

Desarrollo de un vehículo guiado autónomamente tipo triciclo mediante la implementación de un DSPF28335 y Matlab Simulink en tiempo real

Development of an Autonomously Guided Tricycle-Type Vehicle Using Real-Time Implementation of DSPF28335 and Matlab Simulink

- ID** **Juan Manuel Barrera-Fernández** es alumno de doctorado en mecatrónica en el Centro de ingeniería y Desarrollo Industrial CIDESI Querétaro (jbarrera@posgrado.cidesi.edu.mx) (México) (<https://orcid.org/0000-0002-2168-3408>), Maestro.
- ID** **Julio César Solano-Vargas** es Profesor Investigador en el Area de Física Aplicada en el Centro de ingeniería y Desarrollo Industrial CIDESI Querétaro (jbarrera@posgrado.cidesi.edu.mx) (México) (<https://orcid.org/0000-0003-4745-0175>), Doctor.
- ID** **Juan Pablo Manzo-Hernández** es alumno de doctorado en mecatrónica en el Centro de ingeniería y Desarrollo Industrial CIDESI Querétaro (jbarrera@posgrado.cidesi.edu.mx) (México) (<https://orcid.org/0009-0005-7464-0670>), Maestro.

Resumen: En este artículo se presenta una metodología para el desarrollo eléctrico y electrónico, programación y evaluación de un Vehículo Guiado Autónomamente (AGV) de tipo triciclo, diseñado para aplicaciones industriales y logísticas. El sistema fue construido utilizando Matlab Simulink, una herramienta poderosa en la simulación y análisis de sistemas dinámicos, lo que permitió una programación precisa del control de posición y dirección del AGV. El proceso de desarrollo incluye la integración del algoritmo de control Proporcional-Integral-Derivativo (PID) que fue implementado y ajustado para optimizar la respuesta dinámica del vehículo en diversas etapas de su movimiento, como arranque, frenado y cambio de dirección. Los resultados obtenidos a partir de las simulaciones y pruebas experimentales demostraron la eficacia del controlador PID en mejorar la estabilidad y precisión del AGV, lo que subraya su potencial para aplicaciones en entornos automatizados donde la precisión y la fiabilidad son críticas. Finalmente, el artículo propone un análisis exhaustivo de los datos obtenidos, discutiendo las limitaciones y posibles mejoras futuras en la implementación del sistema, con el objetivo de contribuir al avance en la automatización y el control de vehículos autónomos en la industria.

Palabras clave: Vehículo Autónomo, Electrónica, Servo-motor EC, PID, Simulink.

Cómo citar: Barrera-Fernández, J.M., Solano-Vargas, J.C. y Manzo-Hernández, J.P. (2024). Desarrollo de un vehículo guiado autónomamente tipo triciclo mediante la implementación de un DSPF28335 y Matlab Simulink en tiempo real. *Tecnología, Ciencia y Estudios Organizacionales*, 6(12), pp. 38-58. <https://doi.org/10.56913/teceo.5.12.38-58>

Recepción: 21-09-2024
Aprobación: 23-10-2024

Abstract: This article presents a detailed methodology for the electrical and electronic development, implementation, and evaluation of an Autonomously Guided Vehicle (AGV) of the tricycle type, designed for industrial and logistical applications. The system was built using Matlab Simulink, a powerful tool for simulating and analyzing dynamic systems, which enabled precise programming of the AGV's position and direction control. The development process includes the integration of a Proportional-Integral-Derivative (PID) control algorithm, which was implemented and tuned to optimize the vehicle's dynamic response in various stages of movement, such as starting, braking, and changing direction. The results obtained from simulations and experimental tests demonstrated the effectiveness of the PID controller in improving the AGV's stability and accuracy, highlighting its potential for applications in automated environments where precision and reliability are critical. Finally, the article proposes an exhaustive analysis of the obtained data, discussing the limitations and potential future improvements in the system's implementation, with the aim of contributing to the advancement of automation and control of autonomous vehicles in the industry.

Keywords: Autonomous Vehicle, Electronic, Servo-motor EC, PID, Simulink.

Introducción

En los últimos años, el desarrollo de vehículos autónomos ha experimentado un crecimiento significativo, impulsado por avances en tecnologías de sensores, procesamiento de datos y algoritmos de control. Estos vehículos han encontrado aplicaciones en una variedad de entornos, desde el transporte público hasta la logística industrial. En particular, los vehículos autónomos tipo triciclo ofrecen una plataforma versátil y eficiente para operaciones en zonas cerradas controladas, como fábricas, almacenes y campus universitarios. La estructura simplificada (contiene dirección y tracción en un solo eje) y la estabilidad inherente de los triciclos los hacen ideales para maniobras en espacios reducidos y para la implementación de metodologías auxiliares de navegación para el control de dirección y tracción.

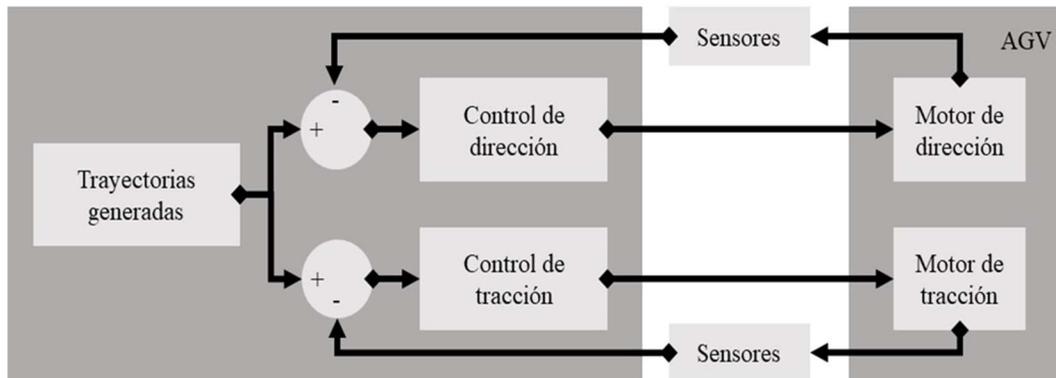
Este artículo presenta un proyecto de desarrollo de un vehículo autónomo tipo triciclo diseñado específicamente para operar en entornos cerrados controlados y el cual puede usarse para implementar diferentes metodologías de generación y seguimiento de trayectorias. El sistema de control se basa en la implementación de un controlador proporcional integral derivativo (PID), que proporciona una solución sencilla y efectiva para el mantenimiento de la trayectoria y la regulación de velocidad. La plataforma de desarrollo se centra en el uso del microcontrolador DSPF28335, conocido por su capacidad de procesamiento en tiempo real y su idoneidad para aplicaciones de control de motores y sistemas embebidos. Los motores utilizados son motores Maxon motor EC 30 de 40 watts con una configuración inicial 272768 a 8 Nm, que proporcionan el torque y la velocidad adecuada para el buen funcionamiento del vehículo.

Para el desarrollo del código en lenguaje C, se utiliza la herramienta de software Code Composer Studio, que ofrece un entorno integrado para la programación, depuración y optimización de aplicaciones de tiempo real. El objetivo principal de este proyecto es demostrar la viabilidad y eficiencia de un sistema de control simple para la navegación autónoma de vehículos tipo triciclo en zonas cerradas, proporcionando una base para futuras mejoras y aplicaciones más complejas.

Método

Este trabajo presenta una metodología para el control de tracción y dirección en sistemas autónomos utilizando servomotores EC de Maxon Motor (Figura 1). La metodología se divide en tres etapas: primero, la generación de trayectorias mediante vectores Spline; segundo, la conversión de estas trayectorias en señales de control a través de un controlador PID; y tercero, el acondicionamiento de las señales para los servomotores, utilizando retroalimentación de encoders. Todo el sistema es gestionado en tiempo real por un procesador DSPF28335, asegurando un control eficiente y preciso del movimiento del vehículo.

Figura 1. Diagrama del sistema de control de tracción y dirección de un AGV tipo triciclo.



