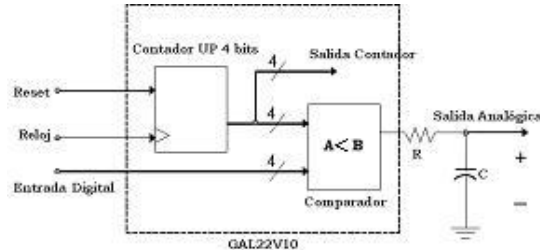


Teoría PWM (Modulación por Ancho de Pulsos)

Por lo general, todo sistema que procesa información binaria para controlar un proceso analógico requiere una etapa de entrada analógico – digital y una etapa de salida digital –analógica (convertidores ADC y DAC). Para reducir costos en los diseños que no requieren alta resolución en la etapa de salida, es posible sustituir el DAC por un algoritmo de Modulación por Ancho de Pulsos (**PWM** - Pulse-Width Modulation). Una *unidad PWM* permite asignar cierta duración de tiempo en alto o en bajo a un dato digital de n bits que se considera salida de la etapa procesadora. Lo anterior se logra conectando un contador y un circuito comparador, tal y como se aprecia en la siguiente figura:



El comparador determinará si el dato aplicado a la entrada de la unidad es igual al valor binario del contador que cambia constantemente. El tiempo que durará la señal en alto depende de la cantidad de pulsos de reloj que se apliquen hasta que el contador presente un dato binario mayor o igual al de la entrada. A la salida de la unidad PWM es necesario conectar un filtro RC (Pasa Bajas) para determinar el nivel analógico propuesto por el filtro. El periodo completo de un ciclo PWM es igual al producto del periodo del reloj de la señal de referencia (reloj del sistema) con 2^n , donde n es el número de bits del contador propuesto. Obsérvese la relación siguiente:

$$T_{PWM} = (T_{reloj})(2^n) \quad (\text{eq. 1})$$

Es posible advertir que si el contador es de 4 bits, se tendrá que $T_{PWM} = 16 T_{reloj}$, por lo que al aumentar el número de bits del contador para mejorar la resolución, el periodo PWM tenderá a hacerse más grande reduciendo drásticamente la frecuencia de salida. Lo anterior es sumamente importante cuando se desee realizar una aplicación que sí requiera sincronizar la señal analógica para controlar un sistema.

Para los propósitos del diseño planteado, considérese una frecuencia de referencia de 60 Hz, con un contador de 4 bits, y sustituyendo en *eq. 1*, se obtiene una frecuencia PWM de 3.75 Hz, equivalente a un T_{PWM} de 0.26 seg.

La relación de los valores del filtro está dada por:

$$RC = \frac{1}{2\pi F_{PWM}} \quad (\text{eq. 2}),$$

por lo que eligiendo $C = 1\mu\text{F}$, se obtiene un valor para R de aproximadamente 42 K Ω (es posible ajustar a un valor comercial más exacto). En nuestro caso particular, se diseñará una unidad PWM de frecuencia baja por lo que no es necesario considerar aspectos de diseño más formales en el filtro. En diseños formales se recomienda estimar una frecuencia de corte menor a la frecuencia PWM y no es común considerarlas iguales, tal y como se hizo en la aproximación anterior. En tal situación será necesario sustituir la frecuencia PWM por la frecuencia de corte en la *eq. 2* ó tal vez, hasta considerar un filtro de mayor orden que permita una mejor estabilidad de la señal.

Implementa tu diseño en un GAL22V10 y mide los voltajes analógicos obtenidos en la salida. Un valor 0000₂ debe estar en los 0 Volts, mientras que un valor 1111₂ debe concebir una salida de 5 Volts. Para mejorar el diseño es necesario incluir una etapa final de amplificación (un transistor o un OPAM) a la salida.

Control de Motores a Pasos Unipolares

Un motor a pasos (*stepper motor*, traducido también al castellano como motor paso a paso) tiene la propiedad de moverse de un paso a otro, por cada pulso de reloj que se le aplique. Así, puede realizar 15 pasos en un mismo sentido, si se le aplican 15 pulsos de reloj. Dependiendo de las características del motor, es posible tener pasos muy pequeños (por ejemplo de 1.8°, por lo que después de 200 pulsos completará una vuelta a razón de 1.8° x

200 = 360°) o pasos muy grandes (por ejemplo de 90°, completando una vuelta con 4 pulsos a razón de 90° x 4 = 360°).

