

# PROTOTIPO DE UNA ESTACIÓN METEOROLÓGICA

Trabajo para optar el título de Tecnólogo en Electrónica Industrial

SERGIO ANDRÉS HERRERA MENDOZA  
RICARDO CARAZO GARCÍA

Asesor:  
MSc. José Antonio Araque Gallardo

UNIVERSIDAD DE SUCRE  
FACULTAD DE INGENIERÍA  
TECNOLOGÍA EN ELECTRÓNICA INDUSTRIAL  
Sincelejo  
2016

## AGRADECIMIENTOS

A nuestro director de trabajo de grado, José Antonio Araque Gallardo, magíster en controles industriales. Quien con sus conocimientos, experiencia y paciencia nos ayudó en la realización del presente trabajo y por consiguiente, a la culminación satisfactoria de nuestros estudios.

Los autores

## RESUMEN

El presente trabajo se realizó en los laboratorios de electrónica de la Universidad de Sucre en la ciudad de Sincelejo durante el periodo comprendido entre el segundo semestre del año 2015 y el primer semestre del año 2016. El propósito de este trabajo consistió en el diseño y realización de un prototipo de una estación meteorológica de bajo costo capaz de medir la temperatura ambiente, la presión atmosférica, la humedad relativa, la velocidad y dirección del viento y la altitud, con sistema de envío de datos por SMS a un móvil o por comunicación serial o GPRS a un ordenador con Windows. En el caso del móvil, los datos son gestionados por una aplicación para Android hecha con la herramienta *App Inventor* de *Google*, y, en el caso del ordenador, por una aplicación hecha en *Lenguaje C#*. Independientemente del dispositivo, los datos son visualizados y tabulados en un fichero CSV o de Excel. Para la realización de la estación, primero, se hizo necesario la definición de los requerimientos técnicos específicos del prototipo, luego, se determinó la instrumentación a implementar y la integración de la misma en el sistema, y, finalmente, se desarrolló el *firmware* de los sistemas embebidos y de las aplicaciones para la gestión de datos en la computadora y el dispositivo móvil. El presente trabajo se justificó en base a su utilidad práctica, valor teórico y conveniencia económica respecto al costo que supondría la adquisición de un sistema comercial para el mismo fin. Se basa principalmente en el empleo de una tarjeta *CY8CKIT-042 PSoC 4 Pioneer Kit* de *Cypress* como unidad de control y un módulo GSM/GPRS (de referencia *M95* fabricado por *Quectel*) para el envío de datos inalámbricamente. En síntesis, el resultado obtenido no fue más que la realización satisfactoria del prototipo previamente descrito.

*Palabras claves:* prototipo de estación meteorológica, PSoC, GSM, sistemas embebidos, Android, Windows, C#.

## Tabla de Contenido

AGRADECIMIENTOS.....	2
RESUMEN.....	3
Capítulo 1 INTRODUCCIÓN .....	12
1.1    Objetivos.....	15
1.1.1    General.....	15
1.1.2    Específicos.....	15
Capítulo 2 ESTADO DEL ARTE.....	16
2.1    Antecedentes.....	16
2.1.1    Construcción de una estación meteorológica de acceso remoto (EMAR).....	16
2.1.2    Estación meteorológica inalámbrica.....	16
2.2    Bases teóricas.....	17
2.2.1    GSM (Global System for Mobile Communications). ....	17
2.2.2    Familia PSoC 4200.....	17
2.2.3    I2C (Inter-Integrated Circuit).....	19
2.2.4    UART (Universal Asynchronous Receiver/Transmitter). ....	22
2.2.5    TCP/IP (Transmission Control Protocol/Internet Protocol). ....	22
2.2.6    Microcontroladores PIC. ....	23
Capítulo 3 DISEÑO DEL PROTOTIPO DE ESTACIÓN METEOROLÓGICA.....	24
3.1    Descripción general y criterios de diseño.....	24
3.2    CY8CKIT-042.....	27
3.2.1    Descripción.....	28
3.2.2    Alimentación. ....	29
3.2.3    Interfaces de usuario. ....	29
3.2.4    Puente USB-UART. ....	31
3.3    Medición de la presión atmosférica, la temperatura y la altitud.....	32
3.3.1    Características.....	33
3.3.2    Asignación de pines.....	33
3.3.3    Circuito típico de conexión. ....	34
3.3.4    Direccionamiento como esclavo.....	35

3.3.5	Contenido de la EEPROM.....	35
3.3.6	Modos de operación.....	36
3.3.7	Obtención de la presión, la temperatura y la altitud.....	37
3.3.8	Diagrama de flujo.....	39
3.3.9	Conectar BMP085 a CY8CKIT-042.....	41
3.4	Medición de la humedad relativa.....	42
3.4.1	Especificaciones técnicas y características.....	42
3.4.2	Asignación de pines y circuito típico de conexión.....	43
3.4.3	Obtención de la humedad relativa y de la temperatura.....	43
3.4.4	Diagrama de flujo.....	45
	Conectar DHT11 a CY8CKIT-042.....	47
3.5	Medición de la velocidad del viento.....	48
3.5.1	Descripción.....	48
3.5.2	Principio de funcionamiento.....	49
3.5.3	Conexión del Anemómetro con la tarjeta CY8CKIT-042.....	50
3.5.4	Diagrama de flujo.....	51
3.6	Medición de la dirección del viento.....	52
3.6.1	Descripción.....	52
3.6.2	Principio de funcionamiento.....	53
3.6.3	Conexión con la tarjeta CY8CKIT-042.....	53
3.6.4	Lectura del sensor.....	54
3.7	Comunicación con el ordenador.....	55
3.8	Módulo de visualización.....	55
3.8.1	Descripción.....	55
3.8.2	Diagrama esquemático.....	57
3.8.3	Funcionamiento.....	58
3.8.4	Diagrama de flujo.....	58
3.8.5	Conexión del módulo de visualización con la tarjeta CY8CKIT-042.....	61
3.9	Módulo de comunicación.....	61
3.9.1	Descripción.....	62
3.9.2	Conexión del módulo de comunicación a la tarjeta CY8CKIT-042.....	67
3.9.3	Funcionamiento.....	68
3.10	Teclado.....	72

3.11	Diseño del firmware para la tarjeta CY8CKIT-042.....	73
3.11.1	Consideraciones.....	73
3.11.2	Descripción.....	74
3.12	Diseño de la aplicación en C#.....	80
3.12.1	Consideraciones.....	80
3.12.2	Descripción.....	81
3.12.3	Configurar red del equipo anfitrión.....	85
3.13	Diseño de la aplicación para Android.....	91
3.13.1	Consideraciones.....	92
3.13.2	Descripción.....	92
Capítulo 4 RESULTADOS Y ANÁLISIS.....		93
4.1	Envío de datos a través de comunicación serial.....	94
4.2	Envío de datos a través de GPRS.....	95
4.3	Envío de datos a través de SMS.....	97
4.4	Exportando datos.....	98
4.5	Fiabilidad de los datos.....	100
4.6	Análisis económico.....	101
Capítulo 5 CONCLUSIONES.....		103
Capítulo 6 RECOMENDACIONES.....		104
6.1	Técnicas.....	104
6.2	Para trabajos futuros.....	104
GLOSARIO.....		106
REFERENCIAS.....		107

## Índice de Figuras

Figura 2-1. Familia PSoC 4200, diagrama de bloques internos .....	18
Figura 2-2. Estructura del bus I2C .....	20
Figura 2-3. Transferencia completa de datos .....	21
Figura 3-1. Diagrama de bloques del prototipo de estación meteorológica .....	27
Figura 3-2. CY8CKIT-042, detalles .....	28
Figura 3-3. CY8CKIT-042, diagrama de bloques .....	29
Figura 3-4. CY8CKIT-042, puente USB-UART .....	31
Figura 3-5. Sensor BMP085, aspecto .....	32
Figura 3-6. Sensor BMP085, diagrama de pines .....	33
Figura 3-7. Sensor BMP085, circuito típico de conexión.....	34
Figura 3-8. Sensor BMP085, ejemplo de inicio de la medición de la presión en el modo standard .....	37
Figura 3-9. Sensor BMP085, ejemplo de lectura desde el bit MSB .....	38
Figura 3-10. Sensor BMP085, compensar temperatura (código en lenguaje C) .....	38
Figura 3-11. Sensor BMP085, compensar presión (código en Lenguaje C) .....	39
Figura 3-12. Diagrama de flujo seguido para comunicarse con el sensor BMP085.....	40
Figura 3-13. Sensor BMP085, conexión con la tarjeta CY8CKIT-042 .....	41
Figura 3-14. Sensor DHT11, aspecto.....	42
Figura 3-15. Sensor DHT11, circuito típico de conexión.....	43
Figura 3-16. Sensor DHT11, protocolo de transmisión.....	44
Figura 3-17. Sensor DHT11, diagrama de flujo seguido para recuperar los datos del sensor.....	46
Figura 3-18. Sensor DHT11, diagrama de flujo de la interrupción .....	47
Figura 3-19. Sensor DHT11, conexión con la tarjeta CY8CKIT-042 .....	48
Figura 3-20. Anemómetro, aspecto.....	49
Figura 3-21. Anemómetro, descripción .....	49
Figura 3-22. Anemómetro, conexión con la tarjeta CY8CKIT-042 .....	51
Figura 3-23. Anemómetro, diagrama de flujo .....	52
Figura 3-24. Veleta, detalles .....	53
Figura 3-25. Veleta, aspecto .....	53

Figura 3-26. Veleta, conexión con la tarjeta CY8CKIT-042.....	54
Figura 3-27. Módulo de visualización .....	56
Figura 3-28. PIC16F877A-20, aspecto físico .....	56
Figura 3-29. GLCD 128x64.....	56
Figura 3-30. Módulo de visualización, diagrama esquemático .....	57
Figura 3-31. Módulo de visualización, diagrama de flujo de la función main .....	59
Figura 3-32. Módulo de visualización, diagrama de flujo de la interrupción.....	60
Figura 3-33. Módulo de visualización, conexión con la tarjeta CY8CKIT-042.....	61
Figura 3-34. M95, aspecto .....	62
Figura 3-35. Módulo de comunicación.....	62
Figura 3-36. Módulo de comunicación, alimentación .....	63
Figura 3-37. Módulo de visualización, tarjeta SIM .....	64
Figura 3-38. Módulo de comunicación, antena .....	64
Figura 3-39. Módulo de comunicación, circuito de acople para las interfaces seriales.....	65
Figura 3-40. Módulo de comunicación, encendido/apagado .....	66
Figura 3-41. Módulo de comunicación, Netlight/Status.....	67
Figura 3-42. Módulo de comunicación, conexión con la tarjeta CY8CKIT-042 .....	68
Figura 3-43. Módulo de comunicación, aplicación TCP .....	71
Figura 3-44. Teclado, circuito de conexión con la tarjeta CY8CKIT-042 .....	72
Figura 3-45. Diagrama de flujo de la función Main .....	75
Figura 3-46. Diagrama de flujo de la función para enviar los datos a través de comunicación serial RS232.....	77
Figura 3-47. Diagrama de flujo de la función para el envío de datos a través de GPRS.....	78
Figura 3-48. Diagrama de flujo de la función para el envío de datos a través de SMS.....	79
Figura 3-49. Interfaz Estación Meteorológica versión 2 .....	81
Figura 3-50. Interfaz Estación Meteorológica, barra de menú .....	83
Figura 3-51. Interfaz Estación Meteorológica, gráfica .....	83
Figura 3-52. Interfaz Estación Meteorológica, tabla .....	84
Figura 3-53. Interfaz Estación Meteorológica, panel.....	84
Figura 3-54. Interfaz Estación Meteorológica, barra de estado .....	85
Figura 3-55. Propiedades del adaptador de red.....	87



Figura 3-56. Configurar IP privada como estática.....	87
Figura 3-57. Mapear puerto a la aplicación servidor .....	89
Figura 3-58. Validar uso del puerto por la aplicación servidor en el Firewall de Windows .....	90
Figura 3-59. DUC v4.1.0 .....	91
Figura 3-60. Aplicación móvil del prototipo de estación meteorológica.....	92
Figura 4-1. Prototipo de estación meteorológica .....	93
Figura 4-2. Prototipo de estación meteorológica, envío de datos a través de comunicación serial .....	94
Figura 4-3. Prototipo de estación meteorológica, envío de datos a través de GPRS .....	96
Figura 4-4. Prototipo de estación meteorológica, envío de datos a través de SMS .....	98
Figura 4-5. Datos tabulados en un fichero de Excel de pruebas realizadas el 20 de julio de 2016 .....	99
Figura 4-6. Datos tabulados en un fichero CSV .....	99
Figura 4-7. Tabla del IDEAM de la precipitación y temperatura para el 20 de julio de 2016 ...	101

## Índice de Tablas

Tabla 3-1. Distribución de pines para el sensor BMP085 .....	33
Tabla 3-2. Sensor BMP085: direccionamiento como esclavo .....	35
Tabla 3-3. Sensor BMP085: contenido de la EEPROM.....	35
Tabla 3-4. Sensor BMP085: modos de operación .....	36
Tabla 3-5. Sensor BMP085: escritura.....	36
Tabla 3-6. Sensor BMP085: lectura.....	37
Tabla 3-7. Sensor DHT11: distribución de pines.....	43
Tabla 3-8. Secuencia de inicio del módulo de comunicación.....	69
Tabla 3-9. Comandos básicos para la aplicación SMS .....	69
Tabla 3-10. Comandos básicos para la aplicación TCP.....	70
Tabla 4-1. Costo de materiales.....	102

## **Capítulo 1**

### **Introducción**

La posición geográfica de Colombia posibilita que en este país se presenten una gran variedad de climas a lo largo de todo su territorio. Si bien a diario se reciben a través de los medios de información un reporte del estado del tiempo facilitado por el Instituto de Hidrología, Meteorología y Estudios Ambientales (IDEAM), esta información no representa el conjunto de factores atmosféricos que caracterizan un entorno reducido.

El interés en el estudio del clima radica en que para el desarrollo de ciertas actividades económicas como la agricultura, la industria, la navegación, la investigación, entre otras; es muy importante tener conocimientos acertados sobre el comportamiento de éste, ya que de este conocimiento se fundamentan decisiones de trascendencia e importancia en aspectos relacionados con dichas labores. Por ejemplo: en la aviación se sustentan las decisiones que pueden llevar a los pilotos a tomar rutas distintas según el tiempo que se espera; en la agricultura (una de las principales actividades económicas del país y del departamento de Sucre) facilita la toma de decisiones para llevar a cabo la realización del cultivo y las distintas actividades para su sostenimiento, lo que garantiza una mayor probabilidad de éxito en el mismo. Sin embargo, la falta o distribución irregular de estaciones meteorológicas en la región no permite la elaboración de un modelo numérico del comportamiento del clima que permita hacer predicciones meteorológicas.

Aunque la motivación y razón para la realización del prototipo de la estación que describe este documento no pretende satisfacer específicamente ninguna de las actividades expuestas anteriormente (por ser precisamente un prototipo experimental), mediante el uso de la misma se podría realizar un monitoreo de los factores climáticos locales; un análisis de los datos obtenidos

de éste permitiría la elaboración de un modelo del comportamiento del clima a nivel local, que favorecería a las personas o entidades que a partir de ciertas condiciones climáticas deban tomar decisiones para satisfacer algún requerimiento.

Además de la utilidad práctica comentada en el párrafo anterior, la realización de este proyecto también posee valor teórico en el manejo de las tecnologías de vanguardia empleadas en el mismo, como lo son el manejo de sistemas embebidos y la conexión de dispositivos o sistemas a Internet.

Si bien se consiguen en el mercado sofisticadas estaciones meteorológicas que hacen uso de sistemas avanzados para el cálculo exacto de los distintos factores que afectan el estado de la atmósfera en un momento y lugar determinado (como es el caso de las estaciones meteorológicas automáticas en las cuales no se necesita de una persona que esté tomando los datos diariamente a determinada hora, sino que éstos son procesados, almacenados y transmitidos a un servidor local), la adquisición, instalación y mantenimiento de éstas representa un costo excesivo. El prototipo de estación meteorológica que describe el presente proyecto es una opción considerablemente económica respecto a una estación comercial.

El presente proyecto apuntó a ser el primer paso para la realización de sistemas locales de bajo costo que satisfagan las necesidades expuestas anteriormente. El objetivo general fue la realización de un prototipo de una estación meteorológica capaz medir las variables físicas más comunes, específicamente: la temperatura ambiente, la humedad relativa, la presión atmosférica, la rapidez y dirección del viento y la altitud. Además, debía implementar un sistema propio para exportar los datos obtenidos a un ordenador o móvil, donde pueden ser visualizados y almacenados para su posterior análisis. El resultado no fue menos que el propuesto. Para la realización del

mismo se requirió principalmente el manejo de sistemas embebidos, y, por consiguiente, el manejo de *Lenguaje C* y de diferentes *IDE's* de programación. Así como el aprendizaje de programación orientada a objetos bajo Windows y conocimientos en el área de instrumentación, comunicaciones digitales y de electrónica en general.

## 1.1 Objetivos

### 1.1.1 General

Diseñar e implementar un prototipo de estación meteorológica con adquisición de datos inalámbrica usando módulos PSoC (Programmable System on Chip) y tecnología móvil de segunda generación.

### 1.1.2 Específicos

- Establecer los requerimientos técnicos específicos para el prototipo.
- Determinar la instrumentación a implementar y la integración de la misma en el sistema.
- Medir la temperatura ambiente, la presión atmosférica, la altitud sobre el nivel del mar, la humedad relativa y la velocidad y dirección del viento.
- Realizar un subsistema para la visualización de datos de los procesos internos del prototipo.
- Realizar un subsistema para el envío de datos inalámbricamente a través de la red de telefonía celular o de forma alámbrica a través de comunicación serial.
- Desarrollar el firmware del dispositivo embebido.
- Desarrollar el software interfaz para la gestión de datos en la computadora y/o el dispositivo móvil.
- Realizar pruebas de campo y verificar la fiabilidad de los datos de las variables medidas y el correcto funcionamiento de todos los subsistemas.

## Capítulo 2

### Estado del Arte

#### 2.1 Antecedentes

Referente al proyecto de investigación que se realizó no se encontraron antecedentes o estudios similares realizados utilizando la misma tecnología en el departamento de Sucre. A nivel nacional se encontraron investigaciones que tuvieron como objetivo principal la realización de una estación meteorológica con características y funciones similares a las que describen el presente trabajo. Éstas son:

##### **2.1.1 Construcción de una estación meteorológica de acceso remoto (EMAR).**

Presentado en la Fundación Tecnológica Antonio de Arévalo (Cartagena de Indias, Colombia) por Genadiz Cardona, G. y Bolaños, M., en 2014. Esta estación permite reconocer el estado climático y atmosférico en los alrededores de la Fundación Tecnológica Antonio de Arévalo a través del uso de sensores y monitoreo vía web (Fundación Tecnológica Antonio de Arévalo [TECNAR], 2015).

##### **2.1.2 Estación meteorológica inalámbrica.**

Presentado en la Universidad Manuela Beltrán (Bogotá, Colombia) por Arango Parrado, Daniel A.; Rincón Infante, Freddy E.; Manrique Daza, Lisandro; Buitrago Rojas, Diana C., en diciembre de 2010. En este proyecto se implementó una estación meteorológica para monitorear cinco variables físicas: la temperatura, la presión atmosférica, la humedad relativa y la dirección y velocidad del viento. El sistema utiliza un microcontrolador *PIC* en el que se procesan las señales pre-acondicionadas de los sensores adjuntos y un celular en el cual se obtienen los valores medidos

de las variables climáticas estudiadas a través del servicio SMS; estos datos son registrados y visualizados (Arango, Rincón, Manrique, & Buitrago, 2010).

El diseño del prototipo de estación meteorológica que describe el presente documento no se basa en ninguno de los procedimientos o métodos seguidos para el desarrollo de las estaciones meteorológicas anteriormente citadas.

## **2.2 Bases teóricas**

### **2.2.1 GSM (Global System for Mobile Communications).**

En síntesis, GSM (originalmente *Groupe Spécial Mobile*) es un estándar desarrollado para las comunicaciones móviles. Se denomina de segunda generación (o 2G) porque a diferencia de la primera generación la comunicación entre los dispositivos móviles se produce de un modo digital. Soporta el *roaming*, la transmisión de voz y de datos de pequeño volumen (envío de mensajes de texto a través de *Short Message Service* o SMS). GPRS (*General Packet Radio Service*) es una extensión para GSM que permite la transmisión de datos mediante la conmutación de paquetes (CCM, s.f.).

### **2.2.2 Familia PSoC 4200.**

PSoC 4 (PSoC o *Programmable System on Chip*) es como se designa a la arquitectura de sistemas embebidos fabricada por *Cypress* ([www.cypress.com](http://www.cypress.com)). La familia PSoC 4200 describe a los productos o chips basados en la arquitectura antes mencionada. Los chips de ésta familia consisten en un microcontrolador de lógica programable que integra bloques analógicos y digitales y periféricos e interfaces de comunicación estándar, configurables mediante software. De ésta manera, la principal característica de estos dispositivos es la flexibilidad con la que ofrecen al



diseñador la posibilidad de satisfacer los requerimientos de muchas de las aplicaciones que involucran sistemas embebidos. La Figura 2-1 ilustra el diagrama de bloques internos para ésta familia.

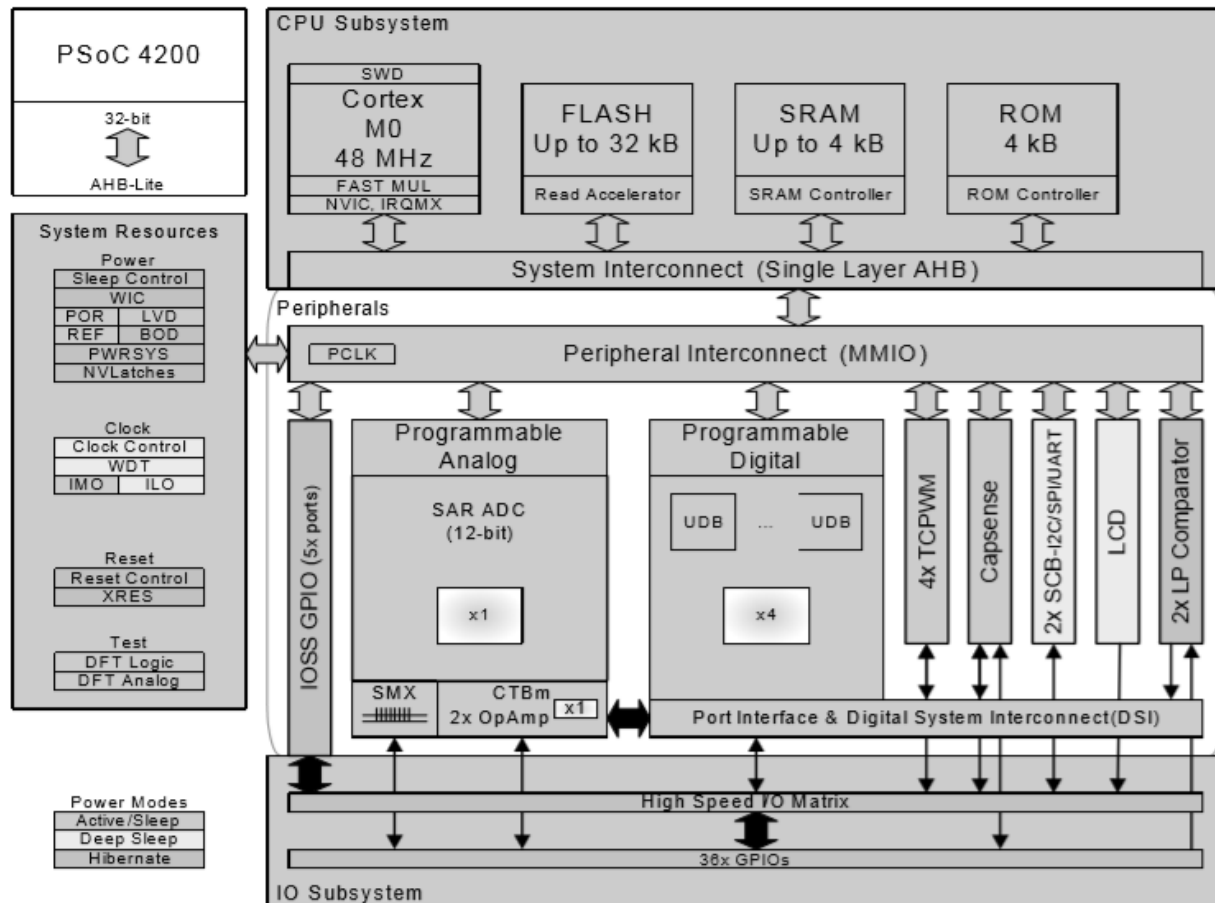


Figura 2-1. Familia PSoC 4200, diagrama de bloques internos

Fuente: adaptada de (Cypress Semiconductor Corporation, 2016, pág. 4)

Entre las principales características de la familia PSoC 4200 están (Cypress Semiconductor Corporation, 2016):

- CPU: reloj de 48 MHz, 32 KB de memoria *flash*, 4 KB de memoria SRAM e instrucciones de 16 bits. Se puede usar parte de la memoria flash para emular mediante código una EEPROM si es necesario.

- Integración de bloques analógicos: ADC SAR de 12 bits y 1-Msbps, 2 amplificadores operacionales, sensor de temperatura, dos comparadores de bajo consumo.
- Integración de bloques digitales: 4 UDB's (*Universal Digital Blocks*) e interfaces o puertos de comunicación.
- Puede suplir funciones de Temporizador, Contador, PWM y comunicación serial mediante I2C, UART y SPI.
- Cada pin o interfaz atiende a diferentes configuraciones para su operación (Análogo, Digital, entre otros) editadas mediante software.
- Funciones especiales para operar periféricos: *LCD segment driver* y *CapSense*.
- Bajo nivel de operación (1.71V – 5.5V).
- Cinco diferentes encapsulados: 48-pin TQFP, 44-pin TQFP, 40-pin QFN, 35-ball WLCSP y 28-pin SSOP.

### 2.2.3 I2C (Inter-Integrated Circuit).

Es una interfaz en forma de bus serie desarrollada por *Philips Semiconductors* ([www.semiconductors.philips.com](http://www.semiconductors.philips.com)). Emplea dos líneas para la comunicación entre dos o más dispositivos compatibles, atendiendo a una serie de procedimientos para el buen funcionamiento del sistema. Soporta velocidades de 100 Kbps (*Standard Mode*), 400 Kbps (*Fast Mode*) y más de 3,4 Mbps (*High-Speed Mode*). Normalmente las funciones que suelen realizar los periféricos asociados al bus no demandan una alta tasa de transferencia de datos. El sistema dispone, como mínimo, de un microcontrolador y de un periférico asociado. El bus está diseñado de tal modo que permite hacer fácilmente modificaciones al sistema de comunicación sin afectar a los dispositivos

previamente conectados. La Figura 2-2 ilustra la estructura del bus I2C. El protocolo seguido es el de maestro/esclavo (*master/slave*), nótese que puede haber más de un maestro conectado al bus (Palacios, Remiro, & López, 2004).

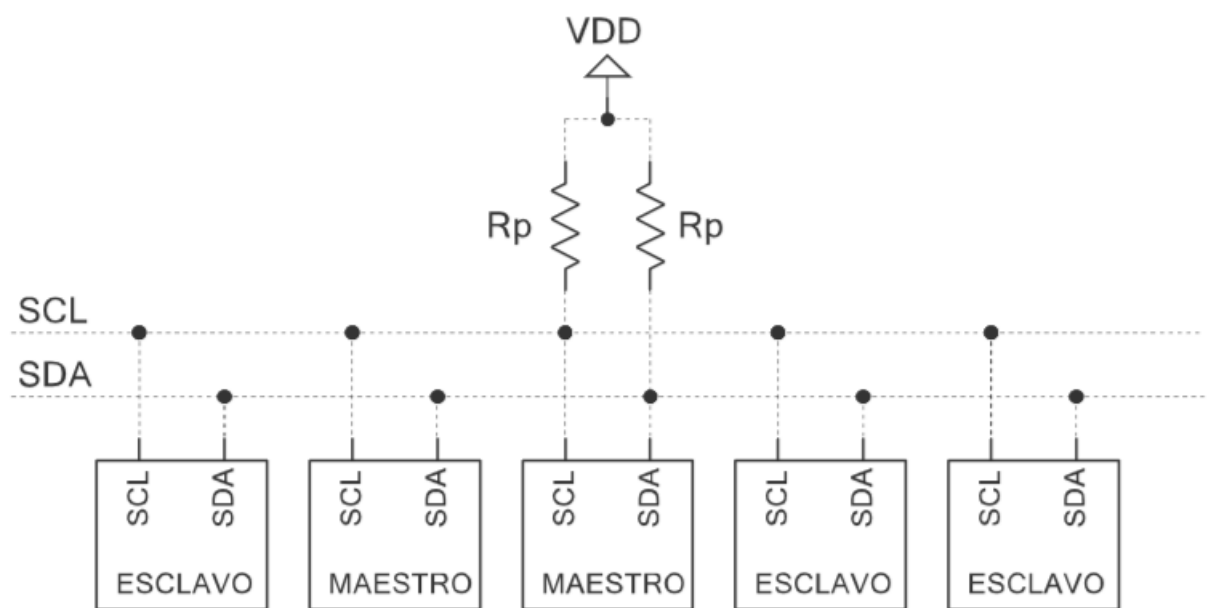


Figura 2-2. Estructura del bus I2C  
Fuente: basada en (Palacios, Remiro, & López, 2004, pág. 332).