Primera Etapa: Generación de Trayectoria mediante Vectores Spline

En la primera fase del proceso, se utiliza la metodología de generación de trayectorias por vectores Spline para definir la trayectoria deseada del vehículo cuya presentación de la metodología y resultados fueron presentados por (Miramontes et al., 2021). Este método permite crear trayectorias suaves y continuas, ideales para el movimiento preciso del sistema autónomo. Los vectores Spline se calculan en Matlab Simulink, donde se define el perfil de velocidad y la orientación en función del tiempo. Esta trayectoria generada sirve como referencia para el control de tracción y dirección de los motores.

Segunda Etapa: Transformación de Vectores a Señales de Control

Una vez generada la trayectoria, el siguiente paso consiste en transformar estos vectores Spline en señales de control para los servomotores. Para ello, se emplea un controlador Proporcional-Integral-Derivativo (PID), que ajusta las señales de control basadas en el error entre la trayectoria deseada y la posición actual de los motores. El controlador PID es clave para asegurar que las respuestas de los motores en términos de tracción y dirección sean precisas y rápidas, minimizando el error y optimizando el rendimiento dinámico del sistema.

Tercera Etapa: Acondicionamiento de Señales de Control y Retroalimentación

En esta etapa, las señales de control generadas por el controlador PID se acondicionan para ser compatibles con el servocontrolador Maxon Motor. Esto incluye la adaptación de las señales de control para asegurar que sean interpretadas correctamente por los servomotores. Además, se implementa un sistema de retroalimentación basado en un encoder incremental para la tracción y un encoder absoluto de tipo analógico para el control de dirección. Estos dispositivos proporcionan información en tiempo real sobre la posición y velocidad de los motores, que es utilizada por el sistema de control para ajustar las señales de manera continua.

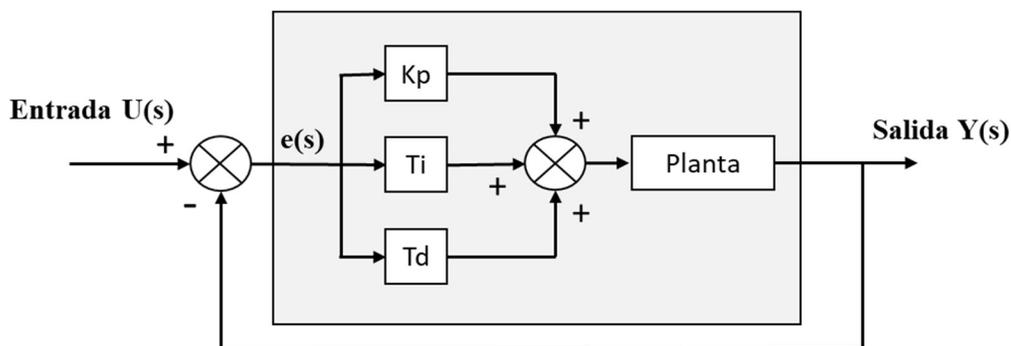
Finalmente, todo el sistema de control es operado por el procesador DSPF28335, que gestiona tanto la generación de trayectorias como el procesamiento de las señales de retroalimentación. Las señales de control son desplegadas en tiempo real para su análisis, permitiendo realizar ajustes y mejoras durante la operación del vehículo autónomo. Este enfoque asegura una alta precisión y estabilidad en el control de tracción y dirección, optimizando el rendimiento general del sistema.

La metodología de integración del software Matlab Simulink para monitorear las señales de control de dirección y velocidad, así como para la programación en tiempo real del vehículo autónomo, se basa en un enfoque sistemático que combina la simulación, modelado y despliegue directo en hardware. Inicialmente, se desarrollan modelos de programación del sistema de control de dirección y tracción en Matlab Simulink, donde se representan las dinámicas del vehículo y el monitoreo de las señales de control. Estos modelos de control son validados a través de pruebas en tiempo real para garantizar su comportamiento correcto bajo diversas condiciones operativas. Posteriormente, se emplea la funcionalidad de generación de código automático de Simulink para traducir los modelos de control en código C, que es entonces desplegado en el procesador DSPF28335. Durante la operación del vehículo, Matlab Simulink se utiliza para monitorear y ajustar las señales de control en tiempo real, permitiendo la adquisición de datos y la afinación de los parámetros del controlador. Este enfoque integrado asegura una comunicación eficiente entre el software y el hardware, optimizando el rendimiento del vehículo autónomo.

Diseño del controlador PID

En la automatización de vehículos guiados autónomamente (AGV), el control de posición y velocidad es fundamental para garantizar un desplazamiento eficiente y seguro en áreas de trabajo dinámicas. Este trabajo introduce un algoritmo de control Proporcional-Integral-Derivativo (PID), ver Figura 2, diseñado específicamente para gestionar simultáneamente dos sistemas clave en un AGV: el control de posición y el control de velocidad. El algoritmo PID ajusta continuamente las señales de control en respuesta a las desviaciones de la trayectoria y la velocidad deseadas, proporcionando correcciones en tiempo real que permiten al AGV mantener su curso y velocidad con alta precisión. Esta implementación asegura que el vehículo responda de manera rápida y estable a las variaciones en el entorno operativo, mejorando así su desempeño y fiabilidad en aplicaciones industriales y logísticas.

Figura 2. Diagrama del controlador PID.



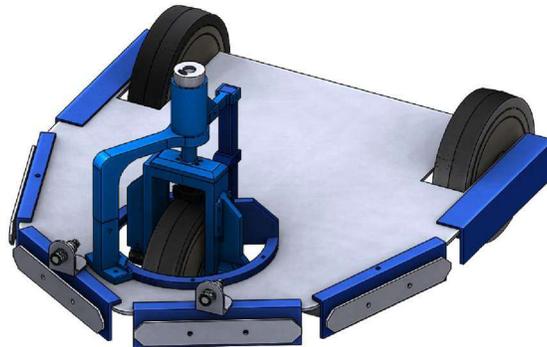
Resultados

En la sección de resultados, se presenta un análisis detallado del rendimiento del prototipo de AGV tipo triciclo, incluyendo la construcción del vehículo, la configuración de los drivers para los motores de tracción y dirección, y la implementación del controlador PID. El prototipo fue diseñado con un enfoque en la simplicidad y eficiencia, integrando un motor directamente en el eje de tracción y otro en el eje de dirección, ambos controlados por drivers configurados para maximizar el rendimiento del sistema.

Construcción de prototipo

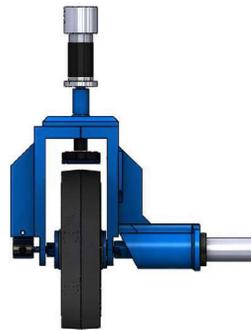
El prototipo del AGV (Figura 3) tipo triciclo fue diseñado con un enfoque en la simplicidad y eficiencia mecánica, integrando el motor directamente en el eje de tracción y dirección. Esta configuración elimina la necesidad de acoplamientos mecánicos adicionales, lo que reduce las pérdidas de energía y mejora la transmisión de potencia. Al colocar el motor directamente en el eje, se logra un mayor torque, optimizando la capacidad de tracción y maniobrabilidad del vehículo. Esta disposición permite un control más preciso y una respuesta más rápida a las demandas de movimiento, mejorando el rendimiento general del AGV en aplicaciones industriales.

Figura 3. Prototipo de vehículo AGV tipo triciclo.



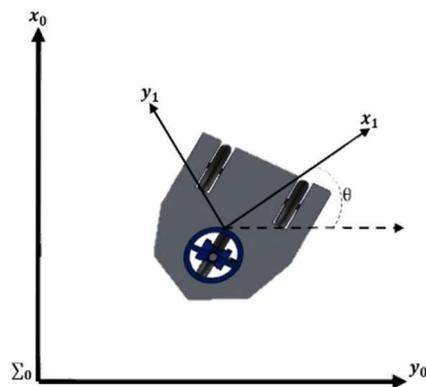
En el diseño del sistema de dirección del AGV tipo triciclo, se instaló un sensor magnético analógico en el eje de dirección ver Figura 4. Este sensor funciona mediante la interacción entre un imán montado en el eje y el propio sensor, que detecta la rotación relativa entre ambos componentes. A medida que el eje del AGV gira, el campo magnético cambia, lo que genera una señal analógica proporcional al ángulo de rotación del eje. Esta señal analógica proporciona una medida precisa y continua del ángulo de dirección del vehículo, que es crucial para el control exacto de la orientación y la trayectoria del AGV en aplicaciones dinámicas.

