

# IMPLEMENTACIÓN EN FPGA DE SENSOR DE TEMPERATURA Y HUMEDAD RELATIVA PARA UNA FUTURA PLATAFORMA DE MONITOREO AMBIENTAL

***Julio César Regalado Sánchez***

Tecnológico Nacional de México/Instituto Tecnológico de Celaya  
*11030432@itcelaya.edu.mx*

***Alejandro Espinosa Calderón***

Tecnológico Nacional de México/Instituto Tecnológico de Celaya  
*alejandro.espinosa@itcelaya.edu.mx*

***José Alfredo Padilla Medina***

Tecnológico Nacional de México/Instituto Tecnológico de Celaya  
*alfredo.padilla@itcelaya.edu.mx*

***José Javier Díaz Carmona***

Tecnológico Nacional de México/Instituto Tecnológico de Celaya  
*javier.diaz@itcelaya.edu.mx*

## **Resumen**

Este trabajo presenta los avances en el diseño y desarrollo de lo que será un prototipo de una plataforma de integración de sensores de variables ambientales, basada en FPGA. Se eligió el utilizar FPGAs debido a que éstos destacan por su capacidad de procesamiento paralelo, la portabilidad de su código, y su alta velocidad de trabajo. La plataforma que se describe en este trabajo cuenta con la capacidad de conectar múltiples sensores para mediciones ambientales como son temperatura, humedad y luminosidad. Todos estos sensores utilizan diferentes protocolos digitales de comunicación, por ejemplo: RS232, I2C y SMBus, entre otros. Por el momento se reportan avances específicamente en la implementación del sensor de temperatura y humedad relativa SHT75. Este sensor utiliza un protocolo basado en I2C, pero con algunas variantes. Este artículo reporta la

implementación y ajuste del protocolo de medición, así como algunas mediciones logradas hasta el momento.

Esta plataforma está pensada para ser reconfigurable en hardware, realizar procesamiento paralelo y permitir la integración de nuevos sensores. Parte del trabajo a futuro consistirá en describir e implementar en el FPGA los protocolos de comunicación y procesamiento digital de señales correspondiente a esos nuevos sensores.

**Palabras Clave:** I2C, Medición ambiental, RS232, VHDL.

## 1. Introducción

En la actualidad existen diferentes sistemas de medición de variables ambientales mediante plataformas de integración. Algunas de estas plataformas hacen uso de la tecnología de módulos ZigBee que se utilizan como router en puntos distantes, formando con estos dispositivos una red de sensores que permiten supervisar algunas variables físicas [1]. Algunas otras plataformas utilizan técnicas de control basadas en sistemas de conjuntos difusos las cuales además utilizan una red de internet para poder monitorear los sensores mediante la nube (*Cloud*) a través de una red alámbrica [2]. Existen otras plataformas que hacen uso de los microprocesadores para realizar el monitoreo de temperatura mediante sensores analógicos para facilitar la adquisición de datos mediante las entradas analógicas de los microcontroladores [3]. Algunos otros proyectos desarrollados actualmente combinan la tecnología de módulos ZigBee con módulos de radio frecuencia XBee-PRO que convierten la red alámbrica ZigBee en inalámbrica lo cual contribuye a una mejora a la red [4]. También se han desarrollado tarjetas de adquisición de datos basadas en microcontroladores y software libre, para obtener una tarjeta de bajo costo [5]. Además, se han desarrollado plataformas, basadas en el uso de software y hardware libres, que consideran la construcción y adaptación de sensores para medir las variables climatológicas dentro y fuera invernaderos. Estas plataformas utilizan la construcción y adaptación de interfaces electrónicas para capturar los valores de

los sensores y el desarrollo de software para la interpretación de los datos mediante plataformas de software y hardware libres como Java y Arduino [6]. Las plataformas antes mencionadas tienen la desventaja de que trabajan de forma serial, lo que ocasiona retrasos en el procesamiento. Los FPGA (acrónimo de *Field Programmable Gate Array*) son dispositivos digitales que permiten realizar procesamientos en paralelo, resultando en mayor velocidad de trabajo. Por este motivo se pretende crear una plataforma basada en FPGA, que nos permita conectar diferentes dispositivos y realizar mediciones de manera simultánea, además de tener la posibilidad de expandir la cantidad de sensores integrados, al tener diferentes bloques de programación, con protocolos de comunicación establecidos como I2C, RS232 y protocolos propios de diferentes fabricantes [7]. Al tener los bloques de programación con diferentes protocolos de comunicación, bastará con agregar un nuevo bloque a la estructura y habilitar las entradas y salidas de la FPGA para poder integrar un nuevo dispositivo de medición sin intervenir con los demás bloques de programación de los demás dispositivos [8]. El primer objetivo para lograr esta plataforma es la implementación en FPGA del protocolo RS232, I2C y el protocolo del sensor SHT75, que sentarán las bases para la plataforma de monitoreo ambiental.

## 2. Desarrollo

Para la implementación de la plataforma de integración de sensores, se utilizó una tarjeta Basys 2, la cual cuenta con un reloj interno de 50 MHz, conector VGA, puerto PS/2, 8 led's, 4 display's, 4 botones, 8 switches y 16 pines de I/O. en la figura 1 se muestra la FPGA mencionada [9].

