

# SISTEMA EMBEBIDO EN FPGA PARA LA MEDICIÓN DEL ÁNGULO DE INCLINACIÓN

Bedolla Niño, Emmanuel (1), Rodríguez Doñate, Carlos (2)

1 [Licenciatura en Comunicaciones y Electrónica, Universidad de Guanajuato, DICIS Irapuato-Salamanca Sede Yuriria] [[e.bedollanogto.mx]

2 [Departamento de Estudios Multidisciplinarios, DICIS, Campus Irapuato-Salamanca, Universidad de Guanajuato] [[c.rodriguezdonate@ugto.mx,]

## Resumen

En la actualidad para dotar al robot de sensibilidad con el mundo que le rodea, se necesita dotar a este, de sensores para detectar obstáculos y evitar colisiones o hacer los movimientos necesarios según la programación que se haga. Muchos robots con ayuda de un inclinómetro pueden saber su posición para realizar diferentes tareas. En este trabajo se presenta el diseño de un sistema embebido en FPGA para medir el ángulo de inclinación a partir de las señales de un acelerómetro mediante el algoritmo CORODIC bajo una arquitectura iterativa. Las pruebas experimentales se realizaron con un acelerómetro MMA8452Q, la implementación del sistema embebido se realizó en el kit de desarrollo DE1 de altera que tiene el FPGA Cyclone II EP2C20F484.

## Abstract

Nowadays, in order to provide the robot with sensitivity to the world around it, it needs to be equipped with sensors to detect obstacles and avoid collisions or to make the necessary movements according to the programming that is done. Many robots with the help of an inclinometer can know their position to perform different tasks. This paper presents the design of a system embedded in FPGA to measure the angle of inclination from the signals of an accelerometer through the algorithm CORODIC under an iterative architecture. Experimental tests were performed with an MMA8452Q accelerometer, implementation of the embedded system was performed in the alteration DE1 development kit that has the FPGA Cyclone II EP2C20F484.

## Palabras Clave

Sistema Embebido; VHDL; Inclinómetro; Acelerómetro; Arquitectura Iterativa; Algoritmo "CORDIC"; Inclinómetro

## INTRODUCCIÓN

En la actualidad, los robots son parte fundamental en el sector industrial ya que con ellos se puede incrementar la producción, eficiencia y precisión de la tarea que se les asigne; sin embargo, al ser una máquina carece de los sentidos de percepción por lo cual es necesario dotarlos con sensores que les permita conocer el entorno de trabajo. En algunos casos el conocer la inclinación en un robot es parte fundamental; por ejemplo, para mantener el control de sus movimientos el robot seguidor de líneas debe de conocer el ángulo de inclinación y en base a esto debe de incrementarla o disminuirla la velocidad y la aceleración [1].

Particularmente, un acelerómetro se ha utilizado para monitorear fallas en máquinas industriales [2]; no obstante, a partir de las señales del acelerómetro se puede inferir el ángulo de inclinación. Otras formas de obtener la inclinación se basan en la deformación de una varilla de acero que experimenta torsión respecto de la inclinación a la que se le somete. Otros inclinómetros están basados en un lastre libre que actúa por la acción de la fuerza gravitacional, y otros se basan en el desplazamiento de una burbuja de aire en un medio líquido confinado [3].

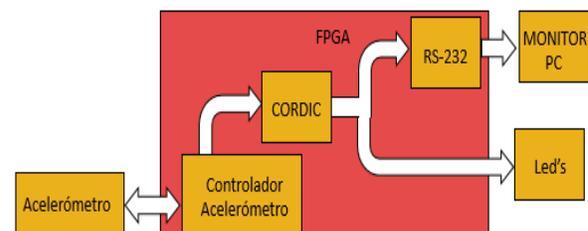
En este proyecto se ha optado por el uso del acelerómetro para construir un inclinómetro, por lo cual es necesario conocer algoritmos que puedan ser utilizados para estimar el cálculo de funciones trigonométricas de forma eficiente y con bajos recursos computacionales. Entre estos algoritmos se destaca el CORDIC (Coordinate Rotation Digital Computer, Computadora Digital para Rotación de Coordenadas) que proporciona un método iterativo para rotar vectores y su operación solo se basa en sumas y desplazamientos que lo hace idóneo para ser implementado en hardware, y se pueden estimar funciones trigonométricas circulares, hiperbólicas y funciones lineales. El algoritmo original fue propuesto por Jack Volder en el año 1959 [4]. Normalmente, el CORDIC se puede operar en modo rotación o modo vectorización, en este trabajo solo se utilizó el segundo modo, el cual rota el vector de entrada hacia el eje X, minimiza la magnitud del componente Y, ya que si  $Y = 0$  entonces el vector se encuentra sobre el eje X. Asimismo se utiliza el signo del componente Y para determinar la dirección de rotación. Si el

