

OBTENCIÓN DEL MÁXIMO ANCHO DE BANDA PARA LA ADQUISICIÓN Y RECONSTRUCCIÓN DE SEÑALES ANALÓGICAS CON LA TARJETA SPARTAN 3E

Enrique Gerardo Hernández Vega

Tecnológico Nacional de México/Instituto Tecnológico de Chihuahua
ehernand@itchihuahua.edu.mx

Jorge Alberto Ortiz Gallo

Tecnológico Nacional de México/Instituto Tecnológico de Chihuahua
jaortiz@itchihuahua.edu.mx

Daniel Eduardo Morales Fernández

Tecnológico Nacional de México/Instituto Tecnológico de Chihuahua
demorales@itchihuahua.edu.mx

Alejandro Verduzco Hernández

Tecnológico Nacional de México/Instituto Tecnológico de Chihuahua
averduzco@itchihuahua.edu.mx

Resumen

En este trabajo se presenta un sistema en el cual se obtiene el máximo ancho de banda posible para la adquisición y reconstrucción de señales analógicas en la tarjeta de desarrollo Spartan-3E del fabricante Xilinx, utilizando los convertidores que contiene dicha tarjeta, tanto el ADC como el DAC. El sistema está desarrollado en VHDL empleando el concepto de máquina de estados finitos (FSM) y el administrador digital de reloj (DCM) incluido en el FPGA de la tarjeta. En aplicaciones tales como procesamiento digital de señales en tiempo real, comunicaciones digitales y control digital, por mencionar algunas, es muy importante tener un ancho de banda considerable en el sistema. El valor máximo obtenido para el ancho de banda del sistema fue de 161 kHz.

Palabras Claves: ADC, ancho de banda, DAC, FPGA, Spartan-3E.

Abstract

This paper presents a system in which the maximum bandwidth possible for the acquisition and reconstruction of analog signals is obtained in the Spartan-3E development board of the Xilinx manufacturer, using the converters contained in this board, both the ADC and The DAC. The system is developed in VHDL using the concept of finite state machine (FSM) and the digital clock manager (DCM) included in the FPGA. In applications such as digital processing of real-time signals, digital communications and digital control, to mention a few, it is very important to have a considerable bandwidth in the system. The maximum value obtained for the system bandwidth was 161 kHz.

Keywords: ADC, bandwidth, DAC, FPGA, Spartan-3E.

1. Introducción

En los planes y programas de estudio de Ingeniería Electrónica, no sólo en los Institutos Tecnológicos del país sino en todos los otros subsistemas educativos, se considera el uso de VHDL y FPGAs para el desarrollo de sistemas digitales y sus aplicaciones. Es común encontrar en las diversas instituciones educativas tarjetas de desarrollo como la Spartan-3E de Xilinx o sus equivalentes.

En el caso de la tarjeta Spartan-3E, la comunicación entre el FPGA y los convertidores de datos ADC y DAC, es a través de una sola interfaz serial SPI lo cual limita el ancho de banda del sistema. Además, la salida digital de 14 bits del ADC es en complemento a 2, mientras que el DAC es de 12 bits, tal cual se señala en la guía de usuario de la tarjeta [Xilinx, 2011]. Por estas razones, son pocos los trabajos de procesamiento de señales con la tarjeta Spartan-3E que utilicen ambos convertidores, la mayoría ha utilizado convertidores externos, ya sea uno o ambos [Domínguez, 2011].

La guía de usuario de la tarjeta contiene información muy escueta y algunos errores conceptuales como en el caso de la función de transferencia del ADC, lo cual puede generar errores en el cálculo de los valores de salida digital y su linealización dado que está en complemento a 2, como en [Mascharak, 2012].

El cuello de botella en el sistema es la interfaz SPI compartida por ambos convertidores cuyo límite máximo es de 50 MHz según el fabricante de los convertidores, por lo que a mayor frecuencia de la señal de reloj de esta interfaz se tendrá una mayor razón de muestreo y en consecuencia un mayor ancho de banda. En el resto de los trabajos similares consultados, la frecuencia de la señal de reloj de la interfaz SPI es menor a 10 MHz, ejemplo [Khedr, 2013].

