**work**

### **Why Arduino?**

Thanks to its simple and accessible user experience, Arduino has been used in thousands of different projects and applications. The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Teachers and students use it to build low cost scientific instruments, to prove chemistry and physics principles, or to get started with programming and robotics. Designers and architects build interactive prototypes, musicians and artists use it for installations and to experiment with new musical instruments. Makers, of course, use it to build many of the projects exhibited at the Maker Faire, for example. Arduino is a key tool to learn new things. Anyone - children, hobbyists, artists, programmers - can start tinkering just following the step by step instructions of a kit, or sharing ideas online with other members of the Arduino community.

There are many other microcontrollers and microcontroller platforms available for physical computing. Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, and many others offer similar functionality. All of these tools take the messy details of microcontroller programming and wrap it up in an easy-to-use package. Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems:

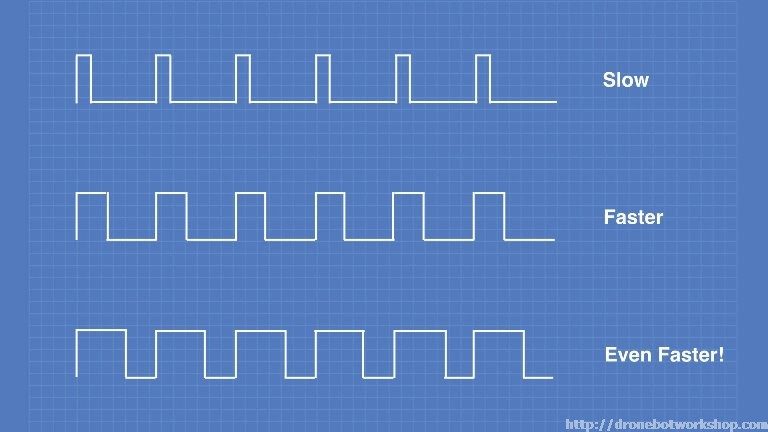
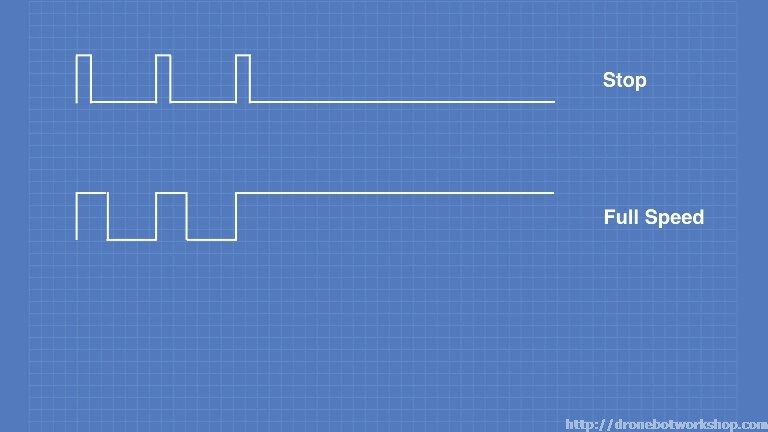
* **Inexpensive** - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than $50
* **Cross-platform** - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
* **Simple, clear programming environment** - The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.
* **Open source and extensible software** - The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to.
* **Open source and extensible hardware** - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the [breadboard version of the module](https://www.arduino.cc/en/Main/Standalone) in order to understand how it works and save money.

# **why ARDUINO MEGA 2560 REV3**

The MEGA 2560 is designed for more complex projects. With 54 digital I/O pins, 16 analog inputs and a larger space for your sketch it is the recommended board for 3D printers and robotics projects. This gives your projects plenty of room and opportunities.

The **Arduino Mega 2560** is a microcontroller board based on the [ATmega2560](http://www.atmel.com/Images/Atmel-2549-8-bit-AVR-Microcontroller-ATmega640-1280-1281-2560-2561_datasheet.pdf). It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega 2560 board is compatible with most shields designed for the Uno and the former boards Duemilanove or Diecimila.

Pulse Width Modulation (PWM)

A far better method of controlling DC motors is to use pulse width modulation or PWM. If you’ve read up on controlling LED’s with your microcontroller you probably have already run into PWM as it’s also a good method of controlling the brightness of an LED.

