

1. Arquitectura de la solución

1.1 Requerimientos

Se desea controlar la rapidez del movimiento generado por un motor DC. Esto debe hacerse mediante una interfaz que permita al usuario determinar la rapidez del movimiento del eje, según sea la necesidad del mismo.

1.2 Solución propuesta

Para la comunicación usuario-motor se propone un sistema que establezca un control FeedForward, cuya entrada sea el valor de rapidez lineal deseada por el usuario, y sus salidas sean la respuesta física del motor y el valor medido, por el sistema mismo, de dicho comportamiento, el cual se visualice en la interfaz.

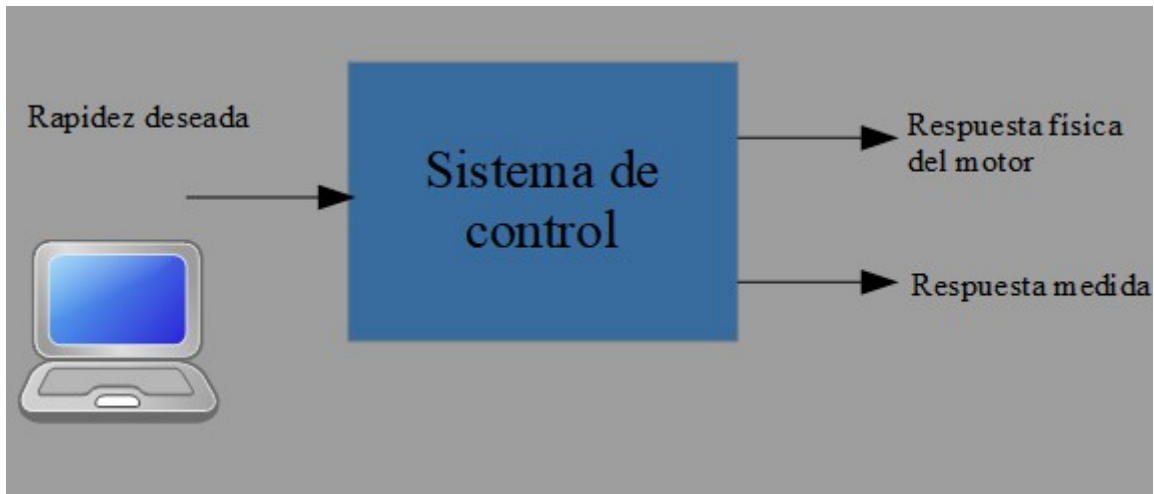


Figura 1. Arquitectura nivel 0.

El sistema de control, descrito como componente en la Figura 1, se compone de la siguiente manera: Unidad de procesamiento, temporizador, módulo encoder, generador señal PWM, módulo GPIO, una GUI en una aplicación de computador y un motor DC. La interconexión entre estos componentes debe ser como se muestra en la Figura 2.

La entrada suministrada al computador se trasmite a la Unidad de procesamiento, la cual con un conjunto de instrucciones ordena al generador de señal PWM un ciclo de trabajo, que varía la tensión promedio suministrada al motor DC. Después de esto, la Unidad de procesamiento en conjunto con el temporizador y el módulo encoder se encargarán de medir la respuesta del motor, la cual se transmite a la aplicación de computador, donde se mostrará la relación funcional tiempo-rapidez en una gráfica.

El módulo motor DC tiene incorporado el propio motor DC junto a un circuito de control, el

cual se encarga de que el motor responda a la señal PWM.