El mayor ancho de banda obtenido usando los convertidores de datos, ADC y DAC, en la tarjeta Spartan-3E para la adquisición y reconstrucción de señales analógicas, es de 141 kHz con una razón de muestreo de 282 kmuestras/segundo. La frecuencia de la señal de reloj para la interfaz SPI es de 41.67 MHz para el manejo del ADC y DAC [Silage, 2008]. Es posible aumentar el ancho de banda del sistema de adquisición y reconstrucción de señales mediante el uso de un sintetizador digital de frecuencia a través de los administradores digitales de reloj (DCM), para aumentar la frecuencia de trabajo del SPI. Además, es posible utilizar un protocolo SPI de 24 bits en lugar de 32 bits para la comunicación con el DAC como se establece en la hoja de datos [Linear Technology Corporation, 2004].

2. Métodos

El hardware que se utilizó para llevar a cabo esta implementación fue la tarjeta de desarrollo Spartan-3E con un FPGA XC3S500E. La tarjeta contiene, además, entre otros muchos recursos, un circuito de captura análoga y un circuito de conversión digital a análoga. El diagrama a bloques del sistema se muestra en la figura 1. El software utilizado fue el ambiente de desarrollo ISE 14.7 de Xilinx.

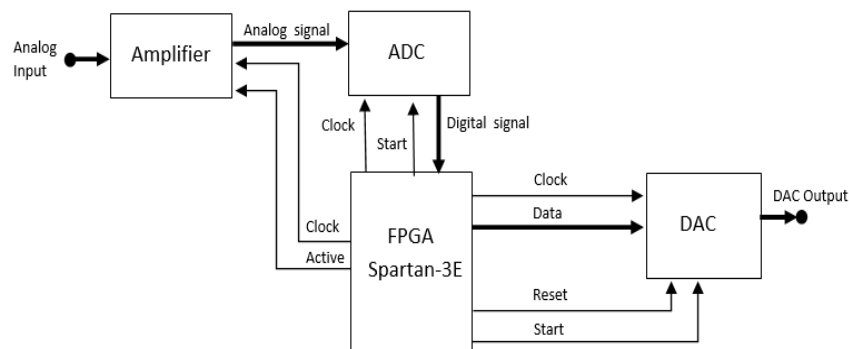


Figura 1 Diagrama a bloques del sistema.

El circuito de captura análoga consiste en un amplificador de ganancia programable (PGA LTC 6912) y de un convertidor análogo a digital (ADC LTC 1407A). Ambos son controlados por el FPGA a través de una interfaz SPI. El diagrama a bloques del circuito de captura análoga se muestra en la figura 2.

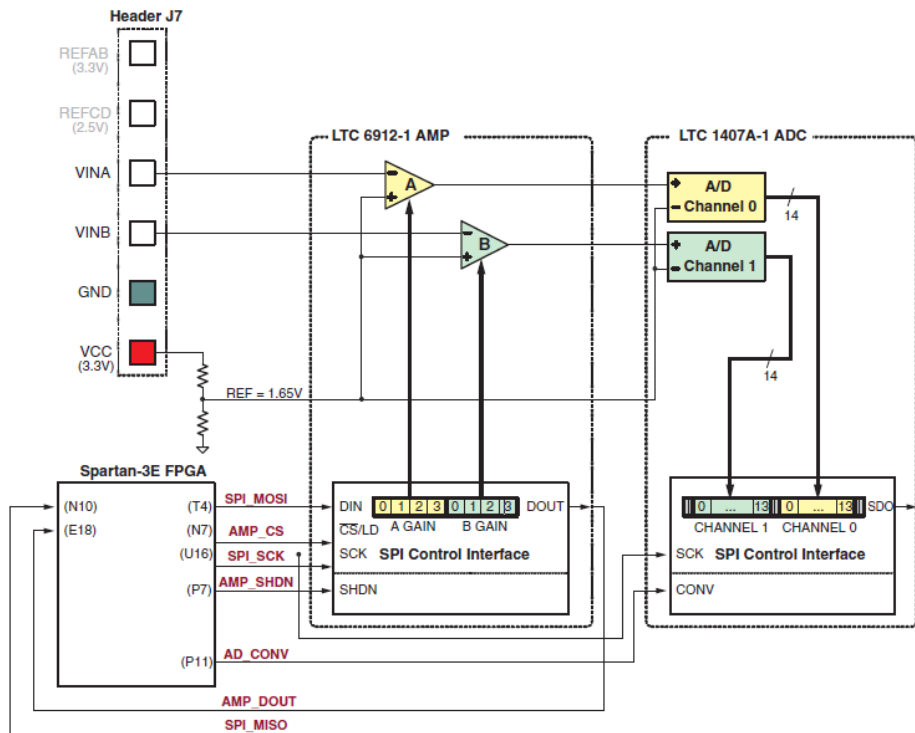


Figura 2 Vista detallada del circuito de captura análoga.