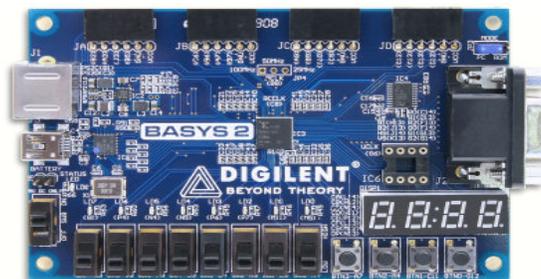


Figura 1 Tarjeta de desarrollo Basys 2.

Para poder realizar las mediciones con los diferentes sensores, se utilizaron diferentes protocolos de comunicación, (RS232 e I2C). Se empezó con el desarrollo de los drivers de estos protocolos de comunicación, figura 2.

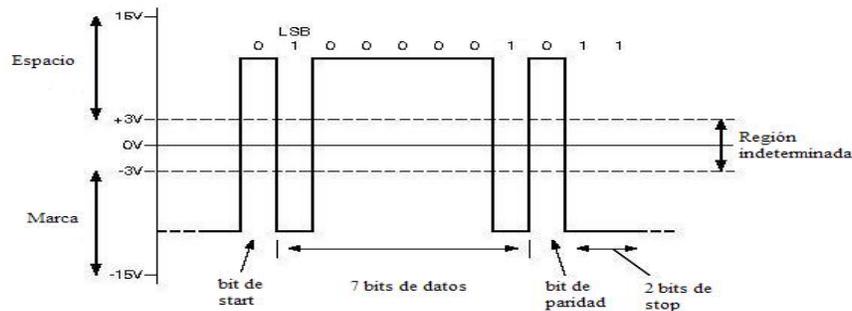


Figura 2 Trama de comunicación del bus RS232.

### RS232

El primer paso fue describir en código VHDL el diagrama de bloques del RS232 de la figura 3. Para la visualización se usó un osciloscopio marca HAMEG modelo HMO 1002, el cual cuenta con siete entradas digitales para realizar pruebas de funcionamiento de protocolos de comunicación como son el RS232, I2C, CAN, LIN, SPI y SSPI. Se conectó la FPGA al probador de protocolos para comprobar su funcionamiento y verificar la correcta transmisión de datos [10].

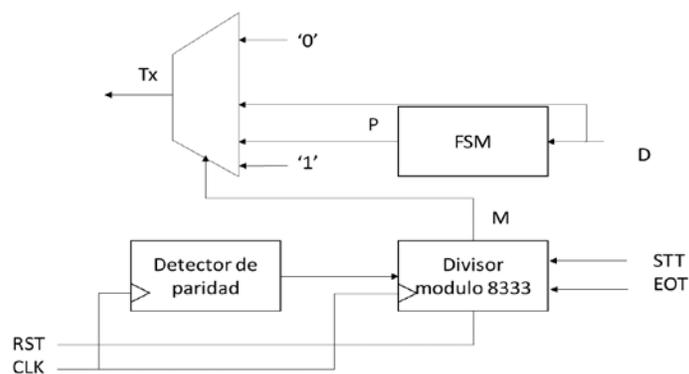


Figura 3 Diagrama de bloques para FPGA del transmisor RS232.

RS232 es un protocolo de comunicación serial asíncrono, que tiene la característica de ser punto a punto. Este protocolo de comunicación no permite la

creación de redes [6]. Las especificaciones eléctricas del puerto serial están contenidas en el estándar RS232 de EIA (Electronics Industry Association) que establece los siguientes parámetros:

- Un “Espacio” (0 lógico) está entre +3 y +15 volts.
- Una “Marca” (1 lógico) estará entre -3 y -15 volts.
- La región entre +3 y -3 volts es indefinida.
- Un circuito abierto nunca debe exceder 25 volts (con referencia a tierra).

La interfaz está limitada normalmente a velocidad de 20 kbps para distancias del orden de 15 metros. Para el diseño de la plataforma propuesta se eligió una velocidad de 1.2 kbps. En la figura 2 se muestra la trama de comunicación que utiliza el protocolo RS232 donde se observa la secuencia de inicio y de fin, además del bit de paridad para la comprobación del correcto envío de datos. En la figura 3 se observa el diagrama de bloques para el protocolo RS232, donde se utiliza un divisor de frecuencia para generar la velocidad de transmisión de 1.2 kbps, una máquina de estados donde se encuentra el control del bloque principal, el detector de paridad y el multiplexor donde se introduce la información y la entrega por el bus Tx [11].

## **I2C**

Se utilizó la estructura del diagrama de bloques del RS232 para diseñar el protocolo I2C, modificando la máquina de estados y el divisor de frecuencia para obtener además el bus de sincronización del protocolo I2C. Una vez que se desarrolló el código en VHDL para el protocolo I2C, se realizaron pruebas con el probador de protocolos del osciloscopio para verificar la correcta transmisión de datos y el buen funcionamiento del protocolo.

