

Tutorial teórico práctico con servos y Arduino.

[DronProfesional](#) / [Blog](#) / [Tutoriales, trucos, consejos, bricolajes](#)

/ Tutorial teórico práctico con servos y Arduino.



Tutorial teórico práctico con servos y Arduino.

📅 9 septiembre, 2019

5 / 5 (4 votos)

En esta ocasión presentamos, a nivel teórico y práctico, un tutorial con **servos y Arduino** muy completo y complementado con un videotutorial para que se entienda mejor.

Para ello estudiaremos algunos circuitos de ejemplos para ilustrar con diferentes tipos de necesidades de alimentación y diferentes **servomotores** y para lo cual usaremos un controlador Arduino a 5V.

📞 Llámanos



servomotores de uso corriente para distintas necesidades.

Índice de contenidos

- [Nuestro vídeo tutorial sobre servos.](#)
- [Qué es un servo motor \(servo\).](#)
 - [Partes de un servomotor.](#)
- [Razones de interés del uso de servomotores en drones.](#)
 - [Aplicaciones prácticas de servomotores en drones.](#)
 - [Servomotores y radio-control \(RC\).](#)
 - [Hablemos de los variadores ESC 4](#)
- [Par motor o fuerza de torque de un servo y otras consideraciones mecánicas.](#)
 - [Relación entre el ángulo de la fuerza y el torque.](#)
 - [Cálculo de la fuerza de torque de un servo en función de la posición y el ángulo de la fuerza.](#)
 - [Herrajes y partes mecánicas auxiliares.](#)
- [Señal PWM \(Pulse Width Modulation\) para servos.](#)
 - [Servos analógicos y servos digitales.](#)
 - [Servomotores analógicos compatibles con Arduino a 5v.](#)
- [La importancia de una buena alimentación en el uso de los servos.](#)
 - [Comprendiendo las necesidades de alimentación de los servomotores.](#)
 - [Uso de un condensador electrolítico de gran capacidad en paralelo con la fuente de alimentación.](#)
 - [Pruebas de esfuerzo de máximo torque.](#)
- [Como realizar el montaje de nuestro primer proyecto Arduino con servos.](#)
 - [Notas previas al montaje de nuestro prototipo con Arduino Pro Mini.](#)
 - [Como alimentar Arduino.](#)
 - [La librería de Arduino para servos.](#)
 - [Conexión de un servomotor a un Arduino.](#)
 - [Tutoriales para uso de servos en Arduino.](#)
- [Ejemplos que ilustran la versatilidad del control de servos con un sistema basado en Arduino.](#)
 - [Algunos ejemplos prácticos con servos.](#)
 - [Circuito 1: Lo que no se debe hacer.](#)
 - [Circuito 2: Alimentando el circuito a través de un adaptador USB-FTDI.](#)
 - [Circuito 3: Alimentando servos más potentes.](#)
 - [Circuito 4: Usando una fuente entre 7V y 12V y un regulador de voltaje 7805V.](#)
 - [Circuito 5: Usando una fuente a 12V y un servo a 12V.](#)
 - [El fuente del programa es el siguiente:](#)



Llámanos



Complementar este artículo. Documentaremos todo lo relativo a la alimentación de los servos con algunos ejemplos en el siguiente **videotutorial**.

Videotutorial con servomotores y Arduino



Visto el vídeo empecemos por el principio.

Qué es un servomotor (servo)

Los servomotores son un tipo de motores que tienen un propósito diferente al de aquellos que son capaces de permanecer girando por tiempo indefinido.

Los servomotores son gobernados electrónicamente ya sea conectados a un receptor de radio control, o a un controlador electrónico como **Arduino**.

Todos los motores gobernados por un controlador electrónico se consideran dispositivos actuadores. Esto significa que estarán conectados en algún conector de salida de la controladora y tras recibir algún tipo de señal eléctrica permiten realizar algún tipo de acción.

Los cuadricópteros suelen tener una controladora. Algunos incluso usan controladoras basadas en Arduino.

Existen además de los motores otros tipos de actuadores tales como:

- Emisores de luz tales como diodos leds.
- Emisores de sonido tales como zumbador electromagnéticos o piezo-eléctricos.
- Conmutares de circuitos (relés electromecánicos).



Llámanos



indicamos el ángulo de giro deseado del eje del motor y el servo se encarga de posicionarse en este ángulo. Tienen internamente unos topes y su giro está generalmente limitado de 0 a 180° (otros modelos más atípicos pueden realizar giros diferentes tales como los de 360 grados).

Nos vamos a centrar en los aspectos generales, pero actualmente disponemos en el mercado de una gran variedad de servos de diferentes tamaños, aplicaciones, tensiones, fuerza de torque, velocidad de giro, formas de control y por supuesto precios.

Por ello, aunque los servomotores ya no existan físicamente en muchos cuadricópteros, hablar de todo ello nos da un contexto que nos ayuda a conocer el por qué de algunos protocolos de comunicación. En el caso de los motores de un cuadricóptero también se usan señales PWM para controlar las revoluciones de cada motor.

A modo de práctica, en este artículo vamos a proponer el primer montaje de un prototipo muy sencillo usando un controlador Arduino para gobernar un servo.

Partes de un servomotor

Las partes principales internas de un servomotor son:

- Un par de topes: Simplemente limitan mecánicamente el giro del eje.
- Un mecanismo reductor: Consta de una serie de engranajes para desmultiplicar la velocidad de giro del motor. Esto hace que el giro sea más lento pero también más potente (proporcionan un alto par) y más preciso(pueden llegar a moverse solo a décimas de grado).
- Un controlador electrónico: Recibe una señal modulada por ancho de pulso (PWM), la descodifica y la compara con la posición de giro del eje indicada por un sensor y actúa en consecuencia para llevar el eje a la posición indicada por la señal (PWM).
- Un potenciómetro: Hace las veces de sensor del ángulo de giro. El controlador lee la posición del potenciómetro porque a partir de esa posición se puede saber la posición del eje ya que están mecánicamente vinculadas entre sí.

Todo ello simplifica el uso de los servomotores ya que el control del giro se realiza internamente. Basta indicarle al servo la nueva posición y se moverá hasta ella desde la posición actual. Esa indicación, como ya hemos dicho, se hace mediante un tipo de señal (PWM).

Razones del interés para el uso de servomotores en drones



Llámanos



voladores.

- Si bien muchos cuadricópteros no necesitan usarlos, algunos otros tipos de drones sí que los usan.
- La señal PWM para servos es la que entregan los receptores de radio-control a la controladora y también esta señal es la que usa la controladora para regular la velocidad de los motores.
- Algunos accesorios para cuadricópteros sí que pueden hacer uso de servomotores.

Aplicaciones prácticas de servomotores en drones

Gracias al uso de señales PWM en las receptoras de radio-control, no todo tiene por qué funcionar a través de la controladora de vuelo.

Podemos usar canales extras del receptor de radio-control y enchufarlos directamente a algún servomotor que controle algún mecanismo auxiliar, como por ejemplo la activación de un tren de aterrizaje, una llave de paso para inflar un flotador y recuperar un dron hundido (no es ninguna tontería ¿verdad?), abrir o cerrar una llave de paso para insuflar insecticidas, la liberación de un paracaídas para evitar estamparse contra el suelo, o cualquier dispositivo mecánico adicional que se le pase por la cabeza y que no suponga un peso excesivo para su dron.

Servomotores y radio-control (RC)

Antiguamente un vehículo teledirigido usaba únicamente como elementos electrónicos de control un receptor que controlaba directamente los servos adecuados. En los modelos de aeronaves que funcionan con gasolina o diésel, sí que se usa todavía un servo para accionar el acelerador, y si son de tipo ala fija también se usan para mover tanto el timón de cola como los alerones.

Se suelen transmitir esos movimientos mediante varillas y cables desde los servomotores. En realidad, el radio-control es previo al uso de las sofisticadas controladoras que tenemos hoy en día, y dependiendo de su sofisticación, estas controladoras tienen precios que van desde cantidades bastante asequibles, como otras con precios realmente elevados.

El tipo de señal para el control de servos mediante **pulsos de anchura variable** (PWM) se ha mantenido en estas receptoras porque se ha convertido en un tipo de señal eléctrica estándar muy sencilla y perfectamente funcional. Los controladores pueden traducir diferentes señales recibidas usando una gran variedad de protocolos de comunicación, pero los receptores de radio-control (RC) continúan usando estos estándares típicos para el control de servos.

Tal como dijimos las controladoras pueden usar diferentes tipos de señales y diferentes protocolos de comunicación entre la controladora y otros dispositivos. **El PWM para Servo se**

Llámanos



sencillo y bien definido.

Hablemos de los variadores (ESC)

Este es un tema al margen de los servomotores, pero conviene aclararlo para tener una mejor visión de conjunto.

Como intermediario entre la señal PWM y la señal de potencia trifásica de los motores de un cuadricóptero, se usa un elemento llamado variador (ESC).

Estos son elementos generalmente inteligentes que merecen un tratamiento aparte. Por ello, ahora simplificaremos, diciendo que la función principal de los variadores es la de proporcionar potencia trifásica a los motores a partir de la indicación recibida en forma de señal PWM.

