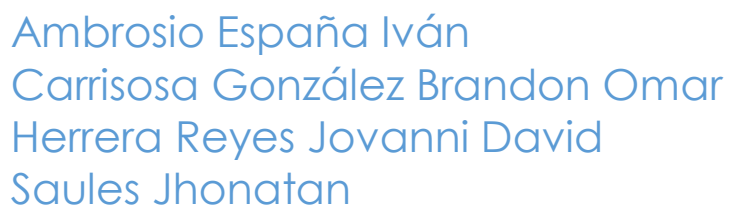


## Microcontroladores.

## Microcontroladores.



Instituto Politécnico Nacional – Escuela Superior de Cómputo.

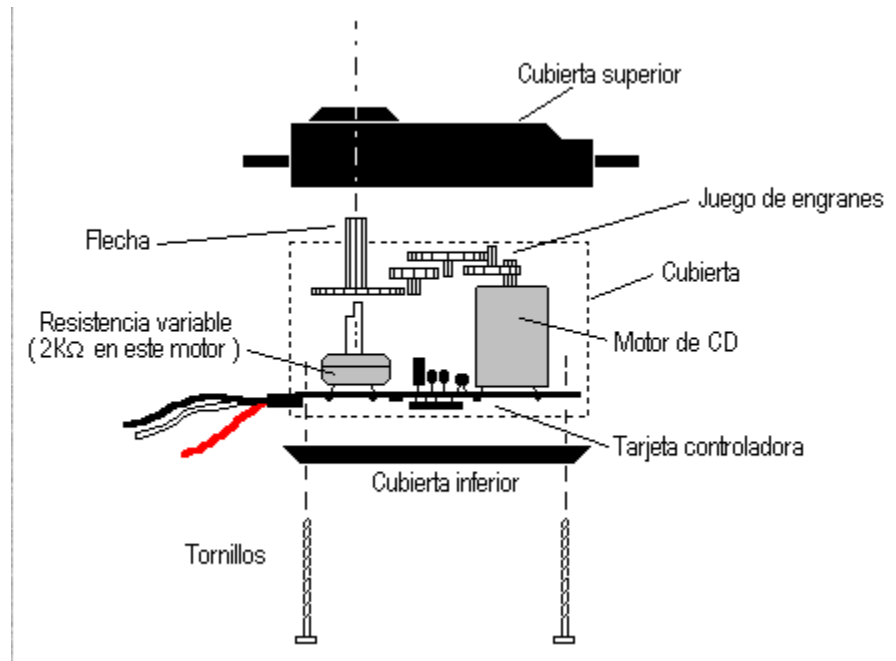
Profesor: Pérez Pérez .

## **Servomotor.**

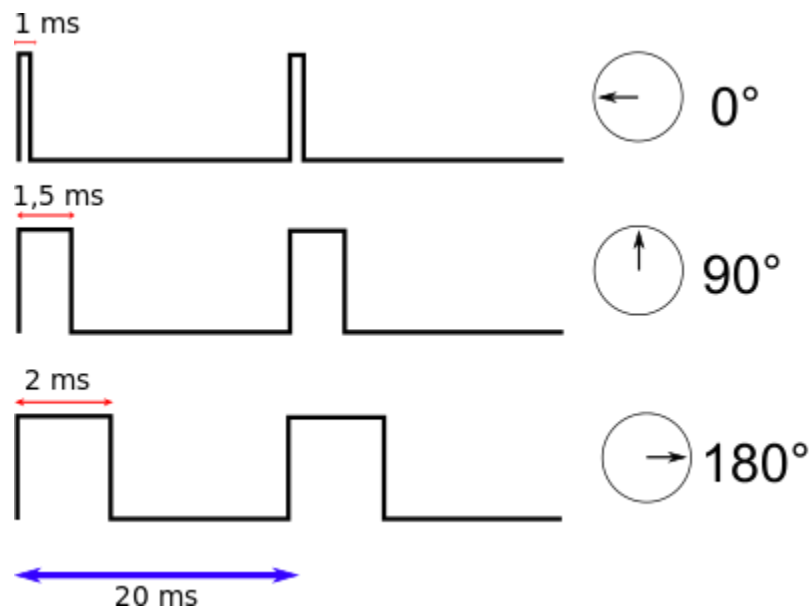
Un servomotor es un tipo de servomecanismo, y un servomecanismo es un dispositivo que tiene un sistema de corrección con retroalimentación para el control de un parámetro específico.

