

Práctica de Electrofisiología: Desarrollo de una mano con movimiento a partir de un EMG.

Daniel Gómez Pérez

Facultad de Ciencias, Universidad Nacional Autónoma de México

Resumen

Cuando realizas un movimiento con alguna de nuestras extremidades la mayoría del tiempo no estamos conscientes de todo el proceso que conlleva hacerlo. En esta práctica se da un ejemplo sobre como es todo ese proceso electrofisiológico para realizar el movimiento. Se desarrolla una mano que con ayuda de un servomotor, es capaz de cerrarse y abrir cada vez que el biceps se contrae o relaja respectivamente. Todo esto con ayuda de un electromiografo (EMG) y programación en ARDUINO. El alumno sera capaz de realizar esta sencilla practica para poder comprender mejor el funcionamiento del sistema nervioso y locomotor. Así como comenzar a adentrarse en el desarrollo de mecanismos más complejos como lo son protesis mioeléctricas.

Objetivo

Que el alumno desarrolle un mano que se mueva con un servomotor programado en ARDUINO y gracias a la señal adquirida por un EMG. Así como que obtenga una mejor comprensión sobre como es el proceso electrofisiológico para poder realizar un movimiento con una extremidad.

1. Introducción

Cuando estamos disfrutando de una tarde calurosa en el parque con nuestro perro y le arrojamos una pelota para que vaya por ella, normalmente no somos conscientes de todo el trabajo que hace nuestro cuerpo para realizar un movimiento que, desde nuestra perspectiva, es bastante simple y sin mucha complejidad. Si analizamos un poco más de cerca esta simple acción seremos capaces de comprender un poco, lo fascinante y complejo que es el cuerpo humano. Para poder lanzar una pelota, nuestro cerebro tiene que ser capaz de comunicarse y coordinar las contracciones de todos los músculos de nuestro bra-

zo; eso sin tomar en cuenta que al mismo tiempo debe de percibir todo su entorno, para saber como se deben de realizar dichas contracciones y que la pelota llegue a donde la persona quiera. Para que el cerebro, que se puede ver como el centro de control de nuestro cuerpo, se comunique con los músculos, utiliza el sistema nerviosos.

El sistema nervioso es un conjunto de células especializadas en conducir señales eléctricas, basicamente lo que hace el sistema nervioso es captar estímulos externos o internos, procesar la información y generar una respuesta acorde a la situación. El sistema nervioso está compuesto por dos tipos de células, conocidas como: neuronas y células gliales, a su vez desde el punto de vista anatómico se divide en: Sistema nervioso central (SNC) y Sistema nervioso periférico (SNP); el primero, compuesto por la médula espinal y el encéfalo, es el encargado de recibir estímulos, memorizar la información recibida, gestionarla y crear una respuesta que se enviará y provocará el movimiento, por otro lado, el SNP, está formado por el sistema sensor y motor, el primero se encarga de percibir la información y de comunicar la respuesta a los músculos utilizando el segundo; desde el punto de vista funcional se divide en: Sistema nervioso autónomo y Sistema nervioso somático. [1] [2]

Retomando el ejemplo del lanzamiento de la pelota, el movimiento para lanzarla se puede resumir como: 1) El sistema sensor de la persona ha detectado que se tiene que lanzar la pelota, debido a el lugar y las condiciones en las que se está. 2) Siguiendo las vías aferentes del sistema sensor, esta información va hacia la médula espinal, a través de la cual llegará a los lugares en los que se gestionará y se decidirá lanzar la pelota. 3) La decisión transcurre por la médula espinal en dirección al sistema motor (en forma se estímulo eléctrico), que a través de las vías eferentes de este último, es comunicado a los músculos agonistas del movimiento para el lanzamiento de la pelota, siendo este el resultado del proceso.