El bus I2C es un estándar que facilita la comunicación entre microcontroladores, memorias y otros dispositivos con cierto nivel de "inteligencia", sólo requiere de dos líneas de señal y un común o masa. Fue diseñado a este efecto por Philips y permite el intercambio de información entre muchos dispositivos a una velocidad aceptable, de unos 100 Kbps, aunque hay casos especiales en los que el reloj

llega hasta los 3,4 MHz. Para el diseño del protocolo se eligió una velocidad de 100Kbps [6]. La metodología de comunicación de datos del bus I2C es en serie y sincrónica. Una de las señales del bus marca el tiempo (pulsos de reloj) y la otra se utiliza para intercambiar datos [7]. Descripción de las señales:

- SCL (System Clock) es la línea de los pulsos de reloj que sincronizan el sistema.
- SDA (System Data) es la línea por la que se mueven los datos entre los dispositivos.
- GND (Masa) común de la interconexión entre todos los dispositivos "enganchados" al bus.

En la comunicación I2C existen dos modalidades de comunicación. La primera modalidad es el modo maestro la cual se encarga de establecer la comunicación con los demás dispositivos conectados al sistema mediante la generación de estados de inicio y parada, además genera el tren de pulsos para la sincronización. La segunda es en modo esclavo, en este modo sólo se recibe el tren de pulsos para la sincronización y no puede generar inicio de comunicación con otro dispositivo, sólo puede responder a las peticiones de lectura o escritura del maestro [12].

En la figura 4 se observa la trama de comunicación del bus I2C, el modelo que se propone es el modo maestro para tener control sobre los dispositivos que se tienen planeado conectar en la FPGA.

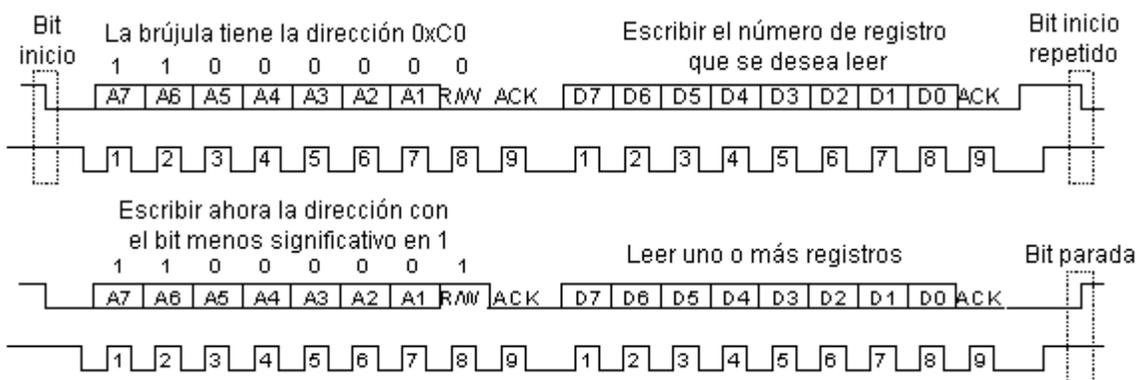


Figura 4 Trama de comunicación del bus I2C.

## Sensor de temperatura y humedad SHT75

El primer paso fue diseñar el bloque de programación para la secuencia de inicio de comunicación y reinicio de comunicación. Luego se programó la máquina de estados, el módulo del divisor de frecuencia y el multiplexor para la salida y entrada de datos de la FPGA. Se utilizó el osciloscopio para realizar pruebas con el protocolo.

Se agregó un nuevo bloque de programación para poder utilizar el display de la FPGA. Después se procedió a conectar el sensor SHT75 en la FPGA y se realizaron diferentes mediciones dentro de un refrigerador y en una habitación cerrada con un calentador encendido, además se utilizó el display de la FPGA para poder mostrar los datos obtenidos por el sensor SHT75.

Mediante la ecuación 1 se calculó la temperatura, siendo  $SO_t$  la medición obtenida del sensor de temperatura. Con las ecuaciones 2 y 3 se calcularon la humedad lineal (sin compensación de temperatura) y la humedad compensada, siendo  $SO_{rh}$  la medición que entrega el sensor de humedad y  $T$  la temperatura calculada [13].

$$T = -39.7 + 0.01 * SO_t \quad (1)$$

$$RH(\text{lineal}) = -2.0468 + (0.0367 * SO_{rh}) + (-1.5955E - 6) * SO_{rh}^2 \quad (2)$$

$$RH(\text{compensada}) = (T - 25) + (0.01 + 0.00008 * SO_{rh}) + RH(\text{lineal}) \quad (3)$$

El SHT75 es un sensor de temperatura y humedad de la marca SENSIRION, utiliza la comunicación serial para el envío de datos obtenidos de las diferentes variables físicas que mide. El sensor cuenta con un rango de operación de -40 °C a 123.8 °C, una resolución típica de 0.01 °C y un tiempo de espera de medición de 320 ms para el caso del sensor de temperatura. En cuanto a la humedad, cuenta con una resolución de 0.05%RH, un rango de operación de 0% a 100% y un tiempo de espera de medición de 80 ms. Se alimenta con un voltaje típico de 3.3 V, aunque admite de 2.4 V a 5.5 V [13].

El protocolo de comunicación que utiliza el sensor SHT75 es similar al I2C, sin embargo, hay varios cambios en la trama de comunicación como es la secuencia de inicio que se observa en la figura 5, la secuencia de reinicio de comunicación

que se observa en la figura 6 y el tiempo de espera de medición de 80 ms, tiempo que utiliza el sensor para realizar la medición de la variable ambiental y preparar la secuencia de transmisión de información [13].

