

**PROYECTO FIN DE CARRERA**  
**INGENIERÍA DE TELECOMUNICACIONES**



**SISTEMA DE MONITORIZACIÓN PARA EL  
APOYO AL TRATAMIENTO DEL TRASTORNO  
POR DÉFICIT DE ATENCIÓN CON  
HIPERACTIVIDAD**

AUTOR: LUCAS MULEY VILAMÚ  
DIRECTOR: RUBÉN BLASCO MARÍN  
PONENTE: ROBERTO CASAS NEBRA  
DEPARTAMENTO DE INGENIERÍA  
ELECTRÓNICA Y COMUNICACIONES  
CENTRO POLITÉCNICO SUPERIOR

FEBRERO 2011



## Sistema de monitorización para el apoyo al tratamiento del trastorno por déficit de atención con hiperactividad

### RESUMEN

El presente proyecto fin de carrera ha desarrollado un sistema de bio-feedback basado en sensores inerciales, con el fin de ayudar a niños con trastorno por déficit de atención con hiperactividad en la mejora de su autocontrol. Este sistema detecta los estados de actividad del niño con el objeto de que él sea consciente de que su estado de nerviosismo/ansiedad está aumentando y sea capaz de controlarlo.

El sistema de bio-feedback consiste en uno o más sensores inerciales que porta el usuario del sistema, estos sensores se comunican inalámbricamente con un PC, donde se realiza un análisis en tiempo real de las señales recibidas y se genera un estímulo visual y/o auditivo.

Para la realización de este sistema se ha utilizado el kit comercial de sensores inalámbricos "EcoKit" (1).

Las principales tareas realizadas han sido las siguientes:

- Programación del Firmware de los sensores inalámbricos.
- Programación del Firmware de la Estación Base.
- Programación del software y algoritmos del PC que procesará los datos.
- Validación del sistema.



## Índice

RESUMEN .....	3
Índice.....	5
Capítulo 1 Introducción.....	9
1.1 Introducción.....	9
1.2 Marco del desarrollo del proyecto.....	9
1.3 Objetivos y alcance del proyecto .....	9
1.5 Estructura de la memoria .....	10
1.6 Distribución temporal del desarrollo del proyecto.....	11
Capítulo 2 Arquitectura del sistema y definición de requisitos.....	13
2.1 Introducción.....	13
2.2 Especificaciones técnicas.....	13
2.2.1 Especificaciones en el diseño del firmware de los ECO Nodes y la Estación Base .....	13
2.2.2 Especificaciones en el diseño del software del PC.....	13
2.2 Arquitectura del sistema .....	14
2.2.1 Descripción del entorno de uso .....	14
2.2.2 Descripción del sistema .....	15
2.3 Implementación de la arquitectura.....	15
2.4 Tareas desarrolladas.....	17
Capítulo 3 Sistema de captación de movimiento .....	19
3.1 Introducción.....	19
3.2 ECO nodes.....	19
3.2.1 Descripción física.....	19
3.2.2 Programación del Firmware.....	21
3.3 Estación Base .....	23
3.3.1 Descripción física.....	23
3.3.2 Programación del firmware .....	24
3.4 Comunicación entre los ECO nodes y la Estación Base .....	26
3.4.1 Introducción .....	26
3.4.2 Protocolo de comunicación .....	27
Capítulo 4 Análisis del grado de actividad y bio-feedback.....	31
4.1 Introducción.....	31
4.2 Script en Matlab para el estudio de errores .....	31

4.3 Script en Matlab para la toma de datos de usuarios.....	32
4.3.1 Objetivo.....	32
4.3.2 Diagrama de bloques.....	32
4.4 Desarrollo y prueba de distintos algoritmos para el análisis del grado de actividad.....	33
4.4.1 Objetivo.....	33
4.4.2 Diagrama de flujo.....	33
4.5.3 Validación con base de datos.....	39
4.6 Desarrollo software del sistema de bio-feedback.....	41
4.6.1 Objetivo.....	41
4.6.2 Diagrama de flujo.....	42
4.6.4 Menú de configuración.....	44
Capítulo 5 Conclusiones y perspectivas.....	47
Bibliografía.....	49
Figuras.....	51
Gráficos.....	52
Tablas.....	52
Anexo 1 Descripción del estado del arte.....	55
Dispositivos de bio-feedback para hiperactivos.....	55
Dispositivos comerciales de detección de movimiento.....	55
Xsense.....	55
InterSense.....	55
Ascension.....	56
Microstrain.....	56
Nec-tokin.....	56
Kits de desarrollo de bajo coste con RF y sensores inerciales.....	57
eZ430-Chronos.....	57
G-Link®mXRS™Wireless Accelerometer Node.....	57
Comunicaciones.....	57
Zigbee.....	57
Bluetooth.....	58
Wi-Fi.....	58
RF del ECO Kit.....	59
Anexo 2 Puesta en marcha del ECO Kit.....	61
Introducción.....	61

Puesta en marcha paso a paso .....	63
El ECO Node .....	63
Development Board .....	64
Anexo 3 Firmware del ECO Node.....	67
Librerías, variables globales y funciones .....	67
Inicialización de variables y configuraciones iniciales.....	69
Bucle principal .....	71
Configuración del Timer 0 .....	74
Anexo 4 Firmware de la Estación Base. ....	79
Anexo 5 Scripts en Matlab y software del PC.....	83
Script para la toma de datos de usuarios .....	83
Script en Matlab para validar los detectores de actividad .....	90
Script en Matlab del software final.....	107
Anexo 6 Estudio de errores de los ECO Nodes .....	127
Transmisión de 1 ECO Node .....	127
Transmisión 2 ECO Nodes a 40 Hz .....	129
Transmisión 3 ECO Nodes a 40 Hz .....	135
Anexo 7 Toma de datos.....	137
Introducción .....	137
Objetivos .....	138
Metodología .....	138
Análisis de resultados y cálculo de los umbrales .....	139
Anexo 8 Predicción y reconstrucción de la señal.....	147
Interpolación lineal .....	147
Filtrado adaptativo.....	148
Conclusiones.....	150





## Capítulo 1 Introducción

### 1.1 Introducción

El presente proyecto fin de carrera (PFC) desarrolla un sistema de bio-feedback, basado en sensores inerciales, con el fin de ayudar a niños con Trastorno por Déficit de Atención con Hiperactividad (TDAH) en la mejora de su autocontrol. Este sistema detectará los estados de actividad del niño con el objeto de que él sea consciente de que su estado de nerviosismo/ansiedad está aumentado y sea capaz de controlarlo.

El sistema de bio-feedback consistirá en dos sensores inerciales que deberá portar el usuario del sistema. Estos sensores se comunicarán inalámbricamente con un PC, donde se realizará un análisis en tiempo real de las señales recibidas y se generará un estímulo visual y/o auditivo.

### 1.2 Marco del desarrollo del proyecto

El PFC se ha desarrollado en el seno del grupo de investigación TECNODISCAP (2), de la Universidad de Zaragoza. TECNODISCAP persigue la mejora de la calidad de vida de las personas con discapacidad, dependencia o necesidades especiales, mediante servicios basados en la aplicación de las Tecnologías de la Información y Comunicaciones (TIC).

Con la finalidad de conseguir un mayor impacto en su entorno, TECNODISCAP colabora con varias instituciones y asociaciones aragonesas (DGA, Ayuntamiento de Zaragoza, ATADES, DFA, ATENCIONA, etc). En concreto, fruto de la colaboración existente con ATENCIONA (3), se plantea el presente PFC. ATENCIONA es una asociación sin ánimo de lucro formada por un grupo de familias con hijos diagnosticados de TDAH y profesionales implicados en el trastorno. Esta institución busca la integración y la adaptación de los niños en la sociedad. Trabajan y colaboran con colegios, asociaciones, sistema sanitario, educativo, etc. para ayudar a los niños con TDAH a eliminar las barreras con las que se encuentran a diario en su entorno como consecuencia de sus dificultades. ATENCIONA, como objetivo principal persigue *“poner todos los medios a su alcance para poder llegar a la detección temprana de los niños con TDAH, procurándoles tratamiento y seguimiento adecuado a sus necesidades para su correcto desarrollo”* (4).

Dentro de las líneas de colaboración establecidas con ATENCIONA se incluye el bio-feedback, con la intención de mejorar el auto-control de los niños con TDAH, siendo este PFC el primer paso en este sentido.

### 1.3 Objetivos y alcance del proyecto

El proyecto persigue conseguir una herramienta para mejorar el tratamiento de niños con TDAH. Esta herramienta está pensada para ser usada en clases de mejora del autocontrol de la asociación ATENCIONA. Este PFC pretende ser una primera prueba experimental, dando un prototipo que permita validar el concepto.

Posteriormente será analizado por psicólogos de la asociación que evaluarán su eficacia en los niños con TDAH proponiendo mejoras y cambios. Una vez aprobado el sistema por la asociación, se dará paso a la implementación definitiva para su posterior explotación.

Los objetivos que se han marcado con este proyecto son los siguientes:

1. Puesta a punto del ECO Kit y de su API.
2. Programación del Firmware del ECO Node y de la Estación Base, para la transmisión de los datos obtenidos en el ECO Node al ordenador.
3. Diseño de un script en Matlab que permita capturar los datos de los sensores y registrarlos.
4. Diseño de un script en Matlab que permita realizar un post-proceso de los datos capturados con el fin de detectar distintos grados de actividad de la persona que porta el sensor.
5. Realización de un script en Matlab que analice en tiempo real los datos procedentes de los sensores y genere un estímulo visual y auditivo según programación del terapeuta.
6. Validación del sistema con usuarios.

### 1.5 Estructura de la memoria

En este apartado se va a explicar cómo se ha estructurado la memoria dando una pequeña introducción de lo que se puede encontrar en cada uno de los capítulos y anexos.