La línea SCL (*Serial Clock*) sincroniza el envío de datos a través de la línea SDA (*Serial Data*), que es bidireccional. El dispositivo maestro debe iniciar y terminar la transferencia de datos y sincronizar y direccionar al dispositivo esclavo. El valor de las resistencias  $R_p$  es función del número de dispositivos conectados y de VCC, normalmente éste es de 4K7 en la mayoría de los casos. Los dispositivos que soportan esta interfaz poseen una dirección de siete bits de largo mediante la cual el maestro identifica a cuál de los periféricos asociados al bus se va a dirigir (Palacios, Remiro, & López, 2004).

La transferencia de datos entre un maestro y un esclavo puede resumirse en los siguientes pasos ilustrados en la Figura 2-3 (Palacios, Remiro, & López, 2004):

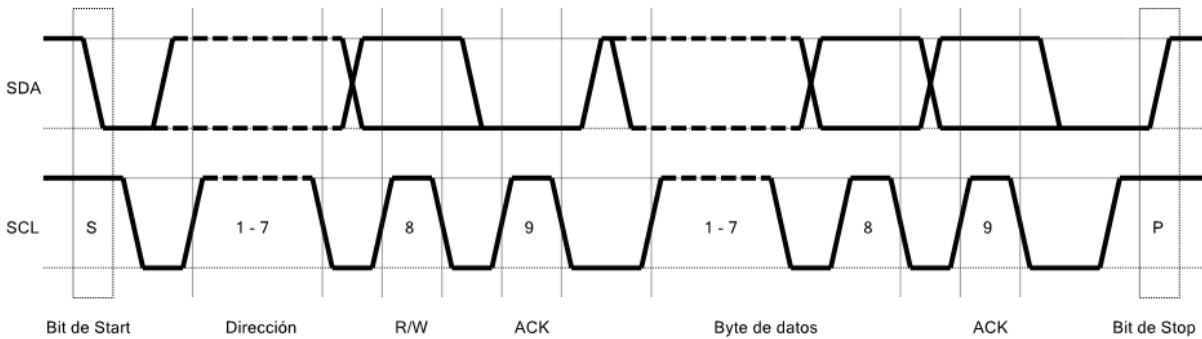


Figura 2-3. Transferencia completa de datos  
Fuente: basada en (Palacios, Remiro, & López, 2004, pág. 337)

- El maestro comprueba que el bus no esté ocupado.
- Inicio de la transferencia; para ello, el maestro pone a SDA en flanco de bajada mientras SCL está a nivel alto (bit de *Start*). Entonces el bus se considerará ocupado.
- El maestro realiza el direccionamiento del esclavo a partir de siete bits que identifican a éste en el bus.
- El maestro envía un bit de R/W (*Read/Write*) que determina si el esclavo es receptor ( $R/W = 0$ ) o transmisor ( $R/W = 1$ ).
- El receptor envía un bit de reconocimiento o ACK (*acknowledgement*) poniendo la SDA a nivel bajo en el pulso de reloj correspondiente. Éste bit se envía luego de cada byte transmitido.
- El receptor envía un bit por cada pulso de reloj hasta completar un byte que representa los datos transmitidos.

- El maestro pone a SDA en flanco de subida mientras SCL está a nivel alto (bit de *Stop*).

#### **2.2.4 UART (Universal Asynchronous Receiver/Transmitter).**

Es un componente para la comunicación serial asíncrono entre equipos terminales compatibles, basándose comúnmente en el estándar RS232. Antes de transmitir los datos es necesario configurar la palabra de control en el registro correspondiente del bloque UART para indicar la naturaleza de los datos, como la velocidad de transmisión, el formato de los datos, si hay o no paridad y el tipo de esta, y los bits de parada. Como es de suponerse, ésta debe tener la misma configuración en los equipos que se comunican. Ésta configuración se hace a través del firmware que ejecuta el sistema donde se encuentra embebido el componente. El componente UART puede ser configurado para dúplex, semidúplex, solamente RX o TX.

#### **2.2.5 TCP/IP (Transmission Control Protocol/Internet Protocol).**

TCP/IP es una arquitectura que consta de protocolos estándares bajo los que opera Internet y que se pueden agrupar u organizar en cinco capas relativamente independientes de acuerdo a las funciones que éstas realicen en el proceso de comunicación a través de la red. De menor a mayor nivel, estas capas son (Stallings, 2000, pág. 52):

- Capa física: define la interfaz física entre el dispositivo transmisor de datos y la red.
- Capa de enlace: se ocupa de las funciones concernientes al intercambio de datos entre el sistema final y la subred a la cual está conectado.
- Capa de red: realiza el encaminamiento de datos desde el origen hasta el destino a través de varias redes conectadas por dispositivos de enrutamiento.

- Capa de transporte: proporciona servicios para la transferencia de datos extremo a extremo.
- Capa de aplicación: proporciona la comunicación entre procesos de dispositivos separados.

Cada capa define una serie de protocolos que intervienen en las funciones que cumple la misma, y cada una de ellas interacciona con las capas adyacentes. La información pertinente al tema es extensa y para este apartado se decidió no entrar en detalle; pero, básicamente, para conectarse a Internet todo dispositivo (computador, router, etc.) debe tener una única dirección que permita identificarlo en el sistema global (LAN o WAN), y cada proceso de la capa de aplicación que se ejecute en dicho dispositivo debe tener a su vez una dirección o puerto que lo identifique dentro del mismo para que pueda realizarse la transferencia de datos extremo a extremo proporcionada por la capa de transporte. Los dispositivos de enrutamiento sólo implementan hasta la capa de red. Así, el protocolo IP de la capa de red se implementa en todos los dispositivos finales y de encaminamiento. Por su parte, TCP de la capa de transporte sólo se implementa en los dispositivos finales.

### **2.2.6 Microcontroladores PIC.**

Los microcontroladores PIC (*Peripheral Interface Controller*) pertenecen a una serie de dispositivos fabricados por *Microchip Technology Inc.*, ([www.microchip.com](http://www.microchip.com)). Son unos circuitos integrados con la arquitectura de una computadora, esto es, CPU, RAM, EEPROM, buses e interfaces de entrada y salida. Estos dispositivos disponen de un set de instrucciones por medio de las cuales deben ser programados para que ejecuten una tarea en específico. Como la memoria de programa es de tipo *Flash*, pueden ser regrabados un número limitado de veces.

## Capítulo 3

### Diseño del Prototipo de Estación Meteorológica

Este capítulo está dedicado al diseño del prototipo de estación meteorológica. Presenta una descripción completa de los elementos y subsistemas que conforman a la misma atendiendo a la metodología seguida para su elaboración y la integración de la misma en el sistema. De igual forma, presenta una justificación para el uso de dichos elementos y los procedimientos o métodos que se siguieron.

#### 3.1 Descripción general y criterios de diseño

El objetivo general del presente proyecto señala el uso de sistemas embebidos PSoC (*Programmable System on Chip*) y tecnología celular de segunda generación como base en la construcción del prototipo. Atendiendo a este principio, se eligió a la tarjeta CY8CKIT-042 como unidad central de proceso y al módulo M95 para la etapa de comunicación inalámbrica. La tarjeta CY8CKIT-042 ejecuta el *firmware* que opera a los periféricos asociados que conforman a la estación, también procesa los datos provenientes de los sensores acerca del valor actual de las variables medidas, les da formato y los envía al módulo de comunicaciones M95 para que sean transmitidos inalámbricamente y posteriormente almacenados en el dispositivo remoto (celular u ordenador). También ofrece la opción de realizar el envío de los datos a un ordenador local de forma alámbrica mediante comunicación serial basada en el protocolo RS232. Por sesión, sólo se puede hacer uso de una de las tres formas para enviar los datos. La elección de la opción y cancelación de la misma se realiza a través de un teclado matricial 4x4.

El uso de sistemas embebidos PSoC se justifica por las altas prestaciones que ésta tecnología ofrece, su flexibilidad y su bajo costo. Integra bloques analógicos y digitales y

diferentes interfaces para la comunicación con el exterior suficientes para el desarrollo del prototipo. En cuanto al uso de GSM, aunque no es una tecnología reciente, se consideró factible su empleo en el prototipo como medio para exportar los datos inalámbricamente; usar la red de telefonía celular resuelve el problema de la distancia o el alcance a la hora de realizar dicha labor.

Por tratarse de un prototipo cuyo fin no es satisfacer las necesidades para el desarrollo de ninguna actividad comercial en específico, se acordó medir las variables físicas más comunes o que aplican a la mayoría de actividades. Éstas fueron la temperatura ambiente, la humedad relativa, la presión atmosférica, la rapidez y dirección del viento y, adicionalmente, la altitud.

Otro de los requerimientos para el desarrollo del prototipo es el empleo de hardware de bajo costo y con las prestaciones adecuadas. A la fecha, se encontraron en el mercado sensores con salida digital (el uso de sensores inteligentes ahorró realizar etapas para el acondicionamiento de señal a la salida de estos) económicos; con un rango de medición, resolución, precisión y exactitud satisfactorios; previamente calibrados o capaces de ejecutar sus propias rutinas de calibración y capaces de sensor más de una variable. Aunque su presentación no es lo suficientemente robusta para aplicaciones industriales o aplicaciones en los que el sensor es sometido a drásticas condiciones ambientales, se consideró acertado o adecuado el uso de estos sensores en el prototipo de estación meteorológica. Para medir la presión atmosférica se eligió el sensor BMP085, además de medir la variable ya mencionada también puede medir la temperatura ambiente, y, como la presión atmosférica y la altitud se relacionan, a partir de la primera se calcula el valor de la segunda por medio de una expresión matemática de uso general. Para medir la humedad relativa se eligió al sensor DHT11, que además puede medir la temperatura, y para medir la rapidez y dirección del viento se hizo necesario la construcción de un sensor Anemómetro y Velea que realizaran dicho cometido.



La estación debía ser capaz de interactuar con un celular bajo Android, para tal fin se elaboró una aplicación utilizando la herramienta *App Inventor*, de Google. A través de la aplicación se piden los datos de las variables a la estación usando SMS, ésta recibe la petición y envía la respuesta a través del módulo M95 al equipo remitente. Para solicitar el envío de datos el equipo móvil debe ser del mismo operador al que pertenece la *SIM Card* de la estación. La tarjeta CY8CKIT-042 dispone de un bloque UART, se aprovechó ésta característica para enviar los datos a un ordenador local bajo Windows. En el ordenador se ejecuta una aplicación hecha en C# encargada de recibir los datos de la estación, visualizarlos y tabularlos en un fichero de Excel. Como además el módulo M95 soporta GPRS, se optimizó el firmware de la estación y la aplicación interfaz en Windows para enviar y recibir (respectivamente) los datos a través de Internet.

Los procesos internos llevados a cabo por la estación y los mensajes de estado o eventos se muestran en el módulo de visualización, basado en un PIC16F877A-20 y una GLCD 128x64.

En el diseño no se consideró la realización de una fuente de alimentación dedicada a la estación. La CY8CKIT-042 puede ser alimentada a través de un puerto *USB* del ordenador o a través de una fuente externa. La Figura 3-1 ilustra el diagrama de bloques del prototipo de estación meteorológica, el módulo de alimentación que allí se ilustra corresponde a un adaptador que se recicló de un ordenador portátil en desuso.

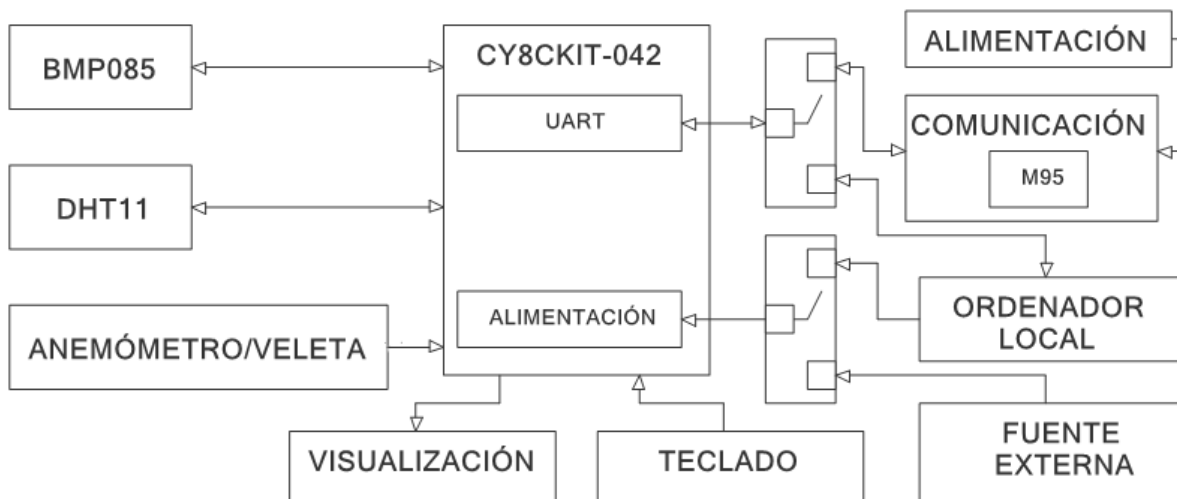


Figura 3-1. Diagrama de bloques del prototipo de estación meteorológica  
Fuente: elaboración propia.

En los subcapítulos siguientes a éste se presenta una descripción de cada uno de los elementos y funciones previamente mencionados/as y cómo se integraron los/as mismos/as al sistema.

### 3.2 CY8CKIT-042

La CY8CKIT-042 o PSoC 4 Pioneer Kit (Figura 3-2), es una tarjeta de desarrollo y prototipado basada en un chip de la familia PSoC 4200, fabricada por *Cypress* ([www.cypress.com](http://www.cypress.com)). Integra el hardware suficiente para suplir diversas aplicaciones usando sistemas embebidos.

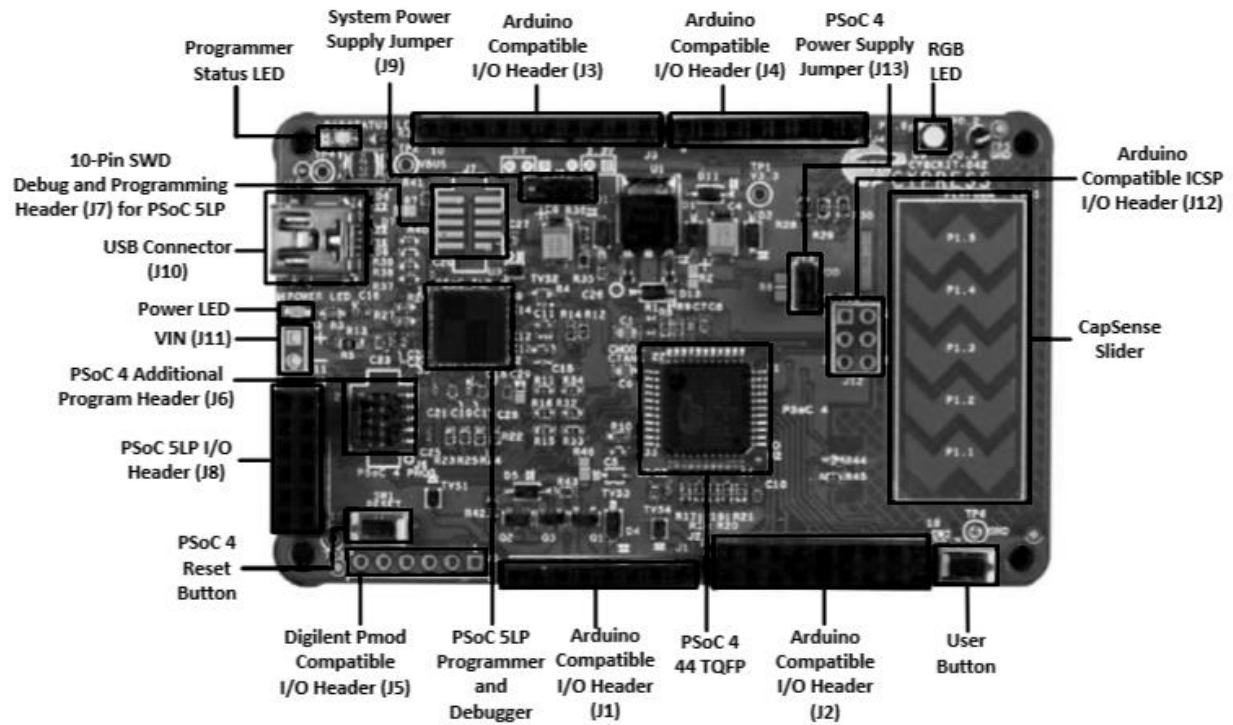


Figura 3-2. CY8CKIT-042, detalles

Fuente: adaptada de (Cypress Semiconductor Corporation, s.f., pág. 31)

### 3.2.1 Descripción.

La CY8CKIT-042 consta, en esencia, de un chip *44-pin TQFP* como unidad central de proceso, un programador basado en *PSoC 5LP*, un regulador *LDO* e interfaces de usuario (*Headers*, *CapSense Slider*, pulsadores y *LED's* de estado). La Figura 3-3 ilustra el diagrama de bloques de la tarjeta.

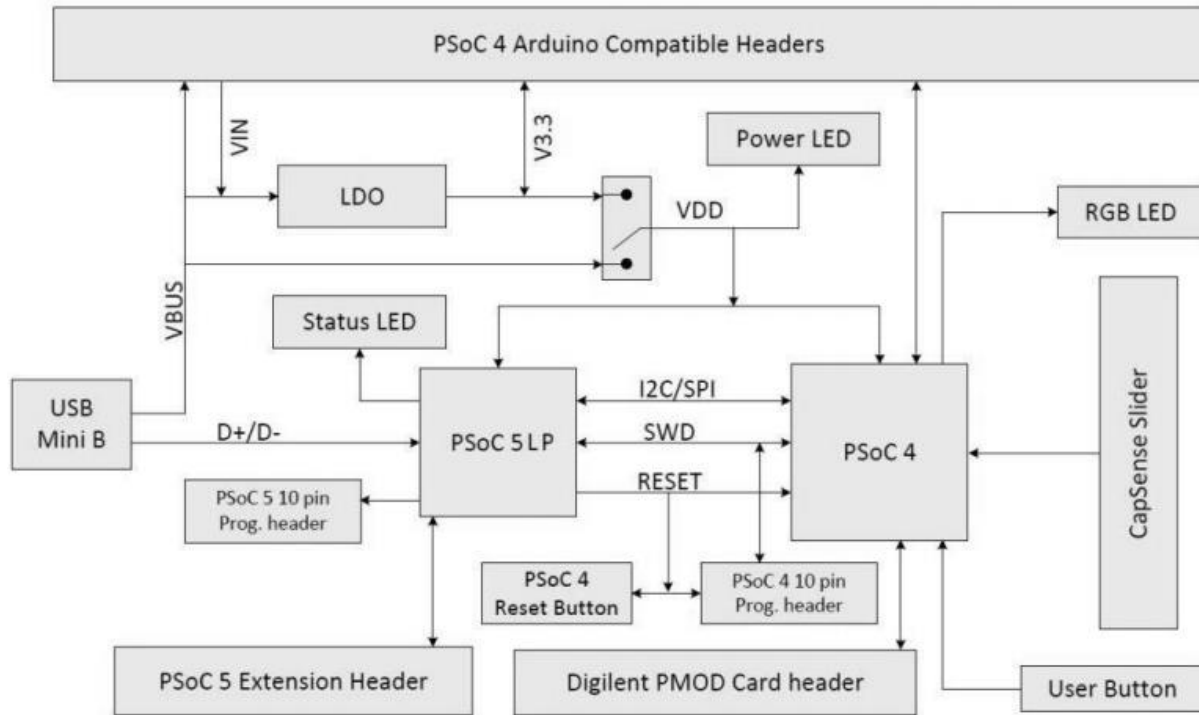


Figura 3-3. CY8CKIT-042, diagrama de bloques

Fuente: adaptada de (Cypress Semiconductor Corporation, s.f., pág. 33)

### 3.2.2 Alimentación.

La CY8CKIT-042 puede ser alimentada de tres formas diferentes:

- Con 5V a través del puerto USB adjunto a la tarjeta (J10, Figura 3-2).
- De 5V a 12V a través de VIN (pin 01 del puerto J1 o J11, Figura 3-2).
- A través del grabador externo, ya sea usando el puerto J6 o J7 (Figura 3-2).

La tarjeta trabaja con niveles de tensión de 3.3V o 5V, la selección se hace a través del puente o jumper J9 adjunto (Figura 3-2). Los niveles de tensión a través de los puertos de entrada y salida se corresponden con el nivel seleccionado. Además, la etapa de alimentación implementa un circuito de protección contra voltajes en reversa, sobretensión y cortocircuitos.

### 3.2.3 Interfaces de usuario.

La CY8CKIT-042 consta de trece *headers* para la comunicación con el exterior (refiérase a la Figura 3-2 para su identificación):

- J1: dispone de pines para la alimentación, voltajes de referencia, y *reset*. Compatible con *shields* Arduino.
- J2: éste puerto es análogo; dispone de pines para trabajar con el ADC, comparadores y amplificadores operacionales. Compatible con *shields* Arduino.
- J3: éste puerto es digital; dispone de pines para trabajar con PWM, I2C, SPI, entre otras. Compatible con *shields* Arduino.
- J4: éste puerto es digital; dispone de pines para trabajar con UART y PWM. Compatible con *shields* Arduino.
- J5: soporta interfaces Digilent Pmod.
- J6: programador y depurador adicional para PSoC 4.
- J7: programador y depurador para PSoC 5 LP.
- J8: habilitado para PSoC 5LP.
- J9: consiste en un *jumper* que se usa para seleccionar el nivel de voltaje con el que trabajará la tarjeta.
- J10: Conector USB para la programación y depuración, o alimentación.
- J11: VIN.
- J12: Arduino ICSP compatible. Sólo funciona en el modo de 5V.
- J13: jumper entre la alimentación para medir la corriente.

La CY8CKIT-042 además incluye las siguientes interfaces de usuario (Figura 3-2):

- CapSense Slider (CSS1) de cinco segmentos (para mayor información refiérase a la documentación de Cypress en [www.cypress.com](http://www.cypress.com)).
- LED verde, para indicar el estado del programador (D10).
- LED de power (D3).
- LED RGB de propósito general (D9).
- Pulsador para reset (SW1).
- Pulsador de propósito general (SW2).
- Cuatro tomas a negativo: TP3, TP4, TP5 y TP6.

### 3.2.4 Puente USB-UART.

El PSoC 5LP adjunto a la tarjeta también se puede usar como un puente USB-UART para la transferencia de datos entre PSoC 4 y el ordenador a través del terminal COM (Figura 3-4).

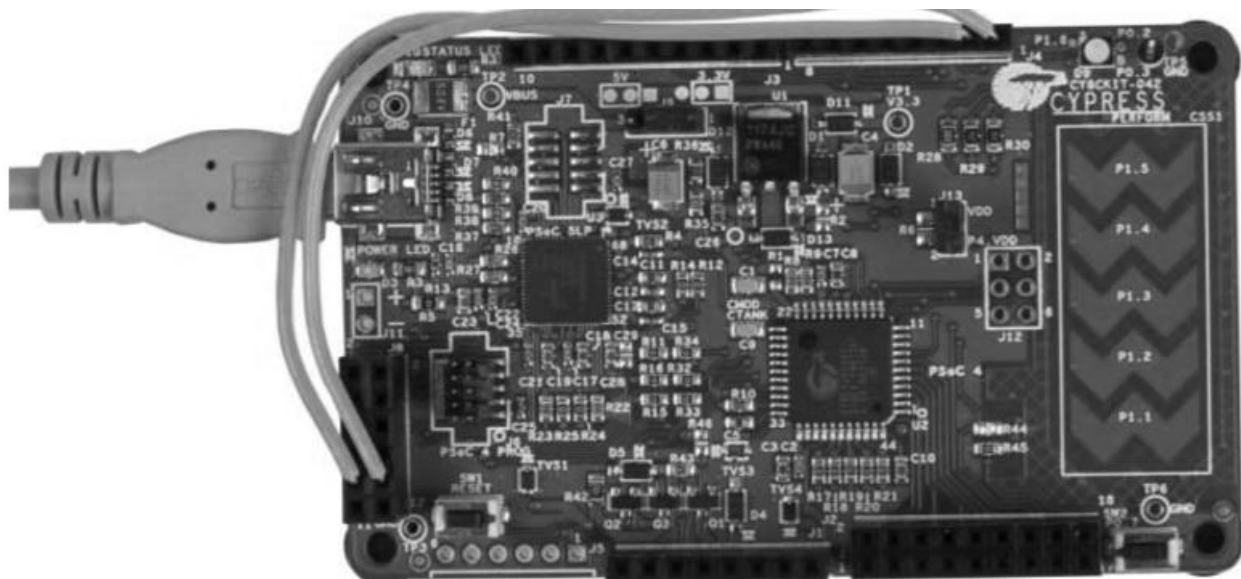


Figura 3-4. CY8CKIT-042, puente USB-UART

Fuente: adaptada de (Cypress Semiconductor Corporation, s.f., pág. 27)

Las especificaciones soportadas por el puente USB-UART son:

- Baud rate: 1200, 2400, 4800, 9600, 19200, 38400, 57600 y 115200.
- Formato de un byte.
- Sin paridad.
- Un bit de parada.
- Sin control de flujo.

### 3.3 Medición de la presión atmosférica, la temperatura y la altitud

Para la medición de la presión atmosférica y la temperatura ambiente se utilizó el sensor inteligente BMP085 (Figura 3-5). Como la presión atmosférica se relaciona con la altitud, a partir del valor obtenido de primera se calcula el valor de la segunda. Es fabricado por *Bosch Sensortec* ([www.bosch-sensortec.com](http://www.bosch-sensortec.com)). Entrega en la salida el valor digital obtenido de la medición de la presión atmosférica y de la temperatura ambiente del lugar en donde se encuentre ubicado el dispositivo, con una alta exactitud. Consiste de un sensor piezo-resistivo, un ADC, una unidad de control con una interfaz serial I2C a través de la cual se comunica con el exterior y una memoria EEPROM que almacena coeficientes de calibración usados para compensar el *offset* en el valor obtenido de las magnitudes físicas medidas y otros parámetros del sensor.



*Figura 3-5. Sensor BMP085, aspecto*  
*Fuente: (Bosch Sensortec, 2008, pág. 1)*

### 3.3.1 Características.

Entre las principales características para este sensor se pueden señalar (Bosch Sensortec, 2008):

- Voltaje de alimentación de 1.8V a 3.6V.
- Rango para la presión: 300 a 1100 hPa.
- Rango para la temperatura: - 40 a 85 °C.
- Resolución de la presión: 0.01 hPa.
- Resolución de la temperatura: 0.1 °C.
- Alta precisión.
- Interfaz de comunicación I2C.
- Bajo consumo.
- Presentación SMD.

### 3.3.2 Asignación de pines.

La Tabla 3-1 muestra la asignación de pines y la Figura 2-1 ilustra el diagrama de pines para el sensor BMP085.

Tabla 3-1  
Distribución de pines para el sensor BMP085

Pin. No.	Nombre	Función
1	GND	Power
2	EOC	Salida digital
3	VDDA	Power
4	VDDD	Power
5	NC	
6	SCL	Entrada digital
7	SDA	Digital bidireccional
8	XCLR	Entrada digital

Nota: NC es no conectado.

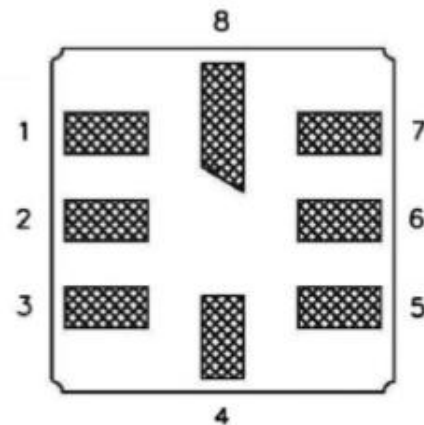


Figura 3-6. Sensor BMP085, diagrama de pines  
Fuente: adaptada de (Bosch Sensortec, 2008, pág. 18)



Fuente: (Bosch Sensortec, 2008)

### 3.3.3 Circuito típico de conexión.

La Figura 3-7 ilustra el circuito típico de conexión. VDDA y VDDD pueden operar al mismo nivel de tensión, el condensador de 100 nF es de desacople, en caso de que VDDA y VDDD se alimenten con fuentes diferentes se debe colocar un condensador de desacople en ambos pines. Las líneas SDA, SCL y las resistencias  $R_p$  se corresponden con el protocolo I2C, el XCLR o master clear puede dejarse flotando y el terminal EOF (*End of Conversion*) avisa a la unidad maestra de que puede leer los datos del sensor.