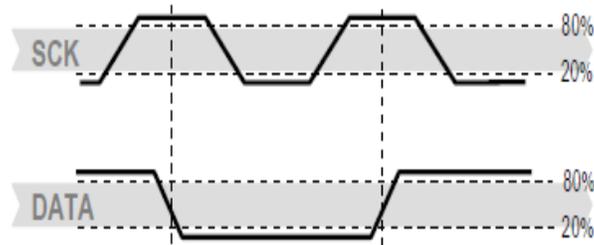


Figura 5 Secuencia de inicio de comunicación.

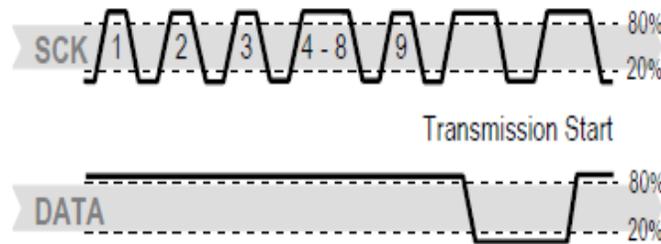


Figura 6 Secuencia de reinicio de comunicación.

En la figura 7 se observa el ejemplo de comunicación donde se mide la humedad, en este ejemplo el sensor devuelve un dato de 16 bits donde sólo los 12 bits menos significativos contienen la información de la medición realizada.

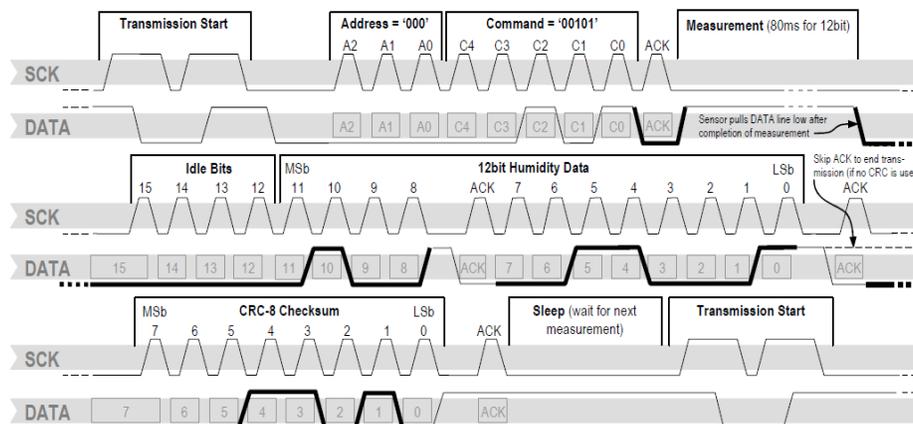


Figura 7 Ejemplo de comunicación para lectura de humedad.

### 3. Resultados

#### Protocolo RS232

Con la ayuda de osciloscopio HAMEG HMO 1002 series, el cual cuenta con entradas digitales para realizar pruebas de funcionamiento de protocolos de comunicación, se probó el protocolo realizado en la FPGA enviando el dato “0001 0001” al osciloscopio para verificar el correcto funcionamiento del protocolo RS232. En la figura 8 está la captura del osciloscopio donde se observa que el dato enviado fue leído correctamente. Después, se realizó una nueva prueba cambiando el dato “0001 0001” por “1010 1010”. La figura 9 muestra la captura del osciloscopio donde se ve la tabla de datos leídos por el probador digital.

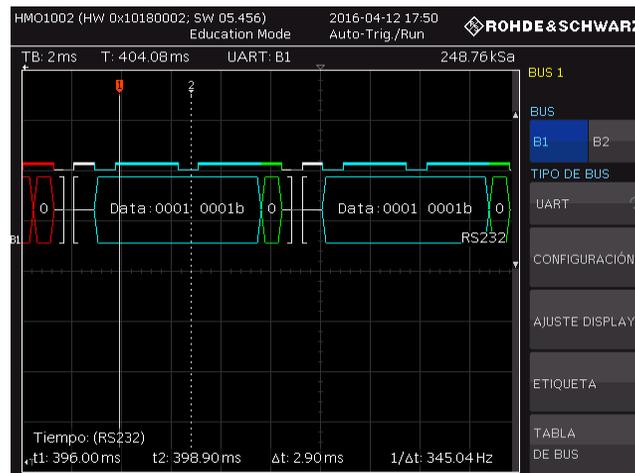


Figura 8 Señal obtenida de la FPGA por el probador digital de protocolos.

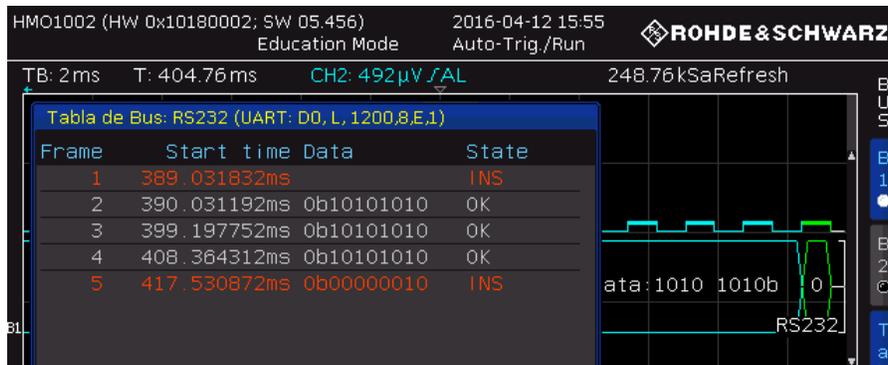


Figura 9 Tabla de datos leídos por el probador digital, para el protocolo RS232.