Figura 4. Estructura básica del prototipo del eje principal tracción y dirección.



En el análisis del movimiento del AGV tipo triciclo, se definen dos marcos de referencia: el fijo y el móvil. El marco de referencia fijo está asociado con el entorno en el que opera el vehículo, mientras que el marco de referencia móvil se mueve junto con el vehículo. El desplazamiento angular del AGV respecto al marco fijo se denota como θ (theta) como se muestra en la Figura 5. Este ángulo θ describe la orientación del vehículo en relación con su posición inicial y es crucial para determinar su trayectoria y dirección. Al considerar ambos marcos de referencia, se puede modelar con precisión el comportamiento dinámico del AGV, lo que permite un control efectivo del sistema de dirección y tracción en tiempo real.

Figura 5. Marcos de referencia fijo y móvil en el vehículo móvil AGV



Programación Simulink

Antes de configurar el DSP se necesitan 4 programas clave.

1. Instalar Matlab 2020 o posteriores con el toolkit: Support package C2000 TI.
2. Se puede instalar en el apartado de Add On: Get Hardware: C2000 TI: Install
3. Instalar Control Suite Texas Instruments.
4. Instalar C2000ware Texas Instruments
5. Instalar Code Composer V8.2

Salidas/Entradas Digitales Matlab Simulink

1. Abrir un archivo nuevo de Simulink Matlab.

2. En la pestaña de Modeling y con el DSP conectado vía USB, configurar el framework de Simulink.

En este paso se recomienda localizar la pestaña de Hardware Implementation

- i. Hardware Board: TI Delfino f2833x

1. Texas Instruments: Device vender
2. Device type: C2000
3. Clic “Apply”. (Se puede dar la opción de (boot from flash) esto ejecuta el programa sin la necesidad de conexión de matlab hacia el DSP.
4. OK.

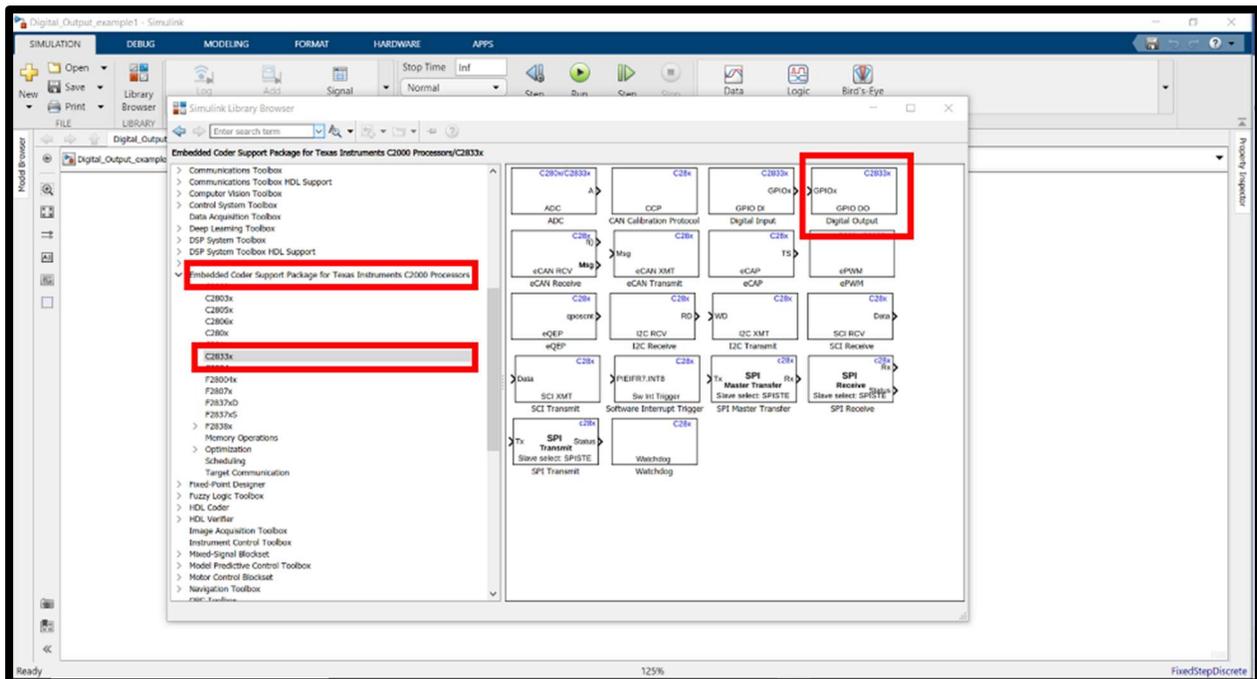
Esta configuración inicial, se debe de realizar para todo programa en donde se requiera programar el DSP. Una vez hecho esta configuración, inicializamos las librerías de programación.

1. Library Browser

Ubicar: Embedded Coder Support Package for Texas Instruments C2000 Processors

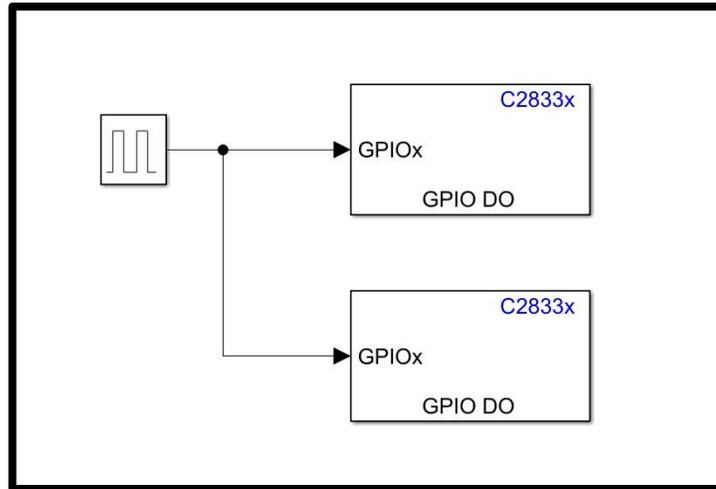
- Abrir: C2833x
 - Localizar: Digital Output, como se muestra en la Figura 6.

Figura 6. Selección del bloque “Digital output”.



A continuación, se muestra el funcionamiento y configuración para cada una de las diferentes modalidades que implementa el bloque de salidas digitales. El ejercicio que se propone para este apartado es el de la figura siguiente. Donde consta de una señal cuadrada de entrada y unida hacia las salidas configuradas con los puertos GPIO0 y GPIO1, como se muestra en la Figura 7.

Figura 7. Ejemplo 1, señales de salida digitales



Para la configuración del GPIO0, se habilita la señal digital, como se muestra en la figura 13, mientras que en la configuración del GPIO1 se habilita la señal de Toggle GPIO Figura 8 y Figura 9; este es un funcionamiento adicional a las salidas digitales, en donde cambiara su estado en cada evento o cambio del bit de entrada (toggle bit). La configuración de la señal cuadrada con una amplitud de 1, periodo 2, ancho de pulso 50% y un retraso de fase de 0 segundos.

Figura 8. Configuración y selección de las salidas digitales GPIO.

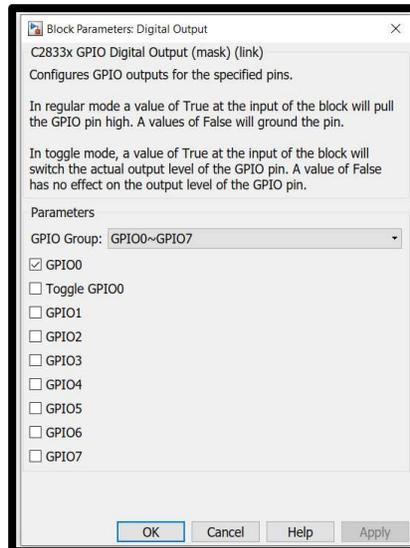
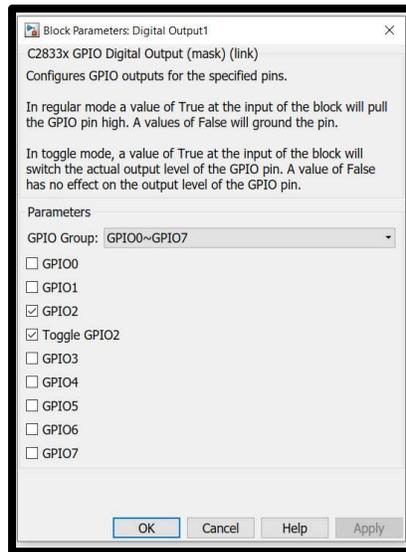


Figura 9. Configuración y selección de las salidas digitales GPIO toggle bit.



