**BRAZO ROBOTICO**



**Integrantes:**

* **Jostin Navarro**
* **Thiago Ramirez**
* **Ludmila Rovner**
* **Tomas Herrero.**

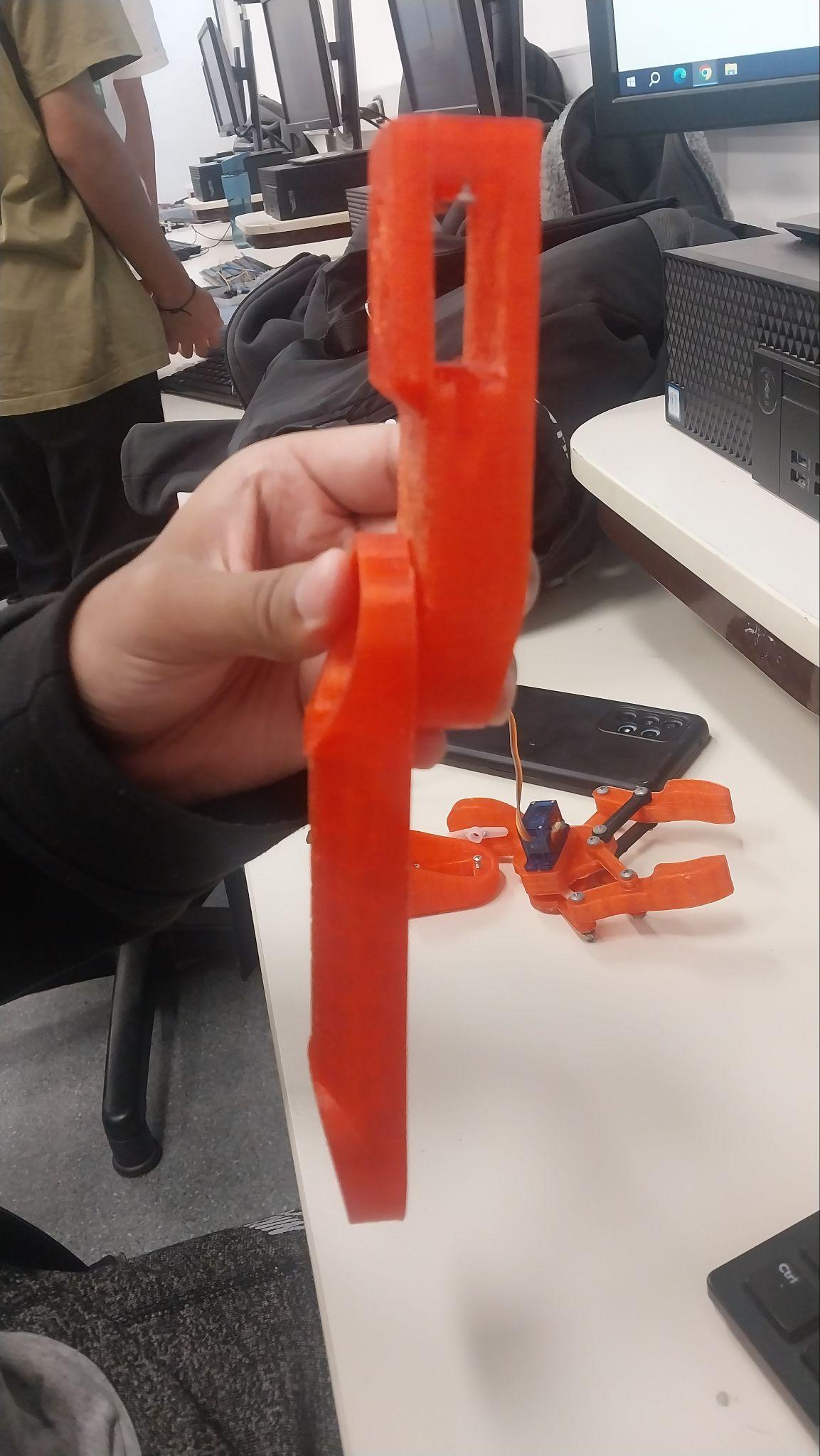
**Profesor: Gonzalo Consorti**

**Descripción: En nuestro caso vamos a realizamos o tratamos de hacer un brazo robótico que funcione y se maneje mediante un joystick analógico para poder mover cada parte de dichoso brazo hacia la derecha o izquierda dependiendo de para donde muevas el joystick.**

**Dia 25 de Octubre:**

**En nuestro caso vamos a realizamos o tratamos de hacer un brazo robótico que funcione y se maneje mediante un joystick analógico para poder mover cada parte de dichoso brazo hacia la derecha o izquierda dependiendo de para donde muevas el joystick.**

**El día 25 de Octubre este día en el trabajo primero nos dividimos por partes el poder armar el robot juntando las partes para poder tenerlo armado para el futuro y poder usar el código que ya fue investigado o mejor dicho que ya habían realizado previamente mi compañero.**

****

**Esta es la primera parte que armamos que es la parte del medio del brazo la cual consta de 2 piezas, esta parte con Thiago fue que la terminamos armando mientras que por otro lado nuestra compañera estaba con la otra parte la parte superior del brazo, ya luego mostrare una foto de como es esa parte y de cómo queda todo junto o bueno casi todo ya que nos faltaba la base del trabajo que no había sido imprimida.**

****

**Esta es la parte superior del servo brazo la cual consta de la pinza y de digamos que lo que pudiera ser la mano, está parte la inició un compañero empezando a armar para luego nosotros seguir o terminar de armar lo que faltaba y poder tomarle la foto para poder tener guardado el proceso de como vamos armando y como va quedando el brazo parte por parte, ya luego explicaremos las partes una por una junto a su función aunque algunas no hace mucha falta la explicación ya que es medio obvio en algunas partes pero igual se explicaran.**

