



Proyecto 3. Análisis en sistemas embebidos

Bioseñales

Orman Jesús Contreras Mulford

Maria Camila Melan Guerra

Facultad de Ingeniería

Programa de Bioingeniería

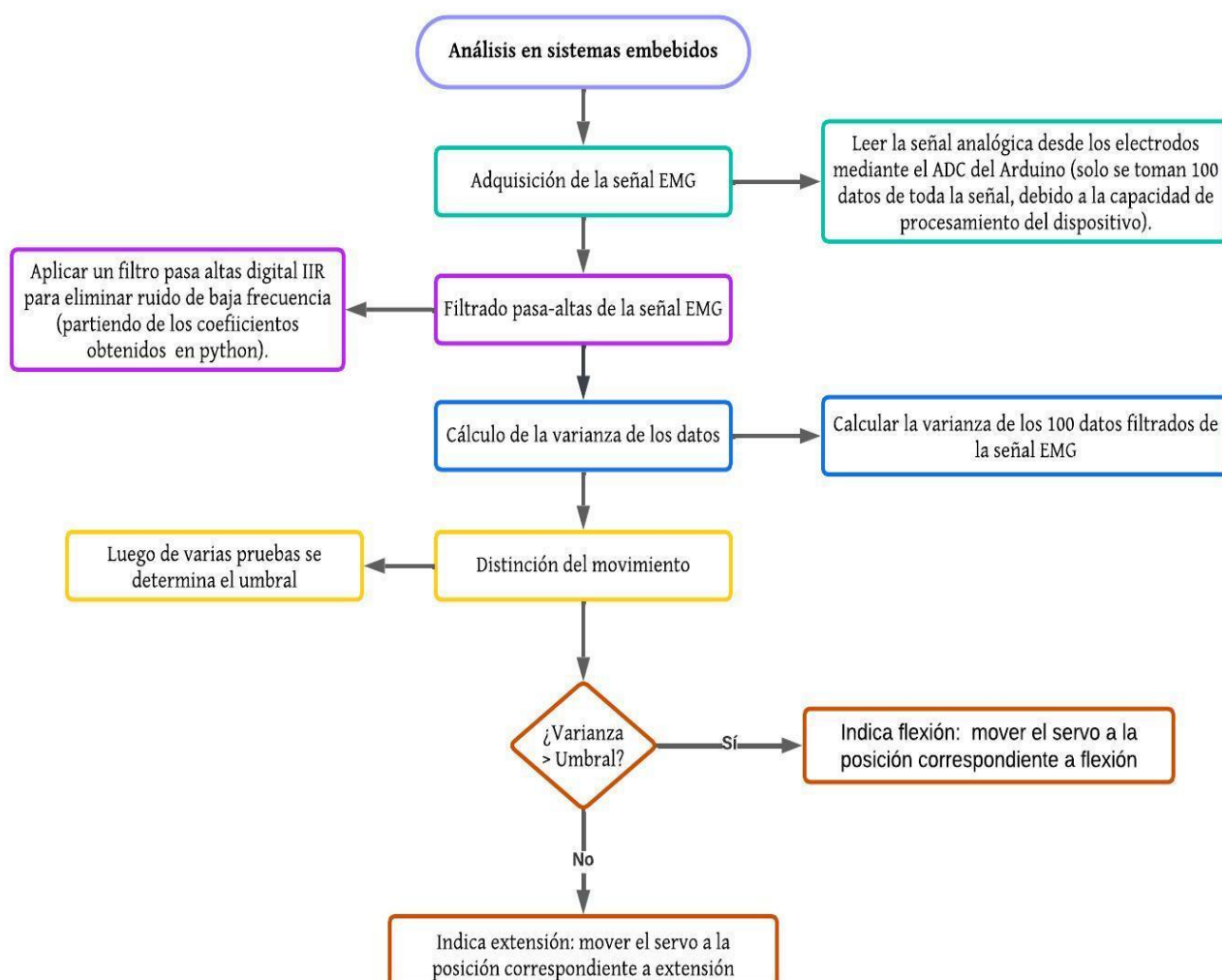
Medellín

2024

Análisis en sistemas embebidos

I. Diagrama de flujo

El siguiente diagrama ilustra el flujo de trabajo propuesto para la adquisición, filtrado, procesamiento de la señal EMG y, finalmente, la generación de la salida para indicar el movimiento de flexión o extensión. Este proceso se divide en varias etapas importantes que incluyen la captura de la señal EMG, la aplicación de filtros para eliminar el ruido no deseado, el procesamiento de la señal para extraer características relevantes y, finalmente, la determinación del tipo de movimiento muscular basado en estas características procesadas.



