

University of Nebraska at Omaha

From the Selected Works of Philippe Malcolm

Summer July 17, 2018

aGonabadiEtAl_2018_EduExoEMGdelayTutorial_13.pdf

Arash Mohammadzadeh Gonabadi

Philippe Malcolm, *University of Nebraska at Omaha*

Available at: <https://works.bepress.com/philippe-malcolm/2/>

Tutorial: Using EduExo to visualize electromechanical delay in *m. biceps brachii*

Tutorial version: 1.0

Authors: Arash, M. Gonabadi, Philippe Malcolm

Department of Biomechanics and Research in Human Movement Variability,
University of Nebraska at Omaha

Contact: amgonabadi@unomaha.edu, pmalcolm@unomaha.edu

1. Required materials and software

- Assembled EduExo educational elbow exoskeleton **with muscle control extension-kit** <https://www.beyond-robotics.com>
- A laptop with MS Excel
- Arduino Integrated Development Environment <https://www.arduino.cc/en/Main/Software>

2. Introduction on electromechanical delay in skeletal muscles

Skeletal muscle contractions are initiated by electrical signals from the nervous system. We can measure this electrical activity via electromyography (EMG) sensors that placed on the skin (Basmajian, 1962). These measurements are useful to evaluate the activation level and timing of muscle contractions, for example in clinical gait analysis.

EMG studies have shown that there is a delay of 30 to 100ms between the onset of electromyographical activity and the onset of measurable tension in the muscle (Cavanagh and Komi, 1979). This delay appears to be due a combination of electrochemical and mechanical processes (Lacourpaille et al., 2013).

Knowledge of the existence of this delay can be important for multiple applications. For example, in exoskeletons that are controlled via EMG activity the control mechanism is sometimes designed such that the exoskeleton-delay is similar to the electromechanical delay in human muscle in order to make the exoskeleton act more intuitive, similar to the wearer's own muscles (Ferris et al., 2005).

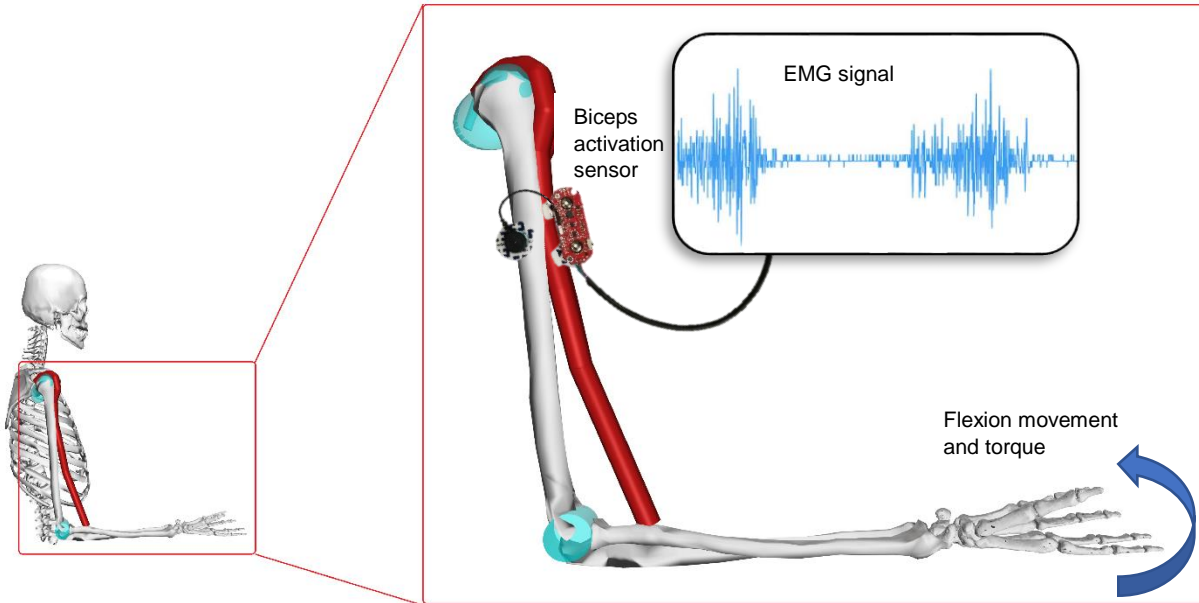


Figure 1: Schematic explanation of EMG measurement. Bipolar surface electrodes are placed on the skin and measure electrical activity associated with muscle contractions.

3. Tutorial

EduExo is an educational elbow exoskeleton with different sensors and a motor. The purpose of this tutorial is to use EduExo as to demonstrate electromechanical delay by measuring EMG and joint angle data.