La memoria dispone de 5 capítulos:

- El capítulo 1 da una visión general del proyecto situándolo en el contexto y exponiendo lo que en él se va a desarrollar, así como sus objetivos y alcance. También se muestra una distribución temporal del mismo, mostrando así la carga de trabajo asociada.
- En el capítulo 2 describe las especificaciones del sistema y como se ha implementado la arquitectura del sistema acorde a esas especificaciones.
- El tercer capítulo describe el sistema de captación de movimiento. Se muestra la arquitectura y la programación del firmware de los sensores inerciales y la Estación Base.
- El capítulo 4 detalla el análisis de los datos proporcionados por los sensores inerciales, explicando el desarrollo del software de la aplicación final.
- Por último, en el capítulo 5 se hace una valoración global de todo el trabajo realizado. Se sacarán las conclusiones finales obtenidas y las perspectivas de continuidad del PFC.

Hay un total de 8 anexos:

- ❖ El primer anexo da una descripción del estado del arte.
- ❖ El segundo anexo describe las características del EcoKit y su puesta en marcha.
- ❖ En el anexo 3 tenemos una explicación del firmware de los ECO Nodes.
- ❖ El cuarto anexo es el firmware de la Estación Base.
- ❖ El anexo 5 son los scripts en Matlab de los algoritmos y del software.
- ❖ El anexo 6 describe un estudio realizado de los errores de transmisión nodo-Estación Base.
- ❖ En el anexo 7 se detallará las pruebas de campo realizadas para la validación de los algoritmos de detección.

- ❖ El anexo 8 detalla dos algoritmos utilizados para la reconstrucción de la señal. Estos dos algoritmos no se han implementado finalmente.

### 1.6 Distribución temporal del desarrollo del proyecto

A continuación se puede ver el diagrama de Gantt de la distribución temporal del desarrollo del proyecto. El número estimado de horas dedicadas al proyecto ha sido 700 horas.

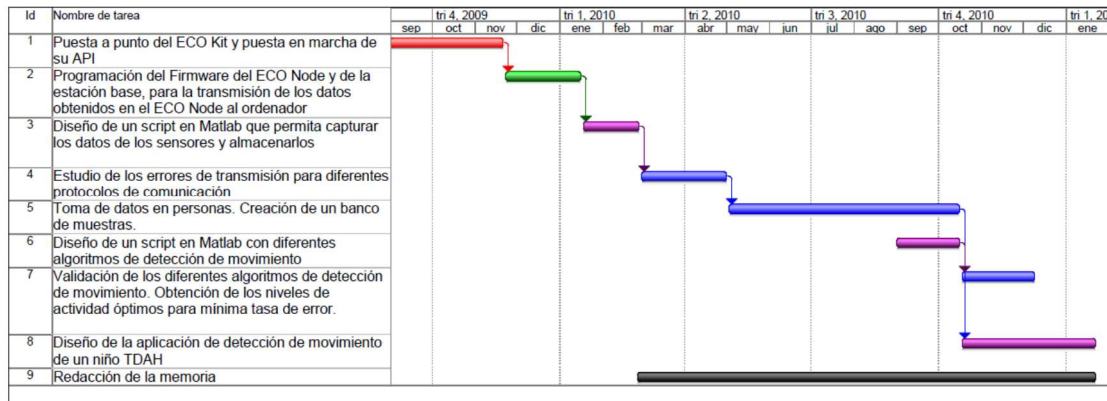


Figura 1 Diagrama de Gantt del desarrollo del proyecto

- █ Puesta a punto del ECO Kit
- █ Programación del Firmware
- █ Diseño de scripts en Matlab
- █ Estudio
- █ Redacción de la memoria



## Capítulo 2 Arquitectura del sistema y definición de requisitos

### 2.1 Introducción

A continuación se describe las especificaciones técnicas y cómo se ha implementado la arquitectura del sistema de detección del grado de actividad.

### 2.2 Especificaciones técnicas

Para realizar el sistema de bio-feedback va a ser necesario contar con sensores inerciales que muestreen el movimiento del niño con TDAH. Deben ser de pequeño tamaño e inalámbricos ya que el uso de cables aumenta la intrusión. La información de los acelerómetros será enviada por RF a una Estación Base que reenviará los datos al PC.

Los sensores inerciales deben tener la capacidad de enviar la información a una distancia de al menos 3 metros de distancia (relacionada con el tamaño de las aulas donde se usará el sistema). Deben ser reprogramables para poder modificar la frecuencia de muestreo y el protocolo de comunicación en función de las necesidades que tengamos. Una de las especificaciones técnicas predeterminadas del PFC ha sido el uso del “Eco Kit” (1). El Eco Kit es un kit de desarrollo fabricado en el seno de la National Tsing Hua University de Taiwán y la Universidad de California. Este kit dispone de sensores de movimiento inerciales (llamados ECO Nodes) con capacidad para enviar información de forma inalámbrica. Además, incorpora dentro del kit el DEMO9S12NE64 de Freescale (5), una Estación Base que hará de puente entre los nodos y el PC.

Las especificaciones de diseño han venido marcadas por tres factores: el ámbito en el que el sistema se va a utilizar (clase para la mejora del autocontrol en niños TDAH de la asociación Atenciona), las limitaciones del hardware (la falta de un protocolo de acceso al medio en los nodos) y el planteamiento inicial del proyecto final de carrera (por ejemplo, dentro de las especificaciones iniciales está la de usar el “EcoKit”).

#### 2.2.1 Especificaciones en el diseño del firmware de los ECO Nodes y la Estación Base

Se necesita que los ECO Nodes sean capaces de:

1. Muestrear los valores de la aceleración en los tres ejes del espacio para aceleraciones normales del cuerpo humano.
2. Enviar las muestras a una distancia de 3 metros (las clases donde se va a utilizar el sistema será de 10 m<sup>2</sup>) por RF.

La Estación Base debe ser capaz de:

1. Recibir datos por RF de los nodos, y comunicarse con ellos.
2. Reenviar al PC los datos recibidos de los nodos.

#### 2.2.2 Especificaciones en el diseño del software del PC

Durante todo el desarrollo del PFC se han programado diversos scripts en Matlab. Cada uno de los scripts ha tenido que realizar tareas diferentes. A continuación se describe brevemente la utilidad de cada uno de los scripts y sus especificaciones:

1. Script en Matlab para el estudio de errores.

Este script obtendrá la tasa de error en la comunicación con el fin de mejorar el protocolo de comunicación.

- a. Implementación de una rutina de interrupción para capturar bytes del puerto serie.
  - b. Procesamiento de los bytes y detección de muestras.
  - c. Cálculo de la tasa de error en recepción en tiempo real.
2. Script en Matlab para la toma de datos.

La función principal de este script es crear una base de datos de muestras para validar posteriormente los algoritmos de detección de actividad. La especificación principal es:

- a. Almacenaje de muestras en una base de datos diferenciando el número de persona, el nivel de actividad, y la fecha y hora de la captura de los datos.
3. Script en Matlab para validar distintos algoritmos.

El objetivo de este script ha sido probar y validar los algoritmos de detección:

- a. Adquisición de muestras de una base de datos almacenadas por el script anterior.
  - b. Filtrado paso bajo de la señal.
  - c. Algoritmos de detección del grado de actividad para cada nodo.
  - d. Cálculo de la probabilidad de error en función de los 3 umbrales de actividad de cada nodo.
4. Script en Matlab de la aplicación final.
- a. Comunicación bidireccional entre nodos-Estación Base-PC.
  - b. Implementación de una rutina de interrupción para capturar bytes del puerto serie.
  - c. Procesamiento de los bytes y detección de muestras.
  - d. Filtrado paso bajo de la señal.
  - e. Algoritmos de detección del grado de actividad de cada nodo.
  - f. Algoritmo de detección del grado de actividad final del usuario en función del nivel de actividad de cada nodo. Este algoritmo debe usar el grado de actividad de cada nodo, y el tiempo que lleva cada nodo en ese grado de actividad para determinar la actividad del usuario portador de los sensores.
  - g. Menú de opciones para el control del sistema.
  - h. Bio-feedback visual y auditivo.

## 2.2 Arquitectura del sistema

### 2.2.1 Descripción del entorno de uso

El ámbito en el que se va a usar este sistema de detección de actividad va a ser las clases de la Asociación Atenciona. Como ya se ha descrito anteriormente, este prototipo solo nos va a servir para validar el sistema. Por tanto el ámbito de actuación va a ser reducido en comparación al sistema final. Concretamente, el sistema se va a usar en las clases de mejora del autocontrol que tienen un tamaño de 10m<sup>2</sup>. Durante la clase, el usuario que lleve puesto el/los nodos estará realizando una tarea solo o junto con algunos otros usuarios. El sistema debe detectar el grado de actividad del usuario dando un estímulo visual y auditivo que le permita ser consciente de su actividad y lo reduzca dinámicamente. Esto ayudará a los psicólogos de la asociación a enseñar a los niños con TDAH a auto-controlarse. Los nodos que realizan el muestreo de la aceleración deberán estar en diferentes partes del cuerpo del niño

que no impidan la realización normal de las tareas ni tampoco puedan ser una distracción para él. Buscamos un sistema no invasivo de detección del movimiento que pase desapercibido para el usuario.