El circuito de captura análoga toma el voltaje de las entradas VINA o VINB y lo convierte a una representación digital de 14 bits, D[13:0], como se expresa en la ecuación 1.

$$D[13:0] = GANANCIA \times \frac{(V_{IN} - 1.65 V)}{1.25 V} \times 8192 \quad (1)$$

El voltaje de referencia del amplificador PGA y del ADC es de 1.65 V. El máximo rango del ADC es de ± 1.25 V, por lo que los límites de voltaje inferior y superior de la señal análoga serían 0.4 y 2.9 V respectivamente, para una ganancia de -1.

El hecho de que se comporte de esta manera hace que la salida digital sea un número con signo, por lo tanto, el bit más significativo será el signo y los 13 bits

restantes serán la magnitud, ya sea magnitud normal si el signo es positivo o magnitud en complemento a 2 si es negativo. Es por esto que la ecuación 1 sólo aplica para los números negativos y para los números positivos se utilizará la ecuación 2.

$$D[13:0] = GANANCIA \times \frac{(V_{IN} - 1.65 V)}{1.25 V} \times 8191 \quad (2)$$

En cambio, la salida del DAC (LTC 2624) es el análogo equivalente de un valor de 12 bits sin signo, escrito por el FPGA al DAC a través de una interfaz SPI, D[11:0], como se observa en la figura 3.

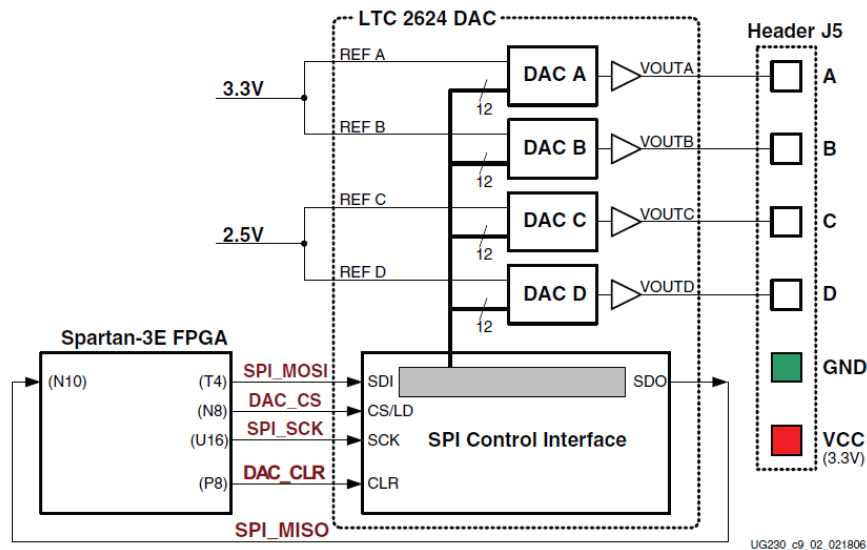


Figura 3 Vista detallada del circuito de conversión digital a análogo.

El voltaje de salida del DAC se describe en la ecuación 3. El voltaje de referencia es diferente en las 4 salidas del DAC. Los canales A y B usan un voltaje de referencia de 3.3 V, mientras que los canales C y D usan un voltaje de 2.5 V de referencia. En este caso se utilizó la salida A, por lo que el voltaje de referencia es de 3.3 V.

$$V_{OUT} = \frac{D[11:0]}{4096} \times V_{REFERENCIA} \quad (3)$$

El principal reto de esta aplicación es el uso de la interfaz SPI, ya que sólo se puede utilizar en un dispositivo a la vez: PGA, ADC o el DAC, por lo que fue

necesario crear una máquina de estados finitos (FSM) para el manejo total del sistema, con la cual fuera utilizada esta interfaz de la mejor manera posible. El diagrama de estados de la FSM se muestra en la figura 4.

