

Capítulo 3

Sistemas de control

3.1 Control Simple Proporcional

Al tener los programas para cada uno de los sensores y los 4 motores es necesario tratar de unificarlos en un solo programa principal que pueda realizar las 3 funciones que se necesitan para esta tesis: controlar los 4 motores, detectar la distancia del vehículo hacia el suelo y detectar la aceleración para compensar la velocidad de los motores.

La parte más fácil de controlar fue el sensor ultrasónico ya que este funciona sin tantas complicaciones y es sencillo incorporarlo a otro programa, en este caso se incorporo fácilmente al programa de velocidad de los 4 cuatro motores, el cual era el segundo en nivel de organización y facilidad programación. Dada esta facilidad se decidió trabajar con el programa del Nunchuk el cual es el más complejo de los 3 porque se tiene que controlar un dispositivo que entrega señales digitales.

Teniendo el programa anterior para controlar el Nunchuk y saber la inclinación que presenta el acelerómetro a diferentes ángulos solo bastaba con realizar una acción de control que hiciera que cuando uno de los ejes en los que se encuentra un motor este por debajo de un nivel estable este tenga un incremento en su velocidad para mantener el dispositivo en un nivel de equilibrio.

Para esto se decidió dejar el funcionamiento de los motores a un 94% de su velocidad total para que el 6% restante sirviera como incremento, esto nos deja con un valor de 240 de 255 en el valor de analogWrite() para el valor de PWM de los motores.

Teniendo este 6% de PWM se analizaron los grados que se habían tomado en cuenta de inclinación que son valores entre 15° y 17° dependiendo de cada eje del acelerómetro. Los primeros 4 valores de cambio correspondientes al nivel de PWM registrados en el acelerómetro se dejaron para tener una base de estabilidad de aproximadamente 3.4% por cada eje.

Tabla 1: Grados de inclinación respecto a incremento PWM eje X

PWM	X>50°	X<50°
+1	0°	0°
+2	1°	0.3°
+3	1.7°	1.8°
+4	2.8°	2.7°

Tabla 2: Grados de inclinación respecto a incremento PWM eje Y

PWM2	Y<50°	Y>50°
1	0°	0°
2	2.1°	2.6°
3	2.4°	3.8°
4	3.3°	4.7°

Tomando en cuenta las consideraciones anteriores los 15 valores correspondientes a PWM se dividieron entre los 11 valores de grados de inclinación que se tienen registrados para cada uno de los ejes dando los siguientes incrementos: 1, 3, 4, 6, 7, 8, 10, 11, 12, 14, 15; los cuales sumados a los 240 dan un total de 255 que es el valor máximo de PWM y representa el motor trabajando al 100%.

Tabla 3: Incremento de PWM en motores del Eje X

X<50°			x>50°		
Inclinación	PWM	incremento-	Inclinacion2	PWM3	incremento+
4.5°	46	47	4.7°	54	55
5.5°	45	48	5.9°	55	58
6.5°	44	48	6.9°	56	60
7.5°	43	49	7.9°	57	63
8.5°	42	49	8.9°	58	65
9.5°	41	49	9.9°	59	67
10.5°	40	50	10.9°	60	70
12.6°	39	51	12.9°	61	73
13.9°	38	50	13.9°	62	74
14.8°	37	51	15.3	63	77
15.9°	36	51	16.3	64	79