Esta señal es entregada en algunos pines de salida de la controladora del dron y existe un minicontrolador en el propio variador que es capaz de reconocer comandos codificados en PWM. Por ello algunos variadores necesitan ser configurados mediante estos comandos para así poder activarse o para ser calibrados.

Resumiendo, los variadores son generalmente elementos inteligentes que son gobernados por la controladora del dron. Ambas partes (controladora y variador) son inteligentes y necesitan usar un mismo protocolo de comunicación para ponerse de acuerdo y entenderse.

Podréis entender mejor todo esto en un vídeo que explica estos conceptos de [radio-control para principiantes \[ENG\]](#).

Par motor o fuerza de torque de un Servo y otras consideraciones mecánicas

El par motor de un servo, o torque de un servo, se mide habitualmente en Kilos por centímetro. Si un servo es de 1 Kg x cm, significa que podrá levantar un peso de 1 Kilo con una carga situada como máximo a una distancia de un centímetro del eje de giro.

Si se coloca a más distancia, no tendrá fuerza para levantarlo. Los servos, tal como ya hemos dicho, son motores que para lograr su fuerza, usan un sistema de reductora por engranajes. En otras palabras, reducen la velocidad de giro en favor del aumento de la fuerza de torque.

Relación entre el ángulo de la fuerza y el torque.

En la mayoría de los sitios se describe el torque de un servo en los términos que acabamos de explicar. Lo que conviene tener claro es que la fuerza es un vector.

¡Llámanos



servo.

Si nosotros estamos sujetando un cubo de agua con el brazo hacia abajo. Necesitaremos muy poca fuerza para separar un poco el cubo del cuerpo y que no nos salpique el agua al andar, pero si intentamos continuar levantando el cubo con los brazos en cruz (levantados hasta la altura del hombro), en esa posición el brazo quedará en horizontal y perpendicular a la fuerza de la gravedad.

El torque a 90° tendrá un valor máximo. Concretamente el torque será el peso del cubo multiplicado por la longitud de nuestro brazo. Si hacemos mucho el bestia y superamos nuestro límite, podemos rompernos el tendón del hombro y terminar en el quirófano.

Si no somos muy fuertes no podremos subir tanto el brazo porque el torque requerido es proporcional al seno del ángulo por la posición del brazo con la vertical, y a 90° el seno valdrá 1 (que es el máximo valor para el seno de un ángulo). Veamos a continuación esto con más detenimiento ilustrando el concepto con una imagen.

Cálculo de la fuerza de torque de un servo en función de la posición y el ángulo de la fuerza.

Veamos en la imagen siguiente, lo que ocurre en un servo con su torque al variar la posición y el ángulo de la fuerza. Para entenderlo es casi imprescindible ilustrar este concepto con una imagen.

Tenemos dibujadas en dicha imagen un par de posiciones del brazo a 90° y a 30° .

Sabiendo que ese servomotor tiene un torque máximo de $2\text{Kg} \times \text{Cmt}$, podremos colgar como máximo en el extremo del brazo de 2 cmts de largo una pesa de 1Kg. (¡OJO! sería una situación límite al igual que la del resto de las pesas).

Con ese peso podremos elevar el brazo hasta cualquier posición, ya que donde más fuerza tendrá que hacer será con el brazo a 90° y tendrá la fuerza justa para llegar a esa posición.

Si pusiéramos un peso de 2Kg en el extremo del brazo no podríamos elevar el brazo más que hasta su posición de 30° (el $\sin(30^\circ) = 1/2$).

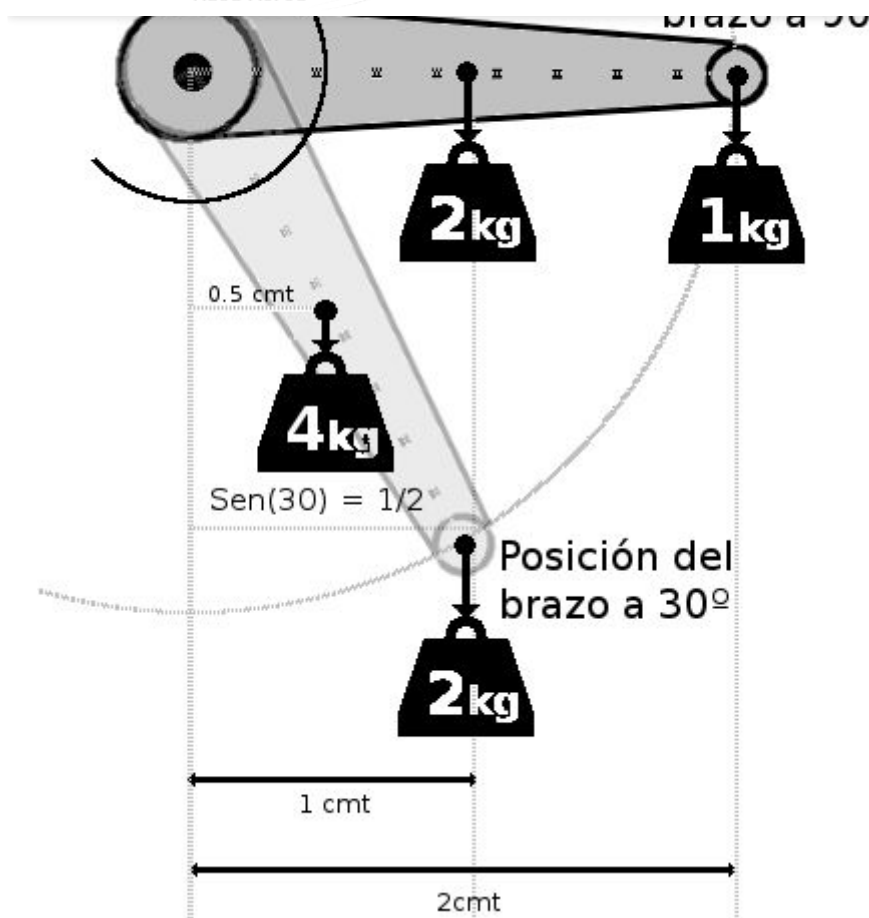
Podemos ver en la imagen que colgando el peso más cerca del eje de rotación, a la mitad de la longitud del brazo, el servo podrá levantar el doble de peso.

En la imagen hemos colocado cuatro pesas con distinto peso y en distintas posiciones de forma que todas ellas están ejerciendo idéntica fuerza de torque. La ilustración muestra cuatro alternativas posibles para un mismo valor de torque.

Existe un caso en el cual la fuerza de torque es cero. En la imagen no hemos representado la posición del brazo a cero grados (es decir, colocado en vertical hacia abajo).



Llámanos



servo de la imagen en dos situaciones diferentes. Una de ellas será con el brazo a cero grados:

1. Hasta una posición de 60° con 2kg en su extremo sería: $2\text{Kg} * 2\text{Cmt} * \sin(60^\circ) = 2\text{Kg} * 2\text{Cmt} * 0,866 = \mathbf{3.464 \text{ Kg} \times \text{Cmt.}}$
2. **Con un ángulo a 0°** , (brazo en vertical hacia abajo, que tampoco lo hemos dibujado en la imagen), **el torque necesario sería cero** porque $\sin(0^\circ)=0$. Es decir, el torque valdría **0 Kg * Cmt.**

Expresado de otra forma, en la segunda posición un peso se puede sostener sin fuerza ninguna y se puede empezar a mover con una fuerza muy

pequeña que irá aumentando proporcionalmente al seno del ángulo formado por el brazo.

Hay que tener en cuenta que existen fuerzas de fricción en los engranajes para mover el servo que en estos ejemplos no estamos considerando.

Si planteamos este ejercicio al revés, podríamos calcular el máximo valor del torque de un servo haciendo que levantara un peso desde el brazo en posición cero, girando hasta donde pueda.

Si el servo no es suficientemente potente la posición del brazo no pasará de un determinado ángulo. El servo se detendría al alcanzar su máximo valor de torque (lo cual no debe intentarse porque forzaría peligrosamente el motor).

El torque conseguido serán los kilos por la longitud del brazo por el seno del ángulo alcanzado. Esto es en teoría, porque en la práctica si llevamos a un servo a su máximo torque y lo aguantamos más de un segundo probablemente lo que conseguiríamos es averiarlo. Tenga en cuenta que algunos servos se quemarán casi instantáneamente.

Herrajes y partes mecánicas auxiliares

Los servomotores deben unir partes móviles entre sí haciendo la función de articulación electromecánica en muchas ocasiones y para ello se recurre a una serie de herrajes para las

Llámanos



(40mm..40.5mm)x(19.5mm..20.5mm)x(35mm..45mm). Faltaría conocer las medidas y posición de los cuatro agujeros para usar tornillos de sujeción.

Para ello podemos ver la descripción de las medidas del [Servo HS-311 \(servo estándar con todas sus medidas perfectamente detalladas\)](#).

Muchas veces lo único que hace falta es poder sujetar de una forma rígida y segura el servo a una superficie. [BANCADA SERVO ALUMINO AZUL 6STARHOBBY \(Un soporte estándar muy sencillito\)](#).

Un segundo ejemplo de soporte sencillo sería este: [Soporte Multipropósito Para Servo Estándar Mg995 \(40.6mm x 19.8mm x 42.9mm\)](#)

Hemos descrito el uso de herrajes de propósito general, porque esto nos proporciona mayor versatilidad, pero para la realización de proyectos menos originales es muy frecuente recurrir al uso de kits que ya vienen con la parte mecánica, los servos, tornillería, controladora, fuente de alimentación, etc.