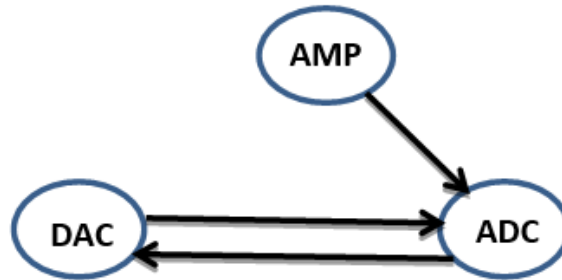


Figura 4 Diagrama de estados del sistema.

En el sistema se utilizó como entrada la señal de reloj de 50 MHz incluida en la tarjeta de desarrollo y a través del uso del administrador digital de reloj (DCM), se generó una señal de reloj de 80 MHz, obteniendo así una velocidad de 40 MHz en la operación de la interfaz SPI. Además, se generó otra señal de reloj de 2 MHz para el manejo del PGA, el cual sólo se configura una sola vez, antes que empiecen a trabajar el ADC y el DAC. El diagrama a bloques de la entidad del sistema se muestra en la figura 5.

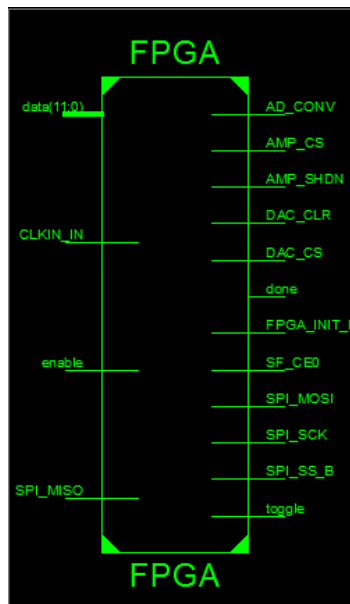


Figura 5 Diagrama de la Entidad.

Cada elemento del sistema (PGA, ADC, DAC) requirió de una FSM para su manejo particular. Para el caso del amplificador fueron necesarios 6 estados como se muestra en la figura 6. Estos estados se basan en el protocolo de la interfaz de comunicación serial SPI tal cual se muestra en la figura 7.

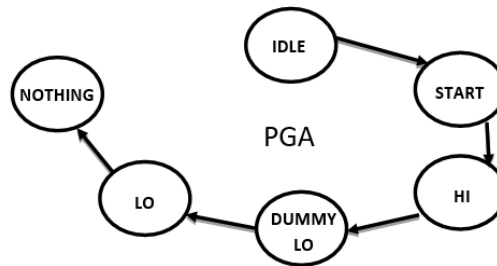


Figura 6 Diagrama de estados para el amplificador.

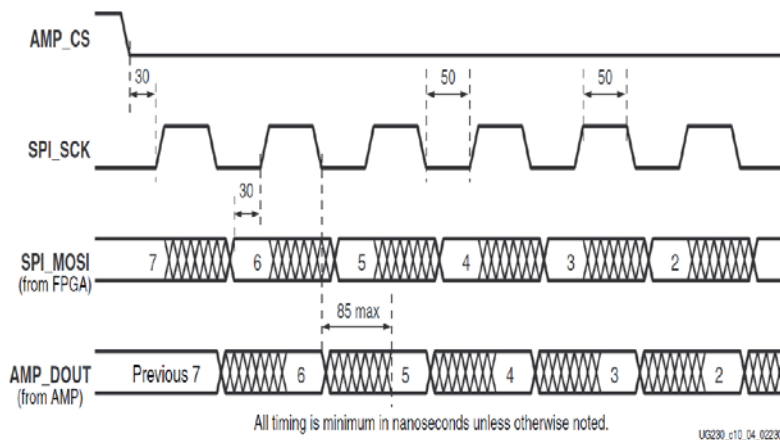


Figura 7 Comunicación SPI con el Amplificador.

