



# DESARROLLO DE UN ELECTROMIÓGRAFO APLICADO AL CONTROL DE UNA MANO BIÓNICA IMPRESA EN 3D.

---

## MOTIVACIÓN

- ▶ Auge de la impresión 3D
- ▶ Disponibilidad de modelos 3D
- ▶ Alto costo de prótesis existentes en el mercado
- ▶ Tecnología disponible para reducir costos

## MOTIVACIÓN

- ▶ Auge de la impresión 3D
- ▶ Disponibilidad de modelos 3D
- ▶ MUY alto costo de prótesis existentes en el mercado
- ▶ Tecnología disponible para reducir costos

## ETAPAS

- ▶ Desarrollo electromiógrafo
- ▶ Impresión y movimiento de la mano prostética
- ▶ Control de movimiento por electromiografía

## ANTECEDENTES

- ▶ Electromiografía
- ▶ Prótesis
- ▶ Tecnologías

# ELECTROMIOGRAFÍA

- ▶ La electromiografía es la técnica de registro gráfico de la actividad eléctrica producida por los músculos esqueléticos
- ▶ Muy bajo voltaje ( $50\mu\text{V}$  a  $30\text{mV}$ )
- ▶ Método intramuscular o superficial

# ELECTROMIOGRAFÍA - INSTRUMENTAL

- ▶ Electrodos
- ▶ Amplificador
- ▶ Sistema de registro

## ANTECEDENTES

- ▶ Electromiografía
- ▶ Prótesis
- ▶ Tecnologías

# PRÓTESIS

► Mecánicas



AtomicLab

► Electrónicas



TouchBionics

# PRÓTESIS

► Mecánicas



AtomicLab

► Electrónicas



TouchBionics

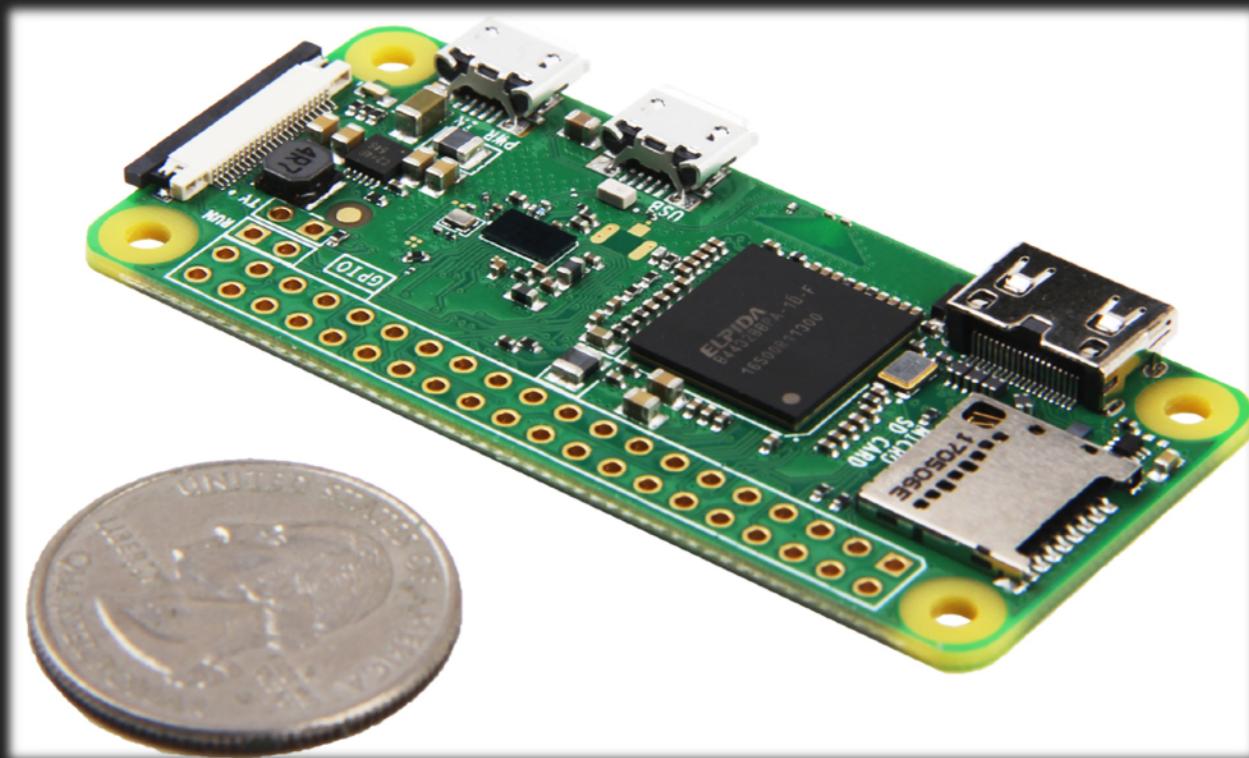
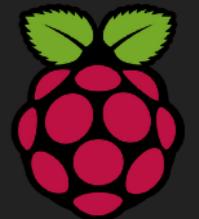
► u\$s17

► u\$s80.000 - 120.000

## ANTECEDENTES

- ▶ Electromiografía
- ▶ Prótesis
- ▶ Tecnologías

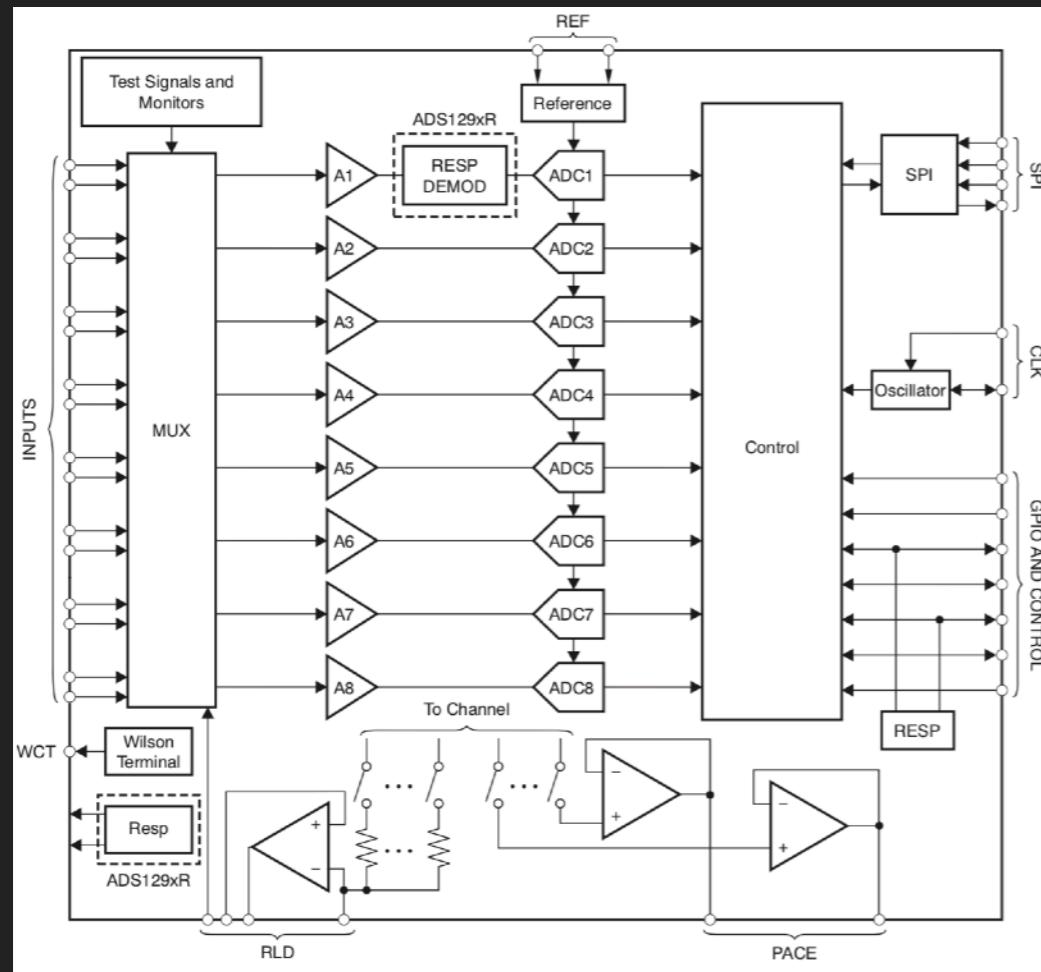
# TECNOLOGÍAS - RASPBERRY PI ZERO



- ▶ ARM11 a 1GHz
- ▶ 512MB RAM
- ▶ 1 puerto USB OTG
- ▶ Mini HDMI
- ▶ 40 Pines GPIO
- ▶ 802.11 b/g/n wireless LAN
- ▶ Bluetooth 4.1
- ▶ Bluetooth Low Energy (BLE)