Para la implementación se recomienda utilizar “jumpers” o cables dupont para observar las señales de salida; recordando que el 1 lógico se observa a la salida con un voltaje de 3.3V en cada uno de los GPIO como se muestra en la Figura 10 y Figura 11.

Figura 10. Diagrama de conexión

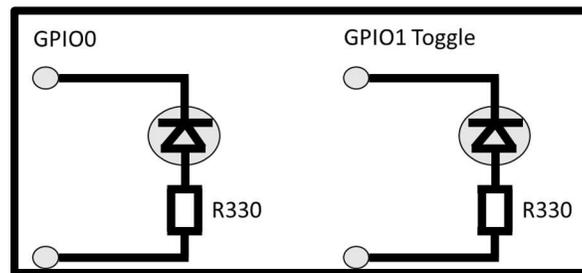
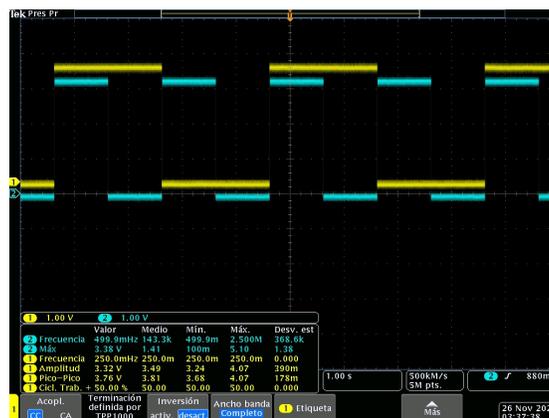


Figura 11. Gráfica de señales de salidas digitales GPIO y GPIO toggle bit



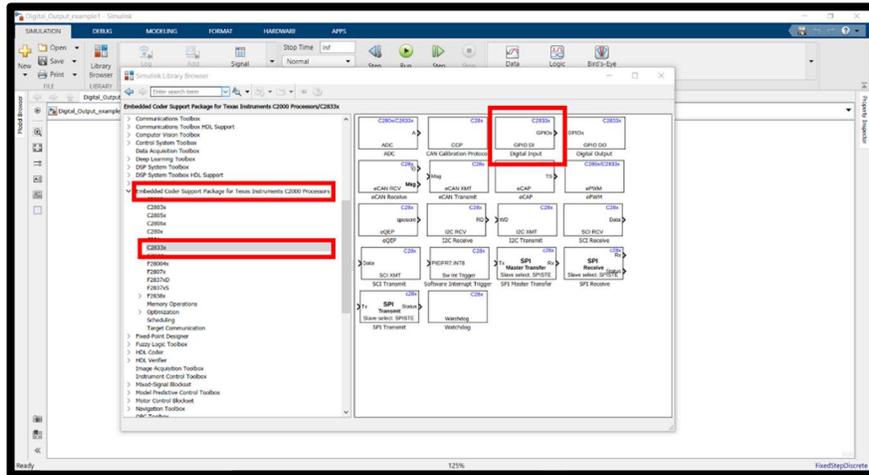
Library Browser

1. Library Browser

Ubicar: Embedded Coder Support Package for Texas Instruments C2000 Processors

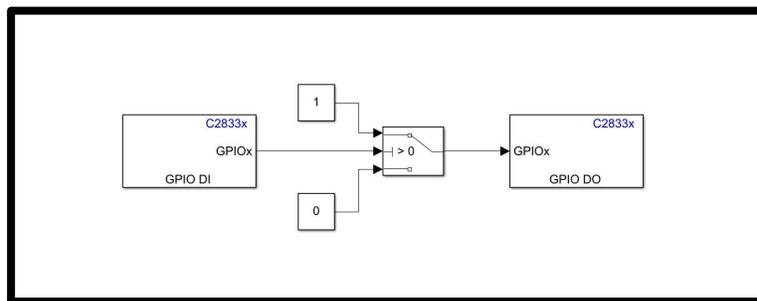
- Abrir: C2833x
 - Localizar: Digital Input como se muestra en la Figura 12.

Figura 12. Inicialización entradas digitales



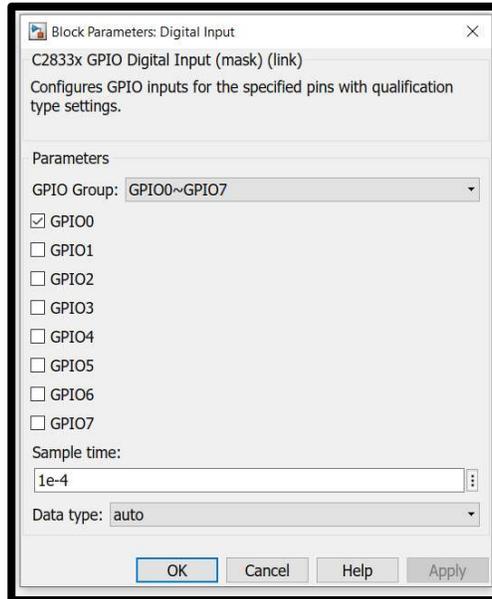
Se propone al estudiante un ejercicio sencillo el cual muestra sin la necesidad de conexión “online” USB y scope la configuración y funcionamiento de las entradas digitales. Hay que recordar que un cero y uno lógico están caracterizados por 0 volts y 3.3 volts, si al dispositivo DSP le llegase un voltaje por arriba de 5 volts, este puede perder funcionalidad y causar la falla total del dispositivo, Figura 13.

Figura 13. Configuración entrada/salida digital simple



En la figura siguiente, se muestra la configuración básica entrada y salida de una señal digital, en donde si la señal de entrada contiene un valor diferente mayor a cero este será igual a un 1 lógico, y arrojará este resultado al GPIO de salida, como se muestra en la Figura 14.

Figura 14. Configuración interna para GIPO de entrada.



Se recomienda se establezca un valor de $1e-4$ para el tiempo de muestreo, se observó que un valor por arriba de este valor de muestreo, el DSP no es capaz de visualizar el valor digital de entrada y omite su funcionamiento. De igual manera que las salidas digitales, se selecciona el GPIO de entrada, se recomienda seleccionar una entrada por cada bloque de programación “digital input” como se muestra en la Figura 15 y Figura 16.

Figura 15. Diagrama de conexión.

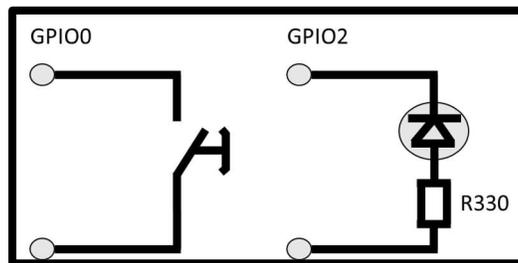
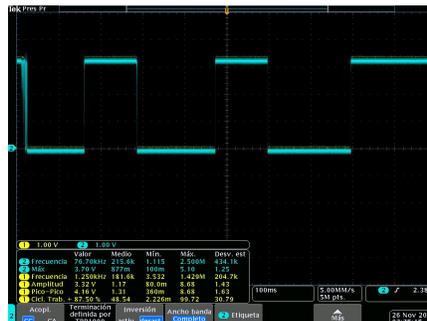


Figura 16. Gráficas de las señales de entrada y salida digitales.