acumulador angular se inicializa con cero, al final del proceso contendrá el ángulo necesario para efectuar dicha rotación. [5]

En este trabajo se presenta el diseño de un sistema embebido en FPGA (field programmable gate array) para medir el ángulo de inclinación a partir de las señales de un acelerómetro. Donde la arquitectura digital del algoritmo CORDIC fue implementado de forma iterativa. Las pruebas experimentales se realizaron con un acelerómetro MMA8452Q, la implementación del sistema embebido se realizó en el kit de desarrollo DE1 de altera que tiene el FPGA Cyclone II EP2C20F484.

## MATERIALES Y MÉTODOS

El diagrama general del sistema embebido diseñado para la medición del ángulo de inclinación se muestra en la imagen 1, el cual está integrado por: un Acelerómetro, un controlador para el acelerómetro, la implementación del algoritmo CORDIC, el protocolo RS-232 para mostrar el ángulo obtenido en el monitor y por último también se mostrará el resultado en los leds de la tarjeta.



**IMAGEN 1: Diagrama de la Metodología utilizada que obtiene el ángulo de 0° a ±180°.**

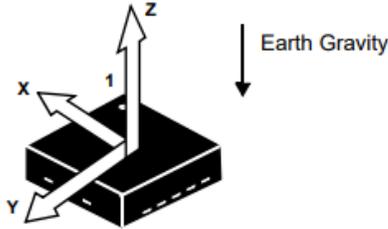
A continuación, se describirán los principales elementos del sistema embebido

### Acelerómetro:

La detección de inclinación con un acelerómetro se basa en que el único estímulo de aceleración es asociado con la gravedad, por lo tanto, la detección de inclinación utiliza el vector de gravedad y su proyección sobre los ejes del acelerómetro para determinar el ángulo de inclinación.

El acelerómetro como tal no conoce en ningún momento su posición, sin embargo, conociendo las

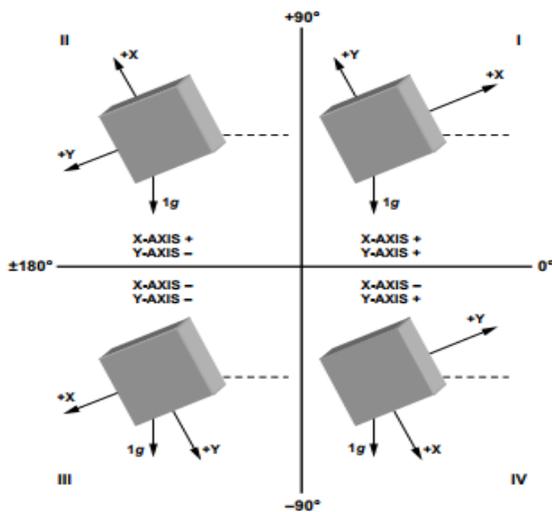
componentes de la fuerza G aplicado al acelerómetro se puede conocer el ángulo [6], como se muestra en la imagen 2.



**IMAGEN 2: Dirección de las aceleraciones detectables [6]**

Al utilizar un segundo eje se logra tener la capacidad de distinguir entre cada cuadrante y de medir ángulos a lo largo de todo el arco de 360°. Como se muestra en la Imagen 3, cada cuadrante tiene una combinación diferente de signos asociados con la aceleración de eje X e Y.