## Protocolo I2C

En la figura 10 se presenta el diagrama final realizado para el protocolo I2C, el cual se diseñó para ser el dispositivo maestro debido a que los sensores que se van integrar a la plataforma son esclavos.

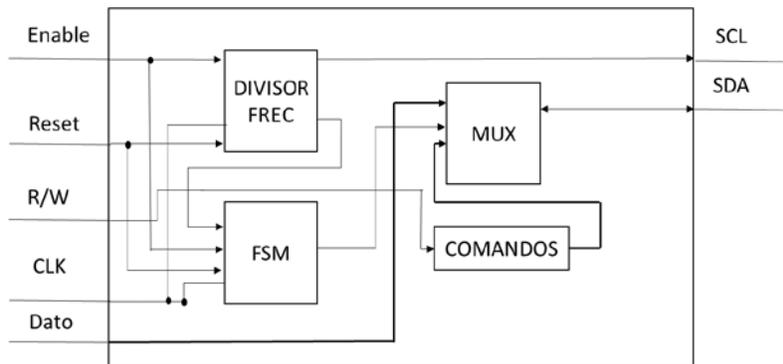


Figura 10 Diagrama de bloques final del bus I2C para FPGA.

Primero se probó el modo escritura en el cual se envió el dato binario “1010 1010”, que corresponde al valor hexadecimal “AA”, al dispositivo con la dirección “0000 1111”, que corresponde a un valor hexadecimal “0F”. En la figura 11 se observa la información recibida por el probador digital de protocolos. Para comprobar el modo de lectura se utilizó la misma dirección “0F” de dispositivo y se leyó el dato “C4” en hexadecimal del osciloscopio, en la figura 12 se observa la señal obtenida de la FPGA y el dato enviado por el osciloscopio.

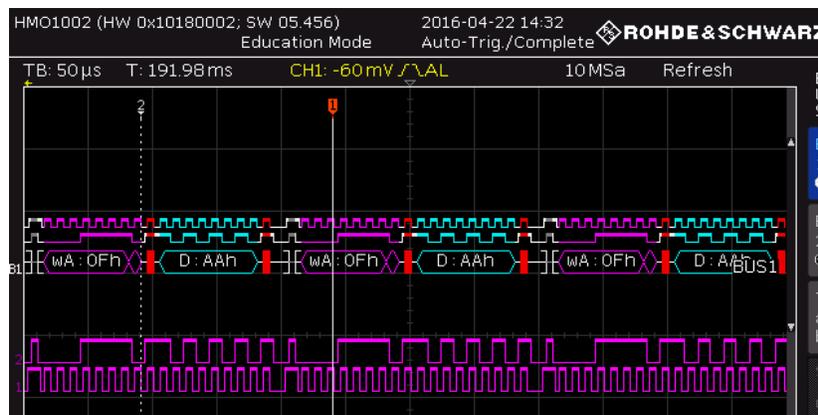


Figura 11 Señal obtenida de la FPGA en modo escritura, probador digital de protocolos.

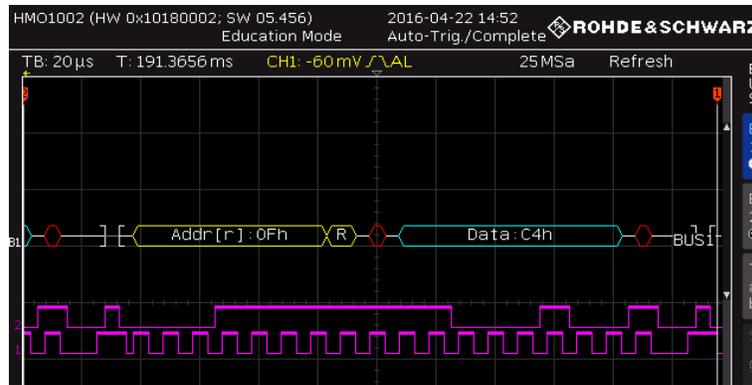


Figura 12 señal obtenida de la FPGA y el dato enviado por el osciloscopio.

### Protocolo Sensor SHT75

Antes de realizar la comunicación serial con el se comprobó que la señal enviada por la FPGA fuera correcta, para esto se utilizó el probador digital de protocolos. La figura 13 muestra la, las tramas de inicio y reinicio de comunicación concuerdan con las señales SCK y data, de la figura 7.



Figura 13 Señal obtenida de la FPGA, sensor SHT75.

Cabe mencionar que solamente se utilizó el 17% de los recursos de la FPGA Basys 2 que es una tarjeta de gama baja.

Las mediciones de prueba del sensor se realizaron dentro de un refrigerador (figura 14), donde se tomaron mediciones de temperatura (figura 15) y humedad (figura 16):

- 1ª sección se abrió y cerró la puerta del refrigerador conectado.
- 2ª sección se desconectó el refrigerador para aumentar temperatura.