Justificación de cada una de las operaciones:

- **Adquisición de la señal EMG:** Es necesario obtener los datos de la señal EMG para su procesamiento posterior en el microcontrolador.
- **Filtrado pasa-altas de la señal EMG:** Dado que el Arduino ya tiene incorporado un filtro pasa-bajas de 40 Hz, es necesario aplicar un filtro pasa-altas de 10 Hz para eliminar las componentes de baja frecuencia no deseadas.
- **Cálculo de la varianza de los datos:** La varianza se utilizará como medida para determinar si hubo flexión o extensión, por lo que es necesario calcularla a partir de los datos filtrados.

- **Definición del umbral de la varianza:** Comparar la varianza con un umbral predefinido permite determinar si hubo actividad muscular significativa (flexión) o no (extensión).
- **Movimiento del servo:** Mover el servo a la posición correspondiente a flexión o extensión según el resultado de la varianza, para indicar visualmente el movimiento detectado.

Alternativas:

- **Adquisición de la señal EMG:** Adquirir un número diferente de muestras dependiendo de los requisitos.
- **Filtrado pasa-altas de la señal EMG:** Utilizar un filtro pasa-banda en lugar de un pasa-altas.
- **Cálculo de la varianza de los datos:** Utilizar otras medidas estadísticas como el valor RMS.

II. Informe de funcionamiento de la propuesta.

Actualmente para muchas aplicaciones biomédicas, es importante la detección y clasificación precisa de movimientos. Las técnicas tradicionales disponibles suelen ser costosas o requerir equipos especializados que se deben importar de otros países. Este contexto motivó la creación de un dispositivo asequible y de fácil operación que permitiera discriminar de manera efectiva los movimientos de flexión y extensión del codo mediante la medición de la contracción del bíceps de un individuo. Este dispositivo es muy útil para muchas aplicaciones, incluyendo su potencial uso en terapias de rehabilitación para personas con distrofia muscular u otras discapacidades relacionadas.

Estos movimientos se evidencian con el desplazamiento de un servomotor, lo que permite evaluar con precisión la actividad muscular en tiempo real.

A continuación, se enmarcan los resultados obtenidos en pruebas realizadas con dos sujetos diferentes, donde se llevaron a cabo un total de 20 ensayos para cada individuo. Durante estas pruebas, se registró el número de veces que el dispositivo detectó correctamente cada tipo de movimiento, permitiendo así evaluar su precisión y efectividad en situaciones reales.

Tabla 1. Resultados obtenidos

Sujeto	Umbral	Muestras Tomadas	Muestras Positivas	% Extensión	%Flexión
1	800	20	20	100%	100%
2	800	20	20	100%	100%

% de error = 0%

Los resultados obtenidos revelan un alto grado de acierto en la detección de movimientos de flexión y extensión en ambos sujetos evaluados. Este hallazgo respalda la eficacia y

confiabilidad del dispositivo, destacando su potencial para aplicaciones biomédicas, terapias de rehabilitación y otros campos donde la detección precisa de movimientos musculares sea relevante. Estos resultados alientan futuras investigaciones y desarrollos en esta línea, con el objetivo de seguir mejorando y adaptando estas tecnologías a las necesidades específicas de los usuarios y profesionales de la salud.