La función de tangente inversa devuelve un valor en el Cuadrante I si el operando, AX, OUT / AY, OUT, es positivo; Si el operando es negativo, la función tangente inversa devuelve un valor en Cuadrante IV. Debido a que el operando en el cuadrante II es negativo, se debe agregar un valor de 180 ° al resultado del cálculo cuando el ángulo está en ese cuadrante. Debido a que el operando en el Cuadrante III es positivo, se debe sustraer un valor de 180 ° del resultado del cálculo cuando el ángulo está en ese cuadrante. El cuadrante correcto del ángulo calculado se puede determinar examinando el signo de la aceleración medida en cada eje.



**IMAGEN 3: Ángulo de Inclinación y Signo de Aceleración para la Detección de Cuadrante**

Se utilizó el software Active-HDL 7.1 para realizar la simulación y arquitecturas digitales, el acelerómetro utilizado es un MMA8452Q, la implementación del sistema embebido se realizó en el kit de desarrollo DE1 de altera que tiene el FPGA Cyclone II EP2C20F484.

Para trabajar el algoritmo CORDIC el modo de vectorización se utilizan las siguientes ecuaciones.

$$X_{i+1} = X_i - Y_i d_i 2^{-i} \quad (1)$$

$$Y_{i+1} = Y_i + X_i d_i 2^{-i} \quad (2)$$

$$Z_{i+1} = Z_i - d_i \arctan(2^{-i}) \quad (3)$$

$$\text{En donde } d_i = \begin{cases} -1, & \text{si } Y_i \geq 0 \\ 1, & \text{si } Y_i < 0 \end{cases} \quad (4)$$

Además, cabe recalcar que el algoritmo solo puede calcular el ángulo desde +90° a -90° por lo cual se agregaron las siguientes condiciones para que pudiera calcular el ángulo de 0° a ±180°.

Primer cuadrante ángulo de 0 a 90°:

Si  $X_{inicial} \geq 0$  y  $Y_{inicial} \geq 0$  entonces

$$Z_{Final} = Z \quad (5)$$

Segundo cuadrante ángulo de 90° a 180°:

Si  $X_{inicial} < 0$  y  $Y_{inicial} \geq 0$  entonces

$$Z_{Final} = \pi - Z \quad (6)$$

Tercer cuadrante ángulo de -90 a -180°:

Si  $X_{inicial} < 0$  y  $Y_{inicial} < 0$  entonces

$$Z_{Final} = -\pi - Z \quad (7)$$

Cuarto cuadrante ángulo de 0 a -90°:

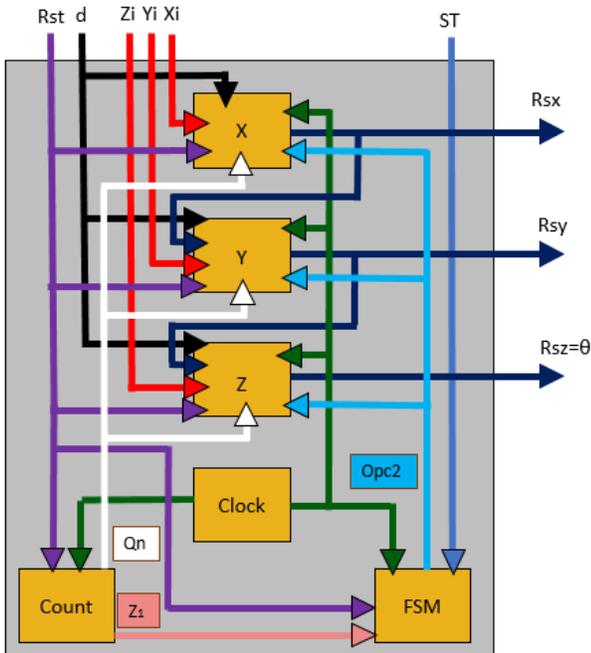
Si  $X_{inicial} \geq 0$  y  $Y_{inicial} < 0$  entonces

