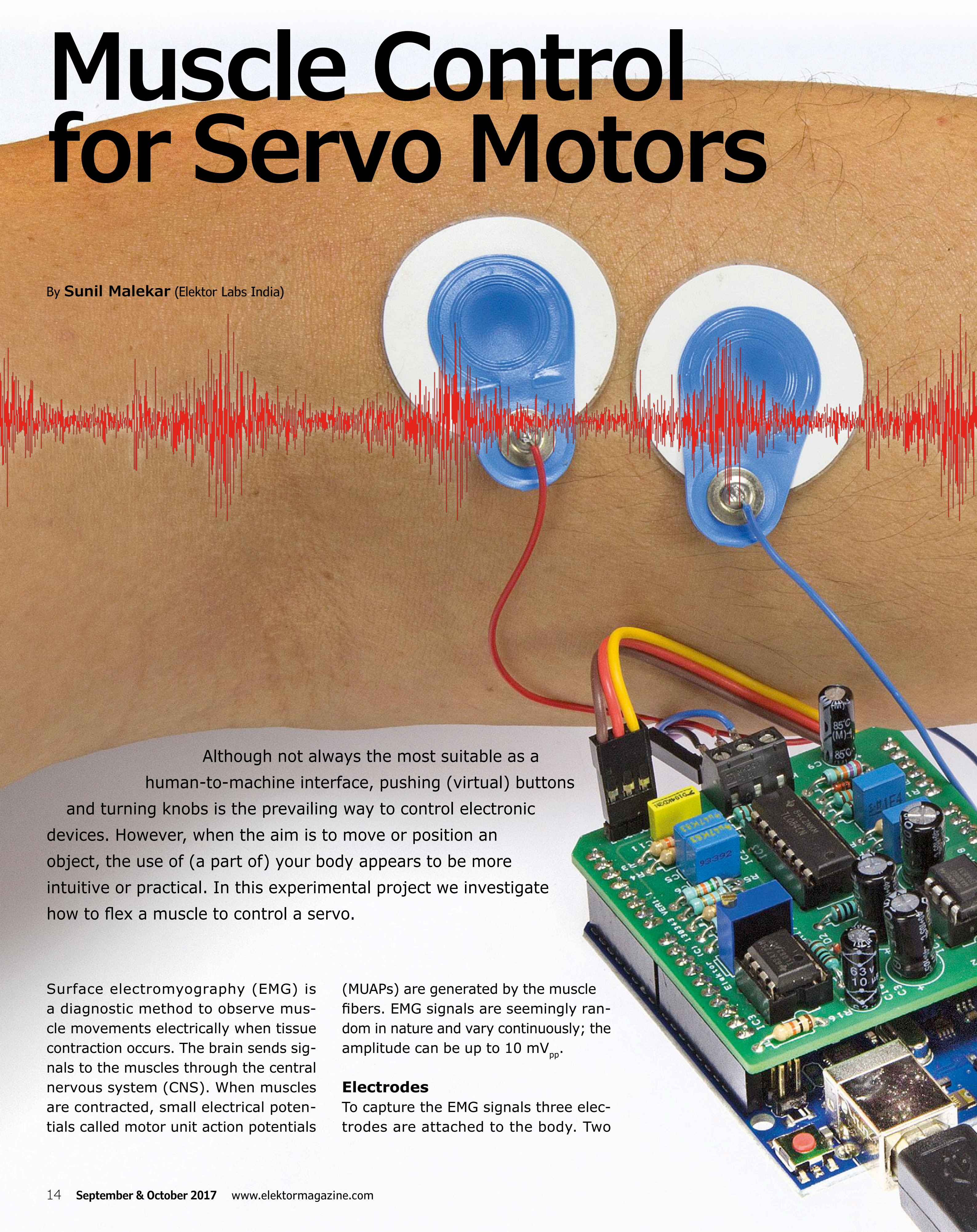
Muscle Control for Servo Motors



By **Sunil Malekar** (Elektor Labs India)

Although not always the most suitable as a human-to-machine interface, pushing (virtual) buttons and turning knobs is the prevailing way to control electronic devices. However, when the aim is to move or position an object, the use of (a part of) your body appears to be more intuitive or practical. In this experimental project we investigate how to flex a muscle to control a servo.