Señal PWM (Pulse Width Modulation) para servos.

Siempre que hablemos de PWM estamos hablando de una señal que a intervalos de tiempo regulares, (intervalos cíclicos), entrega pulsos con una forma de onda cuadrada (o rectangular) y con una anchura que puede ser variada para que su valor se situé entre el 0% del periodo del intervalo de tiempo en un ciclo completo, o el 100% del periodo del mismo.

Así dicho queda algo confuso. Entenderemos mejor esto más tarde con algunas ilustraciones. Una señal PWM como la que acabamos de describir se podría usar para hacer que un LED varíe su luminosidad entre el 0% de la potencia y el 100% de su potencia.

Es como si electrónicamente y con una frecuencia alta que hace imperceptible el parpadeo del led, estuviéramos encendiendo y apagando el LED con un porcentaje de tiempo apagado y un porcentaje de tiempo encendido que podemos variar entre el 100% del tiempo encendido y el 100% del tiempo apagado.

Para lograr una potencia media usaríamos el 50% del tiempo apagado y el otro 50% del tiempo encendido. Las situaciones extremas no nos servirían para gobernar un servo, ya que tanto con el 100% del tiempo encendido, como con el 100% del tiempo apagado, la onda desaparece y la electrónica pierde el sincronismo de la señal.

Por ello, a diferencia de una señal PWM genérica, que acabamos de describir y que es utilizable para gobernar un LED, en la **señal PWM para servos**, existen una serie de restricciones que definen este estándar como un caso particular de señal PWM genérica. Para mantener un servo en una posición fija, lo que se hace es mantener fija la forma de esa onda PWM.

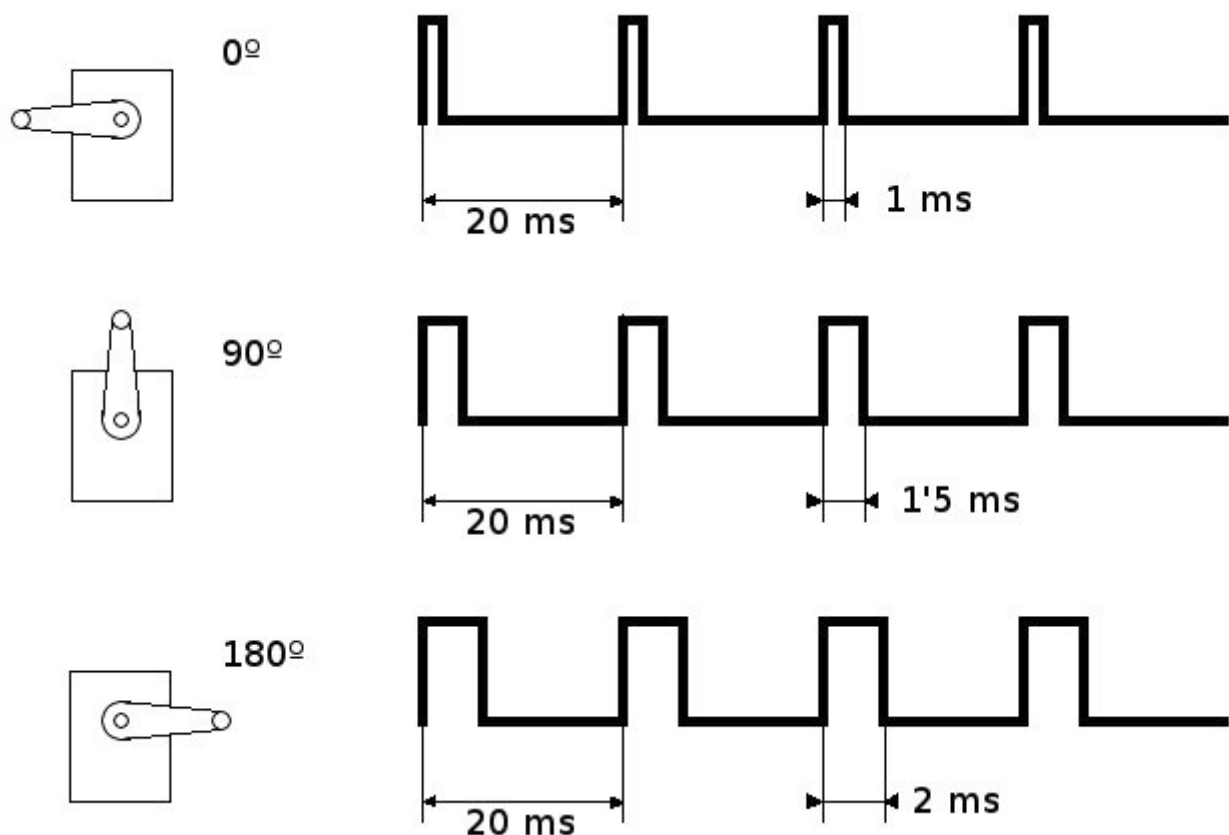
Llámanos



sería el periodo).

También podemos expresar lo mismo diciendo que, la frecuencia de 50Hz para el ciclo de trabajo significa que la señal se repetirá de forma cíclica cada 20 milisegundos. Este es el primer requisito del PWM para servos, pero tiene algunos requisitos más que vamos a explicar en detalle.

Cuando la señal PWM viaja de un punto a otro, la recepción de la misma necesita establecer unos puntos de sincronismo para decodificar y así poder medir correctamente la anchura de los pulsos. En los servos más corrientes, donde las posiciones van de 0 a 180°, la correspondencia entre los grados y el ancho del pulso es la que podemos apreciar en la imagen siguiente:



Este es el caso más habitual pero dependiendo de los servos, puede haber variaciones.

- 0° = Generalmente pulso de 1ms (dependiendo del servo podrías ser de 0.5ms).
- 90° = pulso de 1.5ms.
- 180° = Generalmente pulso de 2ms (dependiendo del servo podrías ser de 2.5ms).

Así pues, el ciclo de pulsos suele durar entre 15 y 25 milisegundos dependiendo del servo, pero generalmente, como ya hemos dicho, es de 20 milisegundos (en microsegundos serían 20.000 μ s).

Si pasa un tiempo superior a 20ms (es decir, si para un ciclo de 20ms transcurre un tiempo de 21 milisegundos) sin recibir pulsos, el servo entrará en reposo y su consumo bajará de forma muy

Llámanos



para determinados casos, pero no para otros.

Para forzar el máximo ahorro de energía por software, en la librería Arduino existe un comando que desconecta el servo, pero para garantizar activamente la permanencia del servo en una determinada posición, hay que mandar continuamente la señal PWM con el ancho de pulso correspondiente al valor del ángulo requerido.

El controlador en el interior del servo siempre estará verificando la posición del mismo y en caso necesario llevará nuevamente el eje a la posición de giro correcta.

Una vez alcanzada la posición correcta, la parte del motor (que es la que más energía gasta) se detiene y deja de consumir, pero el controlador interno (salvo que desconectemos por software el servo) seguirá verificando la señal PWM de entrada y la cotejará con la posición actual del eje para en caso de algún desvío corregirlo.

Por ello, en la mayoría de los casos lo habitual es mantener el servo siempre activo. El consumo en reposo de un servo es muy bajo.

Si el servomotor debe mantener su posición pese a una fuerza ejercida sobre su eje que supere la capacidad de retención del servo en reposo, el motor quedará vibrando haciendo fuerza y consumiendo electricidad.

Hacer esto por un breve espacio de tiempo puede ser admisible, pero tener el motor del servo siempre haciendo una fuerza para contrarrestar a una fuerza contraria, ya sea un tope, un muelle o la gravedad, puede provocar un desgaste excesivo o incluso un fallo del servomotor por excesivo esfuerzo.

Servos analógicos y servos digitales

Los servos más normalitos son los servos analógicos. A nivel de usuario no presentan grandes diferencias, ya que ambos se controlan con las señales PWM que ya hemos mencionado.

Los servos digitales llevan internamente microprocesadores que los hacen más versátiles, a diferencia con los analógicos donde sus parámetros son fijos.

En los servos digitales se pueden configurar varios parámetros para su funcionamiento tales como: posición central inicial, sentido de giro, topes en el recorrido del servo, velocidad de respuesta del servo y resolución, etc. Además pueden trabajar con señales PWM con una frecuencia 10 veces más alta y tienen un menor tiempo de respuesta.

Estos servos digitales son bastante más sofisticados y más complejos. Muchos necesitarán controladoras especiales. Las librerías de Arduino están pensadas para su uso con servos analógicos normalitos, así que la compatibilidad con servos digitales no está asegurada.

Llámanos

Servomotores analógicos compatibles con Arduino a 5v.

Lo primero que hay que advertir es que, con excepción de los microservos, alimentar un servo mediante un adaptador USB no es buena idea, ya que estos adaptadores no suelen entregar más de 500mA.

Ocurre algo parecido si intentamos alimentar un servo a través del regulador de de tensión interno incluido en el módulo Arduino. Salvo que se trate de un microservo, hará falta alimentarlo con una fuente capaz de entregar corrientes algo más fuertes.