### 2.2.2 Descripción del sistema

El sistema está compuesto de tres módulos:

1. El primer módulo son los ECO Nodes. Estos nodos estarán colocados en diferentes partes del cuerpo del usuario (en la aplicación final sería el niño TDAH). Son los encargados de enviar la información de la aceleración muestreada a 40 Hz hasta la Estación Base. Transmiten esta información por RF a la frecuencia de 2,4 GHz. Dado que el sistema tiene que ser poco invasivo se ha optado por poner sobre el usuario solo 2 nodos (suficiente para la aplicación que queremos). Los algoritmos que se han desarrollado se han validado para poder funcionar con uno o dos nodos.
2. El segundo módulo es la Estación Base. Es la responsable de recibir los datos por RF de los nodos. La antena de la Estación Base debe estar colocada en un sitio lo más visible para los nodos, de esta forma se evitan desvanecimientos de señal en la comunicación nodos-base. La Estación Base se comunicará a través su puerto serie con el PC.
3. El tercer módulo será el PC y el feedback audiovisual. Será el encargado de procesar los datos en tiempo real detectando el grado de actividad del usuario. Generará el feedback visual y auditivo.

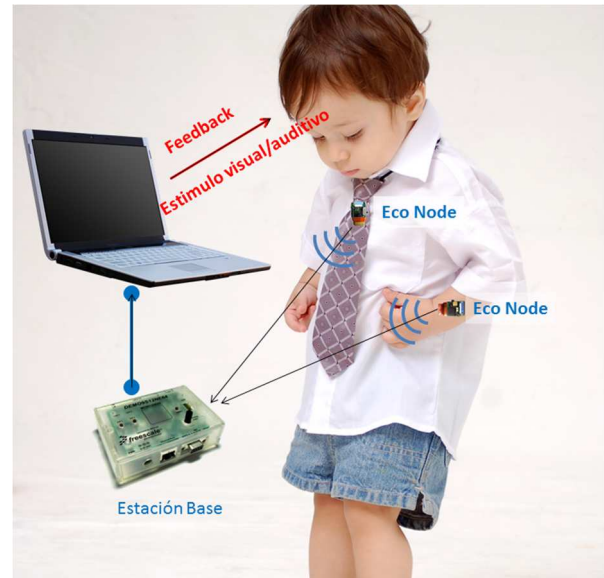


Figura 2 Croquis del sistema de bio-feedback

### 2.3 Implementación de la arquitectura

En este apartado se va a describir desde un punto de vista lógico como se ha implementado la arquitectura en los ECO Nodes, Estación Base, y PC.

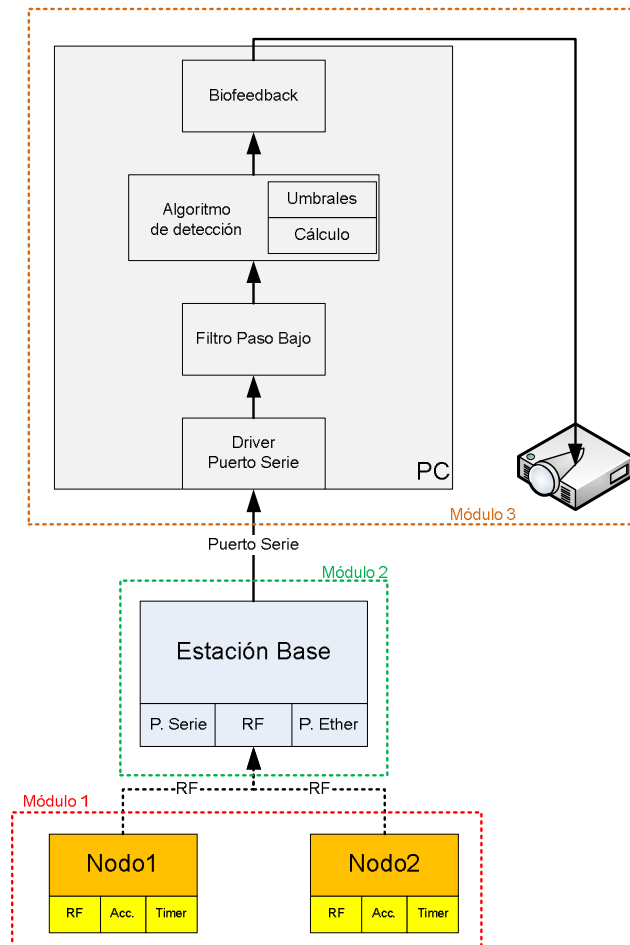


Figura 3 Esquema lógico del sistema

Como se puede ver en la figura 3, tenemos los 3 módulos que ya se han descrito en el apartado anterior.

El módulo 1, que corresponde con los ECO Nodes, se encargará de muestrear la aceleración del movimiento cada 25 ms (inverso de la frecuencia de muestreo) para posteriormente enviarla a la Estación Base. Dispone de tres bloques: El bloque “Timer” es el responsable de contar 25 ms y dar la orden de muestrear. El bloque “Acc.” muestreará la señal de sensor inercial con un ADC (**A**nalog-**t**o-**D**igital **C**onverter). Por último el bloque “RF” es el responsable de enviar las muestras a la Estación Base.

El módulo 2 corresponde con la Estación Base. Es la responsable de hacer de puente entre los nodos y el PC que procesará los datos. La base dispone de diferentes bloques correspondientes con los 3 puertos que hemos usado sistema lo largo del proyecto final de carrera. El boque “P.Serie” es el puerto serie por el que mandaremos al PC las muestras que recibamos de los nodos. El bloque RF recibirá las muestras de los nodos a través de una antena exterior. El bloque “P. Ether” nos servirá para transmitir datos de la Estación Base al PC. Este bloque no se usa en el programa final.

El tercer módulo, que corresponde con el PC, será el encargado de procesar los datos recibidos de la Estación Base y detectar el estado de actividad del usuario. Este módulo tiene 4 bloques principales. El primero de ellos es el driver de recepción del puerto serie. Recibirá los datos por



el puerto serie y verificará que son datos de aceleración correctos. El siguiente bloque realiza un filtrado paso bajo de las muestras. Una vez filtrados los datos, están listos para ser procesados en el bloque “Algoritmo de detección”. En este bloque se realizan dos operaciones. La primera es el cálculo de la actividad con uno de los dos algoritmos de detección disponibles. Esta primera operación proporcionará un nivel de actividad para cada uno de los nodos activos.

La segunda operación que realizará el bloque “Algoritmo de detección” será decidir en que estado se encuentra el usuario en función del grado de actividad de cada nodo. Comparará la actividad de un nodo con unos umbrales. De esa comparación proporcionará uno de los 4 estados de actividad en la que puede estar el nodo: actividad nula, baja, media o alta. Una vez obtenida la actividad de cada nodo, concluirá el nivel de actividad del

El grado de actividad del usuario se enviará al bloque “Bio-feedback”, que será el encargado de generar un estímulo visual-auditivo para que el usuario se dé cuenta que ha aumentado su estado de actividad y que por tanto debe moderar sus movimientos.

## 2.4 Tareas desarrolladas

Para la realización del PFC se ha tenido que realizar una serie de tareas de forma ordenada. A continuación se expone brevemente cada una de las tareas y se engloban en el conjunto del proyecto:

- 1. Programación de los ECO Nodes** (programación del módulo 1): se ha programado el muestreo de los acelerómetros, comunicación por RF, etc.  
Los ECO Nodes se programarán para que realicen un streaming de datos de las aceleraciones y los envíen a la Estación Base.
- 2. Programación de la Estación Base:** se ha programado el envío y recepción por RF y por el Puerto Serie.  
La base hará de puente de comunicación entre los nodos y el PC.
- 3. Verificación de la comunicación entre la Estación Base y los ECO Nodes.**  
En este apartado se ha verificado la comunicación bidireccional de varios nodos con la Estación Base.
- 4. Programación de un script en Matlab para el estudio de errores.**  
En este apartado se ha programado un script en Matlab para recibir los datos de la Estación Base y analizar los errores de la comunicación nodo-base-PC. Se ha realizado un estudio completo de los errores en función de varias variables: distancia entre nodo-base, número de nodos transmitiendo a la vez, uso de diferentes canales de RF, diferentes controles de acceso al medio, etc. Este estudio está detallado en el Anexo 6. Con las conclusiones de este estudio, se han establecido el protocolo de comunicación idóneo para la comunicación.
- 5. Reprogramación de los ECO Nodes y la Estación Base.**  
Una vez definido el protocolo óptimo de comunicación entre la Estación Base y los nodos se han programado los nodos y la base.

Una vez implementado un protocolo de comunicación nodo-base-PC que minimice la tasa de error, se ha procedido a programar el detector de actividad. Este detector debe recibir en tiempo real las muestras de aceleración de los nodos que lleva el usuario, y según esos datos, debe detectar su grado de actividad. Para validar los algoritmos que detectarán el estado de actividad del usuario (que será un niño TDAH), se ha necesitado obtener datos reales de la aceleración en distintos grados de actividad de un niño TDAH. Al estar protegidos los menores de edad por la ley, se ha tenido que crear una base de datos de muestras de aceleración de personas que simularan distintos grados de actividad de un niño TDAH. Los pasos realizados han sido:

**6. Programación de un script en Matlab para la toma de datos.**

Se ha diseñado un script en Matlab para tomar muestras de usuario. Estas muestras servirán para validar los sistemas de detección de actividad.

**7. Programación de un script en Matlab para validar los detectores de actividad.**

Una vez se ha obtenido un banco de muestras de personas en diferentes estados de actividad, se ha procedido a programar un script en Matlab capaz de adquirir datos del banco de muestras y procesarlos para probar los distintos algoritmos de detección de actividad.

Por último se ha procedido a la programación del software final:

**8. Programación de una aplicación en Matlab con biofeedback.**

Se ha procedido a crear la aplicación final. Se ha programado un menú para seleccionar el número de nodos, el sistema de detección de movimiento, y variar los umbrales de actividad. Desde este menú se controla el bio-feedback audiovisual, que será el responsable de avisar al usuario que su estado de actividad ha aumentado (o disminuido).