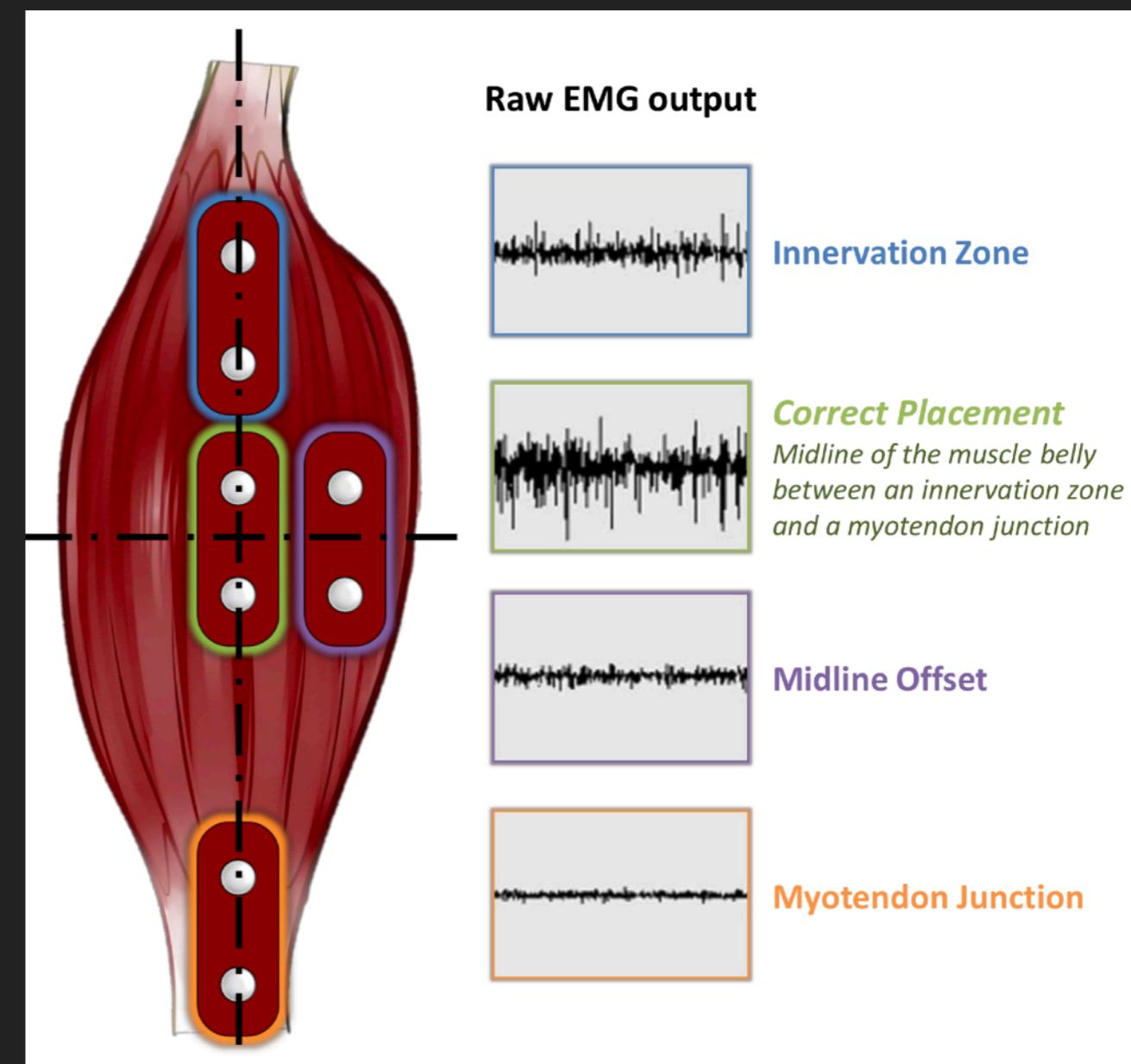
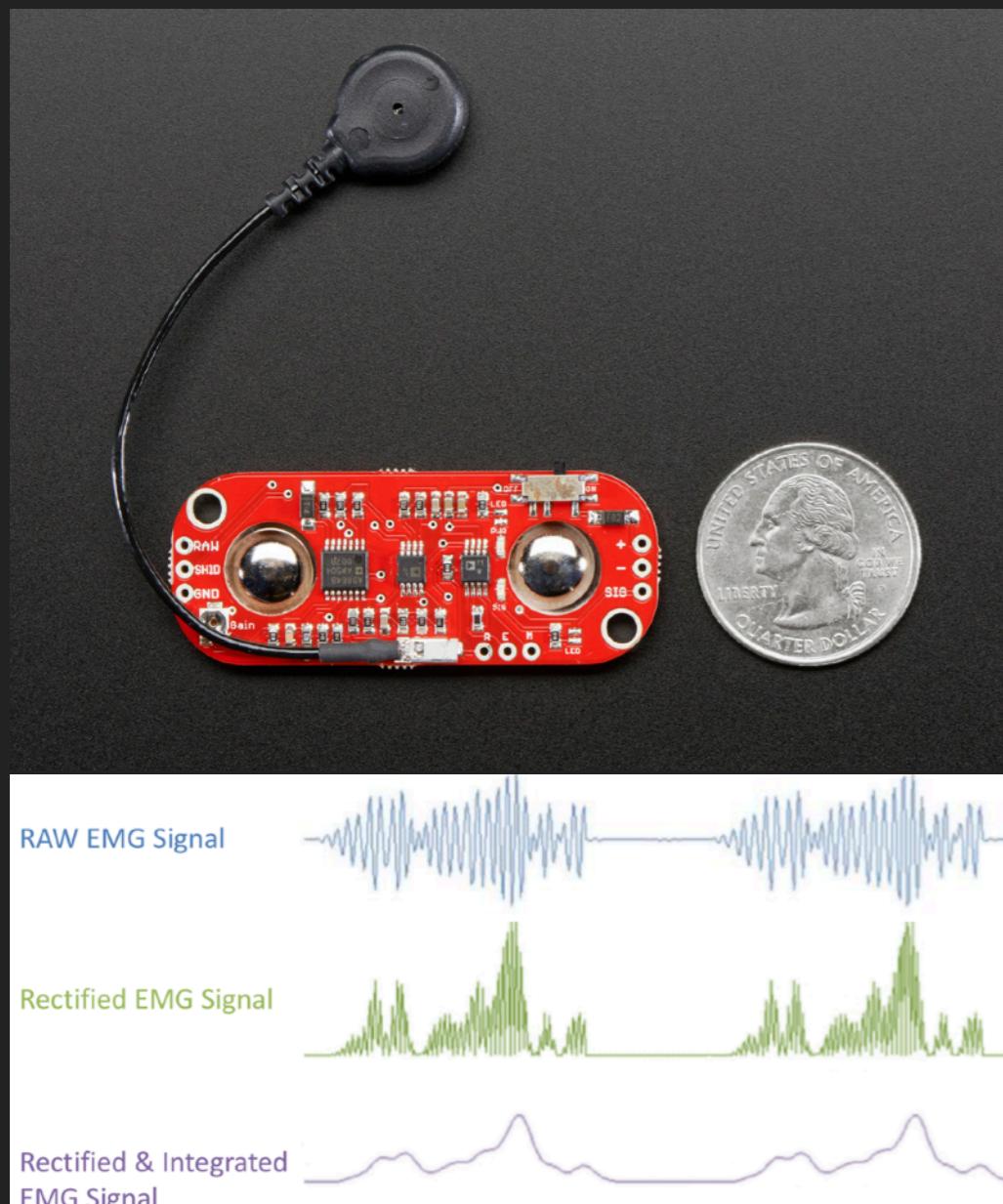


# TECNOLOGÍAS - ADS1294

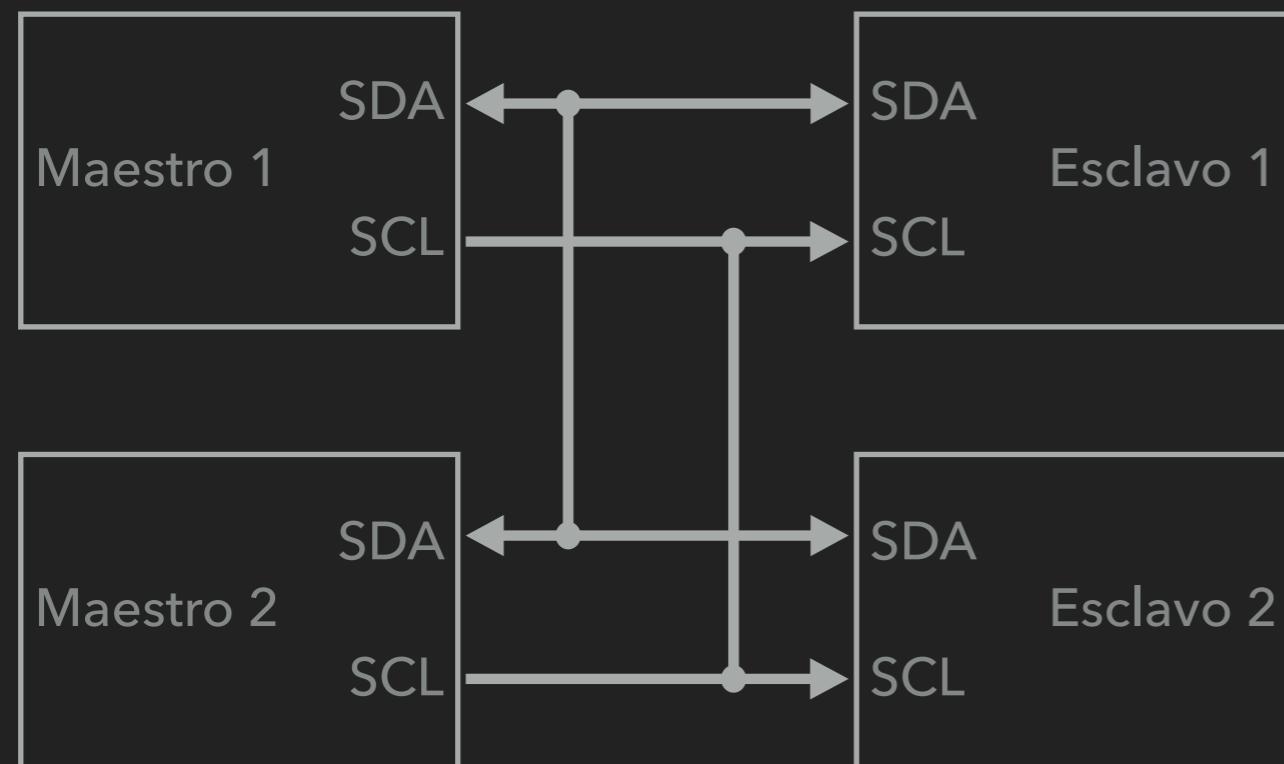


- ▶ 4 canales de alta resolución y de bajo ruido
- ▶ Bajo consumo: 0.75 mW / canal
- ▶ Frecuencia de muestreo: desde 250Hz a 32kHz
- ▶ Ganancia programable: 1, 2, 3, 4, 6, 8, o 12.
- ▶ Alimentación: Unipolar o Bipolar.
  - Analógica: 2.7V a 5.25V
  - Digital: 1.65V a 3.6V.
- ▶ Oscilador y referencia internos.
- ▶ Señales de test integradas.
- ▶ Comunicación mediante interfaz SPI.
- ▶ Rango de temperatura operativo del encapsulado obtenido: 0°C a 70°C.