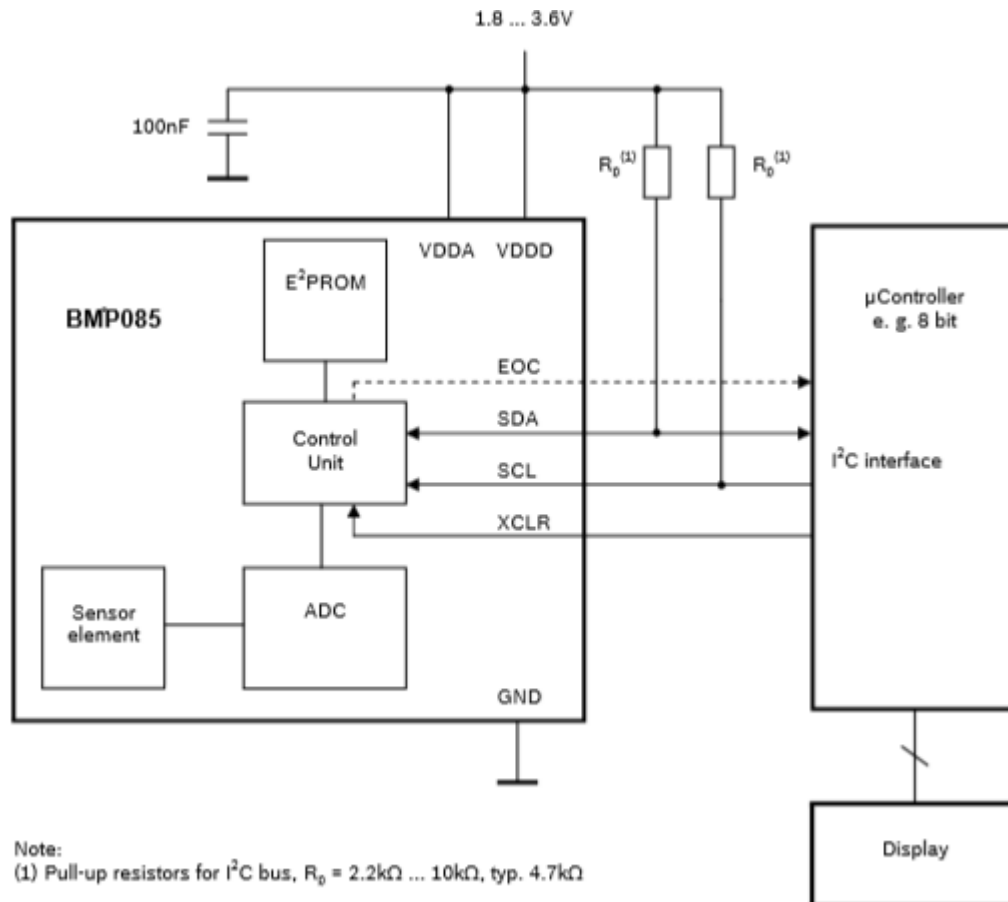


Figura 3-7. Sensor BMP085, circuito típico de conexión  
 Fuente: adaptada de (Bosch Sensortec, 2008, pág. 9)

### 3.3.4 Direccionamiento como esclavo.

Al igual que cualquier otro dispositivo con interfaz serial I2C, el BMP085 se activa cuando recibe la dirección de la Tabla 3-2.

*Tabla 3-2  
Sensor BMP085: direccionamiento como esclavo*

A7	A6	A5	A4	A3	A2	A1	R/W
1	1	1	0	1	1	1	1/0

*Fuente: adaptado de (Bosch Sensortec, 2008)*

Así, atendiendo a las especificaciones técnicas del Bus I2C, para realizar las operaciones de lectura y escritura se deben usar las direcciones 0xEF y 0xEE respectivamente, que se obtienen a partir de los posibles estados del bit de control (LSB).

### 3.3.5 Contenido de la EEPROM.

La EEPROM del BMP085 almacena 176 bits, éstos están particionados en once palabras de 16 bits cada una, y representan a los coeficientes de calibración necesarios para corregir el offset en el valor obtenido de las magnitudes medidas y otros parámetros del sensor. La Tabla 3-3 muestra el contenido de la EEPROM.

*Tabla 3-3  
Sensor BMP085: contenido de la EEPROM*

<b>BMP085 Dir. de registro</b>		
<b>Parámetro</b>	<b>MSB</b>	<b>LSB</b>
AC1	0xAA	0xAB
AC2	0xAC	0xAD
AC3	0xAE	0xAF
AC4	0xB0	0xB1
AC5	0xB2	0xB3

AC6	0xB4	0xB5
B1	0xB6	0xB7
B2	0xB8	0xB9
MB	0xBA	0xBB
MC	0xBC	0xBD
MD	0xBD	0xBF

Fuente: adaptada do (Bosch Sensortec, 2008, pág. 11)

Cada unidad del sensor tiene coeficientes individuales de calibración. Antes de calcular el valor de la temperatura y la presión, la unidad maestra debe leer dichos coeficientes. Para ello, debe enviar el valor del registro del bit MSB para cada uno de los coeficientes y seguidamente, leer la respuesta del sensor y almacenar el valor devuelto en su variable o parámetro correspondiente.

### 3.3.6 Modos de operación.

La tasa de muestreo o *sampling rate* en inglés, se puede configurar mediante programación a través del código de la unidad maestra en cuatro modos diferentes, los cuáles se muestran en la Tabla 3-4. Nótese que a mayor número de muestras por segundo mayor es el tiempo de conversión del ADC y también mayor será el consumo.

Tabla 3-4  
Sensor BMP085: modos de operación

Mode	Oversampling setting	Muestras	Tiempo de conversión (ms)
Ultra low power	0	1	4.5
Standard	1	2	7.5
High resolution	2	4	13.5
Ultra high resolution	3	8	25.5

Fuente: basado en (Bosch Sensortec, 2008, pág. 10)

Nota 1: el tiempo de conversión indicado es para la presión.

Nota 2: el tiempo de conversión para la temperatura siempre será de 4.5 ms.

Nota 3: Oversampling setting (OSS) es la variable para configurar el modo de operación.

Tabla 3-5

*Sensor BMP085: escritura*

Medida	Registro	Comando
Temperatura		0x2E
Presión (OSS = 0)	0xF4	0x34
Presión (OSS = 1)		$0x34 + (OSS \ll 6) = 0x74$
Presión (OSS = 2)		$0x34 + (OSS \ll 6) = 0xB4$
Presión (OSS = 3)		$0x34 + (OSS \ll 6) = 0xF4$

Fuente: (Bosch Sensortec, 2008)

Nota: OSS es oversampling setting

Tabla 3-6

*Sensor BMP085: lectura*

Parámetro	Tamaño	Registros involucrados
UP (Uncompensated Pressure)	16 a 19 bits	MSB = 0xF6, LSB = 0xF7,
UT (Uncompensated Temperature)	16 bits	XLSB = 0xF8 (opcional)
Datos EEPROM	16 bits	0xAA a 0xBF

Fuente: (Bosch Sensortec, 2008)

### 3.3.7 Obtención de la presión, la temperatura y la altitud.

Luego de la condición de *Start*, el maestro debe enviar el valor del registro donde va a escribir, seguidamente envía el comando de la variable a convertir (consultar Tabla 3-5). Este proceso se ilustra en la Figura 3-8. La conversión de la temperatura se tomará 4.5 ms, el tiempo de conversión para la presión depende del modo de operación elegido.

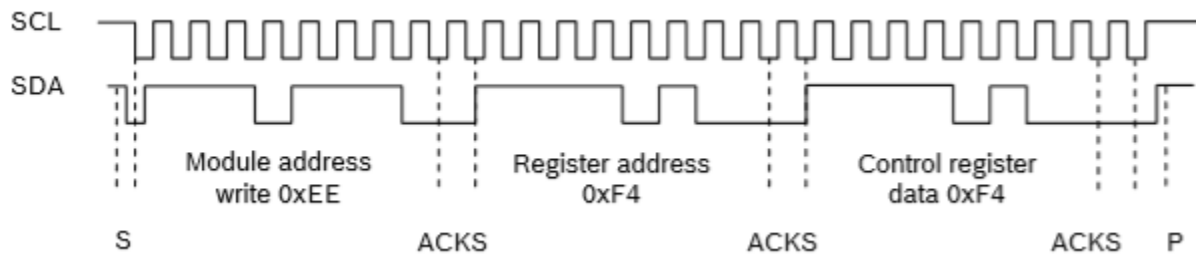


Figura 3-8. Sensor BMP085, ejemplo de inicio de la medición de la presión en el modo standard

Fuente: adaptado de (Bosch Sensortec, 2008, pág. 16)

Inmediatamente el sensor iniciará la conversión ( $EOF = 0$ ) de la variable solicitada. Cuando finalice cambiará el estado del terminal EOF ( $EOF = 1$ ) para dar aviso al microcontrolador maestro que ha finalizado. Entonces éste podrá leer el valor de la variable medida, para ello debe enviar el registro de donde se comenzará a leer (consultar Tabla 3-6) y luego realizar la operación de lectura (Figura 3-9).

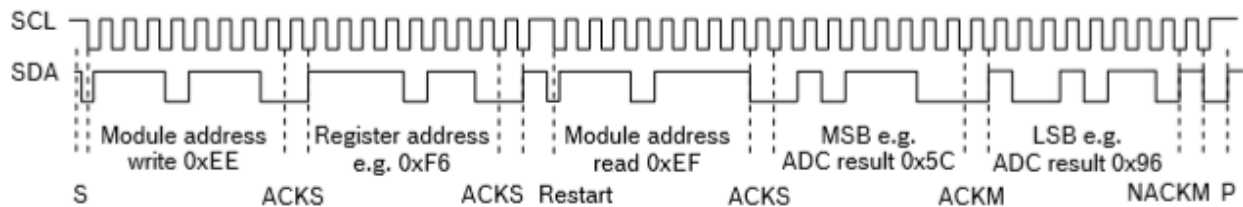


Figura 3-9. Sensor BMP085, ejemplo de lectura desde el bit MSB  
Fuente: (Bosch Sensortec, 2008, pág. 17)

El valor no compensado de la temperatura  $UT$  será  $MSB \ll 8 + LSB$ . El valor no compensado de la presión  $UP$  será  $(MSB \ll 16 + LSB \ll 8 + XLSB) \gg (8-oss)$ .

Para obtener el valor compensado de la temperatura y la presión de deben realizar las operaciones ilustradas en la Figura 3-10 y Figura 3-11 respectivamente.

```

136 long X1, X2, X3, B3, B5, B6, B7, T, P;
137 unsigned long B4;
138
139 /*Calibrando temperatura. Ver hoja de datos.*/
140 X1 = (UT - AC6) * AC5 >> 15;
141 X2 = (MC << 11) / (X1 + MD);
142 B5 = X1 + X2;
143 *Temperature = (B5 + 8) >> 4;

```

Figura 3-10. Sensor BMP085, compensar temperatura (código en lenguaje C)

```

145  /*Calibrando presión. Ver hoja de datos.*/
146  B6 = B5 - 4000;
147  X1 = (B2 * (B6 * B6 >> 12)) >> 11;
148  X2 = AC2 * B6 >> 11;
149  X3 = X1 + X2;
150  //.....
151  T = AC1;
152  T = (T * 4 + X3) << OSS;
153  B3 = (T + 2) / 4;
154  //.....
155  X1 = AC3 * B6 >> 13;
156  X2 = (B1 * (B6 * B6 >> 12)) >> 16;
157  X3 = ((X1 + X2) + 2) >> 2;
158  B4 = (AC4 * (unsigned long) (X3 + 32768)) >> 15;
159  B7 = ((unsigned long) UP - B3) * (50000 >> OSS);
160  if (B7 < (long)0x80000000) {P = (B7 * 2) / B4;}
161  else {P = (B7 / B4) * 2;}
162  X1 = (P >> 8) * (P >> 8);
163  X1 = (X1 * 3038) >> 16;
164  X2 = (-7357 * P) >> 16;
165  *Pressure = P + ((X1 + X2 + 3791) >> 4);

```

Figura 3-11. Sensor BMP085, compensar presión (código en Lenguaje C)

Para calcular la altitud en metros a partir del valor obtenido de la presión atmosférica se usa la siguiente expresión matemática de uso general descrita en Ec. 1; donde  $p$  es la presión en hPa obtenida y  $p_0 = 1013.25 \text{ hPa}$  (presión atmosférica a nivel del mar).

$$\text{Altitud} = 4330 * \left( 1 - \left( \frac{p}{p_0} \right)^{\frac{1}{5.255}} \right) \quad [\text{Ec. 1}]$$

### 3.3.8 Diagrama de flujo.

La Figura 3-12 ilustra el diagrama de flujo que se utilizó para realizar el proceso de comunicación entre la tarjeta CY8CKIT-042 y el sensor BMP085.

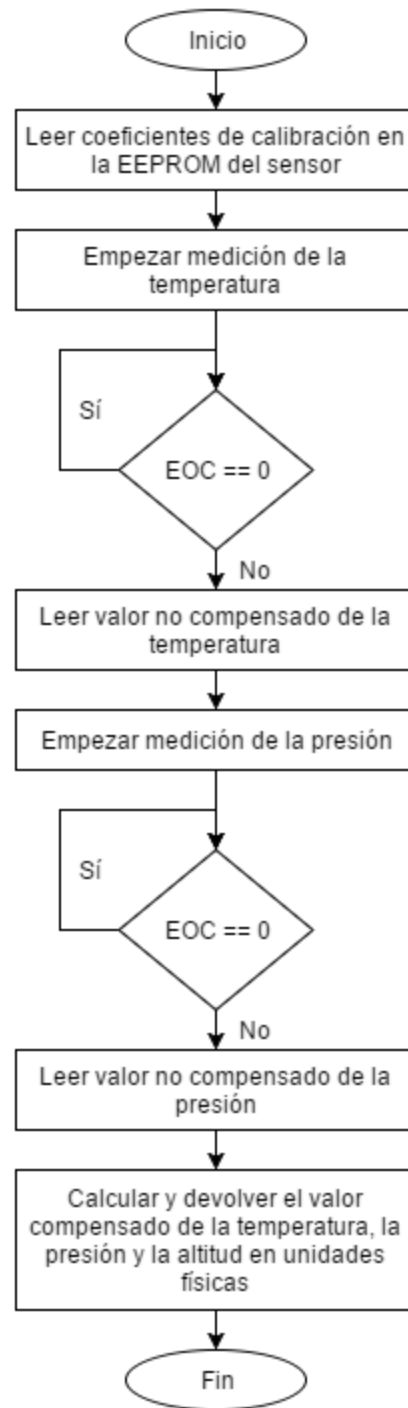


Figura 3-12. Diagrama de flujo seguido para comunicarse con el sensor BMP085

### 3.3.9 Conectar BMP085 a CY8CKIT-042.

La Figura 3-19 ilustra el diagrama esquemático para la conexión del sensor BMP085 a la tarjeta CY8CKIT-042. Nótese que no aparecen las resistencias  $R_p$  que establece la norma para el protocolo I2C, esto es porque estas resistencias ya están habilitadas internamente para P4\_0 y P4\_1 en la Pioneer.

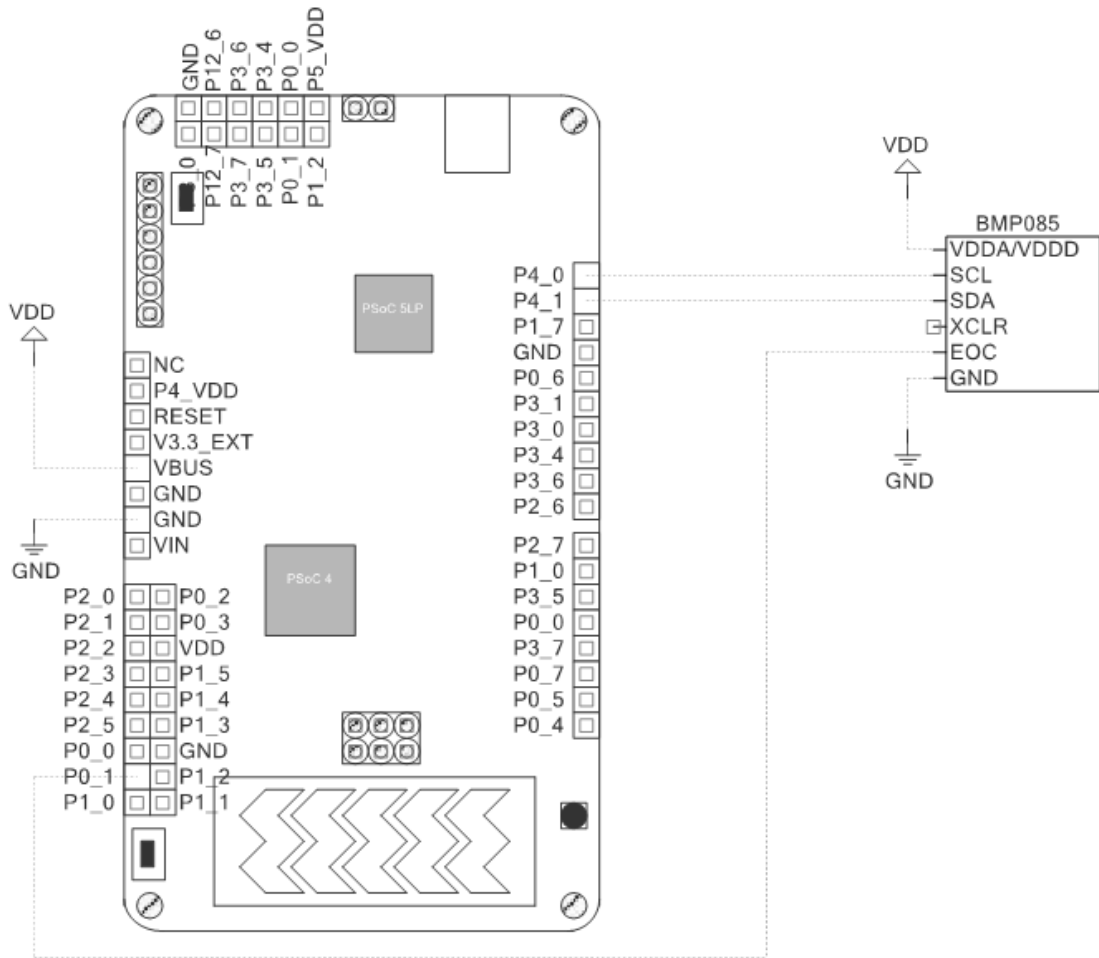


Figura 3-13. Sensor BMP085, conexión con la tarjeta CY8CKIT-042



### 3.4 Medición de la humedad relativa

Para la medición de la humedad relativa se utilizó el sensor inteligente DHT11 como el que se muestra en la Figura 3-14. El DHT11 es un sensor diseñado para la medición de la humedad relativa y de la temperatura ambiente del lugar en donde opere. Es fabricado por *D-Robotics UK* ([www.droboticonline.com](http://www.droboticonline.com)). Consta de un componente para la medición de la humedad y un componente para la medición de la temperatura, además de un microcontrolador de 8-Bits. Está protegido contra interferencias y es de respuesta rápida. Almacena coeficientes de calibración para compensar el valor obtenido de las variables que soporta.



*Figura 3-14. Sensor DHT11, aspecto*

#### 3.4.1 Especificaciones técnicas y características.

Entre las principales características para el DHT11 están las siguientes:

- Alimentación de 3V a 5.5V
- Rango de medida para la humedad: 20-90%
- Error relativo en la medida de la humedad:  $\pm 5\%$
- Rango de medida para la temperatura: 0-50 °C

- Error relativo en la medida de la temperatura:  $\pm 2\%$
- Resolución: 1
- Bajo consumo.
- Encapsulado de 4 pines

### 3.4.2 Asignación de pines y circuito típico de conexión.

La Figura 3-15 ilustra el circuito típico de conexión entre el sensor DHT11 y el microcontrolador. El fabricante recomienda el uso de una resistencia de *Pull-Up* de valor  $5K\Omega$  en el pin de datos y un condensador de  $100nF$  de para filtrar la alimentación. La Tabla 3-7 muestra la asignación de pines para este sensor.

Tabla 3-7  
Sensor DHT11: distribución de pines

Pin. No.	Nombre	Función
1	VCC	Power
2	Data	Digital bidireccional
3	Null	
4	GND	Power

Nota: el pin número 3 no se utiliza.

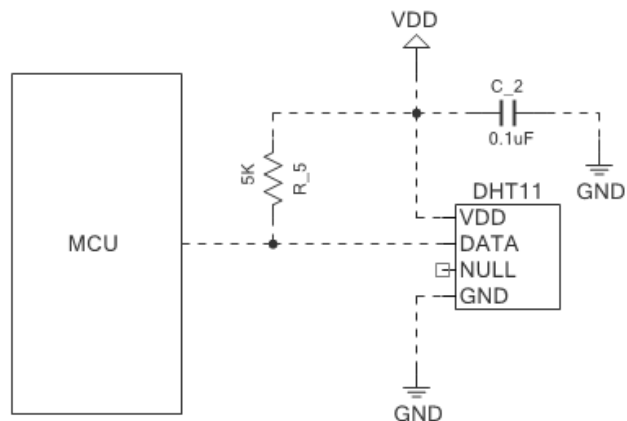


Figura 3-15. Sensor DHT11, circuito típico de conexión

### 3.4.3 Obtención de la humedad relativa y de la temperatura.

El DHT11 no utiliza un protocolo de comunicación estándar para la transmisión de datos con el microcontrolador, éste implementa un protocolo propio, en el cual la comunicación se realiza a través de una sola línea bidireccional (Pin 2). La Figura 3-16 ilustra este proceso.

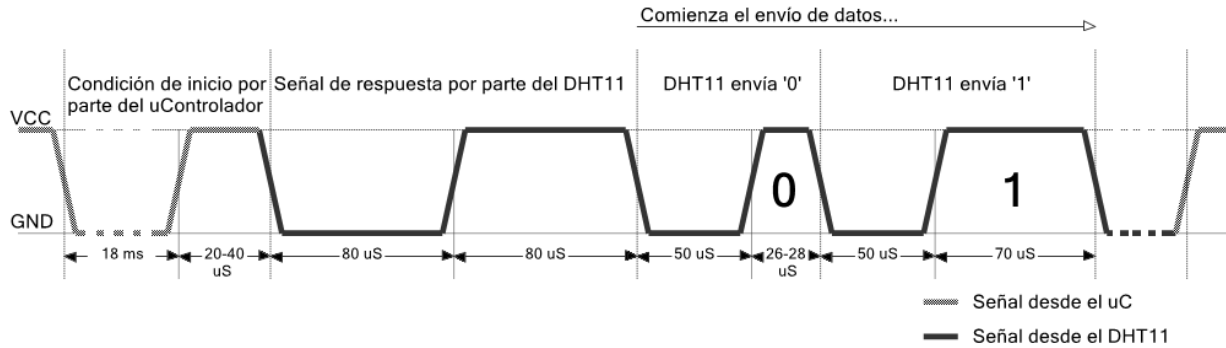


Figura 3-16. Sensor DHT11, protocolo de transmisión

Para solicitar el envío de datos el microcontrolador debe poner la línea de transmisión a cero lógico durante 18 ms y luego ponerla a uno lógico para esperar la respuesta del sensor, éste se tardará entre 20  $\mu$ S y 40  $\mu$ S en responder. La señal de respuesta consiste en un pulso en bajo y un pulso en alto durante 80  $\mu$ S, seguido de la señal de respuesta empezará el envío de datos. Independientemente del estado lógico, antes de enviar un bit el sensor DHT11 pondrá a nivel bajo la línea de transmisión durante 50  $\mu$ S, inmediatamente después pondrá la línea en alto durante 26  $\mu$ S a 28  $\mu$ S si es un cero o durante 70  $\mu$ S si es un uno. El envío de datos terminará luego de haber enviado 40 bits. Al final de la transmisión el microcontrolador debe poner la línea de comunicación a nivel alto para la próxima transmisión.

Los 40 bits que se reciben del sensor DHT11 están divididos en 5 bytes, los cuales representan:

- El primer byte es la parte entera de la humedad relativa.
- El segundo byte es la parte decimal de la humedad relativa. En este sensor no se utiliza y siempre devuelve cero.

- El tercer byte es la parte entera de la temperatura.
- El cuarto byte es la parte decimal de la temperatura. Tampoco se utiliza en este sensor y siempre devuelve cero.
- El último byte es el *checksum* o suma de verificación, y es la suma de los anteriores bytes transmitidos.

#### **3.4.4 Diagrama de flujo.**

La Figura 3-17 ilustra el algoritmo en el que se basa la función programada para la obtención de los datos del sensor. En la CY8CKIT-042 se configuró un pin como bidireccional y se le asignó una interrupción por flanco de subida. Se determina el estado del bit recibido testeando el estado de la línea en función de los tiempos conocidos de estos, esto se muestra en la Figura 3-18.

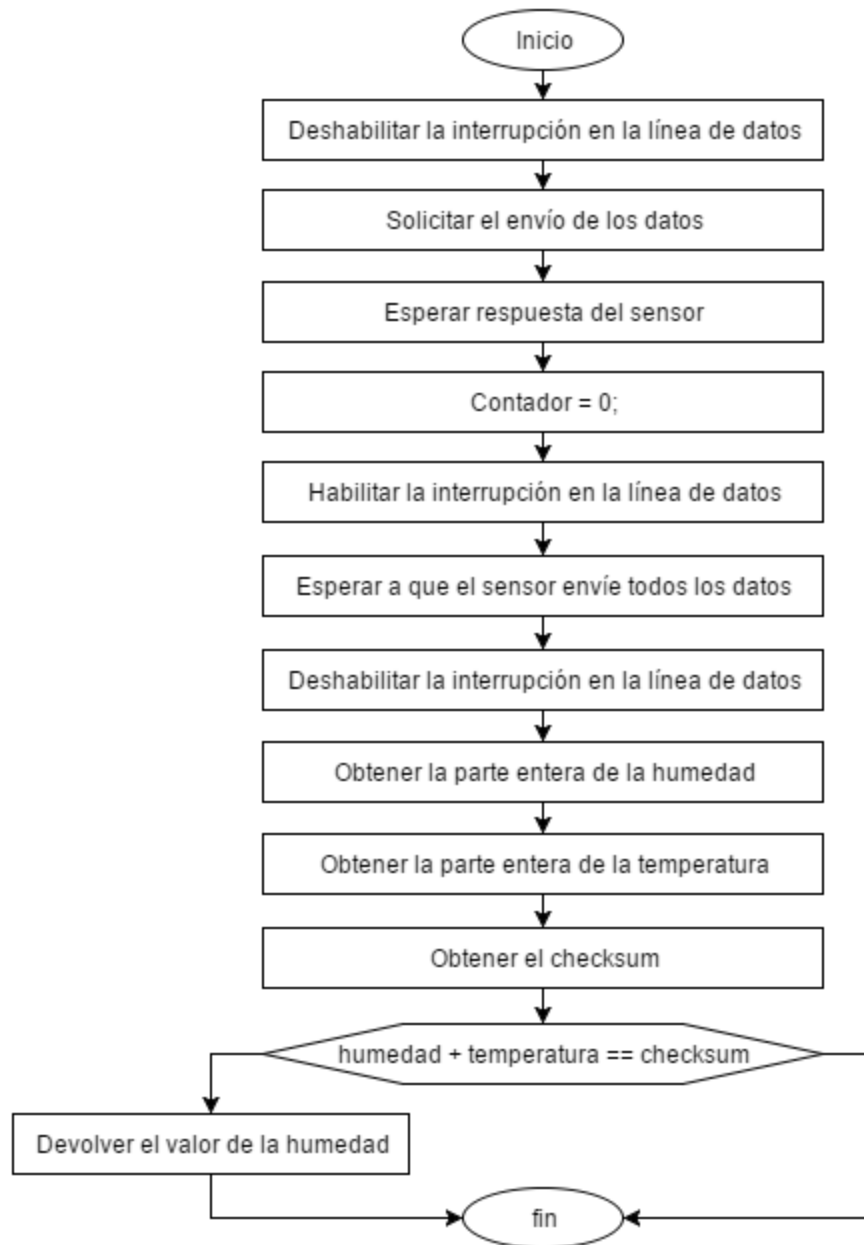


Figura 3-17. Sensor DHT11, diagrama de flujo seguido para recuperar los datos del sensor.