Por ello si alimentamos el Arduino a través del pin *Raw* del módulo (se hace así cuando se usa un voltaje de entre 6.5 y 12v) y sacamos los 5v del pin *VCC* del mismo módulo, cuando el servo se active puede dejar al Arduino sin suficiente corriente ya que el regulador de voltaje de la placa Arduino no es capaz de entregar mucha intensidad.

En los circuitos de ejemplo, al final del artículo, se entenderá más fácilmente






En la tabla siguiente vienen las principales características de los servos que hemos considerado más interesantes para aplicaciones habituales de robótica aunque hay una gran variedad de ellos.

Observe que la velocidad en la tabla siguiente se expresa en segundos/60° (tiempo que tardan en efectuar un giro de 60°).


En realidad, expresado así, es la inversa de la velocidad de giro porque en lugar de espacio/tiempo, es tiempo/espacio y la velocidad de giro será más alta con el valor más bajo en segundos/60°.

Otro detalle es que en esta tabla no se menciona el consumo siendo este dato muy importante, pero más adelante explicaremos el motivo.

Modelo Servo	Torque/Voltaje	Velocidad de giro	Ángulo Máximo	Tamaño	Peso
	1.6Kgxcmm (4.8v)	0.12s/60° (4.8v)	180°	23mm x	9g
	1.8Kgxcmm (6v)	0.10s/60° (6.0v)		12.2mm x 29mm	
SG90: Es el servo que más se vende y el más barato. Es un microservo muy ligero y  muy popular. Tiene engranajes de plástico					
Llámanos					


		(6v)		28.5mm	
MG90	MG90: Muy similar al primero pero más robusto y más rápido. Tiene engranajes metálicos.				
	3.3Kgxcn (4.8v)	0.21/60° (4.8v)	180°	40.39mm x	45.5g
	4.1Kgxcn (6.0v)	0.16/60° (6.0v)		19.56mm x 36.58mm	
Hitec HS422 (*)	Hitec HS422: Es un servo muy popular en la gama de los 45.5 gramos y el fabricante ofrece información bastante completa. (*) Sobre los consumos de corriente del servo Hitec HS422 véase más adelante.				
	8.5Kgxcn (4.8v)	0.20s/60° (4.8v)	120°	40.6mm x	55g
	10kgxcn (6v)	0.16s/60° (6v)		19.8mm x 42.9mm	
MG995	MG995: Es un servo muy popular en la gama de los 55 gramos. Más potente que los anteriores. Gran resistencia y fuerza. Tienen engranajes metálicos.				
	10.5Kgxcn (4.8v)	0.20s/60° (4.8v)	180°	40.7mm x	55g
	14Kgxcn (7v)	0.17s/60° (6.0v)		19.7mm x 42.9mm	
MG946R	MG946R: Es un servo de la misma gama de 55 gramos que el anterior, con similar velocidad, pero con más resistencia y fuerza.				
PDI-6221MG-	17.25Kgxcn	 0.18/60° Llámanos	180°	40.5mm x	62g





0

PDI-6221MG-180: Es un servo algo mas pesado, con mayor torque que el anterior y además más rápido.

 <p>RDS3115</p>	<div> 13.5Kgxcn (4.8v) 15Kgxcn (6v) 17Kgxcn (7.2v) </div>	<div> 6.8s/60° </div>	<div> 270° </div>	<div> 40mm x 20mm x 40.5mm </div>	<div> 55Kg </div>
<p>RDS3115: Un servo potente, ideal para soportes o brazos articulados gracias a poder acoplarse por ambos extremos de su eje a las piezas mecánicas que tiene que mover. Tiene gran torque, su velocidad de giro es bastante lenta pero para ciertas aplicaciones la lentitud puede incluso considerarse una ventaja.</p>					

Estos siete servos pueden dar mucho juego, pero si necesitas más información o servos para un voltaje diferente, puedes consultar este [Amplísimo catálogo de servomotores](#).

OJO! Las características indicadas más arriba, nosotros las consideramos orientativas. Algunos de estos servos tienen clones que se venden con el mismo nombre, pero las características reales en estos clones no serán las mismas dependiendo del fabricante.

La importancia de una buena alimentación en el uso de los servos.

Ya hemos mencionado la importancia de este aspecto en el uso de servos y lo que vamos a hacer ahora es explicar el motivo de ello.

Comprendiendo las necesidades de alimentación de los servomotores

Un conector USB normalito puede entregar unos 500mA a 5V. Eso sería suficiente para un microservo como el **SG90**.

Muchos alimentadores USB pensados para recargar dispositivos móviles están pensados para entregar ese tipo de carga y en previsión de que se les conecte algo con una demanda de corriente



llámanos



un servo rápido y potente lógicamente, se necesitará un mayor consumo de potencia eléctrica y generalmente eso pasa por un motor algo mayor.

Si nosotros conectáramos un servo algo más potente que el SG90 a uno de estos reguladores de baja potencia protegidos, puede que con el servo en reposo no notemos nada raro porque el consumo será bajo, pero cuando se intente accionar el servo y éste demande más potencia, el sistema de protección de la alimentación podría saltar en algún momento creando problemas.

Es decir, lo normal es que necesites buscar una fuente de alimentación apropiada a las necesidades de tu servomotor.

Por desgracia, muchos vendedores no proporcionan una información completa sobre el consumo del servo. En realidad, una sola cifra de consumo no sería suficiente y podría inducir a error porque el consumo de un servomotor varía mucho dependiendo del esfuerzo mecánico realizado en cada momento.

Los datos de consumo de servomotores tampoco aparecen en todas la hojas de especificaciones técnicas denominadas **datasheet**. Recientemente hemos localizado información relativa a consumos de corriente en uno de los servos compatibles con Arduino a 5V que ya hemos referenciado anteriormente. No hemos tenido ocasión de ponerlo a prueba.

Servo Hitec HS422	
Corriente en reposo	8 mA
Corriente en funcionamiento	150 mA
Corriente máxima	1100 mA

Observe la enorme diferencia del consumo de corriente de un servo cuando funciona sin carga a cuando lo hace a máxima carga. Faltaría saber durante cuanto tiempo este servo puede trabajar con la corriente a ese valor de máxima carga sin dañarse.

Probablemente sea menos de un segundo ya que una acción puntual de un servo se limita a cambiar de posición y volver al estado de reposo.

Los datos de consumo de un servo ni son fáciles de obtener ni son siempre todo lo claros o fiables que uno supone. Tenga cuidado, en productos baratos del mercado chino, la cifra de torque puede venir exagerada.

Llámanos



condiciones de trabajo concretas que necesitemos para asegurarnos que no tallara.

Siempre usaremos fuentes con voltaje adecuado, estabilizado y con protección para sobrecargas, (son muchas las fuentes de alimentación que incluyen estas protecciones).

Si deseamos proteger un servomotor frente a una sobrecarga, evitaremos elegir una fuente demasiado potente ya que la protección de sobrecarga no actuaría con la limitación de corriente que nosotros necesitamos y quemará el motor.

Un caso especial sería cuando deseamos alimentar un número elevado de servos con una única fuente de alimentación. Si lo calculamos para garantizar que todos los servomotores puedan actuar simultáneamente sin hacer saltar la protección de sobrecarga, no podremos proteger a los servos frente a un consumo individual excesivo durante un tiempo prolongado.

Es decir, cada caso requiere de un estudio individualizado.

Uso de un condensador electrolítico de gran capacidad en paralelo con la fuente de alimentación.

Volviendo al caso de la alimentación de un solo servomotor y si únicamente deseamos permitir corrientes de picos muy breves que sobrepasen el nivel de protección de nuestra fuente, podemos colocar en paralelo con la fuente de alimentación un condensador que permitirá absorber los picos de consumo.

El condensador conseguirá amortiguar los picos de consumo en la fuente evitando que salte el sistema de protección.

A más capacidad del condensador, más cantidad de carga podrá almacenar y mejor absorberá los picos de mayor carga. El voltaje de trabajo del condensador deberá ser igual o superior al voltaje de la fuente.

Para evitar el deterioro del condensador, mejor elegir uno algo sobrado de voltaje. Al permitir sobrecargas muy breves, inferiores siempre a un segundo, no pondrían en riesgo la integridad del servo ya que no tendría tiempo de calentarse excesivamente.

Todo esto hay que tenerlo en cuenta para poder sacarle el máximo partido a nuestro servo y para optimizar la compra de nuestros componentes.

Cuando no conocemos bien las características de un servo, resultará imprescindible hacer pruebas de consumo en diferentes condiciones, de forma que podamos averiguar como se comportará el servo más tarde en condiciones reales y que fuente de alimentación será la más adecuada.



Llámanos



admite el servo.

Deberías pensarte dos veces si quieres probar el servo a la máxima potencia de torque que vienen en las especificaciones técnicas.

En las especificaciones técnicas podrían referirse al máximo torque por un breve espacio de tiempo, por ejemplo medio segundo, y pasado ese tiempo manteniendo la alimentación sobre el servo, este podría quemarse sufriendo un fallo total o una pérdida de potencia importante.

La alternativa a realizar esta prueba es la de conformarse con probar el servo solo con una potencia de torque suficiente para tus necesidades.

Por el contrario, si pretendes comprar muchos servos y antes de eso quieres hacer una prueba de resistencia para averiguar sus límites, o si necesitas mucha seguridad en ese servo, puedes hacer la prueba de máxima carga y averiguar cuanto aguanta uno de ellos y arriesgarte a fundirlo.