En esta practica se desarrollará un pequeño dispositivo, el cual consiste en un circuito en el cual por medio de la

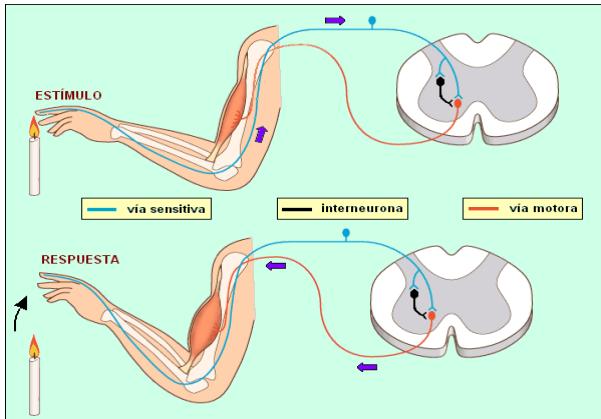


Figura 1: Esquema que muestra el camino general que sigue una señal en el cuerpo humano a partir de un estímulo y hasta su respuesta para generar un movimiento. En este caso el movimiento de un reflejo. Se tiene que el estímulo viaja a través de las vías aferentes hasta la médula espinal y luego de regreso a través de las vías eferentes hasta los músculos. Figura tomada de <https://alejandracardenas18.wordpress.com/2015/04/25/arcos-reflejos/>

señal adquirida por un EMG y un ARDUINO se pueda mover un servomotor que moverá una mano hecha de cartón, que exemplifica el proceso de movimiento de una extremidad de una manera sencilla: se tiene un estímulo (contracción del músculo) que viaja (a través del EMG) y es procesada (en el ARDUINO) generando una respuesta (movimiento del servomotor), que es lo que realiza nuestro cuerpo (Fig. 1).

La práctica estará basada en el dispositivo construido por BackyardBrain, usando su placa y código. Para más información se puede consultar su página. [5]

Ahora que ya se dio una breve introducción sobre lo que consiste la práctica, se dará una breve introducción sobre los principales materiales que se utilizarán, con el fin de familiarizarse con su funcionamiento.

Servomotor. Se conocen popularmente como "servos" y son útiles por su pequeño tamaño, bajo consumo de energía y alta precisión. Un servomotor se compone de un pequeño motor de CC, una transmisión de engranajes que reduce la velocidad y aumenta la resistencia, y un pequeño control de circuito que hace posible mover el motor con precisión utilizando solo tres cables, uno para control y los otros dos para alimentación y tierra. El PWM (Pulse Width Modulation) es el sistema más comúnmente utilizado para el control de servo. Este sistema genera una onda cuadrada que cambia la cantidad de tiempo que el pulso está alto, manteniendo el mismo período; una duración de alto nivel de la señal indica la posición donde queremos poner el eje del motor. Un

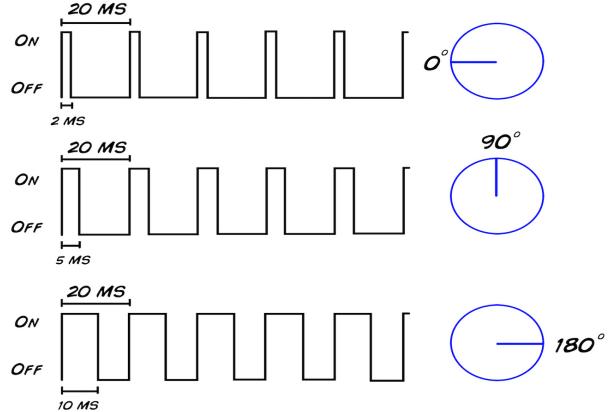


Figura 2: Esquema que muestra los pulsos PWM que mueven el servomotor a los grados que se necesiten. Figura tomada de [5]

potenciómetro integrado en el circuito de control monitorea el ángulo actual del servo: si el eje está en ángulo recto, entonces el motor está apagado. El circuito comprueba que si el ángulo no es correcto, el servo corregirá la dirección hasta que el ángulo sea el correcto. Normalmente, el eje del servo puede alcanzar alrededor de 180 grados, pero en algunos servos alcanza los 210 o 360 grados. La duración del impulso indica el ángulo de rotación del motor (Fig.2). [5]