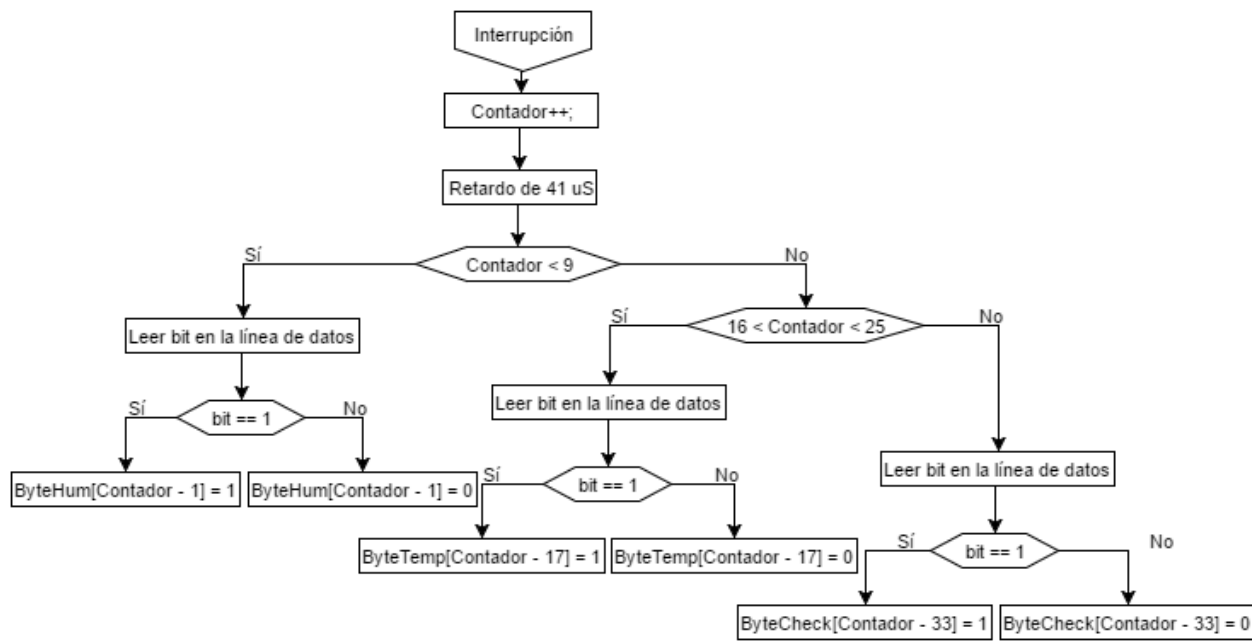


Figura 3-18. Sensor DHT11, diagrama de flujo de la interrupción

### Conectar DHT11 a CY8CKIT-042.

La Figura 3-19 ilustra el diagrama esquemático usado en el prototipo de estación meteorológica para conectar el sensor DHT11 a la CY8CKIT-042.

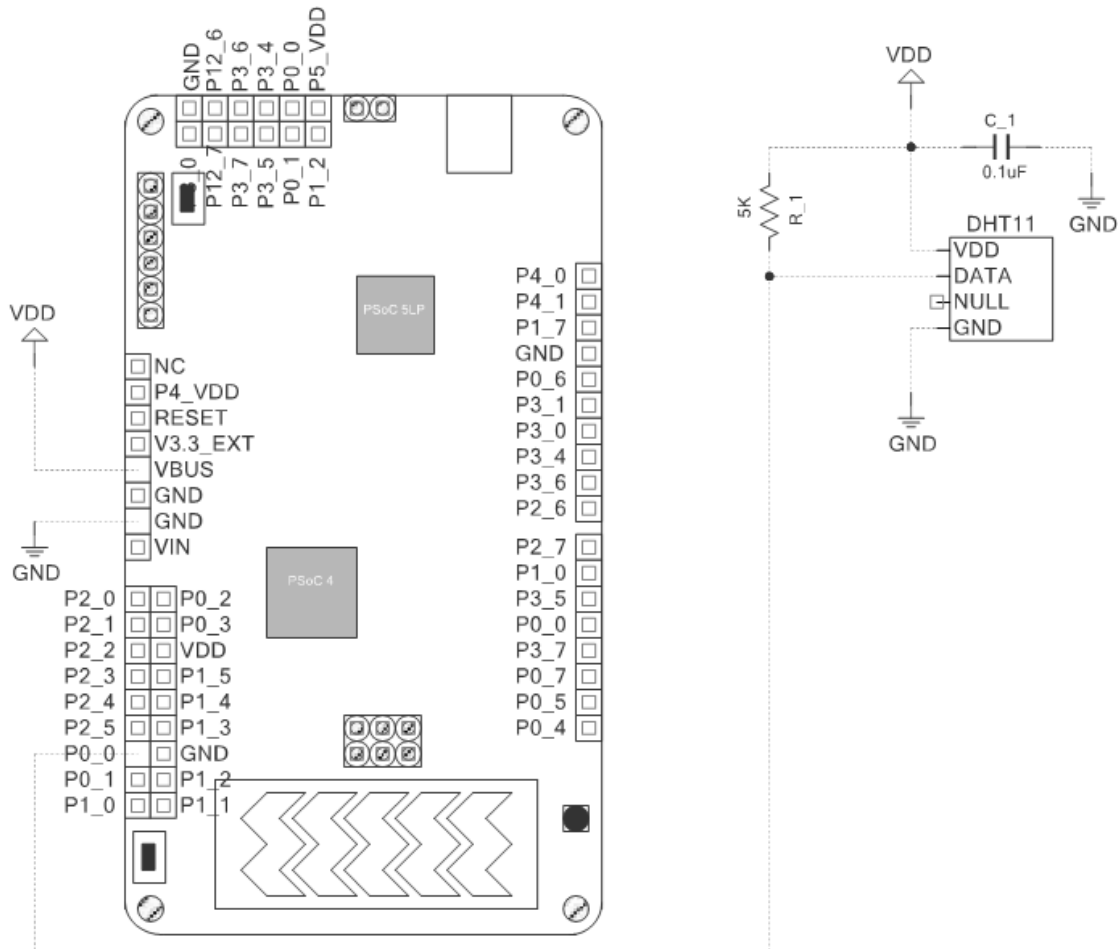


Figura 3-19. Sensor DHT11, conexión con la tarjeta CY8CKIT-042

### 3.5 Medición de la velocidad del viento

Por cuestiones de costo y accesibilidad para la medición de la velocidad del viento no se pudo adquirir del mercado ningún sensor diseñado para tal fin. Así que se optó por el diseño y realización de uno propio.

#### 3.5.1 Descripción.

El anemómetro diseñado se ilustra en la Figura 3-20, consta de tres cazoletas adheridas a uno de los extremos de un eje central, perpendiculares a éste y separadas 120° entre sí.



Figura 3-20. Anemómetro, aspecto

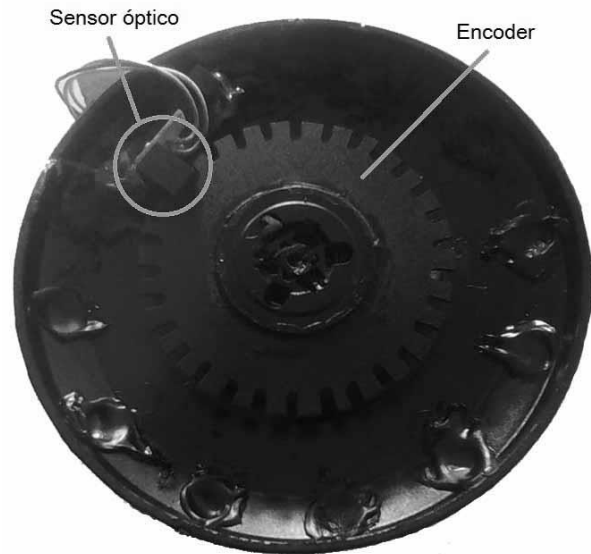


Figura 3-21. Anemómetro, descripción

En el extremo opuesto, se sitúa, paralelamente a las cazoletas, un *encoder* que gira simultáneamente con éstas al ser impulsadas por el viento. Los dientes del *encoder* encajan perfectamente en la ranura de un sensor óptico de barrera, de manera que el haz infrarrojo queda perpendicularmente a éstos. Como se muestra en la Figura 3-21.

### 3.5.2 Principio de funcionamiento.

La forma semiesférica de las cazoletas, su distribución y la reducida resistencia mecánica del eje facilitan el impulso de éstas por acción del viento. El *encoder* gira simultáneamente con las cazoletas y sus dientes interrumpen el haz infrarrojo del sensor óptico. Acción que es capturada por una interrupción en la tarjeta CY8CKIT-042. Considerando que durante el tiempo de muestreo del anemómetro (en el orden de los ms) la velocidad del viento será constante debido a que éste no presente cambios drásticos en periodos de tiempo tan cortos, se puede analizar el movimiento



del disco como circular uniforme. Por consiguiente, la velocidad lineal o tangencial del disco, que corresponde a la velocidad del viento que lo impulsa, viene dada por Ec. 2.

$$V_L = \frac{Arco}{T} \text{ [Ec. 2]}$$

Donde  $T$  es el periodo de muestreo y el arco corresponde a Ec. 3. De esta forma se calcula la velocidad del viento.

$$Arco = \left( \frac{2\pi r}{No. \text{ dientes}} \right) (No. \text{ interrupciones}) \text{ [Ec. 3]}$$

### 3.5.3 Conexión del Anemómetro con la tarjeta CY8CKIT-042.

La Figura 3-22 ilustra la conexión del anemómetro con la tarjeta CY8CKIT-042. Además del sensor óptico de barrera también consta de un transistor bipolar como acondicionador de señal. Sólo es necesario una línea de comunicación con la tarjeta. El pin que captura la interrupción está configurado en el modo *Resistive Pull Up*.

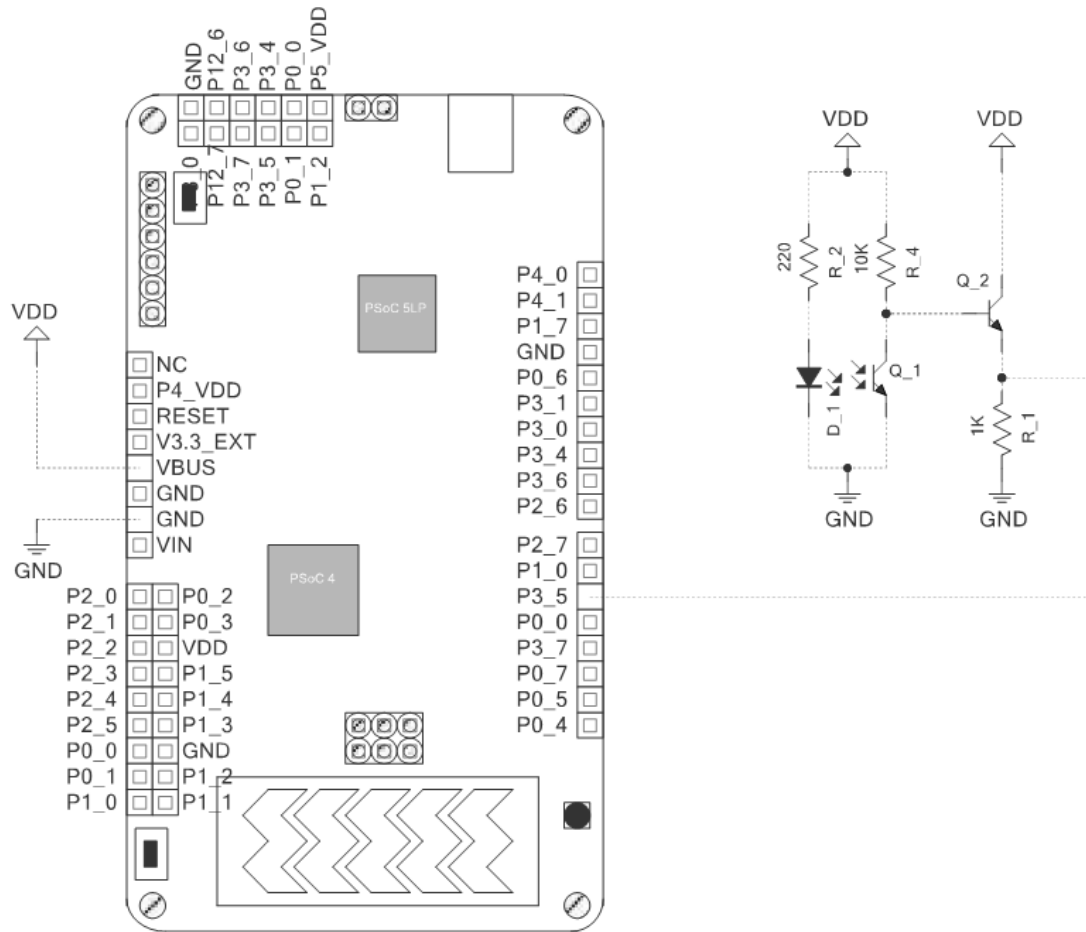


Figura 3-22. Anemómetro, conexión con la tarjeta CY8CKIT-042

### 3.5.4 Diagrama de flujo.

La Figura 3-23 ilustra el diagrama de flujo en el que se basa la función para recuperar los datos del sensor.

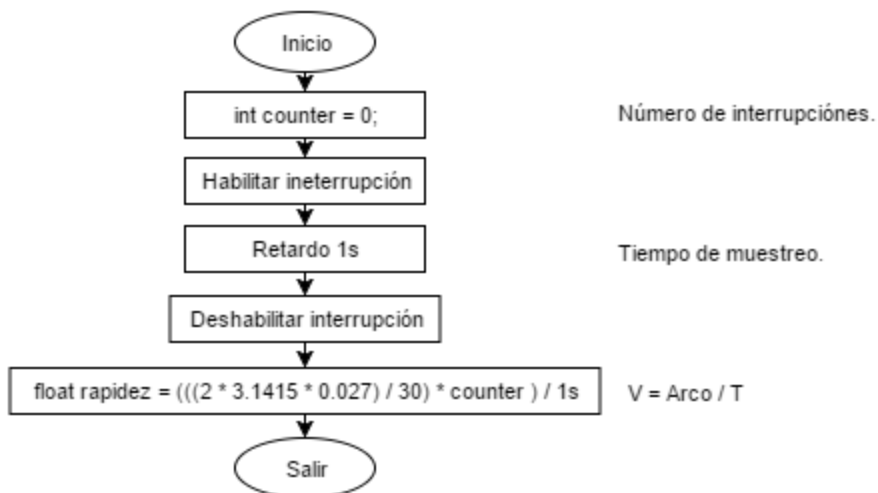


Figura 3-23. Anemómetro, diagrama de flujo

### 3.6 Medición de la dirección del viento

Al igual que con la velocidad del viento, para obtener la dirección de éste se hizo necesario la construcción de un sensor propio. El cual se describe a continuación.

#### 3.6.1 Descripción.

Consta de una veleta fija a un eje que gira libremente y un contrapeso en forma de flecha para equilibrar el peso de ésta (véase Figura 3-25). En la base, se encuentra situado un imán que se mueve con el eje en la misma dirección de la velera e inmediatamente a éste, una placa con ocho *Reed Switches* ubicados de acuerdo a las direcciones Norte, Noroeste, Oeste, Suroeste, Sur, Sureste, Este y Noreste. Como se muestra en la Figura 3-24.



Figura 3-25. Veleta, aspecto



Figura 3-24. Veleta, detalles

### 3.6.2 Principio de funcionamiento.

La reducida resistencia que ofrece el eje al movimiento y la forma que tiene la veleta, posibilitan que ésta sea desplazada por el viento hasta quedar paralelamente a él o en su misma dirección, al igual que el imán permanente situado en la base que sostiene al conjunto, el cual activará a los *Reed Switches* (interruptores magnéticos) correspondientes a su dirección actual. El conjunto de *Reed Switches* está conectado a un puerto de la tarjeta CY8CKIT-042 que leerá una combinación diferente para cada una de las direcciones válidas. El sensor tiene una resolución de  $22.5^\circ$ , o sea que puede leer un total de 16 direcciones.

### 3.6.3 Conexión con la tarjeta CY8CKIT-042.

La Figura 3-26 ilustra el circuito de conexión con la tarjeta CY8CKIT-042. Cada Reed Switch representa una dirección, si el imán se sitúa entre dos interruptores magnéticos, la dirección correspondiente será la media de las direcciones que estos representan. El puerto en la tarjeta

CY8CKIT-042 al que están conectados está configurado en el modo *Resistive Pull Up*, de modo que a los interruptores sólo es necesario conectarlos al positivo de la fuente.

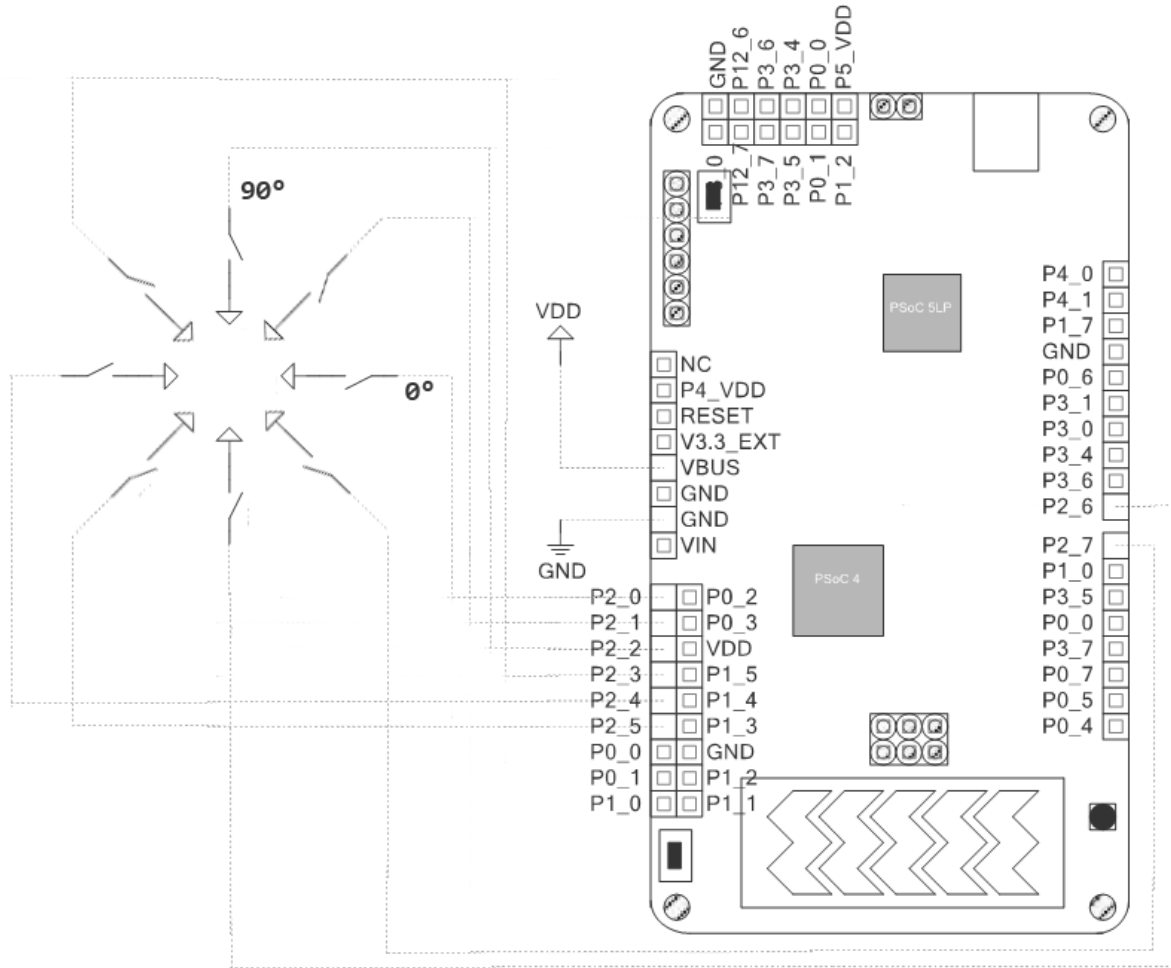


Figura 3-26. Veleta, conexión con la tarjeta CY8CKIT-042

### 3.6.4 Lectura del sensor.

En el código sólo es necesario leer el valor del puerto y compararlo con una tabla en la que se listan las direcciones correspondientes para cada valor.

### **3.7 Comunicación con el ordenador**

Para la transferencia de datos a nivel local hacia el ordenador se hace uso del puente USB-UART (Figura 3-42). La tarjeta sólo soporta un bloque UART en un mismo proyecto debido al limitado número de UDB's con los que cuenta el 44 TQFP; más adelante se verá que la comunicación con el módulo M95 se realiza a través de un bloque UART, debido a que sólo puede haber un bloque de éstos por proyecto, se hizo necesario de un medio físico para seleccionar entre el envío de datos a través del USB-UART o a través del M95, la solución a éste inconveniente fue el uso de jumpers a disposición del usuario para elegir entre ambas opciones.

### **3.8 Módulo de visualización**

Con la intención de hacer amigable la estación al usuario, se diseñó un módulo para la visualización de los procesos y dar conocimiento al operador del estado actual de la misma.

#### **3.8.1 Descripción.**

El módulo de visualización realizado se ilustra en la Figura 3-27. Consta, en esencia, de una GLCD de 128x64 pixeles (véase la Figura 3-29) que posibilita la impresión de un mínimo de información para visualizar a manera de registro los procesos ejecutados por la estación y el estado de la misma, a diferencia de una LCD común de 16x2.

Como la GLCD elegida ocupa 16 pines del microcontrolador que la gobierna, se descartó la posibilidad de conectarla directamente a la tarjeta CY8CKIT-042, debido a que el número de puertos de ésta es limitado y la GLCD no dejaría cabida para conectar el resto de subsistemas que conforman a la estación; o al menos no a todos. En cambio, se optó por utilizar un microcontrolador PIC16F877A-20, éste, se conecta a la tarjeta CY8CKIT-042 haciendo uso de una sola línea de

comunicación, a través de la cual recibe serialmente los datos a visualizar y otra línea para reset. Éste dispone de los recursos necesarios para operar la GLCD.

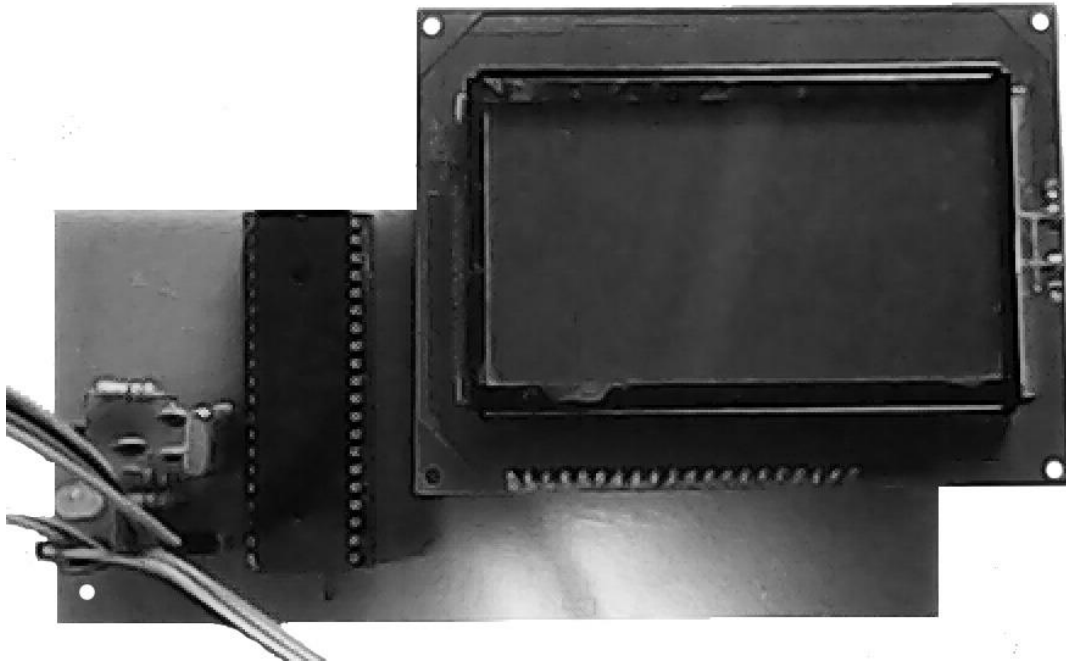


Figura 3-27. Módulo de visualización



Figura 3-29. GLCD 128x64



Figura 3-28. PIC16F877A-20, aspecto físico

El PIC16F877A-20 es un microcontrolador de gama media, bajo costo y reducido tamaño. Manufacturado por *Microchip Technology*. Integra una variedad de recursos que favorecen su

inclusión en diversas aplicaciones. Para mayor información consulte la hoja de datos facilitada por el fabricante en [www.microchip.com](http://www.microchip.com). La Figura 3-28 ilustra a este dispositivo.

### 3.8.2 Diagrama esquemático.

La Figura 3-30 ilustra el diagrama esquemático del módulo de visualización; este ocupa cuatro líneas para su funcionamiento: dos para la alimentación, una de reset y otra para la recepción de los datos a visualizar mediante comunicación serial asíncrona.

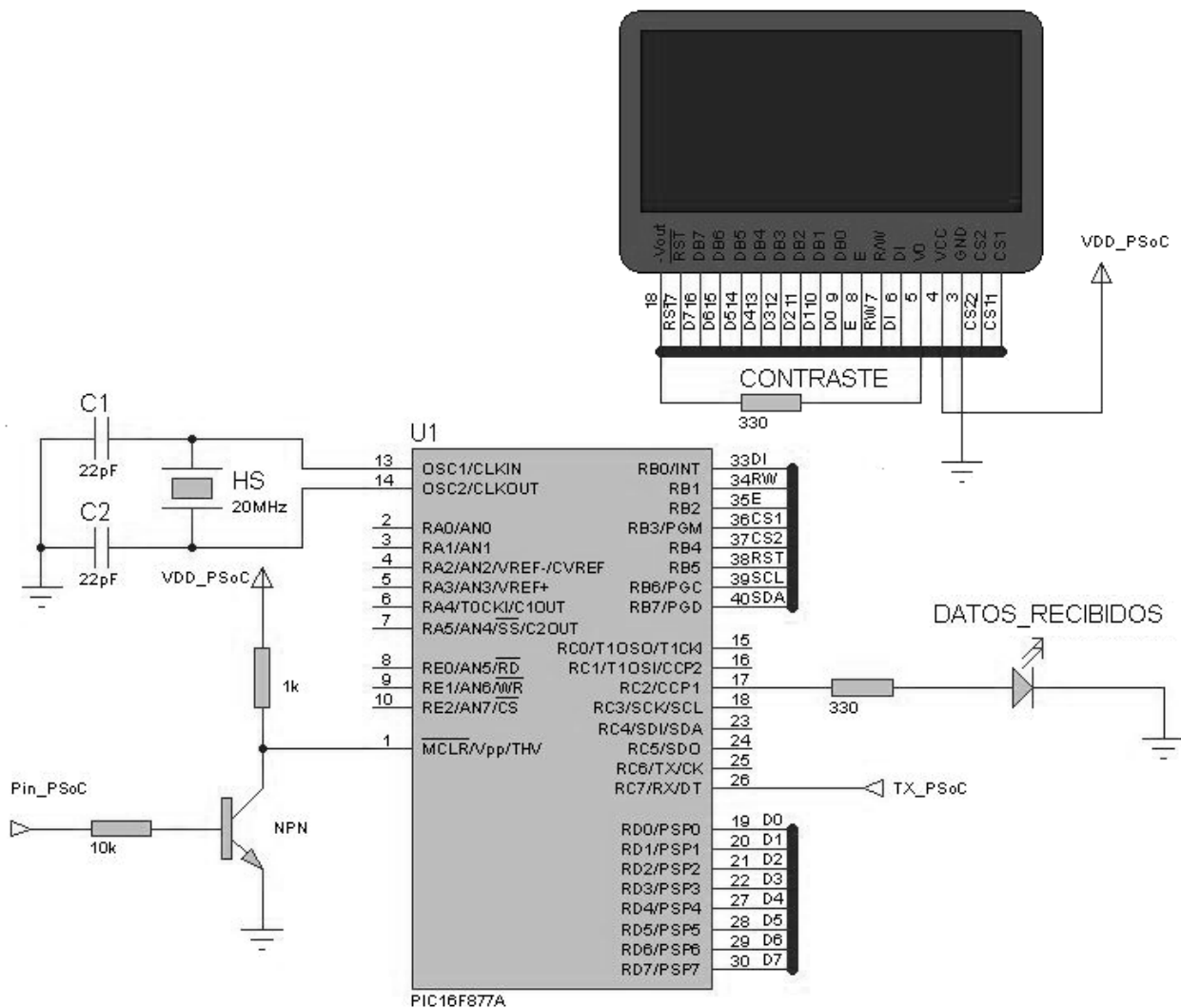


Figura 3-30. Módulo de visualización, diagrama esquemático



### 3.8.3 Funcionamiento.

El módulo de visualización ocupa una línea para la recepción de los mensajes a visualizar; la comunicación es serial asíncrona, por lo cual el PIC hace uso del bloque UART embebido. Para que los datos puedan ser visualizados la unidad maestra debe enviar un retorno de carro tras cada mensaje. Como sólo se puede hacer uso de un bloque UART por aplicación en la tarjeta CY8CKIT-042 y éste ya está siendo ocupado por el módulo de comunicación, entonces se emuló mediante código una línea de transmisión basada en la norma RS232 para este dispositivo. El módulo de visualización imprimirá en pantalla los mensajes que se sean enviados y actualizará la misma conforme estos vayan llegando. Los mensajes que se le envíen deben tener máximo 20 caracteres, incluyendo los no imprimibles.

### 3.8.4 Diagrama de flujo.

La Figura 3-31 ilustra el diagrama de flujo en el que se basó la programación de la función Main para el PIC16F77A-20. En esta se imprimen en la GLCD un menú de las opciones que ofrece la estación para exportar los datos. Luego habilita la interrupción por byte recibido y las interrupciones globales. Por último entra en un bucle infinito a la espera de mensajes a visualizar.

