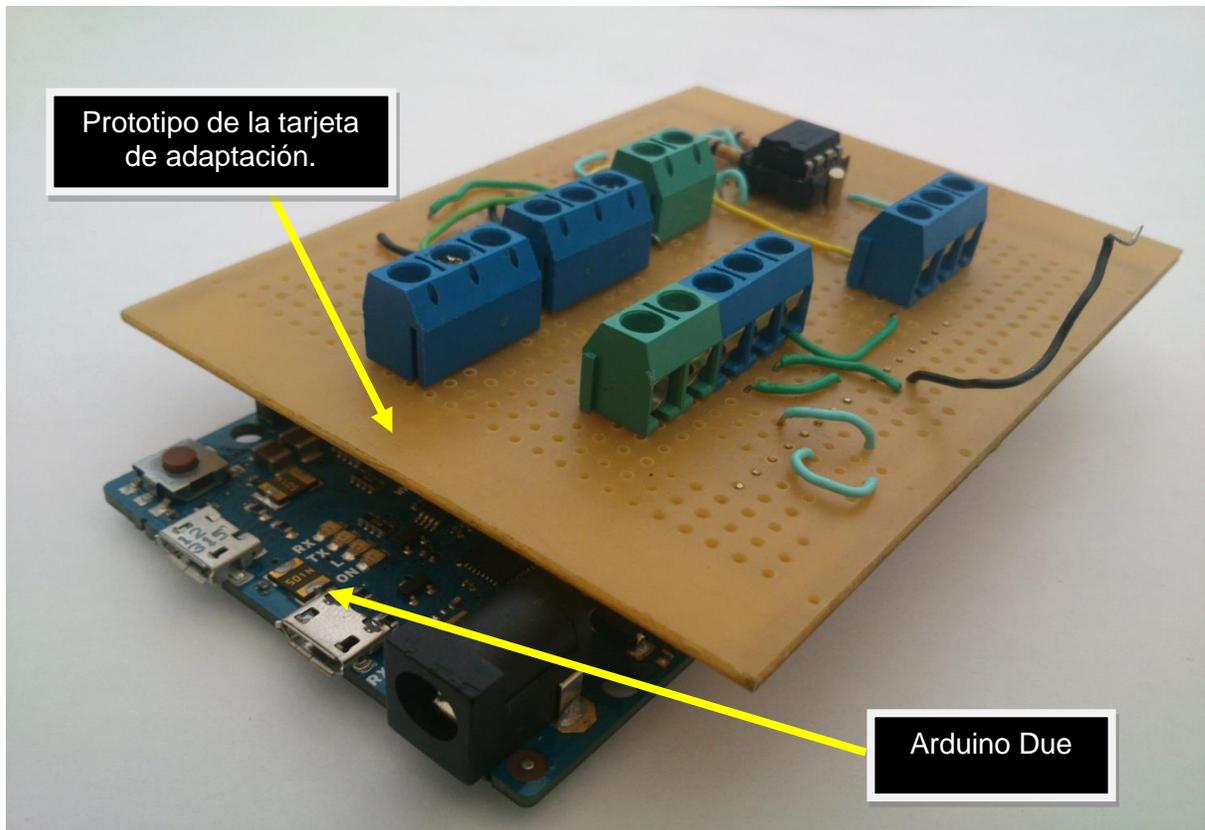


Figura. 34. Prototipo de tarjeta de interconexión compatible con el Arduino Due Fuente propia. La figura 35 muestra el diseño del impreso para la tarjeta de adaptación mediante el software Eagle.

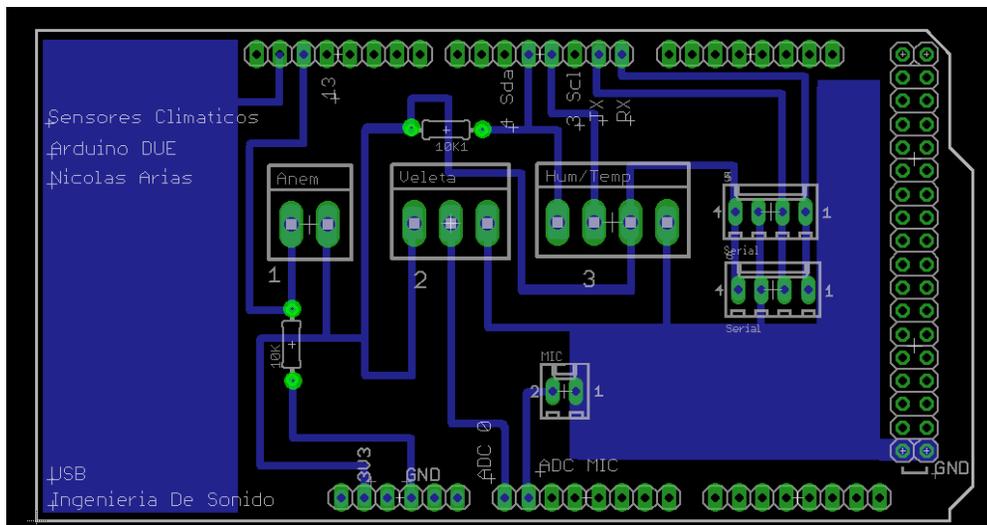


Figura. 35. Tarjeta de acople de sensores para Arduino DUE Fuente: propia.

Una vez diseñado y desarrollado el circuito impreso, se implementa en el Arduino Due. Esta tarjeta se conecta directamente en el Arduino como se muestra en la figura 34, lo cual permite la interconexión sin necesidad de cables. La imagen de la figura 36 muestra la tarjeta de acople totalmente terminada.

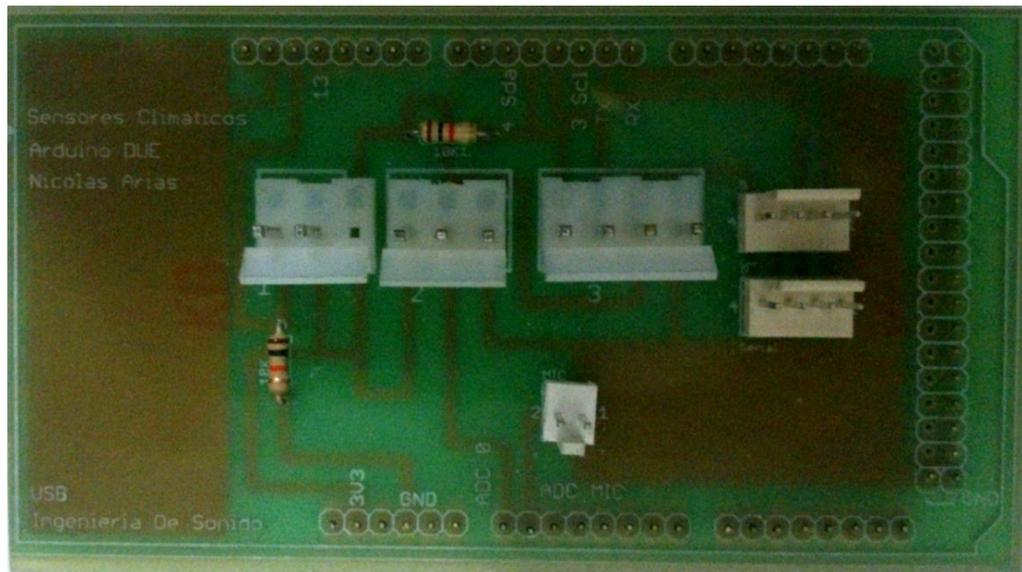


Figura. 36. Tarjeta de acople para Arduino Due.

La ubicación de los sensores en los pines del Arduino Due fue la siguiente:

- El sensor SHT11 de humedad y temperatura, posee 4 pines de conexión 2 son de datos y 2 de alimentación, los dos de datos se conectaron en el pin 3 y 4 (Clock y Data) respectivamente en el Arduino.
- La veleta de viento se conectó a la alimentación y al pin análogo 0 del Arduino.
- El anemómetro se conectó al pin digital 13 del Arduino.
- El micrófono del sonómetro se conectó al pin análogo 1 del Arduino.

- **Integración en Software.**

El entorno de programación Arduino, es una utilidad que facilita la interacción entre el programador y el sistema Arduino Due.

Para recibir los datos de los sensores fueron necesarias algunas librerías para Arduino Due, como la librería “TEMPHUME” [33] y “METRO” [33] estas librerías son de código abierto y libre para cualquier implementación. [8].

El código de integración está conformado por 2 secciones como se muestra en la figura 37.



Figura. 37. Secciones de la integración en software.

- **Lectura de los datos.**

Para la lectura de los datos que provienen del sensor de humedad y temperatura se utilizó la librería “TEMPHUME” disponible en el entorno de programación. Esta librería permite la comunicación directa del Arduino Due y el sensor con 2 funciones específicas, lectura de temperatura y lectura de humedad.

La figura 38 muestra el proceso que se empleó para obtener los datos de humedad y temperatura.

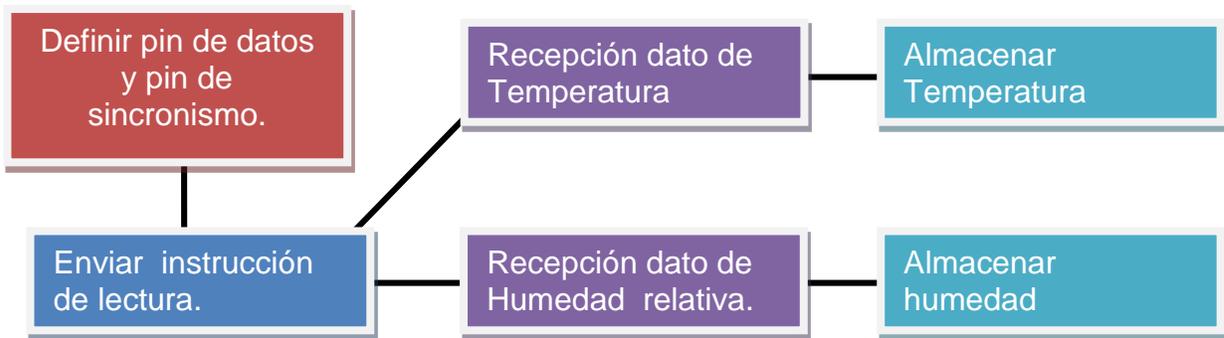


Figura. 38. Procedimiento para la lectura del sensor SHT11.

Para obtener los datos de la veleta se utiliza el conversor análogo digital 0 del Arduino Due, el funcionamiento de la veleta es similar al de una resistencia variable. A medida que la veleta cambia su posición directamente cambia la resistencia interna del sensor, lo que indica un cambio de voltaje el cual es leído por el conversor.

Este valor de lectura se escala de 0 a 365, siendo 0 – 0° y 4095 – 365°, (0 a 4095, valores de representación digital de un rango de voltaje análogo de 0 a 3.3V con una resolución de conversión de 12bits) y se almacena. La figura 39 muestra el proceso que se utiliza para la lectura de la dirección del viento.

```

//Funcion para la adquisicion del valor de direccion del viento
void Direccion()
{
gra = ADC->ADC_CDR[7]; //Lectura del conversor analogo digital "0"
gra = (gra*365)/4095; // Escalamiento de la señal rango de 0° a 365°
}
  
```

Figura. 39. Código para obtener el valor de dirección del viento.

Para obtener el valor de velocidad del viento es necesario implementar en el desarrollo del código la librería “METRO” disponible en el entorno Arduino, esta librería permite realizar operaciones y funciones en un intervalo de tiempo específico.

El anemómetro genera pulsos rectangulares de voltaje proporcionales a la velocidad del viento, para calcular el valor de velocidad se utiliza la función de transferencia brindada por el fabricante donde indica que es necesario identificar la frecuencia de esta señal, por lo cual se realiza una cuenta de los pulsos producidos por el anemómetro en un intervalo de un segundo.

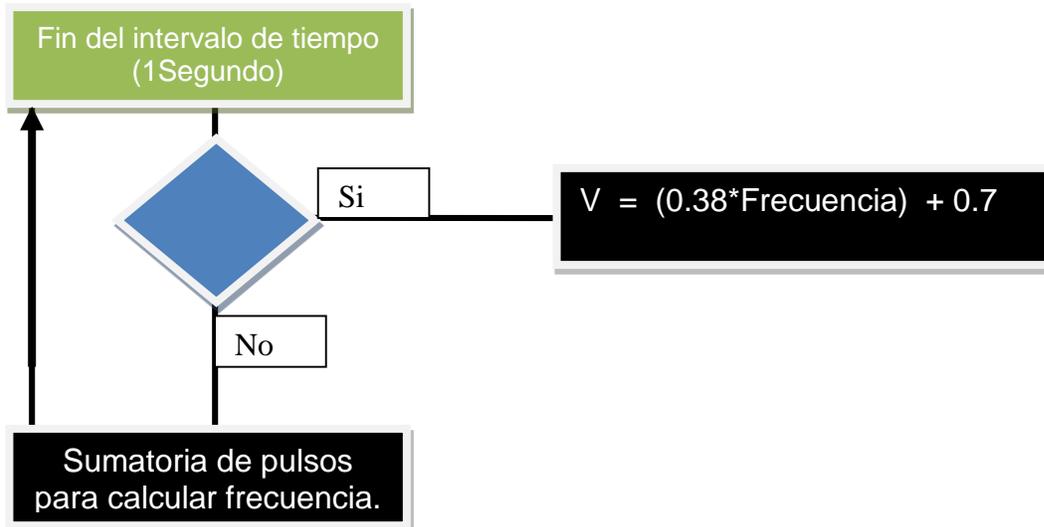


Figura. 40. Procedimiento para obtener el valor de la velocidad de viento.