# TECNOLOGÍAS - MYOWARE MUSCLE SENSOR



# TECNOLOGÍAS - I<sup>2</sup>C



- ▶ Dos líneas de señal: reloj (SCL, Serial Clock) y la línea de datos (SDA, Serial Data)
- ▶ El nivel alto debe ser de al menos  $0,7 \times VDD$  y el nivel bajo no debe ser más de  $0,3 \times VDD$
- ▶ Reloj generado por el maestro

# TECNOLOGÍAS - BLUETOOTH DE BAJA ENERGÍA (BLE)

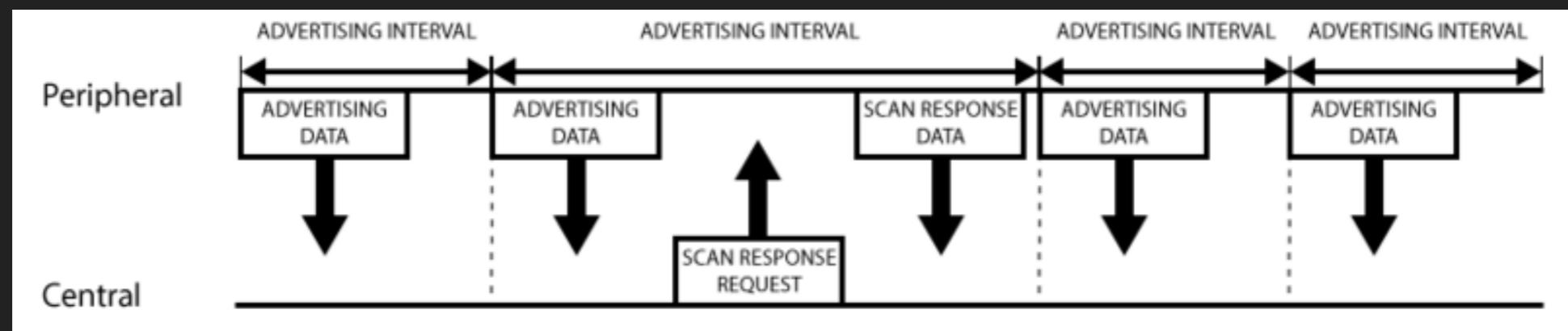


- ▶ Es parte de la especificación de Bluetooth 4.0
- ▶ Sencillo para implementar la comunicación entre pequeños dispositivos y aplicación móvil
- ▶ Servidores GAP y GATT

# TECNOLOGÍAS - BLE - GAP



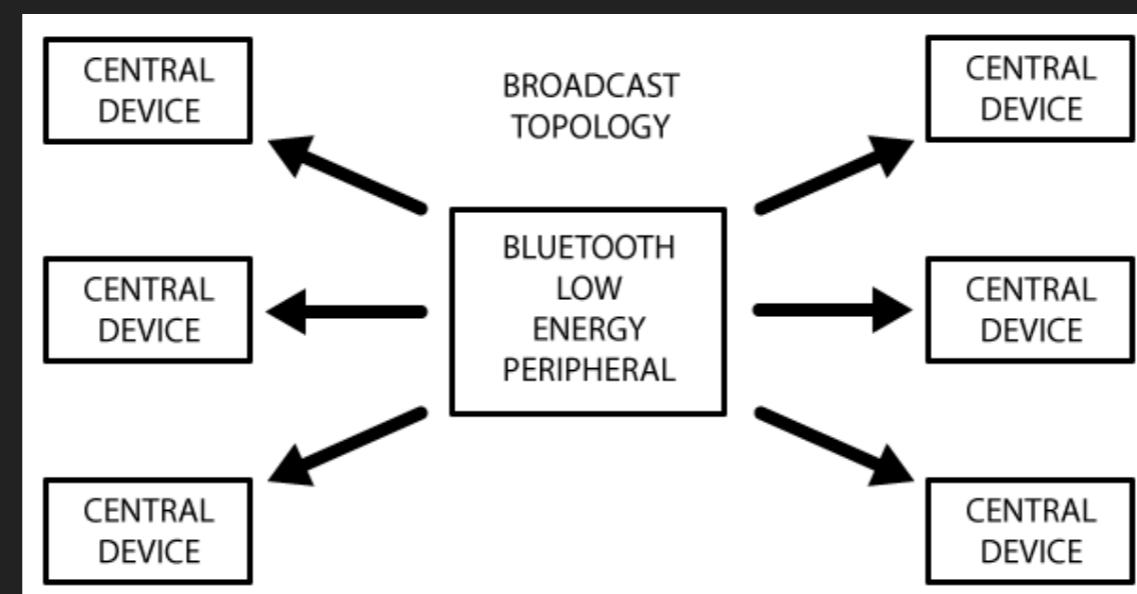
- ▶ Generic Access Profile
- ▶ Controlar las conexiones y los anuncios en BLE
- ▶ Periféricos y dispositivos centrales
- ▶ Advertising Data payload y el Scan Response payload



# TECNOLOGÍAS - BLE - MODO BROADCAST



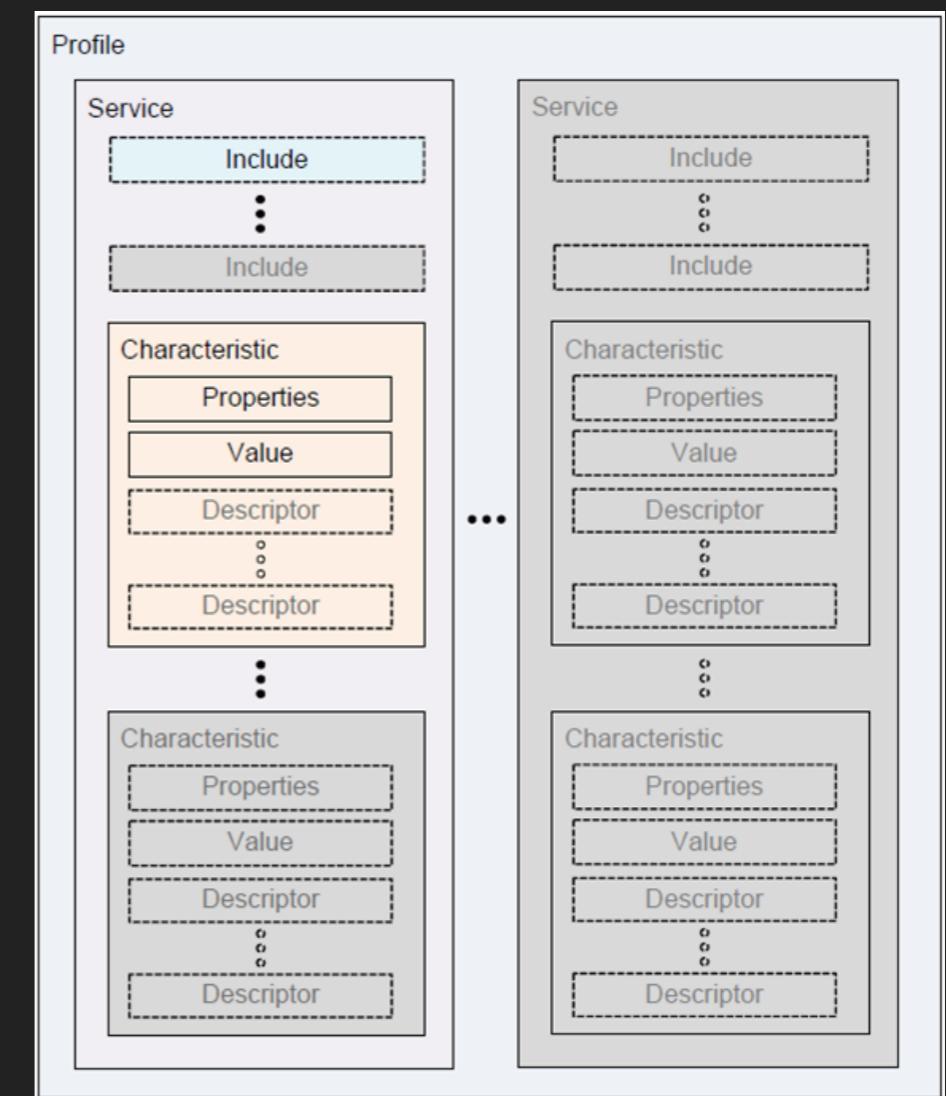
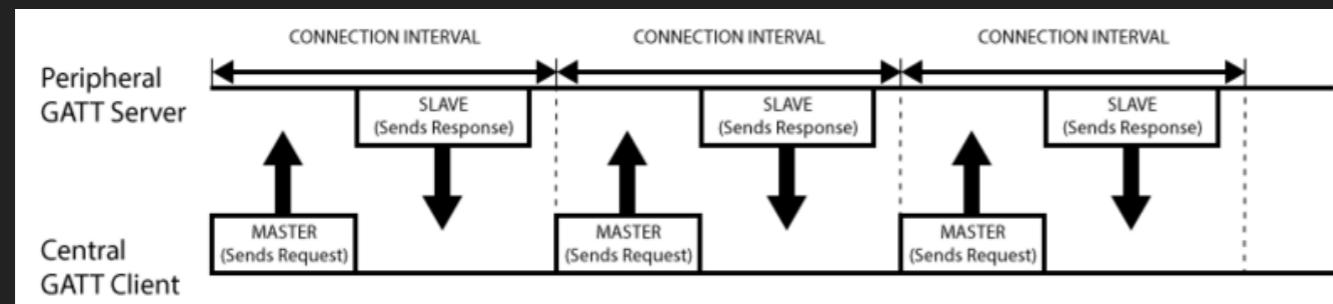
- ▶ Cuando se desea que un periférico transmita datos a más de un dispositivo a la vez
- ▶ Especificación iBeacon
- ▶ Al establecer una conexión (periférico y dispositivo central), el proceso de advertising se detiene y usará los servicios y características del GATT.



# TECNOLOGÍAS - BLE - GATT



- ▶ Generic Attribute Profile
- ▶ Las conexiones son exclusivas



# TECNOLOGÍAS - NODEJS



- ▶ Entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor basado en el lenguaje de programación JavaScript, asíncrono, con I/O de datos en una arquitectura orientada a eventos, que lo hace liviano y eficiente.