****

**Esto ya seria el como quedo el brazo armado luego de “Juntar las piezas” ya que realmente estaban sueltas solo que las sostenemos con las manos ya que le faltaban los servomotores como se puede apreciar en la foto que tienen huecos que es por falta de ellos luego en un futuro ya se los agregaremos y quedara todo bien y junto y con suerte tambien estara junto a la base ya que si, eso es solo la parte de arriba todavía falta todo lo de abajo o mejor dicho la base del brazo.**

**Acá hay un link sobre información que de mi parte me puse a buscar sobre servomotores y cómo funcionaban junto a sus conexiones y códigos.**

[**https://programarfacil.com/blog/arduino-blog/servomotor-con-arduino/**](https://programarfacil.com/blog/arduino-blog/servomotor-con-arduino/)

**Ahora mencionare sobre lo que llegue a ver de la pagina esta llega a hablar sobre un poco de los servo motores pequeños los azules explicando sus características junto a qué cosas pueden hacer o hasta que tanto pueden llegar a realizar o moverse, como por ejemplo.**

* **Que tiene un ángulo de giro de 180º**
* **Puede moverse con una resolución de más de 1 grado, este es el máximo de resolución que vamos a conseguir.**

**Otra cosas que nos habla esta página o mejor dicho nos muestra es una foto de sus conexiones ósea del servomotor a la placa arduino junto a donde iría cada cable.**

**Otro tema es que nos explica y nos enseña cómo se pueden llegar a programar un servo motor sin usar los joysticks mediante una página que se suele usar para cuando uses los códigos y los componentes en físico y no mediante tinkercad.**

**En este caso nos enseña algo importante y nos da una ayuda que son:**

1. **Cómo activar para que la página sea para servo motores incluyendo el código básico que debería de tener**
2. **Y un código de cómo podemos mover los servo motores uno a uno mediante esa página.**

**Al final esto no terminó siendo de mucha ayuda osea a medias ya que mis compañeros ya tenían planeado como seria ya el código el cual lo bueno es que fue parecido a lo que llegue a investigar pero que al final no llegamos a utilizar realmente ya que el código ese iba a ser muy largo cuando realmente se podía acortar usando un joystick en vez de como nosotros pensamos originalmente que seria, que solo tendríamos que darle la señal para que se moviera y ya.**

**Dia 7 de Noviembre:**

**El dia jueves 7 de noviembre con nuestro grupo nos quedamos un tiempo extra para poder hacer y avanzar con el brazo robótico probando un código y ver si funciona para cada parte y terminar o seguir chequeando si funcionaban todo los servo motores sin ninguna complicación, utilizamos este código de inicio y ver si funcionaban las piezas armadas previamente.**

**Este fue el primer código que decidimos probar para nuestra clase con el servo brazo y poder ver si el código le funcionaba a nuestro brazo y ver si se movían las piezas y cumplian codo lo que le llegábamos a pedir y con eso ya saber que tan lejos estábamos de como llegar a ser nuestro código final para este proyecto.**

| **void loop()  {  Giro.write(0);//Movimiento 1  delay(2000);  Hombro.write(0);  delay(2000);  Codo.write(180);  delay(2000);  Muneca1.write(0);  delay(2000);  Muneca2.write(0);  delay(2000);  Pinza.write(0);  delay(2000);** |
| --- |

**Esta parte de nuestro código se encarga de cómo marcamos el movimiento uno, nuestro plan era que al arrancarlo todos los servos estén en los 0° claro cada uno respetando su turno mediante el delay que hace que hagan el movimien todo rápido y se dañen los servos sino que se tomen su tiempo para no dañarse y hacer que todos estos servos motores inicien ya desde una posición normal o común en vez de en una posición avanzada o mejor dicho en un ángulo alto y así hacer todos los movimientos bien.**

| **Giro.write(90);//Movimiento 2  delay(2000);  Hombro.write(90);  delay(2000);  Codo.write(0);  delay(2000);  Pinza.write(90);  delay(2000);** |
| --- |

**Este de acá sería nuestro movimiento 2, la función de esta parte es asegurar que los servomotores, luego de haber alcanzado la posición previamente indicada, continúen con el proceso de movimiento sin interrupciones. Es decir, una vez que los servos han llegado a la posición deseada en el primer paso, el código se encarga de asegurar que estos se ajusten con precisión a esta nueva posición. Y que luego de eso lograr avanzar al siguiente movimiento de una buena forma y no que le cueste al hacer un movimiento grande o largo sino que se divide por partes que serían los diferentes movimientos que usamos para este código.**

| **Muneca2.write(90);//Movimiento 3  delay(2000);  Codo.write(90);  delay(2000);  Hombro.write(45);  delay(2000);  Pinza.write(0);  delay(2000);** |
| --- |

**Este es el tercer movimiento, el cual mantiene algunos de los ángulos empleados en los movimientos anteriores. De hecho, algunos de estos ángulos se modificaron, adoptando nuevas inclinaciones ayudando a que el brazo se acerque poco a poco al movimiento final o a nuestro objetivo en esos momentos, que en estos momentos realmente no tenemos uno sino que estamos es más probando si es que funcionan sin ningún problema y con fluidez, que no pegue un salto ni nada que se tome su tiempo el cual fue indicado en su delay y que el código funcione sin tirarnos un error por alguna tontería que nos falto agregar o alguna cosa de ese estilo que no vimos.**

| **Muneca2.write(0);//Movimiento 4  delay(2000);  Codo.write(180);  delay(2000);  Hombro.write(90);  delay(2000);  Giro.write(0);  delay(2000);** |
| --- |