Si se trata de un servo muy barato comprado en el mercado chino, puedes descubrir que no es tan fuerte como pensabas. Eso no le descalifica necesariamente, depende para qué lo necesites. Son por lo general baratos y pueden adecuarse a muchos proyectos.

Por todo ello, lo mejor es hacer prototipos y probar los componentes como mínimo en condiciones reales o incluso en condiciones ligeramente más duras.

Esto resulta imprescindible para garantizar la seguridad de nuestro desarrollo. Muchos bricolajes terminan fallando al cabo del tiempo porque las pruebas iniciales fueron insuficientes.

El desarrollo puede no ser todo lo robusto que uno supone por desconocimiento exacto de algún componente y su fiabilidad puede ser baja en determinadas condiciones.

Recuerda que no es lo mismo usar un servomotor para mover un robot de juguete que para que dispare un paracaídas de un dron que podría estar cayendo en una zona con personas que no llevan un casco puesto.

El nivel de pruebas no puede ser el mismo dependiendo de los riesgos en juego en caso de fallo del servo.

Como realizar el montaje de nuestro primer proyecto Arduino con servos

Todo esto ha sido teoría, pero no tenemos por qué conformarnos con la teoría.

Como ya hemos dicho, tenemos en el mercado un controlador Arduino del que ya hemos hablado y que nos permite poner en práctica todo lo que sabemos e incluso salir de dudas a la hora de probar un servo para saber como se comportará en las condiciones que nosotros pretendemos usarlo.



Llámanos



Notas previas al montaje de nuestro prototipo con Arduino Pro Mini.

Para este proyecto usaremos un **Arduino Pro Mini** que es totalmente compatible con el **Arduino Uno**, solo que el primero es más pequeño y para programarlo necesita un **adaptador USB-FTDI**.

Lo hacemos así porque lo que vamos a hacer es un prototipo y resulta muy cómodo poder pinchar el Arduino Pro Mini en una placa de prototipado también denominada **Protoboard**. Se podrían usar diferentes tipos de placas Arduino sin demasiado problema.

Como alimentar Arduino

Hay diferentes formas de alimentar un Arduino. Para ello hay dos pines que aceptan la entrada de la alimentación para Arduino.

Por un lado tenemos un pin denominado VCC que acepta el voltaje de trabajo del chip Arduino que generalmente es a 5v aunque hay algunas placas que trabajan a 3,3v. Este pin siempre debe de conectarse a una fuente que entregue un voltaje estabilizado.

Cuando la placa se alimenta enchufándola a un puerto USB recibirá 5v estabilizados como si los conectáramos a ese pin.

La mayoría de la placas Arduino incluyen internamente un pequeño regulador de voltaje que requiere a la entrada voltajes algo más altos que el voltaje de trabajo en VCC. Este pin se denomina RAW. Podría bastar alimentarlo con 6 voltios pero seguramente con ello forcemos a trabajar al procesador a un poco menos de sus 5v.

El voltaje de entrada recomendado para este pin RAW es de entre 7 y 12v. Una pequeña pila de 9v puede bastar para ello, pero una pila de este tipo puede que no sea suficiente para asumir la carga de algunos servos.

Profundizaremos algo más sobre alimentación de servos más adelante en este mismo artículo.

La librería de Arduino para Servos.

Estas librerías están descritas en <https://www.arduino.cc/en/Reference/Servo>. En ellas se describe cada una de las funciones de esta librería que son:

- **attach()** Asocia al servo un pin y lo prepara para actuar.
- **write()** Cambia el valor de la señal PWM para mover el servo a una determinada posición (habitualmente entre 0 y 180).

Llámanos



tránsito, sino la posición final).

- `attached()` Informa de si el servo está asociado al pin.
- `detach()` Desconecta y libera el pin del servo.

En la práctica con las dos primeras funciones se puede manejar el servo, ya que las restantes no tienen tanta utilidad.

Conexión de un servomotor a un Arduino.

Los servomotores habitualmente suelen tener tres cables. En un extremo tienen un cable negro o marrón que habrá que conectar al negativo de la alimentación y justo al lado el hilo central será rojo que habrá que conectar al positivo de la alimentación.

El hilo restante será un cable que llevará la señal PWM y que será de color amarillo, naranja o blanco. Las variaciones de todo ello dependen de las marcas de los servos.

Consulte la ficha de las especificaciones técnicas si tiene dudas. Usando servos a 5v y Arduinos a 5v podrá compartir la misma alimentación para el Arduino y para el servo pero recuerde que los servos pueden generar grandes picos de consumo de corriente en determinados momentos.

Tutoriales para uso de servos en Arduino

En el propio **IDE de Arduino** en la sección de ejemplos ya tiene un par de ellos muy sencillos para ilustrar el manejo de la librería.

- **Ejemplos->Servos->Knob.** Control de la posición del servomotor con un potenciómetro.
- **Ejemplos->Servos->Sweep.** Movimientos de ida y vuelta entre ambos topes de giro desde 0° a 180°.

Desde un punto de vista didáctico esos ejemplos sirven perfectamente para un primer contacto, pero nosotros vamos a proponer un ejemplo propio que es muy parecido al mencionado knob. Hemos preferido hacer lecturas promediadas del potenciómetro.

Ejemplos que ilustran la versatilidad del control de servos con un sistema basado en Arduino

Algunos ejemplos prácticos con servos

Recomendamos echar un vistazo a estos ejemplos: [TOP 10 Arduino-Servo Motor Projects 2018](#).

Llámanos



y amperajes, porque las combinaciones que se podrían considerar son demasiadas.

La idea es mostrar unas cuantas de ellas para ilustrar la versatilidad del uso de un Arduino en diferentes circunstancias de potencia, de voltaje de alimentación, o de servos de diferentes voltajes.

A efectos didácticos hemos preferido elegir un montaje muy sencillo en el cual usaremos un potenciómetro de entre 15K y 30K para entregar en un pin de entrada analógico del Arduino un voltaje de entre 0V y 5V que serán usados por el programa para determinar la posición de giro del eje del servo.

De esta forma, tendremos total control para forzar todo tipo de movimientos en el servomotor y observar su respuesta. En determinadas condiciones de alimentación y de carga en el servo podremos comprobar como reacciona nuestro servo frente a movimientos suaves o bruscos del potenciómetro.

Para estas pruebas no necesitamos hacer apenas soldaduras. Nosotros solo hemos realizado soldadura para hacer un pequeño adaptador a fin de enchufar el potenciómetro en la protoboard, pero podríamos haber utilizado para esas conexiones cable unifilar y pincharlo en la protoboard.

Circuito 1: Lo que no se debe hacer.

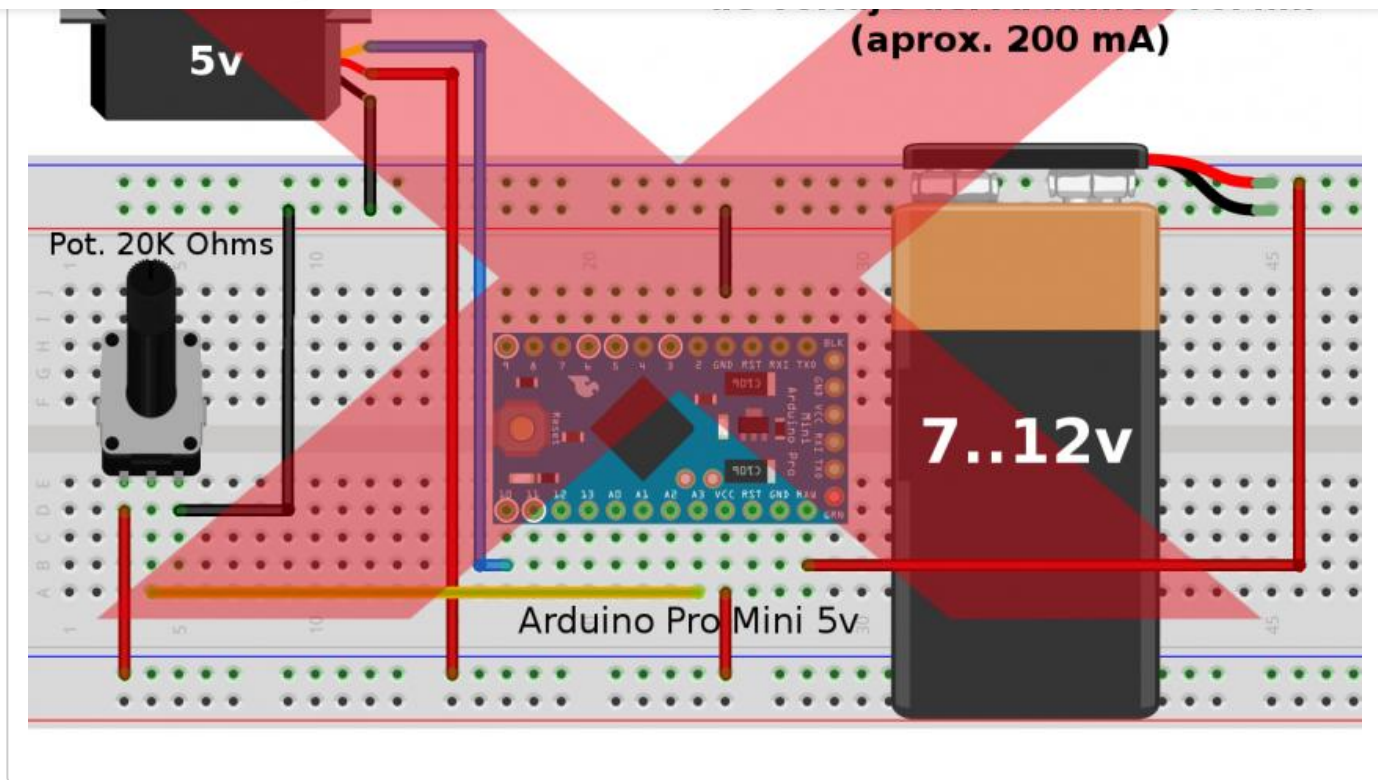
Empezamos explicando con un ejemplo de **lo que no se debe hacer**. Explicaremos un circuito alimentado con una fuente de entre 7V y 12V conectada al pin RAW.