## TECNOLOGÍAS - SWIFT

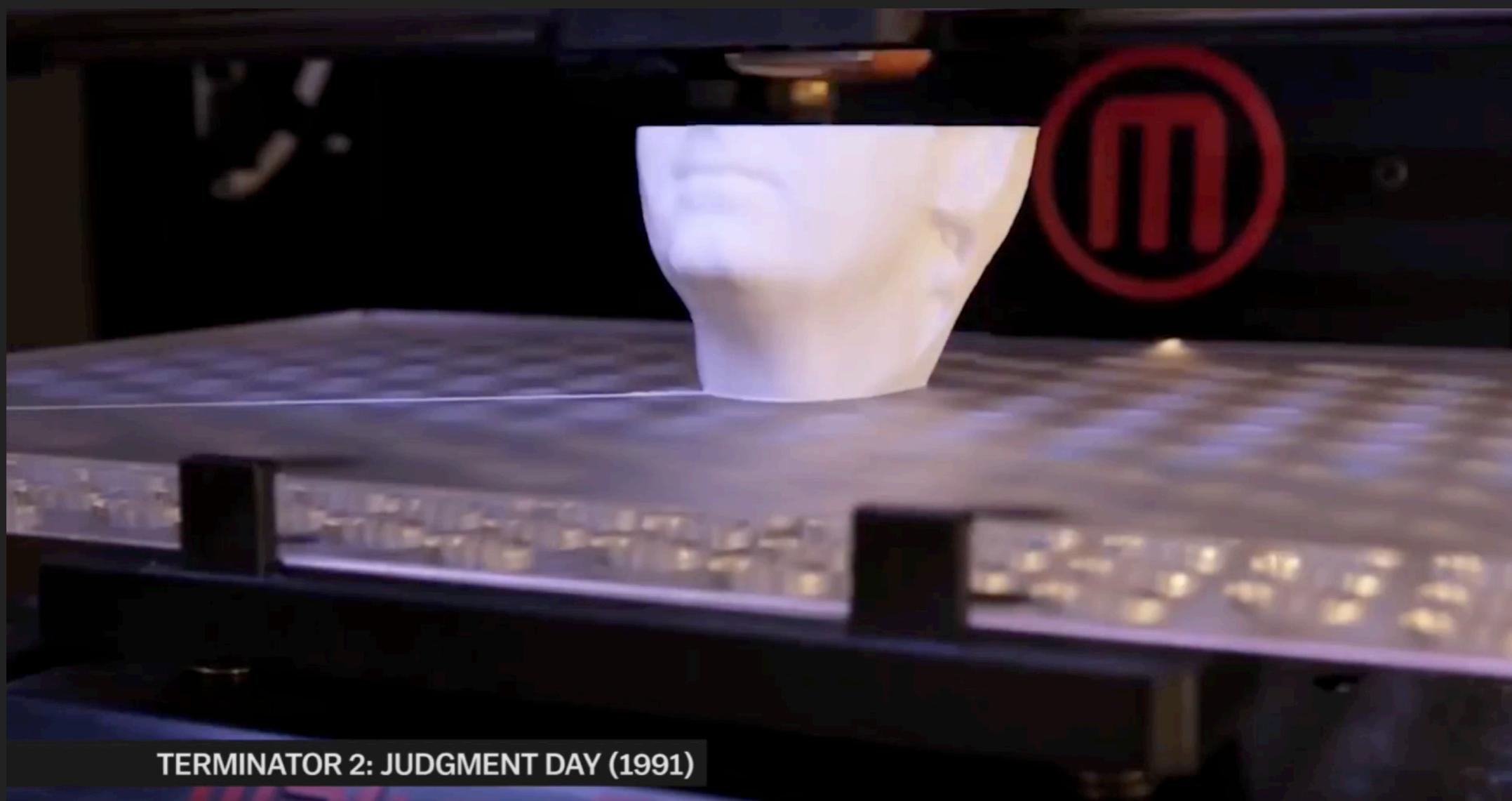


- ▶ Swift es un lenguaje de programación creado por Apple enfocado en el desarrollo de aplicaciones para iOS y macOS.
- ▶ Fue presentado en 2014.
- ▶ En el año 2015 pasó a ser de código abierto.

## TECNOLOGÍAS - IMPRESIÓN 3D

- ▶ Tecnologías de fabricación por adición donde un objeto tridimensional es creado mediante la superposición de capas sucesivas de material.
- ▶ Impresión por extrusión.
- ▶ Impresión por fotopolimerización.

# TECNOLOGÍAS - IMPRESIÓN 3D POR EXTRUSIÓN



TERMINATOR 2: JUDGMENT DAY (1991)

# TECNOLOGÍAS - IMPRESIÓN 3D POR FOTOPOLIMERIZACIÓN



# TECNOLOGÍAS - IMPRESIÓN 3D POR FOTOPOLIMERIZACIÓN

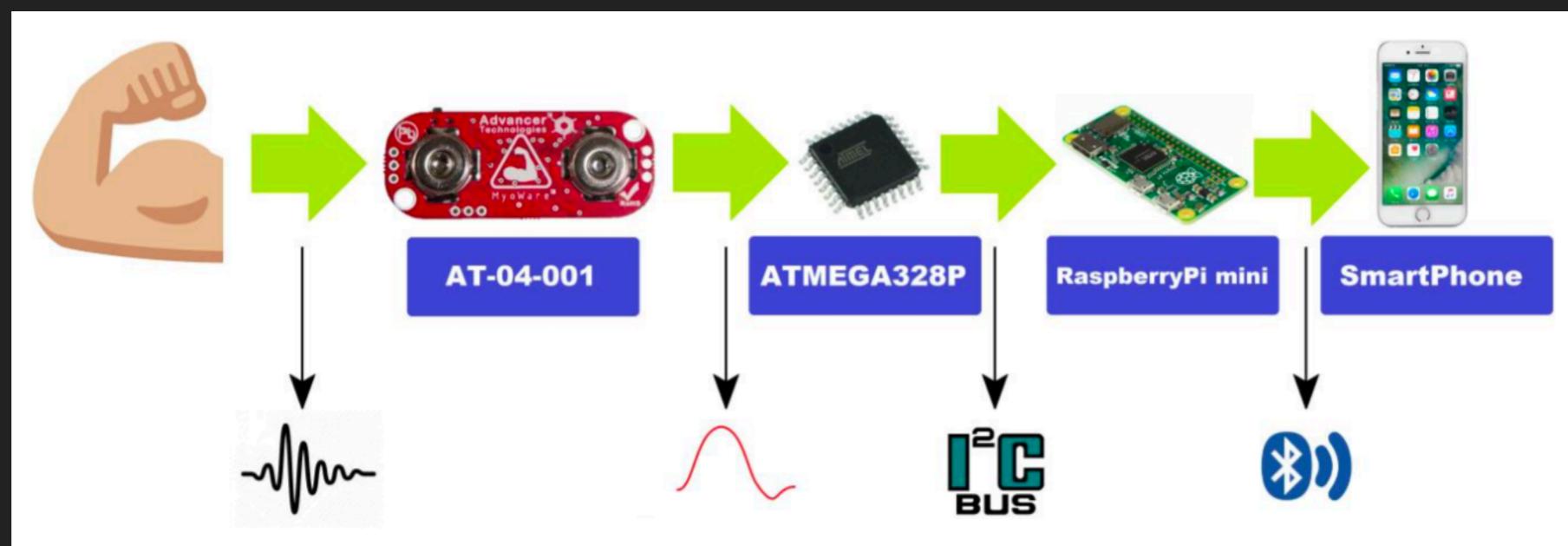


## CRITERIOS DE DISEÑO

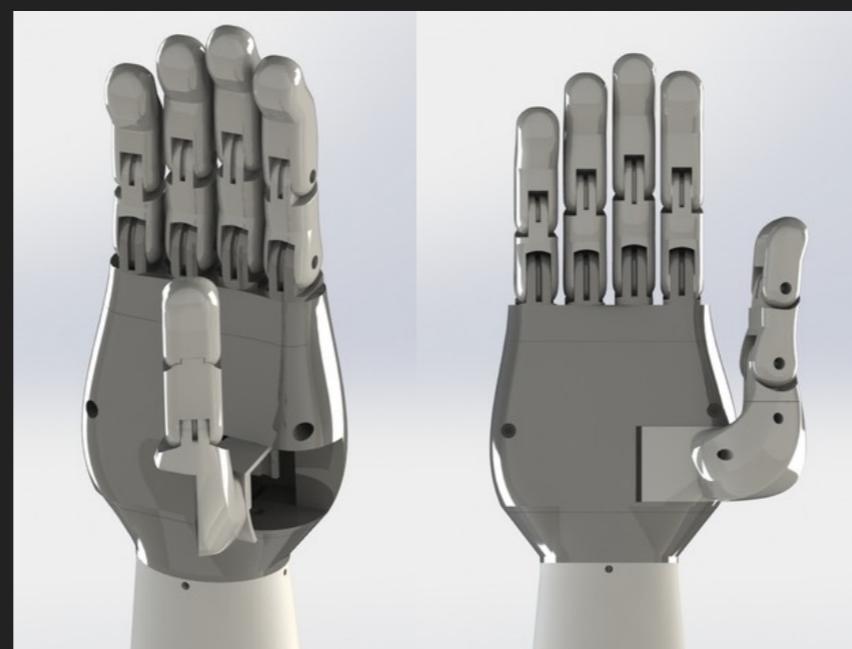
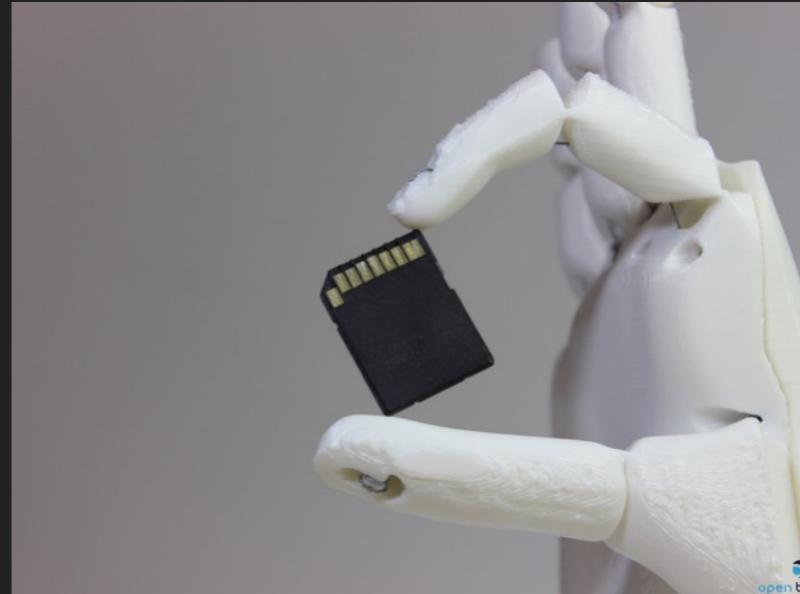
- ▶ Raspbian - Sistema Operativo
- ▶ MyoWare - Sensor actividad muscular
- ▶ JavaScript - Lenguaje servidor
- ▶ Swift - Lenguaje aplicación