Como se muestra en la figura 40, se almacena el valor de velocidad de viento para así finalizar con la lectura de todos los sensores y proceder a la parte de transmisión.

- **Transmisión de los datos.**

Al momento de calcular y almacenar las 5 variables climáticas, temperatura, humedad, dirección del viento, velocidad del viento y nivel RMS, se procede a realizar una trama de datos para él envío por medio del protocolo serial, esta comienza, divide y termina con un carácter de sincronismo “/”.

/ DireccionViento / VelocidadViendo / Temperatura / Humedad / NivelvoltajeRMS/

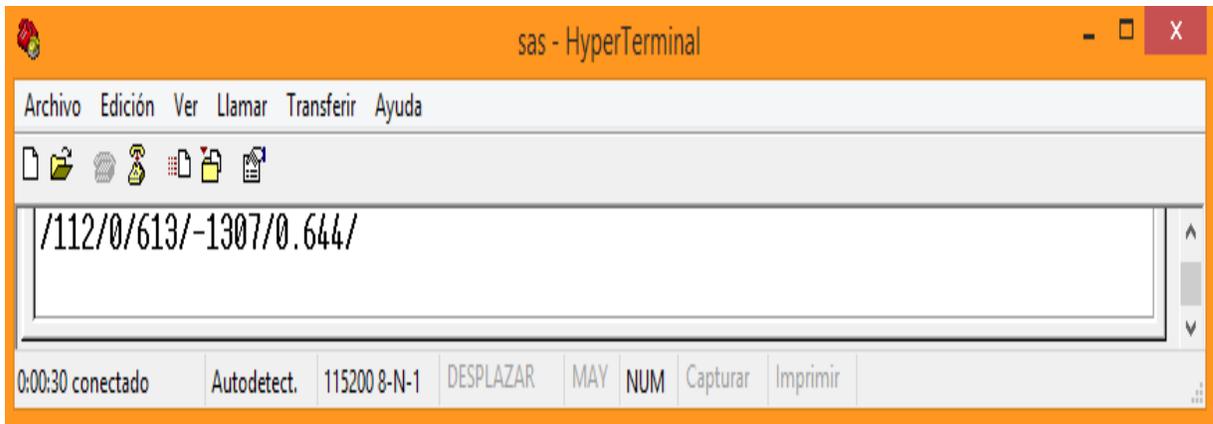


Figura. 41. Envío de la trama de datos desde el Arduino Due. Fuente: propia

En la figura 41 se evidencia la integración de los datos en una sola trama, el sistema total permite visualizar los datos en tiempo real, y realizar mediciones del nivel de presión sonora con tres diferentes tiempos de integración, por lo cual el Arduino Due se programó para 4 diferentes modalidades de envío.

- Monitoreo: para esta modalidad el Arduino Due envía la trama de datos con todas las variables climáticas incluyendo el nivel de presión sonora.
- Medición Slow: esta modalidad envía datos del nivel de presión sonora al sistema embebido cada segundo.
- Medición Fast: esta modalidad envía datos del nivel de presión sonora al sistema embebido cada 125 mili-segundos.
- Medición Impulsive: esta modalidad envía datos del nivel de presión sonora al sistema embebido cada 32 mili-segundos.

Estas modalidades son seleccionadas por el usuario, en la interfaz del sistema embebido, utilizando el protocolo serial se envía "0", "1", "2", "3" de acuerdo a la modalidad que se desea respectivamente.

5.2.3. Adaptación al sistema embebido.

La comunicación serial es el transporte de la información y a su vez es el puente que permite la interacción del Arduino Due y el sistema embebido. Para realizar la conexión física de los dos dispositivos es necesario utilizar un instrumento que permite acoplar las señales seriales. En la figura 42 se muestra el dispositivo de interconexión.

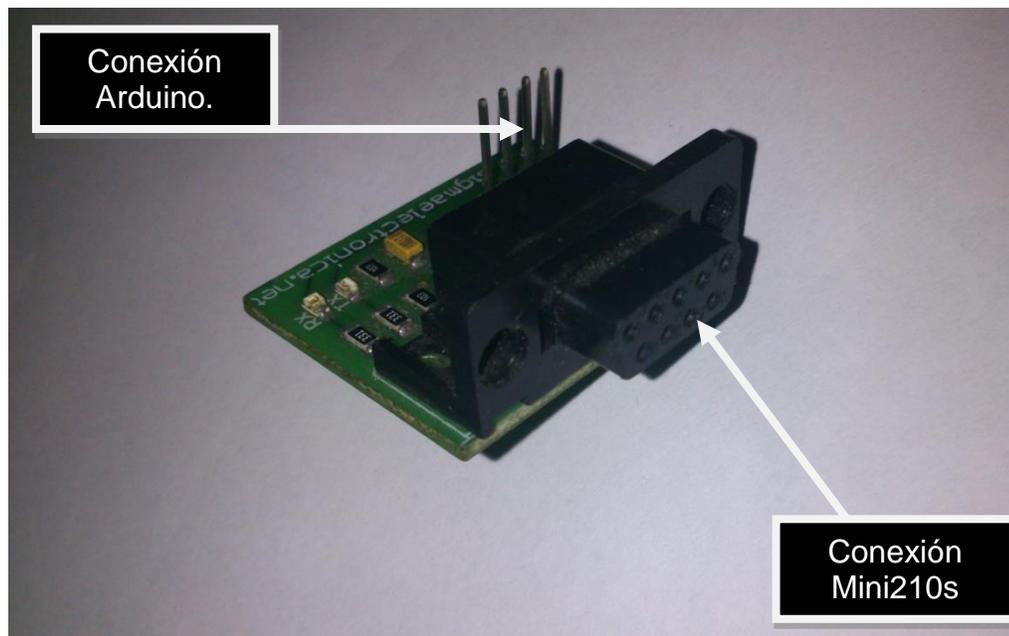


Figura. 42. Dispositivo de interconexión serial.

5.3. Integración del sistema y visualización en tiempo real.

La integración del sistema y la visualización de los datos se realizan desde el sistema embebido con ayuda de visual basic, para efectuar este proceso el sistema embebido debe estar adecuado con el sistema operativo Windows CE como indica el fabricante, los pasos para la adecuación del sistema están descritos en el anexo A.

La interfaz de control del sistema tiene integrado los datos de los sensores y los datos del sonómetro, y los transmite por medio del protocolo serial como se mencionó anteriormente.

La interfaz de control se divide en tres etapas, como se muestra en la figura 43.



Figura. 43. Etapas de la interfaz de usuario.

La interfaz de usuario es diseñada y desarrollada en visual basic 2005, debido a su compatibilidad directa para realizar aplicaciones en Windows CE.

La interfaz de usuario tendrá tres ventanas o se dividirá en tres secciones,

- Ventana Principal.
- Ventana de monitorización.
- Ventana de Medición del sonómetro.

La primera es la ventana principal donde se podrá elegir la modalidad de funcionamiento del sistema y la presentación de la aplicación.



Figura. 44. Ventana principal de la interfaz de usuario. Fuente: propia.

Como se muestra en la figura 44 esta interfaz posee dos botones que habilitan las funciones de la aplicación, el botón “Visualización” habilita la interfaz de monitoreo y el botón “Medición” habilita la interfaz del sonómetro.

- **Desarrollo interfaz del sonómetro**

La interfaz del sonómetro permite medir en un tiempo específico y con un tiempo de integración determinado, el nivel de presión sonora del lugar donde está ubicado el sistema.

También permite visualizar y almacenar el nivel continuo equivalente (LEQ), el nivel máximo RMS y el nivel mínimo RMS, del tiempo total de medición.

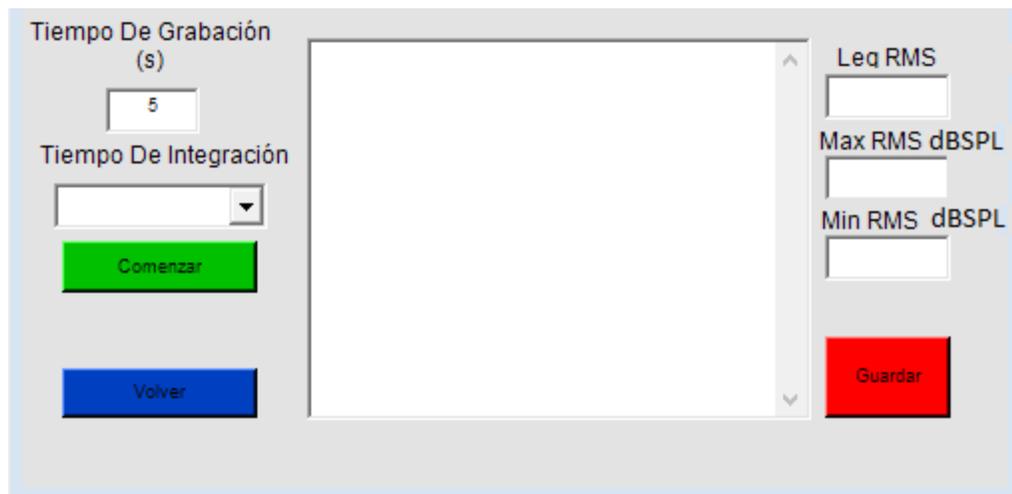


Figura. 45. Interfaz del sonómetro. Fuente: propia.

En la figura 45 se evidencia que la interfaz del sonómetro está dividida en dos secciones la primera es una sección de configuración y conexión y la segunda una sección de visualización de los datos obtenidos.

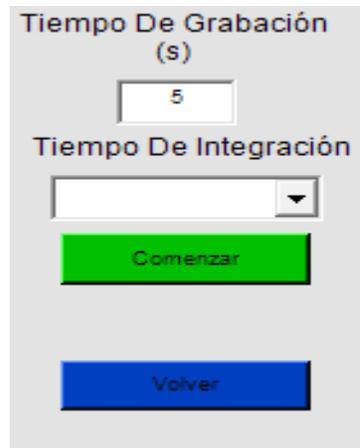


Figura. 46. Sección de configuración del sonómetro. Fuente: propia.

En la figura 46 se muestra el contenido de la sección de conexión. El botón “comenzar” tiene como función habilitar la comunicación serial con el Arduino Due para enviar un mensaje de configuración. Este mensaje debe ser 0, 1, 2, 3 o 4, para seleccionar una de las modalidades de trabajo del sistema. Para el caso de la interfaz del sonómetro se utiliza únicamente 1, 2 y 3, según el tiempo de integración seleccionado (1 – Slow, 2-Fast, 3-Impulsive) como se muestra en la figura 47.

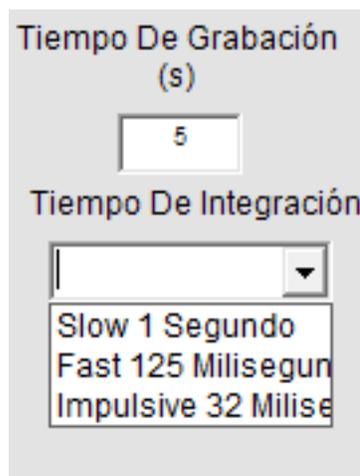


Figura. 47. Menú desplegable para el tiempo de integración. Fuente: propia.

Una vez que se selecciona la modalidad y se envía el mensaje de configuración se procede a recibir los datos correspondientes al sonómetro.

Los datos que envía el Arduino Due son valores de voltaje RMS, por esta razón se deben representar en niveles de presión sonora, para esto se utiliza la ecuación 1.

La figura 48 muestra el procedimiento interno de la interfaz del sonómetro para calcular el nivel de presión sonora, una vez el usuario haya estipulado el tiempo de medición y el tiempo de integración.

