

MATERIA DE SISTEMAS EMBEBIDOS

PROYECTO PRIMER BIMESTRE

Guerrón Carlos - Solano Edwin

15 de diciembre de 2020

1. Introducción

Se realiza un programa con el uso de Arduino uno que realiza un sistema de filtrado a una señal ECG y a una señal de Sensor Flex en el cual aplicando diferentes algoritmos de suavizado se identifica cual de estas realiza el mejor filtro para la señal de entrada, para realizar cada uno de estos algoritmos de filtrado se realiza un código en el entorno de Arduino el cual muestra como se comporta la señal con un filtro diferente. Los datos de entrada de los sensores son proporcionados de una base de datos existente el cual se analiza la

señal ECG que es una señal cardiaca y una señal diferente de un Sensor Flex con un máximo de 100 datos ya que al usar mas Arduino uno usado en la practica colapsa su memoria.

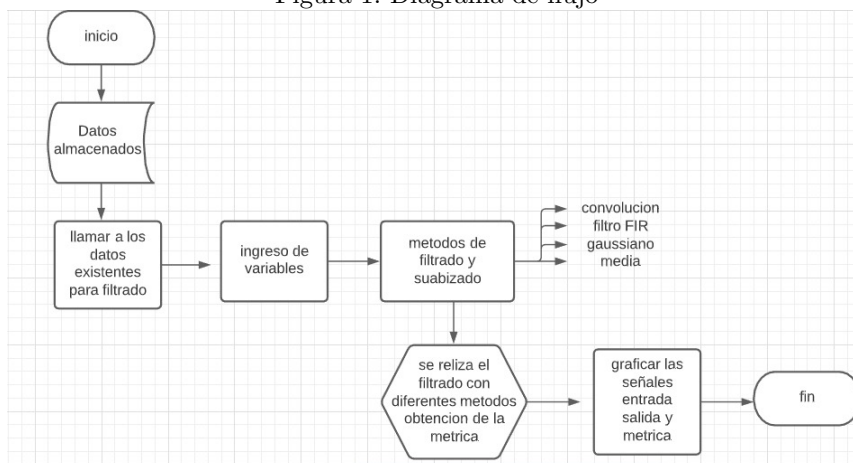
Teniendo los datos agregados al entorno de Arduino para la señal ECG se realiza un filtrado FIR que es la convolución para la señal ECG, y se obtiene las gráficas de relación señal ruido para poder analizar cual algoritmo de filtrado es el mas optimo a usar.

2. Diseño del Sistema

2.1. Diagrama de Flujo

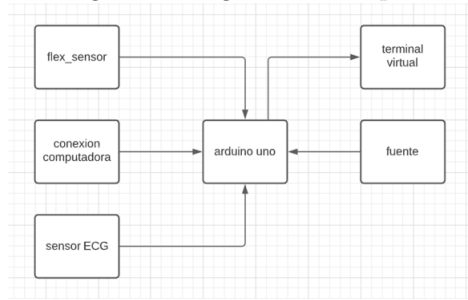
Se diseña el diagrama de flujo de cómo está constituido el programa para obtener el paso a paso del funcionamiento de este y su comportamiento al momento de implementar algoritmos de filtrado y suavizado de señal.

Figura 1: Diagrama de flujo



En este diagrama se identifica que tanto está conectado a Arduino y de relaciona que son de entrada y cuales son de salida para tener claro los componentes a usar.