## Capítulo 3 Sistema de captación de movimiento

### 3.1 Introducción

En este capítulo se explica cómo se ha desarrollado el sistema de captación de movimiento. Se hará una descripción física de los nodos y la Estación Base junto con una explicación de cómo se han programado los firmware. Por último, se abordará la comunicación nodo-base.

### 3.2 ECO nodes

Los ECO Nodes son los nodos inalámbricos que se van a utilizar para muestrear el movimiento inercial del usuario. Tienen unas características que los hacen adecuados para la aplicación que queremos:

1. Ocupan un volumen de  $1\text{cm}^3$  y pesan 2 gramos incluyendo batería y antena.
2. Disponen de un dispositivo RF con una tasa de transmisión de 1 Mbps y un alcance de 10 metros.
3. Duración de la batería mayor a 4 horas (más de 1 hora en transmisión continua).
4. Dispone de un acelerómetro.

#### 3.2.1 Descripción física

El hardware que dispone el ECO Node es el que se puede ver en la siguiente figura:

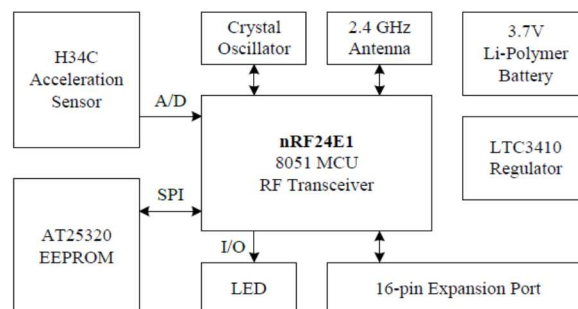


Figura 4 Diagrama de bloques del hardware del ECO Node

Cada uno de los ECO Nodes dispone de las siguientes partes:

- Microcontrolador nRF24E1 (Nordic Semiconductor)

Este microcontrolador dispone de una unidad de RF que funciona a 2,4 GHz y un convertidor analógico-digital que es el que usaremos para obtener muestras de los acelerómetros en determinados instantes de tiempo.

La arquitectura interna del microprocesador se puede ver a continuación (6):

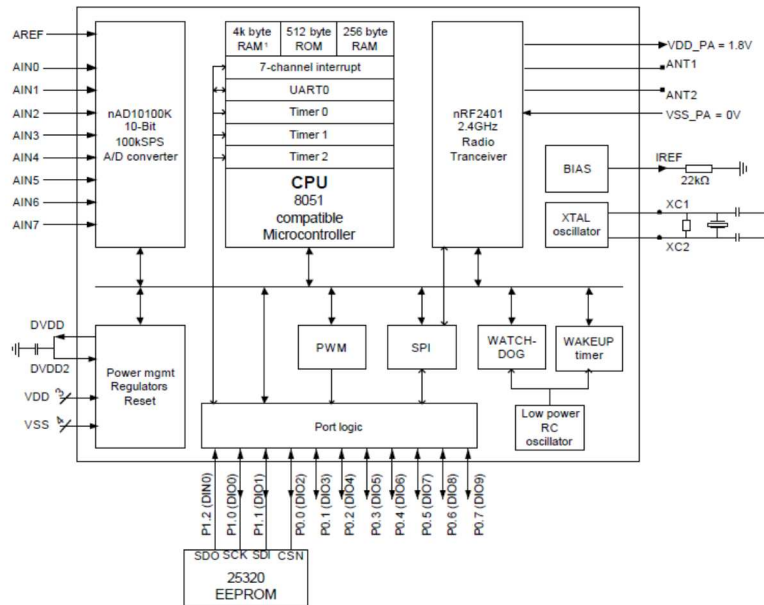


Figura 5 Diagrama de bloques del microprocesador nRF24E1

Como se puede ver en la figura 5, el microcontrolador nRF24E1 dispone de una CPU compatible con la familia 8051 (7). Como se puede ver en el diagrama de bloques, se dispone de un ADC (Analog-to-Digital Converter) que va conectado al sensor de aceleración H34C. Para muestrear la señal del acelerómetro a la frecuencia que deseamos necesitaremos configurar la interrupción de un timer para que cuente hasta 25 ms. Como se puede ver en la figura de arriba el nRF24E1 dispone de tres Timers con sus interrupciones que nos servirán para este fin. Solo usaremos el Timer1.

El nRF24E1 también dispone del transceptor nRF2401 que generará la señal de RF para ser emitida por las antenas.

- H34C Acceleration Sensor (8)

Este sensor de aceleración permite medir la aceleración inercial en las tres direcciones del espacio hasta valores de  $\pm 3g$ . El microcontrolador será el encargado de muestrear la señal producida por el sensor con un ADC. Este conversor dispone de una resolución de 12 bits. Por tanto, los valores oscilarán entre 0 y  $2^{12}-1$ .

- 2.4 GHz Antenna

El ECO Node dispone una antena para la transmisión y recepción que funciona a la frecuencia de 2.4 GHz.

- Crystal Oscillator

Hay un oscilador de cristal funcionando a 16 MHz que es el que generará la señal de reloj del microcontrolador.

- 3.7 V Li-Polymer Battery

La batería que dispone cada uno de los nodos le permite tener una autonomía de 4 horas (40 mAh).

- LED

Los nodos disponen de un led cada uno. Gracias a este led, se podrá ver visualmente si el ECO Node está transmitiendo o por el contrario existe algún problema.

- AT25320 EEPROM

Se dispone de una memoria EEPROM de 4kBytes para almacenar datos del programa.

- 16-pin Expansion Port

Es el puerto de expansión que permite comunicar el nodo por puerto serie con el ordenador. Va a servir para poder cargar a través de él los programas y también para comunicar los nodos directamente con el ordenador.

- LTC3410 Power Regulator

Regulador de tensión que convierte los 3,7 V de salida de la batería en los 2,1 V que usa el microcontrolador.

### 3.2.2 Programación del Firmware

Toda la programación del firmware del ECO Node se ha llevado a cabo en C, con el programa de libre distribución Eclipse (9). La carga del programa se ha llevado a cabo a través de un puerto serie virtual del ordenador que se conecta a la placa de desarrollo proporcionada por el fabricante. Esta a su vez está conectada a los ECO Nodes a través de un adaptador.

#### *Descripción de librerías*

Para la programación del firmware del ECO Node se cuenta con las librerías en C facilitadas por los fabricantes del ECO Kit. Esas librerías han facilitado la programación de los ECO Nodes.

Las librerías disponibles son las siguientes:

- La librería reg24e1: contiene las direcciones de memoria de los registros que se van a modificar para programar el ECO Node.
- La librería eco\_sys.h: incorpora funciones para controlar el led de los nodos.
- En la librería adc.h: se encuentran las funciones necesarias para controlar el ADC que muestreará la señal del acelerómetro.
- La librería isr\_timer.h: permite configurar interrupciones en el timer a la frecuencia que se desee. La subrutina de la interrupción será la encargada de ordenar el muestreo de los acelerómetros. Esta librería se ha tenido que reprogramar entera ya que la proporcionada por el fabricante ha dado problemas al utilizarse.
- Librería serial.h: esta librería permite enviar por el puerto de expansión datos al ordenador. Esta librería la hemos usado durante la programación de los ECO Nodes.
- La librería rf/rf.h incorpora las funciones necesarias para poder enviar y recibir datos por radiofrecuencia.

#### *Diagrama de flujo*

A continuación se muestra una figura que describe el funcionamiento del ECO Node:

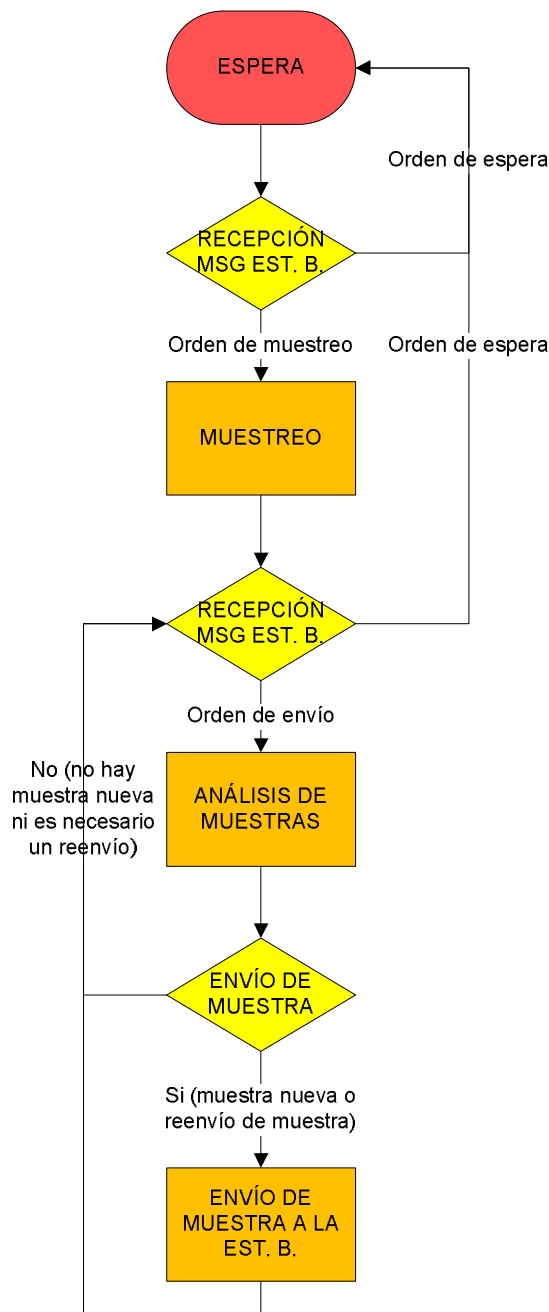


Figura 6 Diagrama de flujo del ECO Node

El ECO Node dispone de cuatro estados: el estado de ESPERA, de MUESTREO, de ANÁLISIS DE MUESTRAS, o de ENVÍO DE MUESTRA A LA ESTACIÓN BASE. Al iniciarse el ECO Node se encuentra en el estado de ESPERA. En este estado el nodo espera la recepción de un mensaje de la Estación Base y una vez recibido mira si contiene una orden de empezar a muestrear datos del acelerómetro.

