

[HOME](#)[SHOP](#)[DEVELOPER](#)

MUSCLE CONTROL V1.2

August 22, 2016

OVERVIEW

This tutorial will guide you through the process of using Electromyography (EMG) muscle sensors to control the Brunel hand.

This tutorial is for:

- Hand: [Brunel](#)
- Board: [Chestnut](#)
- Firmware: [Beetroot](#)

You will need:

- Brunel hand, running Beetroot firmware
- [12V DC power supply](#)
- Micro USB cable
- Computer running [Arduino IDE](#), with the Chestnut PCB installed ([Step 4 of the README](#))
- Soldering iron + solder
- Small flat-head screwdriver (for adjusting the sensor gain)
- 1 or 2 x [MyoWare Muscle Sensors](#) (this tutorial covers both single and dual muscle control)
- 3 x [Sticky electrode pads](#)

- 1 x 4 Pole 3.5mm headphone jack (**IMPORTANT**, must be 4 Pole)
- 1m Red wire
- 1m Black wire
- 1m Green wire

If you are using 2 MyoWare sensors for dual muscle control, you will also need:

- 20cm Red wire
- 20cm Black wire
- 1.2m Blue wire
- 1 x MyoWare Muscle Sensors (this tutorial covers both single and dual muscle control)
- 3 x Sticky electrode pads

WARNING, THE MYOWARE SENSORS ARE NOT ISOLATED. BEFORE ATTACHING THE ELECTRODES, BE SURE TO READ THROUGH THE MYOWARE DATASHEET

PREPARING THE SENSORS

Electromyography (EMG) sensors are used to detect the electrical potential of your muscles. A signal is picked up from a surface electrode placed on your skin, the signal is then passed through a series of amplifiers and filters to produce a 'clean' signal. The resulting 'clean' signal is within the usable range of a microcontroller, so can be fed into an Analogue to Digital Converter (ADC), where the magnitude of the signal represents the magnitude of the muscle activation, i.e. the greater the signal magnitude, the harder the muscle is tensing. This signal can then be used to open or close the hand.

CONSTRUCTING THE SENSOR

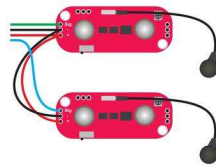
EMG sensors are becoming more widely available; this tutorial involves using either 1 or 2 [MyoWare Sensor Boards](#) from [Advancer Technologies](#).

The first step is to attach the power, ground and signal wires from the MyoWare board to the 4-pole headphone jack.



1. Solder the 1m black wire (ground) to the pad next to the negative symbol (-) on the MyoWare board
2. Solder the 1m red wire (positive) to the pad next to the positive symbol (+) on the MyoWare board
3. Solder the 1m green wire (signal) to the pad next to the signal symbol (SIG) on the MyoWare board

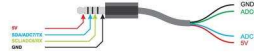
If you are using 2 MyoWare sensors for dual muscle control, you will also need to complete steps 4 - 6. If you only have one MyoWare board, you can skip straight to step 7.



4. Solder the 20cm black wire (ground) to connect the negative pads (-) between the first and second MyoWare boards
5. Solder the 20cm red wire (positive) to connect the positive pads (+) between the first and second MyoWare boards

6. Solder the 1.2m blue wire (signal) to the pad next to the signal symbol (SIG) on the second MyoWare board

The next step is to connect the other end of the wires to the 4-pole headphone jack so that the MyoWare boards can connect to the Chestnut board.



7. Solder the other end of the 1m black wire (ground) to the GND connection of the headphone jack

8. Solder the other end of the 1m red wire (positive) to the 5V connection of the headphone jack

9. Solder the other end of the 1m green wire (signal) to the ADC6 connection of the headphone jack

10. If you are using 2 MyoWare boards, solder the other end of the 1.2m blue wire (signal) to the ADC7 connection of the headphone jack

The assembly of the sensors is now complete.

POSITIONING THE ELECTRODES

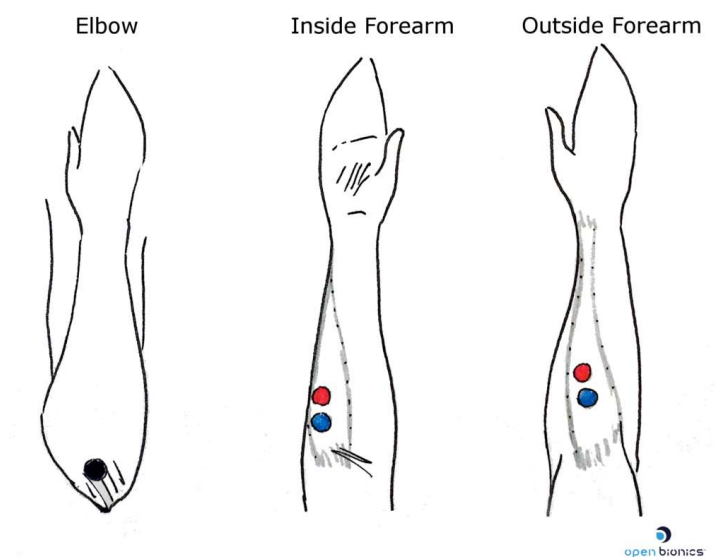
WARNING, THE MYOWARE SENSORS ARE NOT ISOLATED. BEFORE ATTACHING THE ELECTRODES, BE SURE TO READ THROUGH THE MYOWARE DATASHEET