En el diagrama de estados del PGA, figura 6, el primer estado, IDLE, se mantiene hasta que se active el bit de habilitación. Una vez que esto sucede pasa al estado START, donde se pone en cero la señal amp_cs, se inicia una cuenta en cero y se hace un retardo para posteriormente pasar al estado HI.

En este estado, se manda el bit más significativo del valor de la ganancia del amplificador, se pone en 1 la señal del reloj SPI, aumenta en uno la cuenta para después pasar al estado de DUMMY_LO.

Aquí, se evalúa que la cuenta no haya llegado a 8, número de bits a enviar, si no ha llegado, pone en 1 el reloj SPI y se pasa al estado LO. En este estado se

selecciona el bit siguiente a enviar, se pone en cero el reloj SPI y se regresa al estado HI donde esta vez se enviará el bit seleccionado.

Cuando la cuenta llega a 8, se han enviado los ocho bits del amplificador, por lo que la señal amp_cs se vuelve a poner en 1, se pasa al estado NOTHING y de ahí se pasa a la máquina de estados del ADC.

En la figura 8 se muestran los 6 estados requeridos para el manejo del ADC. En el estado IDLE_ADC se verifica que el sistema esté habilitado, lo cual se hace a través de un interruptor deslizable de la tarjeta.

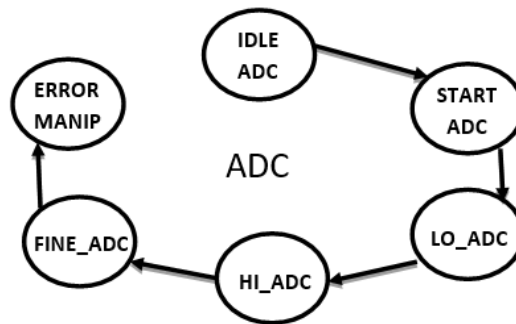


Figura 8 Diagrama de estados para el ADC.

Si no está habilitado el sistema, se regresa al estado inicial para configuración del PGA. En caso contrario se manda al estado de inicio START_ADC además de poner en alto AD_CONV que comienza simultáneamente la lectura de los dos canales del ADC.

En el estado START_ADC se pone en bajo AD_CONV, cambiando así al estado LO_ADC donde se verifica el valor de count para disminuirlo en uno, aumentar counter en 1, y cambiar al estado HI_ADC en el cual se monitorea por medio de rangos que el counter dure 34 ciclos de reloj para que se deje la señal SPI_MISO en alta impedancia, evitando el bloqueo de la comunicación SPI para los demás periféricos, y cambiando al estado FINE_ADC cuando el ADC ha terminado dando paso al estado siguiente el cual es ERROR_MANIPULATE donde se hace la linealización de la lectura, terminando así la máquina de estado del ADC, para luego continuar con la del DAC.

Para realizar la linealización de la lectura del ADC, se tomó en cuenta que la conversión análoga-digital se representa en un número de 14 bits y la salida del

DAC es de 12 bits. Se optó por descartar los valores de los 2 bits menos significativos del valor del ADC. Para linealizar la lectura se realizó el complemento a uno de los 11 bits menos significativos y manteniendo el valor del bit más significativo. En la figura 9 se muestra la salida normal del ADC y en la figura 10 se muestra ya linealizada.

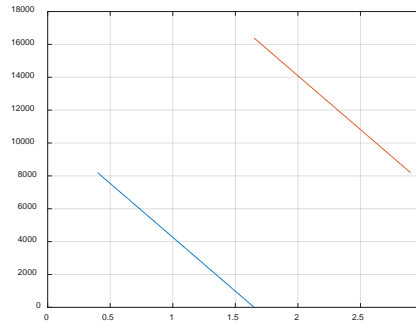


Figura 9 Lectura del ADC sin linealizar.

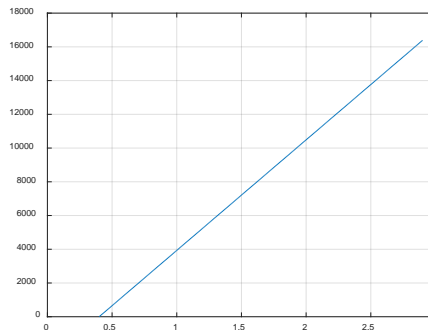


Figura 10 Lectura del ADC ya linealizada.