## MODELO DEL SISTEMA EMG

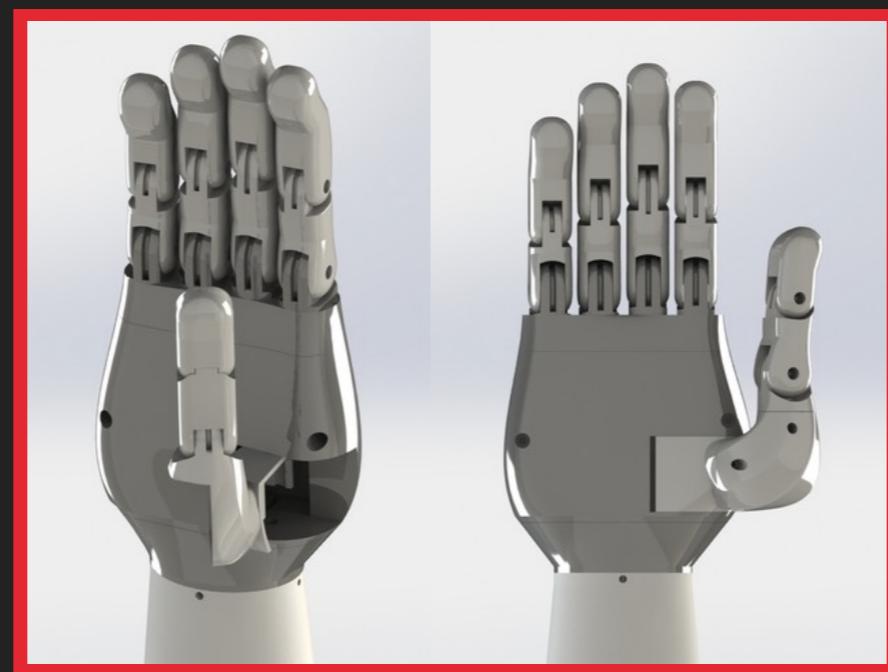
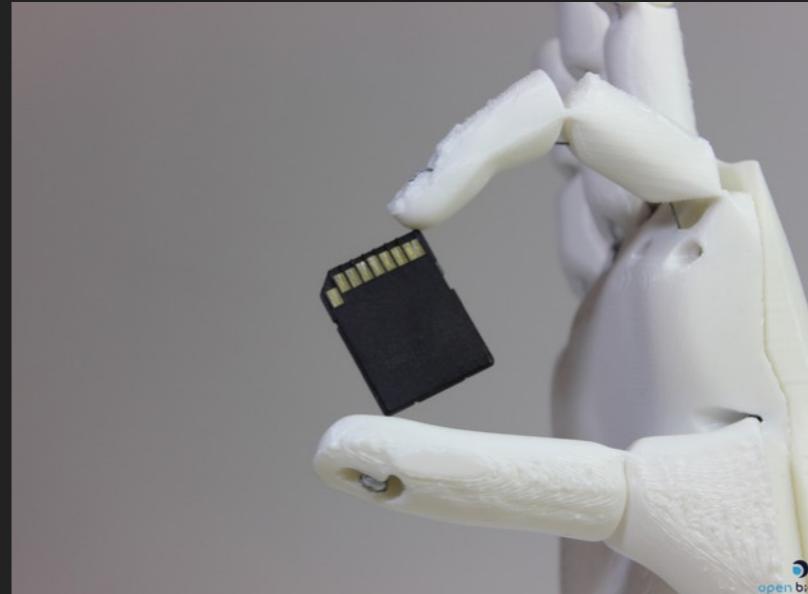
- ▶ Calidad de captura y análisis
- ▶ Facilidad de adquisición y uso
- ▶ Bajo coste



## CRITERIOS DE DISEÑO - MODELO 3D



## CRITERIOS DE DISEÑO - MODELO 3D



## ETAPAS

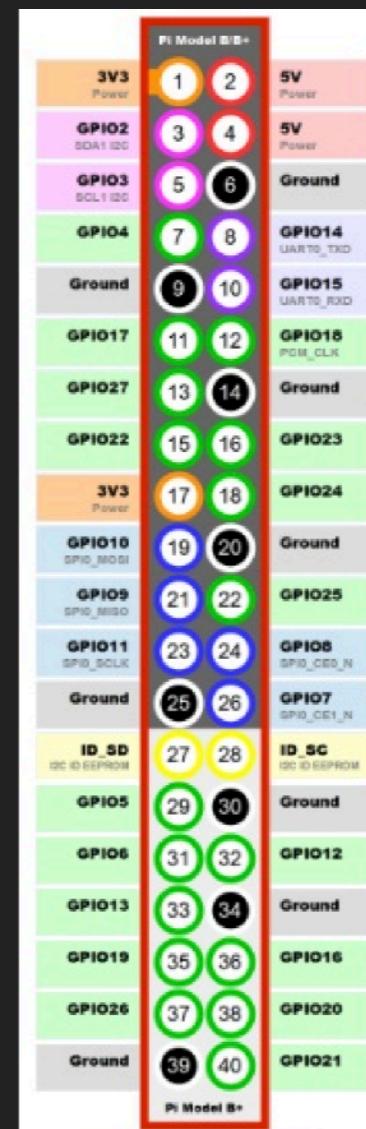
- ▶ Desarrollo electromiógrafo
- ▶ Impresión y movimiento de la mano prostética
- ▶ Control de movimiento por electromiografía

# DESARROLLO ELECTROMIÓGRAFO

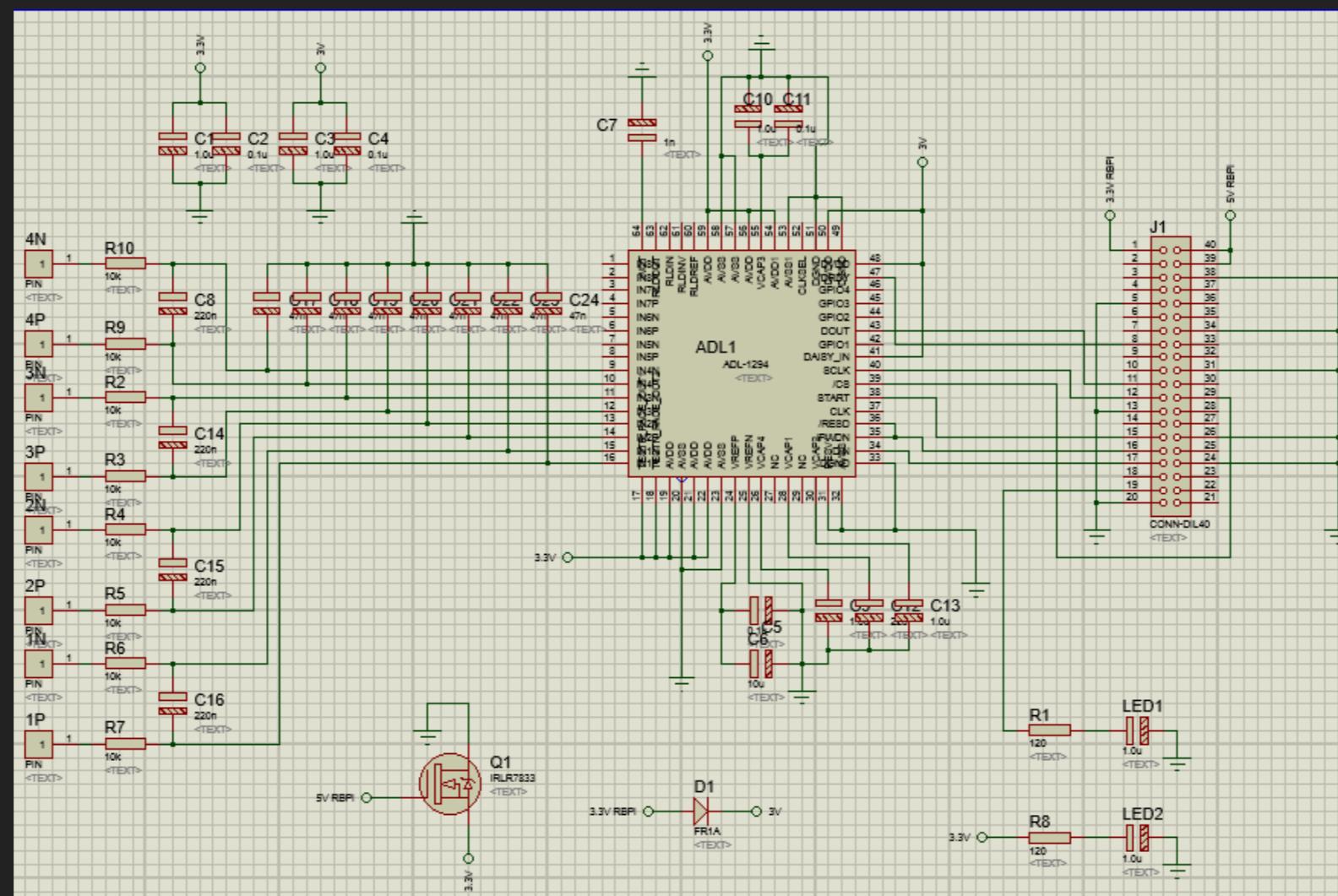
- ▶ Medición de la actividad muscular y comunicación mediante I2C.
- ▶ Procesamiento de datos y comunicación mediante BLE.
- ▶ Visualización de los datos.