En el caso de haber recibido la orden de la Estación Base (“Orden de muestreo” en la figura 6), el nodo cambiará al estado de MUESTREO. En este estado, se activará el timer, y con ello el dispositivo empezará a muestrear cada 25 ms.

Una vez iniciado el muestreo, el nodo se pondrá a la espera de recibir un mensaje de la Estación Base. Si se recibe un mensaje de “Orden de Espera” el ECO Node volverá al estado inicial. En el caso de que el mensaje recibido sea de “Orden de envío”, pasará al estado ANÁLISIS DE MUESTRAS.

En el estado ANÁLISIS DE MUESTRAS, el nodo mirará si tiene una nueva muestra que no haya enviado. Si dispone de una muestra nueva, pasará al estado de ENVÍO DE MUESTRA A LA ESTACIÓN BASE. En el caso de no disponer de una muestra nueva, comprobará si la última muestra que envió se perdió al enviarse (lo podrá comprobar gracias a que el mensaje recibido de la Estación Base contiene un campo con la última muestra recibida de cada nodo). En el caso de haberse perdido la última muestra, pasará al estado de ENVÍO DE MUESTRA A LA ESTACIÓN BASE. Si no se perdió la última muestra, pasará de nuevo a esperar un envío de orden de la Estación Base.

El estado ENVÍO DE MUESTRA A LA ESTACIÓN BASE es el encargado de enviar las muestras a la Estación Base.

### 3.3 Estación Base

La Estación Base servirá de puente entre los nodos y el ordenador. Recibirá los datos de los nodos de forma inalámbrica para después transmitirlos al PC por un puerto serie. El ordenador recibirá los datos por un puerto serie virtual y posteriormente los procesará.

#### 3.3.1 Descripción física

Para la implementación de la Estación Base, se ha utilizado el Kit del Freescale DEMO9S12NE64 (5). Este kit contiene todo lo que un diseñador necesita para desarrollar y evaluar las aplicaciones de la familia MC9S12NE64. El microprocesador incluye gran cantidad de módulos de periféricos: EMAC (Ethernet Media Access Controller), 2 SCI (Serial Communication Interface) asíncronos, y un SPI (Serial Peripheral Interface). A continuación se puede ver una figura con la arquitectura del microprocesador:

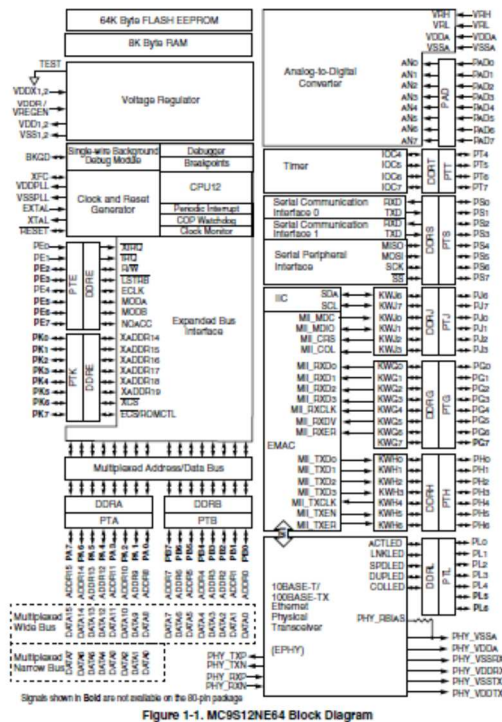


Figura 7 Diagrama de bloques del microcontrolador



Figura 8 Modulo Sparkfun WRL 00705

Durante el desarrollo se ha usado todos los módulos de interfaz, aunque finalmente la comunicación se hará a través de dos puertos: el SCI y SPI. La recepción y envío de la señal de RF será a través del módulo de Sparkfun, WRL-00705 (10). Este módulo usa el transceptor nRF24L01. Funciona a la frecuencia de 2,4 GHz con 126 canales y modulación GFSK. Transmite a una tasa de 2Mbps con una potencia de emisión de 0dBm y una sensibilidad en recepción de hasta -85dBm. A través de él recibiremos la señal que será enviada al SPI del Freescale.

### 3.3.2 Programación del firmware

La programación de la Estación Base se ha desarrollado en C. Se ha utilizado un software proporcionado por el fabricante llamado CodeWarrior Development Studio. La empresa Freescale proporciona con el DEMO9S12NE64 un proyecto de desarrollo llamado Connector App. Este proyecto ha servido de base para la programación de la aplicación.

Las modificaciones que se han hecho en el proyecto Connector App han sido fundamentalmente incorporar dentro del bucle principal del programa dos funciones que se



irán alternando. Una de ellas es la encargada de enviar las órdenes al ECO Node, y otra es la encargada de recibir los datos del ECO Node. Estas dos funciones componen un diagrama de flujo de cinco estados que se puede ver en la figura siguiente:

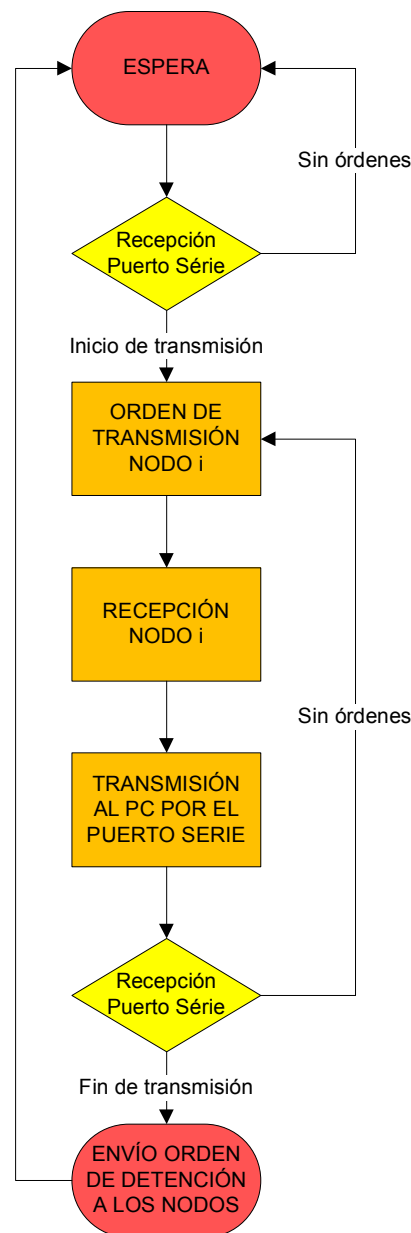


Figura 9 Diagrama de flujo de la Estación Base

Inicialmente, la Estación Base se va a encontrar en el estado de ESPERA. En este estado, lo único que debe hacer es comprobar si en el puerto serie ha llegado alguna orden de iniciar el muestreo de los nodos y la transmisión. Cuando llega esa orden, pasa al estado de ORDEN DE TRANSMISIÓN NODO i. En este estado, la Estación Base envía una orden de envío de mensaje al nodo “i” (el número de nodo “i” se irá alternando entre los dos nodos que tiene el sistema). Una vez enviado el mensaje pasará al estado RECEPCIÓN NODO i. En este estado, recibirá la muestra enviada por el nodo “i”. Una vez recibida la muestra la enviará por el puerto serie al PC (estado TRANSMISIÓN AL PC POR EL PUERTO SÉRIE). A continuación comprobará si hay una

nueva orden del PC de parar de muestrear. En el caso de haber una orden de parada, pasará al estado ENVÍO ORDEN DE DETENCIÓN A LOS NODOS, donde hará un último envío a los ECO Nodes para que se paren y de esta forma consuman menos energía. En el caso de no recibir ninguna orden del PC de parar la transmisión, volverá al estado ORDEN DE TRANSMISIÓN NODO i.

### 3.4 Comunicación entre los ECO nodes y la Estación Base

#### 3.4.1 Introducción

La comunicación entre la Estación Base y los ECO Nodes se realiza a la frecuencia de 2,4 GHz. Los ECO Nodes disponen de una antena para enviar y recibir datos por RF. La Estación Base realiza lo mismo a través de un módulo externo de RF.

Al implementar la comunicación entre los nodos y la base, se ha encontrado que al poner en funcionamiento más de un ECO Node a la vez, la tasa de error (número de muestras perdidas en la transmisión) aumenta de forma significativa.

Esto se debe a que en la librería rf.h no implementa un control de acceso al medio, y por tanto, al ponerse a transmitir varios nodos a la vez colisionan provocando errores en la transmisión. Al no ser objeto del proyecto final de carrera la programación de unas nuevas librerías, se ha hecho un estudio de diferentes protocolos de comunicación entre la Estación Base y los nodos para ver el sistema que produce menor tasa de error.

Una solución al problema es realizar un control de acceso al medio desde la Estación Base. La Estación Base “dirigirá” la comunicación para evitar interferencias entre los ECO Nodes.

Se han probado diferentes técnicas para implementar este protocolo de comunicación. Para cada una de las diferentes posibilidades se ha hecho un estudio del número de errores de transmisión para hallar la mejor solución<sup>1</sup>.

La primera prueba consistió en verificar que el alcance de un ECO Node transmitiendo podía llegar a la distancia exigida en las especificaciones (al menos 3 metros de distancia). Como se puede ver en el Anexo 6, se alcanzan sin problemas. En el mismo experimento, se han comparado diferentes frecuencias de muestreo. La conclusión ha sido que disminuir la frecuencia de muestreo no reduce la tasa de error significativamente, y que por tanto, la frecuencia de muestreo será 40Hz, más que suficiente para la aplicación que queremos.