III. Análisis de microcontroladores

A continuación, se presenta un cuadro comparativo de varios microcontroladores disponibles en el mercado colombiano, enfocándonos en las características de memoria, velocidad de procesamiento y costo relevantes para la adquisición y procesamiento de señales EMG. Estos criterios son relevantes para discriminar con precisión entre movimientos de flexión y extensión muscular, lo que permite seleccionar el dispositivo más adecuado para el desarrollo de sistemas de detección y control de movimientos con base en señales EMG.

Tabla 2. Cuadro comparativo

Microcontrolador	Memoria Flash	Memoria RAM	Velocidad de Procesamiento	Costo Aproximado (COP)	Referencia
Arduino Uno	32 KB	2 KB	16 MHz	\$30.000 - \$90.000	[1]
Arduino Mega	256 KB	8 KB	16 MHz	\$150.000 - \$300.000	[1]
Raspberry Pi Pico	264 KB	16 KB	133 MHz	\$30.000 - \$120.000	[2]
ESP32	512 KB	520 KB	240 MHz	\$50.000 - \$70.000	[3]
STM32F103C8	64 KB	20 KB	72 MHz	\$26.000 - \$50.000	[4]
PIC18F4550	32 KB	2 KB	48 MHz	\$30.000 - \$50.000	[5]
PsoC 5LP	64 KB	8 KB	33 MHz	\$144.00 - \$200.000	[6]
Teensy 3.6	1 MB	256 KB	180 MHz	\$200.000 - \$300.000	[7]
BeagleBone Black	512 MB	512 MB	1 GHz	\$350.000 - \$450.000	[8]
ATSAM3X8E	512 KB	96 KB	84 MHz	\$60.000 - \$80.000	[9]
LPC1768	512 KB	64 KB	100 MHz	\$70.000 - \$90.000	[10]
dsPIC33FJ256GP710	256 KB	30 KB	40 MHz	\$80.000 - \$100.000	[11]
ATmega2560	256 KB	8 KB	16 MHz	\$60.000 - \$80.000	[12]
Raspberry Pi Zero W	512 MB	512 MB	1 GHz	\$150.000 - \$200.000	[13]

TM4C123GH6 PM	256 KB	32 KB	80 MHz	\$90.000 - \$120.000	[14]
------------------	--------	-------	--------	----------------------	------

Teniendo en cuenta que el objetivo principal es adquirir y procesar una señal EMG para discriminar entre un movimiento de flexión y extensión, y que el costo es un factor importante, las mejores opciones de microcontroladores comercialmente disponibles en Colombia, según el cuadro comparativo, serían:

1. ESP32:

- Costo aproximado: \$50.000 - \$70.000
- Memoria Flash: 512 KB
- Memoria RAM: 520 KB
- Velocidad de procesamiento: 240 MHz

El ESP32 ofrece una buena relación entre capacidad de procesamiento, memoria y costo accesible. Su cantidad de memoria RAM y Flash, junto con su alta velocidad de procesamiento, lo convierten en una opción muy sólida para el procesamiento de señales EMG y algoritmos de discriminación de movimientos.

2. STM32F103C8:

- Costo aproximado: \$30.000 - \$50.000
- Memoria Flash: 64 KB
- Memoria RAM: 20 KB
- Velocidad de procesamiento: 72 MHz

El STM32F103C8 es una opción económica y con una buena capacidad de procesamiento. Su memoria RAM y Flash puede ser suficiente para la aplicación de adquisición y procesamiento de señales EMG, y su velocidad de procesamiento es adecuada, aunque menor que el ESP32.

3. PIC18F4550:

- Costo aproximado: \$30.000 - \$50.000
- Memoria Flash: 32 KB
- Memoria RAM: 2 KB
- Velocidad de procesamiento: 48 MHz

El PIC18F4550 es otra opción económica, aunque su memoria RAM y Flash son más limitadas que las opciones anteriores. Sin embargo, su velocidad de procesamiento podría ser suficiente para la aplicación de adquisición y procesamiento de señales EMG.

También se tiene en cuenta que, aunque opciones como el Teensy 3.6, BeagleBone Black y Raspberry Pi Zero W ofrecen capacidades de procesamiento y memoria superiores, sus costos son más elevados y podrían no ser necesarios para esta aplicación específica.