# MEDICIÓN DE LA ACTIVIDAD MUSCULAR

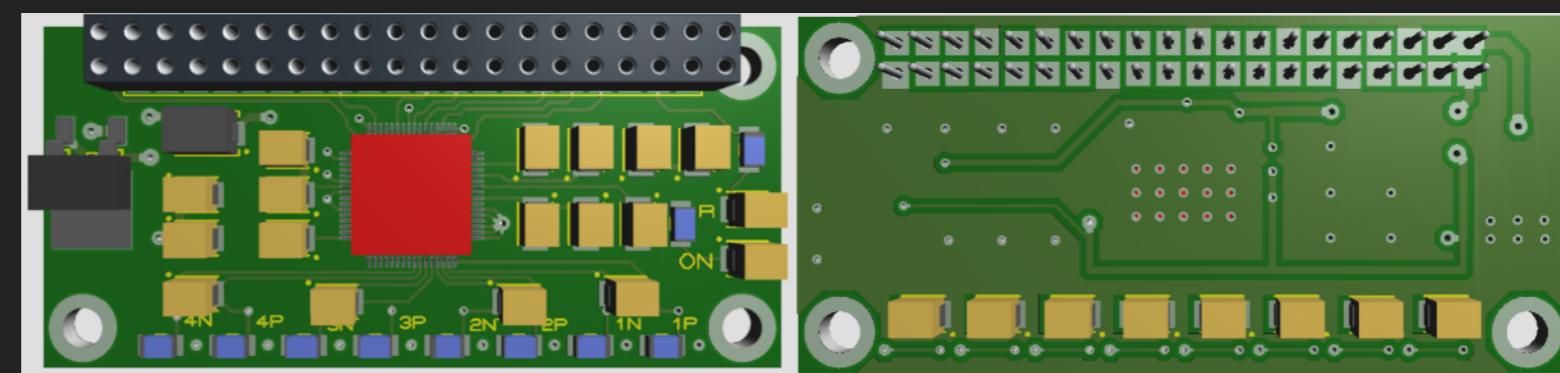
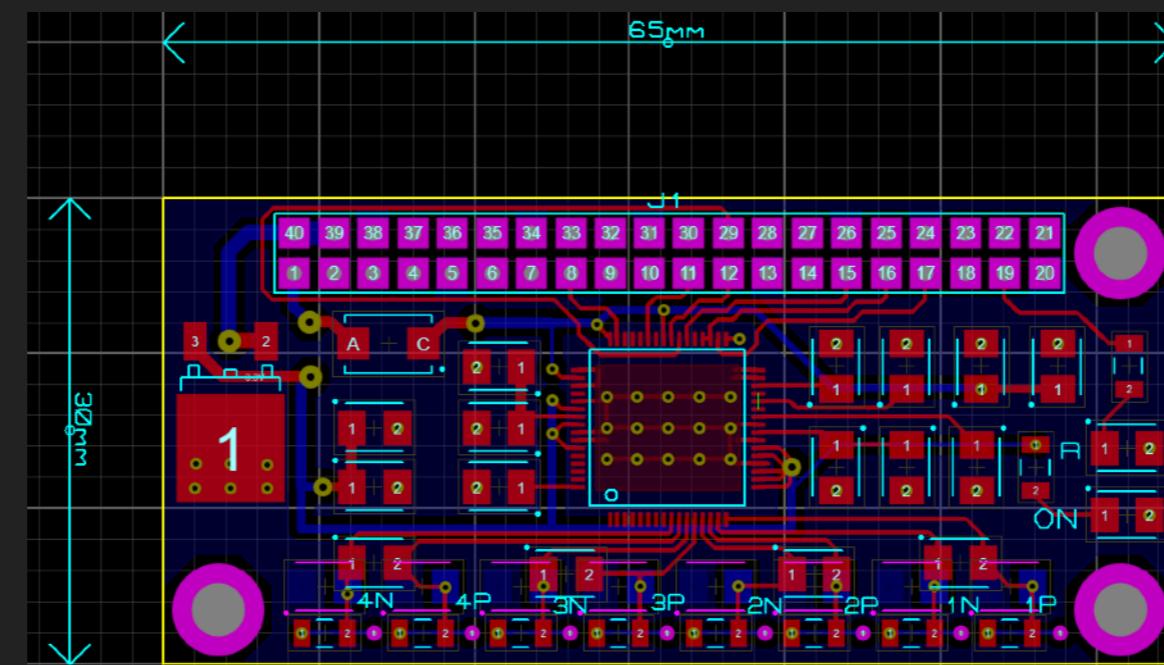
- ▶ ADS1294
- ▶ Lograr una HAT para la Raspberry Pi (65x30mm)



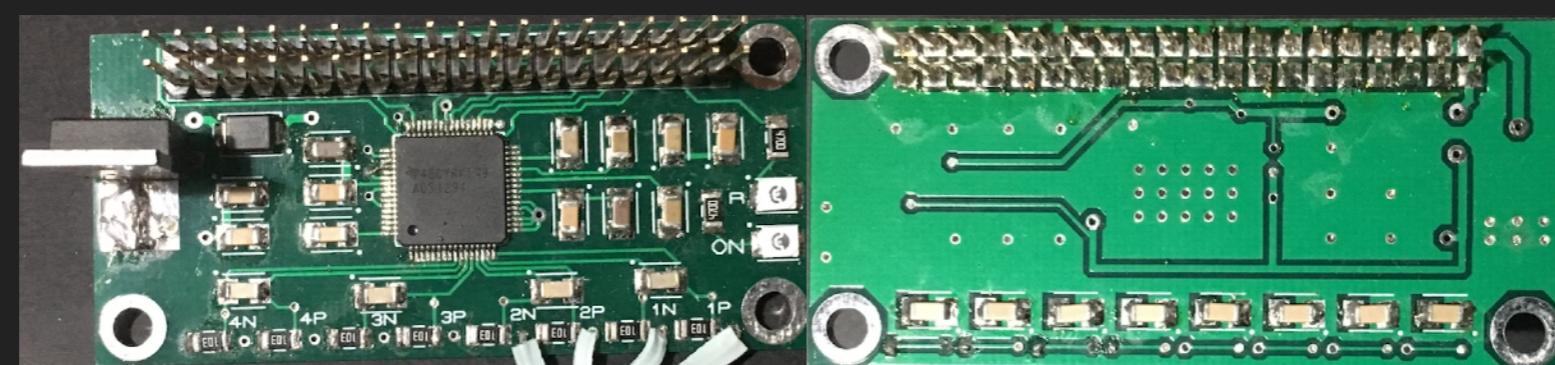
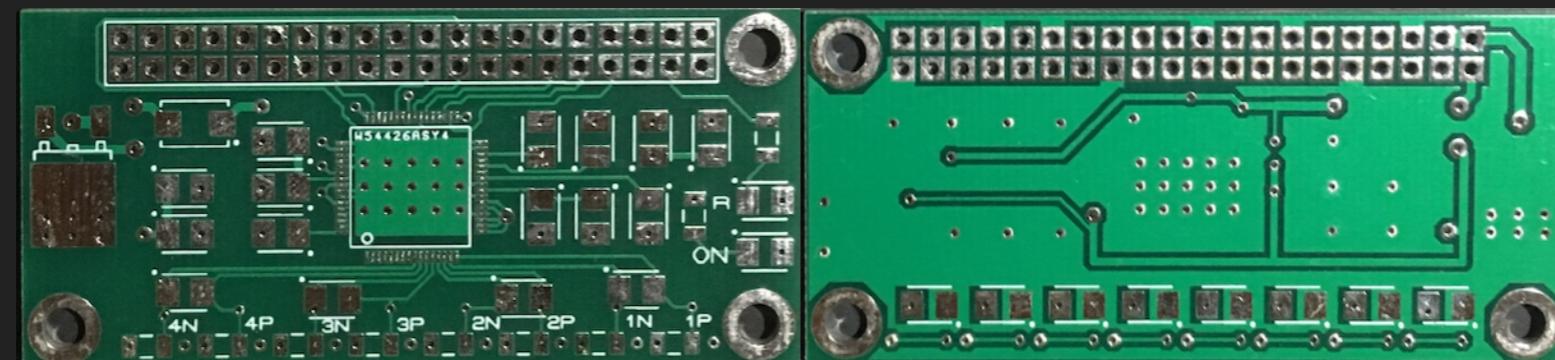
# ESQUEMÁTICO ADS1294



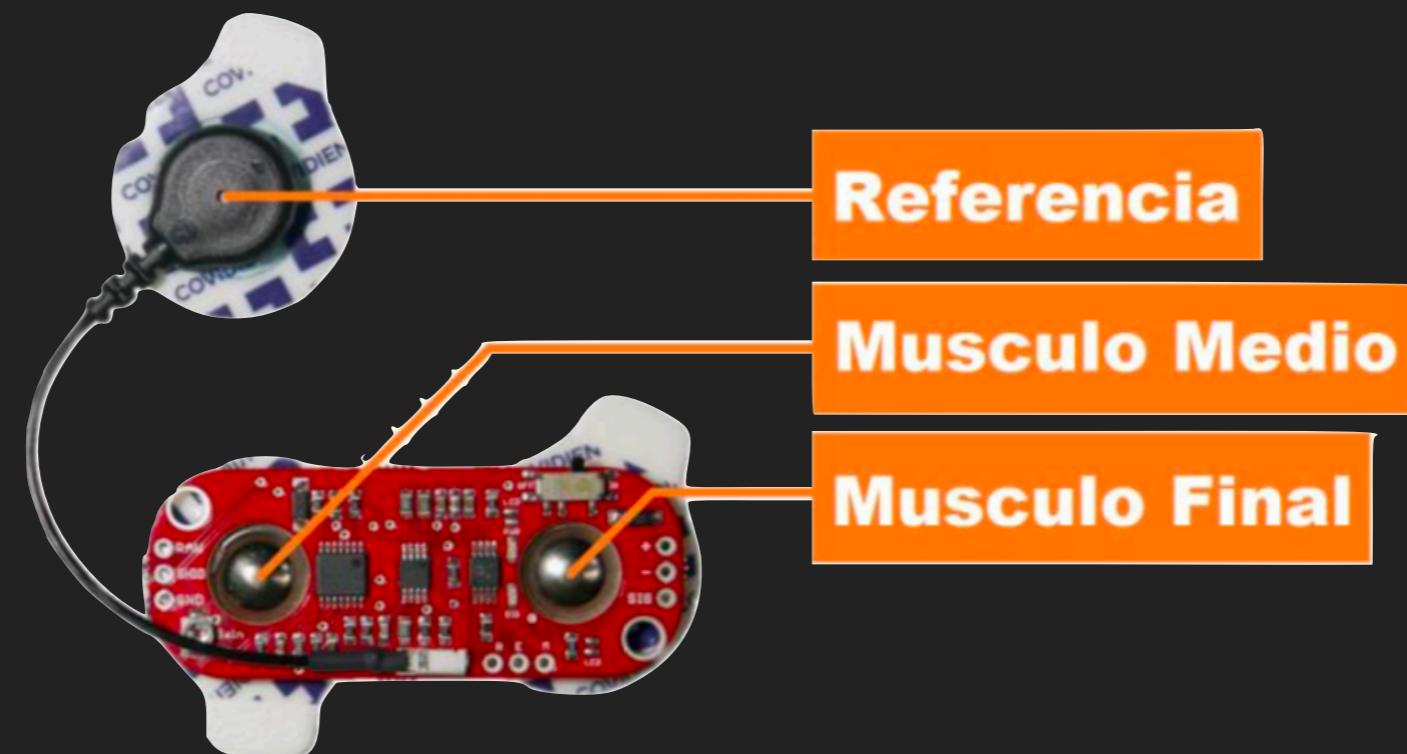
# ESQUEMÁTICO ADS1294



# PLACA FINAL



# MYOWARE MUSCLE SENSOR + ATMEGA328



# COMUNICACIÓN ENTRE DISPOSITIVOS POR I<sup>2</sup>C

- ▶ Raspberry Pi como maestro.
- ▶ Atmega328 como esclavo.
- ▶ Pedido de los datos cada 10ms.