La siguiente prueba ha consistido en poner a transmitir a dos nodos simultáneamente. En este modo de transmisión los ECO Nodes solo reciben la orden de inicio de muestreo de la Estación Base. Una vez recibida la orden, empiezan a emitir sin escuchar nuevas órdenes de la Estación Base. Esto ha provocado que, en muchas ocasiones, uno de los nodos no consiguiera transmitir correctamente ninguna muestra.

Como consecuencia de todo esto se ha cambiado el protocolo de comunicación. La comunicación pasa a ser bidireccional. Cada nodo debe esperar la orden de la Estación Base para enviar su mensaje. Esto permite que la comunicación de los dos nodos no se interfiera. El

---

<sup>1</sup> Ver Anexo 6 Estudio de errores en los ECO Nodes

resultado ha sido que se disminuye significativamente la tasa de error, hasta unos niveles aceptables para nuestro sistema.

Para el sistema que se ha desarrollado, es más que suficiente disponer de 2 nodos funcionando a la vez. Por tanto, en este proyecto final de carrera no nos hemos adentrado más en la programación de la transmisión RF. Se deja para PFC posteriores una optimización de los recursos RF disponibles en los ECO Nodes y la Estación Base.

### 3.4.2 Protocolo de comunicación

La estructura del paquete de datos que van a transmitir los ECO Nodes a la Estación Base es el siguiente:

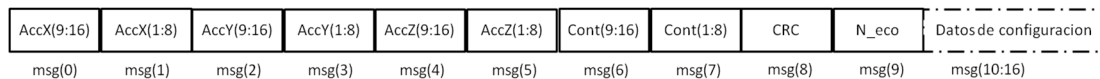


Figura 10 Trama de datos de la comunicación nodo-base

Como se puede ver en la figura 10 se disponen de 17 bytes. En ellos se introducirán los siguientes datos: aceleraciones de los tres ejes del espacio, un contador (número de secuencia para saber si se ha perdido alguna muestra), CRC para corregir errores, número del ECO Node que transmite, y otros datos de configuración (como la dirección de destino, etc.).

La comunicación se realiza a una frecuencia de 2,4 GHz. Se usa uno de los 125 canales disponibles para la comunicación. Cada uno de los dispositivos (la base y los dos nodos) tendrán una dirección única que es la que los identificará. La base tiene la dirección {0x65, 0x65, 0x65} mientras que los nodos tienen las direcciones {0xb7,0xb7,0xb7} y {0xb5,0xb5,0xb5}.

El protocolo de comunicación propuesto finalmente es el que se describe en la figura 11:

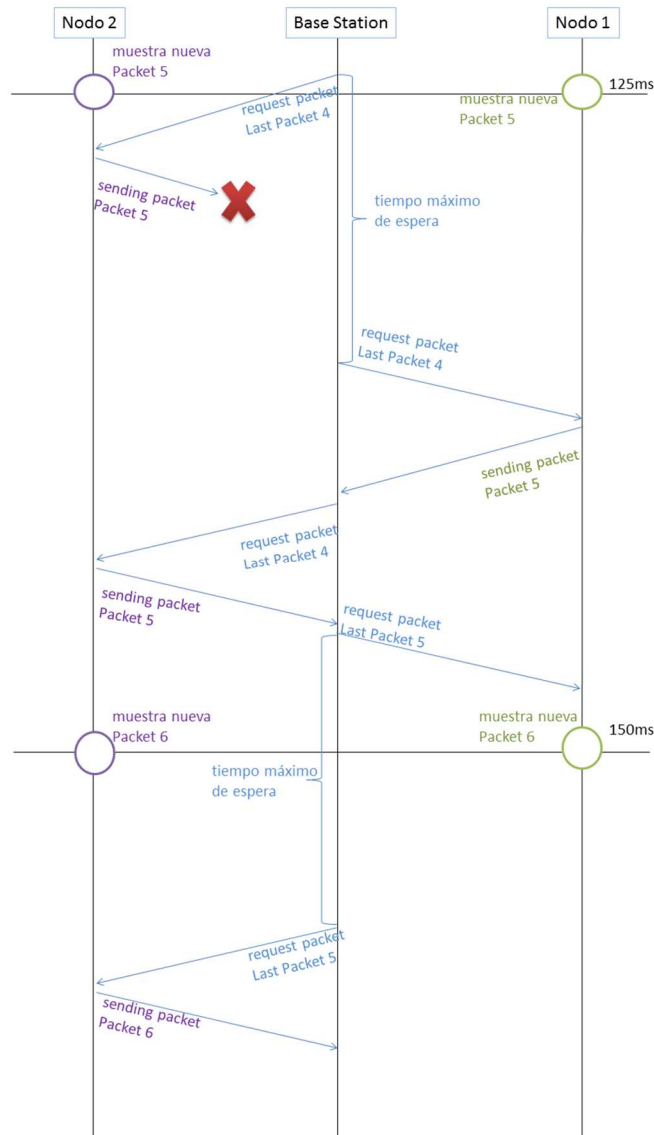


Figura 11 Diagrama del proceso de la comunicación entre dos nodos y la base

A continuación se va a describir el diagrama paso a paso para entender cómo funciona este protocolo.

Como se puede ver en el diagrama, la Estación Base envía al nodo 2 una orden de petición de muestra nueva, diciéndole que la última muestra recibida suya ha sido la 4 (Está indicado donde dice Last Packet 4). El nodo comprueba que tiene una muestra nueva (Packet 5), y la envía a la Estación Base. Sin embargo, la base no recibe el mensaje por un fallo en la transmisión.

Una vez finalizado el tiempo de espera de la Estación Base da por perdida la muestra y envía otra petición de muestra al nodo 1. El nodo 1 comprueba primero si tiene un mensaje nuevo, y como lo tiene, envía la muestra nueva a la Estación Base (Packet 5).

A continuación, la base volverá a enviar al nodo 2 una petición de muestra (indicando que la última muestra recibida ha sido la 4). El nodo 2, primero de todo, comprueba si ha muestreado

una nueva muestra. Como no es el caso, comprobará si la última muestra recibida por la Estación Base es la muestra 5 (la última muestra que envió, que se perdió en la transmisión). Como la última muestra recibida es la 4, el nodo llegará a la conclusión que la muestra 5 se perdió. Por tanto, reenviará una vez más esa muestra a la Estación Base.

Este ejemplo nos muestra cómo se podrán corregir errores de transmisión de forma dinámica.



## Capítulo 4 Análisis del grado de actividad y bio-feedback

### 4.1 Introducción

En este capítulo se va a describir el módulo 3 del sistema desarrollado. Es decir, se va a explicar la parte de recepción y procesado de señal del PC, y el bio-feedback.

Para la recepción y procesamiento de los datos se ha escogido Matlab (11). Este programa aporta gran capacidad de cómputo y permite probar distintos algoritmos de una forma sencilla.

Vamos a usar Matlab para todo el procesamiento de los datos que llegan al PC. Se va a encargar de recibir los datos del puerto serie, adquirir los valores de los acelerómetros, detectar errores, filtrar las muestras, crear un algoritmo para medir la actividad del usuario, y por último, generar el bio-feedback.

Como se ha contado en el capítulo 2, se han programado diferentes scripts a medida que se ha ido desarrollando el PFC. Cada uno de estos scripts ha tenido su función en el proyecto. Los scripts son:

1. Script en Matlab para el estudio de errores
2. Script en Matlab para la toma de datos de usuarios
3. Script en Matlab para validar los algoritmos de detección de actividad
4. Script en Matlab de la aplicación final con bio-feedback

A continuación se van a describir cada uno de estos scripts y se contará como se ha implementado el software del sistema de bio-feedback.

### 4.2 Script en Matlab para el estudio de errores

El objetivo del desarrollo de este primer script ha venido de la necesidad de encontrar el mejor protocolo de comunicación entre la Estación Base y los nodos. Como se ha descrito en el capítulo 3, se han encontrado problemas al poner en funcionamiento varios nodos a la vez. Esto ha provocado que se hayan tenido que probar diferentes métodos de comunicación. La verificación de cada uno de ellos se ha realizado en el PC viendo el número de errores que producía la comunicación.

Los resultados de este estudio se pueden ver en el Anexo 6.

El diagrama de bloques de este script se puede ver en la figura siguiente.

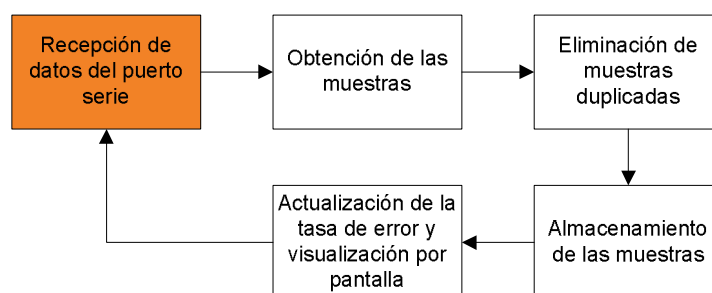


Figura 12 Diagrama de flujo del script para el estudio de errores

Primero de todo se recibe por el puerto serie los datos que envía la Estación Base. Recibidos los datos, se detectan las muestras de los acelerómetros para cada uno de los nodos. Una vez se han obtenidas, se eliminan las muestras duplicadas. Las resultantes se guardan en un archivo para poder analizar los datos posteriormente. Se actualiza la tasa de error y se pasa por pantalla para que dinámicamente el usuario vea los errores que está produciendo la comunicación.

### 4.3 Script en Matlab para la toma de datos de usuarios

#### 4.3.1 Objetivo

El objetivo de este script es crear una base de datos de usuarios en diferentes estados de actividad para probar los diferentes algoritmos de detección.