Figura 2: Diagrama de bloques

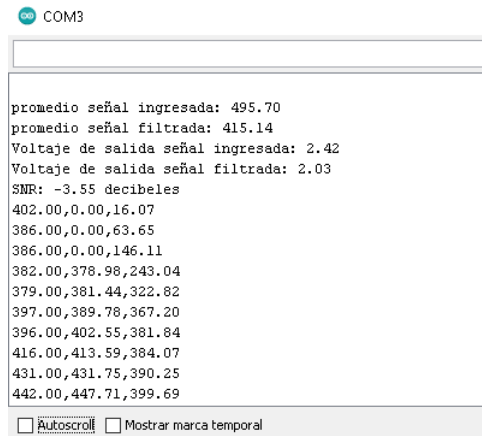


3. Desarrollo

3.1. Simulación

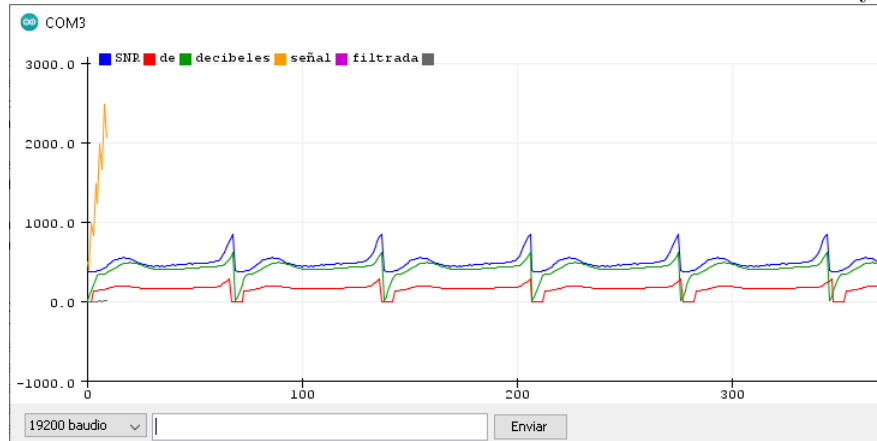
- Señal ECG con filtro FIR, y suavizado Media: Un electrocardiograma (ECG) registra las señales eléctricas del corazón. Es una prueba común e indolora que se utiliza para detectar rápidamente los problemas cardíacos y controlar la salud del corazón.

Figura 3: Datos obtenidos en monitor serial de la señal ECG con suavizado media y filtro FIR



Podemos ver los datos que nos arroja nuestra monitor serial, estos datos estan comparados con una señal impulso generada en Tfilter y una señal ECG que fue proporcionada por el docente. Se uso un Arduino UNO y por eso no se pueden usar muchos datos en la señal ya que se terminaría la memoria de este. Los resultados obtenidos muestran el promedio de la señal ingresada, el promedio de la señal filtrada, el voltaje de la señal ingresada, el promedio de la señal filtrada y el SNR.

Figura 4: Datos obtenidos en Serial Plotter de la señal ECG con suavizado media y filtro FIR



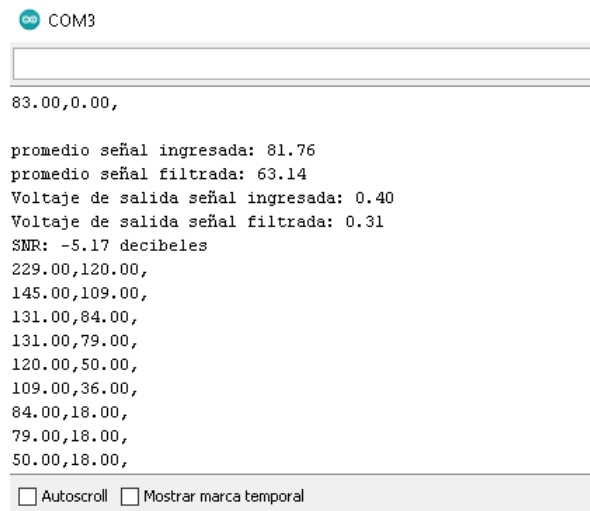
Podemos observar la señal original ingresada (azul) y la señal con el filtro y suavizada, claramente se ve que los picos se reducen y eso hace que nuestro ruido baje y sea eficiente al momento de realizar aplicaciones con estos sensores.

- Señal de un sensor flex con suavizado Gauss: El Sensor Flex (Sensor de Flexion o flex sensor) produce una resistencia variable en funcion del grado al que este doblada.

Convierte la curvatura en distintos valores de resistencia eléctrica.

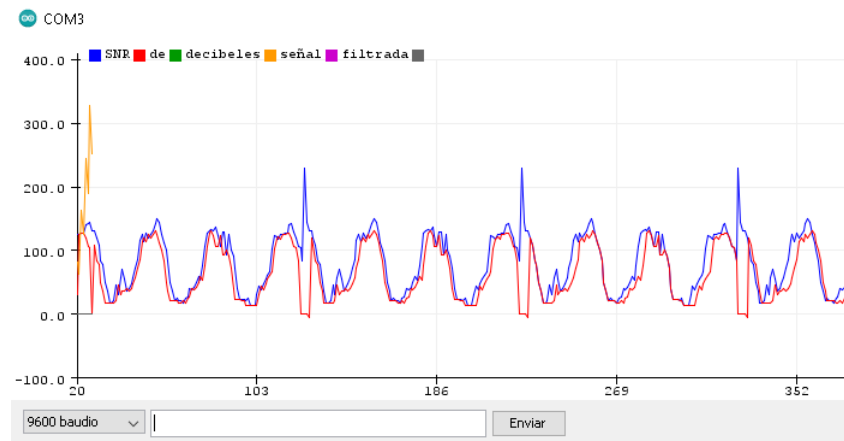
Son por lo general en la forma de una delgada tira de 5 cm de largo que varia en resistencia de aproximadamente 10 a 50 kOhms.

Figura 5: Datos obtenidos en monitor serial de la señal del sensor flex con suavizado Gauss



Aqui observamos los datos que nos arroja el monitor serial de nuestra señal de un sensor Flex, como en la anterior señal nos arroja el promedio de la señal ingresada, el promedio de la señal filtrada, el voltaje de la señal ingresada, el promedio de la señal filtrada y el SNR.