Convertidores ADC Matlab Simulink

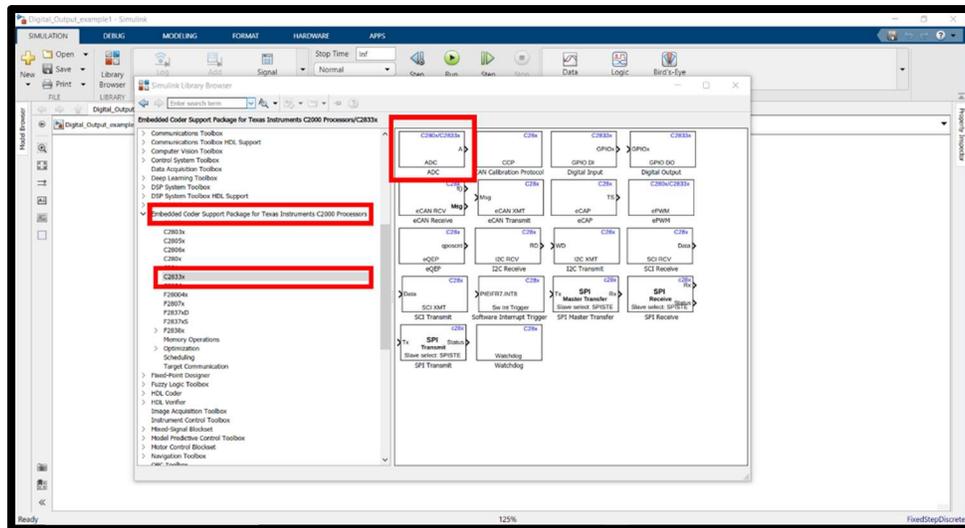
1.- Library Browser

Ubicar: Embedded Coder Support Package for Texas Instruments C2000 Processors

Abrir: C2833x

Localizar: ADC como se muestra en la Figura 17.

Figura 17. Inicialización entrada ADC (Analogic – Digital Converter).

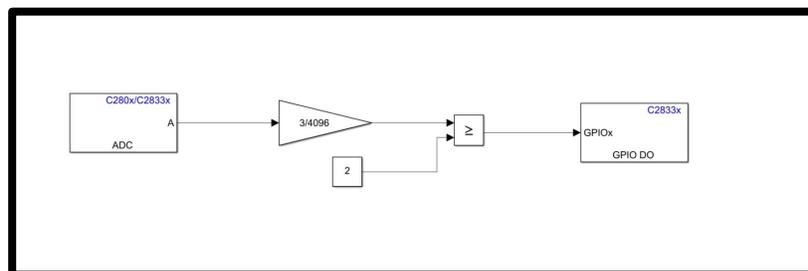


Una vez localizado el módulo ADC para Simulink, en la ventana de configuración tendremos que colocar los siguientes parámetros:

- Selección de módulo: Module A.
- Conversion mode: sequential.
 - Stat of conversion: software.
 - Sample time: 1e-4.
 - Data type: Uint16. (En este apartado se modifica de acuerdo a las necesidades del programador) como se muestra en la Figura 18.

Nota: Se recomienda se utilice el formato de tipo de dato Unit16 ya que proporcionara al ADC una resolución de tipo de valor que representa enteros sin signo con valores que van de 0 a 65535.

Figura 18. Programación de un circuito ADC simple decodificado.



Para la programación propuesta se desarrolla el esquema de la figura anterior, en donde se establece un límite máximo de 3 volts, lo adecuado sería llevarlo a 3.3V, pero debido a la saturación de este por la corriente máxima suministrada, lo limitaremos a 3 volts, dando una corriente nominal de 0.01 Ampers con una resistencia de 330 ohms.

En la programación se observa que se agrega un límite de activación, este límite se establece a un valor de 2 volts; cuando la señal decodificada por la ganancia anterior suministre un valor mayor a 2 volts, este mandara una señal digital de salida y accionara el bloque de salida digital en el GPIO0 como se muestra en la Figura 19 a Figura 21.

Figura 19. Configuración inicial ADCA0.

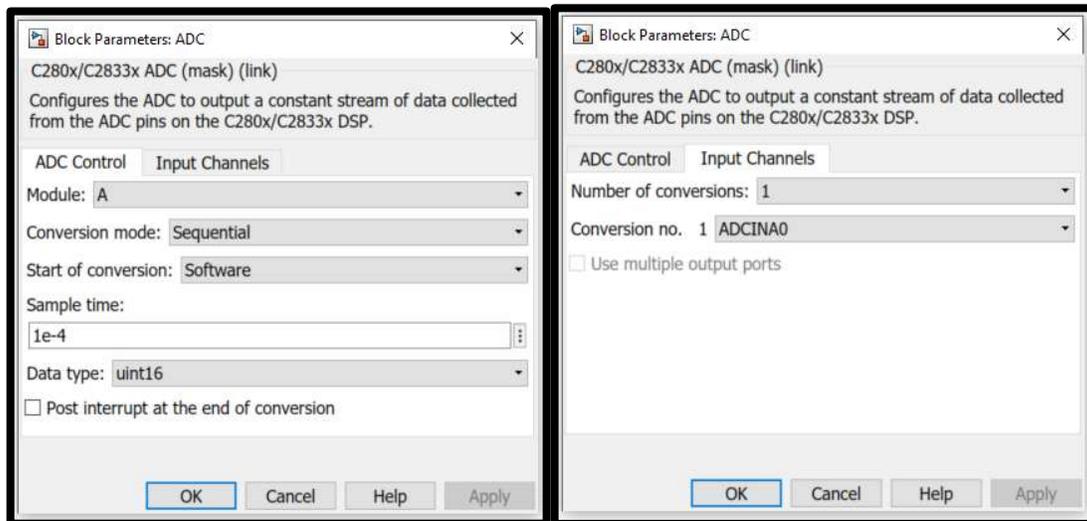


Figura 20. Diagrama de conexión.

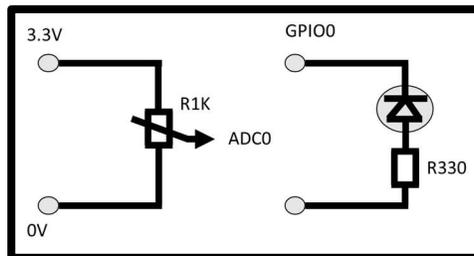
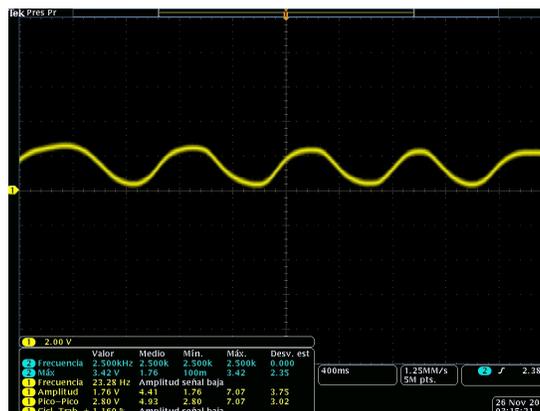


Figura 21. Graficas de respuesta ADC simple.



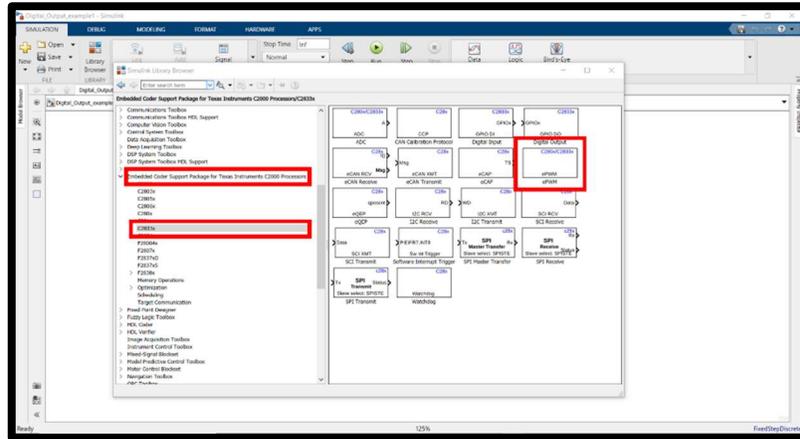
Generadores de señales PWM Matlab Simulink

1.- Library Browser

Ubicar: Embedded Coder Support Package for Texas Instruments C2000 Processors

- Abrir: C2833x
 - Localizar: ePWM como se muestra en la Figura 22.