With PWM the motor is sent a series of pulses. Each pulse is of the full voltage that the motor can handle so a 6-volt motor will be sent 6 volt pulses while a 12-volt motor will be sent 12 volt pulses. The width of the pulses are varied to control the motor speed, pulses with a narrow width will cause the motor to spin quite slowly. Increasing the pulse width will increase the speed of the motor, as illustrated below.

In order to stop the motor completely you just stop pulsing it, essentially sending it zero volts. To run it at full speed you send it the full voltage, again without pulsing it.

You can build a simple PWM generator using a 555 timer and discrete components but it’s a lot easier to use an Arduino. The Arduino has a function called “analogWrite” which is used to drive any of its PWM-capable outputs (the Arduino Uno has 6 digital outputs that are also capable of PWM).

**Miniature Linear Motion actuation** ·

PQ12 Actuonix Motion Devices unique line of Miniature Linear Actuators enables a new generation of motion-enabled product designs, with capabilities that have never before been combined in a device of this size. These tiny linear actuators are a superior alternative to designing your own push/pull mechanisms. Their low cost and easy availability make them attractive to hobbyists and OEM designers alike. The PQ12 actuators are complete, self contained linear motion devices with position feedback for sophisticated position control capabilities, or end of stroke limit switches for simple two position automation.

PQ12 Controller Options

**Option P** – Potentiometer Position Feedback WIRING: (see next page for pin numbering)

1 – Feedback Potentiometer negative reference rail

2 – Actuator Motor Power

3 – Actuator Motor Power

4 – Feedback Potentiometer positive reference rail

5 – Feedback Potentiometer wiper

The –P actuators have no built in controller, but do provide analog position feedback. While voltage is applied to the motor power pins (2 & 3) the actuator extends. Reverse the polarity and the actuator retracts. Position of the actuator stroke can be monitored using the internal linear potentiometer. Provide any stable low and high reference voltage on pins 1 & 4, then read the position signal on pin 5. The voltage on pin 5 will vary linearly between the two reference voltages in proportion to the position of the actuator stroke. Connect to an LAC board for easy interface with any of the following control signals: Analog 0-5V or 4- 20mA, or Digital 0-5V PWM, 1-2ms Standard RC, or USB.



**Option P option**

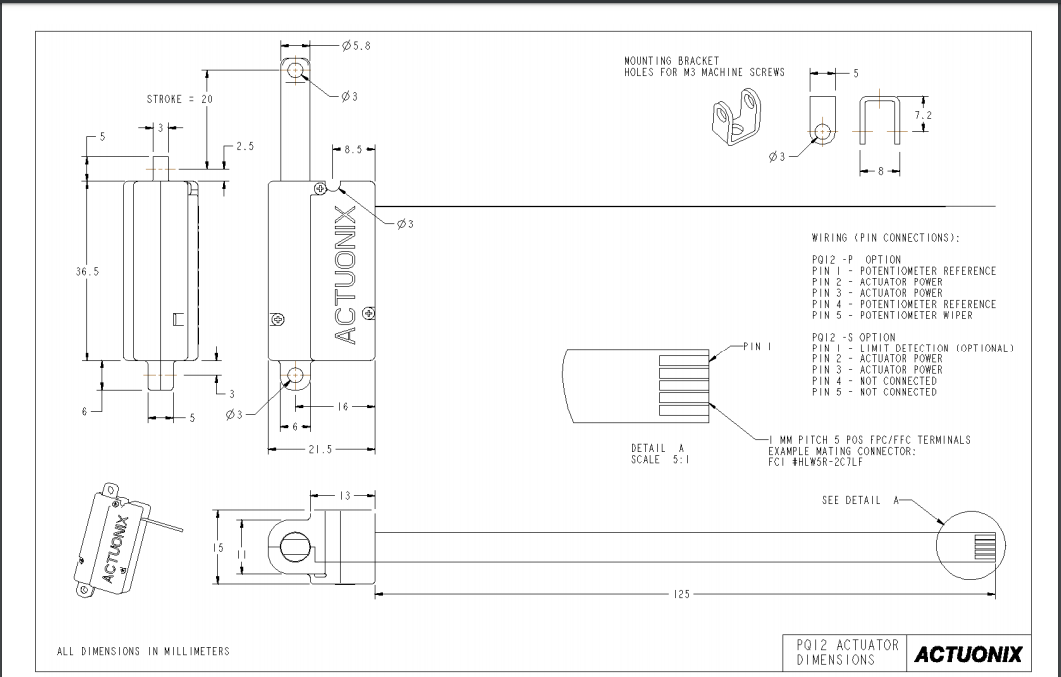
**Option R –** RC Linear Servo WIRING:

1 - RC input signal (RC-servo compatible)

2 - Power (+6 VDC)

3 - Ground

The –R actuators or ‘linear servos’ are a direct replacement for regular radio controlled hobby servos. The desired actuator position is input to the actuator on lead 1 as a positive 5 Volt pulse width signal. A 2.0 ms pulse commands the controller to fully retract the actuator, and a 1.0 ms pulse signals it to fully extend. If the motion of the actuator, or of other servos in your system, seems erratic, place a 1–4Ω resistor in series with the actuator’s red V+ lead wire. The PQ12–R Linear Servos are designed to work with typical RC receivers and battery packs. Consequently, they also are compatible with Arduino control boards, VEX Microcontrollers and many other similar boards designed for robotics.

*Actuator dimensioning and wiring diagram*

Description of MG995 :

Getting an original TowerPro Servo Motor is not an economical option at all!!! And is also a very difficult task to recognize and buy an original TowerPro servo, Because there are many suppliers spread over different online shops who are selling fake servo motors under this Brand name. Robu.in believes in satisfied customers, so whatever the product is we first import them then test them for their supplier defined and standard capabilities. Only after doing all the possible quality checks we make the product available to our customers.

We have imported this copy of original TowerPro MG995 High-Speed Servo from our trusted supplier. Though the motor is the copy of the original High-Speed TowerPro MG995 servo, it is compatible with all the applications that use the genuine TowerPro MG995 Metal Gear Servo Motor because of its same size, comparable quality, and optimized performance. It has metal gears which makes it robust and reliable motors. These TowerPro MG995 Metal Gear Servo Motors are the high-speed servo motors with the mighty torque of 10 kg/cm.

The optimized performance and reliability of this servo have made it the favorite choice of many RC hobbyists. The TowerPro MG995 High-Speed Digital Servo Motor rotates 90° in each direction making it 180° servo motor. It is a Digital Servo Motor which receives and processes PWM signal faster and better. It equips sophisticated internal circuitry that provides good torque, holding power, and faster updates in response to external forces. They are packed within a tight sturdy plastic case which makes them water and dust resistant which is a very useful feature in RC planes, Boats, and RC Monster Trucks etc. It equips 3-wire JR servo plug which is compatible with Futaba connectors too.

Wire Description

RED – Positive

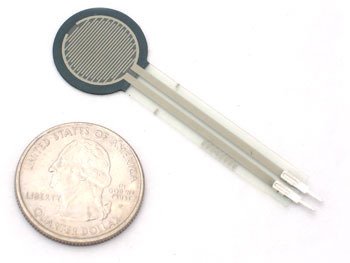
Brown – Negative

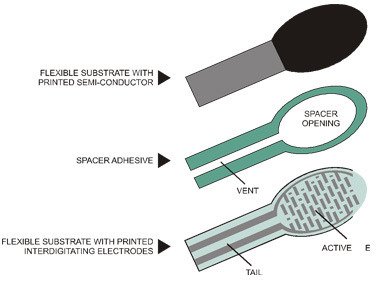
Orange – Signal





# **Overview of fsr**

[](https://learn.adafruit.com/assets/426)FSRs are sensors that allow you to detect physical pressure, squeezing and weight. They are simple to use and low cost.  This is a photo of an FSR, specifically the Interlink 402 model. The 1/2" diameter round part is the sensitive bit.

[](https://learn.adafruit.com/assets/428)The FSR is made of 2 layers separated by a spacer. The more one presses, the more of those Active Element dots touch the semiconductor and that makes the resistance go down.

FSRs are basically a resistor that changes its resistive value (in ohms Ω) depending on how much it is pressed. These sensors are fairly low cost, and easy to use but they're rarely accurate. They also vary some from sensor to sensor perhaps 10%. So basically when you use FSRs you should only expect to get ranges of response. While FSRs can detect weight, they're a bad choice for detecting exactly how many pounds of weight are on them.