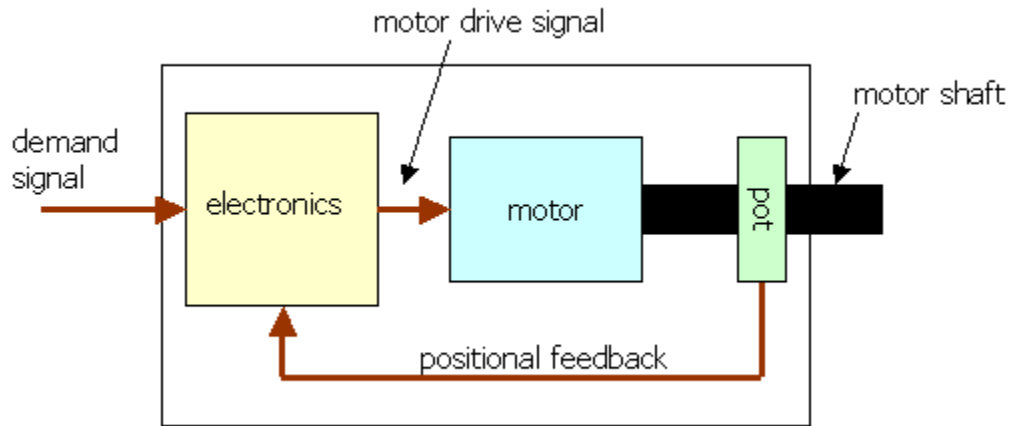


Los servomotores son dispositivos electromecánicos que consisten en un motor eléctrico, un juego de engranes y una tarjeta de control, todo confinado dentro de una carcasa de plástico. La característica principal de estos motores es que la gran mayoría no están hechos para dar rotaciones continuas -algunos sí lo hacen pero se hablarán de ellos más adelante- ya que principalmente son dispositivos de posicionamiento en un intervalo de operación. En esta ocasión se tratará exclusivamente de servomotores para modelismo, excluyendo los servomotores industriales.



Los servomotores funcionan por medio de modulación de ancho de pulso -*pulse-width modulation* (PWM)- Para los servos para modelismo, la frecuencia usada para mandar la secuencia de pulsos al servomotor es de 50 Hz -esto significa que cada ciclo dura 20 ms- Las duraciones de cada pulso se interpretan como comandos de posicionamiento del motor, mientras que los espacios entre cada pulso son despreciados. En la mayoría de los servomotores los anchos de pulso son de 1 ms a 2 ms, que cuando son aplicados al servomotor generan un desplazamiento de  $-90^\circ$  a  $+90^\circ$  por lo que, de una manera más sencilla, el ángulo de giro está determinado por el ancho de pulso; si el ancho de pulso fuera de 1.5 ms, el motor se posicionará en la parte central del rango - a  $0^\circ$ -





La tarjeta electrónica dentro de la carcasa interpreta las instrucciones de un controlador externo. El código de comando especifica el ángulo de rotación deseado medido como un *offset* en ambos lados de la posición central del rango del motor. El motor gira rápidamente a la posición específica y se detiene en ese punto -el potenciómetro, que está acoplado al eje de salida, sirve para enviar una señal de retroalimentación que asegure la posición del motor- En tanto que la señal de comando continúe y se mantenga activado el motor, el motor se mantendrá en la misma posición, incluso si hay una fuerza que lo haga rotar y que sea menor a su torque. Si no hay alguna fuerza, el motor, que estará en una posición estacionaria, consumirá poca corriente.