La programación para el PIC16F877A-20 se realizó en el IDE *PIC C Compiler*, una aplicación destinada a la elaboración de programas para microcontroladores PIC. Dispone de un conjunto de librerías y herramientas que facilitan la realización de los mismos. Desarrollado por *Custom Computer Services, Inc.* ([www.ccsinfo.com](http://www.ccsinfo.com)).

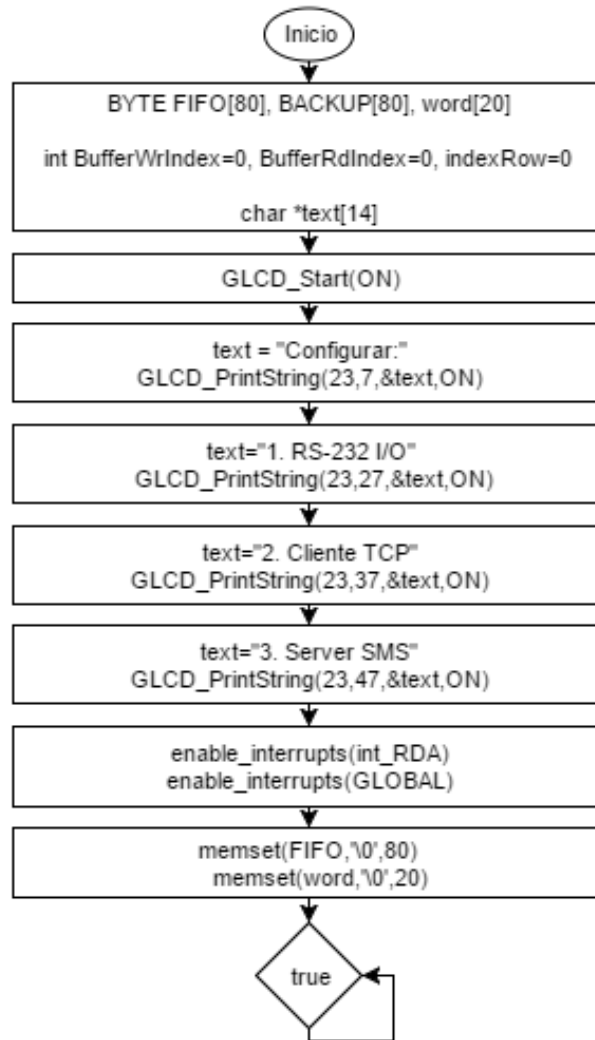


Figura 3-31. Módulo de visualización, diagrama de flujo de la función main

Cuando se recibe un byte se produce la interrupción por byte recibido y se ejecuta el código de la interrupción, el cual se basa en el diagrama de flujo de la Figura 3-32. Al llegar a la interrupción, se enciende el LED del módulo para dar aviso de que se están recibiendo datos. Luego obtiene el carácter correspondiente, si éste es nueva línea, entonces se imprime el mensaje; de lo contrario se almacena en un buffer. Los mensajes antiguos se desplazan hasta conforme llegan los nuevos.

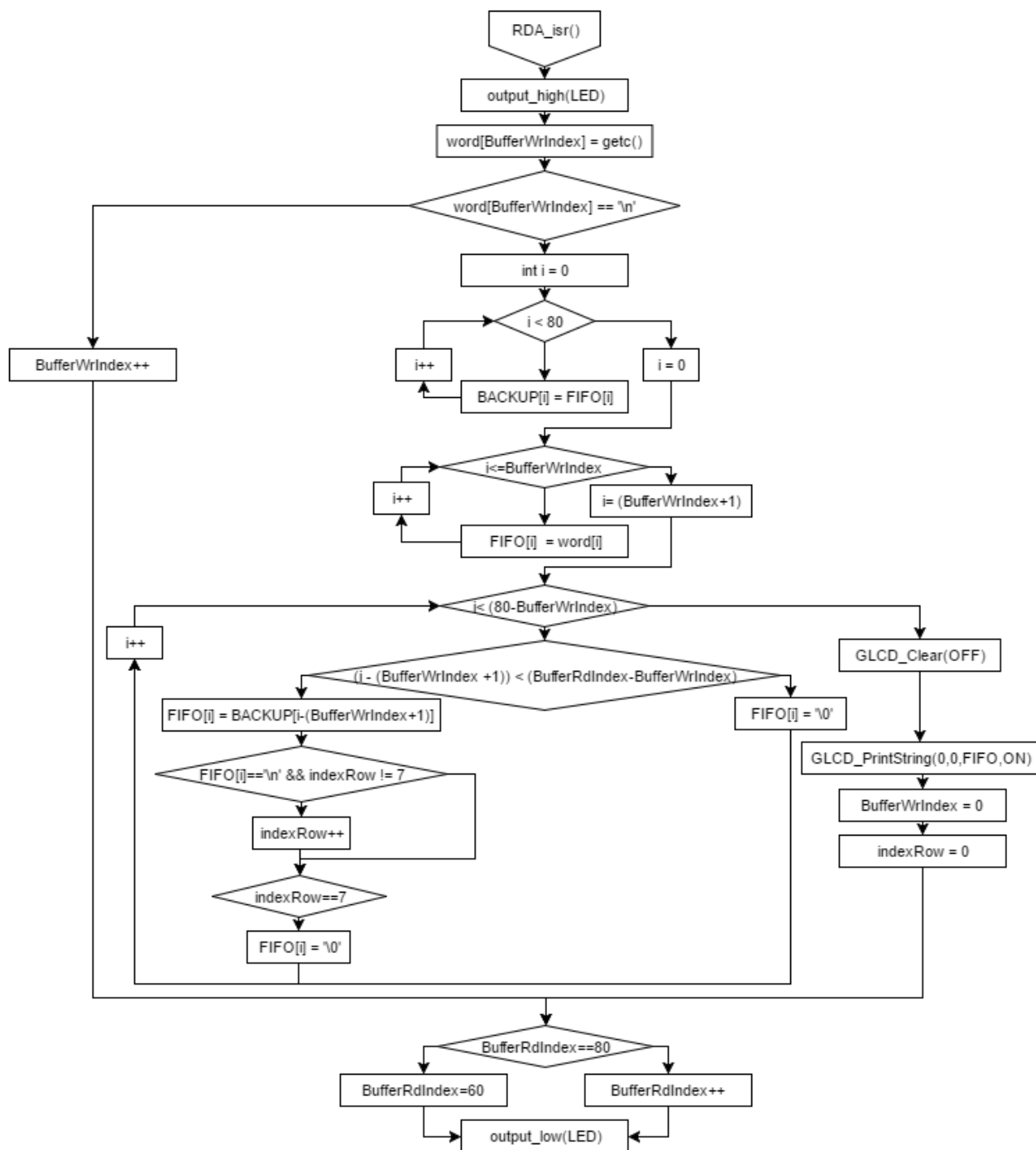


Figura 3-32. Módulo de visualización, diagrama de flujo de la interrupción

### 3.8.5 Conexión del módulo de visualización con la tarjeta CY8CKIT-042.

La Figura 3-33 ilustra el diagrama de conexión del módulo de visualización con la tarjeta CY8CKIT-042.

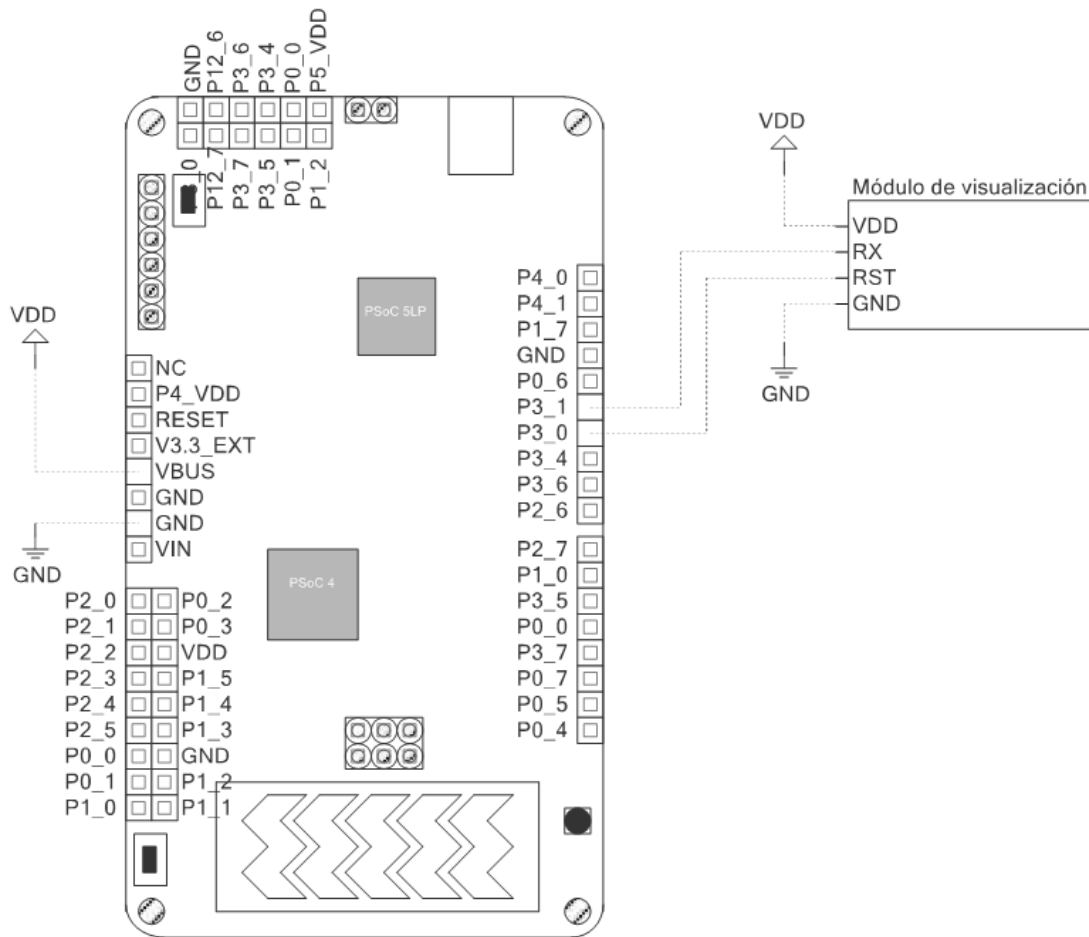


Figura 3-33. Módulo de visualización, conexión con la tarjeta CY8CKIT-042

### 3.9 Módulo de comunicación

Para proveer al prototipo la capacidad de exportar los datos a grandes distancias de donde se encuentra ubicado se hizo uso de una red estándar existente y ampliamente utilizada como lo es

la red de telefonía móvil e Internet. Así, se hizo necesario la utilización de un módem compatible con estas redes. Se trata del módem M95 fabricado por *Quectel* ([www.quectel.com](http://www.quectel.com)), que se ilustra en la Figura 3-34. Éste módem emplea tecnología GSM o 2G y además soporta GPRS. Basado en él se realizó el módulo de comunicación que se ilustra en la Figura 3-35.



Figura 3-34. M95, aspecto  
Fuente: (Quectel, 2014, pág. 72)

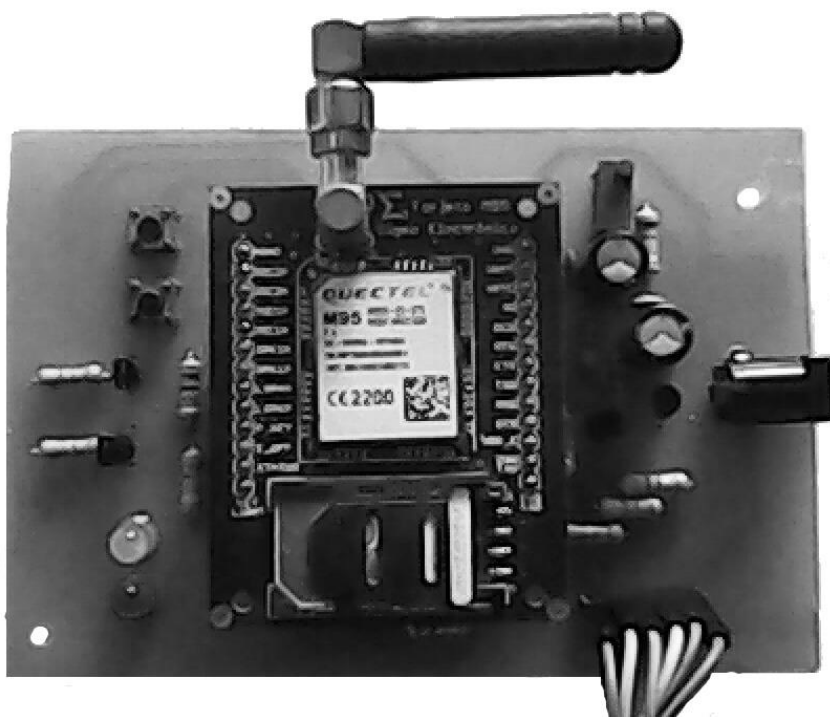


Figura 3-35. Módulo de comunicación

### 3.9.1 Descripción.

Consta, en esencia, del módem M95 y de los circuitos asociados que permiten operar las diferentes interfaces que intervienen en las funciones que éste realiza. Las cuales se describen a continuación:

#### 3.9.1.1 Alimentación.

El módulo puede ser alimentado con niveles de tensión desde 3.3V a 4.6V. El fabricante recomienda utilizar 4V; además advierte que el módulo eleva drásticamente el consumo de

corriente en 1.6A cada 4.615ms durante 577us mientras esté transmitiendo, lo que puede causar un descenso en el voltaje de alimentación. Si éste disminuye a un nivel por debajo del voltaje mínimo permitido el módulo se apagará automáticamente. La caída de tensión durante el periodo de transmisión no debería exceder los 400mV. La Figura 3-36 ilustra el circuito de alimentación propuesto por el fabricante y usado en el prototipo de estación meteorológica. La máxima corriente de carga aceptada por este circuito es de 3A y puede ser alimentado de 5V a 12V, para ajustar el voltaje a 4.1V en la salida se utiliza R\_24. Para la fuente de alimentación se recicló un adaptador de un computador portátil que satisface los requisitos.

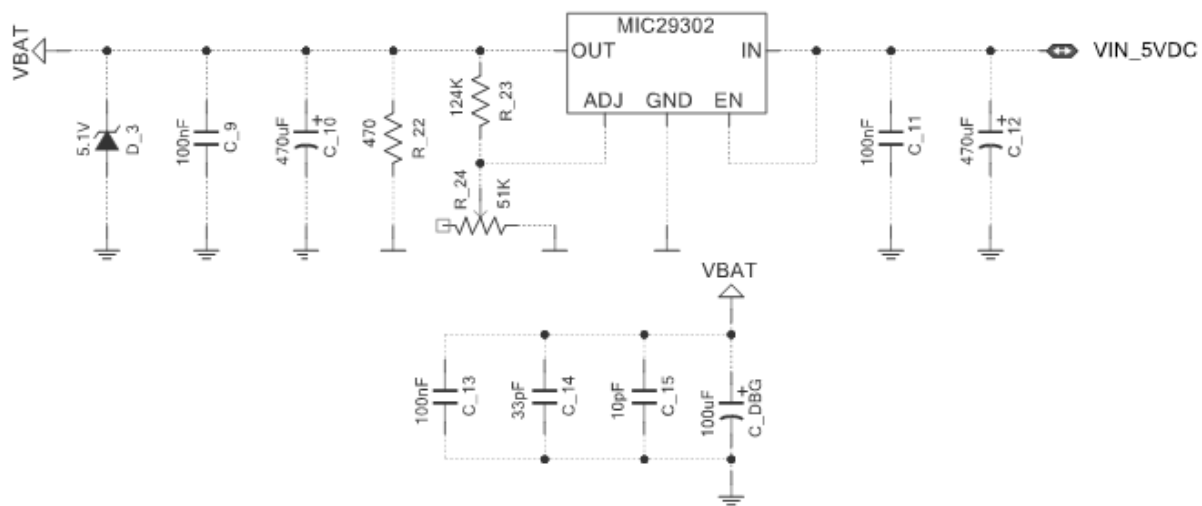


Figura 3-36. Módulo de comunicación, alimentación  
Fuente: basado en (Quectel, 2014, págs. 24-25)

### 3.9.1.2 Tarjeta SIM.

El módulo posee dos interfaces en las que se puede insertar indistintamente una SIM card (SIM1 y SIM2), pero sólo una puede estar habilitada a la vez. Ambas interfaces son alimentadas por un regulador interno del módulo, éste soporta SIM cards que operen a 1.8V y a 3.3V. Además, SIM1 tiene una función de detención que es opcional. En el prototipo sólo se hizo uso de SIM1, el

circuito asociado para ésta interfaz recomendado por el fabricante se ilustra en la Figura 3-37, nótese que no se hace uso de la función de detección.

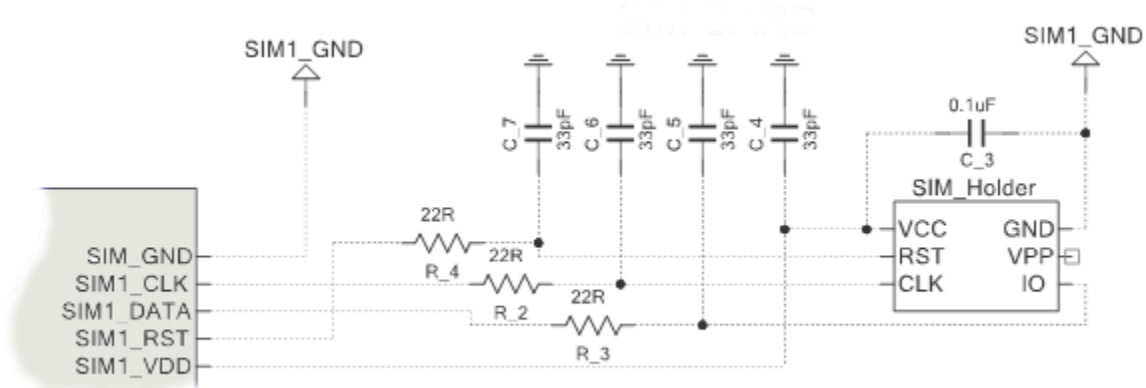


Figura 3-37. Módulo de visualización, tarjeta SIM  
Fuente: basado en (Quectel, 2014, págs. 54-55)

### 3.9.1.3 Antena.

En la hoja de datos el fabricante especifica que la impedancia de la antena que se debe conectar al módulo es de 50Ω. En la Tarjeta M95 se utilizó una antena GSM Quad Band omnidireccional. La conexión de la antena al módulo se ilustra en la Figura 3-38.

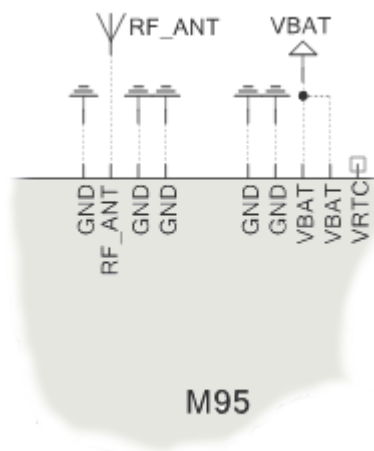


Figura 3-38. Módulo de comunicación, antena  
Fuente: basada en (Quectel, 2014, pág. 60)

### 3.9.1.4 Comunicación serial.

El módulo dispone de un bloque UART para la comunicación con el microcontrolador. Soporta velocidades desde 4800bps a 115200bps y la función Autobauding, que configura automáticamente la velocidad del módulo a la velocidad que opere el microcontrolador. Como la tarjeta CY8CKIT-042 se configuró para trabajar a 5V y el M95 opera a 4.1V, es necesario acoplar los niveles de tensión entre ambos dispositivos. El circuito recomendado por el fabricante se ilustra en la Figura 3-39.

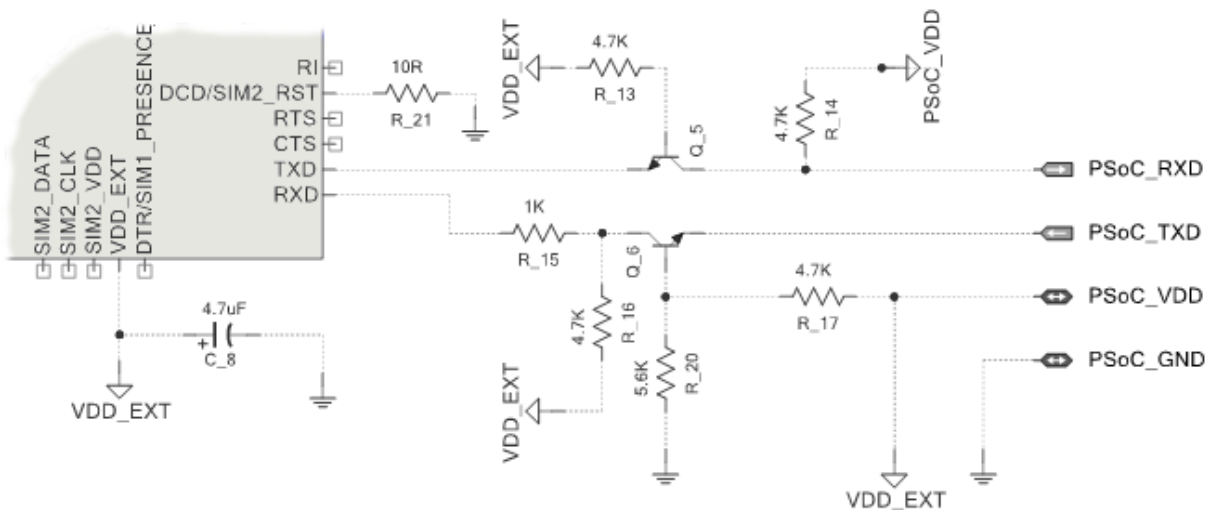


Figura 3-39. Módulo de comunicación, circuito de acople para las interfaces seriales  
Fuente: basado en (Quectel, 2014)

### 3.9.1.5 Encendido/Apagado.

El módulo se enciende poniendo la línea PWRKEY a nivel bajo durante al menos un segundo, esto se puede hacer mediante software (PSoC\_PWRKEY) o mediante hardware (SW\_2) como se ilustra en la Figura 3-40. Una vez alimentado el módulo, el fabricante recomienda esperar al menos 100 ms antes de encenderlo.



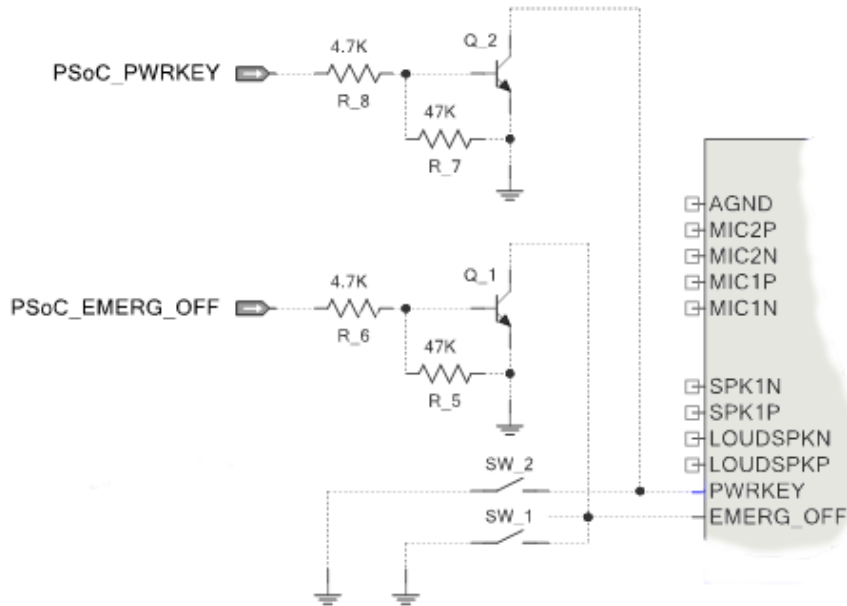


Figura 3-40. Módulo de comunicación, encendido/apagado  
 Fuente: basada en (Quectel, 2014, págs. 26,30)

El módulo puede se puede apagar de distintas formas:

- Si el voltaje de alimentación se sale del rango permitido el módulo enviará una alerta de baja tensión o sobretensión según sea el caso y acto seguido se apagará automáticamente.
- Normal proceso de apagado poniendo a nivel bajo la línea PWRKEY (accionando SW2 o mediante software a través de PSoc\_PWRKEY, Figura 3-40) durante un tiempo mayor a 0.7s y menor a 1s. El módulo tardará de 2s a 12s en salir.
- Normal proceso de apagado por código, enviando “AT+QPOWD=1” al módulo. Éste antes de apagarse responderá con un mensaje “NORMAL POWER DOWN”.

- Apagado de emergencia poniendo la línea EMERG\_OFF a nivel bajo durante 40ms (accionando SW1 o mediante software a través de PSoC\_EMERG\_OFF, Figura 3-40).

### 3.9.1.6 Netlight/Status.

La NETLIGHT es un pin del módulo que emite pulsos de acuerdo al estado de la red (para más información refiérase a la hoja de datos). Puede ser usada como indicador mediante un LED. El M95 también habilita un pin de STATUS para indicar si ha sido encendido. La Figura 3-41 ilustra el circuito de conexión asociado a estas interfaces.

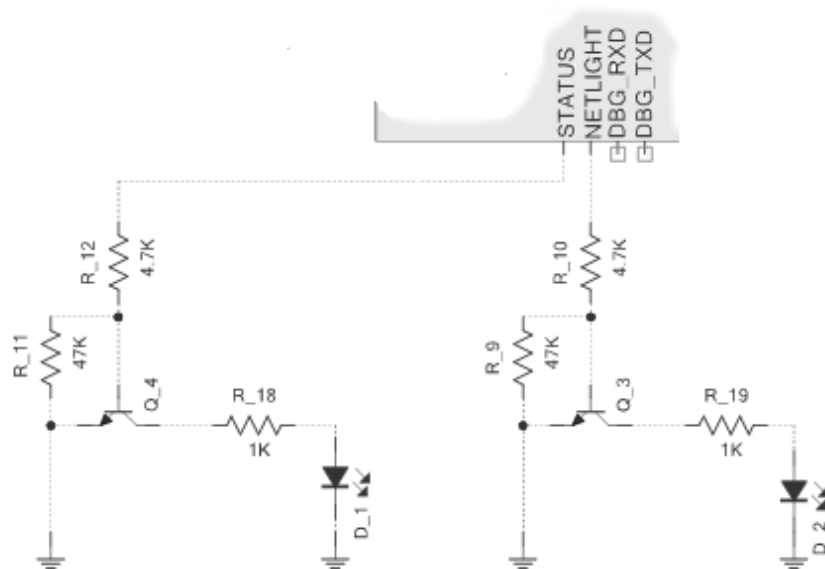


Figura 3-41. Módulo de comunicación, Netlight/Status  
 Fuente: basada en (Quectel, 2014, págs. 58,59)

### 3.9.2 Conexión del módulo de comunicación a la tarjeta CY8CKIT-042.

La Figura 3-42 ilustra el circuito de conexión del módulo de comunicación con la tarjeta CY8CKIT-042.

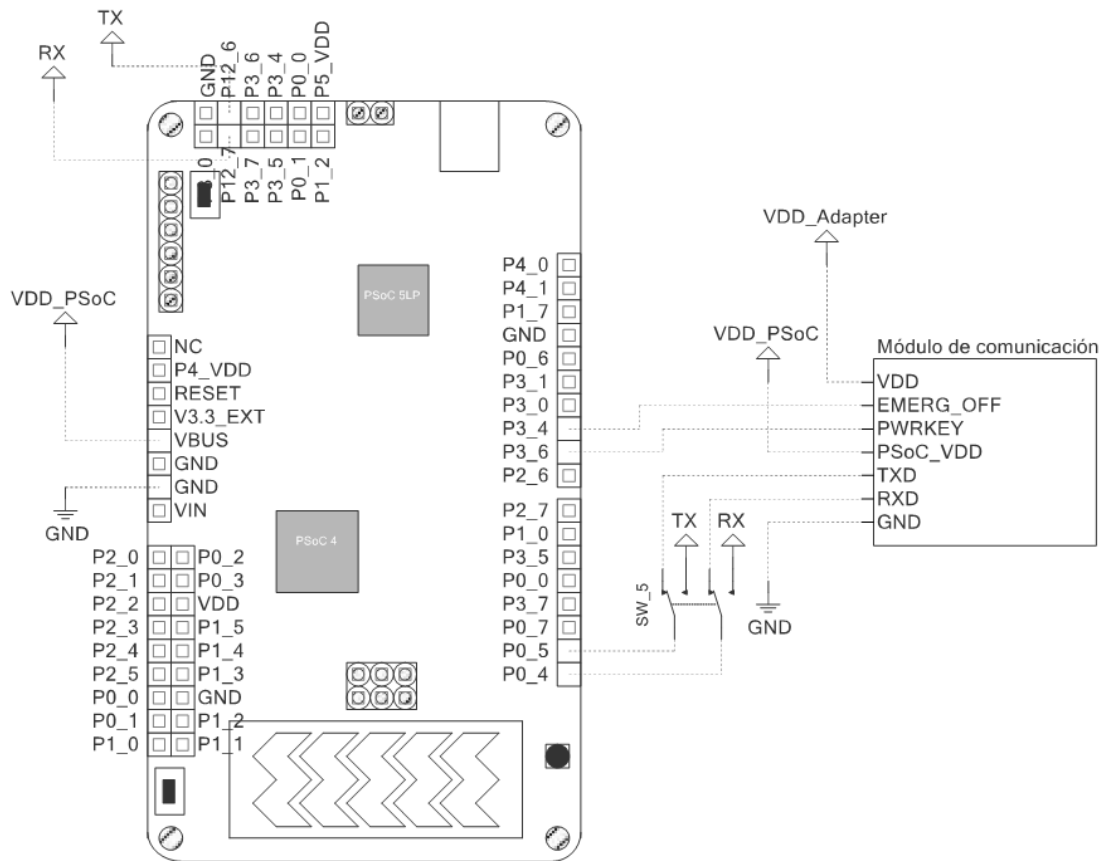


Figura 3-42. Módulo de comunicación, conexión con la tarjeta CY8CKIT-042

### 3.9.3 Funcionamiento.

El módulo obedece a un set de instrucciones propias de GSM conocidas como Comandos AT. Mediante las cuales se puede configurar y acceder a todos los servicios que éste ofrece. El fabricante facilita una lista en la que se incluyen todos los comandos pertinentes al módulo y su respectiva función. En esta sección sólo se hará mención de los comandos que intervienen en la operación de los servicios de los que se vale el prototipo para enviar los datos a través del módulo. Cada que éste recibe un comando por la interfaz serial, ejecuta la acción correspondiente a la instrucción recibida y devuelve la respuesta de la ejecución al dispositivo que lo gobierna. Para verificar esto se puede conectar el módulo al puerto serial del ordenador y utilizar un programa

para la comunicación del ordenador con otros dispositivos a través de este puerto. Si la computadora de la que se dispone no incorpora un puerto serial se puede hacer uso de un módulo conversor de RS232 a TTL y viceversa. La Tabla 3-8 muestra la secuencia de comandos básicos para inicializar el módem.

