

Universidad Simón Bolívar

Departamento de Electrónica y Circuitos

Laboratorio de Proyectos 2

Enero-Marzo 2019

**Anteproyecto:**

**Implementación de acelerómetros y DEMOQE128 para controlar un juego simple**

**Integrantes:**

Gianluque De Simone 12-11412

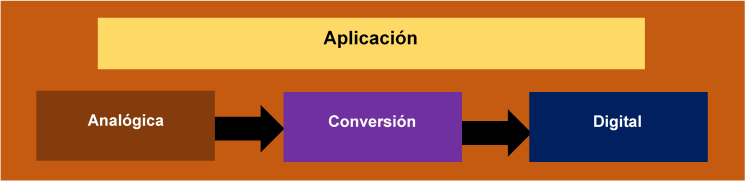
Marco Arroyo 08-10073

# Introducción

El objetivo de este proyecto es utilizar la experiencia adquirida durante la primera parte de la materia para diseñar e implementar la adquisiciónd e datos analógicos y utilizarlos como control de una aplicación digital sencilla.

El prototipo de diseño consta de una parte de adquisición analógica, una parte de conversión analogico-digital y filtrado antialising y una parte de aplicación digital como se puede apreciar en la “Figura 1: Esquema general”.

### Figura 1: Esquema general



# Parte Analógica: Adquisición-Acondicionamiento

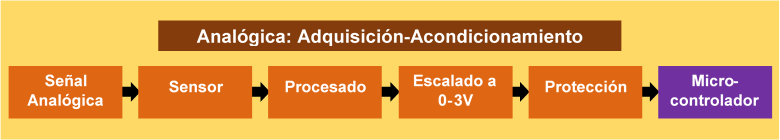
La primera etapa del diseño consta de un sistema de adquisición de señales analógico.

Como se puede apreciar en la “Figura 2: Esquema Analógico” la señal primero debe ser leida por un sensor que convertirá aceleración a voltaje.

Luego de pasar a un procesado con protecciones para entrar montaje y ser escalada a un rango de 0-3V que es lo necesario para que el microprocesador la pueda aceptar.

Finalmente la señal debe pasar por una protección extra que garantice que se mantiene en el rango de voltaje aceptable antes de ser enviada al microcontrolador.

### Figura 2: Esquema Analógico



# Parte de Conversión: Microcontrolador DEMOQE128

Una vez que la señal ha sido acondicionada para un rango adecuado a la entrada del microprocesador DEMOQE128, este debe tomar las muestras necesarias a una frecuencia del doble de la deseada. Esto es 2kHz. La muestra es retenida en una posición de memoria para su lectura y el micro la revisa para pasarla por su conversor analógico digital de aproximaciones sucesivas a 12b.

Es un requisito de diseño que se aplique a la señal un Filtro digital. Tentativamente se escogio el filtro de tipo FIR o Finite Impulse Response. Este filtro tiene una respuesta finita al impulso y dará preferencia a cierta parte de la banda de frecuencias.

Además su diseño tiende a ser más sencillo y se encontraron multiples ejemplos de su implementación disponibles en internet.

Por requisito de diseño el filtro debe poder desactivarse y reactivarse a preferencia al presionar un boton en el microcontrolador. Esto será realizado mediante interrupciones.

En la “Figura 4: Diagrama para filtro FIR tomado de Wikipedia”[1], se puede apreciar la estructura básica para un filtro FIR genérico.

### Figura 4: Diagrama para filtro FIR tomado de Wikipedia

Como se puede ver el filtro FIR funciona mediante una serie de retardos en cascada que se pueden acumular según haga falta. Estos filtros tienen la ventaja de ser estables a diferencia de los filtros IIR[2]. La desventaja es que tienden a ser mas lentos y esto se agrava entre mas retardos se utilicen.

La implementación del filtro digital deberá realizarce en lenguaje C para poder correrla con el microprocesador. El resultado esperado es una sumatoria de la señal y cierto número por definir de sus retardos en una posición de memoria específica que se pueda escribir o leer según haga falta para enviar la señal al computador.

En la “Figura 3: Conversión y Preprocesamiento” se puede apreciar un esquema general de las etapas a realizar en el microprocesador.

### Figura 3: Conversión y Preprocesamiento



Finalmente, tras filtrar un poco mas la señal esta se envía por un bus a la computadora siguiendo un protocolo de comunicaciones RS232 (tentativo) para que la señal sea utilizada y mostrada en la interfaz para los usuarios.

Protocolo de comunicaciones al computador: En principio se propone usar el protocolo estándar RS 232 para nueve conectores, el cual sugiere las conexiones que se pueen apreciar en la “Tabla 2: Protocolo RS 232”

### Tabla 2: Protocolo RS 232

|  |  |  |
| --- | --- | --- |
| **Señal** | **Notación** | **Pines** |
| Common Ground | G | 5 |
| Transmitted Data | TD | 3 |
| Received Data | RD | 2 |
| Data Terminal Ready | DTR | 4 |
| Data Set Ready | DSR | 6 |
| Request To Send | RTS | 7 |
| Clear To Send | CTS | 8 |
| Carrier Detect | DCD | 1 |
| Ring Indicator | RI | 9 |

# Parte Digital: Interfaz para usuario

La parte final del proyecto es el uso de la señal adquirida para controlar una aplicación digital. Tentativamente la aplocicación será un juego simple diseñado en python basado en las mecánicas básicas de “Flappy Bird”[3].

Para ello la señal digital llegará por puerto serial al Bus COM8 (que es el disponible en la computadora utilizada). De ahí la tomará la aplicación y será procesada para poder utilizarla.

En la “Figura 4: Digital: Aplicación” se puede ver el diagrama de etapas de esta parte final del diseño.

### “Figura 4: Digital: Aplicación”



# Bibliografía

[1] “FIR (Finite Impulse Response)”, Wikipedia, 2019. [En linea]. Disponible en: <https://es.wikipedia.org/wiki/FIR_(Finite_Impulse_Response)>. [Accedido: 8-May-2019]

[2] “IIR”, Wikipedia, 2019. [En linea]. Disponible en: <https://es.wikipedia.org/wiki/IIR>. [Accedido: 8-May-2019]

[3] “IIR”, Wikipedia, 2019. [En linea]. Disponible en: https://es.wikipedia.org/wiki/Flappy\_Bird.