Figura 22. Selección de bloque ePWM



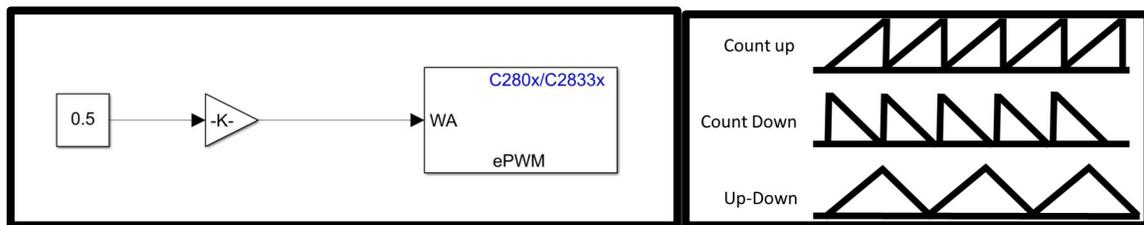
Configuración inicial:

Pestaña General.

- Module: ePWM1
 - Timer period units: clock cycles
 - Specific timer period via: specify via dialog
 - Timer period: 10'000 (con un máximo de 64'000)
 - Reload PRDL: counter equals to zero
 - Counting mode: UP como se muestra en la Figura 23.

En los modos de conteo existen tres parámetros, un conteo ascendente, descendente y up-down, el cual combina ambos modos, el recomendado es el contador ascendente ya que cuenta la unidad en cuanto alcanza el valor determinado.

Figura 23. Configuración simple para ePWM.



En la Figura 23 se recomienda colocar la ganancia K como múltiplo de 10000 o el valor que se haya caracterizado en el valor de “timer period”. Ósea en este ejercicio un $K=10'000$. Y por lo tanto se obtendrá un 50% de su ciclo de trabajo dado que la constante es de 0.5 al iniciar, como se muestra en la Figura 24 a Figura 26.

Figura 24. Configuración simple para ePWM.

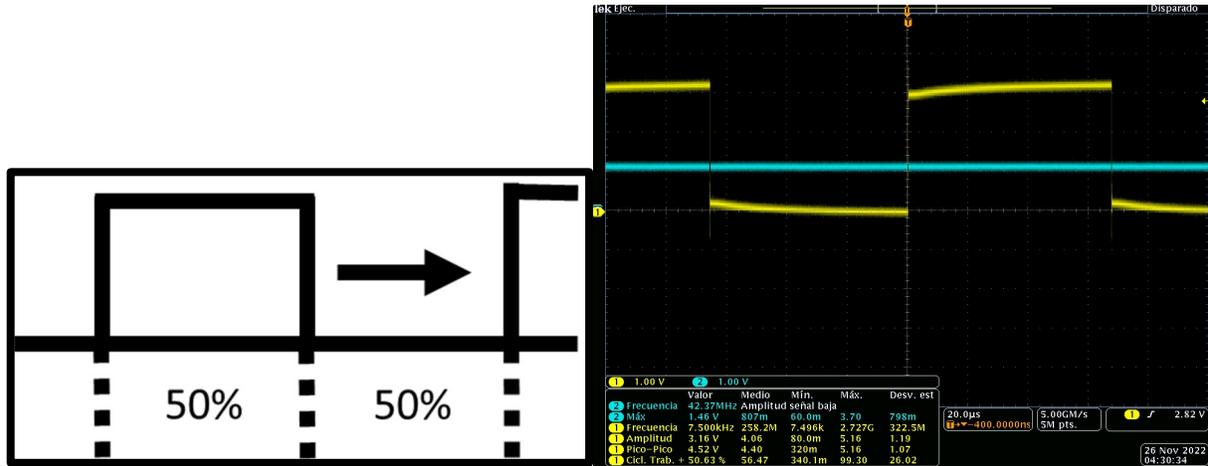


Figura 25. Configuración para ePWM1 en el módulo A.

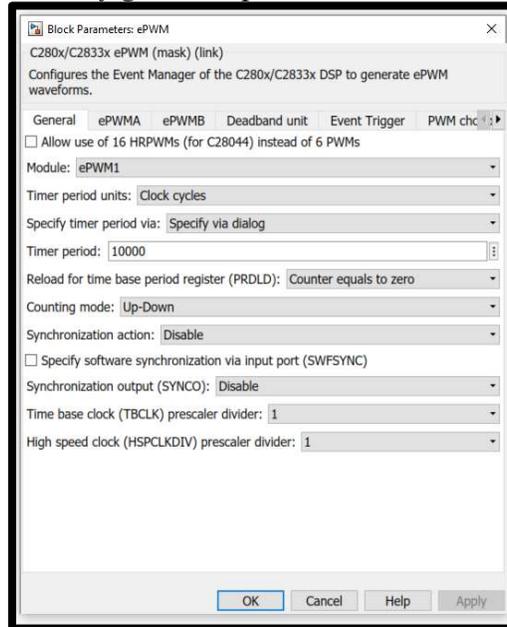
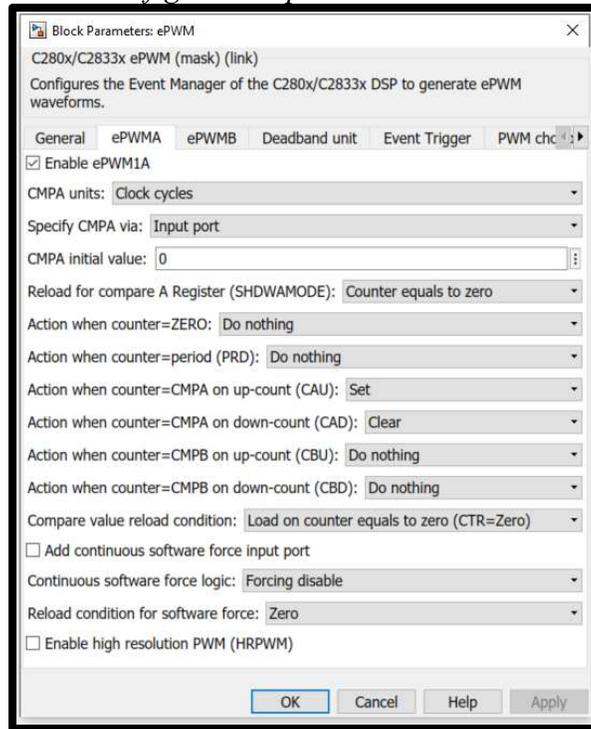


Figura 26. Configuración para ePWM1 en el módulo A.



Para configurar el valor de PWM restante o entero se pueden tener dos combinaciones en los parámetros PRD y CAU. Recaltar que si se busca una combinación entre ePWMA y ePWMB este cambiara de configuración.

Si:

PRD: Clear

CAU:Set

El ciclo CMPA es el restante o sea si se tiene un 0.7 se obtendrá un 30% de su ciclo de trabajo. como se muestra en la Figura 27.

Figura 27. PWM ciclo de trabajo al 30%.



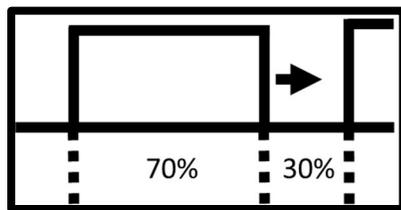
Si:

PRD: Set

CAU: Clear

El ciclo de trabajo entero sería 0.7 y el resultado daría un 70% de su ciclo de trabajo. Como se muestra en la Figura 28.

Figura 28. PWM ciclo de trabajo 70%.



Dependiendo de la configuración inicial, se pueden mover los valores de los ciclos de trabajo enteros y complementarios al 100%. Se recomienda se siga la configuración de las imágenes o el ciclo de trabajo entero como se muestra en el PRD: set y el CAU: clear.

De igual modo se tiene para el modo de UP-DOWN en la pestaña General y counting mode: up-down.

Para un 30% de ciclo de trabajo

- Specify CMPA via: Input port
- CMPA: 0 zero
- Zero: Do nothing
- PDR: Do nothing
- CAU: set
- CAD: Clear (se habilita con el up-down)
- CBU: Do nothing
- CBD: Do nothing (se habilita con el up-down)

Para un 70% de ciclo de trabajo.