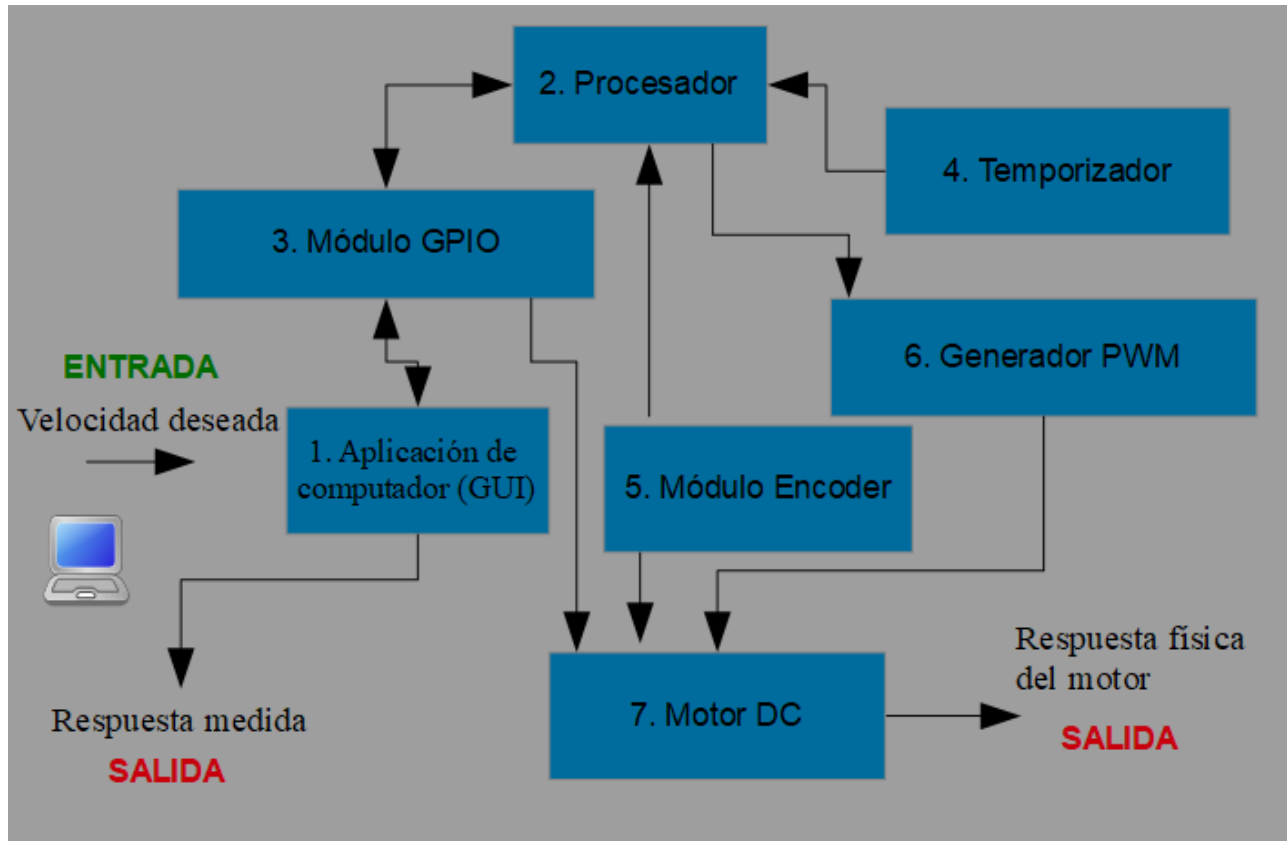


Figura 2. Arquitectura nivel 1

2. Programación del sistema de control

2.1 Inicialización de componentes

Procesador: Contiene un firmware que recibe la velocidad deseada por el usuario y el número de pulsos leídos por el sensor de movimiento del motor DC (Encoder). Al recibir dicha velocidad, el procesador interpreta este valor para así programar una señal PWM a través del generador de PWM, y también determina los valores de salida de los pines digitales del GPIO que van conectados directamente al módulo motor DC. Esta señal hace que el motor DC responda a un valor de tensión promedio determinado. El procesador conoce cuántos pulsos equivale a una vuelta del eje del motor, por tanto, en conjunto con el temporizador calcula la rapidez a la que va dicho eje. Cada vez que el procesador calcule una velocidad, éste lo manda nuevamente a la aplicación. Este proceso se repite siempre mientras el sistema esté encendido.