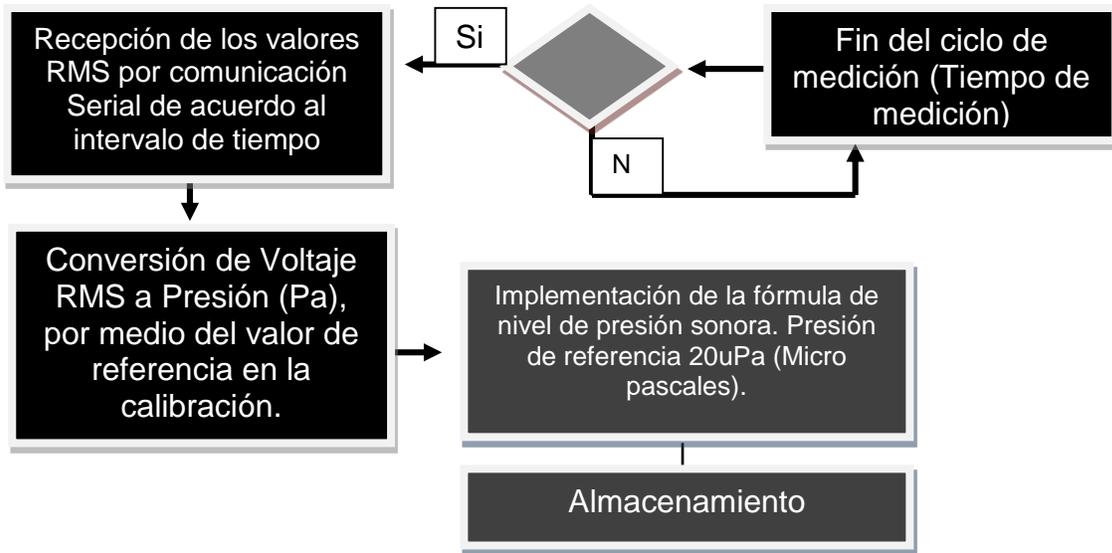


Figura. 48. Proceso interno del botón Comenzar.

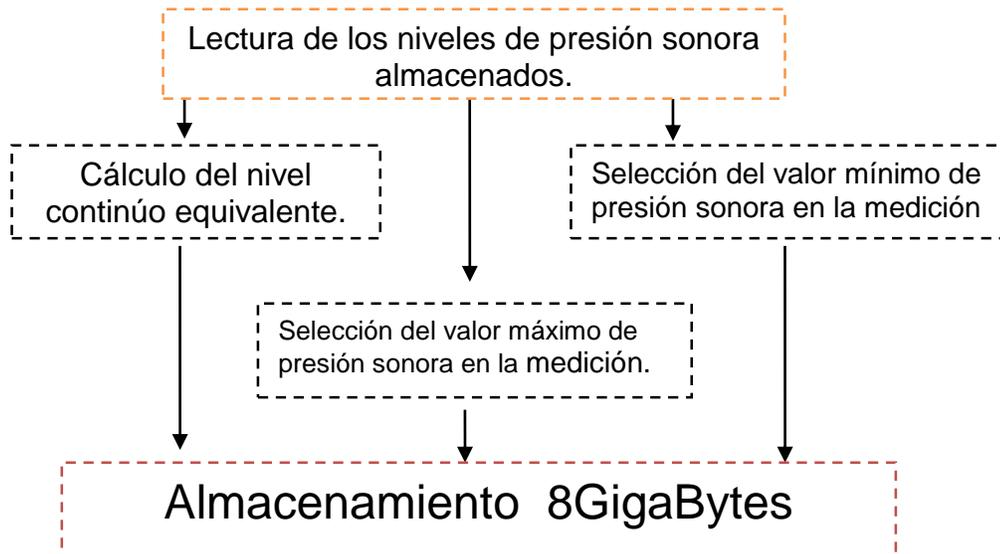


Figura. 49. Proceso interno de botón "resultados".

La figura 49 indica el procedimiento para calcular el nivel continuo equivalente basándose en la expresión de la ecuación 2. En esta sección también se obtiene el valor máximo y

mínimo de la medición y por último se almacenan los datos en una memoria con una capacidad de 8Gbytes.

La siguiente sección de la interfaz del sonómetro es la sección de visualización, donde se imprimen los valores de los niveles obtenidos en las casillas correspondientes como indica la figura 50y 57.

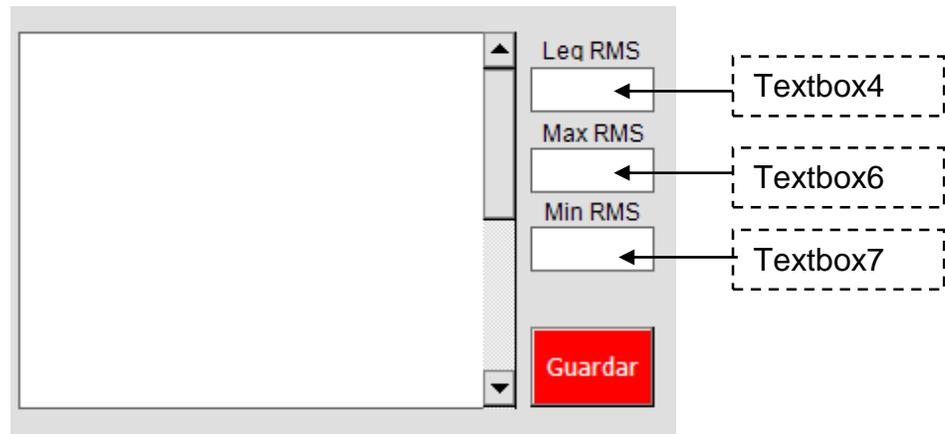


Figura. 50. Sección de visualización en la interfaz del sonómetro. Fuente: propia.

```

TextBox6.Text = Format(max, "##0.00")
TextBox7.Text = Format(min, "##0.00")
TextBox4.Text = Format(leq, "##0.00")
    
```

Figura. 51. Implementación del código para visualizar los datos en las cajas de texto.

El botón de almacenar guarda los registros y resultados de la medición en un archivo de texto ubicado en una ruta específica dentro del sistema embebido.

- **Desarrollo interfaz de monitoreo**

La interfaz de monitoreo permite visualizar los datos del clima y el nivel de presión sonora en tiempo real de dos formas, la primera muestra los valores numéricos de las variables climáticas y la segunda representa estos valores en gráficas.



Figura. 52. Interfaz de monitoreo. Fuente: propia.

El botón de “conectar” que se muestra en la figura 52 tiene la función de habilitar la comunicación serial con el Arduino Due y enviar un mensaje de configuración. Para configurar el Arduino en la modalidad de monitoreo, es necesario enviar como mensaje de configuración el número cero “0”.El proceso de lectura de datos enviados por el Arduino Due se presenta en la figura 53.

El botón “gráficas” permite cambiar la configuración de visualización, a un entorno grafico donde las señales climáticas y de ruido se presentan en cajas gráficas, como muestra la figura 54.

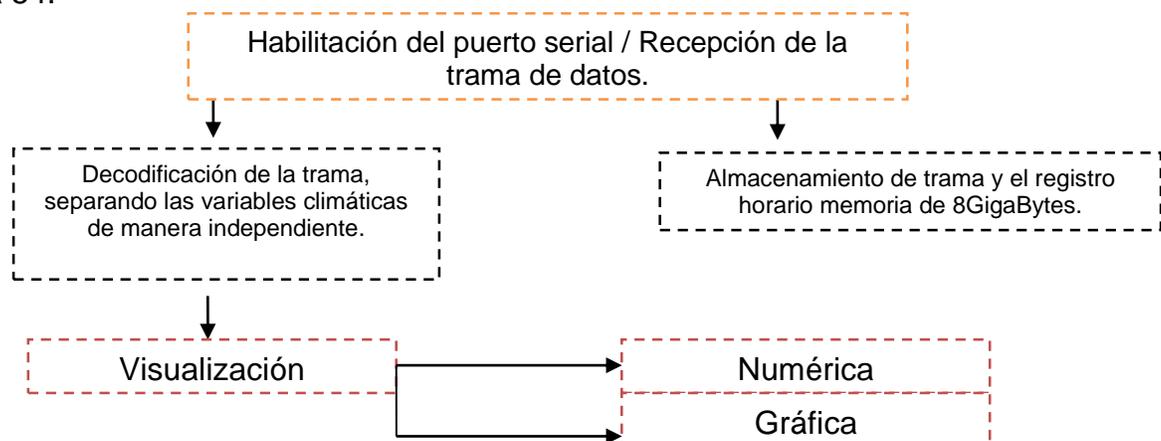


Figura. 53. Proceso de lectura y decodificación de la trama de datos.



Figura. 54. Ventana de visualización de las gráficas en tiempo real. Fuente: propia.

5.4. Medición de la latencia del sistema.

La latencia que presenta el sistema desde el momento que se capturan los datos hasta que los recibe el sistema embebido, se mide como se evidencia en la figura 55.



Figura. 55. Etapas de medición de la latencia del sistema.

Para la primera etapa se implementó la función de arduino “`Millis ()`” y “`Micros ()`” las cuales indican el tiempo en mili segundos y micro segundos, que ha transcurrido desde el momento en que se encendió el Arduino Due, hasta implementar la función. Con estas funciones se calcula cual es el tiempo que se emplea para recolectar los datos. Estas mediciones se realizan de manera independiente para cada una de las variables del sistema. La tabla 3 presenta los tiempos de latencia que presenta la recolección de los datos.

En la tabla 3 se muestran los tiempos que se demora el arduino recolectando los datos en el respectivo orden de la trama de datos, el valor siguiente es el tiempo que demora el arduino en armar la trama de datos y empezar a enviarla.

/ DireccionViento / VelocidadViendo / Temperatura / Humedad / NivelvoltajeRMS/

```
/Tiempo Gra= 3/Tiempo vel= 0/Tiempo tem= 390/Tiempo hum= 620/Tiempo spl= 28/1561
```

Operación	Tiempo de demora
Lectura Dirección del viento	3 Micro Segundos.
Lectura Velocidad del viento	0 Micro Segundos (Menor a 0 microsegundos).
Lectura de Temperatura	390 Micro Segundos.
Lectura Humedad Relativa	620 Micro Segundos.
Ensamble de la trama	1.56 Mili Segundos.

Tabla. 3. Tiempos de ejecución de las operaciones internas del arduino.

En la segunda etapa para calcular el tiempo que se demora en enviar los datos el arduino y recibirlos en el sistema embebido se debe tener en cuenta la longitud de la trama de datos para la modalidad de visualización la trama tiene 22 caracteres y para la modalidad de medición la trama tiene 9 caracteres. Los caracteres que se transmiten son caracteres ASCII de 8Bits, lo que indica que para la modalidad de visualización la trama tiene un tamaño de 176Bits y para la modalidad de medición la trama es de 72Bits.

La velocidad de transmisión serial es de 115200Baudios (Bits por segundo) para calcular el tiempo de trasmisión se relaciona la velocidad de transmisión con el tamaño de trasmisión, en el caso de la modalidad visualización el tiempo es:

$$t = \frac{176\text{Bits}}{115200 \frac{\text{Bits}}{\text{Seg}}}$$

$$t = 1.5 \text{ mS}$$

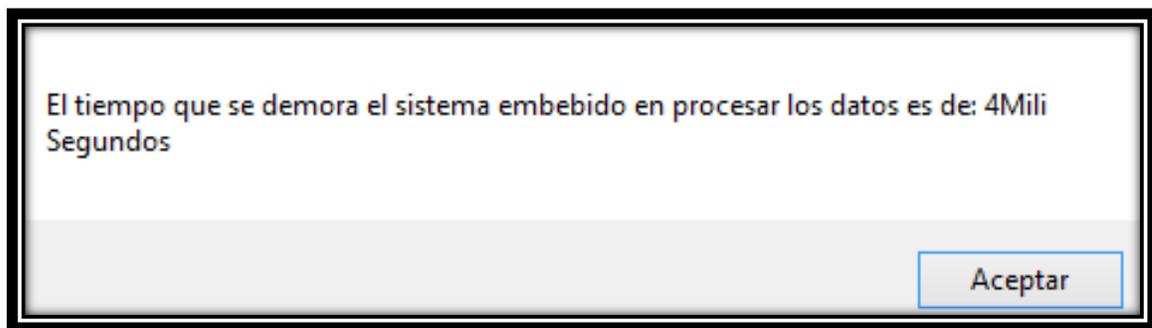
Para la modalidad de medición el tiempo es:

$$t = \frac{72 \text{ Bits}}{115200 \frac{\text{Bits}}{\text{Seg}}}$$

$$t = 0.62 \text{ mS}$$

Después de obtener los tiempos de retardo de cada uno de los procesos del sistema se realiza una sumatoria de los tiempos. Para la modalidad de visualización, la latencia del sistema desde el momento que se capturan los datos de los sensores hasta que se transmiten y reciben es de 4.07mS y para la modalidad de medición es de 3.19mS.

Para medir el tiempo que se demora el sistema embebido en procesar los datos leídos se emplea el mismo método que en el arduino con una función que permite visualizar el tiempo en micro segundos que demora el proceso.



Por lo anterior la latencia total del sistema para modalidad visualización es de 8.07mS y para la modalidad de medición es de 7.19mS.