Dado que muchos de los microcontroladores tienen pines asignados para usar la modulación por ancho de pulso -PWM- y pueden conseguirse fácilmente, son los elementos empleados para el control de las secuencias de pulsos para los servomotores, aunque éstos pueden ser substituidos por un temporizador 555. En otros casos, pueden emplearse tarjetas controladoras que tengan conexión USB para que por medio de ella el servo pueda ser controlado a través de la computadora.

## Código.

*Program to demonstrate the use servo motors with AVR Microcontrollers.*

*For More Details Visit: <http://www.eXtremeElectronics.co.in>*

*Copyright (c) 2008-2010  
eXtreme Electronics, India*

*Servo Motor: Futaba s3003  
Servo Control PIN (white): To OC1A PIN  
Crystal: 16MHz  
LOW Fuse: 0xFF  
HIGH Fuse: 0xC9*

Compiler:avr-gcc toolchain  
Project Manager/IDE: AVR Studio

#### NOTICE

-----  
NO PART OF THIS WORK CAN BE COPIED, DISTRIBUTED OR PUBLISHED WITHOUT A  
WRITTEN PERMISSION FROM EXTREME ELECTRONICS INDIA. THE LIBRARY, NOR ANY  
PART  
OF IT CAN BE USED IN COMMERCIAL APPLICATIONS. IT IS INTENDED TO BE USED  
FOR

HOBBY, LEARNING AND EDUCATIONAL PURPOSE ONLY. IF YOU WANT TO USE THEM IN  
COMMERCIAL APPLICATION PLEASE WRITE TO THE AUTHOR.

WRITTEN BY:  
AVINASH GUPTA  
me@avinashgupta.com

```
*****  
*****/  
#include <avr/io.h>  
  
#include <util/delay.h>  
  
//Simple Wait Function  
void Wait()  
{  
    uint8_t i;  
    for(i=0;i<50;i++)  
    {  
        _delay_loop_2(0);  
        _delay_loop_2(0);  
        _delay_loop_2(0);  
    }  
}  
  
void main()  
{  
    //Configure TIMER1  
    TCCR1A|=(1<<COM1A1)|(1<<COM1B1)|(1<<WGM11);           //NON Inverted PWM  
    TCCR1B|=(1<<WGM13)|(1<<WGM12)|(1<<CS11)|(1<<CS10); //PRESCALER=64 MODE  
14 (FAST PWM)  
  
    ICR1=4999; //fPWM=50Hz (Period = 20ms Standard).  
  
    DDRD|=(1<<PD4)|(1<<PD5); //PWM Pins as Out  
  
    while(1)  
    {  
  
        OCR1A=97; //0 degree  
        Wait();  
  
        OCR1A=316; //90 degree
```

```

Wait();

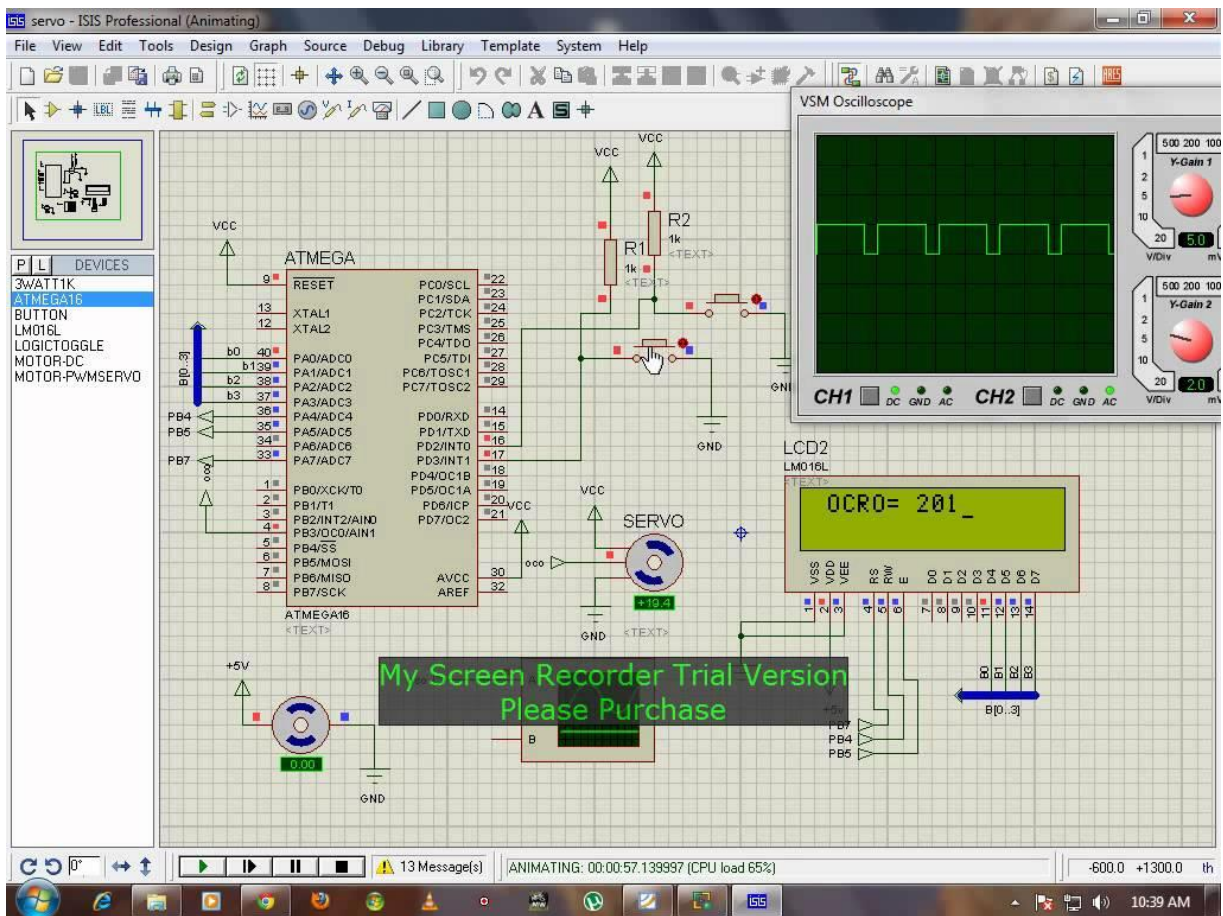
OCR1A=425;  //135 degree

Wait();

OCR1A=535;  //180 degree
Wait();
}
}