Surface electromyography (EMG) is a diagnostic method to observe muscle movements electrically when tissue contraction occurs. The brain sends signals to the muscles through the central nervous system (CNS). When muscles are contracted, small electrical potentials called motor unit action potentials (MUAPs) are generated by the muscle fibers. EMG signals are seemingly random in nature and vary continuously; the amplitude can be up to 10 mV_{pp}.

Electrodes

To capture the EMG signals three electrodes are attached to the body. Two

Arduino reads EMG signals

electrodes are affixed roughly 2-3 cm apart in the region where the muscle movement is to be detected. The third electrode supplies a reference level for the EMG signal to be measured. Preferably this electrode is placed at an electrically neutral tissue like a bony portion not too far from the other two electrodes. At the same time it should not be placed too close either to avoid it from affecting the signal strength.

Suitable electrodes can be found on the Internet — look for foam monitoring electrodes. They usually come in packs of 30 or 50 for around $10. There are gelled and dry type electrodes. We have used gelled electrodes on which an electrolyte solution or gel (also available on the Internet) is put on the side of the electrode in contact with the skin; the other side of the electrode is connected to the circuit's input. A chemical reaction occurs at the interface between the gel and the electrode, causing a potential to be developed at the electrodes. Silver chloride (AgCl) electrodes are best because impedance fluctuations between skin and electrode are minimized, which results in less noise.

Circuit description

The EMG signal captured by the electrodes is weak and needs to be amplified. Usually an instrumentation amplifier with a high-impedance input is employed for this function. However, the drawback of such an input is that noise (AC line hum; RF interference) is easily picked up too, especially when the leads to the electrodes are long. An instrumentation amplifier also requires quite a few components. For these reasons we have opted for a compromise between

Features

- Controls servo with muscle
- Arduino shield
- Through-hole only

PROJECT INFORMATION

Arduino
EMG
biocontrol
servo

➡ entry level
intermediate level
expert level

2 hours approx.

Arduino,
soldering

£27 / €30 / $33 approx.

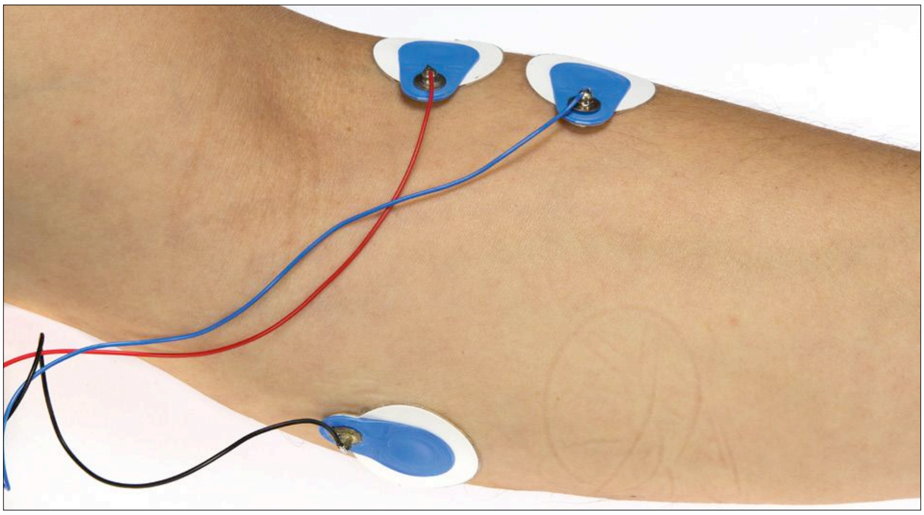


Figure 1. Suitable electrodes for this project and conducting gel can be found easily on the Internet.

simplicity and sensitivity and used a differential amplifier (IC2a) instead of an instrumentation amplifier.
The output of the differential amplifier is fed into a second-order bandpass filter (IC2b and IC2c) to remove unwanted signals. It has a total gain of about 2.4 and passes signals between 20 Hz and 500 Hz, anything outside this range is attenuated at a rate of 12 dB per octave. The filtered signal is amplified further by IC2d, and then rectified by diode D1. Capacitor C9 helps to extract the EMG signal's envelope. Resistor R15 acts as a bleeder resistor for C9, otherwise the latter would "overflow". Amplification of the envelope is done by IC3a before being smoothed a little by capacitor C10