Como se advierte, un motor a pasos es muy socorrido en diseños que requiere posicionar con exactitud el rotor. Lo anterior resulta difícil en los motores de CD, debido a que estos giran libremente al aplicar un voltaje y si se desea detener el rotor es necesario retirar el voltaje de alimentación, lo cual no garantiza que el rotor se detendrá en una posición predefinida. De cualquier modo, existen técnicas que controlan la duración de pulsos de reloj que pueden hacer que un motor a CD se posicione de manera exacta, como sucede con los Servo Motores.

Principio de Funcionamiento

Existen motores a pasos de reluctancia variable y de imán permanente, siendo los segundos los que comúnmente se utilizan para aplicaciones didácticas de robótica. Este tutorial hace referencia a los de imán permanente.

Por lo general, los motores a pasos se clasifican en tres tipos: Unipolares, Bipolares y Multifase. En esta primera parte del documento, nos enfocaremos sólo al motor unipolar.

Un motor a pasos consta de dos partes principales: el rotor y el estator (**Figura 1**). El rotor es la parte central del motor conformada por un imán permanente que gira debido a que el estator tiene bobinas que cuando se excitan adecuadamente generan un campo electromagnético que produce el movimiento del imán en alguna dirección. Lo anterior indica que para que el motor dé un paso, basta excitar la(s) bobina(s) correspondiente(s).

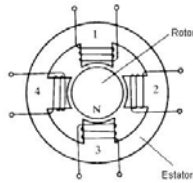


Figura 1. Motor a Pasos de 4 Fases.

El motor unipolar convencional es de 4 fases o bien, 2 bobinas individuales donde cada una de ellas está separada por un tap central (punto común); de ahí que se consideren 4 bobinas en vez de dos (**Figura 2**). Cada bobina tiene asociados dos cables para la alimentación tanto positiva como negativa (se polarizan de manera similar a un LED); sin embargo el tap central es común a todas, por lo que físicamente sólo vemos 5 cables: el común y los cables individuales para cerrar la alimentación de las 4 bobinas.

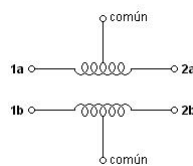


Figura 2. Distribución de las Bobinas de un Motor Unipolar de 4 Fases.

Es importante que notes que si decides conectar el común a GND, excitarás las bobinas con V_m (voltaje nominal del motor). Si por el contrario, conectas el común a V_m , necesitarás excitarlas con GND. A final de cuentas se trata de polarizar correctamente cada bobina.

Consideremos el caso en que se conecta el común a GND y las bobinas se excitan con V_m . De acuerdo a las características eléctricas del motor (12 V, 150 mA, 75 Ω en cada bobina), con el cable negro como **común**, el cable blanco y el verde conformando **1a y 2a** respectivamente, así como el cable rojo y el cable café como **1b y 2b**, es posible sugerir la **Figura 3**:

En la figura anterior se interpreta que para $a1=1$, $b1=0$, $a2=0$ y $b2=0$, el rotor apuntará hacia la bobina **a1**, debido a que es la única que está excitada. Si se desea que el rotor aparezca en la posición marcada por la **Figura 4**, es necesario aplicar $a1=0$, $b1=0$, $a2=1$ y $b2=0$.

La secuencia debe llevar obligatoriamente un orden, se sobreentiende que el rotor no puede pasar instantáneamente de la posición de la primera figura a la posición de la segunda sin haber recorrido las posiciones previas, sin importar el sentido del giro.

Opcionalmente, el rotor puede adquirir una posición intermedia, si se excitan dos bobinas coincidentes (cercanas) al mismo tiempo (**Figura 5**). Lo anterior implica un mayor torque, al igual que una mayor demanda de corriente (lo doble). El estado aplicado es $a1=0, b1=1, a2=1$ y $b2=0$.

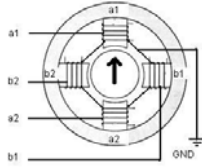


Figura 3. Motor a Pasos de 4 Fases, con Conexiones Propuestas.

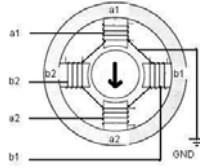


Figura 4. Rotor colocado hacia el sur.