```

## Simulación.



## **Conclusión.**

Con este circuito pudimos situar la posición de un servomotor de los usados en modelismo en 256 posiciones diferentes, la cual se reflejara sobre 8 diodos LED. Manteniendo presionado el pulsador IZQ o DER variaremos la posición del servo en un sentido u otro, al mismo tiempo de que los led van indicando secuencias diferentes.

Aproximadamente mediante la actuación de estos 2 pulsadores tardara 50seg el servomotor en realizar todo el recorrido, pero si actuamos al mismo tiempo sobre el pulsador TURBO, se reducirá este tiempo a solo 5seg.

Tuvimos la posibilidad de accionarlo mediante los microswitch, y llevar al servo a unas posiciones concretas sin necesidad de tener que verlo. Con los pulsadores IZQ y DER llevaremos el servo a la posición deseada, en donde observaremos los LED que permanecen encendidos, colocando en la posición de ON los microinterruptores que coincidan con los LED encendidos, de modo que independientemente de donde se encuentre el servo, este regresará a esa posición siempre que actuemos sobre el pulsador SELEC.

## **Referencias.**

<http://extremeelectronics.co.in/avr-tutorials/servo-motor-control-by-using-avr-atmega32-microcontroller/>

<http://www.avrfreaks.net/forum/atmega8535-servo-motors>

[http://www.sites.upiicsa.ipn.mx/polilibros/portal/Polilibros/P\\_terminados/Servomotor-Etмега8535.htm](http://www.sites.upiicsa.ipn.mx/polilibros/portal/Polilibros/P_terminados/Servomotor-Etмега8535.htm)