Each sensor board requires three electrodes to attach to the skin; 2 electrodes are used for the muscle

signal (on board) and a 3rd electrode used for a body/ground reference (black cable).

1. Press each of the sticky electrode pads into the electrode sensors
2. Peel off the paper backing of each sticky electrode and apply the MyoWare board to the outside of the forearm, shown in the image below. The MyoWare board should be placed on the belly of the forearm muscle ([see the datasheet for more details on sensor location](#))
3. Apply the ground electrode (black) to the elbow

If you are using 2 MyoWare sensors for dual muscle control, you will also need to repeat steps 1 - 3 and place the second MyoWare board on the inside of the forearm.



EMG sensor placement

TESTING THE SENSORS

WARNING, THE MYOWARE SENSORS ARE NOT ISOLATED. BEFORE ATTACHING THE

ELECTRODES, BE SURE TO READ THROUGH THE MYOWARE DATASHEET

UPLOADING THE TEST FIRMWARE

To verify the MyoWare sensors are wired and connected to the skin correctly, you will need to download and run the following Arduino sketch.

1. Download [MyoWare_Sensor_Test.ino](#)
2. Open the sketch using [Arduino](#)
3. If you are using a single MyoWare sensor board, you will need to change the number of EMG channels on line 17 to '1'

```
#define NUM_EMG_CHANNELS 2 // 1 or 2
```

4. Power the Chestnut board and connect it to the computer
5. Compile and upload the sketch to the Chestnut board (**make sure you have installed the Chestnut board, see [step 4 of the README](#)**)
6. Connect the MyoWare board by plugging the 4-pole headphone connector into the Chestnut board
7. Open the Serial Monitor (Tools -> Serial Monitor) and set the baud rate to 38400 bps

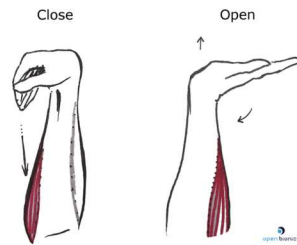
You should now see a the ADC values relating to the EMG signal from each MyoWare board being printed over the serial monitor, for example:

```
Ch0: 16 Ch1: 23
```

(If you are only using a single MyoWare board, you will only see 'Ch0: 16')

USING THE SENSORS

The image on the right shows the two hand motions you should perform to activate the inside and outside forearm muscles.



Hand motions for muscle activation (this is a right hand)

To activate the inside forearm muscle, flex the wrist towards the elbow whilst squeezing the tips of your fingers into the heel of your hand. When you perform this motion, you should see/feel the muscle underneath one of the MyoWare boards tense.

To activate the outside forearm muscle, extend the wrist with an outstretched palm. You should see/feel the muscle underneath the other MyoWare board tense.

TUNING THE SENSOR GAIN



When you perform each of these motions, you should see the ADC value for each channel change.

When your muscle is relaxed, the ADC value should rest around 0 - 300, and when the muscle is tensed it should read around 700 - 900. If the ADC value is too low, you will need to increase the gain of the sensor.

Use the flat-head screwdriver to turn the potentiometer clockwise to increase the gain.

For more details see the [MyoWare sensor board datasheet here](#).

BEETROOT MUSCLE CONTROL

The [Beetroot Firmware](#) allows the [Brunel](#) hand to be controlled via muscle signals. The above EMG sensors detect the activation levels of each muscle, and then feed the activation level (analogue signal) into the ADC of the Chestnut board, via the headphone jack.

There are two main control methods for controlling the hand using the EMG sensors, 'Standard Muscle Control' mode and 'Positional Muscle Control' mode. These modes can be enabled by entering the serial command 'M1' or 'M2' respectively, and will stay enabled until they are disabled by entering the serial command 'M0'.