EMG. Un electromiógrafo puede ser útil para la medición directa de la reacción muscular y muy eficiente, como medidor de las reacciones del sistema nervioso. Este instrumento capta las señales provenientes de los músculos del paciente por medio de electrodos localizados en la piel, mientras el paciente regula de manera consciente o voluntaria la contracción o relajación de los grupos musculares. Las señales provenientes de los electrodos, son filtradas, integradas, amplificadas y graficadas en forma proporcional a los niveles de contracción y relajación del músculo, de manera que cuando el músculo está tenso la gráfica se eleva y cuando el paciente relaja el músculo, desciende (Fig. 3). [6]

2. Metodología

- 1) En un rectángulo de **cartón de 20x30 cm** se dibuja el contorno de la mano y parte del brazo de una persona, y posteriormente se procede a cortarse.
- 2) En la mano de cartón se marcan con una linea los lugares en donde se encontrarían las articulaciones, y se doblan un poco.
- 3) Se cortan **cinco tiras de hilo cañamo de 30 cm** y una de sus puntas se pega con cinta en la punta de un dedo, se hace lo mismo para las cinco cintas.

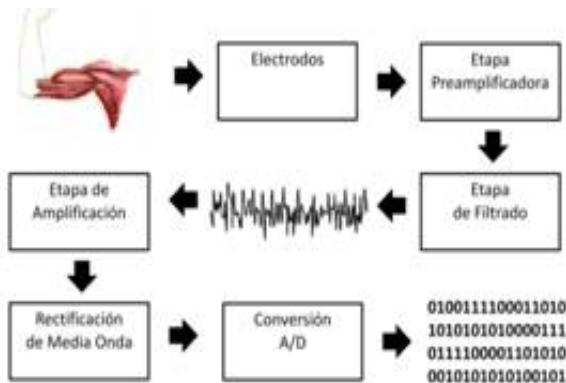


Figura 3: Esquema de bloques para la adquisición de señales EMG. Figura tomada de [6]

- 4) Se colocan los hilos sobre la mano como si fueran los huesos de la mano, y despues con una **engrapadora**, se engrapan en cada una de la articulaciones de tal manera que el hilo quede debajo de la grapa. Tambien se coloca una grapa sobre los hilos en la palma.
- 5) En la parte trasera de la mano de cartón se le colocarán tiras de cartón del grosor de los dedos para darle soporte, serán una por cada dedo y dos tiras más gruesas, una encima de la otra, en la parte del brazo . En la parte del brazo se le hará un corte de un rectangulo de la medida de la base del **servomotor MG995**, en donde se colocará y se sujetará con silicón.
- 6) Se conecta el **EMG spiker shield** a la PC con el cable azul y se abre el programa ARDUINO. (Si no se cuenta con el software, visitar la página <https://www.arduino.cc/en/Main/software> para su descarga gratuita)
- 7) Al abrir el software lo primero será configurar los puertos. En la pestaña de "Herramientas" se selecciona la placa "Arduino/Genuino Uno" y el puerto en donde se haya conectado de la PC.
- 8) Despues en la terminal se escribirá el código que se encuentra en [5] o en el *anexo* de esta práctica. El código lo que hace es usar los valores que se registran por medio de los electrodos para poder mover el servomotor y encender los LED de la placa. Para más detalle sobre el código, consultar los comentarios en el código.
- 9) Una vez escrito el código, se compila y se sube a la placa ARDUINO. Si existe un error en la escritura no se podrá subir, por lo que se tendrá que verificar y corregir.

- 10) Se conecta el cable naranja al EMG spiker shield. Despues con un poco de **alcohol y algodón** se limpian las zonas del brazo en donde se colocarán los electrodos, en este caso serían la parte distal del biceps y la parte anterior de la palma. Una vez limpias las zonas, se coloca un poco de **electrode gel** y se colocan los **tres electrodos**, dos juntos en el biceps y el otro en la palma.