Figura 6: Datos obtenidos en Serial Plotter de la señal del sensor flex con suavizado Gauss



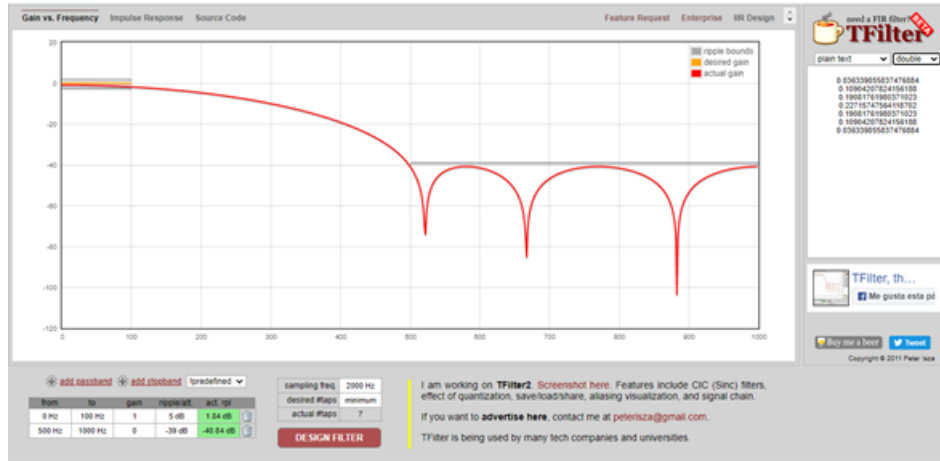
De igual manera se mira como la señal a la cual aplicamos el suavizado Gauss se puede observar una disminución de ruido y esto se refleja en los picos que son cortos y es notorio.

4. Análisis de Resultados

Link del código ubicado en el repositorio de GitHub: <https://github.com/cfguerronp/SistemasEmbebidosProyecto1B>

el filtro que se diseñó es un filtro pasa bajos en la aplicación web llamada Tfilter es una herramienta muy útil y nos da los datos de la señal de una manera muy rápida y fácil para copiar en nuestro script de arduino.

Figura 7: Tfilter



Para ver mejor los resultados que se han obtenido en el proyecto se a realizado una tabla en donde están los tipos de suavizados que se estudio y las señales que usamos, se verán los resultados de estos y la forma en que afecta el SNR.

Suavizado	Voltaje de salida de señal ingresada	Voltaje de Salida de señal filtrada	SNR	Espacio de Memoria	Señal
Media	2.42	2.03	-3.55	60 %	ECG
Gauss	0.40	0.31	-5.17	57%	Sensor Flex

Figura 8: Tabla de resultados

Como se puede observar en el suavizado Gauss sale un menor voltaje en nuestras señales, pero la diferencia entre estas no es mucha, ya que estamos comparando la señal original con la filtrada, lo que si tiene una diferencia es el SNR que el suavizado de Gauss tiene menos decibelios que el suavizado media. Y por ultimo se implemento la comparación de memoria que se uso, esto depende mas de los datos que ingresemos y como solo fue una simulación con datos recolectados de la base de datos que fue proporcionada.

5. Conclusiones y Recomendaciones

- En la SNR (relación señal ruido) mostrada en gráfica indica que mas se acerquen estos valores a cero la gráfica de filtrado es la mas óptima para su uso en diferentes aplicaciones o para el análisis de esta, estos valores se los mide en decibelios.
- Al momento de usar Arduino uno se tiene un gran problema que es el de la memoria ya que dependiendo del dato se puede usar datos entre los 70 a 100 valores dependiendo del tamaño, y las señales contenidas en la base de datos contaban con un muestreo de hasta 900 datos, por eso al momento de graficar se encuentra con ese inconveniente.
- El algoritmo gaussiano es apreciable que el filtrado de la señal es más eficiente ya que los decibelios de esta se aproximan a cero.
- Para el uso de los algoritmos de media y mediana solo se escogen un rango pequeño de valores para comparar por eso entre menos datos se coja el algoritmo funcionara menos eficientemente y la señal ira perdiendo su valor.
- Los valores que maneja Arduino son de 0 a 1023 y los que queremos medir son valores entre 0 y 5 voltios para eso se hace un mapeo a los datos y se convierte de los que mide Arduino al valor que se quiere observar.
- Para el análisis de los datos de sensores usando Arduino es muy eficiente ya que con este prototipo

pado se puede conservar una aplicación para un uso específico.

- Es necesario tener una cantidad suficiente de datos en una señal para su respectivo análisis ya que si son muy pocos los datos analizar la señal estará pobre o carente de información.
- Si se maneja una cantidad grande de datos es re-

comendable trabajar con Arduino mega ya que este tiene mas espacio de memoria y se puede analizar datos mas significativamente.

- Para llamar a una clase aparte en Arduino se la debe hacer con el comando `extend` y el nombre del método exacto si no contiene los datos o el nombre exacto dará un error de compilación