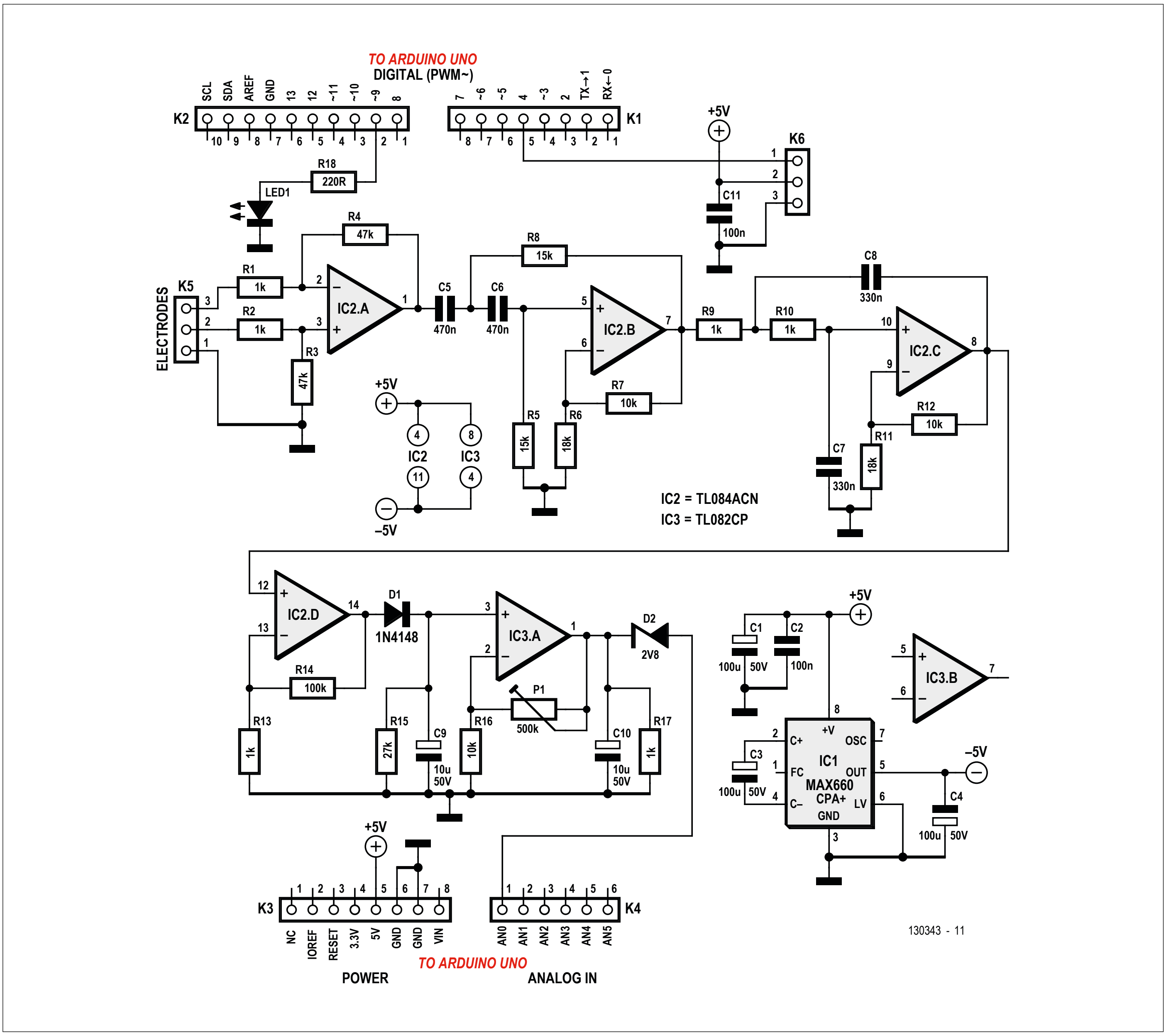
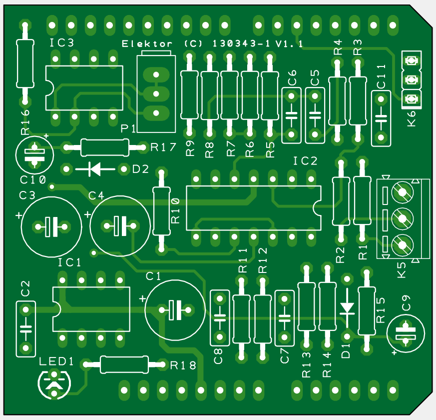
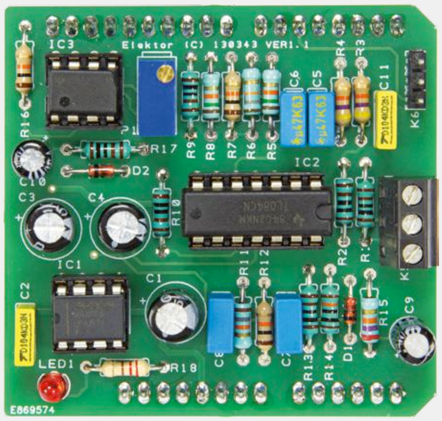


Figure 2. A differential amplifier, a bandpass filter and an envelope follower prepare the EMG signal for digitizing.