Esa base de datos de muestras se debería haber tomado en niños TDAH para que la validación de los diferentes algoritmos fuera más real. Sin embargo, realizar pruebas con menores exige un largo proceso burocrático fuera del alcance del presente proyecto.

Por lo tanto se ha optado por usar “actores” que simularan el estado de actividad de un niño TDAH. Para ello se han escogido 17 personas diferentes que han simulado cada uno de los estados de actividad de un niño TDAH durante 5 minutos. Durante este tiempo, se han tomado datos de su aceleración, y se han guardado en una base de datos. Este estudio está descrito detalladamente en el Anexo 7.

El fin de este script en Matlab es capturar los valores de la aceleración de los actores que simularán diferentes estado de actividad y almacenarlos de forma ordenada en distintos ficheros.

Este script se puede encontrar en el DVD del PFC en /scripts/pserial2.m.

#### 4.3.2 Diagrama de bloques

El diagrama de bloques de este script se puede ver a continuación:



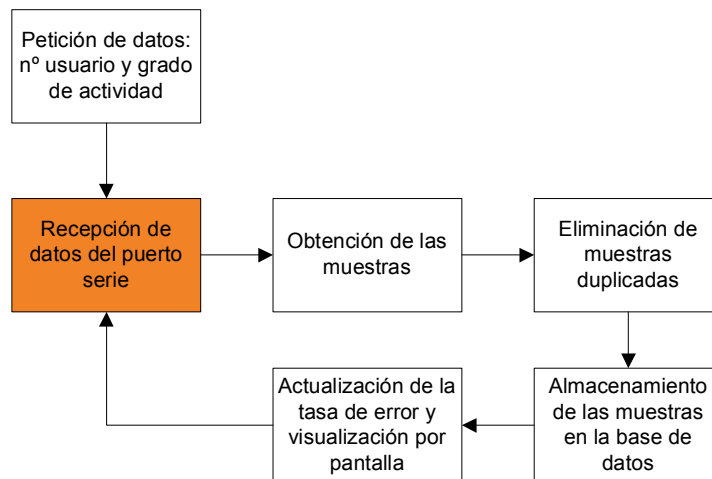


Figura 13 Diagrama de flujo para el script de toma de datos en personas

El diagrama de bloques es muy parecido al del script anterior. Incorpora un estado inicial (Petición de datos: nº de usuario y grado de actividad). Su función es pedir al usuario la información suficiente para poder almacenar los datos indicando el número de usuario, el grado de actividad del usuario, la fecha y hora de la captura (timestamp).

Los resultados de este estudio están descritos en el Anexo 7.

## 4.4 Desarrollo y prueba de distintos algoritmos para el análisis del grado de actividad

### 4.4.1 Objetivo

En este apartado se va a describir como se ha implementado un script en Matlab para poder validar los distintos algoritmos.

En primer lugar se explicará cada uno de los bloques del diagrama del script. Dentro de los bloques se dedicará un espacio extenso al bloque de detección, contando en detalle los dos algoritmos probados.

Por último se hará una comparativa de los dos algoritmos, y las conclusiones a las que se han llegado.

Las dos partes de este script están disponibles en el DVD del PFC:

1. /scripts/pfc.m
2. /scripts/prova.m

### 4.4.2 Diagrama de flujo

Este script en Matlab lo podemos dividir en dos partes:

La primera parte (pfc.m) es la responsable de adquirir los datos de la base de datos de muestras generada en el script anterior y procesarlas en el detector. Los datos a la salida del detector, producirán unos valores que variarán en función del grado de actividad. Para una actividad del usuario alta (movimientos bruscos, saltos, etc.), los valores a la salida del detector de actividad serán altos, para una actividad del usuario baja, los valores serán bajos.

La segunda parte del script (prova.m) se encargará de comparar los datos a la salida del detector de actividad con los umbrales para adquirir el umbral que dé la mínima tasa de error en la detección de la actividad.

A continuación podemos ver el diagrama de flujo de la primera parte del código:

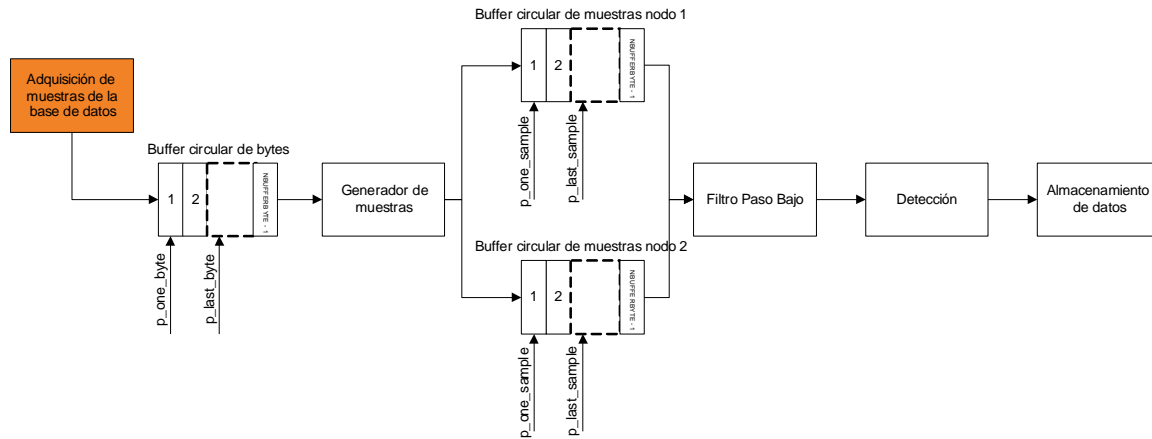


Figura 14 Diagrama de bloques del script de cálculo de los umbrales óptimos

Este diagrama de bloques describe el proceso desde que se adquieren las muestras de la base de datos, hasta la detección. Como el objetivo de este script es validar el sistema de detección de actividad, este sistema debe parecerse lo máximo posible al sistema final. Por tanto, el diagrama de bloques de arriba es muy parecido al diagrama de bloques del sistema final que se podrá ver en los apartados siguientes.

A continuación se van a explicar en detalle cada uno de los bloques del sistema.

### Adquisición de muestras de la base de datos

El primer bloque de todos es el responsable de adquirir las muestras guardadas en la base de datos e introducirlas en un buffer circular de bytes. Este bloque debe simular al bloque de recepción del puerto serie del sistema final.

El formato de la base de datos es el que se ha explicado en el apartado 4.3. Es decir, está almacenado el número de nodo, las aceleraciones en los tres ejes del espacio, el número de secuencia de cada muestra, y el timestamp (hora de recepción de la muestra).

Este bloque adquirirá las muestras de la base de datos y los pasará al formato byte. A continuación, almacenará los bytes en un buffer circular. Se puede ver en la figura 15 la trama que se guardaría en el buffer circular de bytes.

'F'	AccX(1)	AccX(2)	AccY(1)	AccY(2)	AccZ(1)	AccZ(2)	Cont(9:16)	Cont(1:8)	N_nodo	CRC	'F'	^v
line(1)	line(2)	line(3)	line(4)	line(5)	line(6)	line(7)	line(8)	line(9)	line(10)	line(11)	line(12)	line(13)

Figura 15 Trama de datos Estación Base - PC

El buffer circular dispone de dos punteros que marcan el inicio y fin de bytes válidos (*p\_one\_byte* y *p\_last\_byte*).

### Generador de muestras

Este bloque va a ser el encargado de adquirir los bytes guardados dentro del buffer circular, procesarlos, y almacenarlos en distintos buffers en función del número de nodo.

Su primera tarea debe ser diferenciar si los bytes recibidos corresponden a una muestra válida. Para ello dividirá la trama de bytes obtenida en bloques que estén separados por el carácter '\r'. Este carácter es el carácter final de la trama que envía el puerto serie. Por tanto, entre dos tramas completas de muestras se deberá tener ese valor. Una vez se han separado las muestras entre sí, se procede a comprobar si la longitud es correcta y si la muestra inicial y final son 'F'. Entonces se procederá a analizar los valores de las aceleraciones, el número de nodo, y el número de secuencia de cada muestra. Se verificará el CRC, y si es correcto, se procederá a almacenar la información en los buffers correspondientes.

Se va a almacenar los datos en un archivo de texto cuyo formato será similar al usado anteriormente para almacenar datos del acelerómetro<sup>2</sup>. El formato será SVC, por tanto, un archivo separado por comas. Podemos ver un ejemplo debajo:

```
timestamp,time,nodo,ax,ay,az,sec
20110109183416,3416,4,2732,2824,2229,1
20110109183416,3416,4,2739,2830,2235,2
20110109183416,3416,4,2730,2830,2231,3
20110109183416,3416,4,2731,2826,2224,4
20110109183416,3416,4,2733,2822,2227,5
```

En primer lugar irá un campo con el *timestamp*, la fecha y la hora en que se recibió por el puerto serie esa aceleración. En segundo lugar tenemos, los minutos y segundos. Después el nodo, las aceleraciones y el número de secuencia.

El nombre del archivo va a tener un formato similar al usado para almacenar muestras anteriormente<sup>3</sup>. Sera *Date\_YYYY-MM-DD\_Hour\_hh-mm-ss\_Person\_P\_Activity\_A.mat*.

Tener guardadas todas las muestras que se van obteniendo nos va a servir para detectar errores, hacer estadísticas, etc.

Una vez hemos almacenado las muestras dentro del fichero de datos, se copian a cada uno de los buffers de muestras de cada nodo. Los datos que se almacenarán serán: fecha y hora, número de nodo, aceleración en cada eje del espacio, y número de secuencia.

Por último, el bloque "Packet generator" avisará al bloque "Filtro Paso Bajo" cuando haya un número mayor de muestras que NPACKET para que empiece el filtrado.