6. ANÁLISIS DE RESULTADOS.

Al momento de implementar el sistema de adquisición y visualización, se realizaron varios procesos que permiten describir el funcionamiento y la linealidad del sistema.

- Comparación de las características de los sonómetros.
- Comparación de los datos del sonómetro.
- Comparación de los datos ambientales.

Los resultados de las mediciones dependen de las características de hardware y software de los dos sonómetros implementados.

6.1. Comparación de las características de los sonómetros.

Las características de hardware y software de los dos sonómetros se describen en la tabla número 4 y 5.

Características de hardware	Svan 943A	Sonómetro del sistema
Frecuencia de muestreo.	48Khz	50Khz
Resolución de conversión A/D.	20Bits	12Bits
Sensibilidad del micrófono.	51.6mV/Pa	11.42mV/Pa
Protocolo de comunicación.	Serial RS232	Serial RS232
Rango de frecuencia.	20Hz – 20KHz	15Hz-20KHz
Memoria.	8Mbytes	8Gbytes
Rango dinámico.	110dB	60dB

Tabla. 4. Características de hardware de los dos sonómetros.

Características de software	Svan 943A	Sonómetro del sistema
Sound meter level.	SPL, Leq, SEL, Max, Min, Percentiles.	SPL, Leq, Max, Min
Filtros de ponderación.	A, C, Lineal.	Lineal.
Filtros digitales.	30 – 1/3 octava 10 – 1/1 octava	No
Filtros temporales.	Slow, Fast, Impulsive, Peak.	Slow, Fast, Impulsive.

Tabla. 5. Características de software de los dos sonómetros.

6.2. Comparación de los datos del sonómetro con un sonómetro tipo II (Svan-943A).

El sonómetro del sistema se calibro simultáneamente con un sonómetro Svan tipo II y se realizaron tres mediciones comparativas para validar su funcionamiento. Estas mediciones se realizaron como indica el diagrama de la figura 56.

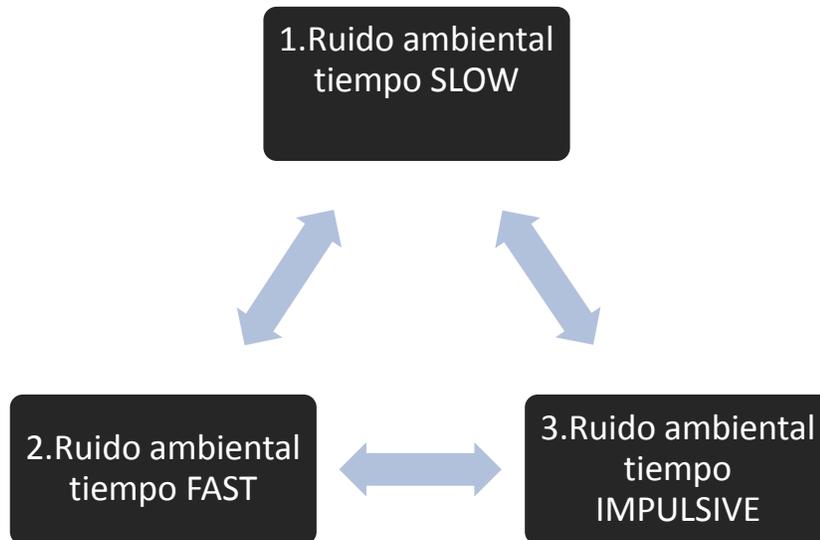


Figura. 56. Tipos de señales, analizadas para la comparación de los sonómetros.

Para cada una de las mediciones se realizó una configuración diferente, 10 segundos de medición para diferentes tiempos de integración, como se muestra en la tabla 6.

Tipo de medición	Duración (Segundos)	Tiempo de integración	Ponderación
• Ruido Ambiental.	10	Slow (1Segundo)	Lineal
• Ruido Ambiental.	10	Fast (125 Milisegundos)	Lineal
• Ruido Ambiental.	10	Impulsive (32 Milisegundos)	Lineal

Tabla. 6. Tabla de configuración para las mediciones.

Los datos a comparar son los niveles continuos equivalentes, los niveles máximos y los niveles mínimos de las tres mediciones. Para estos tres datos se calcula el error absoluto y el error relativo, La tabla 7 muestra los resultados de las mediciones realizadas en conjunto, por los dos sonómetros.

Medición Ruido ambiental 1 – 10 segundos - Slow			
	Leq	Max	Min
Sonómetro del sistema	73.34dB	81.99dB	65.85dB
Sonómetro van tipo II	71.92dB	75.90dB	57.41dB
Medición Ruido ambiental 2 – 10 segundos - Fast			
	Leq	Max	Min
Sonómetro del sistema	73.75dB	78.30dB	64.17dB
Sonómetro van tipo II	71.87dB	79.99dB	62dB
Medición Ruido ambiental 3 – 10 segundos – Impulsive.			
	Leq	Max	Min
Sonómetro del sistema	81.23dB	93.67dB	64.17dB
Sonómetro van tipo II	85.78dB	91.9dB	61dB

Tabla. 7. Datos comparativos de los sonómetros.

El error absoluto está dado por la extensión de la ecuación 7 donde f_m es el valor medido y f_r es el valor de comparación.

$$e_{abs} = |f_m - f_r|$$

Ecuación. 7. Expresión para el cálculo de error absoluto [40].

En los resultados de la tabla 2 se evidencia la diferencia en los valores de Leq obtenidos por los dos sonómetros. Para la configuración Slow, presenta un error absoluto de 1.42dB, para la configuración Fast de 1.88dB y para la configuración Impulsive de 4.55dB.

El cálculo del porcentaje de error relativo para las comparaciones de Leq entre los dos sonómetros está dado por la expresión de la ecuación 8.

$$e_{rel} = \frac{|fm - fr|}{fr} \times 100\%$$

Ecuación. 8. Expresión para el cálculo de error relativo [40].

La tabla 8 presenta los resultados del error relativo para las comparaciones de Leq.

Tiempo De Integración	Porcentaje de error relativo Leq
Slow	1.97%
Fast	2.61%
Impulsive	5.3%

Tabla. 8. Error relativo del nivel continuo equivalente.

Para la comparación de los valores máximos y mínimos obtenidos por los dos sonómetros en las tres diferentes mediciones, se calcula de la misma manera el error absoluto y el error relativo como se muestra en la tabla 9.

Tiempo De Integración	Porcentaje de error relativo Lmax	Error absoluto Lmax
Slow	8.02%	6.09dB
Fast	2.11%	1.69dB
Impulsive	1.92%	1.77dB
Tiempo De Integración	Porcentaje de error relativo Lmin	Error absoluto Lmin
Slow	14.7%	8.44dB
Fast	3.5%	2.17dB
Impulsive	5.19%	3.17dB

Tabla. 9. Error relativo y absoluto para el nivel máximo y mínimo.

Las gráficas 57,58 y 59 representan el comportamiento del nivel de presión sonora en el intervalo de tiempo de 10 segundos, para cada una de las tres mediciones realizadas.

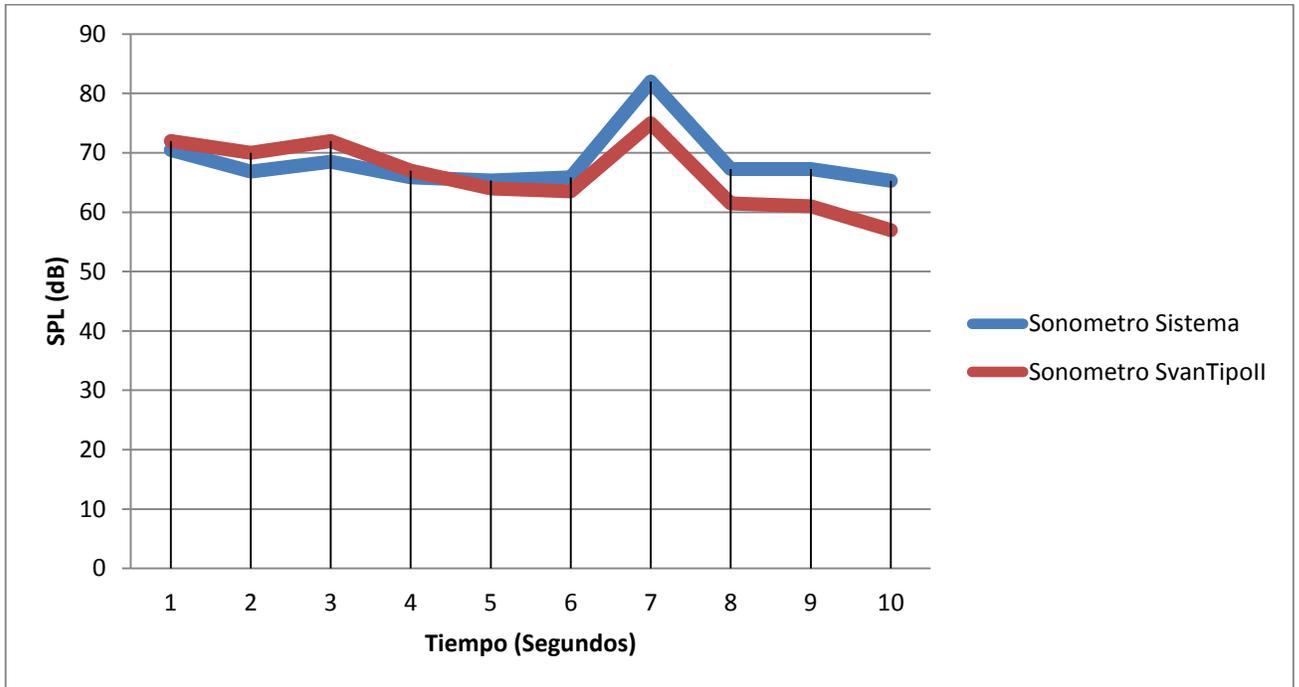


Figura. 57. Medición ruido ambiental tiempo de integración Slow.

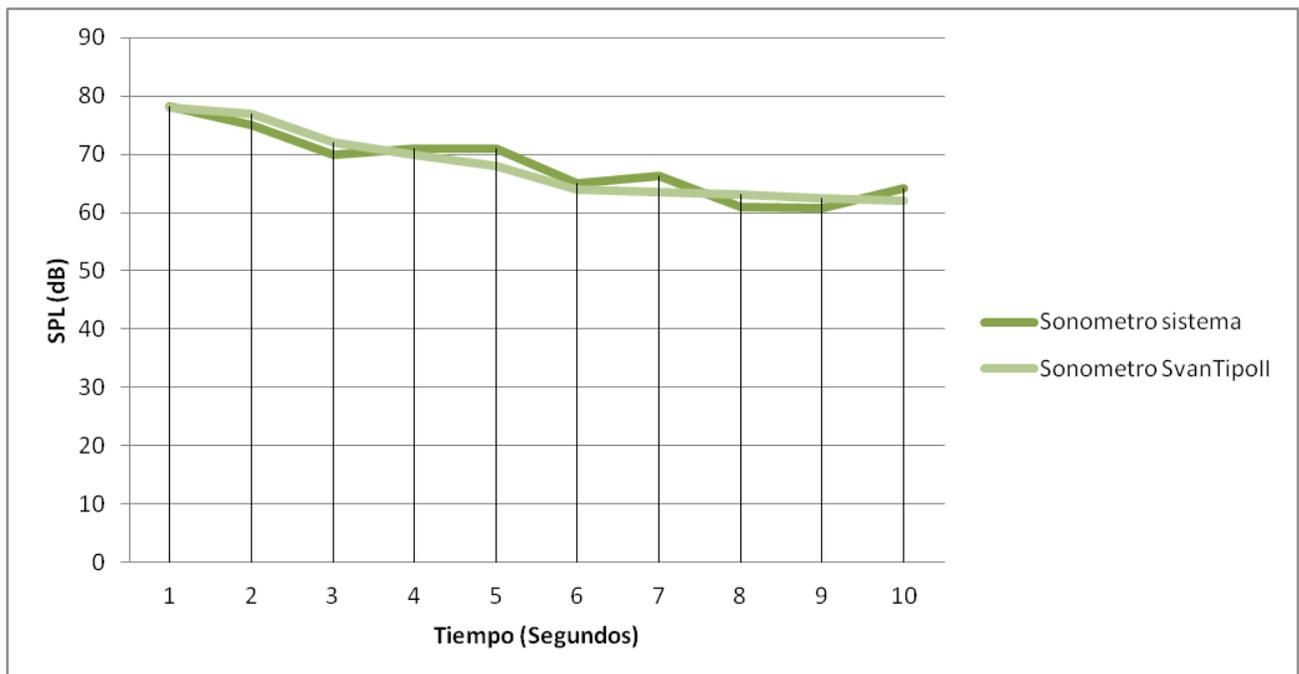


Figura. 58. Medición ruido ambiental tiempo de integración Fast.