IV. Conclusiones y recomendaciones sobre el trabajo desarrollado para futuras mejoras.

Dividiremos esta sección en dos partes, iniciaremos con las conclusiones y hallazgos durante el desarrollo de esta parte del proyecto y una segunda parte que recopilara las recomendaciones de forma autocritica para avanzar en el mismo.

IV.I Conclusiones

1. la tarjeta utilizada (microcontrolador y sensor) mediante rutinas simples de Arduino permiten discriminar entre movimientos básicos y contrarios entre si (contracción - relajación, flexión - extensión) consolidando así una opción viable y asequible para el análisis y procesamiento de señales EMG.
2. La aplicación de filtros IIR demostró ser efectiva para mejorar la precisión en la detección de movimientos, especialmente cuando se utiliza una métrica y no propiamente la señal para discriminar entre estos.
3. La integración con un servomotor u otros componentes actuadores que funcionen como indicadores (leds, buzzer , etc.) permiten una respuesta en tiempo real a los movimientos detectados lo que funciona como base para el diseño y control de otros dispositivos más complejos o elaborados.

IV.II Recomendaciones

1. Optimización de Filtros: Investigar y ajustar los coeficientes de los filtros FIR e IIR para mejorar la precisión del sistema teniendo en cuenta siempre las limitaciones de procesamiento.
2. Optimización de la rutina implementada: Implementar mejoras y optimizaciones de las rutinas para mejorar los tiempos de respuesta y la estabilidad en el funcionamiento.
3. Algoritmos de Clasificación Avanzados: Explorar algoritmos más avanzados de aprendizaje automático para una clasificación más precisa y adaptable de los movimientos.

Link del video:

<https://youtu.be/W6uuv3-VIUg?feature=shared>



Referencias:

[1] ElectroDuino, "Arduino Uno y Mega," [Online]. Available: <https://arduino.cl/producto/arduino-uno/>

[2] Electro-Lab, "Raspberry Pi Pico," [Online]. Available: <https://www.electro-lab.com.co/raspberry-pi-pico.html>.

[3] Cytron Technologies, "ESP32," [Online]. Available: <https://www.cytron.io/p-esp32-dev-board>.

[4] Alldatasheet, "STM32F103C8," [Online]. Available: <https://www.alldatasheet.es/datasheet-pdf/pdf/201590/STMICROELECTRONICS/STM32F103C8.html>

[5] Microchip, "PIC18F4550," [Online]. Available: <https://www.microchip.com/en-us/product/PIC18F4550>.

[6] Cypress Semiconductor, "PSoC 5LP," [Online]. Available: <https://www.infineon.com/cms/en/product/microcontroller/32-bit-psoc-arm-cortex-microcontroller/?term=PSoC&view=kwr&intc=searchkwr>

[7] PJRC, "Teensy 3.6," [Online]. Available: <https://electronilab.co/tienda/teensy-3-6-180-mhz-arm-cortex-m4/>

[8] BeagleBoard.org, "BeagleBone Black," [Online]. Available: <https://beagleboard.org/black>.

[9] Atmel, "ATSAM3X8E," [Online]. Available: <https://www.microchip.com/wwwproducts/en/ATSAM3X8E>.

[10] NXP, "LPC1768," [Online]. Available: <https://www.nxp.com/products/processors-and-microcontrollers/arm-microcontrollers/general-purpose-mcus/lpc1700-cortex-m3/512-kb-flash-64-kb-sram-lpc1768:LPC1768>.

[11] Microchip, "dsPIC33FJ256GP710," [Online]. Available: <https://www.microchip.com/wwwproducts/en/dsPIC33FJ256GP710>.

[12] Atmel, "ATmega2560," [Online]. Available: <https://www.microchip.com/wwwproducts/en/ATmega2560>.

[13] Raspberry Pi, "Raspberry Pi Zero W," [Online]. Available: <https://www.raspberrypi.com/products/raspberry-pi-zero-w/>.

[14] Texas Instruments, "TM4C123GH6PM," [Online]. Available: <https://www.ti.com/product/es-mx/TM4C123GH6PM>