Usaremos un servo de 5V. Se podría pensar que **podemos tomar el voltaje de 5V desde el pin Vcc** por ser un servo pequeño de bajo consumo, pero la realidad es que incluso este pequeño servo puede alcanzar picos de consumo de 250mA mientras que el regulador de Arduino Pro Mini no debería pasar de los 200mA.

Es decir, estaríamos forzando el regulador de voltaje interno del Arduino Pro Mini.



Llámanos



Circuito 2: Alimentando el circuito a través de un adaptador USB-FTDI.

En la siguiente imagen alimentamos el circuito a través de un adaptador USB-FTDI, que no solo nos proporciona los 5V ya estabilizados, sino que en caso de conectarlo a un ordenador nos permitirá ver en pantalla algunas trazas comunicando por el puerto USB el Arduino y el PC.

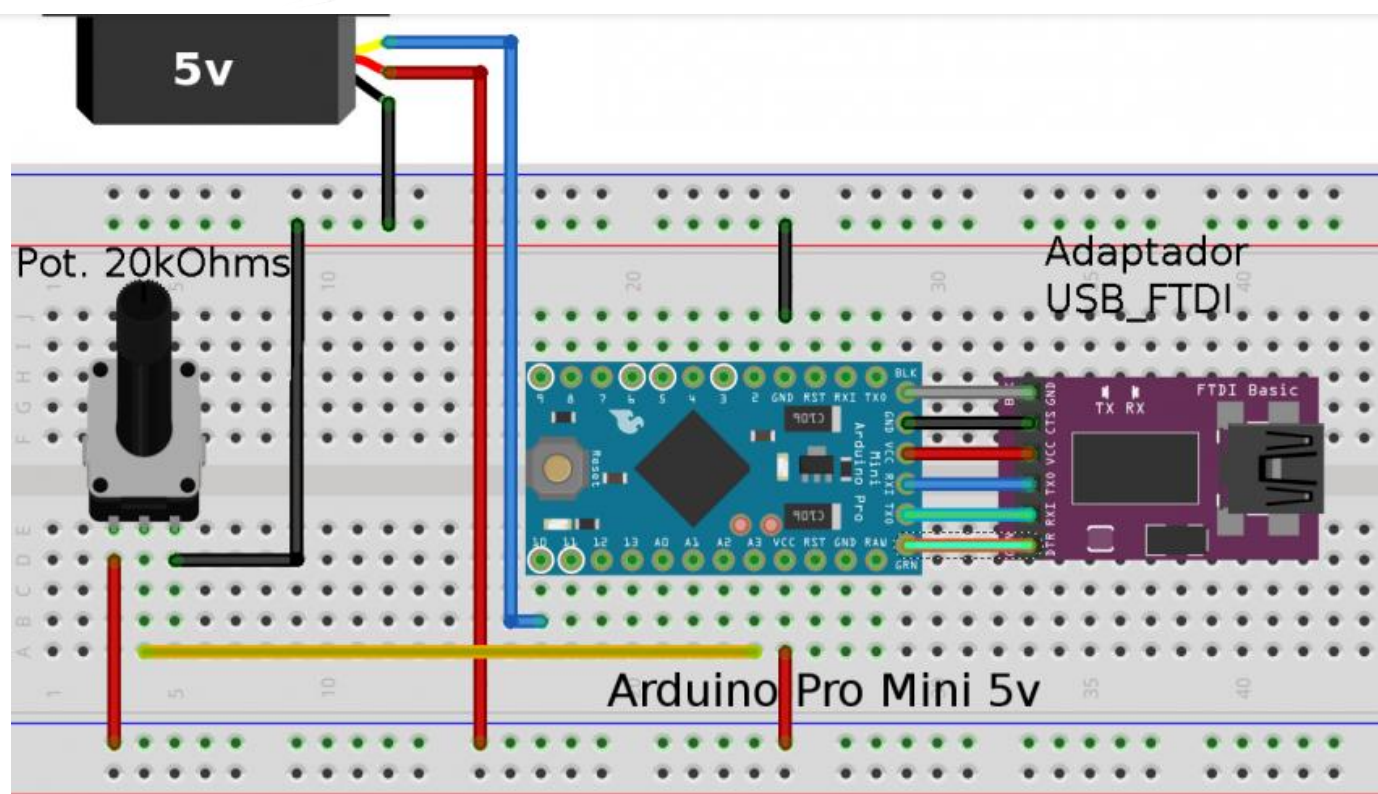
En principio, nada impide alimentar el circuito a través de un adaptador de corriente de 5V de los que se usan para cargar móviles por ejemplo, pero recuerde que la mayoría de los servos, exceptuando los microservos, necesitan una fuente de alimentación más potente.

¡OJO! Nosotros representamos en estos diagramas la fuente de alimentación con una pila, pero en realidad necesitaremos una fuente estabilizada de 5V con un amperaje adecuado para ese servo y además protegida contra sobrecargas.

Aunque no está indicado en la imagen, puede ser muy conveniente conectar un condensador electrolítico de unos 25V (mas vale que sobre) y unos 2000 microfaradios o incluso más.