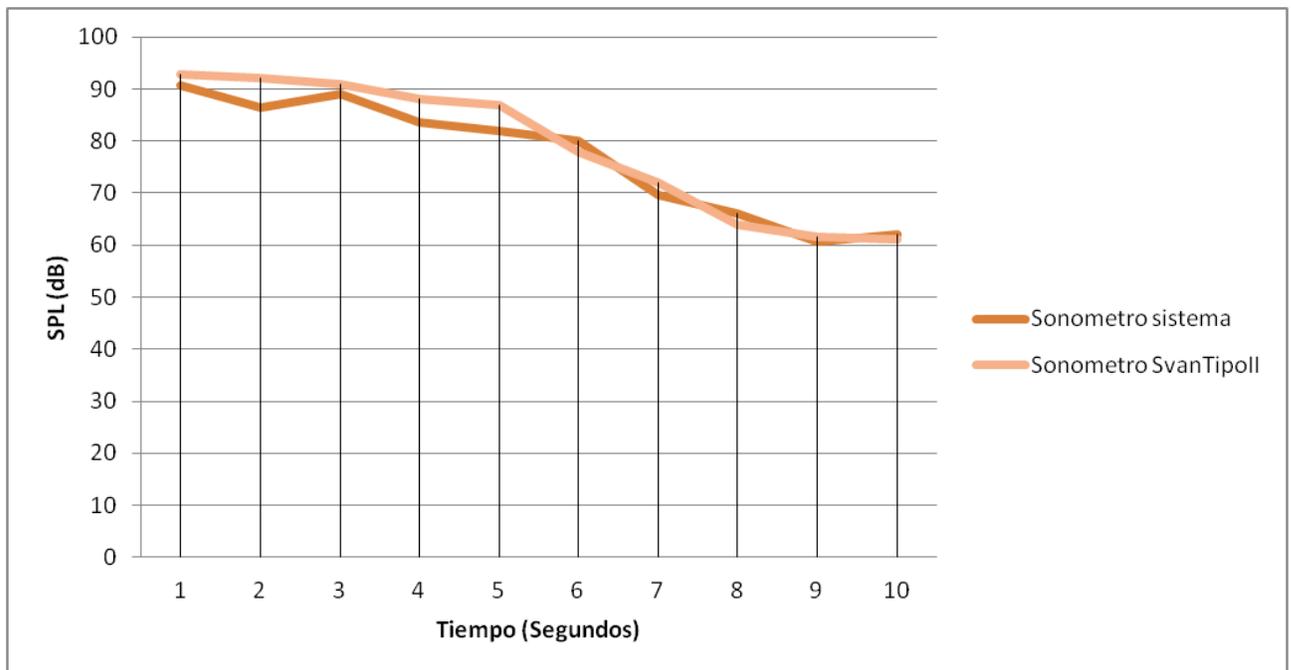


Figura. 59. Medición ruido ambiental tiempo de integración Impulsive.

La comparación de los niveles de presión sonora en cada una de las tres mediciones realizadas presenta un patrón similar en el comportamiento temporal como se evidencia en las figuras 57, 58 y 59.

La medición con tiempo de integración Slow (Figura 57), presenta un realce casi al final de la medición, captado por los dos sonómetros.

En el caso de la figura 58 y 59 el comportamiento de las gráficas tiende a ser un poco más estable, simulando un comportamiento lineal con pendiente negativa.

6.3. Comparación de los datos climáticos con una estación VAISALA.

Se realizó una medición comparativa de los datos climáticos entre el sistema y una estación meteorológica de marca VAISALA “QL150 Sensor Collector” [39], esto con el fin de evidenciar el funcionamiento y la precisión de los sensores implementados en el sistema.

El tiempo de medición seleccionado fue de 15 horas, donde se realizaron registros de los datos climáticos cada hora. En la tabla 10 se evidencian los datos obtenidos en el periodo de tiempo.

Medición comparativa entre el sistema y la estación meteorológica VAISALA								
Hora Fecha 10/02/2014	Estación meteorológica vaisala				Sistema desarrollado			
	Temp°C	Hum %	Vel m/s	Dir °N	Temp°C	Hum %	Vel m/s	Dir °N
5 am	12.7	85	0	332	12	85	0	332
6 am	11.6	87	0	242	12	87	0	242
7 am	13.8	79	1.1	174	13	76	0	174
8 am	22.9	41	0.1	90	23	45	0	98
9 am	25.2	38	0.4	174	25	40	0	174
10 am	27.4	31	2	113	27	29	2	109
11 am	27.4	33	0	304	27	35	0	305
12 pm	25.3	36	0	125	25	36	0	119
1 pm	25.4	34	3	331	25	32	2	330
2 pm	25.4	35	0	75	25	36	0	75
3 pm	22.8	38	0	24	22	37	0	24
4 pm	25.1	31	0	224	25	32	0	231
5 pm	19.8	45	5	66	20	49	7	65
6 pm	16.8	59	5	147	16	57	5	139
7 pm	15.2	75	8	83	16	81	5	90

Tabla. 10. Tabla de resultados de la medición climática.

La resolución de los sensores de la estación vaisala permite obtener datos más precisos debido a su calidad de construcción integral y unificada [39].

Las gráficas de la 60 hasta la 63 representan el comportamiento de las variables climáticas, en la medición de comparación.

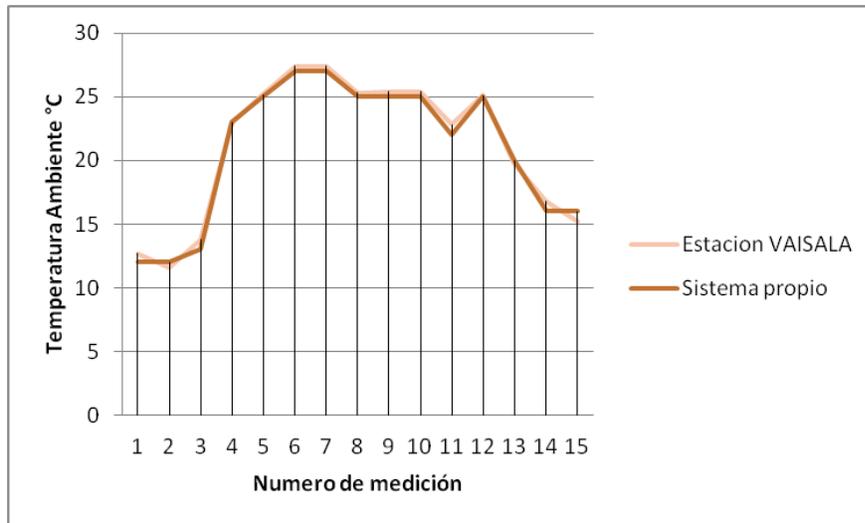


Figura. 60. Comparación de la señal de temperatura.

En la figura 60 se puede evidenciar el comportamiento de la temperatura ambiente, aumento de temperatura en la tarde, y una disminución en la noche. Este fenómeno no presenta cambios abruptos en ninguno de los 2 sistemas.

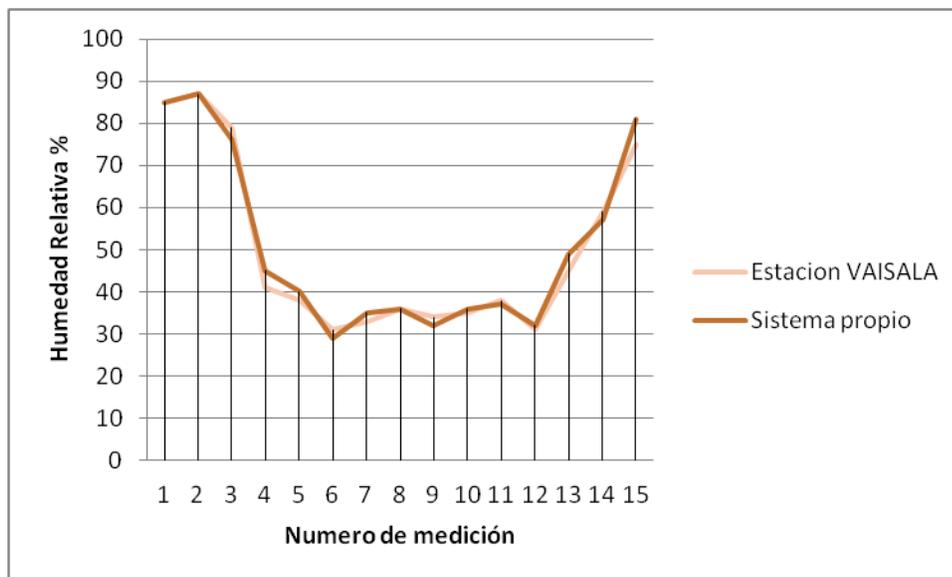


Figura. 61. Comparación de la señal de humedad relativa.

La figura 61 contiene la información de la humedad relativa del ambiente, esta humedad se comporta de manera inversa que la temperatura, por esta razón la sección con menos humedad se presentaron al medio día.

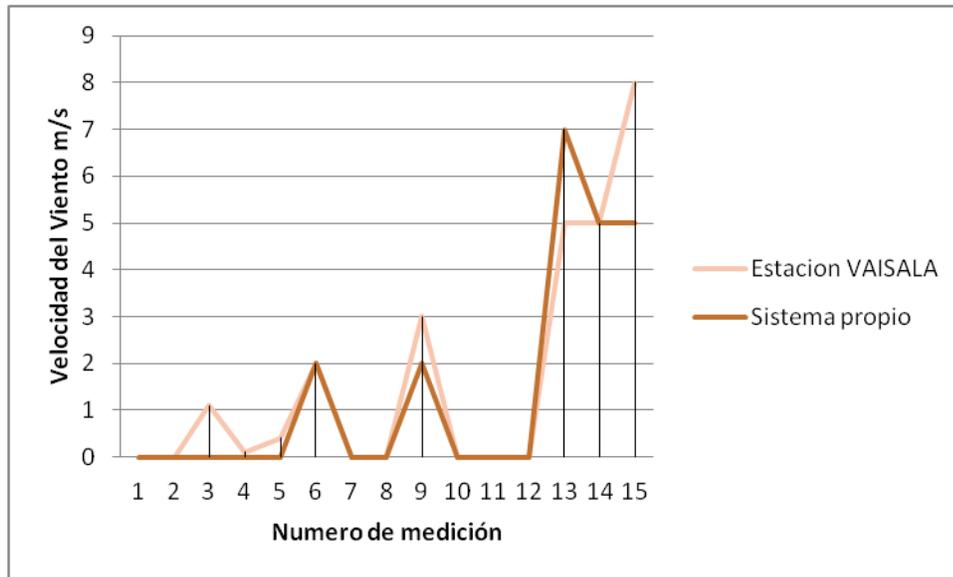


Figura. 62. Comparación de la velocidad del viento.

La velocidad del viento presenta un comportamiento intermitente, debido a las fluctuaciones aleatorias que presenta este fenómeno, en la figura 62 se evidencia mayor velocidad del viento en las horas de la noche.

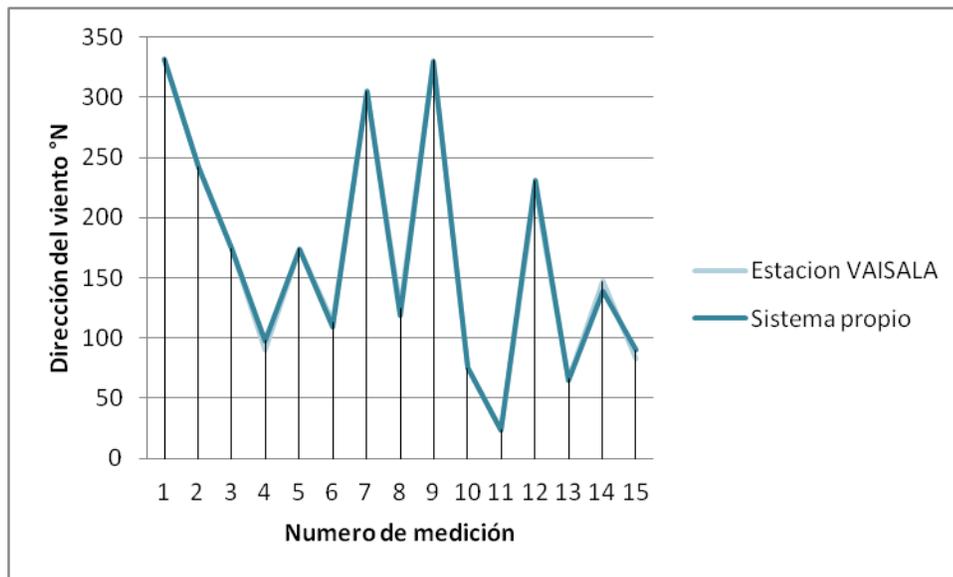


Figura. 63. Comparación de la dirección del viento.