$$Z_{Final} = Z \quad (8)$$

### Algoritmo CORDIC y sus arquitecturas digitales

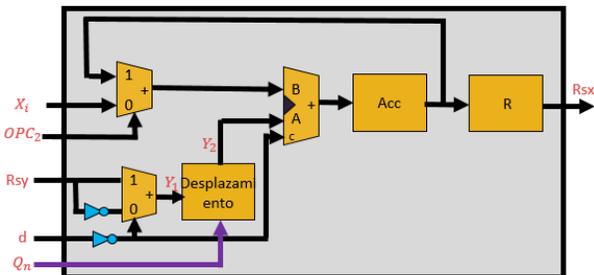
El diagrama general de la imagen 4 muestra la arquitectura para el algoritmo CORDIC este diseño está compuesta por un contador, una máquina de estados. Donde la señal Zi es la condición inicial del ángulo el cual tiene un valor de 0 radianes, la señal Xi es la condición inicial del vector en la dirección X que corresponde a la medición del eje X del

acelerómetro, la señal  $Y_i$  es la condición inicial del vector en la dirección Y que corresponde a la medición del eje Y del acelerómetro, la señal  $d$  corresponde a la comparación de la ecuación 4, la señal  $Rst$  es el reset maestro, la señal  $ST$  es el inicio de operación de la arquitectura,  $Q_n$  es el número de iteración,  $OPC2$  es una señal de control, finalmente  $R_{sx}$ ,  $R_{sy}$  y  $R_{sz}$  es el resultado del algoritmo después de analizar su operación.



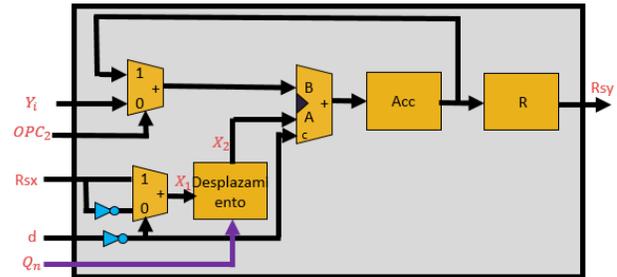
**IMAGEN 4:** Arquitectura de algoritmo Cordic

La imagen 5 muestra la arquitectura para calcular la ecuación (1), que corresponde al vector X a partir de la cual se obtiene la magnitud del vector ( $X_i$ ,  $Y_i$ ).



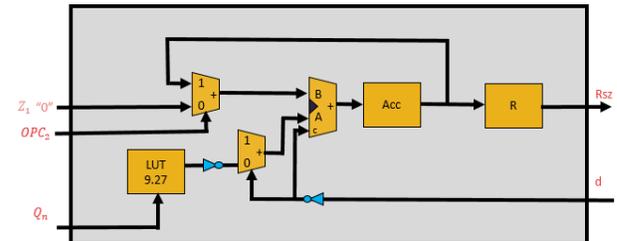
**IMAGEN 5:** Arquitectura del componente "X"

La imagen 6 muestra la arquitectura para calcular la ecuación 2, que corresponde al vector Y, en cuanto el valor de  $R_{sy}$  se acerque a cero, indica que el resultado del algoritmo es más preciso, lo cual depende de las iteraciones que realice la arquitectura.



**IMAGEN 6:** Arquitectura del componente "Y"

Finalmente, el diagrama de la imagen 7 calcula la ecuación (3) que corresponde al ángulo del vector ( $X_i$ ,  $Y_i$ ), el cual determina la inclinación referente a las mediciones del acelerómetro. En esta arquitectura la ROM (LUT) tiene los valores de en un formato de punto fijo de 9.27.



**IMAGEN 7:** Arquitectura del componente "z"

Como se pudo observar en las diferentes arquitecturas solo se requiere de un sumador, un acumulador (Acc), un registro para almacenar el resultado (R) y dos multiplexores, el primero para ingresar las condiciones iniciales o los resultados de cada iteración y el segundo para realizar la operación de la ecuación (4).