COMPONENT LIST





Resistors

Default: 250 mW 5%
R1,R2,R9,R10,R13,R17 = 1kΩ
R3,R4 = 47kΩ
R5,R8 = 15kΩ
R6,R11 = 18kΩ
R7,R12,R16 = 10kΩ
R14 = 100kΩ
R15 = 27kΩ
R18 = 220Ω
P1 = 500kΩ, 24-turn trimpot

Capacitors

C1,C3,C4 = 100µF, 50V, 2mm pitch
C2,C11 = 100nF, 0.2” pitch
C5,C6 = 470nF, 50V, 0.2” pitch
C7,C8 = 330nF, 50V, 0.2” pitch
C9,C10 = 10µF, 50V, 2mm pitch

Semiconductors

D1 = 1N4148
D2 = 1N5224B-TP
IC1 = MAX660CPA
IC2 = TL084ACN
IC3 = TL082CP
LED = red, 3mm

Miscellaneous

K1,K3 = 8-pin pinheader, 0.1” pitch
K2 = 10-pin pinheader, 0.1” pitch
K4 = 6-pin pinheader, 0.1” pitch
K5 = 3-way PCB screw terminal block, 3.5mm pitch
K4 = 3-pin pinheader, 0.1” pitch
PCB # 130343-1

(with its bleeder resistor R17).
A 2.8-V zener diode (D2) is placed in the signal path before the signal reaches the analog-to-digital converter (ADC) of the microcontroller on the Arduino board. This diode prevents false triggering of the software due to variations in the EMG signal.
The gain of IC3a is variable and can be precisely adjusted with trimmer P1. The LED shows when the idle signal (i.e. relaxed muscle) is approximately within range.
The circuit is built as a 5-volt Arduino shield. A MAX660 voltage converter (IC1) inverts the supply voltage in order to obtain a ±5 V symmetrical supply of for the shield.

Software

The software was written within the Arduino IDE for the Arduino Uno. The input signal is connected to an ADC input at pin A0 and it is converted into a numerical value. Using the Arduino map function this value is brought into the 0–180 range corresponding to the number of degrees the servo is supposed to turn. The scaled ADC values are then passed to the servo motor which is controlled drawing on Arduino’s Servo Library. The servo motor should be connected to pin 9 of the Arduino Uno.
While on servo motors, most of them have three wires coming out of them: Positive (+), Negative (–) and Control. Usually Positive is the middle wire but not always, so make sure to connect your servo the right way around.

A word on safety

It is tempting to connect the Arduino to the USB port of a computer. In the case of a battery-powered laptop that should not be a problem, but when the computer is connected to the AC grid a security issue exists as it is — in theory — possible to create a connection between the electrodes and the AC outlet which is dangerous for the person carrying the electrodes. Hence our strong advice **to power the circuit from a 9-V battery or a 5-V USB power bank only**. Do everything you can to avoid accidentally hooking up the test subject to the domestic AC lines!

Bring on the guinea pig!

Plug the shield on the Arduino, and connect the electrodes to the shield. Stick the electrodes on a muscle in an arm or leg of the guinea pig (i.e. the test subject). The skin under the electrodes should be cleaned before attaching them; they should stick well to the skin. The third electrode is important too, and should be connected to the ground of the shield. If this electrode is not connected to ground or improperly attached then the amplifiers will saturate and muscle contractions will not be detectable.
Tell the test subject to relax the muscle to which the system is attached and adjust trimmer P1 so that the LED turns on. It may blink a little, but always on is better.
Contracting the muscle should make the servo rotate while the LED turns off. After relaxing the muscle, the servo motor should reverse back and the LED should turn on again.
Care should be taken to wear shoes when playing with this circuit, i.e. the test subject must be isolated from protective earth (PE) otherwise interference on the reference signal may cause random results.

(130343)

Web Link

[1] www.elektormagazine.com/130343