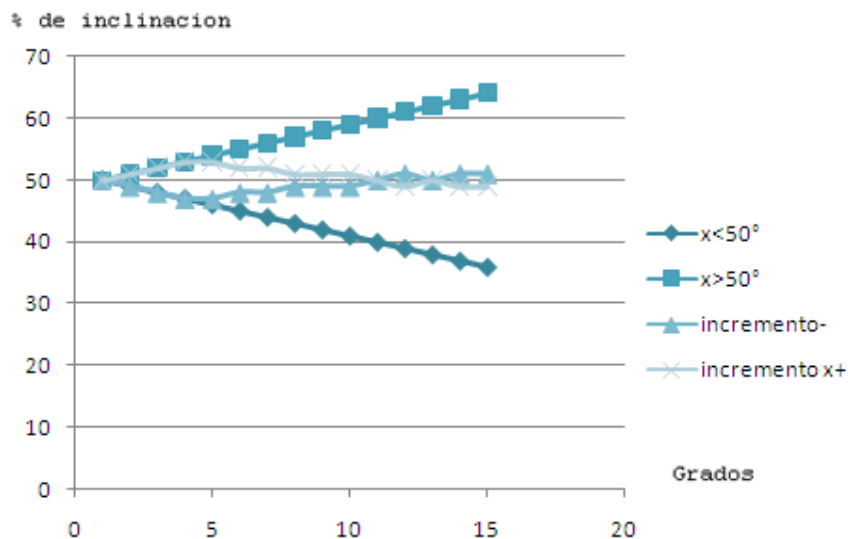


Figura 3. 1: Grafica de incremento de PWM en motores eje X.

Gracias a este pequeño sistema de control los 2 motores que se encuentran en el eje X tienen a regresar a su posición estable mediante el incremento en PWM que cada uno de ellos presenta.

Tabla 4: Incremento de PWM en motores del Eje Y.

Y<50°			Y>50°		
Inclinación	PWM	incremento-	Inclinacion2	PWM3	incremento+
5.8°	46	47	3.9°	54	55
6.8°	45	48	4.8°	55	58
7.9°	44	48	5.9°	56	60
8.9°	43	49	6.9°	57	63
9.9°	42	49	7.9°	58	65
11.1°	41	49	8.9°	59	67
12.2°	40	50	9.9°	60	70
14.3°	39	51	12.3°	61	73
15.5	38	50	13.3°	62	74
16.5	37	51	14.3°	63	77
17.6	36	51	15.7°	64	79

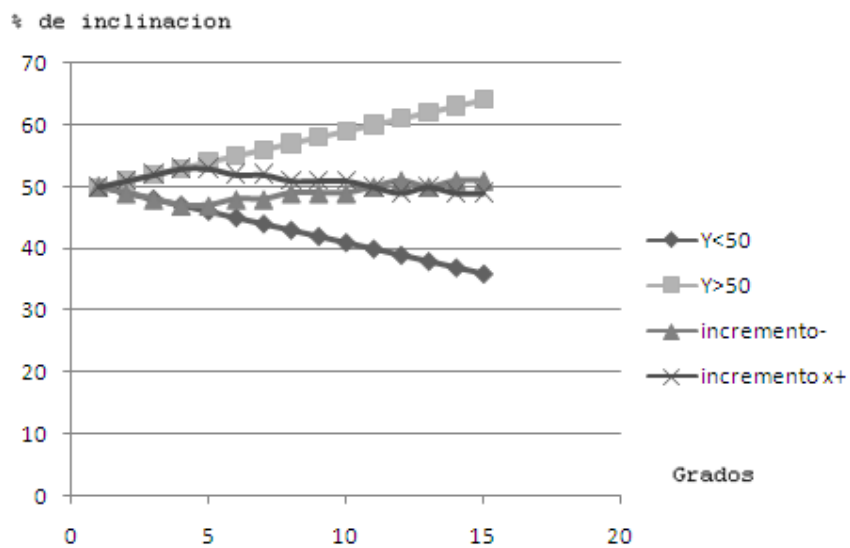


Figura 3. 2: Grafica de incremento de PWM en motores eje Y.

Los grados tomados en cuenta para la inclinación representan un nivel aceptable en términos de inclinación manejable que los motores pueden soportar, de no ser así y en caso de presentarse inclinaciones mayores significa que hay algo que no está funcionando bien en el vehículo ya sean los motores, las hélices o el acelerómetro.