De igual manera, para el manejo del DAC, se utilizaron 6 estados como se muestra en la figura 11.

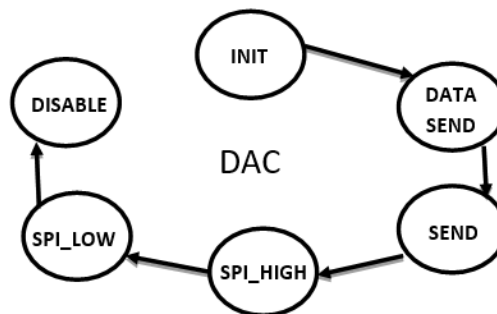


Figura 11 Diagrama de estados para el DAC.

La máquina de estados finitos para el DAC arranca con el estado INIT en el cual se inicia la conversión de datos al poner en alto DAC_CS y se asigna la longitud del dato a enviar, pasando así al estado DATA_SEND que recibe el vector arrojado por el último estado del ADC y este conjunto de bits se concatena con los bits de comando y selección del DAC, continuando al estado SEND, donde por medio de la interfaz SPI se inicia el envío del dato anteriormente mencionado, realizando el envío por medio de los estados SPI_HIGH y SPI_LOW que controlan la señal de reloj de la interfaz, siendo el último estado mencionado, el de DISABLE donde al finalizar el envío de datos se regresa al inicio de la máquina de estados del ADC, al estado IDLE_ADC, mientras que en la máquina del DAC se regresa al estado INIT.

3. Resultados

Una vez terminada la reconfiguración del FPGA, se realizaron las pruebas correspondientes para comprobar el correcto funcionamiento del sistema. Por lo tanto, se utilizó un generador de funciones, un osciloscopio y la tarjeta Spartan-3E que contiene el FPGA ya reconfigurado.

Como patrón de entrada se utilizó el generador de funciones para aplicar la señal analógica a procesar, una señal senoidal en este caso, se ajustó a una frecuencia baja de 100 Hz, de tal forma que se pudiera observar que la señal de salida corresponda a la entrada, dado que el procesamiento fue multiplicar la señal de entrada por 1, como se puede observar en la figura 12.

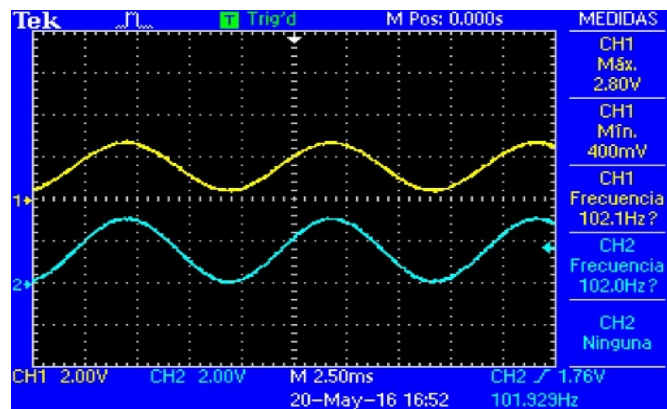


Figura 12 Respuesta del sistema a frecuencias bajas.

En figuras 13 y 14 se puede observar, además de la señal de entrada y la de salida, la señal que representa la activación del ADC que indica la frecuencia de muestreo.

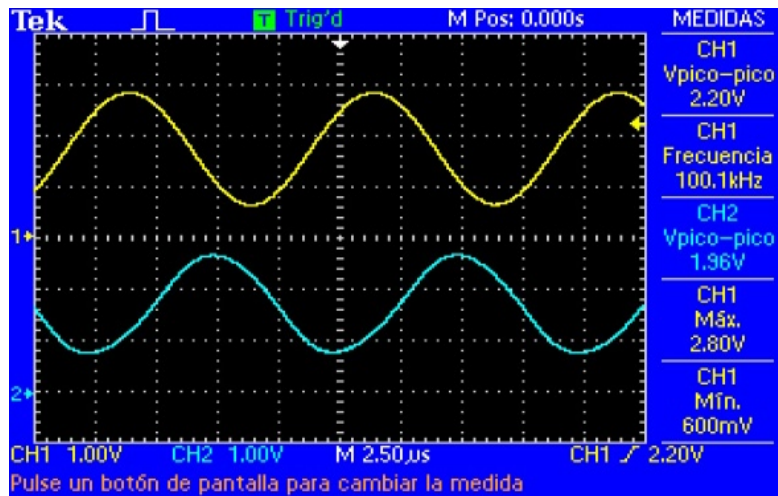


Figura 13 Sistema funcionando a 100 kHz.