Tabla 3-8.  
Secuencia de inicio del módulo de comunicación

Secuencia de inicio		
Comando	Respuesta esperada	Descripción
AT<CR>	OK	Verifica sin el módulo responde.
ATE0<CR>	OK	Desactiva el eco.
AT+CPIN?<CR>	+CPIN: READY OK	Verifica si la SIM card está protegida por código PIN.

Nota: CR = retorno de carro.

### 3.9.3.1 Aplicación SMS.

La Tabla 3-9 describe la secuencia de comandos AT que intervienen en la aplicación SMS de la que se sirve la estación para enviar los datos de las variables medidas a un móvil.

Tabla 3-9.  
Comandos básicos para la aplicación SMS

Aplicación SMS		
Comando	Respuesta Esperada	Descripción
AT+CMGF=1<CR>	Ok	Configura tipo de SMS, si es 1 es texto y si es cero es PDU.
AT+CMGR=?<CR>	Ok	Habilita la recepción de mensajes.
AT+CNMI=2,2<CR>	Ok	Hace que una vez se reciba el mensaje éste se envíe a través de la UART-
AT+CMGS=<"Núm."><CR> <Msg.><\x1A>	+CMGS:<x> OK	Envía mensajes.

Nota: Núm. = destino, CR = retorno de carro, Msg. = mensaje, \x1A = escape y x = referencia del mensaje.

### 3.9.3.2 Aplicación TCP.

En la Tabla 3-10 se describen cada uno de los comandos necesarios para llevar a cabo la aplicación TCP/IP, en la que el módulo se configura como cliente TCP y se conecta a un servidor en un ordenador remoto para enviar los datos correspondientes a las mediciones de las variables físicas.

Tabla 3-10  
Comandos básicos para la aplicación TCP

Aplicación TCP/IP		
Comando	Respuesta esperada	Descripción
AT+QIDEACT<CR>	DEACT OK	Cierra una posible conexión previa a la sesión actual.
AT+IPR=115200&W<CR>	OK	Setea el baudrate.
AT+CPIN?<CR>	+CPIN: READY OK	Verifica si la SIM card está protegida por código PIN.
AT+CREG?<CR>	+CREG:0,1	Verifica si la SIM card está registrada.
AT+QIFGCNT=0<CR>	OK	Setea el context 0 como FGCNT.
AT+QICSGP=1,"APN"<CR>	OK	Setea el APN.
AT+QIDNSIP=1<CR>	OK	Cambia el formato decimal de la IP a formato DNS.
AT+QIMUX=0<CR>	OK	Deshabilita la función MUXIP.
AT+QIMODE=1<CR>	OK	Setea el modo de la sesión a Transparent.
AT+QITCFG=3,2,512,1<CR>	OK	Configura la transferencia del modo Transparent.
AT+QIOPEN="TCP","DNS","Puerto"<CR>	CONNECT OK	Se conecta al extremo remoto subyacente.
AT+QISTAT<CR>	OKSTATE:CONNECT OK	Verifica el estado de la conexión.
AT+QICLOSE	CLOSE OK	Cierra la conexión.
+++	-	Cambia a modo comando.
ATO<CR>	-	Cambia a modo datos.

Nota: CR = retorno de carro.

La Figura 3-43 ilustra el diagrama de flujo para llevar a cabo la aplicación TCP.

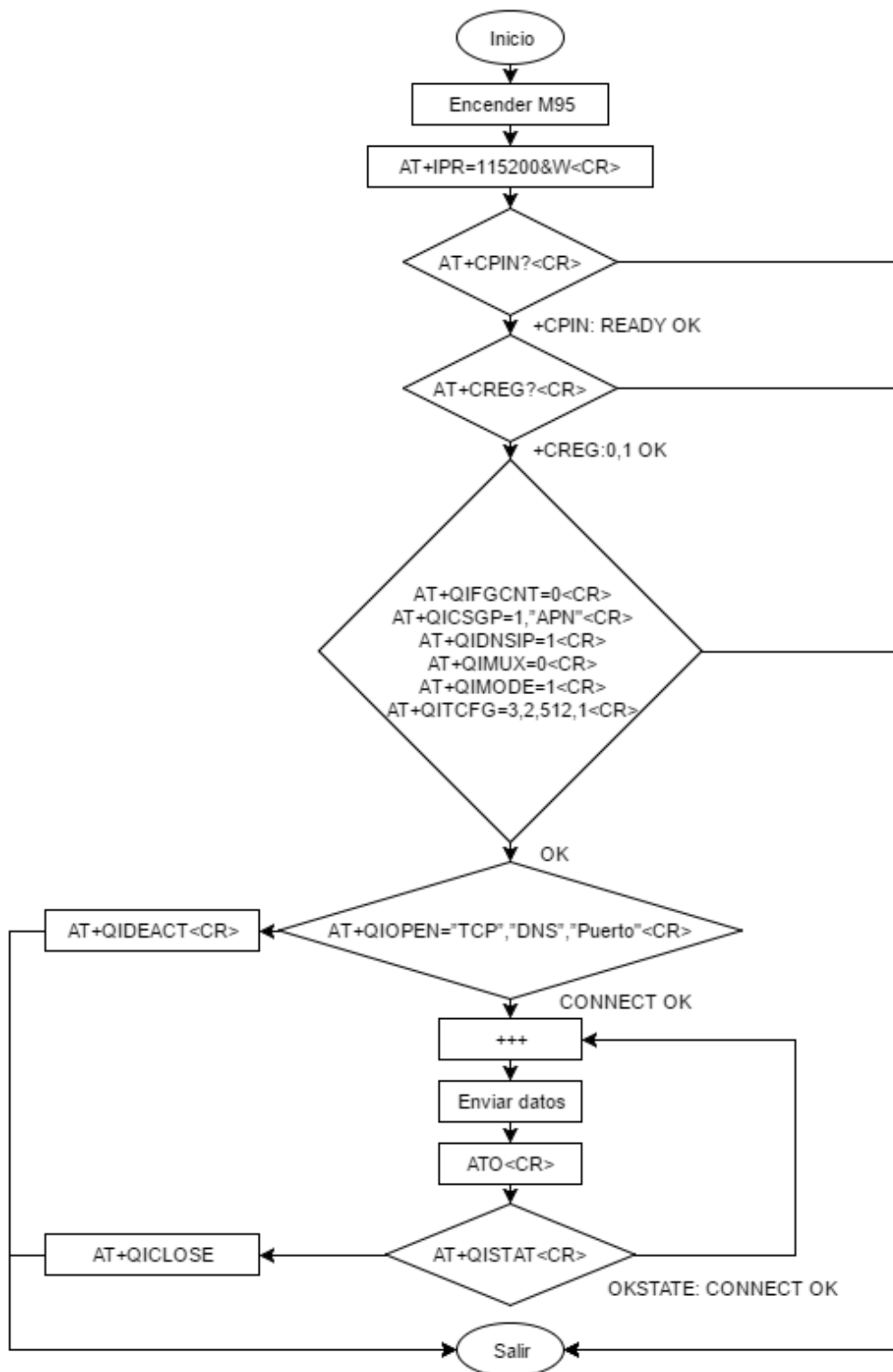


Figura 3-43. Módulo de comunicación, aplicación TCP

### 3.10 Teclado

El teclado es el dispositivo de entrada mediante el cual el operario elige entre las tres opciones que ofrece la estación para enviar los datos al dispositivo remoto. Sólo se utilizan tres de las teclas, cada una de estas tiene asignada una de las funciones de envío respectivamente, que se seleccionarán al usuario presionar la tecla correspondiente. La tarjeta CY8CKIT-042 dispone del hardware necesario para operar este periférico, por lo que sólo es necesario conectarlo directamente a la tarjeta. La Figura 2 42 ilustra el circuito de conexión con la CY8CKIT-042. En el firmware se hace uso de las funciones propias del IDE de programación.

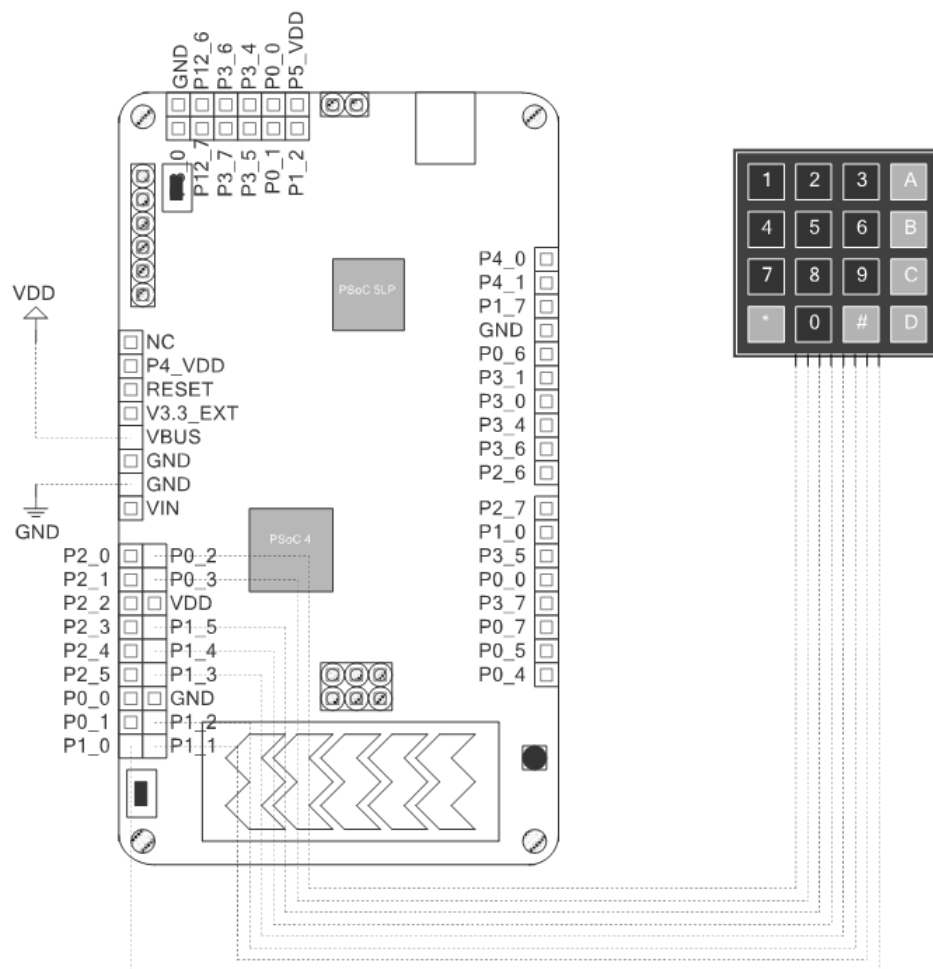


Figura 3-44. Teclado, circuito de conexión con la tarjeta CY8CKIT-042

### 3.11 Diseño del firmware para la tarjeta CY8CKIT-042

La CY8CKIT-042 embebe el hardware necesario para conectar cada uno de los periféricos del prototipo de estación meteorológica, como interfaces I2C, UART, pines de propósito general, entre otros. Naturalmente, ésta ejecuta el firmware que realiza el control sobre estos. El desarrollo del firmware se realizó en *PSoC Creator* de *Cypress*, éste es un software IDE (*Integrated Development Environment*) de programación que integra las herramientas necesarias para el desarrollo de proyectos basados en las familias de dispositivos PSoC manufacturados por *Cypress*. Se puede descargar gratuitamente desde la página web del fabricante en la dirección [www.cypress.com](http://www.cypress.com).

#### 3.11.1 Consideraciones.

El diseño del firmware para el prototipo de estación meteorológica se realizó de acuerdo a los siguientes criterios:

- Hacer uso de una cantidad mínima de memoria Flash y RAM sin perder calidad en la ejecución de las funciones. Esto con el objetivo de dejar recursos disponibles para posibles mejoras o extensiones del prototipo.
- Respuesta ante fallos en el sistema energético, estos es, recuperar la sesión previa a la interrupción en la alimentación sin intervención de un operador. Esto con el fin de evitar desplazamientos hasta el lugar en donde se encuentra el prototipo en caso de estar ubicado en un lugar remoto.
- Proveer un método de selección entre las diferentes opciones para enviar los datos al equipo remoto, así como de una opción de salida de la sesión establecida.



- Visualizar los procesos internos claves a medida que se ejecuten y los mensajes de estado o eventos.
- Realizar la adquisición del valor obtenido en la medición realizada por el respectivo sensor para cada una de las variables físicas involucradas.
- Procesar y dar formato a los datos obtenidos de los sensores acerca del valor de las variables físicas medidas.
- Enviar los datos acerca del valor de las variables físicas medidas a un computador local a través de una interfaz serial compatible con la norma RS232.
- Conectar con un ordenador remoto a través de la red móvil y enviar periódicamente a éste los datos acerca del valor de las variables físicas medidas.
- Enviar los datos acerca del valor de las variables físicas medidas a un celular a petición de un usuario cualquiera mientras éste pertenezca al mismo operador de la estación.
- Bloquear la recepción de llamadas para evitar que se deshabilite el sistema.

### **3.11.2 Descripción.**

Para recuperar la sesión previa en caso de una interrupción en la alimentación el programa emula una EEPROM en la memoria Flash del microcontrolador central de la CY8CKIT-042 y guarda en esta un flag de acuerdo a la opción elegida para enviar los datos al equipo remoto. Al iniciar la estación, luego de inicializar las rutinas básicas del hardware asociado el programa recupera el valor del flag guardado en la EEPROM emulada, si el valor recuperado corresponde al valor por defecto el programa sigue su curso normal y espera a que el usuario realice la elección entre las diferentes opciones para enviar los datos; posteriormente a la elección, guarda el flag

correspondiente en la EEPROM emulada. En caso contrario pasa a seleccionar automáticamente la opción con la que se corresponda el valor del flag recuperado. La Figura 3-45 ilustra el diagrama de flujo seguido para la elaboración de la función principal.

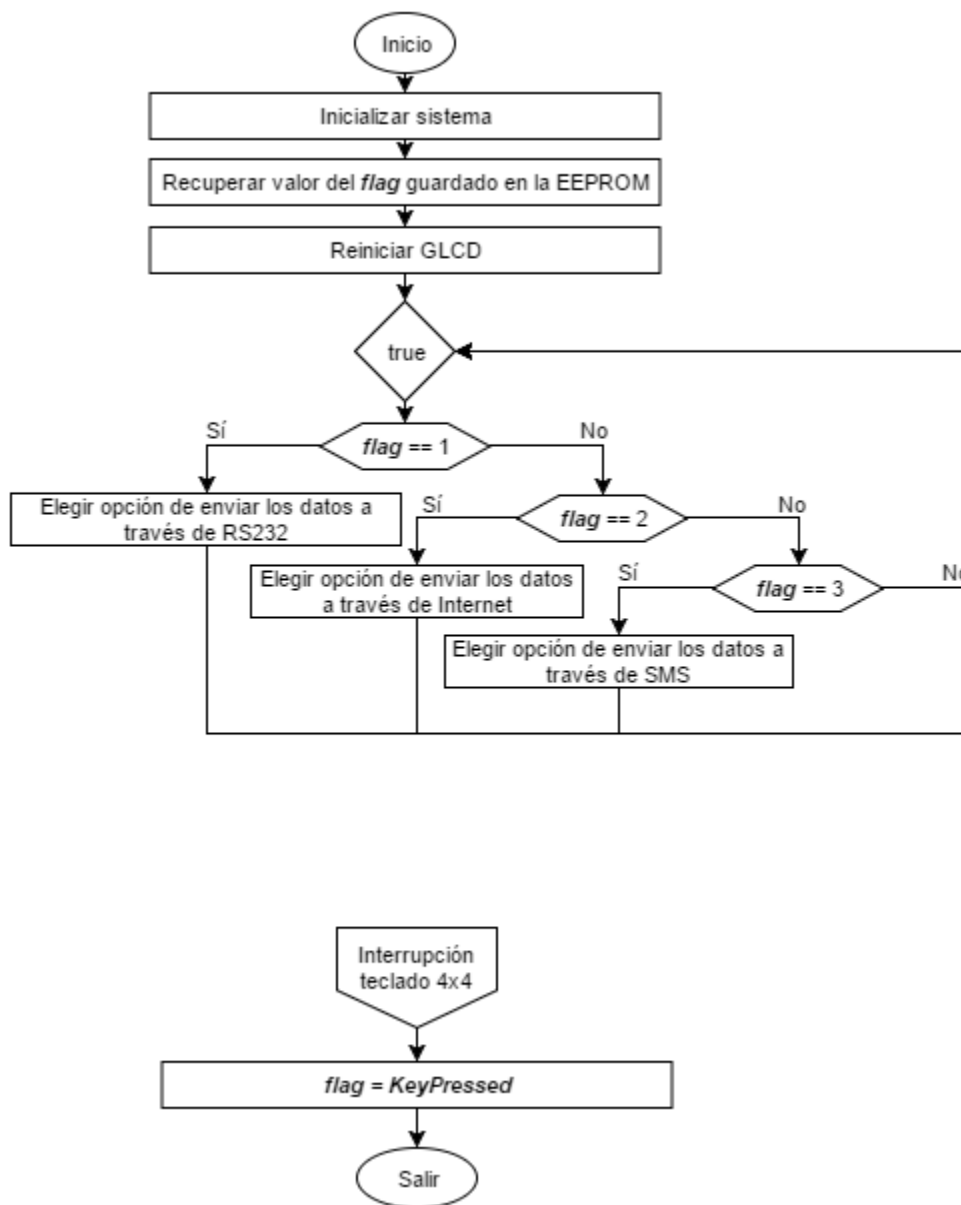


Figura 3-45. Diagrama de flujo de la función Main

La selección de la opción para enviar los datos al equipo remoto se hace a través del teclado matricial. En el módulo de visualización se imprime un menú de opciones en el que se enumeran cada una de las funciones de envío, de acuerdo a la tecla presionada el programa cargará la opción correspondiente. Estas son el envío de datos mediante comunicación serial RS232, envío de datos a través de GPRS y envío de datos a través de SMS. Las Figuras Figura 3-46Figura 3-47Figura 3-48 ilustran el diagrama de flujo seguido para la elaboración del código correspondiente para cada una de estas opciones respectivamente. Una vez se sale de la opción seleccionada se reseteará el valor por defecto del flag en la EEPROM emulada.

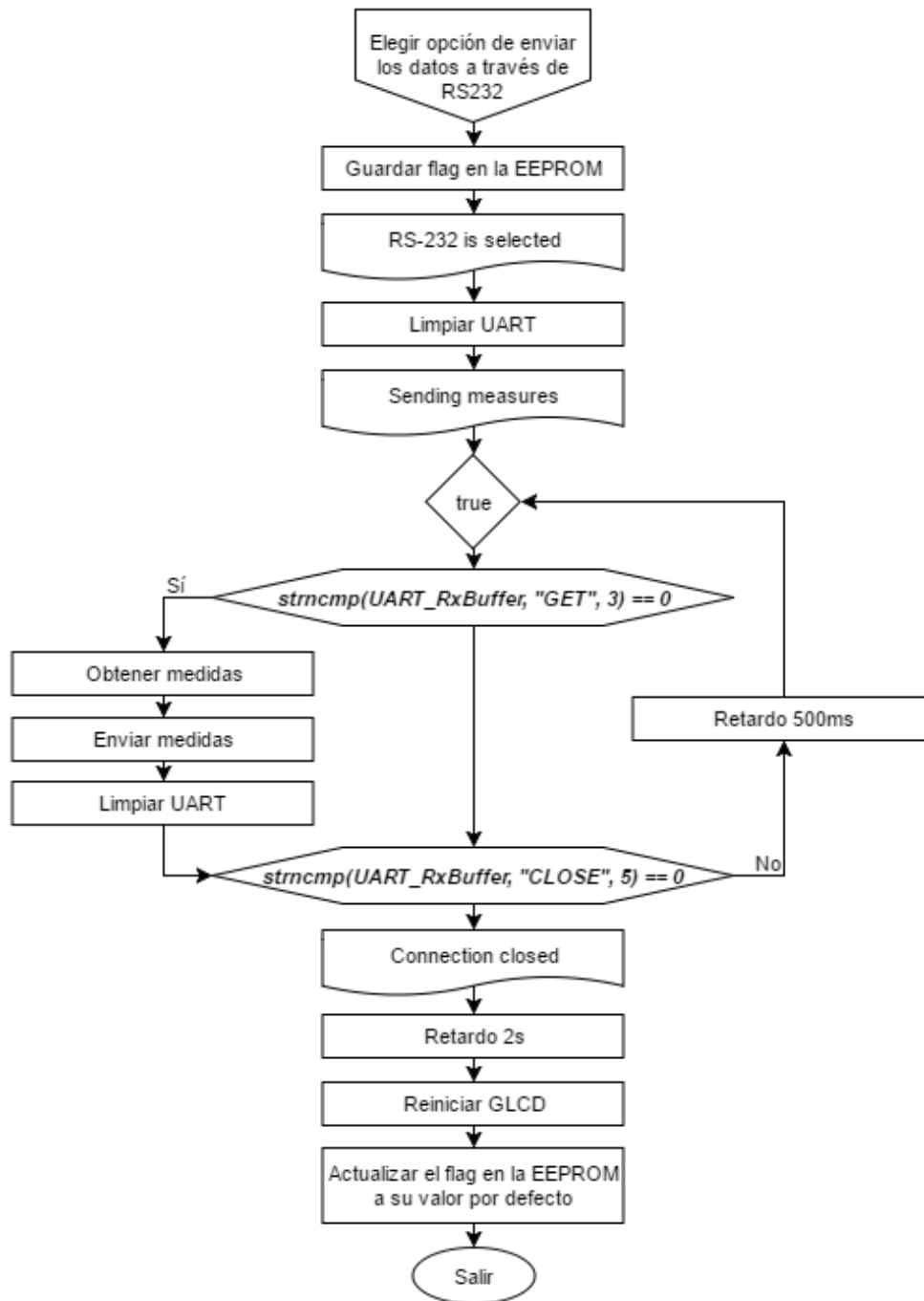


Figura 3-46. Diagrama de flujo de la función para enviar los datos a través de comunicación serial RS232

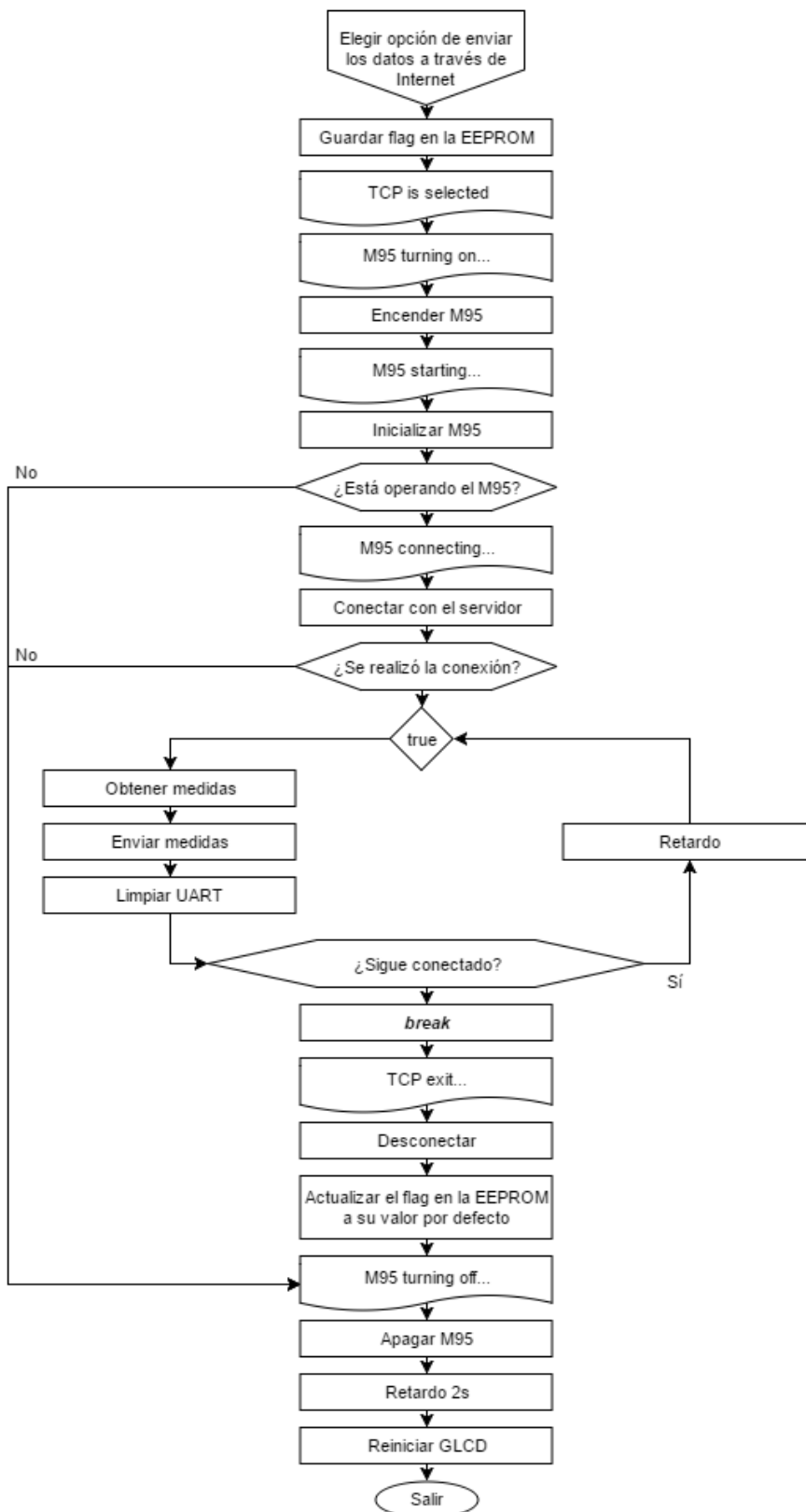


Figura 3-47. Diagrama de flujo de la función para el envío de datos a través de GPRS

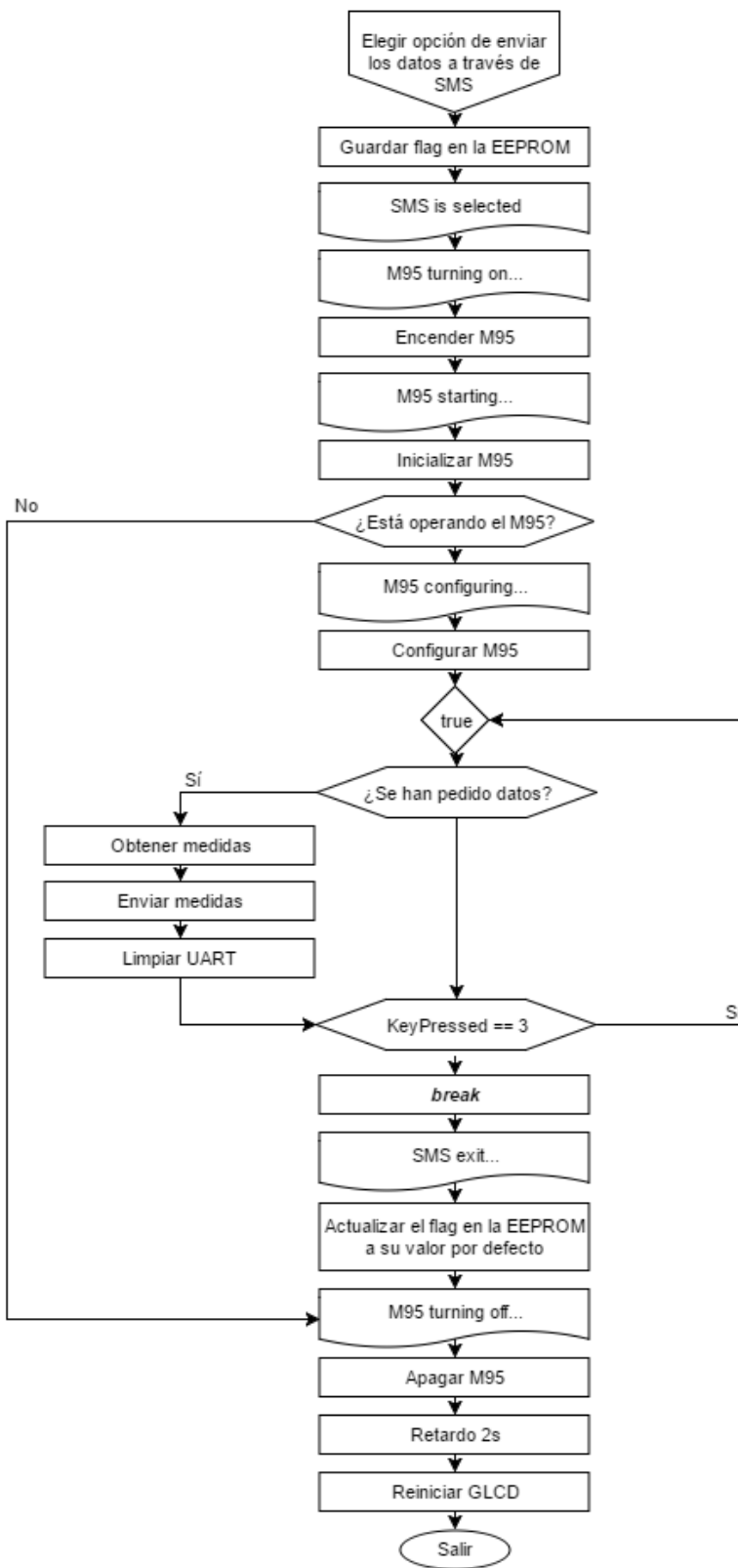


Figura 3-48. Diagrama de flujo de la función para el envío de datos a través de SMS

### 3.12 Diseño de la aplicación en C#

La realización del prototipo de estación meteorológica implica el desarrollo de una interfaz gráfica para visualizar los datos correspondientes a las variables físicas medidas, entre otras funciones. La programación de esta interfaz se hizo en C#, un lenguaje orientado a la programación de objetos, nativo del *Framework .NET* de *Microsoft*, en este caso, la versión 4.5.2. Naturalmente, el sistema operativo de destino es Windows. El desarrollo de la interfaz se hizo en *Microsoft Visual Studio*, un software IDE (*Integrated Development Environment*) facilitado por Microsoft a través de su web, este software integra las herramientas necesarias para el desarrollo de aplicaciones para los productos de Microsoft. Se puede descargar gratuitamente de la página web del fabricante en la dirección [www.microsoft.com](http://www.microsoft.com).