## PROCESAMIENTO DE DATOS

- ▶ Todo el trabajo de filtrado es hecho por el sensor MyoWare.
- ▶ Servidor NodeJS, se encarga de discriminar si se debe realizar o no un movimiento.
- ▶ Actualizar el valor de la característica del servidor GATT.

# COMUNICACIÓN ENTRE DISPOSITIVOS BLE

- ▶ Servidor GATT sobre NodeJS - BLENO.

```
var BlenoPrimaryService = bleno.PrimaryService;  
var EchoCharacteristic = require('./characteristics');
```

- ▶ UUID: ec0e (lectura, escritura y notificación)
- ▶ Cliente GATT sobre aplicación móvil - CoreBluetooth.  
`setNotifyValue(true, for: miCharacteristic)`

# DESARROLLO ELECTROMIÓGRAFO

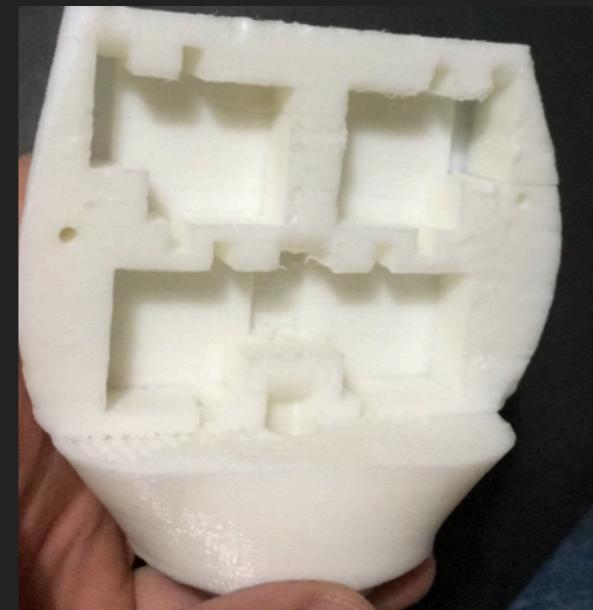
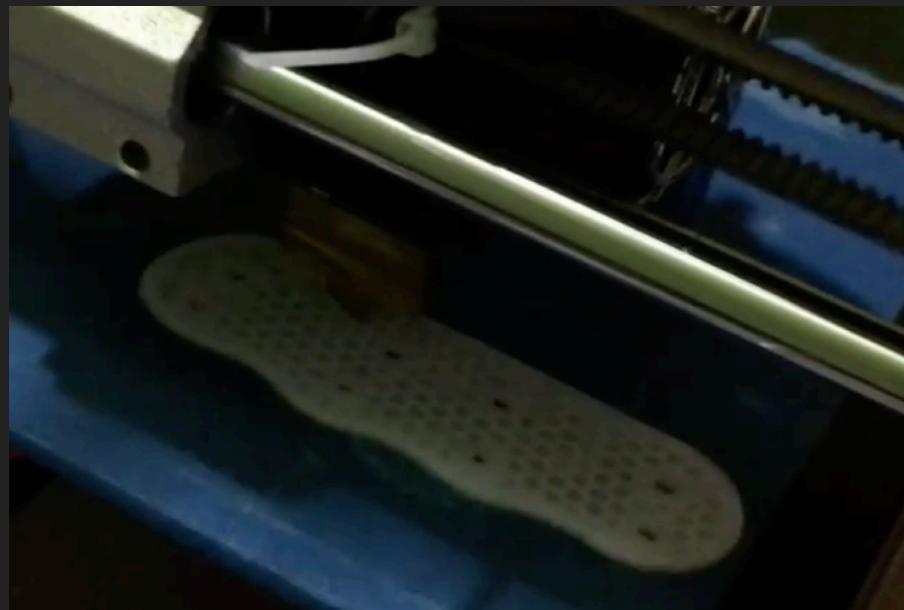
- ▶ Aplicación iOS encargada de graficar los datos recibidos por la RPi.



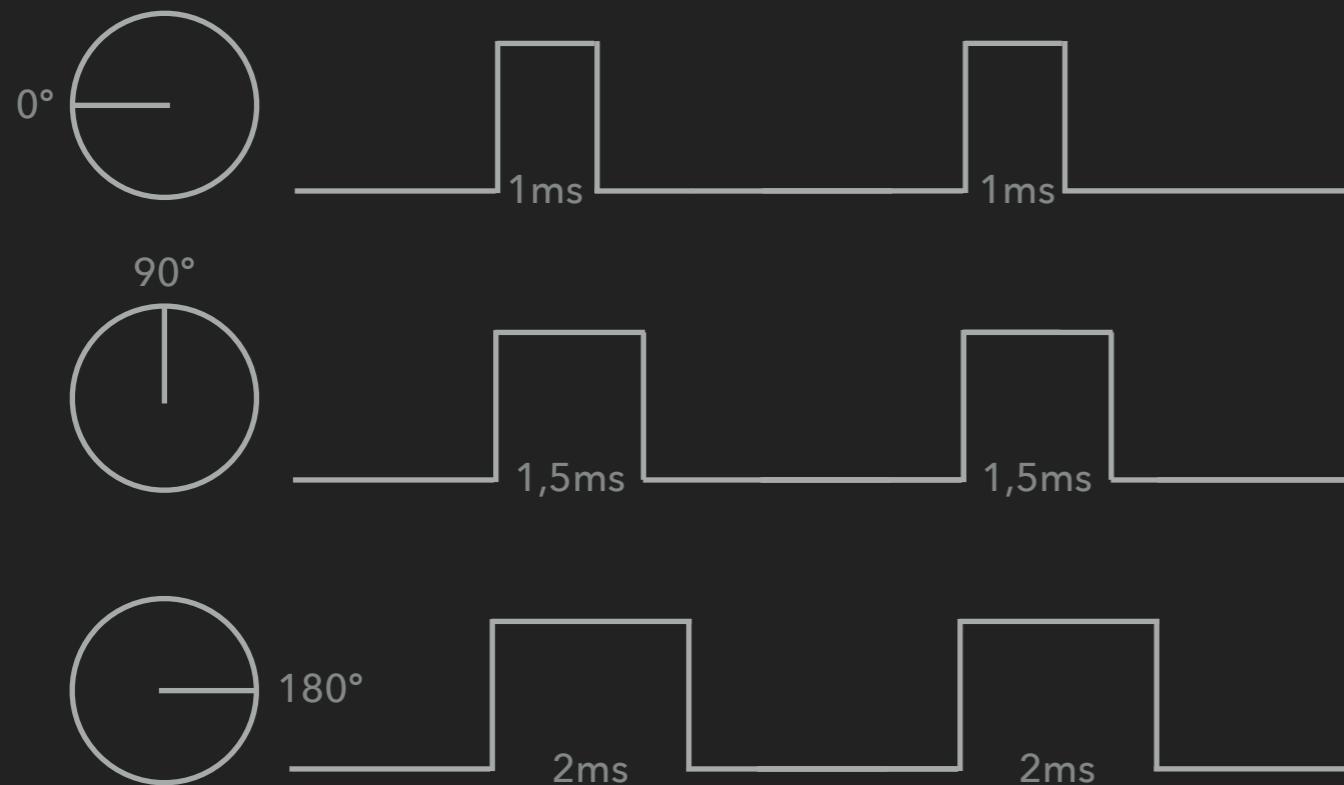
## ETAPAS

- ▶ Desarrollo electromiógrafo
- ▶ Impresión y movimiento de la mano prostética
- ▶ Control de movimiento por electromiografía