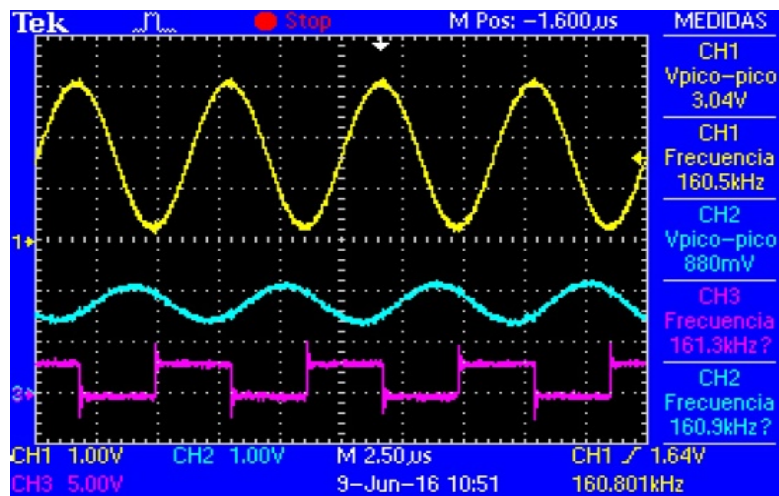


Figura 14 Sistema funcionando a 161 kHz.

4. Discusión

Se observó el rango de valores a los cuales el sistema mantuvo su funcionamiento, tras variar la frecuencia de la señal de entrada obteniendo el ancho de banda máximo soportado por el sistema, el cual fue 161 kHz.

Asimismo, tras aumentar la frecuencia de entrada más allá de la soportada, se generó el fenómeno de “aliasing” como se puede ver en la figura 15, donde se aplica la señal de entrada con una frecuencia de 322.79 kHz y se obtiene a la salida una señal senoidal pero de una frecuencia muy baja, 37.99 Hz, como consecuencia de no cumplir con el teorema de muestreo.

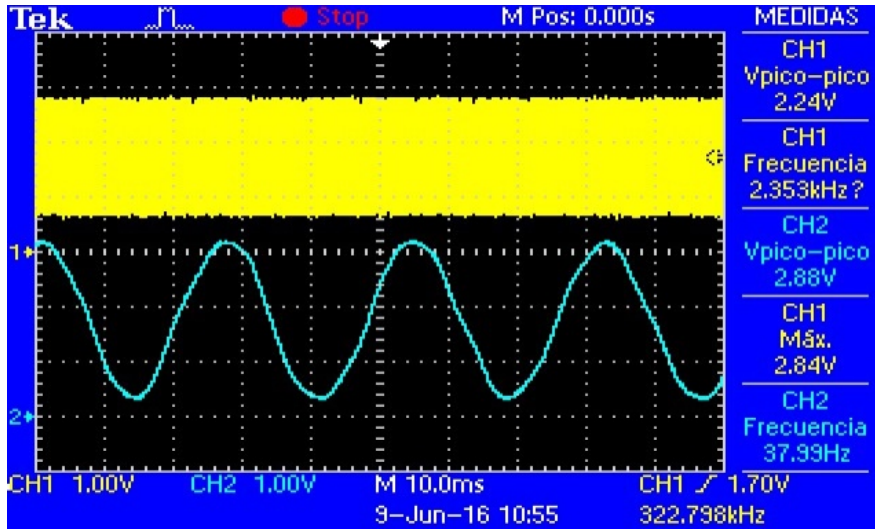


Figura 15 Sistema funcionando a 322 kHz con salida “alias” de 37.99 Hz.

En la figura 16 se aplica una frecuencia de entrada de 641 kHz y se obtiene un alias de 1.48 kHz.