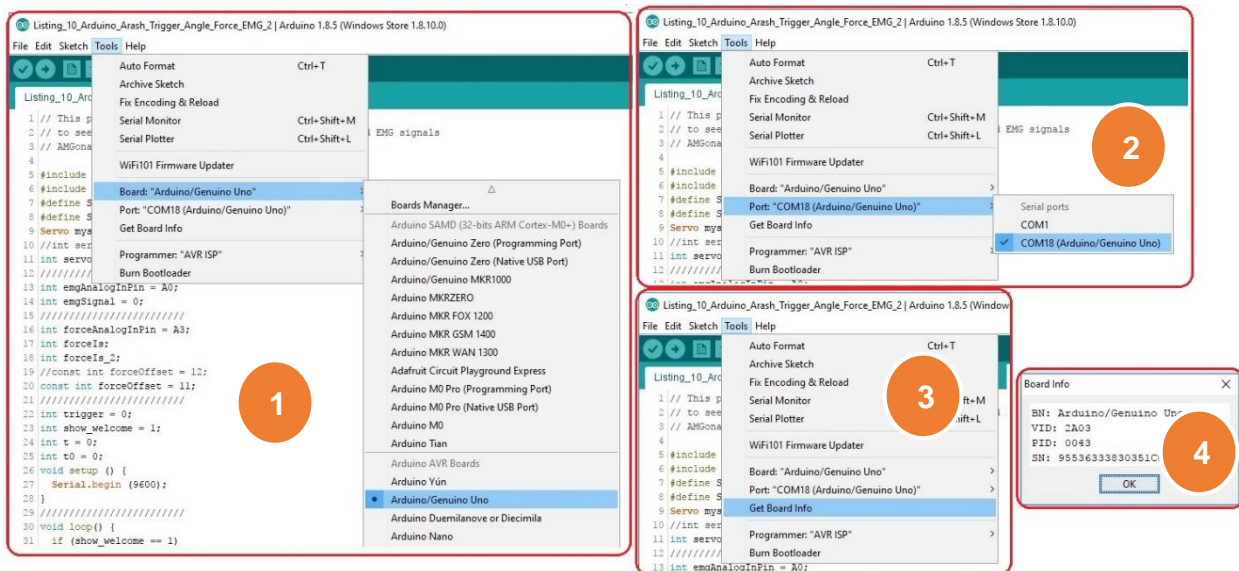


Figure 2: Steps for checking the connection of the Arduino board.

4. Steps

- 1) Connect the EduExo to the Arduino board and connect the Arduino board to your laptop as shown in the EduExo manual.
- 2) Start the Arduino software and follow the steps (1-4) for testing the connection between Arduino board to your computer (figure 2).
- 3) Click "File > New" and copy the code from the "Arduino sketch code" section in this tutorial into the new sketch window. Then click "Sketch > Upload" (figure 3).

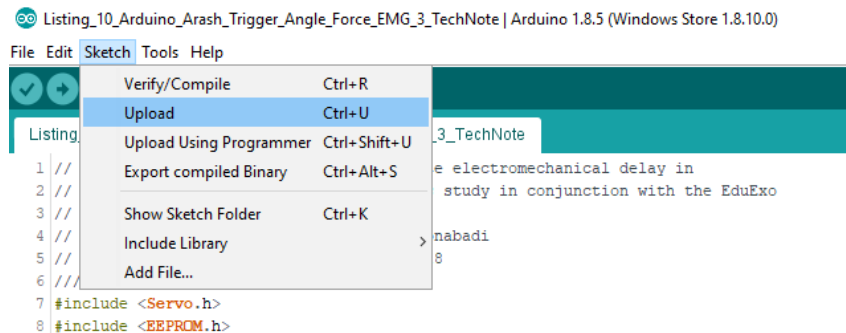


Figure 3: Copy and Upload the code in the board

- 4) Let a participant wear EduExo with electrodes placed on the "medial biceps femoris". The participant has to be able to see the laptop screen. Make sure that the laptop is not connected to the power outlet for safety reasons (figure 4).