**Ahora hablaremos de nuestro cuarto movimiento, en este ya cambiamos un poco más las cosas ya que algunos ángulos los llevamos a los 180° probando si es que también son capaz de llegar a los 180° mientras que con otros los llegamos a devolver a un valor menor para también comprobar si pueden llegar o mejor dicho devolverse o retroceder sin preocupaciones o errores o sin que lleguen a trabarse a mitad de camino y claro para algo que ya e repetido para que haga y pueda hacer lo que le llegamos a indicar que haga osea para llegar a un objetivo el cual realmente no tenemos mucho en mente cual podría llegar a ser aparte de que se mueva.**

| **Hombro.write(45);//Movimiento 5  delay(2000);  Codo.write(80);  delay(2000);  Pinza.write(0);  delay(2000);** |
| --- |

**Este ya es nuestro quinto movimiento y casi nuestro último movimiento, ya este código o mejor dicho esta parte del código es corta ya que al ser de lo último ya trataría solo de tocar lo que queda o mejor dicho mover esas piezas faltantes a la posición que les falta llegar o terminar estando para poder pasar a ya lo ultimo para dar ese último movimiento o mejor dicho para llevarlo ya a su posición final.**

**Claro otra cosa tambien es que cumpla con el hecho de terminar de probar si todo el código funcionaba hasta el momento osea que hasta el movimiento 5 todo va bien sin ningún error y que como ya mencione vaya con fluidez y no que tarde mucho o haga el movimiento muy rápido, pero eso ya esto solo trata de terminar o casi sino que llevar el brazo a que esté ya listo para su último movimiento que sería ya su trazo o camino final y que digamos que ya esté listo o terminado si es que nos funciona bien todo o no falta alguna última cosa nueva que nos saltamos o se nos olvidó agregar en algún momento del código.**

| **Muneca2.write(0);//Movimiento 6  delay(2000);  Codo.write(180);  delay(2000);  Pinza.write(0);  delay(2000);  Hombro.write(180);  delay(2000);  Codo.write(0);  delay(2000); }** |
| --- |

**Este ya sería el último movimiento que claro tiene esos valores pero son más para probar al igual que todos los demás eran para ver si cuando nosotros les pedíamos que hiciera eso se movieran a ese ángulo indicado aunque por el momento no era para un objetivo exacto ni nada.**

**Luego de probar este código vimos que funcionaba pero de cierta forma ya que tiraba algunos errores al ser largo y también ya que tampoco probamos todas las partes en conjunto sino que preferimos ir una a uno ir chequeando las viendo si funcionaban o si hacían el movimiento indicado, luego de estar un rato con eso y matándonos la cabeza ya que a veces (casi siempre) el robot desidia hacer lo que quería, le mandábamos a ir a un lugar x y no se movía y cuando no queríamos que hiciera algo se movía luego de un rato con eso decidimos cambiar el código o mejor dicho achicarlo para ver si de esa forma quería funcionar bien o aunque sea nos hacía caso por una vez.**

**También este era el codigo con el cual pensabamos que teniamos que hacer el trabajo pero eso fue antes de saber que tendríamos que usar los joystick por eso es que tambien nos quedo tan largo esa parte ya que nosotros ya le dábamos el movimiento en sí a cada parte en vez de decirselo con el joystick.**

**Este fue el código que al final terminamos usando en esa clase un código que buscamos para poder usar de guía.**

[**https://www.arduino.cc/en/Tutorial/LibraryExamples/Sweep**](https://www.arduino.cc/en/Tutorial/LibraryExamples/Sweep)

**Usamos esta página para buscar información de cómo podíamos hacerlo y encontramos este código que decidimos utilizar y ver si nos ayudaba para este trabajo en un futuro.**

| **if(analogRead(A5) <= 800 && pos <= 170)  {  pos += 2;  myservo.write(pos);  delay(15);  }else if(analogRead(A4) <= 800 && pos >= 20)  {  pos -= 2;  myservo.write(pos);   delay(15);  } }** |
| --- |

**Luego de probar este código vimos que nos funcionaba de cierta forma ya que seguiamos con los mismos problemas pero que por lo menos si nos llegaba a hacer más caso con las instrucciones y movimientos que le llegábamos a pedir que realizará el por su cuenta al arrancar el código. Aunque a veces no hacía caso y bueno no se movía o se movía cuando quería y no cuando se le indicaba, esa clase nos la pasamos mucho tiempo tratando de solucionar eso.**

**Dia 22 de Noviembre:**

**El dia 22 de noviembre si, mucho tiempo después nos pusimos a seguir con el trabajo tratando de seguir avanzando y viendo si quería funcionar el brazo, la verdad esta clase nos mató mucho en general probamos distintos códigos modificando el original poco a poco para tratar de llegar a un buen resultado, nos la pasamos casi toda la clase tratando de vuelta si el código y el brazo quería funcionar ya que si, descubrimos con todas las clases en las que fui y unas apartes que fue una compañera del equipo vimos que eso, el trabajo funcionaba cuando le interesaba, hoy podía funcionar y mañana no funcionaba pero eso, nos la pasamos viendo como el robot era racista y se quería suicidar todo el rato trataba de eso matarse, nos dimos cuenta que para que funcione no tenemos idea de que hacer realmente por ahora ya que el robot tiene muchos, demasiados, exageradamente grandes problemas mentales.**

**Probamos este código esa clase para chequear si funcionaba pero como ya explique en el párrafo pasado no funcionaba ya que el robot con problemas y racista funcionaba cuando él quería trabajar y funcionar lo peor es que cuando él lo hacía y se movía era a cualquiera sin hacernos el más mínimo caso.**