#### 3.12.1 Consideraciones.

En el diseño de la interfaz gráfica en Windows para prototipo de estación meteorológica se tuvieron en cuenta las siguientes pautas:

- Capacidad de recibir y enviar datos a través del puerto serial que emula la CY8CKIT-042 en el ordenador.
- Habilitar un socket en el ordenador anfitrión para recibir la solicitud de conexión de la estación meteorológica a través de TCP/IP.
- Interpretar, dar formato y visualizar los datos recibidos a través de comunicación serial o a través de TCP/IP y que corresponden a los valores de las mediciones realizadas a las distintas variables físicas de las que se ocupa de medir la estación.
- Opción de exportar los datos previamente procesados a un fichero de Excel.
- Mostrar un registro de los últimos datos recibidos de la estación meteorológica.

- Graficar todos los datos que se reciban acerca del valor de la temperatura ambiente, la humedad relativa, la presión atmosférica y la rapidez del viento.
- Opción de capturar la pantalla ocupada por la aplicación y guardar la captura como una imagen en el ordenador.
- Diseño agradable al usuario y facilidad en el manejo de la aplicación.
- Portabilidad y uso mínimo de los recursos del ordenador anfitrión.

### 3.12.2 Descripción.

La Figura 3-49 ilustra el aspecto físico de la interfaz gráfica para el prototipo de estación meteorológica; ésta consta, en esencia, de una barra de menú, una gráfica, una tabla, un panel con casillas y una barra de estado.

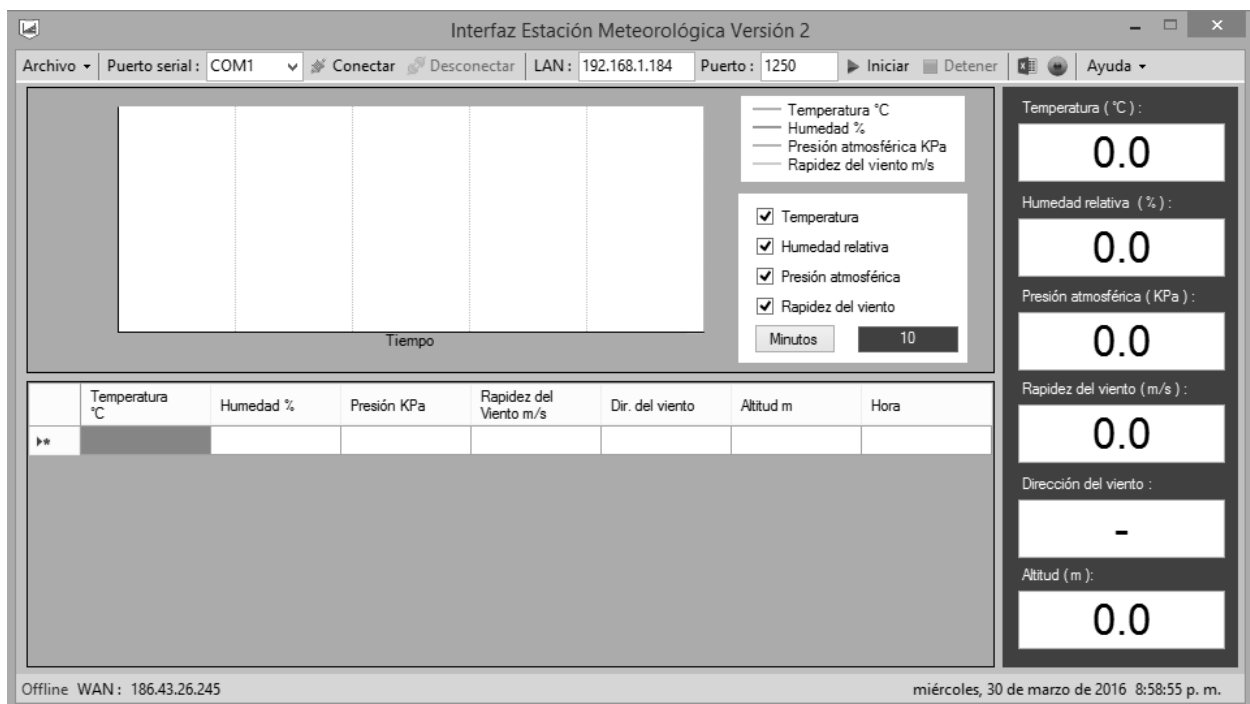


Figura 3-49. Interfaz Estación Meteorológica versión 2



### 3.12.2.1 *Barra de menú.*

La barra de menú se ilustra en la Figura 3-50, ésta se encuentra dividida en cinco secciones:

- Archivo: contiene un botón desplegable en el que se encuentra la opción para cerrar la interfaz.
- Comunicación serial: contiene una caja desplegable que lista automáticamente todos los puertos seriales disponibles en el equipo anfitrión. También contiene un botón para conectar con la estación a través del puerto serial seleccionado en la caja desplegable y un botón para desconectarse de ésta. Cuando se realiza esta conexión se deshabilita la opción de conexión a través de TCP/IPv4.
- Conexión TCP/IPv4: contiene una caja de texto en la que se muestra automáticamente la IP privada del equipo anfitrión y otra caja de texto para asignar el puerto a través del cual se recibirá la solicitud de conexión de la estación. Si se modifica el puerto por defecto también se debe modificar el puerto de destino en el firmware de la estación. Cuando se realiza esta conexión se deshabilita la opción de conexión a través del puerto serial.
- Opciones: contiene un botón para crear el fichero de Excel en donde se guardarán los datos recibidos acerca del valor de las variables físicas medidas por la estación y un botón para hacer una captura de pantalla.
- Ayuda: contiene un botón desplegable en el que se encuentra una opción para visitar la web del programador de la interfaz y una opción para ver la información acerca de la aplicación.

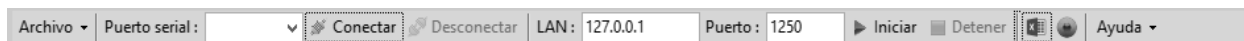


Figura 3-50. Interfaz Estación Meteorológica, barra de menú

### 3.12.2.2 Gráfica.

La grafica se ilustra en la Figura 3-51, en esta se grafica el valor recibido de la temperatura ambiente, la humedad relativa, la presión atmosférica y la rapidez del viento. Del lado derecho incluye un panel en el que se controlan mediante una casilla de validación las variables que se desea mostrar en la gráfica. Como las variables se grafican en función del tiempo, se programó un botón para ajustar el rango de tiempo para el que se muestran los valores en la gráfica de acuerdo al valor especificado en la caja de texto, el programa interpreta este valor como minutos.

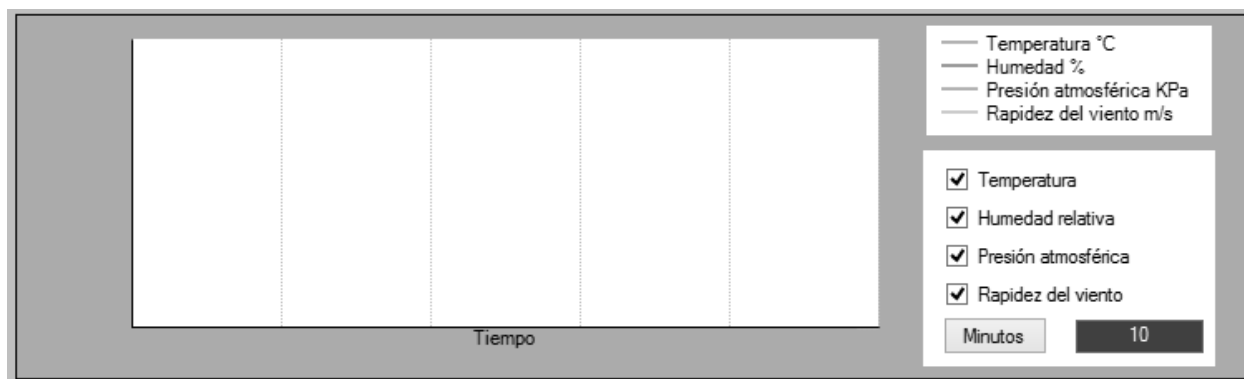


Figura 3-51. Interfaz Estación Meteorológica, gráfica

### 3.12.2.3 Tabla.

La tabla se ilustra en la Figura 3-52, en esta se listan los ocho últimos valores recibidos para cada una de las variables, además de la hora en la que se recibieron.

	Temperatura °C	Humedad %	Presión KPa	Rapidez del Viento m/s	Dir. del viento	Altitud m	Hora
*							

Figura 3-52. Interfaz Estación Meteorológica, tabla

### 3.12.2.4 Panel.

El panel se ilustra en la Figura 3-53, éste consta de una casilla para cada variable física medida por la estación, y en ellas se muestra el último valor recibido para la variable que representan.



Figura 3-53. Interfaz Estación Meteorológica, panel

### 3.12.2.5 Barra de estado.

La barra de estado se ilustra en la Figura 3-54, consta de una etiqueta para mostrar el estado de la conexión; que puede ser “*Offline*” si no ha iniciado o no está esperando conexión con la estación, “*Online*” si está esperando la solicitud de conexión por parte de la estación y “*Estación conectada*” cuando finalmente se ha realizado la conexión. También consta de una etiqueta en la que se muestra automáticamente al iniciar la aplicación la IP pública del equipo anfitrión. En el extremo derecho dos etiquetas muestran la fecha y la hora actual respectivamente.

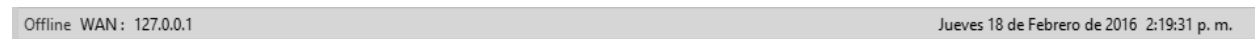


Figura 3-54. Interfaz Estación Meteorológica, barra de estado

### 3.12.3 Configurar red del equipo anfitrión.

En las redes que conforman La Internet los equipos terminales o *host* y los equipos de enrutamiento o encaminamiento se identifican con una dirección única conocida como dirección IP. Esta dirección consta de 32 bits (en el caso de implementar IPv4, *Internet Protocol Version 4*) divididos en 4 bytes y separados por un punto, Ej. 192.168.1.1; una parte de la IP representa la red y el resto representa al *host*. Si se utilizan 8 bits para representar a la red y 24 bits para representar al *host* se dice que es una IP **Clase A**, si se utilizan 16 bits para representar a la red y 16 bits para representar al *host* se dice que es una IP **Clase B** y si se utilizan 24 bits para representar a la red y 8 bits para representar al *host* se dice que es una IP **Clase C**. Hay otros tipos de IP, pero las mencionadas previamente se distinguen como básicas. Para identificar los bits de la IP que corresponden a la red y los bits que corresponden al *host* se utiliza la **Máscara de Subred**, ésta tiene el mismo formato de la dirección IP y marca como uno lógico la cantidad de bits que se

correspondan con los bits de la dirección IP que identifican a la red y como cero lógico al resto. Por ejemplo, si se trata de una IP Clase C la máscara de subred respectiva sería 225.225.225.0. El número de bits utilizados para representar al *host* determinan la cantidad de dispositivos que pueden conectarse a la red, de manera que, si se trata de una IP Clase C, el número de dispositivos que puede albergar la red subyacente es de 256 host (menos un par de direcciones que están reservadas). Cuando una persona natural se suscribe a un ISP (*Internet Service Provider*) normalmente la empresa le otorga una **IP Pública** habilitada para conectarse a Internet, ésta IP es asignada automáticamente por el DHCP (*Dynamic Host Configuration Provider*) o el ISP al iniciar el router/módem. Cada vez que se reinicie el router/módem esta IP cambiará (también se puede configurar para que sea estática), entonces se dice que es una **IP dinámica**. Como esta IP sólo identifica a un host en La Internet, para extender el número de dispositivos que se puede conectar a Internet en el hogar se crea una red LAN (*Local Area Network*) entre equipos que comparten la misma IP pública antes mencionada; de manera que, de un lado del router/módem está una red WAN (*Wide Area Network*) y del otro lado está una red LAN, incompatibles entre sí; para solucionar este inconveniente existe un protocolo conocido como NAT (*Network Address Translation*) que se encarga de realizar la traducción entre las direcciones de redes que son incompatibles. Dentro de la red LAN el usuario puede configurar manualmente la dirección IP (**IP Privada**) para cada equipo de su red o puede habilitar el DHCP para que estas sean asignadas automáticamente. En un *host* son muchas las aplicaciones de Internet que se pueden ejecutar al mismo tiempo, al igual que el *host*, cada aplicación debe tener una dirección que la represente dentro del equipo, así el router/módem puede identificar la aplicación a la que debe dirigir los datos que pasen a través de él, esta dirección se conoce como **Puerto** y sólo puede ser usada por una aplicación a la vez para evitar la ambigüedad.

### 3.12.3.1 Configurar IP privada como estática.

En un equipo con el DHCP activado, cada vez que éste se conecte a la red es posible que se le asigne una dirección IP diferente a la que tenía anteriormente estando conectado a la misma red. Si la nueva IP asignada no coincide con la IP válida para el puerto mapeado a la aplicación servidor en el router/módem, aunque el equipo ejecute la aplicación ésta no será visible desde La Internet, es por ello que se debe asignar una IP estática. Para configurar la IP privada en Windows se debe ir a *Panel de control\Redes e Internet\Conexiones de red*, allí se enlistarán los adaptadores de red del equipo. Luego se debe hacer clic derecho sobre el adaptador de red usado para acceder a Internet, en el menú inmediato que aparece se debe elegir la opción de **“Propiedades”**, acto seguido se abrirá la venta que se ilustra en la Figura 3-55.

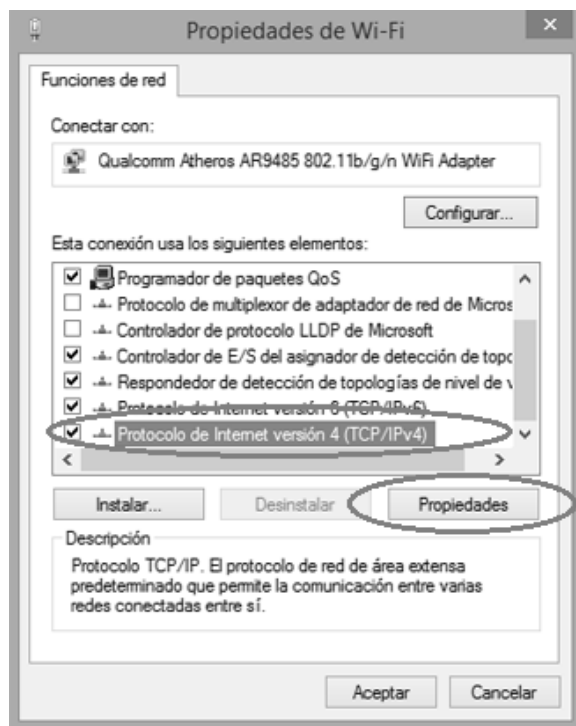


Figura 3-55. Propiedades del adaptador de red



Figura 3-56. Configurar IP privada como estática

A continuación se debe seleccionar la opción “*Protocolo de Internet versión 4 (TCP/IPv4)*” y luego hacer clic sobre “**Propiedades**”. En la nueva ventana que aparecerá se asigna manualmente la IP privada (véase la Figura 3-55). Allí mismo se pueden asignar las direcciones de los servidores DNS (*Domain Name System*). La Puerta de enlace predeterminada es la dirección del router/módem.

### **3.12.3.2 Mapear un puerto en el router/módem a la aplicación servidor.**

Para el funcionamiento correcto de la aplicación servidor se debe mapear en el router/módem el puerto que ésta utiliza. Esto se hace accediendo al equipo. Indistintamente de la marca y el modelo los *setups* suelen tener una interfaz similar. Acceder al *setup* normalmente se hace a través del navegador, escribiendo la dirección de la puerta de enlace predeterminada (para consultar la puerta de enlace predeterminada en Windows se puede abrir el **Símbolo del Sistema** tecleando “**Windows + R**” y luego ejecutar el comando “**ipconfig**”). Luego cargará una interfaz para hacer login (las credenciales para validar la sesión suelen venir escritas en el chasis del dispositivo). En el caso de un módem Huawei HG532e se hace como se ilustra en la Figura 3-57. Los campos que se deben llenar normalmente son los mismos en todos los módems, en este caso: un campo para asignar la interfaz, un campo para asignar el protocolo de transporte, un campo para asignar el *host* remoto del cual se espera recibir la solicitud de conexión (como la IP de la estación es dinámica este campo se dejó vacío, así no restringe la conexión a una sola dirección), campos para asignar el puerto (este debe ser el mismo que utiliza la aplicación servidor), un campo para asignar al *host* en donde se ejecutará la aplicación servidor y un campo para asignar un nombre el puerto mapeado.

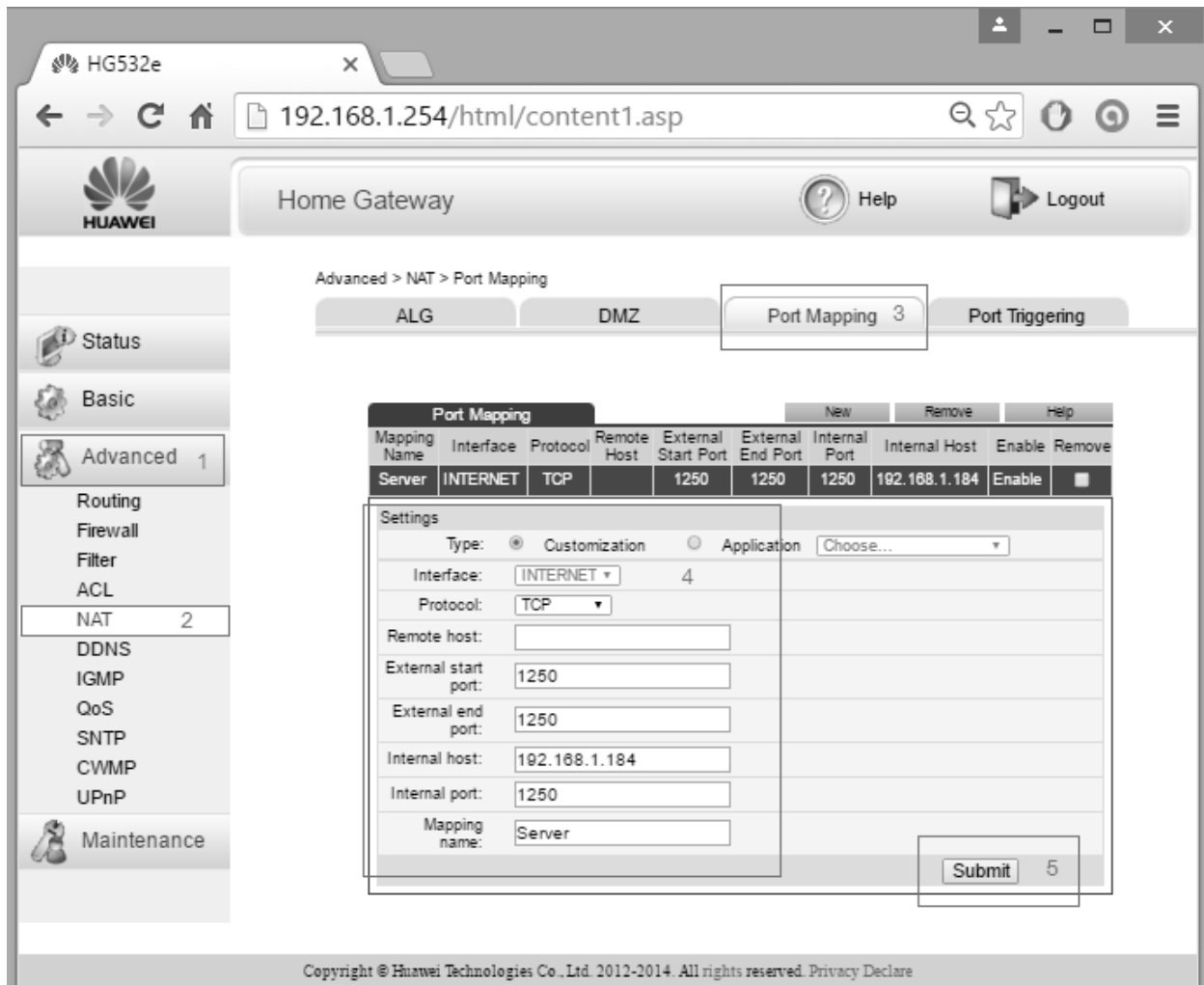


Figura 3-57. Mapear puerto a la aplicación servidor

### 3.12.3.3 Habilitar puerto en el Firewall de Windows.

Al ejecutar la aplicación servidor por primera vez dentro del host el Firewall de Windows suele saltar un mensaje de alerta en el que pregunta al usuario si está de acuerdo en permitir el uso del puerto por dicha aplicación (Figura 2 39). Si se elige aceptar, el sistema operativo habilitará el puerto en el *Firewall* para dicha aplicación.





Figura 3-58. Validar uso del puerto por la aplicación servidor en el Firewall de Windows

#### 3.12.3.4 No-IP.

Es conocido que siempre que se reinicie el router/módem la IP pública asignada por el ISP será diferente de la IP de la anterior conexión. Para poder conectarse al equipo servidor la estación meteorológica debe conocer esta dirección y el puerto asociado a la aplicación, por lo anterior, se hace necesario el uso de un servidor DNS (*Domain Name System*), que asigna un nombre legible a las direcciones IP, así, por más que éstas cambien, sólo se debe memorizar el nombre de dominio para encontrar a los servidores en la red. Para solucionar este inconveniente se creó una cuenta en *www.noip.com*, ésta es una empresa que ofrece un número limitado de nombres de dominio a suscriptores con una cuenta gratis. De esta manera, la estación siempre encontrará al servidor en la red, independiente de la IP pública de éste. Una vez creada la cuenta se debe registrar el *host*, luego se debe descargar una aplicación facilitada en el mismo portal, esta aplicación se encarga de

verificar periódicamente cambios en la dirección de red del *host* y por tanto, debe ejecutarse siempre que se pretenda conectarse con la estación, ésta aplicación se ilustra en la Figura 3-59, al minimizarla se ejecutará en segundo plano.

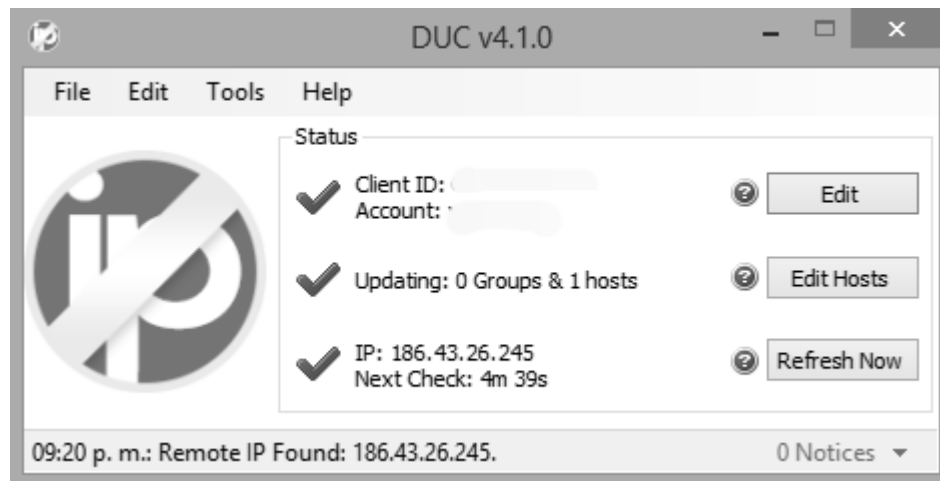


Figura 3-59. DUC v4.1.0

### 3.13 Diseño de la aplicación para Android.

Mientras la opción para enviar los datos al equipo remoto sea la indicada, la estación devolverá una respuesta con el valor medido de todas las variables físicas para las que está capacitada sensar a cualquier móvil del que reciba un carácter como mensaje siempre y cuando este esté suscrito al mismo operador o compañía móvil de la estación, exceptuando también, a los usuarios que hayan cambiado de compañía y se hayan quedado con el mismo número de celular. El formato del mensaje de respuesta es como sigue: *Temperatura-Humedad-Presión-Rapidez del viento-Dirección del viento-Altitud*. Para hacer amena la interpretación de la respuesta de la estación se realizó una aplicación móvil sólo para usuarios Android. Ésta se desarrolló en *App Inventor*, de *Google*; una herramienta online en la que se pueden realizar aplicaciones sencillas para Android mediante programación modular.

### 3.13.1 Consideraciones.

En el desarrollo de la aplicación móvil se hizo de acuerdo a los siguientes criterios:

- Método para asignar el destinatario (número de la estación).
- Método para solicitar a la estación el envío de los datos.
- Método para visualizar el valor obtenido de las variables físicas medidas por la estación.
- Método para almacenar el valor de las variables físicas.

### 3.13.2 Descripción.

La Figura 3-60 ilustra la aplicación móvil para el prototipo de estación meteorológica; ésta consta, en esencia, de una caja para introducir el número celular del servidor, un botón para solicitar el envío de los datos de las variables medidas y un panel de visualización donde se tabulan los datos enviados por la estación una vez son recibidos a través de un mensaje SMS; estos también son almacenados por orden de llegada en un fichero CSV compatible con Excel.