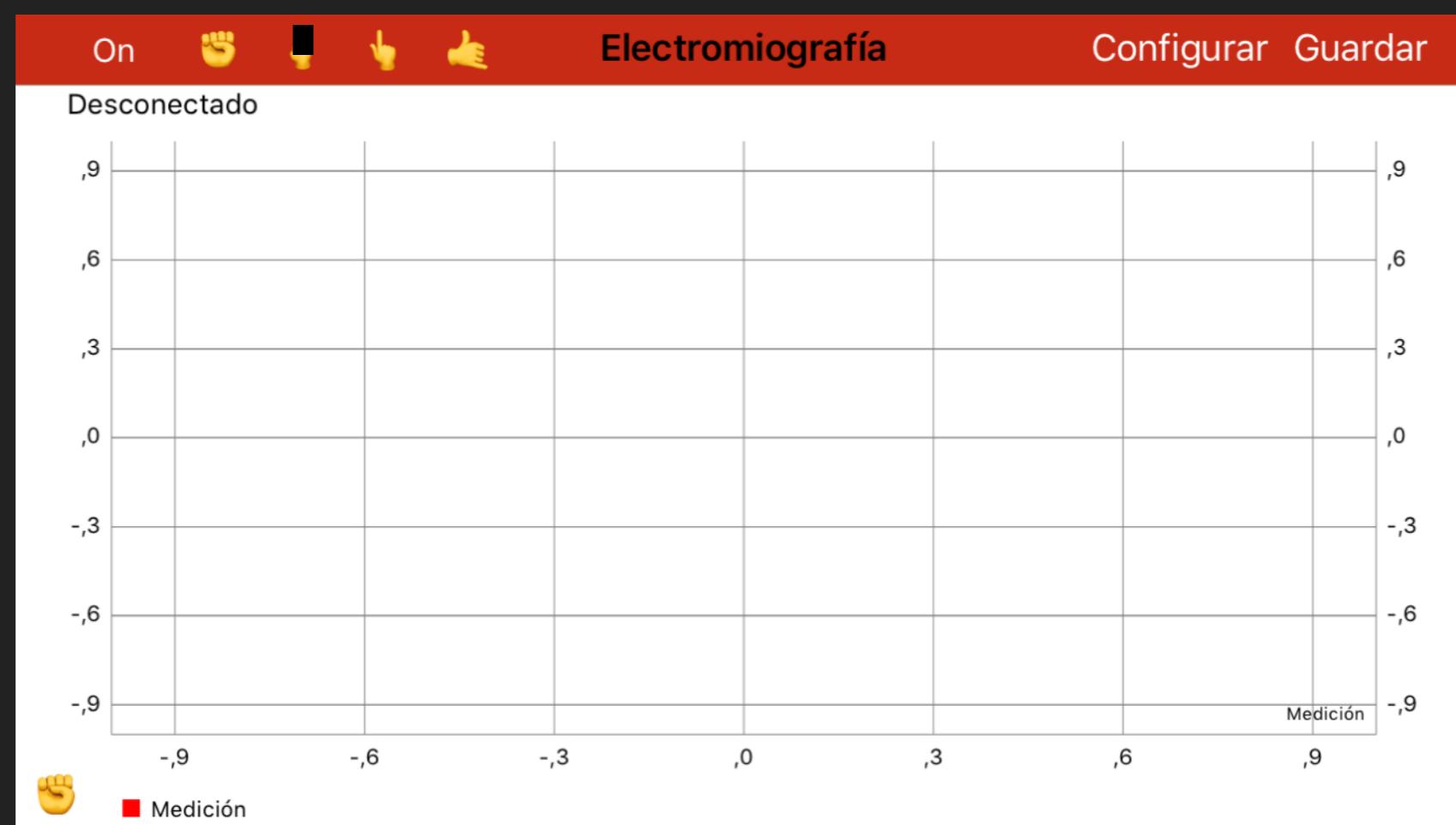
# CONSTRUCCIÓN DE LA MANO PROSTÉTICA



# MOVIMIENTO DE LA MANO



# APLICACIÓN IOS



# APLICACIÓN IOS



# COSTOS

| Descripción  | Monto   | Subtotal       |
|--|---------|----------------|
| 0,5Kg PLA  | \$ 200  | \$ 200         |
| 6 Servomotores (en Argentina)  | \$ 600  | \$ 800         |
| ADS1294 (Finalmente no utilizado)  | \$ 800  | \$ 1600        |
| Raspberry Pi Zero W (en Argentina)   | \$ 800  | \$ 2400        |
| 60 Electrodos (para realización de pruebas, para el uso solo se requieren 3) | \$ 600  | \$ 3000        |
| Fabricación de 5 placas (Finalmente no utilizadas)                           | \$ 200  | \$ 3200        |
| Costos de envío + Costos aduaneros   | \$ 2800 | \$ 6000        |
| Componentes SMD + envío  | \$ 600  | \$ 6600        |
| MyoWare muscle sensor (en Argentina)   | \$ 1000 | \$ 7600        |
| Varios (pegamento, tanza, tornillos)   | \$ 100  | \$ 7700        |
| <b>TOTAL DESARROLLO</b>  |         | <b>\$ 7700</b> |

# COSTOS

| Descripción  | Monto   | Subtotal       |
|--|---------|----------------|
| 0,5Kg PLA  | \$ 200  | \$ 200         |
| 6 Servomotores (en Argentina)  | \$ 600  | \$ 800         |
| Raspberry Pi Zero W (en Argentina)   | \$ 800  | \$ 1600        |
| 60 Electrodos (para realización de pruebas, para el uso solo se requieren 3) | \$ 600  | \$ 2200        |
| MyoWare muscle sensor (en Argentina)   | \$ 1000 | \$ 3200        |
| Varios (pegamento, tanza, tornillos)   | \$ 100  | \$ 3300        |
| <b>TOTAL DESARROLLO</b>  |         | <b>\$ 3300</b> |

## SISTEMA FINAL

- ▶ Sensor electromiográfico.
- ▶ Mano realizada con impresión 3D.
- ▶ \$2800.- → \$1900.-

## PASOS A FUTURO Y POSIBLES MEJORAS

- ▶ Mejorar robustez.
- ▶ Elección de otro sensor o mayor cantidad de sensores.
- ▶ Aplicación para Android.

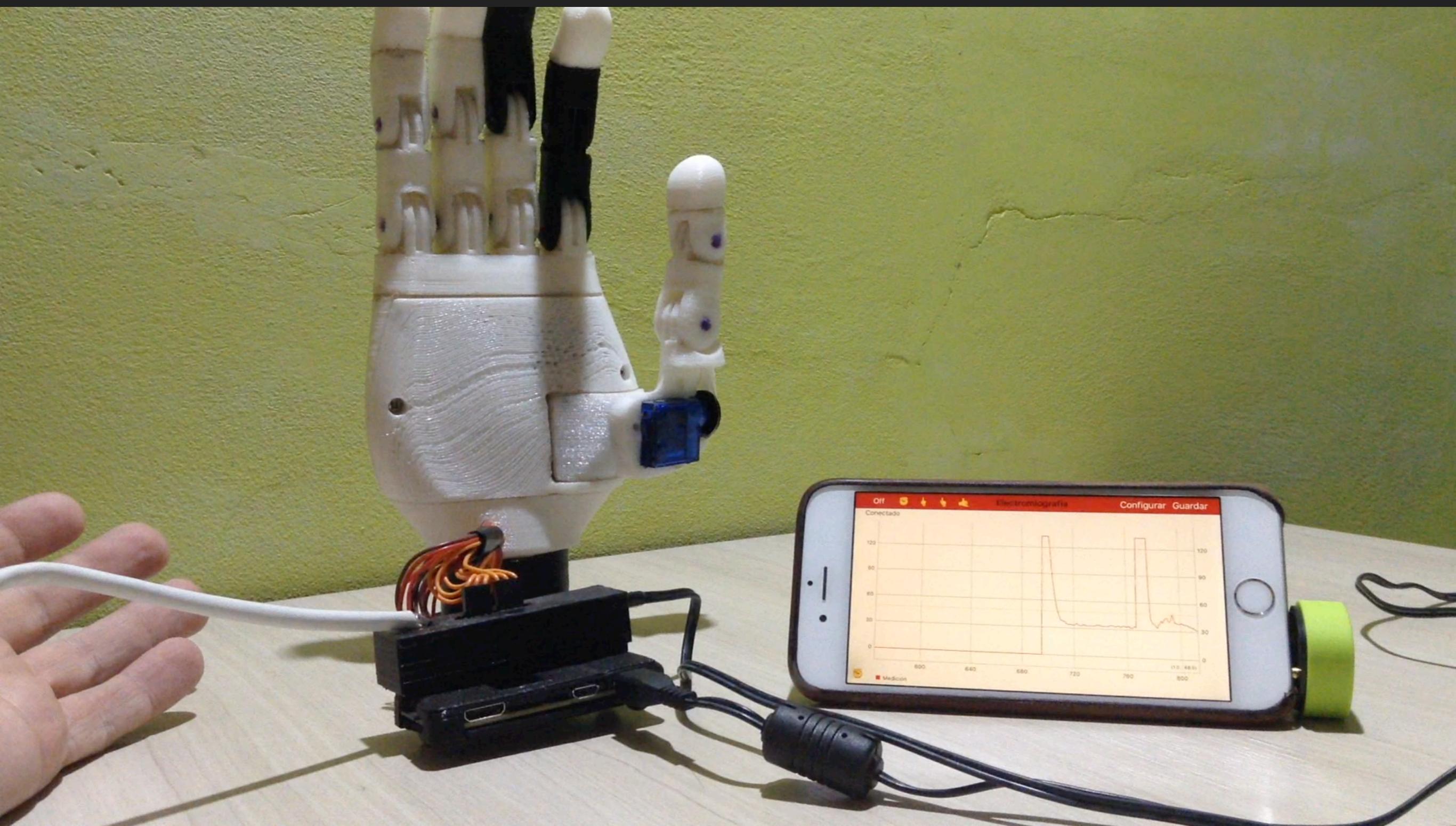


# MUCHAS GRACIAS

GUIDI, JUAN CRUZ - PROYECTO FINAL 2018

# REFERENCIAS BIBLIOGRÁFICAS

- ▶ Desarrollo de dispositivos e-Health de bajo coste para Raspberry Pi - Curso 2013/2014 - Xavier Gallofré Nieve Jesús F. López San José Álvaro Pérez Liaño.
- ▶ Sistema de Adquisición y Procesamiento Inteligente de Señales Biológicas - Norberto Scarone, Damián Marasco, Nicolás Castro, Gustavo Monte.
- ▶ Datasheet ads1298 - Texas Instruments
- ▶ Datasheet ATmega328/P - Atmel
- ▶ Datasheet MyoWare™ Muscle Sensor (AT-04-001) - Advancer Technologies
- ▶ The Swift Programming Language - Apple
- ▶ BCM2835 ARM Peripherals. - Broadcom Corporation. Broadcom Europe Ltd. 406 Science Park Milton Road Cambridge CB4 0WW, 2012.
- ▶ <https://www.thingiverse.com/thing:1103992>
- ▶ <https://www.raspberrypi.org/>
- ▶ <https://www.raspberrypi.org/blog/introducing-raspberry-pi-hats/>
- ▶ <https://nodejs.org/es/about/>
- ▶ <https://github.com/noble/bleno>
- ▶ <https://www.npmjs.com/package/i2c>
- ▶ <https://www.mayoclinic.org/es-es/tests-procedures/emg/pac-20393913>
- ▶ <https://www.sparkfun.com/products/13723>
- ▶ <http://raspi.tv/2017/how-much-power-does-pi-zero-w-use>
- ▶ <https://www.amputee-coalition.org/resources/spanish-history-prosthetics/>
- ▶ <https://atomiclab.org>
- ▶ <http://www.vocativ.com/money/industry/prosthetic-boom-3d-printed-mind-controlled-limbs/index.html>
- ▶ <http://www.monografias.com/trabajos96/bioingenieria-protesis-bionicas-manos/bioingenieria-protesis-bionicas-manos.shtml>
- ▶ <https://solidgeargroup.com/bluetooth-ble-el-conocido-desconocido?lang=es>
- ▶ <https://www.bluetooth.com/specifications/gatt/generic-attributes-overview>
- ▶ [https://es.wikipedia.org/wiki/Swift\\_\(lenguaje\\_de\\_programaci%F3n\)](https://es.wikipedia.org/wiki/Swift_(lenguaje_de_programaci%F3n))
- ▶ <https://developer.apple.com/swift/>
- ▶ <https://es.wikipedia.org/wiki/Xcode>
- ▶ <https://openbionics.com/>
- ▶ <http://exiii-hackberry.com>
- ▶ <https://www.myo.com>



**DEMO**