**CODIGO DEL 22 DE NOVIEMBRE**

| **void loop() { //joystick uno(giro X & hombro Y)  Serial.print ( "x:" ) ;  Serial.print ( analogRead(A0) ) ;  Serial.print ( " y:" ) ;  Serial.print ( analogRead(A1) ) ;  Serial.print ( " w:" ) ;  Serial.print ( analogRead(A2) ) ;  Serial.print ( " z:" ) ;  Serial.print ( analogRead(A3) ) ;** |
| --- |

**Bueno hablando de este nuevo código pues bueno, esta parte del código como se indica nos muestra en la pantalla digamos que los ejes del joystick que utilizamos para cada pieza, ósea nos muestra a qué distancia está el joystick en los momentos luego de mover la palanca para un lado, si lo mueves para arriba subiría el eje “Y” o si lo bajas bajaría hasta ciertos valores los cuales en otro momentos serán más específicos o mejor dicho en el caso que usemos este código al terminar el trabajo al explicarlo mejor se dirán.**

| **Serial.print ( "\t\t") ;  Serial.print ( "G:" ) ;  Serial.print (posG ) ;  Serial.print ( " " ) ;  Serial.print ( "H:" ) ;  Serial.print ( posH ) ;  Serial.print ( " " ) ;  Serial.print ( "C:" ) ;  Serial.print (posC ) ;  Serial.print ( " " ) ;  Serial.print ( "M1:" ) ;  Serial.print ( posM1 ) ;** |
| --- |

**Este código es parecido al de arriba pero a la vez no tanto, esté más que mostrarnos los datos del joystick o donde están sus ejes a qué distancia esté nos muestra la posición o mejor dicho los grados de cada parte del brazo claro cada posición cambia según que tanto se mueva el joystick, para ser más exactos nos muestra cuánto se mueve brazo poco a poco según qué pieza se mueve te muestra qué grados está actualmente si esta en 80° y mientras lo mueves te va mostrando como va subiendo o te muestra cómo es que va bajando ese valor y todo mediante una condición que se menciona en otra parte del código que ya es la que habla o mejor dicho trata del movimiento de cada pieza o mejor dicho cada servomotor que mueve cada una de las piezas individualmente mediante un joystick para las piezas, claro es un joystick para 2 servomotores no todos para un joystick.**

| **giro.write(posG);  hombro.write(posH);  codo.write(posC);  muneca1.write(posM1);**  **if ( analogRead(A0) <= 350 && posG <= 170)  {  posG += 2;}**  **}** |
| --- |

**Luego estaría esta parte que está incompleta pero que se encarga ya del movimiento.**

**Más abajo se hablará en profundidad sobre el código y su funcionalidad exacta para cada parte de este trabajo en vez de hablarlo por encima.**

**Este es el último código que nos quedó y el que utilizamos en esa clase o bueno a medias ya que el código fue cambiado muchas veces en el transcurso de esta clase pero terminó siendo ese el como quedo osea luego de todas las modificaciones que hicimos esa clase para que el robot o mejor dicho el brazo pudiera hacernos caso ya que como comente antes la mayoría de las clases el trabajo no hacía caso o funcionaba cuando quería o algunas piezas o partes se movían solas mientras que las otras no.**

**29 del 11 de 2024**

**En esta clase entre todos nos pusimos a probar el anterior código subido para ver si nos funcionaba ahora si con el trabajo o bueno a medias ya que este código es más para ver si los joystick funcionaban y no tiraban cualquier resultado.**

**Por el momento tenía algunos fallos ya que con ciertas partes el código nos tiraba un valor nada que ver con lo indicado o cuando queríamos hacer que se devuelva a su posición original no quería moverse sino que mantenía su posición nueva.**

**Logramos medio solucionar que se muevan a la posición indicada en este caso 20 y si logra moverse el tema es que al moverse trata de matarse prácticamente, ahora estamos probando el que se ponga en la posición 0.**

**Bueno por lo visto lo que habíamos realizados en clases pasadas qué fue el acomodar cada motor a sus 0 grados para que estén bien pensábamos en su tiempo que estaban bien pero en esta clase nos dimos cuenta que algunos estaban mal o eso parecía.**

**Al final quedó que el último código estaba incompleto entonces nos pusimos a volver a poner lo que faltaba pero poco a poco por parte de cada uno.**

**Luego de un largo rato tratando de solucionarlos lo logramos, pudimos hacer que todos funcionen de vuelta uno a uno y todos juntos también, pudimos terminar el código faltante haciendo que ya estemos cerca de terminarlo, la verdad mucha felicidad para nosotros aunque en algunos momentos le volvían los problemas y todavía se quiere matar pero no importa.**

**Me puse a buscar como usar la pinza como un botón usando el servo para que cada que apretemos el joystick se abra y cierre la pinza cada que lo apretemos por el momento no encontramos nada.**

[**https://www.automatizacionparatodos.com/joystick-arduino-opencv/**](https://www.automatizacionparatodos.com/joystick-arduino-opencv/)

**Encontre esta pagina que parece que te explica como funciona el joystick y cómo podemos usar el joystick como un botón y ver si con eso podemos abrir y cerrar la pinza, también enseña muchas mas cosas como que explica cómo llega a funciona un joystick y explicar el que es.**

**Este fue el código que vamos a utilizar o medio ver que te dice el cómo utilizarlo ahora lo pondremos a prueba y vamos a ver si nos funciona o si no sirve para nada.**

| **void loop() {  valorX = analogRead(VRX);  valorY = analogRead(VRY);  boton = digitalRead(SW);   Serial.print("Valor X: ");  Serial.println(valorX);  Serial.print("Valor Y: ");  Serial.println(valorY);  Serial.print("Boton: ");  Serial.println(boton);  delay(1000); }** |
| --- |