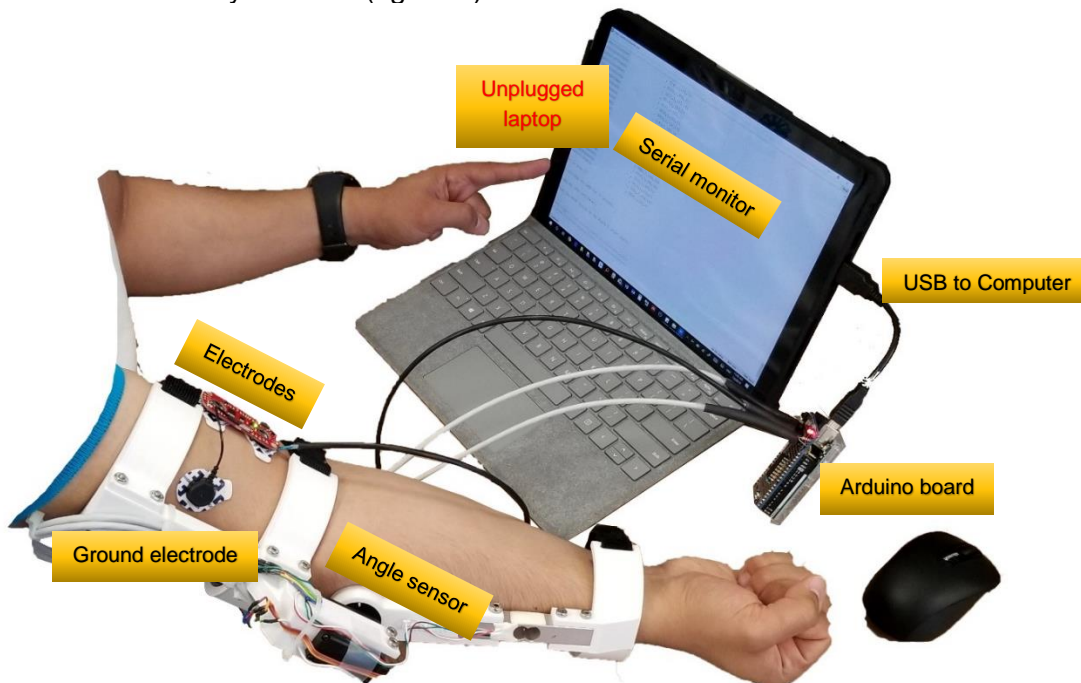


Figure 4: Setup for tutorial for measuring electromechanical delay with EduExo.

- 5) Click on "Tools > Serial Monitor"
- 6) Instruct the participant to wait while holding the arm extended and flex the elbow as fast as possible as soon as the word "GO" appears on the **bottom** of the screen.
- 7) When this script displays "GO" the participant flexes his/her arm as fast as possible.
- 8) Copy the data from the Serial Monitor window into MS Excel (figure 5).

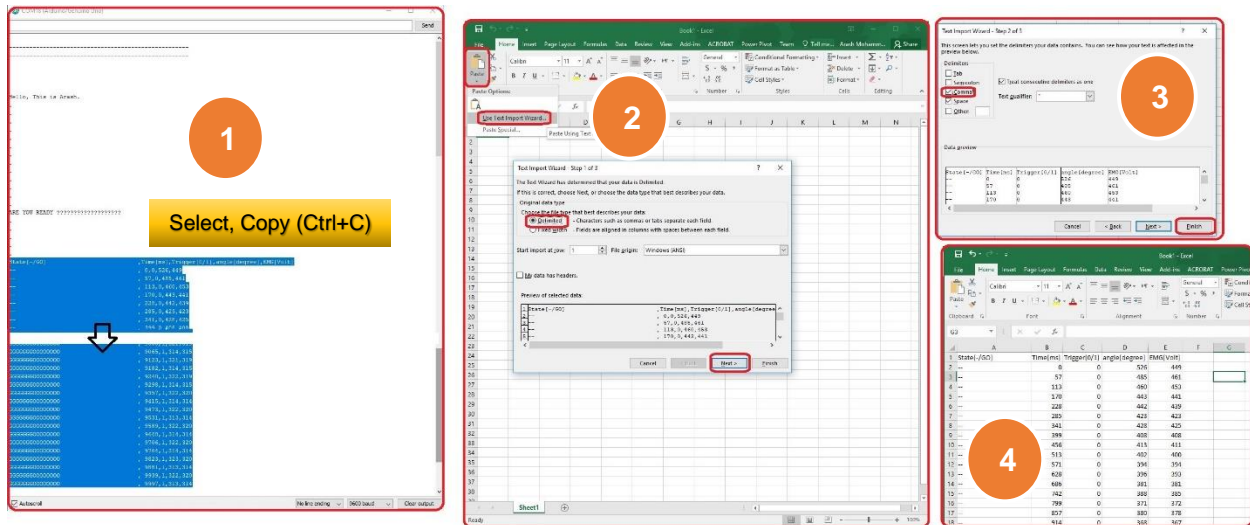


Figure 5: Steps for copying to excel for plotting EMG and angle results.