## RESULTADOS Y DISCUSIÓN

La prueba del sistema consistió en mover el acelerómetro de  $0^\circ$  a  $180^\circ$ . Donde el eje X está en la dirección axial, el eje Y en dirección tangencial y el eje Z en dirección radial, en base a estas condiciones, la medición del acelerómetro del eje Y se conectó en la señal  $X_i$  de la arquitectura de la

imagen 4 y la medición del eje Z en la señal  $Y_i$ . La simulación que muestra el correcto funcionamiento del sistema embebido se muestra en las imágenes 8, 9 y 10, donde la primera corresponde a la medición del eje Y, la segunda a la medición del eje Z y la última corresponde al ángulo estimado con el sistema embebido que se diseñó.

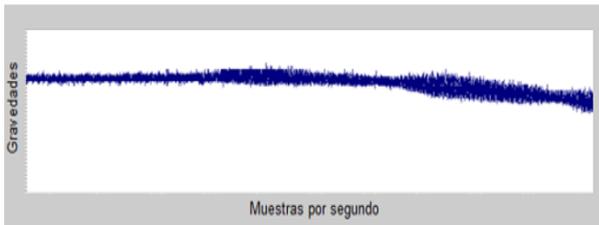


IMAGEN 8: Señal de simulación del Acelerómetro del eje X

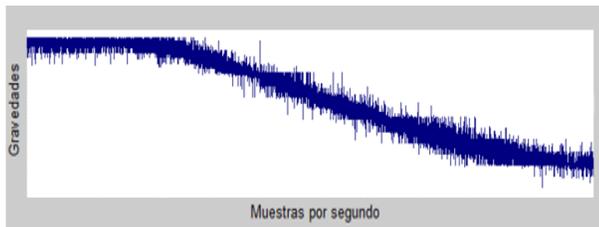


IMAGEN 9: Señal de simulación del Acelerómetro del eje Y

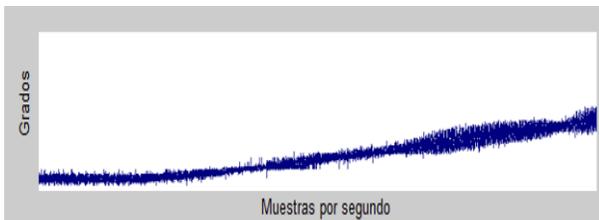


IMAGEN 10: Señal de simulación del de ángulo obtenido

## CONCLUSIONES

En base a los resultados obtenidos se puede demostrar que el sistema embebido diseñado puede estimar el ángulo de inclinación a partir de la medición de las señales de dos ejes de un acelerómetro. Además, la arquitectura digital diseñada del algoritmo CORDIC fue implementada de forma iterativa por lo cual el consumo de recursos fue muy bajo comparado con una arquitectura pipeline. Como prospectivas del trabajo, es integrar este sistema embebido a un robot para que pueda realizar la estimación del ángulo de inclinación de forma precisa.

## AGRADECIMIENTOS

Al DEM-Yuriria DICIS, a la Dirección de Apoyo a la Investigación y Posgrado (DAIP) de la Universidad de Guanajuato, a Altera University Program, al programa NPTC del PRODEP por el proyecto DSA/103.5/16/10374.

## REFERENCIAS

- [1] Sánchez Castro, X., García, J., & Valdivieso, C. (2011). Aplicación con inclinómetro y acelerómetro para mantener control de movimiento de un robot.
- [2] Fuentes Sacha, El Acelerómetro en los móviles, <http://www.xatakamovil.com/variados/el-acelerometro-en-los-moviles>, fecha de consulta 23 abril 2011
- [3] Fisher, C. J. (2010). Using an accelerometer for inclination sensing. *AN-1057, Application note, Analog Devices*.
- [4] Volder, J., "The CORDIC Trigonometric Computing Technique," *IRE Trans. Electronic Computing*, Vol EC-8, pp. 330-334 September 1959.
- [5] Schweers, R. J. (2002). *Descripción en VHDL de arquitecturas para implementar el algoritmo CORDIC* (Doctoral dissertation, Facultad de Informática).
- [6] MMA8452Q, X. (2014). 3-Axis, 12-Bit/8-Bit Digital Accelerometer. 2013-10]. [http://cache.freescale.com/files/sensors/doc/data\\_sheet/MMA8452Q.pdf](http://cache.freescale.com/files/sensors/doc/data_sheet/MMA8452Q.pdf).