- 11) Se conectan los electrodos por medio de los cai-manes del cable naranja, los rojos se conectan en los electrodos del biceos y el negro en el de la palma.
- 12) Se conecta el servomotor al EMG spiker shield, siguiendo el codigo de colores de la placa, y se le conecta una **pila de 9v** a la placa por medio de su adaptador en el conector negro y otra pila al arduino.
- 13) El sujeto deberá de comenzar a realizar contracciones. Se deberá de mover el servomotor con cada contraccií y dependiendo de su amplitud será el movimiento del servomotor. Una vez que el servomotor este en cero grados, se le colocará una de las helices que vienen con él, y se sujetarán los hilos a la hélice lo más estirados que sea posible sin doblar la mano.
- 14) Hacer contracciones con distinta intensidad y observar cuales LED se encienden.

En el *Anexo* se pueden observar algunas fotos que ilustran el procedimiento que se debe de seguir y que sirven como guía.

3. Resultados

Movimiento de la mano de cartón a partir de un EMG (Fig. 4 y 5).

4. Discusión de resultados.

Se construyo un dispositivo que describe de manera muy sencilla el proceso que realiza el sistema nervioso y aparato locomotor para realizar un movimiento. Se hizo una mano con materiales caseros que es capaz de realizar distintos movimientos dependiendo de que tan intensa sea la contracción en el biceps. El sistema que se construyo tiene un estímulo, vías de comunicación, un sistema que gestiona la información y una respuesta.

A pesar de ser un dispositivo sencillo es muy útil para entender y poder hacer analogías entre el cuerpo humano y los principios físicos que lo rigen, nos hace un poco más fácil de entender la idea de el cuerpo humano

como una maquinaria hechas de tejidos.

Preguntas de estudio.

¿Puedes reconocer que parte del dispositivo representa el estímulo, las vías de comunicación, sistema que gestiona la información y la respuesta?

¿Se te ocurre una aplicación para un dispositivo como este?

¿Conoces algún aparato médico que sea parecido o funcione como este sistema?

¿Qué se te ocurre para mejorar el diseño?

5. Conclusión

En la práctica se desarrolló una mano con movimiento aplicando conocimientos básicos de electrónica. A partir de un EMG se registraba el potencial de acción producido por una contracción del biceps el cual era procesado en el ARDUINO, el cual daba las instrucciones que a partir de este potencial se moviera el servomotor y enciendieran ciertos LED, logrando así visualizar la intensidad de la contracción, el movimiento del servomotor sería aprovechado para mover una mano de cartón y simular una mano que se abre y cierra. Con esto se ejemplifica la manera en la que trabaja nuestro cuerpo para realizar movimientos, se recibe un estímulo que se procesa y genera una respuesta. Este dispositivo nos da la idea básica sobre como es la biomecánica de alguna extremidad, y nos plantea el principio básico con el que trabajan las protesis mioeléctricas, que son el futuro en el reemplazo de extremidades.

Referencias

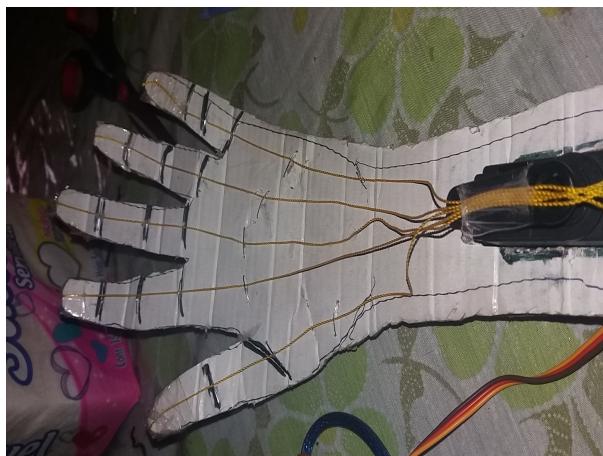


Figura 4: Mano con contracción.

[1] Guyton, Arthur C. *Textbook of medical physiology*. 11th ed.