Para acoplar estos datos en el programa anterior solo fue necesario añadir una pequeña sección que permitiera agregar velocidad a los motores después de una lectura de la inclinación.

```
void loop()
{
  // Cuando el contador alcance 25 leer el Nunchuk
  t++;
  if( t == 25 )
  {

    inclinacion();
    aumentoVelocidad();
    Serial.println();
    Serial.print("xa: "); Serial.print(xa);
    Serial.print("\txb: "); Serial.print(xb);
    Serial.print("\txc: "); Serial.print(xc);
    Serial.print("\txd: "); Serial.println(xd);
    rstInclinacion();

  }

  delay(20);
}
```

En el código anterior se muestra como se hizo la integración del programa anterior que detecta la inclinación que recibe el acelerómetro y posterior a eso se realiza el incremento de velocidad en los motores con el siguiente código:

```

void aumentoVelocidad()
{
    int x,y;

    x=xx2;
    y=yy2;

    if (x>53)
    {

        if(x==54)
            xa=1;
        if(x==55)
            xa=3;
        if(x==56)
            xa=4;
        if(x==57)
            xa=6;
        if(x==58)
            xa=7;
        if(x==59)
            xa=8;
        if(x==60)
            xa=10;
        if(x==61)
            xa=11;
        if(x==62)
            xa=12;
        if(x==63)
            xa=14;
        if(x>64)
            xa=0;
    }
}

```

Gracias a este código se puede asignar un rango de inclinación donde el dispositivo teóricamente esta estable y en caso de salir de este es necesario aplicar el aumento correspondiente para llegar a la estabilidad del dispositivo. El código anterior se realiza para cada uno de los 4 ejes y así tener un aumento constante en cada uno de ellos. El código completo viene en la hoja del apéndice D.¹

¹ El apéndice D se encuentra en la página 86.

Al tener este código listo solo hacía falta la integración de los 3 programas principales. Para la integración de los 3 programas se debió de considerar la manera en la que estos se iban a ejecutar debido a su prioridad; se determino que lo primero que se tenía que hacer era medir la inclinación que tenía el dispositivo para de ahí empezar a manipular los motores y posteriormente obtener la medición de la altura. En el programa que se muestra a continuación se puede apreciar cómo se realiza primero la inicialización del Nunchuk mediante la cual se obtienen los valores actuales de la inclinación del acelerómetro, posteriormente se realiza la acción de control para equilibrar la posición del vehículo (esto se puede ver en las 4 variables: 'a', 'b', 'c' y 'd' cuando se les suman la variables 'xa', 'xb', 'xc' y 'xd'), a continuación se asignan los valores a los motores y se mandan por puerto serial, finalmente las variables de aumento de velocidad se restean y se lleva a cabo la medición de la altura.

```
{  
  
    inclinacion();          //para obtener la inclinación actual del  
//dispositivo  
    aumentoVelocidad();  
  
    a=v + xa;  
    b=v + xb;  
    c=v + xc;  
    d=v + xd;  
  
    analogWrite(act1, a);  // se le asigna un valor determinado a  
//cada motor  
    analogWrite(act2, b);  
    analogWrite(act3, c);  
    analogWrite(act4, d);  
    delay(500);           // se espera un segundo para pasar  
//a una nueva velocidad  
  
    char buf2[4];
```

```

Serial.print (itoa(((a*100)/255), buf2, 10));
Serial.print ("a");
Serial.print (xa);
Serial.print ("h");
Serial.print (itoa(((b*100)/255), buf2, 10));
Serial.print ("b");
Serial.print (xb);
Serial.print ("i");
Serial.print (itoa(((c*100)/255), buf2, 10));
Serial.print ("c");
Serial.print (xc);
Serial.print ("j");
Serial.print (itoa(((d*100)/255), buf2, 10));
Serial.print ("d");
Serial.print (xd);
Serial.print ("k");

    rstInclinacion();
    medirDistancia();
    delay(500);
}

```

El código completo de los 3 programas unidos se muestra en el Apéndice E².

3.2 Control remoto

Como más adelante se menciona, cuando se realizan las primeras pruebas en la estructura es necesario contar con un sistema que permita encender y apagar los motores y el Arduino en caso de presentarse una emergencia o solo para poder encender y apagar el dispositivo de una manera segura.