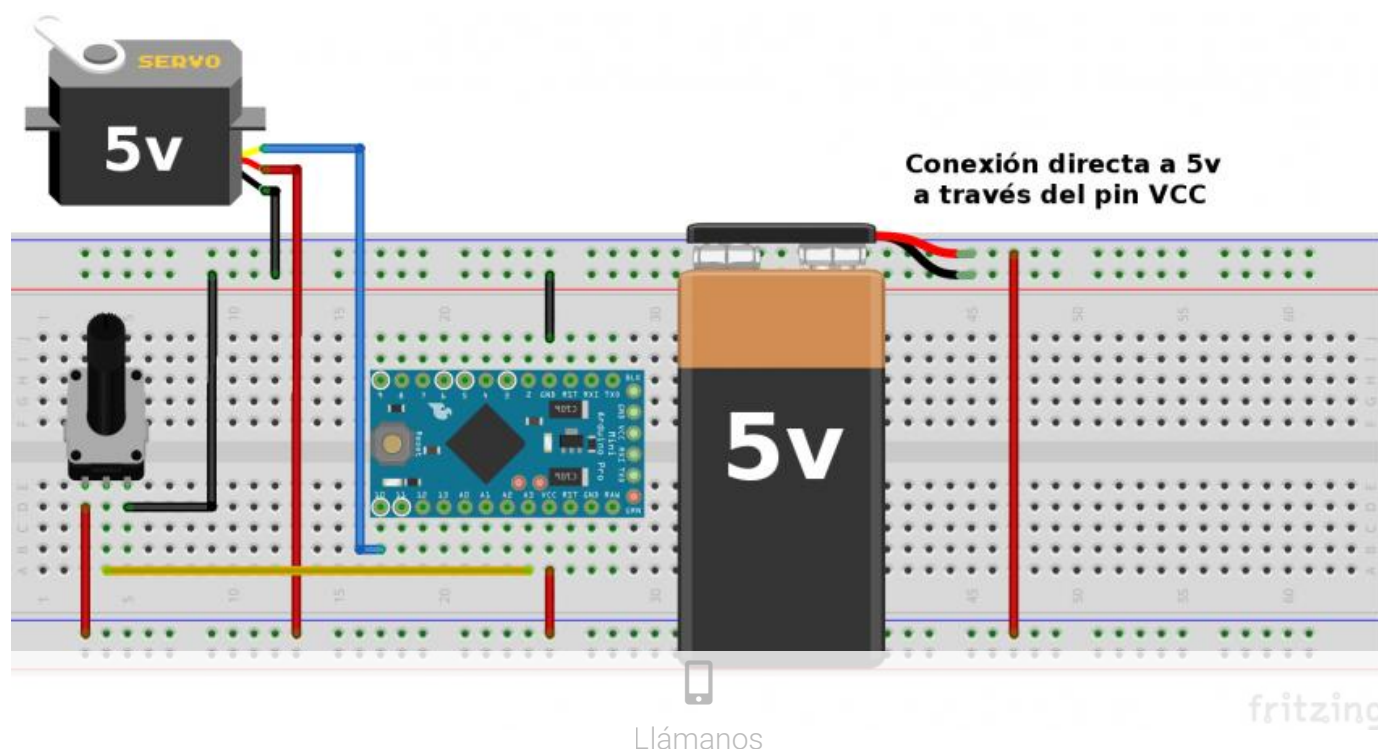


Llámanos

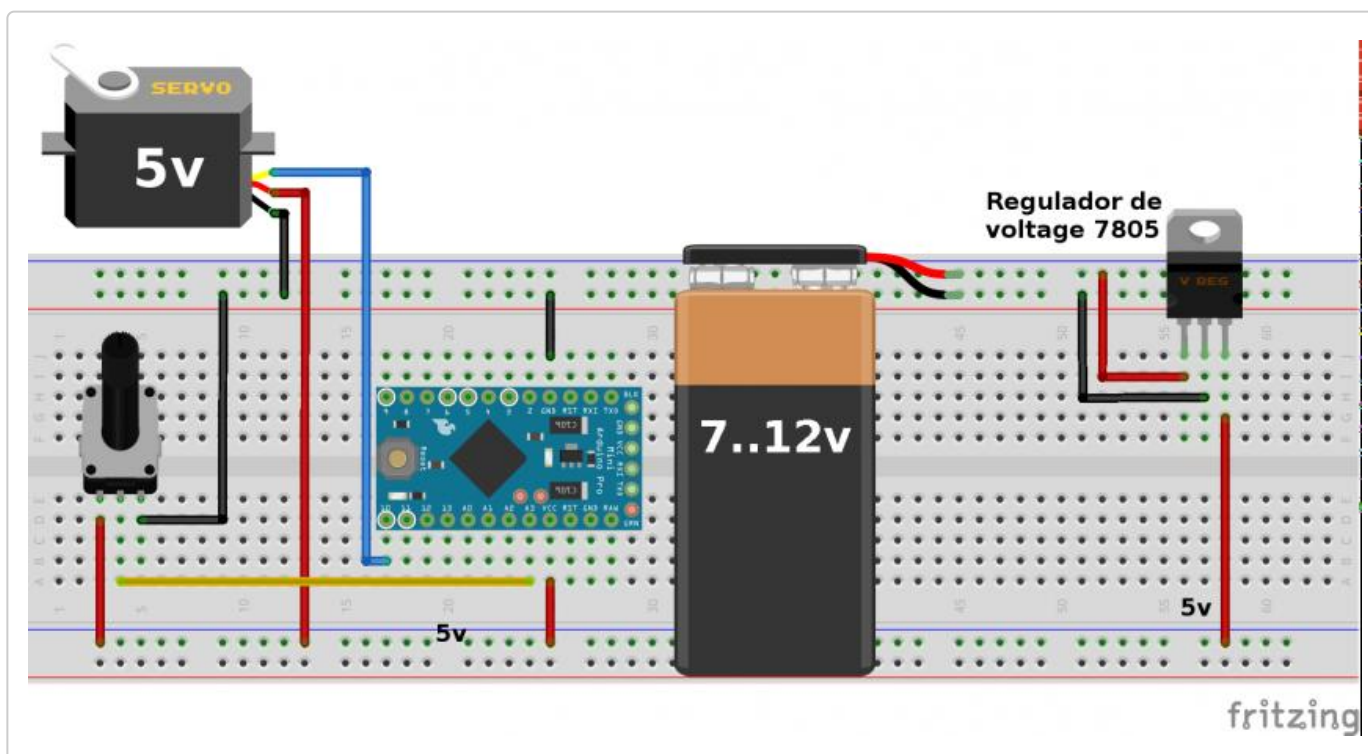


Circuito 3: Alimentando servos más potentes.

La imagen siguiente sirve para ilustrar el uso de servos más potentes que irían conectados directamente a una fuente regulada y estabilizada a 5V. El Arduino deberá conectarse a la alimentación de 5V a través del pin VCC.



Si usamos un servo de 5V y queremos usar una fuente de entre 7V y 12V podemos usar el siguiente circuito que incluye un regulador de voltaje 7805.



Circuito 5: Usando una fuente a 12V y un servo a 12V.

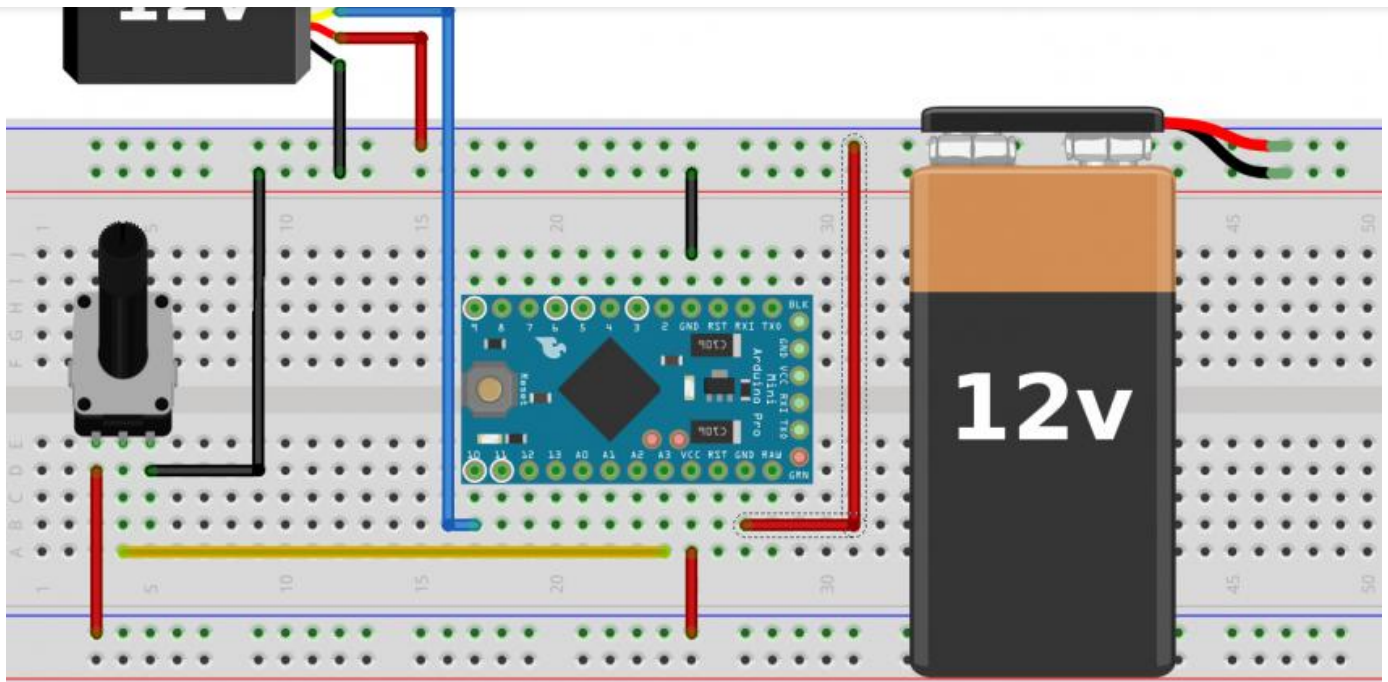
Si usamos una fuente de alimentación a 12V y un servo a 12V, podemos usar el pin Raw para alimentar el Arduino Pro Mini y además en el pin Vcc dispondremos de una toma de 5V.

En este tipo de montaje en el cual usamos el pin Raw para conectar la fuente de alimentación, el pin Vcc quedará limitado a la hora de usarlo para alimentar otras partes del circuito.

En nuestro caso lo podemos usar para conectarlo al potenciómetro porque tendrá una resistencia alta (puede servir potenciómetros de entre 15K y 30V) y no representa un consumo peligroso para el regulador de voltaje interno de Arduino.



Llámanos



El fuente del programa es el siguiente

El programa que usaremos en las pruebas es muy sencillo.

```
//:## #####
//:## Fichero: Demo_Servo_potenciometer.ino Fecha: (21-enero-2019)
//:## Publicado en: https://dronprofesional.com/ (Descarga de Software y documentacion)
//:## -----
//:## * Copyright: (C) 2017 Antonio Castro Snurmacher
//:## * Licencia de uso: (CC) Creative Commons Attribution-NonCommercial-ShareAlike
//:## * https://creativecommons.org/licenses/by-nc-sa/3.0/es
//:## * Este programa no es un producto verificado ni apto para usos prácticos o productivos.
//:## * Carece de todo tipo de garantías y se distribuye únicamente con fines didacticos y
//:## * experimentales. Úselo con prudencia bajo su propia responsabilidad.
//:## * Puede hacer modificaciones en el código, pero esta cabecera debe ser mantenida integramente
//:## * en todas las copias hasta esta última línea.
//:## #####
```

```
/******
```

Este programa sirve para para probar los siguientes elementos de hardware y software.

1 servo

1 Potenciometro

Librería <Servo.h>

Lectura promediada

Funcionamiento:

Permite mover el servo manualmente a cualquier posicon con el potenciometro. Para mayor precisión de la lectura y para filtrar la señal de entrada analógica del potenciometro, se usará una función de lectura promediada que limpiará gran parte del ruido recibido en la entrada analógica que estará

Llámanos



modesta de 1.4 kg.cm. El MG90 es un poco mas caro y más robusto pero muy similar.
Las características entre diferentes tipos de servos pueden variar mucho.