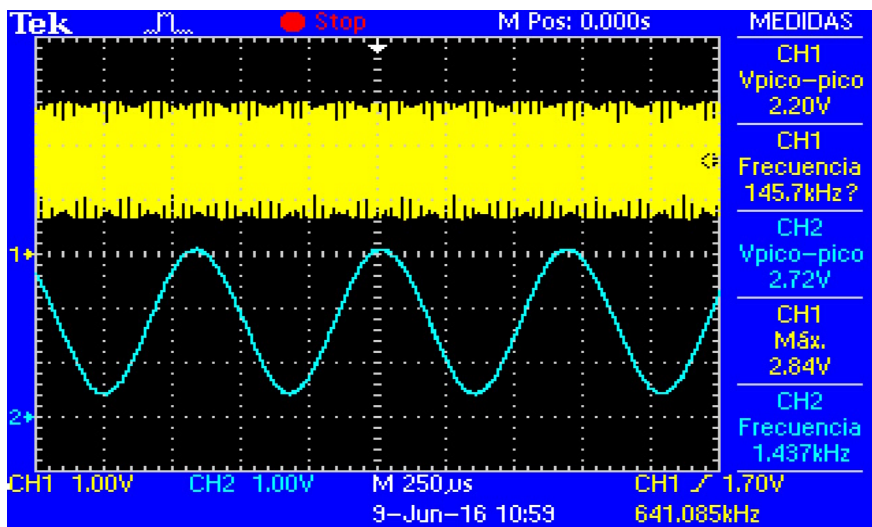


Figura 16 Sistema funcionando a 641 kHz con salida “alias” de 1.43 kHz.

Al inicio, el amplificador se utilizaba dentro del ciclo de captura y reproducción de la señal análoga, pero esto hacía que el ancho de banda se limitara a 10 kHz. Por lo que posteriormente se modificó para que solo se usara una sola vez, también se logró realizar un aumento al hacer uso del protocolo de 24 bits en lugar del de 32 bits que maneja el convertidor digital análogo (DAC). El sintetizador digital de frecuencia se programó a 80 MHz para una frecuencia de operación de la interfaz SPI de 40 MHz.

Gracias a la acción de iniciar el amplificador únicamente al arrancar el sistema y de utilizar el protocolo de 24 bits para el DAC, se obtiene un mayor ancho de banda.

5. Conclusiones

Se logró obtener el máximo ancho de banda permitido por el sistema para el procesamiento de señales hasta una frecuencia de 161 kHz, así como una razón de muestreo de 322 kmuestras/segundo a una velocidad de operación de 40 MHz de la interfaz SPI.

Esto es un aumento de 14% en el ancho de banda con respecto a [Silage, 2008] y a una velocidad 4% menor. La forma de manejo del amplificador y el uso del protocolo de 24 bits del DAC fueron la clave para lograr estos resultados. El manejo del amplificador de ganancia programable no es determinante para el aumento de la frecuencia de operación del sistema ya que sólo se accede a él en una sola ocasión al arranque del sistema.

Un trabajo futuro contemplaría el uso de una señal de reloj externa, en lugar de la de 50 MHz disponible en la tarjeta, con una frecuencia tal que se lleve al límite la velocidad de operación de la interfaz SPI que es de 50 MHz.

6. Bibliografía y Referencias

- [1] Domínguez, I. & Rodríguez, J. Procesador digital sincrónico en tiempo real soportado sobre un circuito FPGA Spartan-3E. *Revista de ciencia y tecnología*, enero/junio, 2011.
- [2] Linear Technology Corporation. LTC2624 DAC Datasheet, 2004.

- [3] Khedr, H.I., & Mostafa, A.G., & Radi, A., & Zidan, W.I. Controlling of Analog Capture Circuit and Digital Analog Converter for Spartan-3E FPGA Starter Kit. *Nature and Science*, 177-182, 2013.
- [4] Mascharak, S. Implementation of the onboard ADC and DAC on the Spartan 3E FPGA platform. Tesis. 2012.
- [5] Silage, Dennis. DSP on the Xilinx Spartan-3E Starter Board, 2008: <http://astro.temple.edu/~silage/pl-edpga.pdf>, Consultado Mayo de 2016.
- [6] Silage, Dennis. Embedded design using programmable gate arrays. Bookstand Publishing, 2008.
- [7] Xilinx Inc. Spartan-3E FPGA Starter Kit Board User Guide, 2011.