² Apéndice E ubicado en la pagina 92.

En las pruebas de vuelo realizadas con la tercera estructura que fue la segunda hecha de tubos de acero inoxidable se contaba con un sistema de paro de emergencia que reseteaba el Arduino por medio de cables lo cual representaba un problema debido a que estos se podían enredar con las hélices además de que era una manera muy insegura de resetear el dispositivo, por eso fue que surgió la necesidad de un control inalámbrico.

Al no contar con el tiempo necesario para armar y diseñar un sistema que hiciera esto se optó por adaptar un sistema comercial utilizado para prender o apagar equipos electrónicos de manera remota mediante un control de señales infrarrojas.



Figura 3. 3: Sistema de encendido y apagado por control remoto

Este dispositivo es de la marca “Illuminoux” y básicamente consta de un emisor y un receptor de señales infrarrojas que accionan un relevador para abrir y cerrar un circuito, esto lo hacen con la opción de temporización de 15 y 30 min, y de 1 y 2 horas.

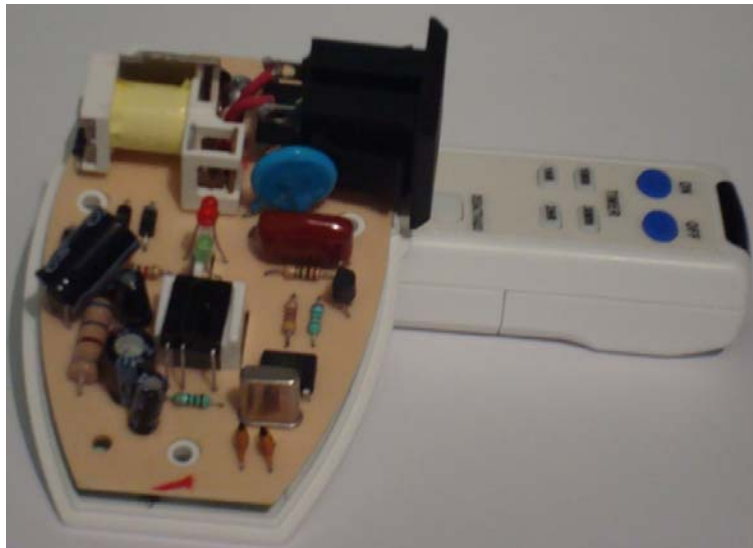


Figura 3. 4: Circuito del interruptor accionado por infrarrojos.

Este circuito funciona por medio de la alimentación de 120 V a 60 Hz porque necesita entregar el mismo voltaje a la salida, sin embargo todo el sistema de control trabaja a un voltaje menor gracias a un puente de diodos, a capacitores y algunas resistencias que bajan el voltaje hasta 8.5 V, pero analizando se observo que se podría trabajar sin ningún problema a 6 V para lo cual eran necesarias unas pequeñas modificaciones, básicamente quitar la parte que reducía el voltaje a 8.5 V.

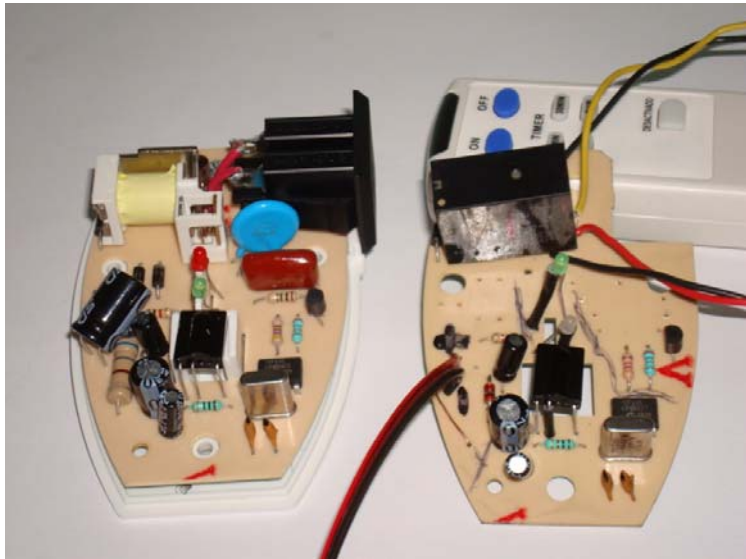


Figura 3. 5: Modificaciones realizadas en el circuito.