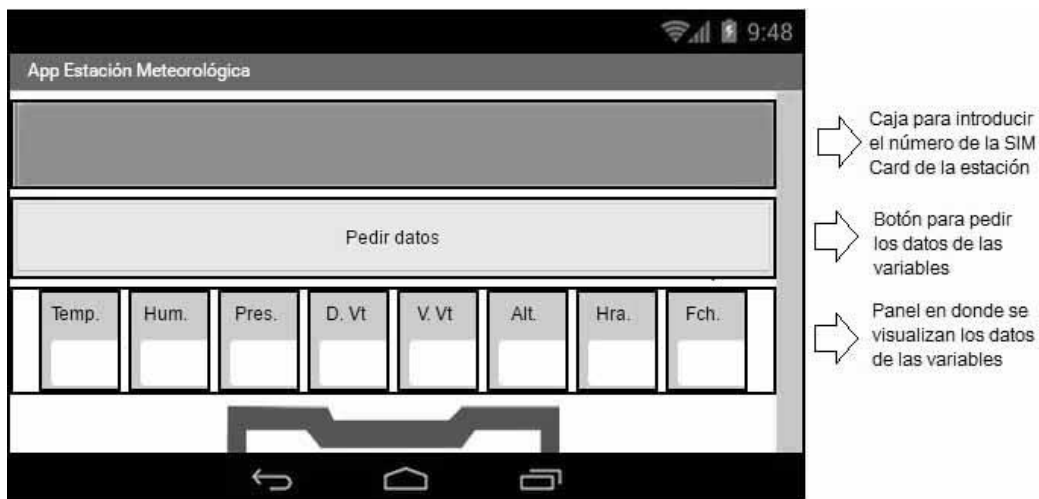


Figura 3-60. Aplicación móvil del prototipo de estación meteorológica

## Capítulo 4

### Resultados y Análisis

Se logró diseñar y desarrollar un prototipo de estación meteorológica que permite sensar la temperatura ambiente, la humedad relativa, la presión atmosférica, la velocidad y dirección del viento y la altitud del lugar en donde se encuentre operando, y enviar la información de las variables medidas a un dispositivo remoto, móvil u ordenador, de forma alámbrica a través de comunicación serial basada en el protocolo RS232 o inalámbrica a través de GSM/GPRS. El prototipo de estación meteorológica realizada se ilustra en la Figura 4-1.



*Figura 4-1. Prototipo de estación meteorológica*

A continuación se presentan datos recogidos de pruebas de campo realizadas.

#### 4.1 Envío de datos a través de comunicación serial

La Figura 4-2 ilustra la interfaz gráfica del prototipo de estación meteorológica operando para la opción de envío de los datos a través de comunicación serial basada en el protocolo RS232. En el prototipo, se debe elegir la opción correspondiente. Al iniciar la interfaz se cargan en el panel correspondiente los puertos seriales activos en la computadora anfitrión. El usuario debe seleccionar el puerto correspondiente a la estación y luego dar clic en el botón “Conectar”. Inmediatamente se deshabilitará este botón y se habilitará el botón de “Desconectar”. En la barra de estado saldrá el estado de la conexión como “Estación conectada”. En la interfaz se visualizarán, tabularán y graficarán los datos enviados por prototipo conforme estos sean recibidos.

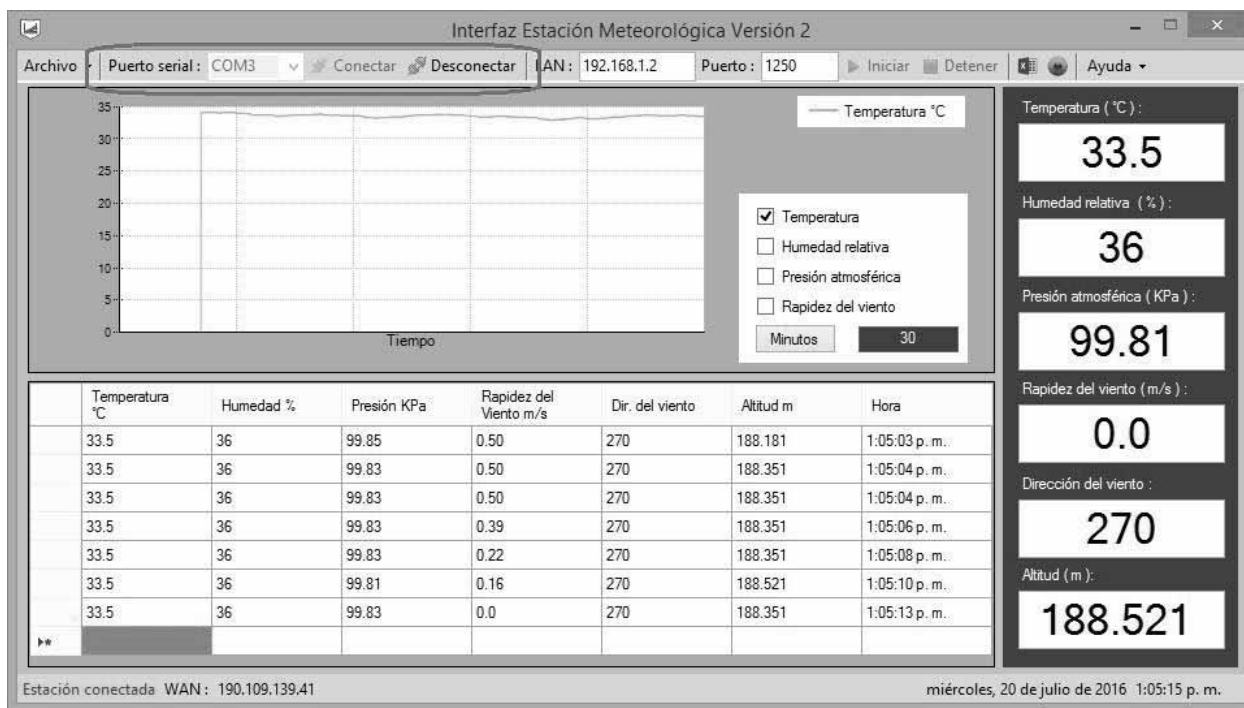


Figura 4-2. Prototipo de estación meteorológica, envío de datos a través de comunicación serial

Por tratarse de comunicación serial los datos se envían en periodos de tiempos muy reducidos debido a que hacer esto no implica costos económicos. De esta forma se obtienen mayor cantidad de muestras.

La distancia máxima a la que puede estar el computador anfitrión del prototipo de estación meteorológica se especifica en la norma RS232.

Antes de conectar con la estación se debe verificar que el puerto correspondiente no está siendo ocupado por otra aplicación y que está funcionando correctamente, esto último se puede hacer a través de **Administrador de Dispositivos de Windows**.

Para salir de esta opción sólo debe darse clic en el botón “**Desconectar**” y la interfaz regresará a su estado inicial. Si se interrumpe la alimentación, en la estación se cargará automáticamente esta opción al reanudarse la energía.

Si el computador anfitrión tiene problemas de lentitud, los datos al ser tabulados podrían colisionar.

#### **4.2 Envío de datos a través de GPRS**

Al iniciar la interfaz ésta comprueba si el equipo anfitrión tiene acceso a Internet, en caso tal, en la barra de estado aparecerá la dirección IP pública y en el panel correspondiente a esta opción en la barra de menú aparecerá la IP privada y el puerto por defecto. En caso contrario aparecerá en la barra de estado “**Medios desconectados**”. Tanto el puerto como la dirección privada deben estar mapeados a la aplicación interfaz en el router. El puerto por defecto es el 1250, si éste se cambia debe regrabarse el firmware de la estación con el valor del nuevo puerto, igual

debe hacerse si se cambia el DNS que idéntica la dirección pública del equipo anfitrión. Para habilitar el DNS debe ejecutarse la aplicación No-IP.

Para recibir los datos a través de Internet, debe colocarse la Interfaz en modo escucha, para ello, debe darse clic en el botón **“Iniciar”** y luego seleccionar la opción correspondiente en la estación. Una vez ésta se conecte aparecerá el estado de la conexión en la barra de estado como **“Estación Conectada”**. El botón de **“Iniciar”** se deshabilitará y el botón de **“Detener”** se habilitará. Los datos enviados por la estación se visualizarán, tabularán y graficarán conforme sean recibidos. Lo anterior se ilustra en la Figura 4-3.

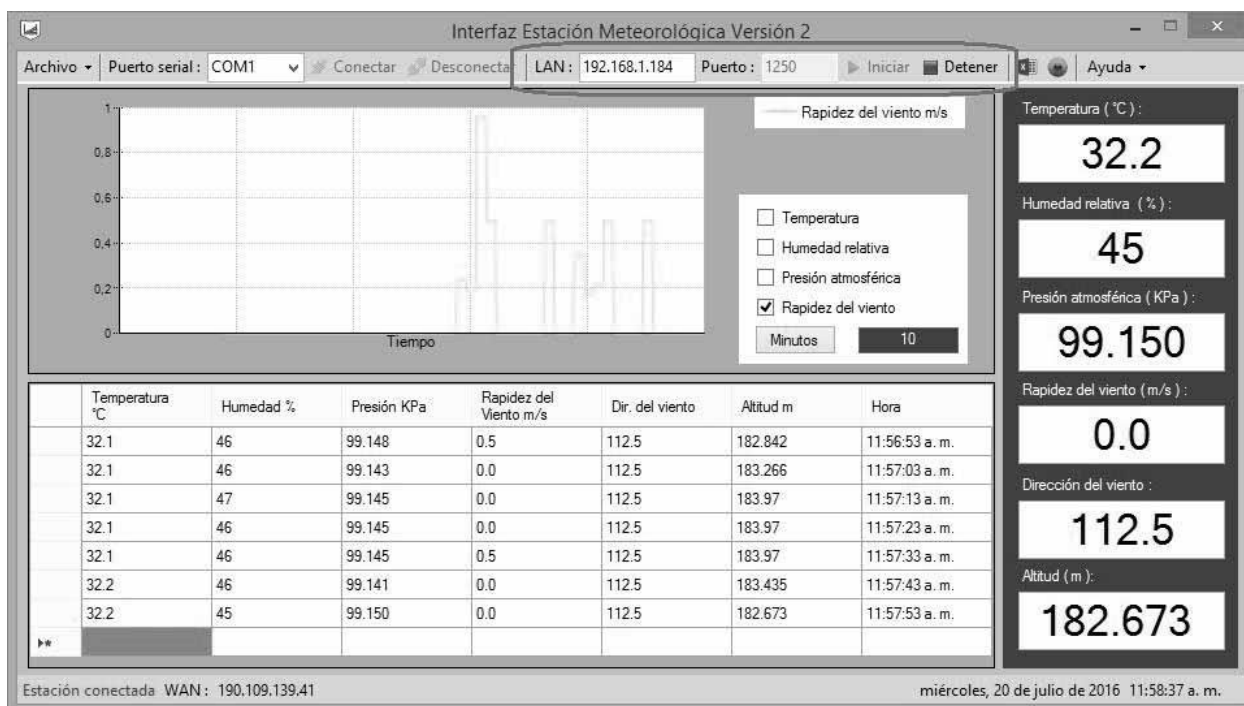


Figura 4-3. Prototipo de estación meteorológica, envío de datos a través de GPRS

Al periodo de envío se le suma la latencia de la red. Como el uso de esta opción implica costos económicos y la tasa de variación de las magnitudes físicas medidas es lenta con respecto al tiempo, puede regrabarse el firmware de la estación con un periodo de tiempo de envío mayor.

Para salir de esta opción se debe dar clic en el botón “**Detener**” y la interfaz regresará a su estado inicial. Si la alimentación en la estación se interrumpe, al reanudarse, el prototipo intentará conectar de nuevo con el servidor.

Cualquier computador con Windows, conectado a Internet y configurado correctamente podría recibir los datos de la estación a través de esta opción.

### **4.3 Envío de datos a través de SMS**

Operando bajo esta opción, cualquier usuario suscrito a la misma compañía de la tarjeta SIM del prototipo podrá solicitar el envío de datos por parte de la estación. Para ello debe enviar a la estación un SMS con cualquier carácter y ésta le responderá con los datos. Si es un móvil con sistema operativo Android puede utilizar la aplicación que se ilustra en la Figura 4-4 para hacer lo descrito anteriormente. La aplicación, además de darle formato a los datos recibidos, también los exporta a un archivo CSV (*Comma-Separated Values*) que puede traducirse a Excel fácilmente, éste se crea al iniciar la aplicación.



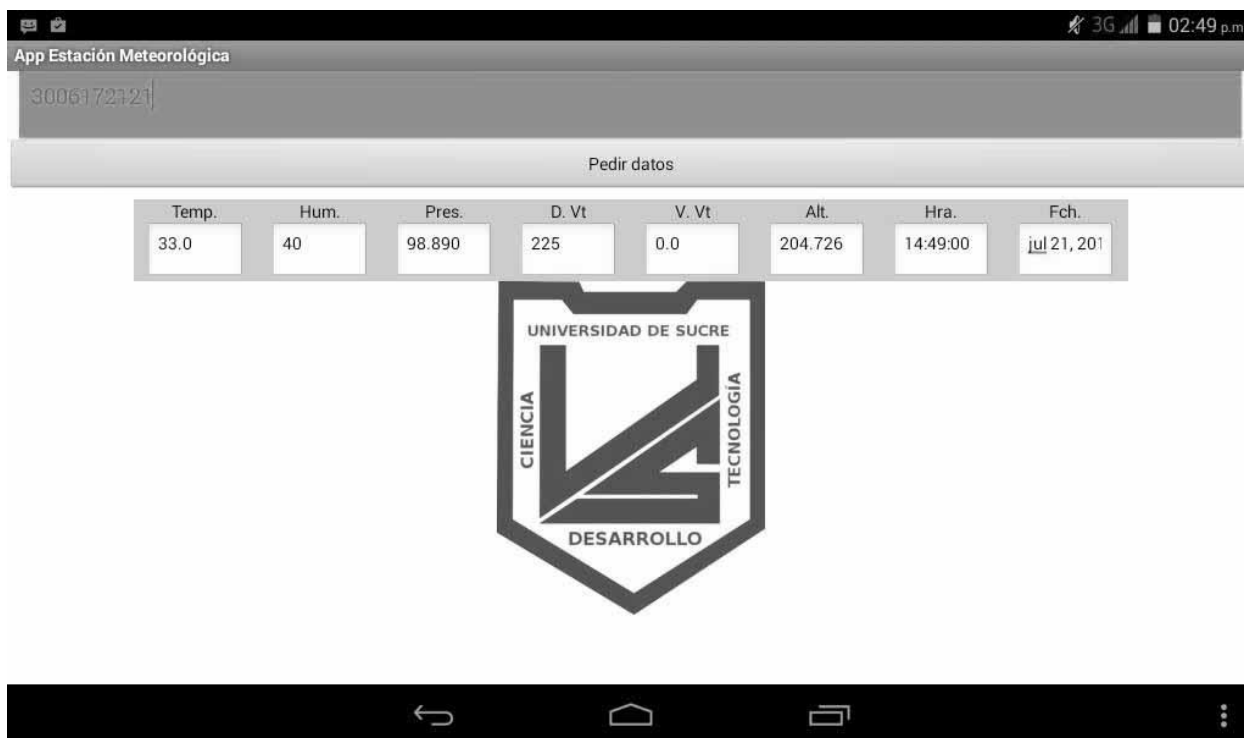


Figura 4-4. Prototipo de estación meteorológica, envío de datos a través de SMS

#### 4.4 Exportando datos

Los datos entregados por la estación son tabulados por las aplicaciones interfaces conforme estos son recibidos. Cuando son enviados al ordenador se exportan a un fichero de Excel, como se ilustra en la Figura 4-5. Por otro lado, si son enviados a un móvil se exportan a un fichero CSV, como se ilustra en la Figura 4-6, éste se puede traducir fácilmente a un fichero de Excel.

	A	B	C	D	E	F	G	H
1	Temp. °C	Hum. %	Pres. KPa	Rap.Vient m	Dir.Vient °	Alt. m	Hora	
2	34,1	37	99,105	0	67,5	186,486	12:40:39 p. m.	
3	34,1	38	99,109	0	67,5	186,147	12:40:41 p. m.	
4	34,1	38	99,109	0	67,5	186,147	12:40:44 p. m.	
5	34,1	38	99,113	0	67,5	185,808	12:40:45 p. m.	
6	34,1	38	99,107	0	67,5	186,317	12:40:47 p. m.	
7	34,1	37	99,105	0	67,5	186,486	12:40:50 p. m.	
8	34,1	38	99,111	0	67,5	185,978	12:40:52 p. m.	
9	34,1	38	99,113	0	67,5	185,808	12:40:54 p. m.	
10	34,1	37	99,109	0	67,5	186,147	12:40:56 p. m.	
11	34,1	38	99,109	0	67,5	186,147	12:40:58 p. m.	
12	34,1	38	99,105	0	67,5	186,486	12:41:00 p. m.	
13	34,1	37	99,111	0	67,5	185,978	12:41:02 p. m.	
14	34,1	36	99,107	0	67,5	186,317	12:41:04 p. m.	
15	34,1	36	99,107	0	67,5	186,317	12:41:06 p. m.	

Figura 4-5. Datos tabulados en un fichero de Excel de pruebas realizadas el 20 de julio de 2016

No.	Temp.	Hum.	Pres.	D. Vt	V. Vt	Alt.	Hra.	Fch.
1	32.9	41	98.896	270	0.0	204.217	14:43:00	jul 21, 2016
2	33.0	41	98.900	270	0.0	203.877	14:44:50	jul 21, 2016
3	33.0	41	98.898	270	0.0	204.47	14:45:50	jul 21, 2016
4	33.0	41	98.886	225	0.0	205.65	14:46:22	jul 21, 2016
5	33.0	41	98.890	225	0.0	204.726	14:47:51	jul 21, 2016
6	33.0	40	98.890	225	0.0	204.726	14:49:00	jul 21, 2016

Figura 4-6. Datos tabulados en un fichero CSV

#### 4.5 Fiabilidad de los datos

En cuanto a la fiabilidad de los datos entregados por la estación, se consideran seguros los que se obtienen de sensores inteligentes, debido a que estos emplean sus propias rutinas de calibración. No obstante, la medición de la temperatura se comparó con los valores arrojados por un multímetro para esta misma variable bajo las mismas condiciones y los datos de la estación coincidieron con los de éste. También se compararon los datos obtenidos por la estación en las mediciones de dicha variable con los datos entregados por el IDEAM en un día determinado y el resultado fue el mismo (comparar registro de la temperatura en la Figura 4-5 con el registro de la misma en la Figura 4-7). Para corroborar los datos de las mediciones de las variables restantes no se emplearon dispositivos calibrados para hacer comparaciones ni tampoco se usó una base de datos confiable que sirviera de referencia como se hizo con la temperatura, debido a que no se disponía de ninguno de los dos. En cambio, se hizo un análisis de acuerdo al ambiente en donde se realizaron las pruebas. Por ejemplo, para la humedad relativa, la estación arrojaba valores menores en ambientes con aire acondicionado (además se mide utilizando un sensor inteligente). Para la presión atmosférica, se sabe que la presión a nivel del mar es de 101325 pascales y Sincelejo, según (Alcaldía de Sincelejo - Sucre, 2010) tiene una altitud media de 213 metros sobre el nivel del mar, por lo tanto, la presión a esta altura debe estar cerca del valor antes mencionado, y así se comprobó al realizar pruebas con la estación en la ciudad de Sincelejo. Además, el cálculo de la altitud a partir del valor obtenido de la presión atmosférica corrobora lo anterior, puesto que en las pruebas realizadas se obtuvieron valores muy cercanos a la altitud antes referenciada (véase los valores registrados para la presión atmosférica y la altitud en la Figura 4-5). Para la medición de la velocidad y la dirección del viento no se usaron sensores inteligentes ni tampoco éstos fueron calibrados debido a que no se contó con las herramientas necesarias para llevar a cabo dicho paso.

Si bien el principio de funcionamiento de los mismos tiene sentido físico no se dispone de datos que evidencien la fiabilidad de las mediciones de éstos.

INSTITUTO DE HIDROLOGIA, METEOROLOGIA Y ESTUDIOS AMBIENTALES

SERVICIO DE PRONOSTICO Y ALERTAS

IDEAM

MUNICIPIO	Precipitación (mm)	Temp-max (°C)	Temp-min (°C)
	JULIO: 20	JULIO: 21	JULIO: 21
Apto. SAN ANDRES	0.0	31.0	26.6
Apto. PROVIDENCIA	0.9	30.9	25.8
Apto. SANTA MARTA	0.0	33.1	26.6
SANTA MARTA (Sede IDEAM)	0.0	X	X
Apto. CARTAGENA	0.0	32.8	28.0
Apto. BARRANQUILLA (Soledad)	0.0	34.3	26.2
BARRANQUILLA (sede IDEAM)	0.0	X	X
Apto. RIOHACHA	0.0	37.4	27.0
Apto. VALLEDUPAR	0.0	38.1	25.2
Apto. MONTERIA	0.4	32.8	22.4
SINCELEJO (Est. - Univ. de Sucre)	0.0	34.1	24.6
Apto. Rafael Barvo (Corozal)	0.0	34.0	24.0
Apto. BARRANCABERMEJA	0.0	32.8	23.5
Apto. BUCARAMANGA (Lebrija)	0.0	26.9	17.8
BUCARAMANGA (sede IDEAM)	0.0	X	X
Apto. CUCUTA	0.0	35.0	26.9
Apto. APARTADO	11.2	31.8	24.3
Apto. EL LIN	0.0	28.6	18.0

Figura 4-7. Tabla del IDEAM de la precipitación y temperatura para el 20 de julio de 2016  
Fuente: (IDEAM - Instituto de Hidrología, Meteorología y Estudios Ambientales, 2016)

#### 4.6 Análisis económico

La Tabla 4-1 es una lista general de los materiales que conforman el prototipo de estación meteorológica realizada y el costo que éstos representan. En el mercado se encontraron estaciones inalámbricas robustas de características similares. Con base a los precios consultados para diferentes estaciones comerciales se estimó que adquirir una de las mismas representa un costo mínimo de aproximadamente 600000 COP, una cantidad significativamente mayor a la descrita en

la Tabla 4-1. Un ejemplo de una estación meteorológica comercial es la *Estación Meteorológica Inalámbrica Davis Instruments 6250*, cuyo precio oscila alrededor de 1400000 COP.

Tabla 4-1  
Costo de materiales.

<b>Costo de materiales</b>	
<b>Rubro</b>	<b>Precio (COP)</b>
Tarjeta CY8CKIT-042	50000
Tarjeta M95	65000
Sensor BMP085	45000
Sensor DHT11	9000
Antena omnidireccional GSM Quad Band	5000
Sensores magnéticos Reed Switch	12000
Regulador MIC29302	7000
Resistor variable	1400
GLCD 128x64	40000
PIC16F877A	9000
Otros	50000
<b>Total</b>	<b>293400</b>

*Nota: dentro de **Otros** se incluyen diferentes componentes pasivos y activos de reducido costo. Así como cables, placas fenólicas, entre otros componentes y herramientas necesarias para la construcción del prototipo.*

## Capítulo 5

### Conclusiones

- Se realizó satisfactoriamente un prototipo de estación meteorológica que puede sensar la temperatura ambiente, la humedad relativa y la presión atmosférica (la medición de la velocidad y dirección del viento no se pudo comparar con un instrumento calibrado y por tanto no se garantiza la fiabilidad de los datos para estas variables). Que ofrece tres opciones diferentes para exportar los datos procesados de las mediciones realizadas a las variables físicas antes mencionadas, éstas son:
  - Envío de datos a través de comunicación serial basada en el protocolo RS232.
  - Envío de datos a través de GPRS.
  - Envío de datos a través de SMS.En el equipo terminal (móvil u ordenador) se visualizan en tiempo real. Para ello, se logró implementar una interfaz gráfica hecha en lenguaje C# para el ordenador y una aplicación Android para el móvil, que crean una hoja de cálculo de Excel en la se registran los datos recibidos de la estación facilitando un análisis futuro.
- El prototipo de estación meteorológica realizado representa un costo considerablemente menor respecto a una estación meteorológica comercial. Además, es un primer paso para el diseño y realización de una estación meteorológica local que cumpla con los estándares comerciales.
- El uso de sistemas embebidos PSoC facilitó el diseño y desarrollo del prototipo de estación meteorológica. Además, permite la mejora del mismo en el futuro.

## Capítulo 6

### Recomendaciones

#### 6.1 Técnicas

- El envío de mensajes tiene cargos, por lo que se recomienda que la tarjeta SIM utilizada en la estación tenga habilitados datos móviles.
- El periodo de tiempo para el envío de mensajes puede ser relativamente largo debido a que la tasa de variación de las variables físicas medidas con respecto al tiempo es pequeña. Evitándose así el gasto excesivo e innecesario de datos móviles.
- El área donde se encuentre operando la estación debe tener cobertura mientras se haga uso del envío de datos inalámbricamente.

#### 6.2 Para trabajos futuros

- Para posteriores mejoras se puede aprovechar el hardware disponible de la tarjeta para la ampliación del prototipo. Se podría, por ejemplo, incorporar un panel solar para medir el índice de radiación, y mediante éste, realizar una fuente de alimentación autónoma que energice a toda la estación.
- Realizar la calibración del anemómetro y veleta o reemplazarlos por modelos comerciales más robustos previamente calibrados.
- Por cuestiones de seguridad se debería realizar encriptación de los datos que se envían por Internet.
- Se debe tener cuidado en la alimentación del módulo de comunicación debido a que genera altos picos de corriente, lo que puede bajar la tensión en la etapa de alimentación y si la tensión se sale del rango permitido el módulo se apagará.

## Glosario

**Altitud:** distancia vertical entre un objeto o punto determinado en el espacio y la superficie del nivel del mar, la terrestre u otro punto tomado como referencia

**Anemómetro:** instrumento que mide la velocidad del viento.

**C#:** es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET, que después fue aprobado como un estándar por la ECMA (ECMA-334) e ISO (ISO/IEC 23270).

**Clima:** es el conjunto fluctuante de las condiciones atmosféricas, caracterizado por los estados y evoluciones del estado del tiempo, durante un periodo de tiempo y un lugar o región dados, y controlado por los denominados factores forzantes, factores determinantes y por la interacción entre los diferentes componentes del denominado sistema climático (atmósfera, hidrosfera, litosfera, criósfera, biosfera y antropósfera).

**DHCP (Dynamic Host Configuration Protocol):** es un protocolo cliente/servidor que proporciona automáticamente a un host de protocolo Internet (IP) una dirección IP y otra información de configuración relacionada como, por ejemplo, la puerta de enlace predeterminada y la máscara de subred.

**DNS (Domain Name System):** es un sistema de nomenclatura jerárquico descentralizado para dispositivos conectados a redes IP como Internet o una red privada.

**Humedad relativa:** relación entre la cantidad de vapor de agua que tiene una masa de aire y la máxima que podría tener.



**IDE (Integrated Development Environment):** es una aplicación informática que proporciona servicios integrales para facilitarle al programador el desarrollo de software.

**NAT (Network Address Translation):** es un mecanismo utilizado por *routers* para intercambiar paquetes entre dos redes que asignan mutuamente direcciones incompatibles.

**Presión atmosférica:** presión que ejerce la atmósfera sobre la superficie de la Tierra.

**TCP (Transmission Control Protocol):** protocolo de transmisión de Internet perteneciente a la capa de transporte del modelo TCP/IP.

**Temperatura ambiente:** temperatura ordinaria en torno a un cuerpo.

**TQFP:** es un microcontrolador que integra bloques analógicos y digitales y periféricos e interfaces de comunicación estándar configurables mediante software, de la familia PSoC 4200 manufacturada por *Cypress Semiconductor*.

**UDB (Universal Digital Block):** es un bloque digital flexible programable dentro de un dispositivo PSoC que está diseñado para realizar máquinas de estado síncronas.

**Veleta:** instrumento que mide la dirección del viento.

## Referencias

Alcaldía de Sincelejo - Sucre. (24 de 9 de 2010). Obtenido de Alcaldía de Sincelejo - Sucre:

[http://www.sincelejo-sucre.gov.co/informacion\\_general.shtml](http://www.sincelejo-sucre.gov.co/informacion_general.shtml)

Arango, D., Rincón, F., Manrique, L., & Buitrago, D. (2010). Estación Meteorológica Inalámbrica.

*Umbral Científico*, 66-72.

Bosch Sensortec. (1 de 7 de 2008). BMP085 Digital Pressure Sensor Data Sheet. Alemania.

CCM. (s.f.). Obtenido de CCM: <http://es.ccm.net/contents/681-estandar-gsm-sistema-global-de-comunicaciones-moviles>

Cypress Semiconductor Corporation. (9 de 6 de 2016). Obtenido de Cypress Semiconductor:

<http://www.cypress.com/file/138656/download>

Cypress Semiconductor Corporation. (s.f.). Obtenido de Cypress Semiconductor:

<http://www.cypress.com/file/46056/download>

D-Robotics UK. (30 de 7 de 2010). *DHT11 Humidity & Temperature Sensor*.

Fundación Tecnológica Antonio de Arévalo [TECNAR]. (3 de 2 de 2015). Obtenido de Fundación

Tecnológica Antonio de Arévalo TECNAR de Cartagena:

<http://orientacion.universia.net.co/universidades/fundacion-tecnologica-antonio-de-arevalo---tecnar-40/noticias/nuevo-reconocimiento-para-tecnar--redcolsi-2840.html>

IDEAM - Instituto de Hidrología, Meteorología y Estudios Ambientales. (08 de 2016). Obtenido

de <http://www.ideam.gov.co/>

Kurose, J., & Ross, K. (2010). *Redes de Computadoras: un enfoque desdoblado* (5 ed.). Madrid, España: Pearson Educación S. A.

Palacios, E., Remiro, F., & López, L. (2004). *Microcontrolador PIC16F84 Desarrollo de Proyectos* (1 ed.). México: Algaomega Grupo Editor.

Quectel. (26 de 11 de 2014). *M95 Hardware Design*.

Quectel. (4 de 11 de 2015). *GSM TCP/IP Application Note*.

Stallings, W. (2000). *Comunicaciones y Redes de Computadoras*. Madrid: PEARSON EDUCACIÓN S. A.