- 3ª sección se abrió y se volvió a cerrar la puerta del refrigerador desconectado. 4ta sección se dejó abierta la puerta hasta estabilización a temperatura

El error promedio de temperatura fue de 18.74%. El error promedio de humedad lineal fue de 20.04% mientras que el error promedio en humedad compensada fue de 16.81%.



Figura 14 Prueba de medición realizada en un refrigerador.

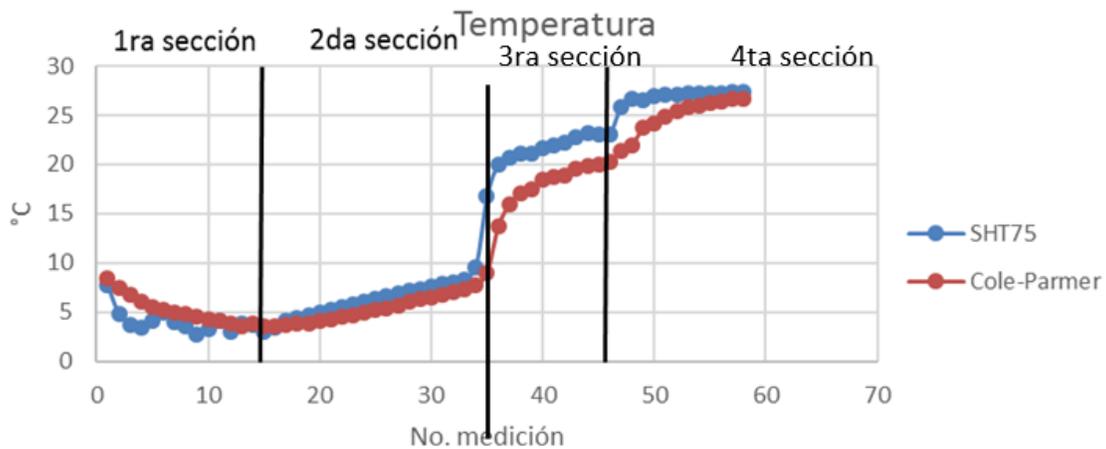


Figura 15 Variación de temperatura en refrigerador. ambiente.

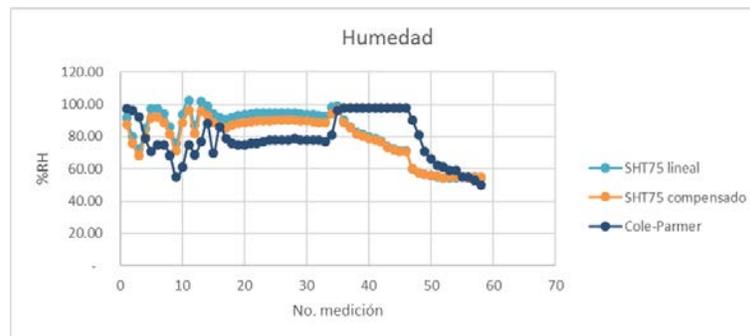


Figura 16 Variación de humedad en refrigerador.

También se realizaron pruebas con el sensor SHT75 en una habitación con temperaturas más elevadas (figuras 17 y 18). El error promedio de temperatura fue de 2.35%. El error promedio de humedad lineal fue de 36.36% mientras que el error promedio en humedad compensada fue de 39.7%.

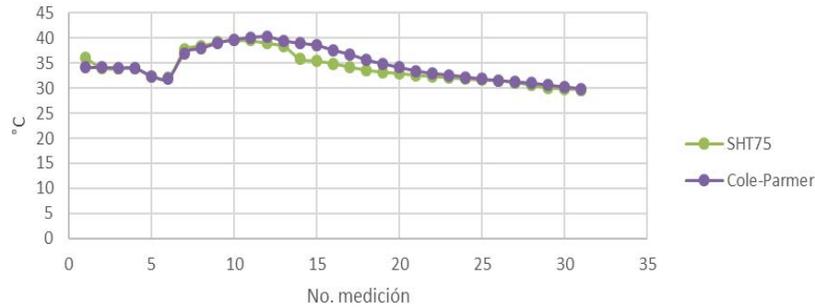


Figura 17 Variación de temperatura en habitación, sensor SHT75.

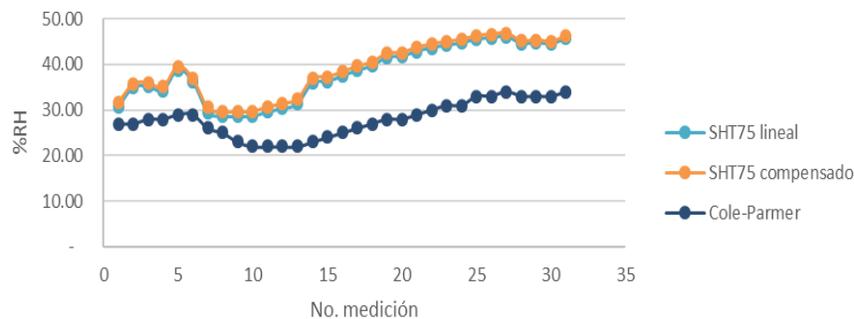


Figura 18 Variación de humedad en habitación, sensor SHT75.