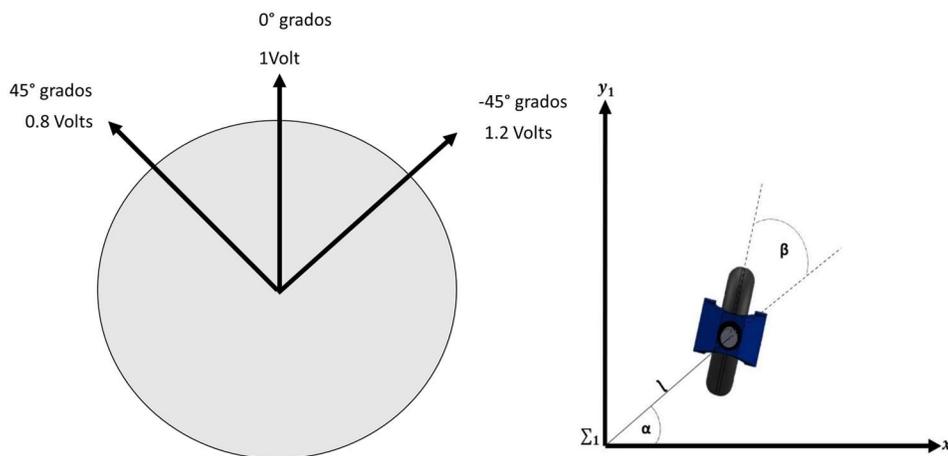
- Specify CMPA via: Input port
- CMPA: 0 zero
- Zero: Do nothing
- PDR: Do nothing

- CAU: Clear
- CAD: Set (se habilita con el up-down)
- CBU: Do nothing
- CBD: Do nothing (se habilita con el up-down)

Caracterización de los sensores

La caracterización del sensor analógico de posición, basado en la detección de magnetismo, se llevó a cabo para evaluar su capacidad de generar señales analógicas precisas en función del ángulo de rotación del eje de dirección del AGV, ver Figura 29. Este sensor utiliza un imán montado en el eje del vehículo que, al girar, altera el campo magnético detectado por el sensor, produciendo una señal analógica que corresponde directamente al ángulo de rotación β y la rotación angular total es definida por α .

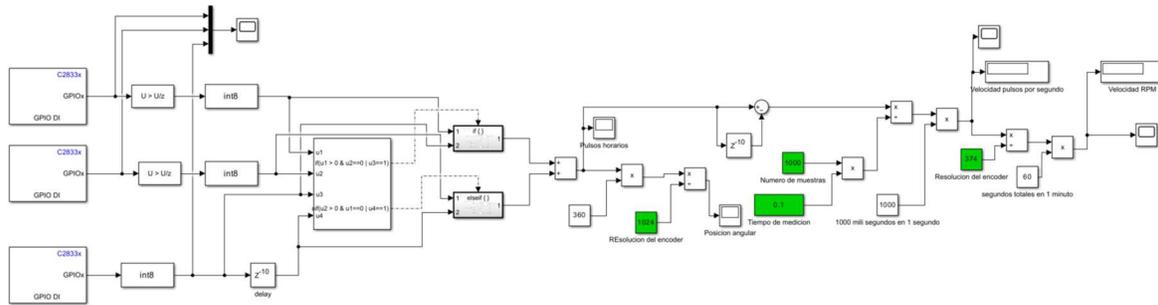
Figura 29. Diagrama de referencia fijo y móvil de rotación para la dirección.



Para asegurar un control preciso de la trayectoria del vehículo, el sensor fue caracterizado en un rango de $\pm 45^\circ$, lo que significa que se evaluó su respuesta tanto a 45° hacia la izquierda como a 45° hacia la derecha desde la posición central. Este rango es crítico para las maniobras del AGV, ya que define los límites dentro de los cuales el vehículo puede ajustar su dirección de manera efectiva. Durante la caracterización, se registraron las señales analógicas correspondientes a diferentes ángulos dentro de este rango, lo que permitió confirmar la linealidad y precisión del sensor en el control de la trayectoria del vehículo.

Además, un encoder incremental es un dispositivo utilizado para medir la posición y velocidad angular de un eje, proporcionando pulsos eléctricos que corresponden a incrementos de movimiento. En este estudio, se realizó la caracterización de un sensor encoder incremental para determinar la velocidad de rotación de la rueda de tracción del AGV en radianes por segundo. El encoder genera una serie de pulsos cuando el eje gira, y al medir la frecuencia de estos pulsos, se puede calcular con precisión la velocidad angular de la rueda. Como se muestra en la Figura 30.

Figura 30. Programación de la estimación de velocidad rotacional de la rueda de tracción del AGV.

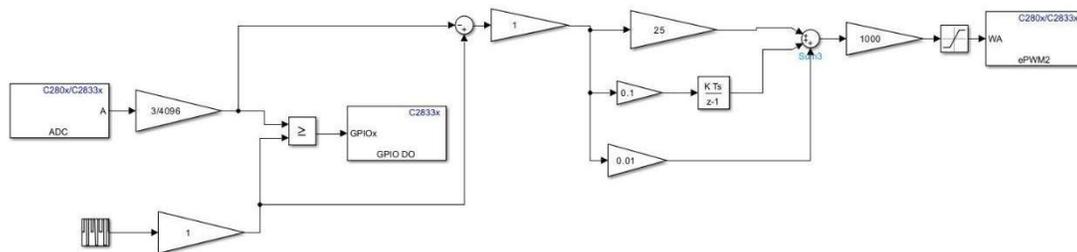


Además, el encoder proporciona dos señales desfasadas entre sí, conocidas como canales A y B. Este desfase es crucial para determinar no solo la velocidad, sino también la dirección de rotación de la rueda. Al analizar la frecuencia y la fase de estas señales, se puede obtener una medida precisa de la velocidad de rotación en radianes por segundo, que es fundamental para el control efectivo del sistema de tracción del AGV. La caracterización del encoder se centró en calibrar y validar la exactitud de estas mediciones en diferentes condiciones operativas.

Diseño controlador PID

En la programación del controlador PID utilizando el software Simulink, se implementó un esquema de control robusto para gestionar la dirección del AGV, ver Figura 31. El controlador PID fue configurado para recibir una señal de retroalimentación proveniente del sensor de posición analógico magnético, el cual proporciona una medida precisa del ángulo de dirección en forma de una señal analógica. Esta señal es utilizada para calcular el error entre la posición deseada y la actual, que luego es procesado por el PID para ajustar la dirección del vehículo.

Figura 31. Diagrama a bloques de la programación de un controlador PID para la aplicación de control de posición del AGV.



Adicionalmente, se configuró un pin específico en el modelo de Simulink para permitir el cambio de dirección del AGV, facilitando la implementación de maniobras complejas. La salida del controlador se traduce en una señal PWM (Modulación por Ancho de Pulso), la cual fue limitada a 10,000 ciclos en su estado de trabajo para asegurar una operación segura y eficiente del sistema. Los parámetros del controlador PID, como se detalla en la Tabla 1, fueron cuidadosamente ajustados para optimizar la respuesta del sistema, proporcionando un equilibrio adecuado entre la rapidez y la estabilidad del control de dirección.

En la programación del controlador PID utilizando el software Simulink, se desarrolló un esquema de control robusto tanto para la dirección como para la velocidad del AGV. Para el control de

dirección, el controlador PID recibe una señal de retroalimentación del sensor de posición analógico magnético, que proporciona una medida precisa del ángulo de dirección en forma de una señal analógica. Esta señal permite calcular el error entre la posición deseada y la actual, que luego es procesado por el PID para ajustar la dirección del vehículo. Además, se configuró un pin específico en Simulink para habilitar el cambio de dirección, y la salida del controlador se traduce en una señal PWM (Modulación por Ancho de Pulso) limitada a 10,000 ciclos en su estado de trabajo, asegurando una operación eficiente del sistema.

Este mismo lazo de control se aplica al control de velocidad del AGV, utilizando el encoder incremental para proporcionar la retroalimentación de la velocidad rotacional en lugar de la señal analógica del sensor de posición. El controlador PID procesa esta retroalimentación para ajustar la velocidad del vehículo, manteniendo un control preciso y estable. Los parámetros del controlador PID, detallados en la Tabla 1, fueron ajustados para optimizar tanto la respuesta de dirección como la de velocidad, garantizando un rendimiento óptimo en ambas funciones críticas del sistema.