**Bueno al final lo que terminamos usando del código que encontramos fue esta parte del código que así nos quedó luego de sacarle las cosas que no eran tan necesarias o que no íbamos a usar para nuestra pinza y bueno ya luego para poder probar la parte de la pinza o mejor dicho el funcionamiento de la pinza y lograr y hacia ese movimiento.**

| **} if (boton, HIGH){  pinBoton = 20; }else if (boton, LOW);{  pinBoton = 60; } }** |
| --- |

**Bueno por el momento no funcionó eso ya que no nos detectaba al presionarlo sino que al presionarlo no detectaba nada y aparte de alguna forma nos afectó con la otra parte del código ya que una pieza o servo no se quería mover, por el momento estamos solucionando el porqué no funciona pero en forma aparte no junto a todo lo demás del código.**

**Luego de un rato probando códigos para saber cómo usar la pinza llegamos a este código con el cual pudimos saber cómo usar la pinza y que se pueda abrir y cerrar hasta la distancia indicada.**

| **if( digitalRead(13) == LOW )  {  pinza.write(0);  }else if( digitalRead(13) == HIGH )  {  pinza.write(50);  }  }** |
| --- |

**Explicación de nuestro código completo**

**Esta parte del código se encarga de declarar parte por parte las piezas o mejor dicho los servomotores que llegamos a utilizar, en nuestro caso llegamos a usar 6 servo motores cada uno para cada movimiento o bueno a medias cada uno se encargará de un movimiento en específico, por ejemplo el “giro” trata de mover la base del trabajo para que gire todo el brazo en si y asi con cada servo motor.**

| **Servo giro;  Servo hombro; Servo codo; Servo muneca1; Servo muneca2; Servo pinza;** |
| --- |

**En esta otra parte trata de escribir y/o mostrarnos a qué distancia está ubicado cada uno, como llegamos a señalar, nosotros vamos a usar 3 joysticks, cada joystick tiene su eje en esta parte de código marcamos cual es cada joystick mediante las letras para saber en qué eje se encuentra cada uno, por eso es que es casi un copiar y pegar ya que todos tratan de mostrar en qué está cada joystick en que distancia desde el número 0 hasta el que llegue que en los joysticks es de 1025 o 523, claro depende de para donde se mueva la palanca haciendo que cambie su eje a un número entre los indicados.**

| **Serial.print ( "x:" ) ;  Serial.print ( analogRead(A0) ) ;  Serial.print ( " y:" ) ;  Serial.print ( analogRead(A1) ) ;  Serial.print ( " w:" ) ;  Serial.print ( analogRead(A2) ) ;  Serial.print ( " z:" ) ;  Serial.print ( analogRead(A3) ) ;  Serial.print ( " m:" ) ;  Serial.print ( analogRead(A4) ) ;** |
| --- |

**En esta otra parte trata de escribir y/o mostrarnos a qué grados está ubicado cada uno, como llegamos a señalar, nosotros vamos a usar 3 joysticks, cada joystick tiene un movimiento que cambiará la posición de una parte o mejor dicho pieza, dependiendo de para donde se mueva y que tanto se sabrá que tantos grados tiene en esos momentos o como nosotros decidimos ponerle sabremos qué posición tiene cada uno al mover el joystick con un eje, como el ejemplo de antes el eje “X” mueve el giro, nosotros al mover ese eje la posición del giro cambia osea sus grados y en la pantalla nos mostrará cómo va cambiando esa posición poco a poco hasta que dejemos de mover ese eje.**

| **Serial.print ( "G:" ) ;  Serial.print (posG ) ;  Serial.print ( " " ) ;  Serial.print ( "H:" ) ;  Serial.print ( posH ) ;  Serial.print ( " " ) ;  Serial.print ( "C:" ) ;  Serial.print (posC ) ;  Serial.print ( " " ) ;  Serial.print ( "M1:" ) ;  Serial.print ( posM1 ) ;  Serial.print ( " " ) ;  Serial.print ( "M2:" ) ;  Serial.print ( " " ) ;  Serial.print ( posM2 );** |
| --- |

**En este momento llegamos a lo que yo diría que es la parte más importante del código ya que esta de aca se encarga de todos los movimientos mediante el joystick osea el movimiento o mejor dicho la parte de los servomotores ya que lo de antes solo es para mostrar su posición pero ya esto es para que se puedan mover.**

**Este código de acá se encarga de indicar que movimiento hará el primer servo motor osea el de la base el “Giro”. Marcando que movimiento hará dependiendo de su posición actual, si está en una posición x que suba y se mueva subiendo su cantidad o mejor dicho su posición, y si está en otra posición que baje o mejor dicho retroceda.**

| **if(analogRead(A0) <= 350 && posG <= 170)  {  posG += 1;  giro.write(posG);  delay(50);  }else if(analogRead(A0) >= 900 && posG >= 20){  posG -= 1;  giro.write(posG);  delay(50);  }** |
| --- |

**Acá es casi lo mismo que el de arriba para no decir que igual solo que este se encarga ya de la parte que le sigue a la base del brazo o como nosotros le decimos el “hombro”, esto ya como se mencionó antes al ser parecido sigue cumpliendo y teniendo el mismo recorrido pero de diferente forma ya que en este caso el “hombro” se moverá de arriba a abajo para que todo el brazo se encuentre ya en un angulo comodo para hacer la acción que le pidamos como por ejemplo agarrar un objeto y moverlo para otro lugar que en nuestro caso es nuestro objetivo, el cual se alcanza con paciencia ya que tenemos que mover poco a poco cada una de las partes a su ángulo como pasaría con esta parte al igual que la de arriba.**

| **if(analogRead(A1) <= 350 && posH <= 170)  {  posH += 1;  hombro.write(posH);  delay(50);  }else if(analogRead(A1) >= 900 && posH >= 20){  posH -= 1;  hombro.write(posH);  delay(50);  }** |
| --- |