#### 4. Discusión

A través de las figuras 8 y 9 se comprueba que la descripción VHDL generada para el protocolo RS232 funcionó correctamente. Esto fue un primer paso necesario para, a partir de aquí, modificar la estructura de este módulo y desarrollar el de I2C.

Las figuras 11 y 12 confirman que la descripción VHDL del módulo I2C funciona correctamente, logrando la comunicación con el osciloscopio y el probador digital en modos de lectura y escritura. Una vez asegurada la correcta comunicación I2C, se modificó este módulo para adecuar la comunicación con el sensor SHT75.

Dado que el SHT75 tiene un protocolo de comunicación cercano al I2C, pero con algunas variantes, se logró generar un módulo VHDL específico para su comunicación. La figura 7 muestra las señales específicas que requiere este sensor para inicio y reinicio de la comunicación (ver señales SCK y Data). La figura 13 muestra cómo es que efectivamente se logró la generación de estas señales, asegurando la comunicación con el sensor.

Una vez que se probó dicha secuencia se procedió a conectar el sensor en la FPGA y realizar mediciones dentro de un refrigerador, como se muestra en la figura 14, además se utilizó un termómetro Cole-Parmer, para comparar las mediciones obtenidas por el sensor SHT75. En la figura 15 se observan los datos de temperatura obtenidos en el refrigerador. Estas mediciones se realizaron cada dos minutos. Al realizar dicha prueba se observó que el termómetro comercial tardaba más tiempo en detectar dichos cambios de temperatura mientras que el sistema con el sensor SHT75 mostró ser más rápido al detectar los cambios de temperatura. Esto se comprueba observando los datos obtenidos en la 1ª sección de la figura 15, donde el refrigerador encendía y apagaba el sistema de enfriamiento; aquí el sensor detectó los cambios de temperatura producidos mientras que el termómetro comercial no detectó ninguno de estos cambios. En la 2ª sección de la figura 15 se apagó el refrigerador para hacer que éste aumentara abruptamente su temperatura y observar el comportamiento del sistema de medición. El refrigerador permaneció desconectado para las siguientes secciones. En estas mediciones se observó un comportamiento lineal por parte de ambos sistemas de medición. En la 3ª sección se abrió y cerró la puerta del refrigerador donde nuevamente el sensor SHT75 detectó de manera más rápida los cambios de temperatura. En la 4ª sección se abrió nuevamente la puerta y se dejó estabilizar la temperatura del refrigerador a la temperatura ambiente donde los dos sensores se igualaron.

En la figura 17 se observan los datos obtenidos en una nueva prueba que se realizó en una habitación cerrada donde se utilizó un calentador para elevar la temperatura y realizar diferentes mediciones con el sistema, en temperaturas más elevadas. En esta gráfica se observa cómo al ser más lento el incremento de

temperatura, ambos sensores presentan mediciones muy similares a diferencia de la prueba del refrigerador donde los cambios de temperatura fueron más rápidos. Las figuras 16 y 18 muestran las mediciones de humedad obtenidas por el sensor y el instrumento de medición comercial en las pruebas realizadas en el refrigerador y la habitación cerrada, respectivamente. En este caso se calcularon dos valores de humedad uno con compensación de temperatura y otro sin la compensación de temperatura, siendo ambos similares. La medición sin compensar arroja valores por arriba del 100% de humedad, es por esta razón que el fabricante recomienda usar la compensación de temperatura para valores de temperatura diferentes de 25 °C.

Cabe resaltar que la medición de humedad se realizó simultáneamente con la de temperatura, esto para poder calcular la humedad con la compensación de temperatura. El error promedio obtenido es muy elevado, esto se puede deber a:

- La lenta respuesta del sensor de referencia.
- A falta de calibración del sistema de referencia.
- O a un desajuste en el algoritmo de medición del SHT75.

Sin embargo, las figuras 15, 16, 17, y 18 muestran que ambos sistemas siguen formas semejantes (hablando de humedad) y convergen en los mismos valores a largo plazo (hablando de temperatura), esto hace suponer que las mediciones no están del todo mal. En un futuro se tendrá que comparar con otros sistemas de medición para hacer los ajustes pertinentes.

## **5. Conclusiones**

Se logró comprobar satisfactoriamente el envío de datos del protocolo RS232 mediante el probador de protocolos, lo cual sienta las bases para poder conectar los sensores que cuenten con este método de comunicación.

Se comprobó el bus I2C en modo maestro y se verificó que escribiera y leyera datos correctamente mediante el probador digital. Se utilizó un “arduino uno” para comprobar que realizara la comunicación correctamente con una plataforma

comercial con el protocolo I2C. Esta prueba resultó satisfactoria debido a que se realizó correctamente el envío de datos.

Se realizó correctamente el protocolo de comunicación propio del sensor SHT75 y además se obtuvieron diferentes mediciones de temperatura y se comprobaron con un termómetro comercial. El sensor SHT75 demostró poder medir cambios de temperatura más rápido que el comercial. De manera semejante, se realizaron pruebas de humedad donde el SHT75 mostró seguir tendencias similares al instrumento comercial. En el trabajo a futuro se planea comparar con otros instrumentos de medición y ajustar el error del sistema desarrollado.