Aplicación de computador (GUI): Proporciona una interfaz amigable al usuario para el control FeedForward. Cuando el computador se conecte a la unidad de procesamiento, debe aparecer un cuadro de texto, en donde el usuario digite la velocidad deseada. Al momento de digitarse, se encuentra un botón de enviar. En el evento de enviar, el computador transmite este dato, e inmediatamente pasa al evento de registrar la medición de velocidad. A partir de acá, se muestra la gráfica de la secuencia experimental velocidad/tiempo. Esta secuencia debe pasar por un filtro digital incorporado en el programa antes de ser graficada.

Motor DC: Contiene un controlador de motor y el motor mismo. El controlador se encarga de

recibir la señal PWM y hace que el motor responda a la tensión promedio determinada por el ciclo de trabajo de dicha señal. Este control también tiene integrado un puente H, el cual a partir del valor de los dos pines del GPIO controlados por el procesador, cambia el sentido de giro del motor y también lo detiene directamente.

Temporizador: Módulo que genere una señal de reloj a una frecuencia eficiente para el cálculo de la rapidez.

Generador PWM: Módulo que genere una señal PWM a una determinada frecuencia, su ciclo de trabajo debe programarse a partir del procesador.

Módulo Encoder: Contiene un sensor de movimiento incremental integrado físicamente al motor y un contador, el cual sea capaz de interpretar el comportamiento de las dos señales de salida del sensor. El contador debe incrementar de 1-1 la cantidad de pulsos cuando detecte un sentido de giro positivo (sentido antihorario), y decrementar de la misma manera cuando detecte un sentido negativo (sentido horario).

Módulo GPIO: Circuitos para pines de salida o entradas que se interconectan con el procesador, para conectarlo con periféricos.

2.2. Instrucciones para el procesador

En la Figura 3 se muestra de manera general las instrucciones que debe ejecutar el procesador en función de las salidas de los módulos al que este se encuentra interconectado, y las acciones que debe hacer sobre éstos.

El firmware del procesador en este caso se divide en dos, el bloque de instrucciones principal y un bloque de manejador de interrupción que se propone para calcular la velocidad del motor.

El bloque de instrucciones principal consta de un ciclo mientras el sistema esté encendido, cuyo propósito es el de cambiar la variable de variable deseada, interpretarla y modificar la respuesta física del motor a través del controlador del motor DC; la velocidad a través del generador de PWM, y el sentido de giro a través de los valores de salida de los pines del GPIO a los que está conectado dicho controlador. Finalmente se transmite la velocidad medida, que será una variable cuyo dato guardado será modificado en el segundo bloque.

El segundo bloque se ejecuta al momento que ha pasado un cierto tiempo en el temporizador. En el manejador se toma el tiempo transcurrido y el cambio en los pulsos que ha entregado el encoder, para así calcular la velocidad. Dicho tiempo debe ser mayor a lo que tarda una iteración dentro del bloque principal de instrucciones. El dato calculado se guarda finalmente en la variable (de tipo volatile en lenguaje C) `vel_medida`. El valor de pulsos anterior y actual se guardarán también en variables de este tipo, para que sean utilizados cada vez que se acceda al manejador de la interrupción, al iniciar por primera vez estos valores deben iniciar en ceros por defecto.

En la Figura 3 y Figura 4 se muestran la descripción general de estos bloques de códigos de manera gráfica, diagramas de flujo.