- 9) Plot the column with EMG data and the column with joint angle data versus time. Normally this will show a small delay between the onset of EMG activity and the first change in joint angle (figure 6). It helps to normalize the angle and EMG data such that these can be plotted on an axis from 0 to 1. It should be taken into account that the sampling rate of the Arduino board is relatively low so the delay cannot be measured very accurately.

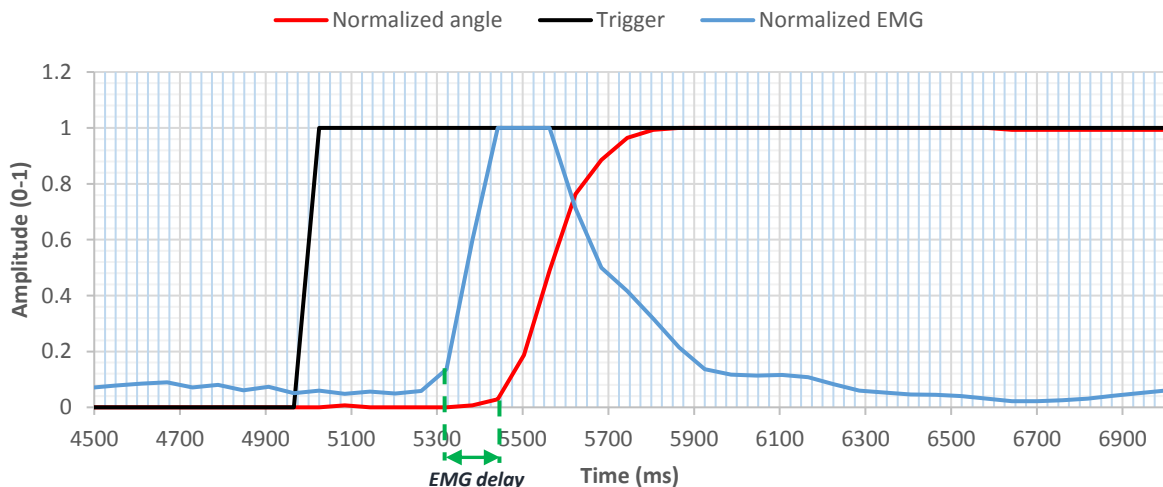


Figure 6: Example plot of EMG signal, joint angle and trigger showing electromechanical delay.

5. Arduino sketch code

```

1 // This program is intended to demonstrate electromechanical delay in
2 // human skeletal muscle for biomechanics study in conjunction with the EduExo
3 // elbow exoskeleton.
4 // This program is written by Arash M. Gonabadi
5 // AMGonabadi@unomaha.edu , 06/09/2018
6 ///////////////////////////////////////////////////////////////////
7 #include <Servo.h>
8 #include <EEPROM.h>
9 #define SAMPLE_DELAY 25
10 #define SAMPLES 200
11 Servo myservo;
12 //int servoPin = 3;
13 int servoAnalogInPin = A1;
14 ///////////////////////////////////////////////////////////////////
15 int emgAnalogInPin = A0;
16 int emgSignal = 0;
17 ///////////////////////////////////////////////////////////////////
18 int forceAnalogInPin = A3;
19 int forceIs;
20 int forceIs_2;
21 //const int forceOffset = 12;
22 const int forceOffset = 11;
23 ///////////////////////////////////////////////////////////////////
24 int trigger = 0;
25 int show_welcome = 1;
26 int t = 0;
27 int t0 = 0;
28 void setup () {
29   Serial.begin (9600);
30 }
31 ///////////////////////////////////////////////////////////////////
32 void loop() {
33   if (show_welcome == 1)
34   {
35     Serial.println(" ");
36     Serial.println("-----");
37     Serial.println("-----");
38     Serial.println(" ");
39     Serial.println(" ");
40     Serial.println(" ");
41     Serial.println(" ");
42     Serial.println("This program is designed to demonstrate electromechanical");
43     Serial.println("delay in skeletal muscle. Hold your arm relaxed. Then flex ");
44     Serial.println("your elbow as fast as you can when the 'GO' signal appears.");
45     delay(500);
46     Serial.println(". ");
47     delay(500);
48     Serial.println(". ");
49     delay(500);
50     Serial.println(". ");
51     delay(500);
52     Serial.println(". ");
53     delay(500);
54     Serial.println(". ");