En el trabajo futuro también se planea describir e implementar en el FPGA protocolos de comunicación SMBus y frecuencímetros para interconectar con sensores de luminosidad. Una vez implementada la comunicación y el procesamiento digital se caracterizará el sistema de iluminación. De esta forma se integrará en FPGA una plataforma de medición ambiental. Otro punto que se deberá lograr en un futuro es la transmisión de datos hacia una computadora, para su almacenamiento, procesamiento, y monitoreo.

## **6. Bibliografía y Referencias**

- [1] C. Vera, J. Barbosa, D. Pabón, “Acople de sensores en la medición de variables ambientales usando tecnología ZigBee”. *Scientia et Technica* Año XIX. Vol. 19. No. 4. 2014. Pp 419-423.
- [2] G. Ampuño, M. Gómez, W. Agila, “Monitoreo y control distribuido de micro-unidades de regulación de humedad y temperatura para invernaderos a través de *Clouds*”. *Tecnológica ESPOL – RTE*. Vol. 28. No. 5. 2015. Pp 258-272.
- [3] A. López, “Medición de temperatura mediante sensores inteligentes basados en microcontrolador”. *UNIMAR*. Vol. 32. No. 2. 2014. Pp 107-127.
- [4] C. Vera, J. Barbosa, D. Pabón, “Parámetros de configuración en módulos XBEE-PRO® S2B ZB para medición de variables ambientales”. *Tecnura*. Vol. 19. No. 45. 2015. Pp 141-157.

- [5] M. Gaspar, C. Gutiérrez, G. Bobadilla, J. Jiménez, P. Quinto, J. Barbosa. "Temperature acquisition board of low building and programing cost". DYNA. Vol. 91. No. 4. 2016. Pp 397-401.
- [6] O.L. Espinosa, G. A. Villavicencio Pérez, S. A. Díaz Luna, "Paquete tecnológico para el monitoreo ambiental en invernaderos con el uso de hardware y software libre". Terra Latinoamericana. Vol. 32. 2013. Pp 77-84.
- [7] P. Venkateswaran, A. Saynal, S. Das, S.K Saynal, R. Nandi, "Design and implementation of FPGA based interface model for scale-free network using I2C bus protocol on Quartus II 6.0." 4th International Conference on Computers and Devices for Communication, IEEE. 978-81-8465-152-2/09. 2009.
- [8] C. L. Herrera, M. Quiñones Cuenca, "Diseño e implementación de un servidor web embebido basado en una arquitectura reconfigurable con FPGAs para el control y monitoreo de periféricos".Escuela de Electrónica y Telecomunicaciones. 2012.
- [9] Basys 2 FPGA, Datasheet, Digilent. Revisión C. USA. 2016.
- [10] Osciloscopio HMO 1002, Datasheet, HAMEG Instruments GmbH. Germany. Version 01.00. 2014.
- [11] R. Romero Troncoso, Electrónica Digital y Lógica Programable. 2007. Ed. Universidad de Gto. México,
- [12] R. Manzano, M. Pérez, R. Ponguillo, "Diseño de un módulo de propiedad intelectual basado en FPGA para el manejo del bus I2C". Revista tecnológica ESPOL.
- [13] Humidity Sensor SHT75, Datasheet, SENSIRION AG, Switzerland. Version 5. 2011.

## **7. Autores**

Julio César Regalado Sánchez estudiante de 10° semestre de la carrera de Ingeniería Electrónica en el Instituto Tecnológico de Celaya, Sus áreas de interés incluyen: procesamiento de sistemas digitales en FPGA y procesamiento de imágenes digitales en el espectro visible.

Alejandro Espinosa Calderón recibió su título de Doctorado de la Universidad Autónoma de Querétaro, en Querétaro, Qro., Mex., en 2012, de donde obtuvo honores, la medalla al mérito académico, y el premio a la mejor tesis doctoral. Ha participado en eventos científicos internacionales en EEUA, Italia y China. Ha recibido premios internacionales de la OSA y USAID, ambos de EEUA. Desde 2013 trabaja en el Dpto. de Ing. Eléctrica y Electrónica del ITCelaya. Sus áreas de interés incluyen: Instrumentación electrónica y procesamiento en sistemas digitales FPGA/microprocesadores.

J. Alfredo Padilla Medina recibió su grado de Doctor en Filosofía del Centro de Investigaciones en Óptica, en León, Gto., Mex., en 2003. Actualmente es profesor-investigador de tiempo completo en el Dpto. de Ing. Eléctrica y Electrónica del ITCelaya. Pertenece al Sistema Nacional de Investigadores con nivel I. Sus áreas de interés incluyen: procesamiento de imágenes digitales en el visible e infrarrojo, el sistema visual humano, y la lógica difusa.

Javier Díaz Carmona recibió su grado de Doctor en Ciencias del Instituto Nacional de Astrofísica, Óptica y Electrónica, en Puebla, Pue., Mex., en 2003. Desde 1991 trabaja en el Dpto. de Ing. Eléctrica y Electrónica del ITCelaya. Ha participado en más de 40 artículos. Pertenece al Sistema Nacional de Investigadores con nivel I. Sus temas de interés incluyen: procesamiento digital de señales, filtros digitales, y aplicaciones de sistemas embebidos FPGA/DSP.

Los autores agradecen al Tecnológico Nacional de México por el apoyo económico para la realización de este proyecto.