El comportamiento de la veleta de viendo del sistema desarrollado presenta el mismo patrón que la veleta de la estación VAISALA, como se muestra en la figura 63 la dirección del viento tubo cambios abruptos en las horas del mediodía.

Hora Fecha 10/02/2014	% Error Relativo				Error absoluto			
	Temp	Hum	Vel	Dir	Temp°C	Hum %	Vel m/s	Dir °N
5 am	5.5%	0%	0%	0%	0.7	0	0	0
6 am	3.4%	0%	0%	0%	0.4	0	0	0
7 am	5.7%	3%	100%	0%	0.8	3	1.1	0
8 am	0.4%	9.7%	100%	8.8%	0.1	4	0.1	8
9 am	0.8%	5.2%	100%	0%	0.2	2	0.4	0
10 am	1.4%	6.4%	0%	3.5%	0.4	2	0	4
11 am	1.4%	6%	0%	0.32%	0.4	2	0	1
12 am	1.1%	0%	0%	4.8%	0.3	0	0	6
1 am	1.5%	5.8%	39.9%	0.3%	0.4	2	2	1
2 am	1.5%	2.8%	0%	0%	0.4	1	0	0
3 am	3.5%	2.6%	0%	0%	0.8	1	0	0
4 am	0.4%	3.2%	0%	3.12%	0.1	1	0	7
5 am	1%	8.8%	40%	1.51%	0.2	4	2	1
6 am	4.7%	3.3%	0%	5.44%	0.8	2	0	8
7 am	5.2%	8%	37.5%	8.4%	0.8	6	3	7

Tabla. 11. Error relativo y absoluto entre las mediciones climáticas realizadas.

La tabla 11 describe el análisis de error relativo y absoluto de las mediciones climáticas realizadas por los dos sistemas.

6.4. Consumo de memoria RAM del sistema.

Para identificar cual es el consumo de memoria RAM del software desarrollado, en el sistema embebido se utiliza la herramienta de administrador de tareas de Windows y se procede a identificar la tarea de la aplicación y visualizar cual es el consumo. La figura 64 presenta el consumo de memoria RAM por parte del software desarrollado.

Nombre	Estado	CPU	Memoria	Disco	Red
EstaciónMeteorologica		1,6%	6,6 MB	0 MB/s	0 Mbps

Figura. 64. Consumo de memoria RAM por la aplicación “EstaciónMeteorológica”.

En la figura tal se evidencia que el consumo de memoria RAM es de 6.6 Mbytes y el consumo de CPU de 1.6%.

7. CONCLUSIONES.

- El documento presenta el desarrollo de un sistema de adquisición y visualización del nivel de presión sonora y variables ambientales, implementado en el sistema embebido Mini210s. En el S.E se implementó el sistema operativo Windows CE6, el cual es compatible con la interfaz desarrollada en visual basic, y con el protocolo de comunicación serial.
- El uso de sensores climáticos con una incertidumbre de $\pm 2\%$ de precisión, como los sensores de la estación meteorológica VAISALA, permiten obtener lecturas representadas en notación decimal.
- Las diferencias de nivel que presentan las tres mediciones realizadas con los dos sonómetros, se deben a las diversas características de los micrófonos y los conversores análogo digital, arrojando un error máximo del 5.3% para los resultados comparativos del nivel continuo equivalente. Por otro lado, el máximo error relativo para las comparación de niveles máximos y mínimos fue 8.02% y 14.7% respectivamente.
- El valor de sensibilidad del micrófono ECM8000, proporcionado por el fabricante Behringer no coincide con el valor real obtenido, ni con las unidades correspondientes de sensibilidad. Ya que el valor suministrado por el fabricante es de 70dB y el valor de sensibilidad real del micrófono es de 11.42mV/Pa.
- La transferencia de tecnología y conocimiento es importante en la innovación y creación de nuevos procesos, gracias a las ayudas técnicas que se brindan de manera libre, al momento de realizar desarrollos ingenieriles y tecnológicos que a su vez sirven como guía para nuevos proyectos.

- Los valores de latencia que presenta el sistema están por debajo del valor de tiempo mínimo de transmisión (filtro temporal Impulsive), por esta razón el sistema no tiene retrasos al momento de transmitir los datos obtenidos.
- Utilizar conversores análogos digitales con una ventana de operación superior a 1.74 Vp (Voltios Pico), permite tener un rango de trabajo sobre el nivel de línea (+4dBu) después de la etapa de pre amplificación.

8. RECOMENDACIONES.

- Implementar algún tipo de seguridad informática al momento de realizar conexiones por sockets ya sea por protocolo WIFI o Ethernet.
- Realizar mediciones de las características del micrófono para validar los datos suministrados por el fabricante.
- Implementar estrategias de almacenamiento de los datos obtenidos por el sistema, como bases de datos en Mysql, SQL, Acces etc.
- En el desarrollo de una interfaz de usuario es importante analizar los pros y contras del software de programación debido a que son realizados para utilidades diferentes.
- Utilizar el mínimo número de interfaces de usuario posible, debido a que cada interfaz de usuario puede ser un punto de caída del sistema.
- Para la transmisión de cadenas cortas de datos seriales entre dos dispositivos, se recomienda utilizar velocidades altas de transmisión como 115200baudios, para poder tener un mayor control de la información, si las cadenas de datos son largas

es preferible utilizar velocidades medias y bajas como 9600 y 4800 baudios para no tener perdida de datos por la latencia propia del sistema.

- Para el diseño y desarrollo del pre amplificador es importante tener en cuenta los factores que afectan las señales de audio, como ruido de alta frecuencia y ruido inducido por la acometida eléctrica.

9. BIBLIOGRAFIA.

- [1] **HERNÁNDEZ JUÁREZ**, José Roberto. *Diseño de un sonómetro*. Instituto Politécnico Nacional. México D.F., México 2009.
- [2] **DICKINSON, DAN**; Leider, Colby; Mann, Doug. *Wireless multisensor monitoring of the Florida Everglades: A pilot project*. Noviembre 2010.
- [3] **ÁLVAREZ G, PICO J VICARIO, L AMUCHÁSTEGUI, AYUSO N. BENÍTEZ N.** *Desarrollo de un protocolo para comunicaciones inalámbricas e implementación en dispositivos de lógica configurable*. Facultad de Informática. San Sebastián, España.
- [4] **SÁNCHEZ ZAPATA, CARMEN ELENA.** *Evaluar y monitorear la calidad del aire en el Valle de Aburrá*. Universidad Nacional de Colombia. De abril de 2011 a mayo de 2012.
- [5] **PLASENCIA SÁNCHEZ, EDSON. CARRANZA CABRERA, CARLOS.** *El ruido en las operaciones mineras: El caso de Yanacocha Oeste*. Revista del Instituto de Investigaciones FIGMMG. Vol. 12, No. 24. Pp. 141-147. Junio de 2009 .
- [6] **BERNABÉ DÍAZ, JOSÉ LUIS.** *Diseño de un sistema embebido para el control en tiempo real de un convertidor elevador alimentado por una celda combustible*. Instituto Politécnico Nacional. México D.F., México. Noviembre de 2009.
- [7] **ROSAS JIMÉNEZ, NELSON FELIPE.** *Diseño e implementación de un sistema embebido para la adquisición y transmisión de señales biomédicas a través de la red celular*. Universidad Nacional de Colombia. Bogotá, Colombia. 2011.
- [8] **CAMARGO BAREÑO, CARLOS IVÁN.** *Transferencia tecnológica y de conocimientos en el diseño de sistemas embebidos*. Universidad Nacional de Colombia. 2011.
- [9] **YAGHMOUR, KARIM.** *Building Embedded Linux Systems*. O'Reilly Media, Inc., 2003.
- [10] **GRALLA, PRESTON.** *Cómo funcionan las redes inalámbricas*. Anaya Multimedia. 2009.

- [11] **MICROSOFT CORPORATION.** *Windows CE 6.0 Based x86 Embedded System.* 2003 (emebido).
- [12] **GLEN BALLOU,** *Handbook for sound engineers. The new audio cyclopedia.* Focal Press 1998. (preamp).
- [13] **M.C CANTO QUINTAL, CARLOS E.** *El Puerto Serie RS232.* 2005.
- [14] **FRIENDLY ARM.** *Mini2440 User Manual.*
- [15] **ESCOBAR DUERTE ANDRÉS.** *Programando en serio con visual basic 6.0 segunda edición.* 2008.
- [16] **INTEL EMBEDDED.** *Using Windows 7 embedded standard 7.* Diciembre. 2011.
- [17] **IEC 61672-1.** *Electroacústica- sonómetros parte 1 especificaciones.* 2002-2005.
- [18] **COBO PARRA PEDRO.** *Contaminación acústica y niveles límite de ruido.* Instituto de acústica, CSIC serrano, Madrid. España 2006.
- [19] **BUENO JORGE,** *Literatura Valvulera Artículo publicado por la revista especializada Guitarra Total en los números 22 y 23.*
- [20] **DR RUIZ PADILLA DIEGO,** *Contaminación acústica Introducción y fundamentos.* Universidad de Granada.
- [21] **SANCHEZ VALENZUELA MAURICIO,** *Metrología Acústica.* Gobierno de chile. 2005.
- [22] **BIDONDO ALEJANDRO,** *Micrófonos.* 2004.
- [23] **ROLDAN ARANDA ANDRES,** *Los micrófonos en acústica.* 2000 – 2001.
- [24] **FRIENDLY ARM.** *Mini210s User Manual.*
- [25] **ABAD TORIBIO LAURA,** *Ruido ambiental seguridad y salud.* Universidad Alfonso X, Madrid. España 2011.
- [26] **MEDIO AMBIENTE EN ESPAÑA,** *Estudio y evolución del medio ambiente.* 2006.

- [27] **ABBOTT, DOUG.** *Linux for Embedded and Real-Time Application.* Newnes. 2009.
- [28] **MADELEINE RENOM.** *Principios básicos de las mediciones atmosféricas.* 2011
- [29] **HUERTA EDUARDO, MANGIATERRA ALDO, NOGUERA GUSTAVO.** *GPS posicionamiento satelital.* Editorial UNR 2005.
- [30] **MOLINA CANTERO JAVIER.** *GPS.* Universidad de Sevilla 2009.
- [31] **MINISTERIO DE AMBIENTE, VIVIENDA Y DESARROLLO TERRITORIAL.** *RESOLUCION 0627.* 2006
- [32] <<FRIENDLY ARM>><http://friendlyarm.net>
- [33] <<ARDUINO DUE>><http://Arduino.cc>
- [34] **CABALLERO ANDRÉS.** *Desarrollo de un controlador midi no convencional implementando un sistema embebido.* Universidad de San Buenaventura 2011.
- [35] <<NATIONAL INSTRUMENTS>><http://digital.ni.com/>
- [36] **SENSIRION.** *SHT11 datasheet.*
- [37] **APRS WORLD.** *#40R datasheet.*
- [38] **TEXAS INSTRUMENTS.** *INA 217 datasheet.*
- [39] <<VAISALA>> <http://es.vaisala.com>
- [40] **MIRANDA MARTÍN DEL CAMPO JAVIER.** *Evaluación de la incertidumbre en datos experimentales.* 2009.

ANEXO A – Compilación del sistema operativo Windows CE en el sistema embebido.

Adecuación Sistema Embebido

El sistema embebido seleccionado como sistema maestro fue el sistema desarrollado por friendly ARM “Mini210s”(ver figura 65), este sistema cuenta con una velocidad de procesamiento de 1GHz, una memoria RAM de 512 Megabytes, y una pantalla touch Screen de 4.3 pulgadas; Esto lo convierte en el sistema embebido idóneo para el desarrollo del sistema en general.

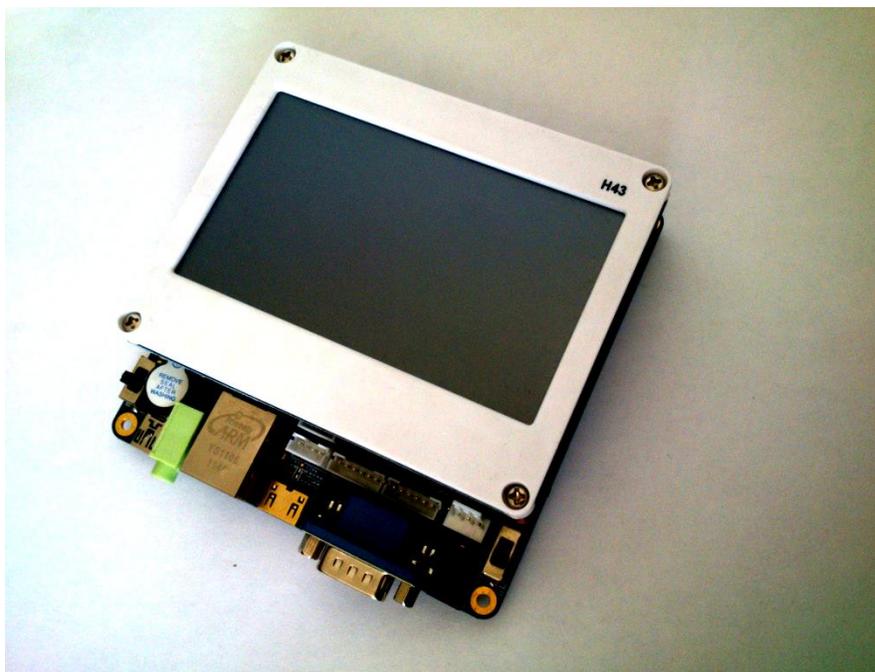


Figura. 65. Mini210s con Touch Screen Fuente: propia

El sistema embebido permite la implementación de tres diferentes sistemas operativos:

- Linux Qtopia.
- Android 4.1
- Windows CE 6.0

Cada uno de estos sistemas es brindado por el fabricante de forma compilado y listo para instalarlo en el sistema embebido.