[2] Irving P. Herman *Physics of the Human Body*. Springer

[3] Frank H. Netter, Alister Brass; *Sistema nervioso: anatomía y fisiología* Volumen de Colección Netter de ilustraciones médicas. Elsevier España, 1994

[4] Daniel P. Cardinali. *Manual de neurofisiología*.

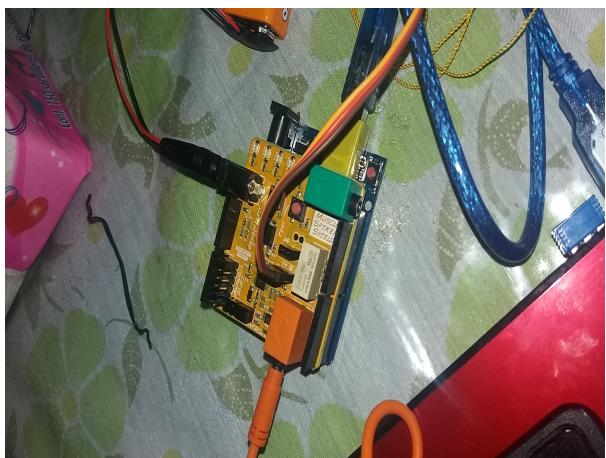
[5] <https://backyardbrains.com/products/diyMuscleSpikerShield>
Consultada el día: 30/Mayo/18

[6] Zecca M., Micera S., Carroza M., Dario P., *Control Of Multifunctional Prosthetic Hands By Processing The Electromyographic Signal*, Critical Reviews in Biomedical Engineering. Vol. 30, pp. 459-485. 2002.