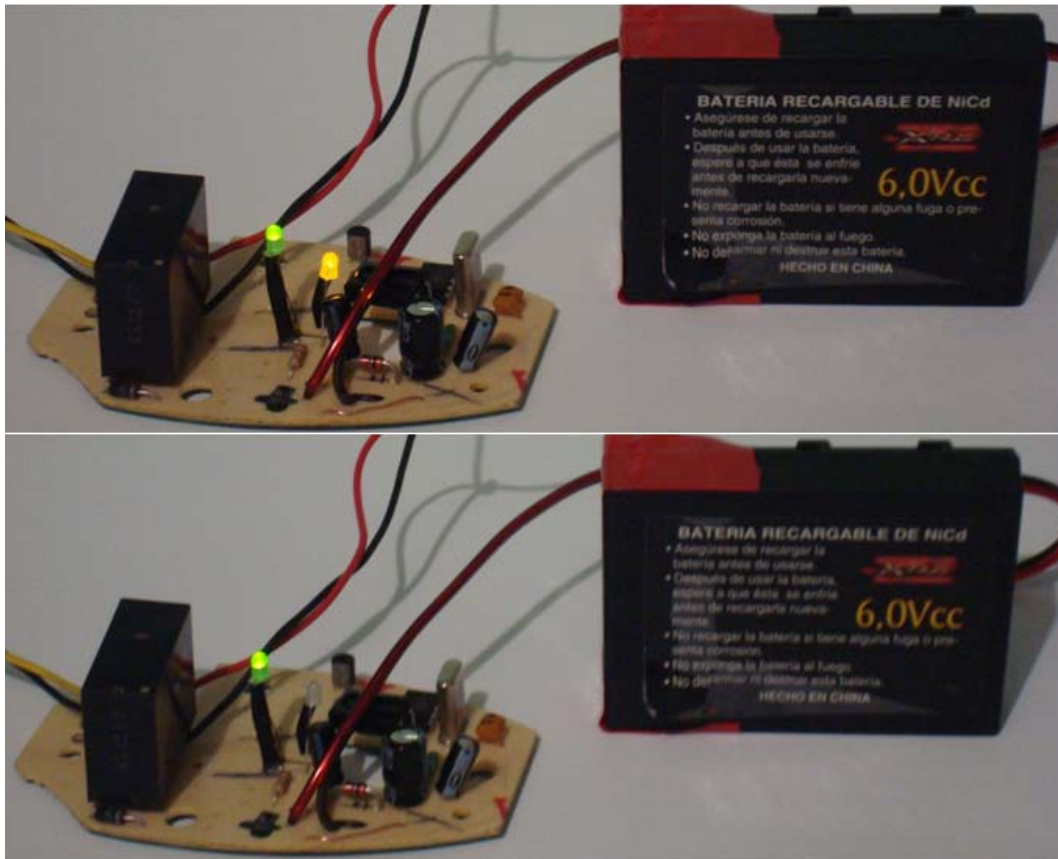


Figura 3. 6: Acción del Relevador preniendo y apagando un LED.

Al haber realizado estos cambios ahora el circuito ya trabajaba con 6 volts y se alimenta de la pila con la que también se alimenta al Arduino. Gracias a este circuito podemos abrir y cerrar los relés mediante la acción de un relevador, estos circuitos serán la alimentación de los motores y del Arduino. De esta manera contamos con dos ventajas, la de poder conectar todo y hacer que entre en funcionamiento cuando nos aseguremos que está correctamente conectado y el de poder apagarlo totalmente en caso de una emergencia; esto nos da la ventaja de realizar las pruebas con una mayor seguridad para los usuarios y para los componentes del dispositivo.