Elección e instalación del Sistema Operativo

Al momento de analizar los beneficios de cada uno de los tres sistemas operativos nombrados anteriormente, Windows CE 6.0 fue elegido como el más idóneo para ser instalado en el sistema embebido, debido a su relación directa con el entorno de programación visual estudio, un entorno de programación dedicado al desarrollo de aplicaciones orientadas a objetos [15].

Los otros sistemas tenían limitantes en el desarrollo de interfaces graficas debido a no tener un entorno propio de programación como la posee Windows.

Instalación de Windows CE 6.0.

Para instalar el sistema operativo en el sistema embebido se debe adecuar la tarjeta de almacenamiento extraíble, en este caso una memoria microSD de 8Gigabytes de almacenamiento como se muestra en la figura 66.



Figura. 66. Memoria de almacenamiento MicroSD 8Gigabytes. Fuente: Propia

El primer paso es dar un formato especial a la tarjeta SD para esto se utiliza la herramienta brindada por el fabricante “SD-Flasher”. Ver figura 67.

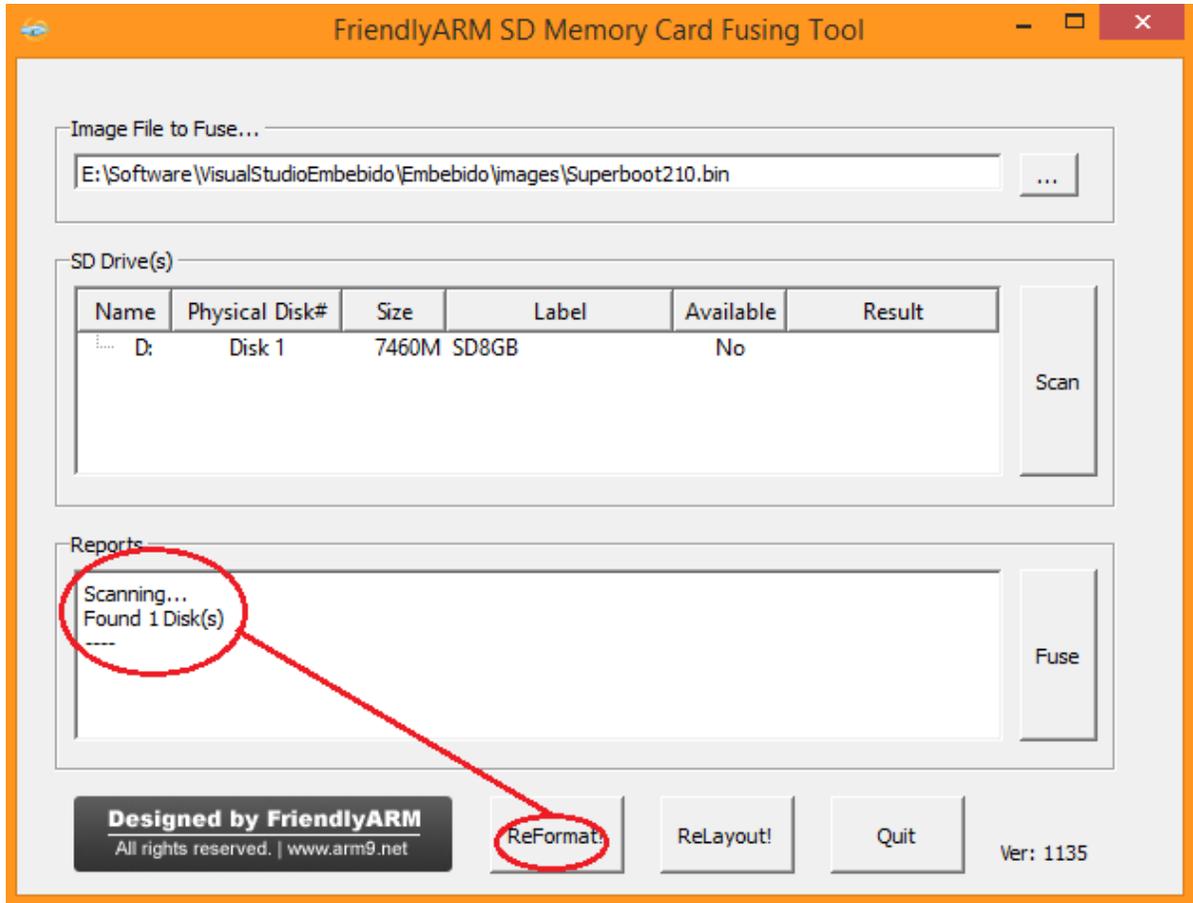


Figura. 67. SDFlasher formato a la memoria. Fuente: propia.

SD-Flasher es una utilidad que le da un formato especial a la tarjeta para ser reconocida por el sistema embebido y además almacena un archivo boot loader para darle instrucciones al sistema embebido como rutas de acceso y configuraciones.

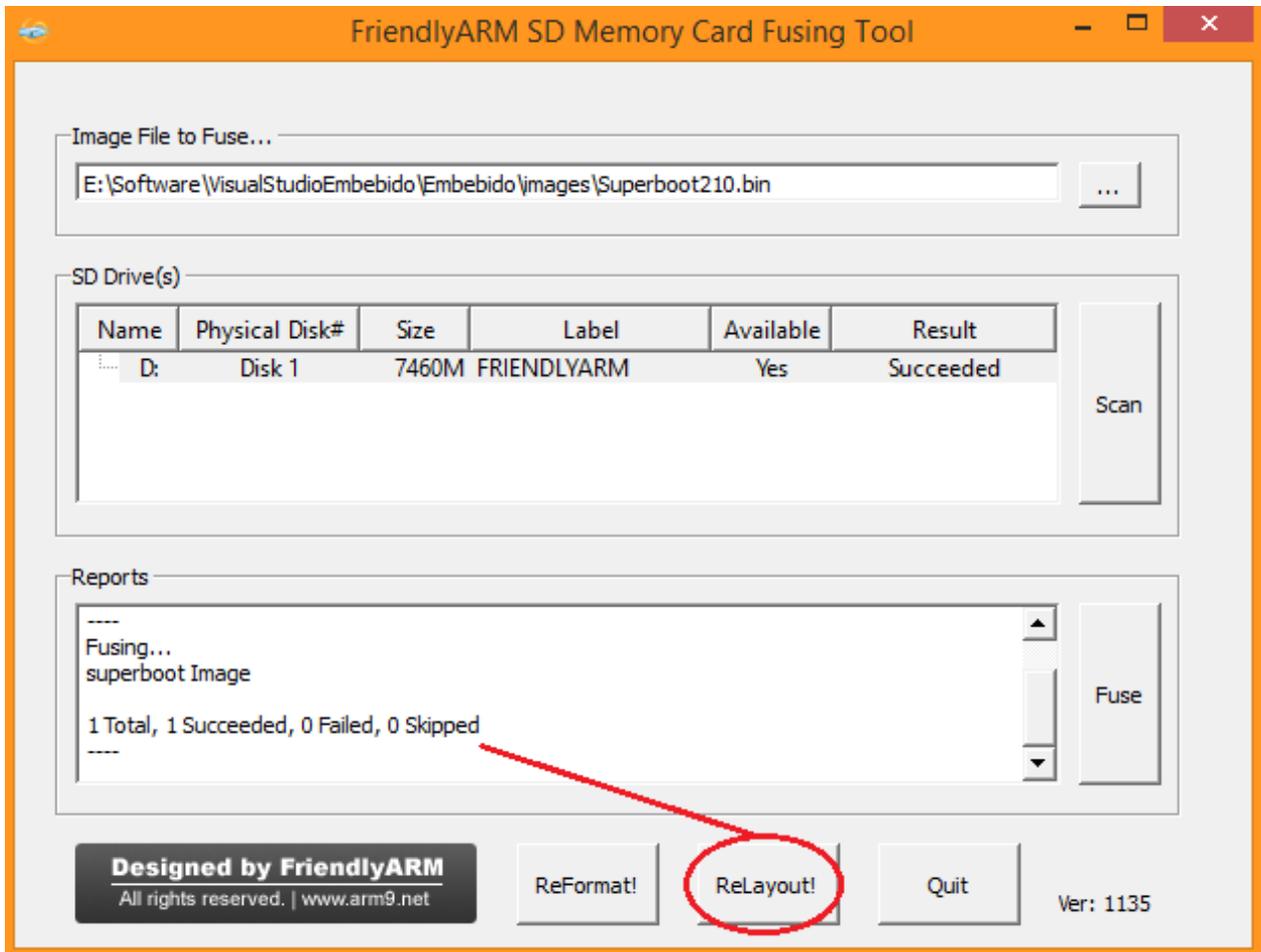


Figura. 68. Relayout en la tarjeta SD. Fuente: propia

El siguiente paso es configurar el archivo de inicio “FriendlyARM.ini” y seleccionar Windows CE 6 como sistema operativo a instalar. Este procedimiento se visualiza en la figura 69.

```

Archivo Edición Formato Ver Ayuda
#This line cannot be removed. by FriendlyARM(www.arm9.net)

CheckOneButton=No
Action = Install
OS = CE6
LCD-Mode = No
LCD-Type = H43

LowFormat = No
VerifyNandWrite = No
CheckCRC32=No

StatusType = Beeper | LED

##### Android 4.0.3 #####
Android-BootLoader = Superboot210.bin
Android-Kernel = Android/zImage
Android-CommandLine = root=/dev/mtdblock4 rootfstype=yaffs2 console=ttySAC0,115200 init=/linuxrc androidboot.console=ttySAC0 gs=0
Android-RootFs-InstallImage = Android/rootfs_android-mlc2.img

##### Android 2.3.1 #####
#Android-BootLoader = Superboot210.bin
#Android-Kernel = Android2.3.1/zImage
#Android-CommandLine = root=/dev/mtdblock4 rootfstype=yaffs2 console=ttySAC0,115200 init=/linuxrc androidboot.console=s3c2410_serial0 gs=0
#Android-RootFs-InstallImage = Android2.3.1/rootfs_android-mlc2.img

##### Linux #####
Linux-BootLoader = Superboot210.bin
Linux-Kernel = Linux/zImage
Linux-CommandLine = root=/dev/mtdblock4 rootfstype=yaffs2 console=ttySAC0,115200 init=/linuxrc
Linux-RootFs-InstallImage = Linux/rootfs_qtopia_qt4-mlc2.img

##### Windows CE6.0 #####
WindowsCE6-Bootloader = Superboot210.bin
WindowsCE6-BootLogo = WindowsCE6\bootlogo.bmp

```



Figura. 69. Configuración del archivo de inicio en el sistema embebido. Fuente: Propia.

Después dentro de la tarjeta SD se crea un directorio llamado “images” y se almacenan los archivos provistos por el fabricante como indica la figura 70, para que el sistema embebido pueda reconocerlos:

FRIENDLYARM (Tarjeta SD)	IMAGES	WINDOWS CE6	NK.bin (imagen compilada)
			Bootlogo.bmp
		FriendlyARM.ini	
		Superboot.bin	

Figura. 70. Diagrama de disposición de los archivos dentro de la memoria SD. Fuente: propia.

Posterior a esto se introduce la tarjeta SD en el sistema embebido, se configura el método de arranque desde la SD como muestra la figura 71. Al encender el sistema embebido aparece el proceso actual de la instalación. Ver figuras 72 y 73.