6. Anexo

Fotos de la práctica y Código servomotor-EMG





```

/*
 * -----
 * Code monitors amplitude of EMG envelope, displays EMG strength on LED bar and controls
 * robotic gripper by controlling servo motor.
 * -----
 */

#include <Servo.h>
#define GRIPPER_STATE_BUTTON_PIN 4           //pin for button that switches default state
                                         //of the gripper (opened/closed)
#define SERVO_PIN 2                         //pin for servo motor
#define SENSITIVITY_BUTTON_PIN 7             //pin for button that selects sensitivity
                                         //number of LEDs in LED bar
#define NUM_LED 6                           //number of LEDs in LED bar
                                         //number of steps (used to avoid
                                         //ringing oscillation)
#define GRIPPER_MINIMUM_STEP 5
#define OPEN_MODE 1
#define CLOSED_MODE 2
#define MINIMUM_SERVO_UPDATE_TIME 1000      //update servo position every 1000ms

Servo Gripper;                          //servo for gripper
byte ledPins[] = {8, 9, 10, 11, 12, 13}; //pins for LEDs in LED bar

//EMG saturation values (when EMG reaches this value
//the gripper will be fully opened/closed)
int sensitivities[] = {200, 350, 500, 600, 800, 1000};
int lastSensitivityIndex = 2;            //set initial sensitivity index

int engSaturationValue = 0;              //selected sensitivity/EMG saturation value
int analogReadings;                    //measured value for EMG
int ledBarHeight = 0;                  //temporary variable for led bar height

unsigned long oldTime = 0;               //timestamp of last servo angle update (ms)
int oldDegrees = 0;                     //old value of angle for servo
int newDegree;                         //new value of angle for servo

unsigned long debounceTimer = 0;          //timer for button debouncer
int gripperStateButtonValue = 0;          //variable that stores state of button
int userReleasedButton = 1;              //flag that is used to avoid multiple
                                         //button events when user holds button

int currentFunctionality = OPEN_MODE;    //current default position of claw

//-----
// Setup servo, inputs and outputs
// -----
void setup(){
    //init servo
    Gripper.attach(SERVO_PIN);

    //init button pins to input
    pinMode(GRIPPER_STATE_BUTTON_PIN, INPUT);
    pinMode(SENSITIVITY_BUTTON_PIN, INPUT);

    //initialize all LED pins to output
    for(int i = 0; i < NUM_LED; i++){
        pinMode(ledPins[i], OUTPUT);
    }

    //get current sensitivity
    engSaturationValue = sensitivities[lastSensitivityIndex];
}

//-----
// Main loop
// -----
// - Checks state of sensitivity button
// - Checks state of default gripper-state button
// - Measure EMG
// - Shows EMG strength on LED bar
// - Sets angle of servo based on EMG strength and current mode (open/closed)
void loop()
{
    //-----
    // Switch sensitivity
    //-----

    //check if button is pressed (HIGH)
    if (digitalRead(SENSITIVITY_BUTTON_PIN))
    {
        //turn off all the LEDs in LED bar
        for(int j = 0; j < NUM_LED; j++)
        {
            digitalWrite(ledPins[j], LOW);
        }

        //increase sensitivity index
        lastSensitivityIndex++;
        if (lastSensitivityIndex == NUM_LED)
        {
            lastSensitivityIndex = 0;
        }

        //get current sensitivity value
        engSaturationValue = sensitivities[lastSensitivityIndex];

        //light up LED at lastSensitivityIndex position for visual feedback
        digitalWrite(ledPins[lastSensitivityIndex], HIGH);

        //wait user to release button
        while (digitalRead(SENSITIVITY_BUTTON_PIN))
        {
            delay(10);
        }

        //wait a bit more so that LED light feedback is always visible
        delay(1000);
    }

    //-----
    // Switch gripper default position open/close
    //-----

    //check if enough time has passed for button contact to settle down
    if((millis() - debounceTimer) > 500)
    {
        gripperStateButtonValue = digitalRead(GRIPPER_STATE_BUTTON_PIN);
        //if button is pressed
        if(gripperStateButtonValue == HIGH)
        {
            //if last time we checked button was not pressed
            if(userReleasedButton)
            {
                debounceTimer = millis();
                //block button events until user releases it
                userReleasedButton = 0;

                //toggle operation mode
                if(currentFunctionality == OPEN_MODE)
                {
                    currentFunctionality = CLOSED_MODE;
                }
                else
                {
                    currentFunctionality = OPEN_MODE;
                }
            }
        }
    }

    //-----
    // toggle operation mode
    // -----
    if(currentFunctionality == OPEN_MODE)
    {
        currentFunctionality = CLOSED_MODE;
    }
    else
    {
        currentFunctionality = OPEN_MODE;
    }
}

//-----
// Measure EMG
// -----
analogReadings = analogRead(A0); //read EMG value from analog input A0

//-----
// Show EMG strength on LED
// -----
//turn OFF all LEDs on LED bar
for(int j = 0; j < NUM_LED; j++)
{
    digitalWrite(ledPins[j], LOW);
}

//Calculate what LEDs should be turned ON on the LED bar
analogReadings = constrain(analogReadings, 30, engSaturationValue);
ledBarHeight = map(analogReadings, 30, engSaturationValue, 0, NUM_LED);

//turn ON LEDs on the LED bar
for(int k = 0; k < ledBarHeight; k++)
{
    digitalWrite(ledPins[k], HIGH);
}

//-----
// Drive Claw according to EMG strength
// -----
//set new angle if enough time passed
if (millis() - oldTime > MINIMUM_SERVO_UPDATE_TIME)
{
    //calculate new angle for servo
    if(currentFunctionality == OPEN_MODE)
    {
        analogReadings = constrain(analogReadings, 40, engSaturationValue);
        newDegree = map(analogReadings, 40 ,engSaturationValue, 100, 180);
    }
    else
    {
        analogReadings = constrain(analogReadings, 120, engSaturationValue);
        newDegree = map(analogReadings, 120 ,engSaturationValue, 100, 180);
    }

    //check if we are in servo dead zone
    if(abs(newDegree-oldDegree) > GRIPPER_MINIMUM_STEP)
    {
        //set new servo angle
        Gripper.write(newDegree);
    }

    oldTime = millis();
    oldDegree = newDegree;
}
}

```