**Ya lo que le sigue a ese código va a ser lo mismo que los de arriba menos uno, pero no pondré los otros ya que van todos de lo mismo pero hablare mas especifico en general con estos códigos de qué es que se encarga de hacer realmente.**

**Estos códigos se encargan de marcar dependiendo de cada parte que tanto es moverá, como ven si se cumple una condición x que al uno mover el joystick su posición actual va sumando en 1 en 1 al mover el joystick por ejemplo hacia arriba, haciendo que se mueva digamos que a la derecha y que al hacer lo contrario con la otra condición al nosotros mover el joystick para abajo su posición va bajando de uno en uno y se movería es para el lado contrario como indica su posición ósea hacia la izquierda. De eso es que tratan estos códigos por eso es que son parecidos ya que todos tienen que cumplir la misma condición para su movimiento y cuando el joystick está quieto no se movería para ningún lado sino que mantiene su posición actual.**

**Ahora hablaremos en profundidad más sobre la pinza que fue la última parte que realizamos para ya poder terminar el brazo robótico con todas sus funciones.**

| **if( digitalRead(13) == LOW )  {  pinza.write(0);  }else if( digitalRead(13) == HIGH )  {  pinza.write(50);  }  }** |
| --- |

**Bueno para empezar como ya mencione antes eso se encarga de que dependiendo de si presionamos o no el joystick va a hacer un movimiento o mejor dicho se va a abrir o cerrar, como está indicado en el código cuando no se está pulsando el joystick osea cuando esta LOW la pinza está en la posición 0 pero cuando lo presionas osea esta en HIGH la pinza se mueve hasta que llegue a los 50 grados y luego volvería a los 0 en el caso que se deje de presionar pero si sigue presionado se quedaría en los 50 grados.**

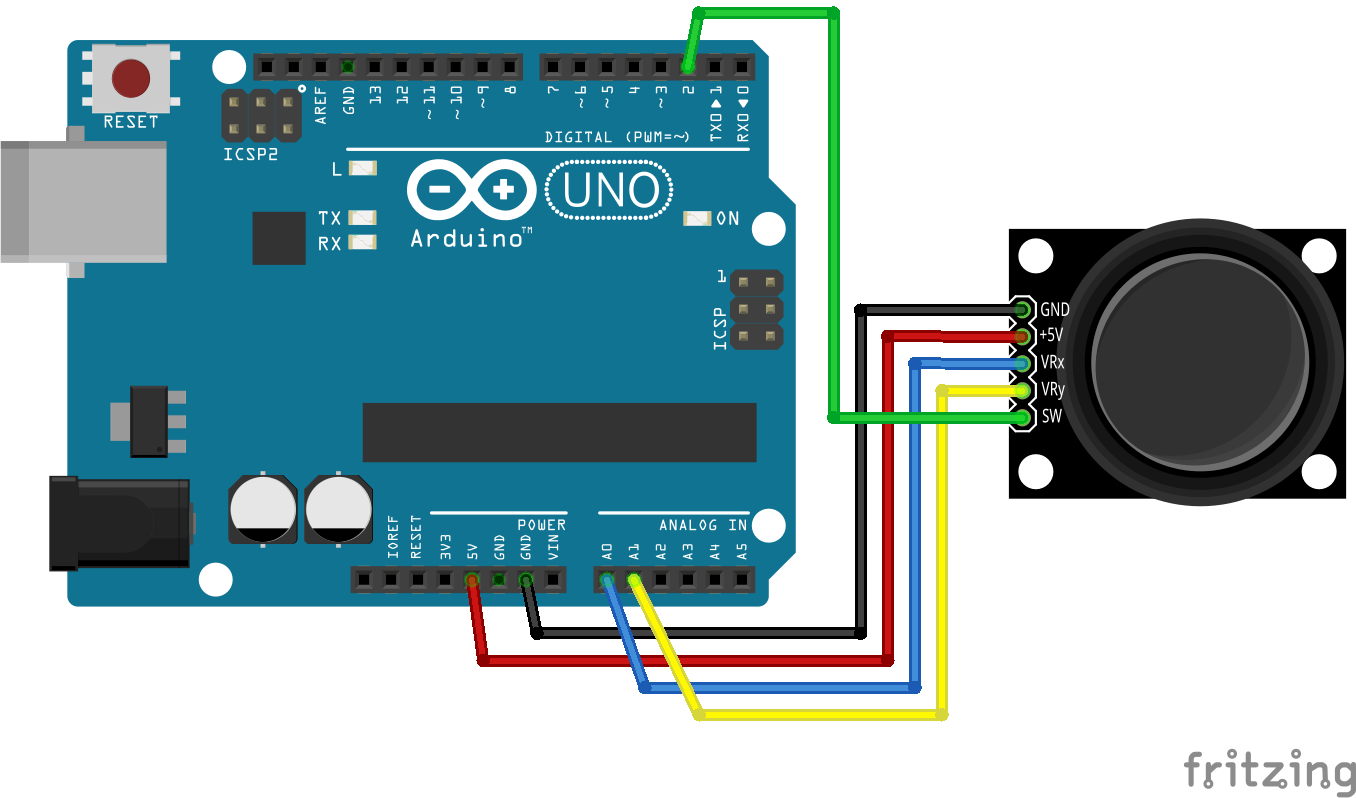
**Partes o piezas/componentes usados para este trabajo junto a su función y el para que los llegamos a utilizar.**

****

**El joystick para Arduino es un dispositivo que nos permite controlar los servomotores para así lograr mover el brazo o mejor dicho los servomotores, dependiendo de para donde muevas la palanca si para un lado o para el otro.**