Figura. 71. Boot desde la tarjeta SD. Fuente: propia

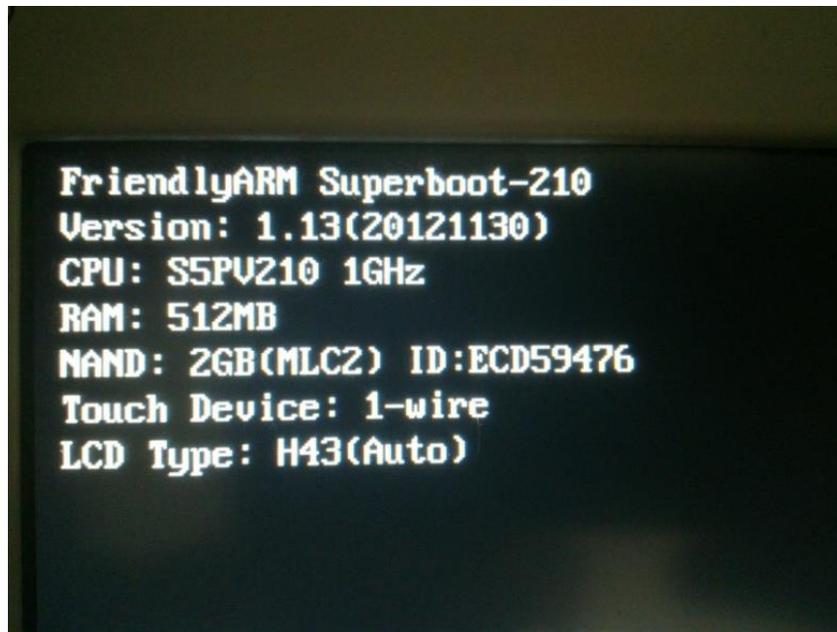


Figura. 72. Información previa a la instalación del sistema operativo. Fuente: propia

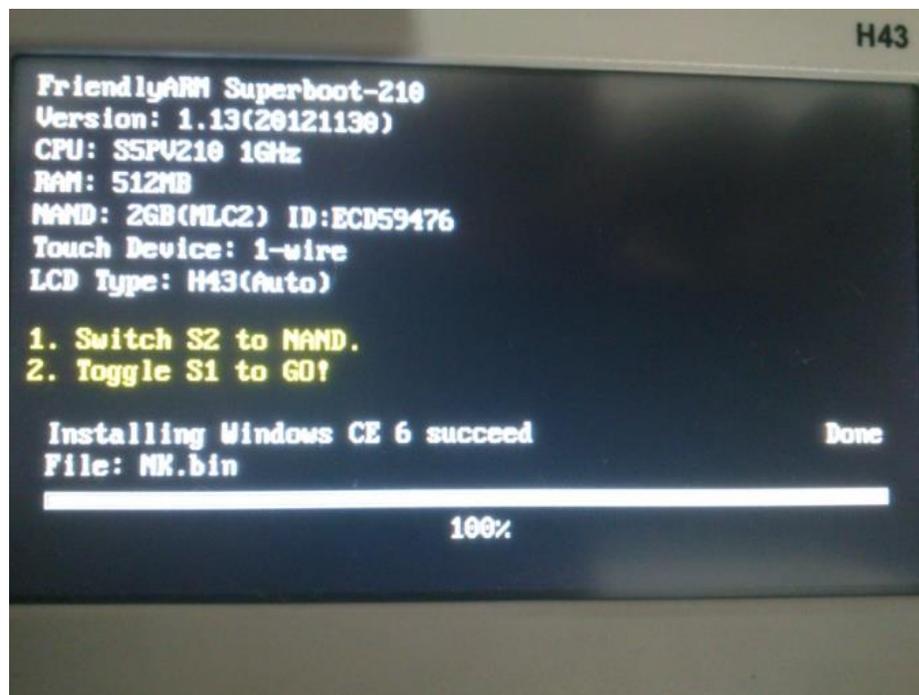


Figura. 73. Proceso de instalación de Windows CE finalizado correctamente. Fuente: propia.

Al completar la instalación se apaga el sistema y se cambia el modo de arranque a memoria interna. Una vez encendido se observa el sistema operativo Windows CE 6.0 ejecutándose internamente en el sistema embebido. Ver figura 74.



Figura. 74. Windows CE desde el sistema embebido. Fuente: Propia.

ANEXO B – Creación del archivo ejecutable para la implementación en el sistema embebido.

Generar un archivo ejecutable a partir de la interfaz de usuario

Visual estudio permite compilar la interfaz de usuario y el código, en un archivo ejecutable para Windows (.exe), este archivo por defecto se compila para arquitecturas de 32Bits.

Como muestra la figura 75 en el menú de la ventana de visual estudio seleccionamos la pestaña “Generar” y luego “Generar InterfazEmbebido”, esto compila todo el proyecto y lo almacena en la carpeta principal.

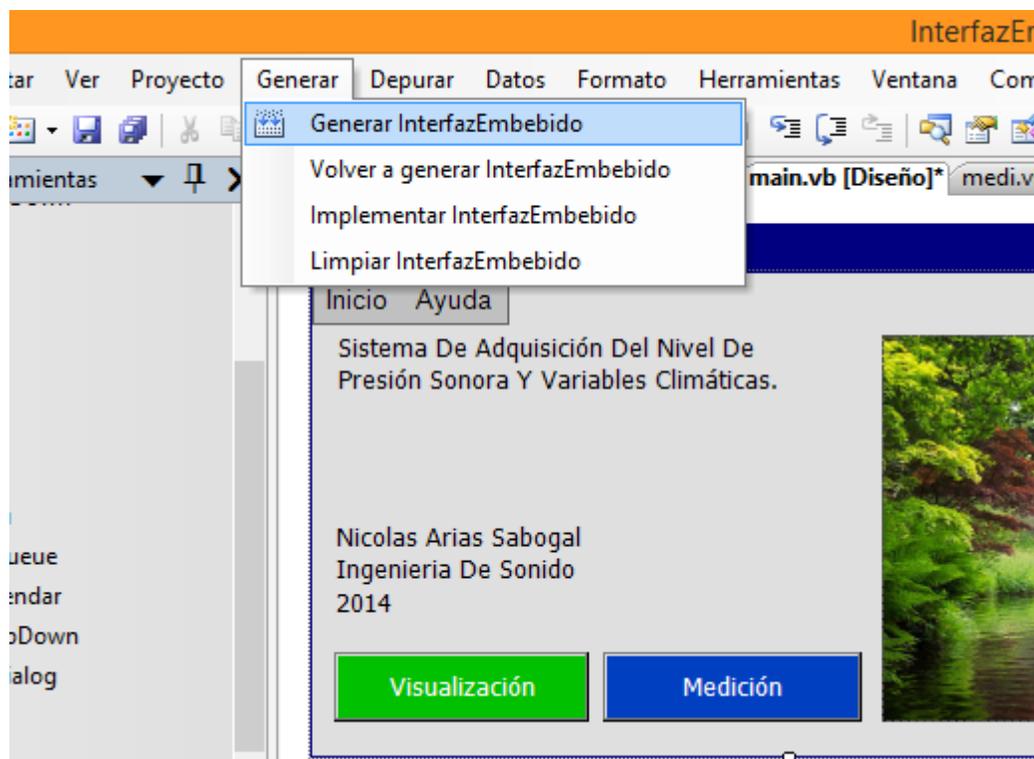


Figura. 75. Compilación de la interfaz de usuario. Fuente: propia

Nombre	Fecha de modifica...	Tipo	Tamaño
bin	31/01/2014 9:24 a. ...	Carpeta de archivos	
My Project	29/01/2014 10:40 ...	Carpeta de archivos	
obj	29/01/2014 10:40 ...	Carpeta de archivos	
InterfazEmbebido	29/01/2014 6:36 p....	Visual Basic Proje...	6 KB
InterfazEmbebido.vbproj	29/01/2014 4:29 p....	Visual Studio Proj...	1 KB
main.Designer.vb	29/01/2014 10:06 ...	Visual Basic Sourc...	5 KB
main	29/01/2014 10:06 ...	Archivo RESX	360 KB
main.vb	29/01/2014 10:06 ...	Visual Basic Sourc...	1 KB
medi.Designer.vb	30/01/2014 7:26 p....	Visual Basic Sourc...	16 KB
medi	30/01/2014 7:26 p....	Archivo RESX	26 KB
medi.vb	30/01/2014 7:26 p....	Visual Basic Sourc...	4 KB
moni.Designer.vb	30/01/2014 7:26 p....	Visual Basic Sourc...	16 KB
moni	30/01/2014 7:26 p....	Archivo RESX	22 KB
moni.vb	30/01/2014 7:26 p....	Visual Basic Sourc...	4 KB
spl.Designer.vb	29/01/2014 10:01 ...	Visual Basic Sourc...	12 KB
spl	29/01/2014 10:01 ...	Archivo RESX	7 KB
spl.vb	30/01/2014 7:43 p....	Visual Basic Sourc...	6 KB

Figura. 76. Archivos generados por la compilación de la interfaz. Fuente: propia.

En la ubicación de origen del proyecto seleccionamos la carpeta “bin” como se muestra en la figura 76 y 77 donde internamente se encuentra la carpeta de nombre “debug” y finalmente el archivo ejecutable de la interfaz de usuario “interfazEmbebido”.

Debug	30/01/2014 7:41 p....	Carpeta de archivos	
Release	29/01/2014 11:39 a...	Carpeta de archivos	

Nombre	Fecha de modifica...	Tipo	Tamaño
InterfazEmbebido	30/01/2014 7:41 p....	Aplicación	311 KB
InterfazEmbebido	30/01/2014 7:41 p....	Program Debug D...	58 KB
InterfazEmbebido	30/01/2014 7:41 p....	Archivo XML	1 KB

Figura. 77. Rutas para acceder al archivo ejecutable .exe. Fuente: propia.

Compilación del ejecutable en el sistema embebido

Para compilar el archivo ejecutable en el sistema embebido utilizamos el software “Windows Mobile 6.1” que se encargara de trasferir y almacenar el archivo ejecutable en el sistema embebido.

El sistema embebido se enciende y se conecta al computador, en la figura 78 se evidencia como el software identifica el sistema embebido y permite el acceso a su memoria interna.



Figura. 78. Windows Mobile Center, dispositivo conectado. Fuente: propia.

Se transfiere el archivo ejecutable a la memoria interna NANDflash. (Ver figura 79 y 80)

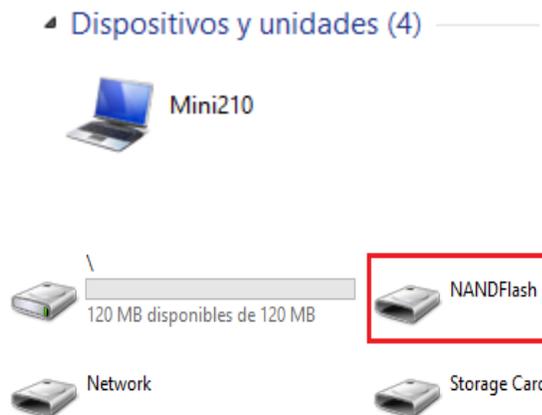


Figura. 79. Dispositivos de almacenamiento en el sistema embebido. Fuente: propia.

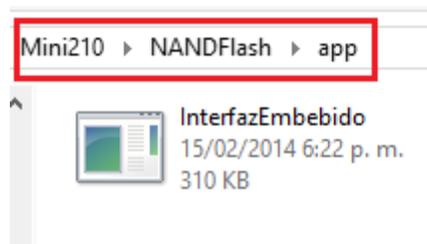


Figura. 80. Archivo ejecutable almacenado en el sistema embebido. Fuente: propia.

Y por último en la imagen de la figura 81 se comprueba que el archivo ejecutable se encuentra en el interior del sistema embebido.

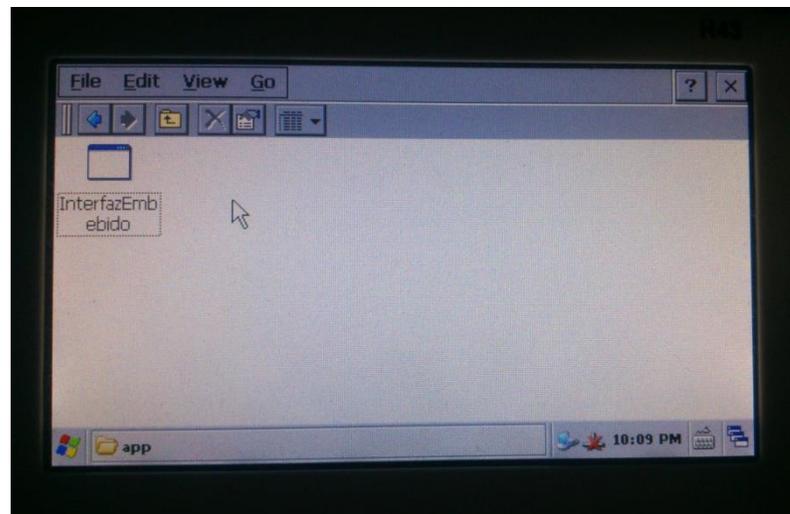


Figura. 81. Archivo ejecutable almacenado en el sistema embebido. Fuente: propia.

ANEXO C – Anexos digitales.

Con la presentación de este documento se incluye un CD de datos el cual posee información y evidencias, del desarrollo y análisis del sistema.