```

```
55 delay(500);
56 Serial.println(". ");
57 delay(500);
58 Serial.println(". ");
59 delay(500);
60 Serial.println(". ");
61 delay(500);
62 Serial.println(". ");
63 delay(500);
64 Serial.println(". ");
65 delay(500);
66 Serial.println(". ");
67 delay(500);
68 Serial.println(". ");
69 delay(500);
70 Serial.println(". ");
71 delay(500);
72 Serial.println("ARE YOU READY?");
73 Serial.println(". ");
74 delay(500);
75 Serial.println(". ");
76 delay(500);
77 Serial.println(". ");
78 delay(500);
79 Serial.println(". ");
80 delay(500);
81 Serial.println(". ");
82 Serial.println("State[-/GO
,Time[ms],Trigger[0/1],angle[degree],EMG[Volt],Force[N]");
83 show_welcome = 0;
84 t0 = millis();
85 }
86 String dataString = "";
87 if (trigger == 1)
88 {
89 dataString += "GGGGGGG00000000 , ";
90 }
91 else
92 {
93 dataString += "-- , ";
94 }
95 dataString += millis() - t0;
96 dataString += ",";
97 dataString += String(trigger);
98 dataString += ",";
99 //////////////////////////////////////
100 int posIs = analogRead(servoAnalogInPin);
101 dataString += String(posIs);
102 dataString += ",";
103 //////////////////////////////////////
104 emgSignal = analogRead(emgAnalogInPin) ;
105 dataString += String(emgSignal);
106 dataString += ",";
107 //////////////////////////////////////
108 forceIs = analogRead(forceAnalogInPin);
109 forceIs -= forceOffset;
110 dataString += String(forceIs);
```

```
111 //////////////////////////////////////////////////
112 Serial.println(dataString);
113 delay(1);
114 //////////////////////////////////////////////////
115 t = abs(millis() - t0);
116 if (t > 5000)
117 {
118   trigger = 1;
119 }
120 //////////////////////////////////////////////////
121 if (t > 10000 && trigger == 1)
122 {
123   Serial.println(" ");
124   Serial.println("Recording done!");
125   Serial.println(" ");
126   Serial.println("10 seconds of data are recorded.");
127   Serial.println(" ");
128   Serial.println("Please copy your DATA to Excel for visualization.");
129   Serial.println(" ");
130   Serial.println("Push the RESET button on the Board to start again.");
131   Serial.println(" ");
132   Serial.println(" ");
133   Serial.println("-----");
134   Serial.println("-----");
135   delay(600000000);
136 }
137 }
138
```



```
-----  
-----  
  
This program is designed to demonstrate electromechanical  
delay in skeletal muscle. Hold your arm relaxed. Then flex  
your elbow as fast as you can when the 'GO' signal appears.  
. . . . .  
ARE YOU READY?  
. . . . .  
State[-/GO],Time[ms],Trigger[0/1],angle[degree],EMG[Volt],Force[N]  
-- , 0,0,544,466,862  
-- , 60,0,712,587,862  
-- , 120,0,792,681,862  
-- , 181,0,826,743,863  
  
GGGGGGG000000000 , 9839,1,857,833,862  
GGGGGGG000000000 , 9901,1,848,821,862  
GGGGGGG000000000 , 9963,1,848,819,862  
  
Recording done!  
  
10 seconds of data are recorded.  
  
Please copy your DATA to Excel for visualization.  
  
Push the RESET button on the Board to start again.  
  
-----  
-----
```

Figure 6: Example of output that you should see on serial monitor window. The part that should be copied into excel is highlighted in blue.

6. References

- Basmajian, J. V., 1962. Muscles Alive: Their Functions Revealed by Electromyography, Academic Medicine. <https://doi.org/10.1001/jama.1967.03130040073037>
- Cavanagh, P., Komi, P., 1979. Electromechanical delay in human skeletal muscle under concentric and eccentric contractions. *Eur. J. Appl. Physiol.* 163, 159–163.
- Ferris, D.P., Czerniecki, J.M., Hannaford, B., 2005. An ankle-foot orthosis powered by artificial pneumatic muscles. *J. Appl. Biomech.* 21, 189–197. <https://doi.org/10.1123/jab.21.2.189>
- Lacourpaille, L., Hug, F., Nordez, A., 2013. Influence of Passive Muscle Tension on Electromechanical Delay in Humans. *PLoS One* 8. <https://doi.org/10.1371/journal.pone.0053159>

7. Acknowledgements

This work was supported by the Nebraska EPSCoR FIRST Award (OIA-1557417). We would like to thank Angel Gonzalez for suggestions on the contents of this tutorial.