**En Arduino, los joysticks se conectan a la placa a través de dos pines analógicos. Uno lee el movimiento hacia la izquierda/derecha y el otro lee el movimiento hacia arriba/abajo. Estos movimientos funcionan como los ejes cartesianos la “Y” sería el movimiento de arriba y abajo mientras que la “X” sería ya el movimiento de los costados osea de derecha a izquierda.**

**Por ejemplo, si usas un joystick para controlar un servomotor, cuando mueves la palanca del joystick hacia la derecha, el servomotor se moverá en esa dirección. Y cuando la mueves hacia la izquierda, el servomotor se moverá en la dirección opuesta, y claro también está el pulsador que es cuando lo presionas este suele usarse más cuando uno necesita es por ejemplo abrir y cerrar una garra o alguna cosa parecida ya que este solo sirve cuando lo presionas como si fuera un botón cuando lo presionas hará una cosa y cuando lo sueltes o dejes de presionar dejara de hacer lo que hacía como en los videojuegos.**

****

**Esta serían las conexiones de cómo quedaría el joystick y como van los cables a cada lugar conectado a la placa de arduino, en este caso el joystick tiene 5 conexiones.**

* **De estas 5 conexiones que llega a tener 2 van para lecturas analógicas o más exactos los A0, A1, etc.**

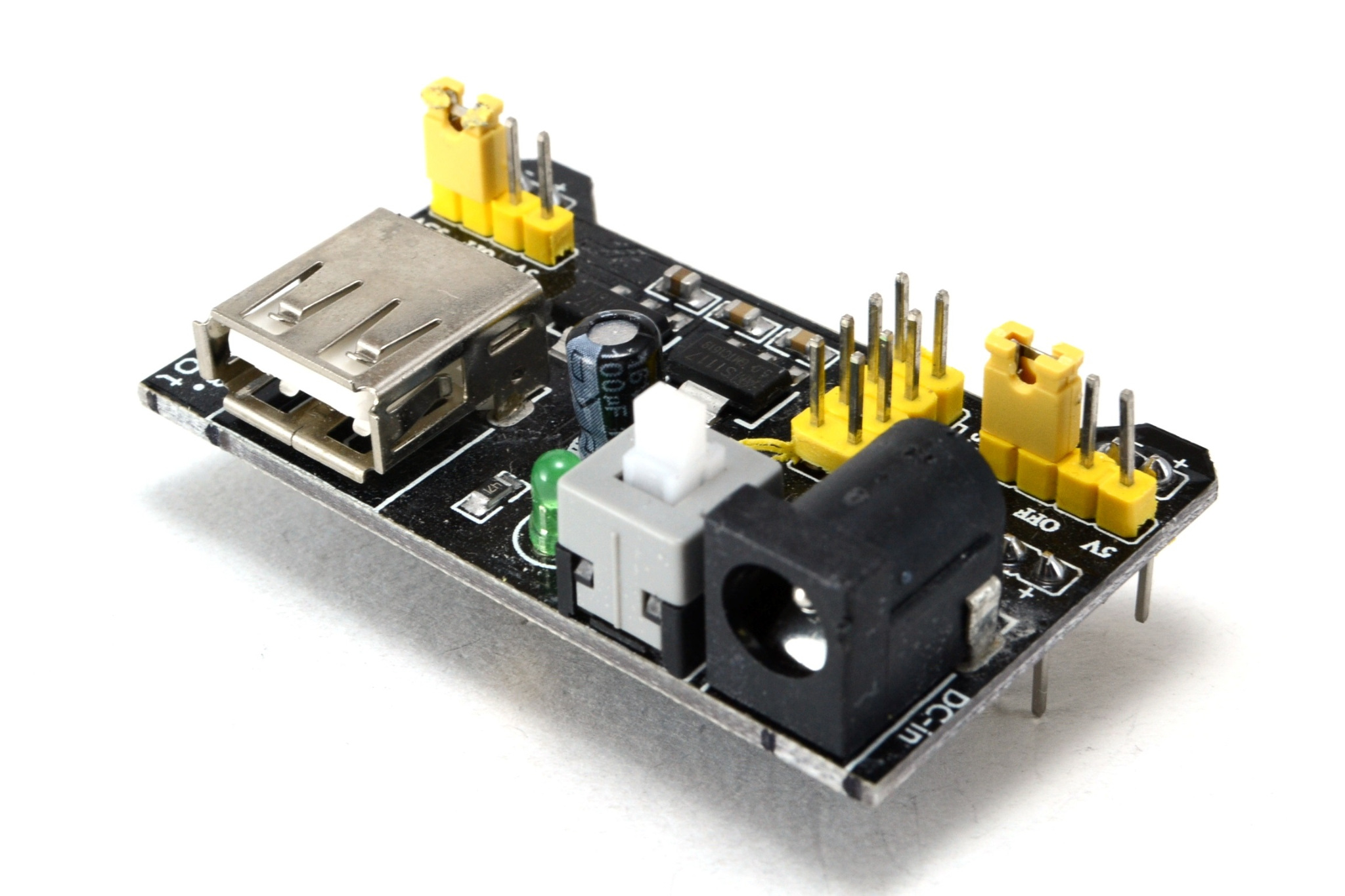
**En este caso esas dos conexiones son de los ejes X e Y que marcarían digamos para que lado va cada uno o mejor dicho los ejes X e Y como ya se mencionó arriba en la explicación del Joystick ensi.**

* **Luego está una conexión que va al GND y la otra que iría al 5V**

**Estas conexiones son típicas que prácticamente todo componente suele usar para recibir la cantidad de energía requerida para su funcionamiento.**