Análisis de respuestas de control

La programación del controlador PID fue realizada para gestionar de manera óptima tanto la velocidad de tracción como la posición angular del eje de dirección, asegurando una respuesta rápida y precisa a las demandas del sistema. En esta sección, se presentan los resultados obtenidos a partir de las pruebas experimentales, analizando la respuesta de posición del eje de dirección, ver Figura 32 y velocidad del sistema de tracción, ver Figura 33. Estos resultados permiten evaluar la efectividad de la configuración implementada y ofrecen una visión clara del rendimiento del AGV en condiciones operativas reales.

Tabla 1.

Valores de los parámetros para el control de posición

<i>PID</i>	
k_P	25
k_I	0.1
k_D	2.5

Figura 32. Gráfica de respuesta del control de posición angular AGV (dirección).



Tabla 2.

Tiempos de respuesta de posición angular AGV (dirección).

PID	Valores
t_d (seg)	0.53
t_r (seg)	0.83
M_p (V)	1.51
t_p (seg)	1.87
t_s 2% (seg)	-----
t_s 5% (seg)	3.78

Tabla 3.

Valores de los parámetros para el control de velocidad.

PID	valores
k_p	12.5
k_I	0.1
k_D	0.01

Figura 33. Gráfica de respuesta del control de velocidad angular AGV (tracción).

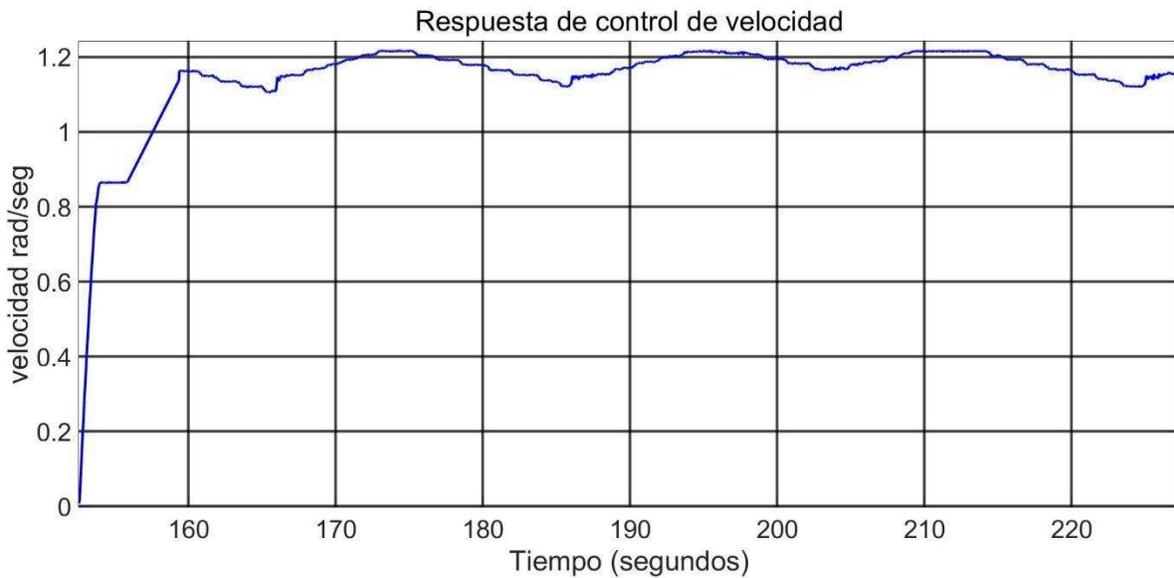


Tabla 4.

Tiempos de respuesta de velocidad angular AGV (velocidad).

PID	Valores
t_d (seg)	1.25
t_r (seg)	9.83
M_p (rad/seg)	10.28
t_p (seg)	10.21
t_s 2% (seg)	22.27
t_s 5% (seg)	12.35

La metodología empleada para el cálculo de los parámetros del controlador PID se basó inicialmente en la herramienta de sintonización de datos de Matlab, utilizando el método de Ziegler-Nichols como punto de partida. Este método proporcionó una estimación preliminar de los valores de los parámetros Proporcional (P), Integral (I) y Derivativo (D), permitiendo establecer un control básico sobre la dirección y velocidad del AGV. Sin embargo, para mejorar el rendimiento en condiciones prácticas, los parámetros obtenidos fueron posteriormente ajustados a través de pruebas experimentales, lo que permitió optimizar la respuesta del sistema en términos de estabilidad, rapidez y precisión. El resultado final de este proceso de ajuste se refleja en la configuración detallada en las tablas correspondientes, donde se especifican los valores definitivos de los parámetros PID tanto para el control de posición como para la velocidad rotacional del vehículo.

Discusión

Los resultados obtenidos en las Tablas 1 a 4, se destaca la efectividad de la metodología aplicada para el control de la dirección y velocidad del AGV tipo triciclo. La implementación del controlador PID, con parámetros ajustados inicialmente mediante el método de Ziegler-Nichols y refinados a través de pruebas experimentales, demostró ser eficaz para mantener la estabilidad y precisión en ambos sistemas de control.

Los datos recopilados y mostrados en la Tabla 2 a partir de las pruebas revelan que la respuesta del sistema de dirección, controlada mediante la señal analógica del sensor magnético de posición, fue rápida y con mínimos overshoots, lo que confirma la correcta sintonización de los parámetros PID. De manera similar, la respuesta de velocidad, monitoreada y presentada en la Tabla 4 a través del encoder incremental, mostró un control preciso y estable, alcanzando la velocidad deseada en un tiempo óptimo sin comprometer la estabilidad del sistema.

Además, la simplicidad en la configuración del sistema, con la eliminación de acoplamientos mecánicos adicionales al integrar el motor directamente en el eje de tracción y dirección, no solo mejoró el rendimiento en términos de torque, sino que también facilitó el proceso de control y ajuste de los parámetros PID. Esta simplificación es reflejada en las gráficas de respuesta de posición y velocidad, donde se observa una clara correspondencia entre las señales de entrada y salida, evidenciando la robustez del sistema bajo diferentes condiciones operativas.

En conclusión, los resultados subrayan la importancia de un enfoque simplificado y bien sintonizado en el diseño y control de AGVs, donde la correcta integración de sensores y motores, junto con un controlador PID adecuadamente ajustado, permite alcanzar un rendimiento superior

en aplicaciones prácticas. Las gráficas obtenidas confirman que la configuración propuesta logra un equilibrio óptimo entre simplicidad y eficacia, asegurando un control fiable tanto en la dirección como en la velocidad del AGV.

Referencias

- Miramontes, K., Solano, J., & Barrera, J. (2021). Simulación de un control de trayectoria para un vehículo guiado autónomamente tipo triciclo con tracción delantera. *Universo de la Tecnológica*, 2(38), 15-20.
- Åström, K. J., & Murray, R. M. (2010). *Feedback systems: An introduction for scientists and engineers*. Princeton University Press.
- Ogata, K. (2010). *Modern control engineering* (5ª ed.). Pearson.
- Nise, N. S. (2020). *Control systems engineering* (8ª ed.). Wiley.
- Maxon Motor AG. (2015). *Maxon motor control: Driving and programming*. Maxon Motor AG.
- Ziegler, J. G., & Nichols, N. B. (1942). Optimum settings for automatic controllers. *Transactions of the ASME*, 64(11), 759–768.
- Guzmán, J. L., & Berenguel, M. (2019). *Control PID y su aplicación en la industria*. Alfaomega.
- Seborg, D. E., Edgar, T. F., Mellichamp, D. A., & Doyle, F. J. III. (2010). *Process dynamics and control* (3ª ed.). Wiley.
- Ollero, A., & Heredia, G. (1995). *Control de robots: Manipuladores y robots móviles*. McGraw-Hill.
- Paden, B., Čáp, M., Yong, S. Z., Yershov, D., & Frazzoli, E. (2016). A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on Intelligent Vehicles*, 1(1), 33–55. <https://doi.org/10.1109/TIV.2016.2578706>
- Slotine, J. J. E., & Li, W. (1991). *Applied nonlinear control*. Prentice Hall.
- Karris, S. T. (2019). *Introduction to Simulink with engineering applications* (4ª ed.). Orchard Publications.
- Attaway, S. (2022). *MATLAB: A practical introduction to programming and problem solving* (6ª ed.). Butterworth-Heinemann.