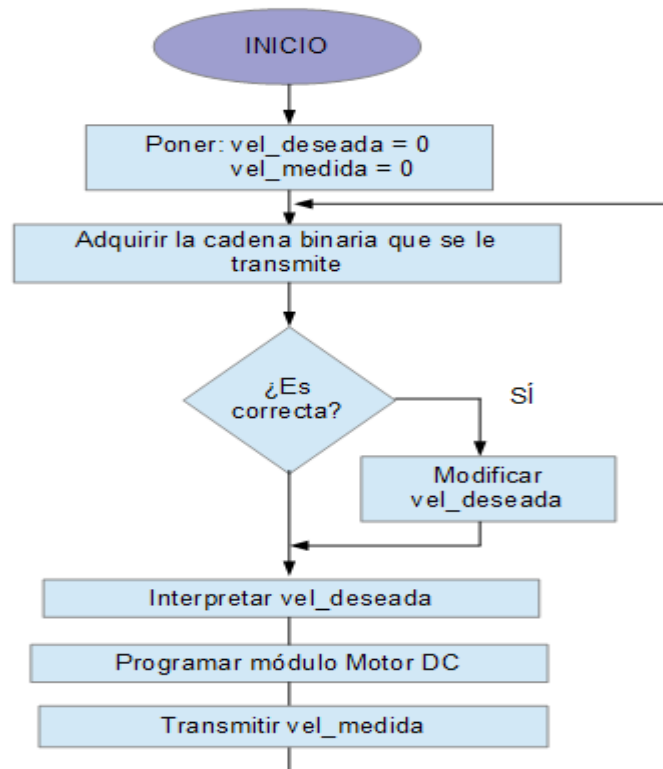


Figura 3. Diagrama de flujo principal

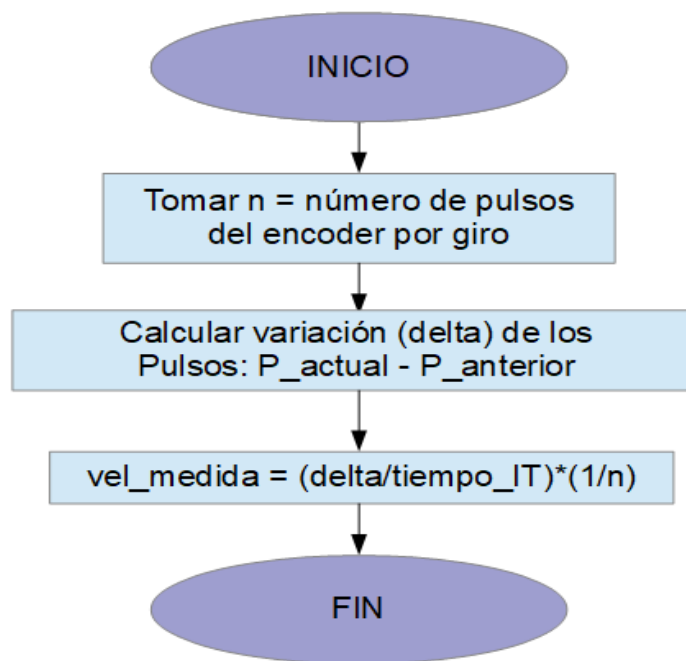


Figura 4. Diagrama de flujo manejador IT

3. Implementación

Para esta práctica de laboratorio la solución propuesta se implementará en un sistema embebido, se hará uso de un microcontrolador STM32F411CEU06, el cual contiene los siguientes

módulos: GPIO, temporizador, ADC, procesador. El módulo IR contiene cuatro de los circuitos descritos anteriormente dispuestos de una determinada forma. Finalmente, la aplicación de computador se hará mediante el framework Qt.

Se realizaron dos librerías en el IDE de STM32Cube, una correspondiente con la transmisión de datos por USB, y otra para el control de motor DC. Ambas proporcionan funciones que cumplen con las descritas en el diagrama de flujo.

4. Conclusiones

Finalmente, se logró implementar un sistema embebido que cumple con los requerimientos del inicio. Esto permitió al grupo conocer y entender el funcionamiento de un encoder de cuadratura y aprender a utilizarlo para futuros proyectos.