* **Luego por último pero no menos importante está la conexión “SW”**

**Este iría para un pin digital, este pin no siempre se suele usar, ya que realmente este se encarga de devolver un 0 o 1 dependiendo si el Joystick está siendo presionado o no estilo HIGH y LOW o mejor dicho se suele usar más cuando vos queres o necesitas usar el hecho de presionar el joystick.**

****

**Bueno, en nuestro caso llegamos a usar esta fuente de poder de alimentación externa por un tiempo ya que la potencia que nos daba la placa de arduino no era suficiente para nuestro trabajo o mejor dicho para todos los componentes que requerimos utilizar en nuestro proyecto por eso fue que tuvimos que recurrir a una fuente externa.**

**Como mencione igual fue por unas clases ya que nos dimos cuenta que al final tampoco nos daba la potencia suficiente ya que si la verdad es que en nuestro caso usábamos 6 servo motores y 3 joystick exigiendo mucha potencia para la placa.**

**Esta fuente sirve es para eso realmente, para poder darnos más poder o potencia a la placa de arduino que luego es potencia pueda alimentar todo los componentes que uno llegue o necesite utilizar, no siempre se necesita usar sino que depende más de que tanto uses y que tanto puedan llegar a exigir esas piezas, si por ejemplo usas como 10 leds y 3 buzzers es posibles que llegues a necesitar usarla ya que entre todo eso usará una cantidad x de poder que puede que tu placa no llegue a tener lo suficiente para todo entonces allí uno recurriría a usar una fuente externa que puede ser esa o otra.**

****

**Bueno aca llegamos con el tema de los servos motores, este es uno de los servos que llegamos a utilizar para el movimiento de nuestras partes aunque bueno, los usamos más para los movimientos más tranquilos por eso es que en nuestro caso no llegamos a utilizar tantos estos servos para nuestro trabajo.**

**Ya en tema de porque lo usan pues estos servos son útiles en general ya que no exigen mucho para su uso o más exacto la potencia que llegan a pedir no es tanta siendo capaz de usar uno solo con la placa arduino sin necesitar de una fuente externa, haciendo que por eso su uso sea uno recomendado para cuando necesites o inicias con el tema del uso de los servos gracias a eso, a su fácil uso y a que no te llega a exigir mucha potencia.**

**Ahora hablando un poco mejor de qué hacen estos servos o bueno los de este tipo son un pequeño actuador rotativo o bien un motor que permite un control preciso en posición angular, este servomotor puede rotar de 0° hasta 180° lo cual lo hace utiliz si queres digamos que mover una pieza a los 180° o alguna distancia entre estos valores.**

**Estos servos se llaman SG90 servo motor arduino.**

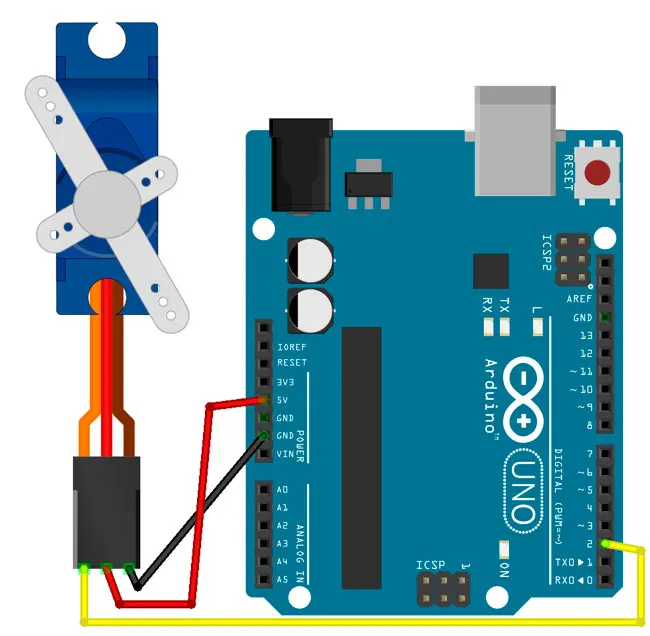
****

**Este sería ya el otro servo motor que nosotros llegamos a utilizar para este trabajo que en nuestro caso lo usamos ya más para los movimientos fuertes o mejor dicho las piezas y/o movimientos que más potencia van a poder necesitar para cumplir la función que le indiquemos.**

**Ya en tema de porque se suele usar pues estos servos son útiles en general por su potencia y los grados que van a poder llegar ya que en este caso este servomotor puede llegar a rotar mucho más que el que mencionamos antes y gracias a su potencia es capaz de mover piezas más grandes como en nuestro caso llegamos a usar.**

**Ahora hablando un poco mejor de qué hacen estos servos o bueno los de este tipo son un motor que permite un control preciso con los ángulos, este servomotor puede rotar de 45° hasta 360°lo cual lo hace utiliz si queres digamos que mover una pieza y que pueda llegar a dar la vuelta entera ya que esos 45° pueden llegar a ser modificados asi que si puede llegar a cambiar.**

**Este servomotor se llama Servo SG-5010.**

****

**Esta serían las conexiones de cómo quedaría en un servo motor y como van los cables a cada lugar conectado a la placa de arduino, en este caso el servomotor tiene 3 conexiones.**

* **Una conexión va al GND**
* **La otra que iría al 5V**
* **Y la última debería de ir a un pin digital o analógico.**

**Estas conexiones la de GND y 5v son típicas que prácticamente todo componente suele usar para recibir la cantidad de energía requerida para su funcionamiento junto a su digamos resistencia ósea positivo y negativo.**

**Luego por último pero no menos importante está la conexión que iría al pin digital o analógico, que este es más para poder llamar al servo motor cuando se le indique cumplir una función solo a ese claro cuando se le llame mediante el pin que esté conectado.**