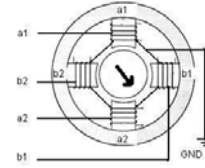


Figura 5. Excitación Simultánea de Dos Bobinas.

Se tienen tres posible secuencias a seguir: *Paso Completo (Wave Drive)*, *Paso Completo (Normal)* y *Medio Paso*. Para la primera secuencia de movimiento se plantea la siguiente tabla, excitando de manera individual cada bobina.

Estado	a1	b1	a2	b2
0	1	0	0	0
1	0	1	0	0
2	0	0	1	0
3	0	0	0	1

Se observa que la secuencia implica cuatro estados, donde cada estado entra con un pulso de reloj. El conteo será ascendente para un giro en el sentido de las manecillas del reloj y será descendente en sentido antihorario. La variable que permita cambiar el sentido del giro se declara como parte de una maquina de estados de manera muy similar a un contador UP/DOWN.

Para la segunda secuencia se respeta la siguiente tabla, excitando dos bobinas a la vez.

Estado	a1	b1	A2	b2
0	1	1	0	0
1	0	1	1	0
2	0	0	1	1
3	1	0	0	1

La tercera secuencia, permite dividir los pasos a la mitad e implica la siguiente tabla:

Estado	a1	b1	a2	b2
0	1	0	0	0
1	1	1	0	0
2	0	1	0	0
3	0	1	1	0
4	0	0	1	0
5	0	0	1	1
6	0	0	0	1
7	1	0	0	1

Obsérvese que se trata de una máquina de 8 estados, donde cada estado representa un paso.

Diseño de un Inversor de Giro

Un código aproximado en ABEL (dirigido a un GAL16V8) para resolver esta última secuencia se lista a continuación. En la máquina de estados se ha incluido un estado inicial para garantizar el inicio del conteo (E0 = 0, 0, 0, 0), por lo que los estados reales de la secuencia se desplazan una posición. La variable *dir*, permite cambiar el giro del motor.

```

MODULE mi_motor
"el cable blanco es a1, el rojo es b1, el verde
"a2 y el café b2. El cable negro es el "común.

"entradas
reloj,dir pin 1,2;

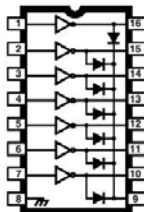
"salidas secuenciales registradas
a1,b1,a2,b2 pin 14,15,16,17 istype'reg,dc';

"declaración de set
sreg=[a1,b1,a2,b2];
E0=[0,0,0,0];
E1=[1,0,0,0];
E2=[1,1,0,0];
E3=[0,1,0,0];
E4=[0,1,1,0];
E5=[0,0,1,0];
E6=[0,0,1,1];
E7=[0,0,0,1];
E8=[1,0,0,1];

equations
sreg.clk=reloj;

state_diagram sreg
state E0:
IF dir THEN E1 ELSE E8;
state E1:
IF dir THEN E2 ELSE E0;
state E2:
IF dir THEN E3 ELSE E1;
state E3:
IF dir THEN E4 ELSE E2;
state E4:
IF dir THEN E5 ELSE E3;
state E5:
IF dir THEN E6 ELSE E4;
state E6:
IF dir THEN E7 ELSE E5;
state E7:
IF dir THEN E8 ELSE E6;
state E8:
IF dir THEN E1 ELSE E7;
END
    
```

El circuito de control no sólo requiere el GAL programado. Hay que recordar que el motor necesita un voltaje de 12 Volts y una corriente de 150 mA por bobina. El GAL no entrega estos valores nominales por lo que es necesario agregar una etapa de potencia a su salida, para amplificar la corriente. Así mismo, se debe considerar un voltaje de 5 Volts para el GAL y uno de 12 Volts para el motor. La etapa de potencia (driver) se puede implementar con un integrado ULN2003 que es un driver monolítico que entrega hasta 500mA. El ULN2003 presenta la característica de ser un driver con trabajo invertido, es decir, complementará los datos entrantes. En realidad no existe problema alguno, por que sólo basta con conectar el común a los 12 Volts nominales y conservar la misma tabla de secuencias establecida, ya que un "1" al entrar al driver se convertirá en un "0".



ULN2003

Finalmente el circuito a armar es el siguiente. El diodo zener debe ser de 12 Volts.