*****/

```
#include <Arduino.h>
#include <Servo.h> // Librería para poder controlar el servo
#define Pin_SERVO_OUT 10 // Pin Actuador Servo
#define Pin_LED_BUILTIN 13 // Pin Led Interno Arduino
#define Pin_POTENCIOMETRO A3 // Pin de entrada analógica para el Potenciómetro
#define Servo_Min 5 // Valor ángulo mínimo para el servo (cerca a 0°)
#define Servo_Max 176 // Valor ángulo máximo para el Servo (cerca a 180°)
#define NUM_LECT_PROMED 50 // Numero de lecturas promediadas
#define LAPSE_LECT_PROMED 1 // Tiempo entre lecturas promediadas en milisegundos

Servo servoMotor; // Instanciamos el objeto servoMotor de tipo Servo. (Dicho de otra forma:
// Creamos un objeto de tipo Servo y lo guardamos en la variable servoMotor).

long ValPot; // Variable para lectura del potenciometro
unsigned long ContLoop=0L; // Contador del bucle principal
// *****
// Recibe un valor ValPot que será mapeado para ajustarlo a los valores maximos y minimos del
// valor Pot recibido para entregarg valores al servo dentro de sus valores máximos y minimos
// Antes de retornar espera el tiempo TimeDelay en milisegundos para completar el movimiento
// *****
void ServoWrite(long ValPot, long ValInputMin, long ValInputMax){
    long Angle;
    // Convierte el rango 0..1023 en un rango Servo_Min, Servo_Max
    Angle = map(ValPot, 0, 1023, Servo_Min, Servo_Max);
    servoMotor.write(Angle);
}

// *****
// Lectura promediada de la entrada analógica conectada al potenciómetro.
// Pese a retornar un valor entre 0 y 1023, haremos cálculos con datos de tipo double para
// así poder promediar un número de lecturas bastante elevado, si así lo quisiéramos, a fin
// de filtrar el ruido de los datos recibidos en la entrada analógica.
// La lectura promediada tardará NUM_LECT_PROMED * LAPSE_LECT_PROMED milisegundos y
// retornará un dato entero de tipo long
// *****
long LecturaPromedPotenciom(){
    double acum=0.0;
    for (int i=0; i<NUM_LECT_PROMED; i++){ //
        acum += analogRead(Pin_POTENCIOMETRO); // Leerá valores entre 0 y 1023 (0v y +5v)
        //delay(LAPSE_LECT_PROMED);
    }
    // Añadiremos 0.5 a la división antes de suprimir decimales para redondear el resultado
    return (long) ( 0.5 + (acum/NUM_LECT_PROMED) );
}

// *****
// Inicializacion
// *****
void setup(){
```



Llámanos



```
pinMode(Pin_LED_BUILTIN , OUTPUT);
pinMode(Pin_POTENCIOMETRO , INPUT);

// Iniciamos el servo para que empiece a trabajar con el pin Pin_SERVO_OUT
servoMotor.attach(Pin_SERVO_OUT); // Servo activado con un pin
servoMotor.write(Servo_Min); // Antes de que se mueva lo llevamos a la posicion inicial
Serial.print(F("attach(")); Serial.print(Pin_SERVO_OUT); Serial.println(')');
}

// *****
// Blucle principal (servo accionado por potenciometro)
// *****

void loop(){
  ContLoop++;
  // Cada 20 iteraciones del bucle verificamos si el potenciometro tiene un valor muy bajo
  // y en tal caso cambiamos el estado del led interno para producir una señalizacion
  // intermitente que identifique el extro de valor bajo del potenciometro.
  if ( (ValPot<20) && (ContLoop % 20)==0){
    digitalWrite(Pin_LED_BUILTIN, !digitalRead(Pin_LED_BUILTIN));
  }
  ValPot = LecturaPromedPotenciom(); // ValPot tomará valores entre 0 y 1023 (0v y +5v)
  Serial.println(ValPot); // Traza del valor leído en el potenciometro
  // Ajustar el valor maximo y el minimo en el servo para los valores de entrada
  // entre 0 y 1023 y accionar servo.
  ServoWrite(ValPot, 0, 1023);
}
```

PD 12-nov-2019:

Si quieres entender [como funciona un servomotor](#), puedes mirar lo que sucede por dentro mientras se mueve.

Se trata de una animación 3D muy lograda.

4 Comentario



Alex on 9 octubre, 2019

Gran artículo chicos. 🙌😊



David Castrillon on 26 abril, 2020

Cordial saludo,

Muy interesante el vídeo y explícito, 📱 puede poner a funcionar un servo de forma inalámbrica?

Llámanos



Me gustaría saber cómo puedo invertir el giro de un servo . Es para usar un dashboard de un auto en un simulador ETS Y ATS . El tema que solo 1 microservo tengo que invertir que es el del combustible ya que al llenar el tanque me marca vacío y viceversa . Ya he probado casi todo . Les agradezco la ayuda



DronProfesional on 14 octubre, 2020

No hay una única respuesta para eso debes mirar en Google para localizar cual es tu caso.

[https://www.google.com/search?](https://www.google.com/search?client=ubuntu&channel=fs&q=como+invertir+el+giro+de+un+servo&ie=utf-8&oe=utf-8)

[client=ubuntu&channel=fs&q=como+invertir+el+giro+de+un+servo&ie=utf-8&oe=utf-8](https://www.google.com/search?client=ubuntu&channel=fs&q=como+invertir+el+giro+de+un+servo&ie=utf-8&oe=utf-8)

Leave a Reply

Name *

Email *

Website

 CAPTCHA



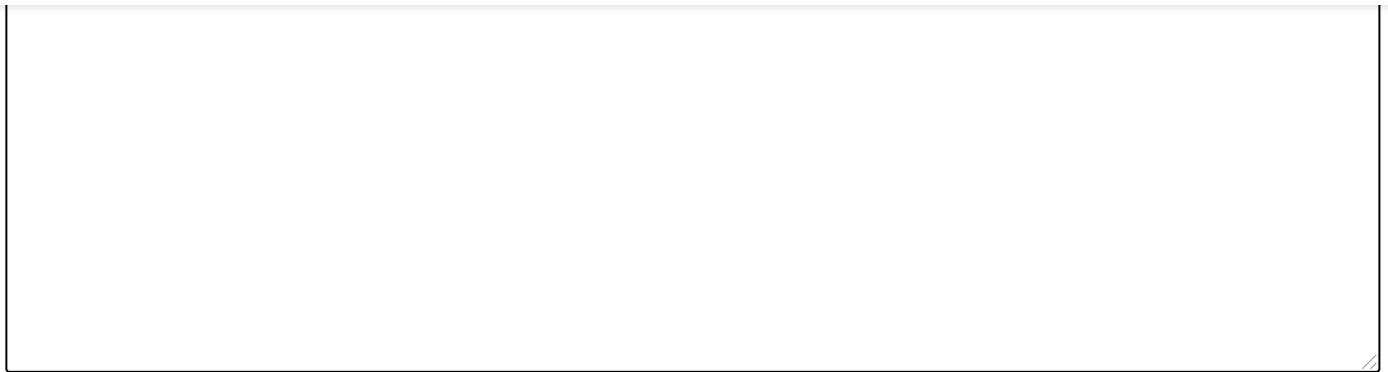
Código CAPTCHA

*

Comment*



Llámanos

[Post Comment](#)

CATEGORÍAS

[Consejos de bodas](#)[DronProfesional](#)[Marcas y modelos](#)[Normativa](#)[Sector audiovisual](#)[Sector drones](#)[Tutoriales, trucos, consejos, bricolajes](#)[Vídeos aéreos](#)

ACCEDE AL FORMULARIO DE CONTACTO

[Accede al formulario de CONTACTO](#)

POSTS RELACIONADOS

[Arduino Pro IDE \(La nueva versión del IDE de Arduino\)](#)

Se trata de un nuevo IDE (Entorno de Desarrollo In...

[Tutorial teórico práctico con servos y Arduino.](#)

5 / 5 (4 votos) En esta ocasión presentamos, a ...

[Llámanos](#)



Cómo comprar un dron de segunda mano y no morir en el intento (guía)

5 / 5 (222 votos)...



¿Vale la pena adquirir DJI Care Refresh cuando compras un dron?

4.8 / 5 (6 votos) Reflexionamos sobre la utilida...



Tutorial: cómo grabar coches con dron en movimiento

5 / 5 (4 votos) Grabar coches con dron es una es...

TU PRODUCTORA AUDIOVISUAL CON DRONES PROFESIONALES EN ESPAÑA



Operamos por toda España peninsular:

Madrid, Guadalajara, Toledo, Segovia, Ávila, Soria, Alicante, Valencia, Albacete, Cuenca, Ciudad Real, León, Zaragoza...

RAZONES PARA CONFIAR EN DRONPROFESIONAL.COM

Consulta nuestra página de Clientes y nuestro Portfolio.

Disfruta de la calidad de nuestros **vídeos corporativos, vídeos de bodas**, etc.

Así comprobarás nuestra capacidad como productores audiovisuales y la calidad de nuestros trabajos de filmación en general y la de filmación aérea en particular.

CONTÁCTANOS

Accede al formulario de
CONTACTO



+34 606 411 729

contacto@dronprofesional.com



Llámanos



POSTS QUE TE PUEDEN INTERESAR



[Cuanto cuesta un video con Dron](#)



[Alquiler de Drones con Piloto](#)



[Tutorial de Realización audiovisual con drones](#)