If you are using a single MyoWare sensor board, you will need to change the number of EMG channels by navigating to '**EMGControl.h**' and changing the following on **line 22**.

```
#define NUM_EMG_CHANNELS 2 // select the  
    number of EMG channels to use (1 or 2)
```

STANDARD MUSCLE CONTROL MODE (M1)

Once 'Standard Muscle Control Mode' (M1) is enabled, the hand processes the ADC value of each EMG channel. If the muscle is not active, a rolling average is taken of the resting ADC value, which is used as the

noise floor. This noise floor is a rolling threshold that is used to remove the EMG signal of a relaxed muscle, which simply makes the EMG signal 0 when the muscle is relaxed, and >0 when the muscle is active.

When the muscle is tensed and relaxed it produces a peak in the EMG signal, which is detected by the hand.

If you are using a two MyoWare sensors, the hand will open fully when a peak is detected on the open muscle, and will close fully when a peak is detected on close muscle. If the open muscle is held tensed for more than 400ms, the hand will cycle to the next grip.

The possible grip patterns are listed in order below:

1. Fist - all fingers and thumb move
2. Palm - all fingers move, thumb stays open
3. Thumbs up - all fingers stay closed, thumb moves
4. Point - all fingers remain closed, only the index finger moves
5. Pinch - all fingers remain open, only the thumb and index move
6. Tripod - ring and pinky remain open, index middle and thumb move

If you are using a single MyoWare sensor, the hand will toggle between being fully open and fully closed when the falling edge of an EMG peak is detected. The grip can also be changed by holding the muscle tensed.

The value for the 'peak threshold', above which the EMG signal is deemed to be at a peak, can be manually changed by entering the serial command 'U###', where # is a value between 0 - 1024. If you simply enter the command 'U', you will see the current value for the 'peak threshold'.

The value for the 'hold duration', which is the amount of time the muscle must remain tensed for the grip to change, can be manually changed by entering the serial command 'T###', where # is a time in ms. If you simply enter the 'T' command, you will see the current value for the 'hold duration'.

PROPORTIONAL MUSCLE CONTROL MODE (M2)

Positional muscle control is designed to achieve finer control of the hand when using muscle sensors, this mode also more closely reflects how a robotic prosthetic hand is controlled. The hand only moves when a muscle is tensed, where the speed of movement is proportional to how much the muscle is tensed. This means that the hand can be moved slowly by tensing gently and can be moved quickly when tensing more firmly.

If you are using a two MyoWare sensors, the hand will only move when either the open or close muscles are tensed.

If you are using a single MyoWare sensor, the hand will open when the muscle is relaxed, and will close when the muscle is tensed.

To enable 'Proportional Muscle Control Mode' enter 'M2' over serial, this mode can be disabled by entering 'M0'. For this mode you may need to decrease the 'peak threshold' (U###) to allow for the 'gentle' tenses to be picked up by the hand.

You can change grip in this mode in exactly the same way as in 'Standard Muscle Control' mode.

MUSCLE CONTROL DATA DUMP (M3)

'Muscle Control Data Dump' mode is used to view the EMG and muscle control data in real-time. Upon entering 'M3', you should see the following:

**M0 = 178 T0 = 369 N0 = 169 A0 = 0 M1 = 200
T1 = 384 N1 = 184 A1 = 0 DIR None**

- M# - raw muscle signal
- T# - calculated threshold (noise floor + sensitivity value)
- N# - noise floor
- A# - whether the muscle is determined to be active (0 = inactive, 1 = active)
- DIR - calculated direction of the combined muscle signal (open, close, none)

If you are only using a single MyoWare sensor, you will only see the data relating to the single channel.

With your muscles relaxed, both A0 and A1 should be 0, showing that the muscles are determined to be inactive. If any of them is detected as active, even when the muscle is relaxed, try increasing the sensitivity value by entering 'U###', where # is the new value, typically between 100 - 300, where the lower the number relates to an increase in sensitivity.

EXTRA

I2C MUSCLE SENSORS

An alternative solution for muscle sensing is to use an I2C ADC. Instead of passing the raw analogue signals down through a long cable to the ADC on the Chestnut board, which could result in an increase in noise, an I2C ADC could be located as close as possible to the muscles/EMG sensor. This would result in a shorter cable length between the output of the

EMG sensors and the ADC, thus reducing the chance of noise.

IMPORTANT NOTE ABOUT I2C

The headphone jack is set to use ADC signals by default. If you wish to use I2C across the headphone jack, you will need to write digital pin 10 (PA21) to be LOW.

```
pinMode(10, OUTPUT);  
digitalWrite(10, HIGH);
```

Within Beetroot, this can be configured by writing the following:

```
setHeadphoneJack(JACK_I2C);
```

In Tutorial Tags Active Tutorial

Share

9 Likes

← BEETROOT V1.0	ARTICHOKE V1.2
FIRMWARE USER GUIDE	FIRMWARE USER GUIDE
FOR BRUNEL HAND	→

[Blog](#) [Contact Us](#)

© Open Bionics 2018