### Filtro Paso Bajo

El bloque "Filtro Paso Bajo" es el encargado de hacer un filtrado preliminar para eliminar las altas frecuencias que rompen la estabilidad del sistema. El filtro elegido ha sido un filtro de Bessel. Los filtros de Bessel tienen la peculiaridad de ser los que tienen la fase más lineal en la banda pasante. Este filtro debe ser muy sencillo (pocos coeficientes) para no aumentar la carga

<sup>2</sup> Ver Anexo 2: Toma de datos en personas

<sup>3</sup> Ver Anexo 2: Toma de datos en personas.

computacional. A continuación se puede ver los coeficientes de la matriz SOS (Second Order Sections):

1	0.205353559727552	0.999921834772360	1	-0.541496914075217	0.735336500914686
1	0.999341071456313	0	1	-0.422882160496947	0

Como el número de coeficientes diferentes de 0 y 1 son 6, en total se realizarán 6 multiplicaciones por cada muestra nueva.

A continuación se puede ver el modelo del filtro:

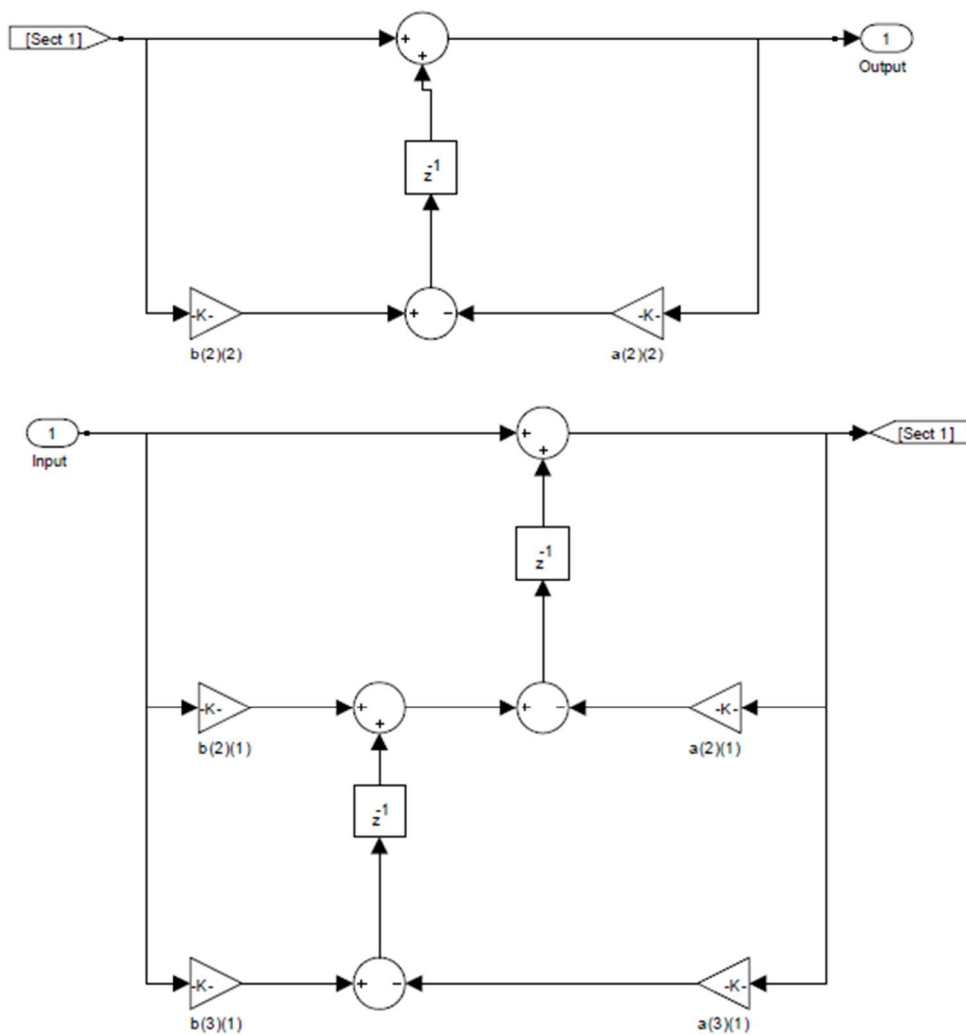
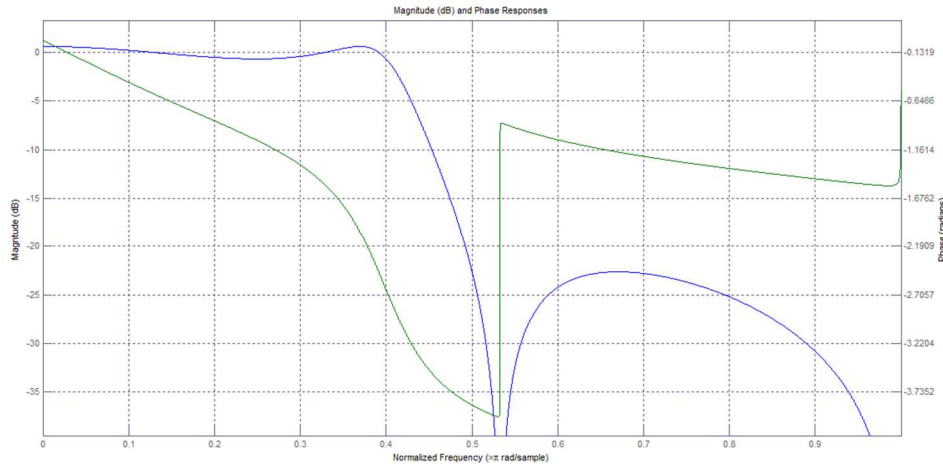


Figura 16 Modelo del filtro de Bessel

La respuesta frecuencial del filtro es la que se ve a continuación:



Gráfica 1 Respuesta frecuencial

Este filtro se ha diseñado para filtrar señales a frecuencia mayores de 16KHz. Como un usuario no puede realizar movimientos a mayor frecuencia, se filtran esas frecuencias. A los 20 Hz la atenuación será superior a 20 dB.

Como se puede ver en la gráfica, apenas hay distorsión de fase en la banda pasante.

### Detección

El bloque detección será el responsable de procesar las aceleraciones para obtener un nivel de actividad para cada nodo. Posteriormente, otro bloque se encargará de concluir en qué estado de actividad se encuentra el usuario, en función de la actividad de cada nodo.

Se han implementado dos algoritmos de detección. A continuación se va a explicar el funcionamiento de cada uno de ellos.

### Cálculo de la varianza

La figura de abajo esquematiza el sistema de detección “Cálculo de la varianza”.

Este sistema de detección calcula la varianza del módulo de las aceleraciones. De esas varianzas se hace una media.

El resultado de la media será el valor final que será introducido en el detector. El detector comparará la media obtenida con los diferentes umbrales (o niveles) Con esto sabrá en qué estado de actividad se encuentra cada nodo que lleva el usuario. Además, los datos de las medias de las varianzas serán guardados en un

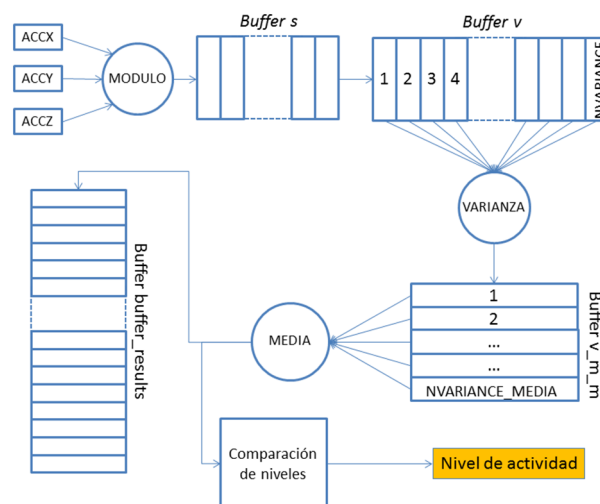


Figura 17 Diagrama de flujo del detector de cálculo de la varianza

buffer para poderlos visualizarlos en tiempo real. Mirando la figura 17, podemos ver el funcionamiento:

Primero de todo se calcula el módulo de las aceleraciones (ACCX, ACCY, ACCZ). El resultado del cálculo se va introduciendo secuencialmente (como un registro de desplazamiento) en el buffer  $v$ . Este buffer es el encargado de hacer el cálculo de la varianza de NVARIANCE valores. Cada vez que se hace el cálculo de la varianza, se desplaza el buffer una posición de manera que entra un dato nuevo, saliendo otro. A continuación se vuelve a calcular la varianza. Cada cálculo de la varianza se guarda en otro buffer ( $v\_m\_m$ ). Este buffer almacena cada uno de las varianzas para posteriormente calcular la media de las últimas NVARIANCE\_MEDIA varianzas. Funciona igual que el buffer  $v$ . Al introducirse una nueva varianza se eliminará la primera y se calculará la media de las NVARIANCES\_MEDIA varianzas. Cada vez que obtenemos una de estas medias, se compara con los diferentes niveles de detección de actividad y de esta manera se concluye en qué estado de actividad se encuentra cada uno de los nodos.

Existen 4 niveles de actividad en el que se puede encontrar el nodo: nivel de actividad nulo, bajo, medio y alto. Para concluir el nivel de actividad de cada nodo se comparará la media obtenida del buffer  $v\_m\_m$  con unos umbrales.

Los valores de NVARIANCE\_MEDIA y NVARIANCE son ajustables por el usuario.

#### Diferencia muestra a muestra

Este método del cálculo de la actividad se fundamenta en medir la pendiente del módulo de las aceleraciones. Con el valor de la pendiente se calculará el grado de actividad de los nodos.

A continuación se puede ver un diagrama con el funcionamiento principal:

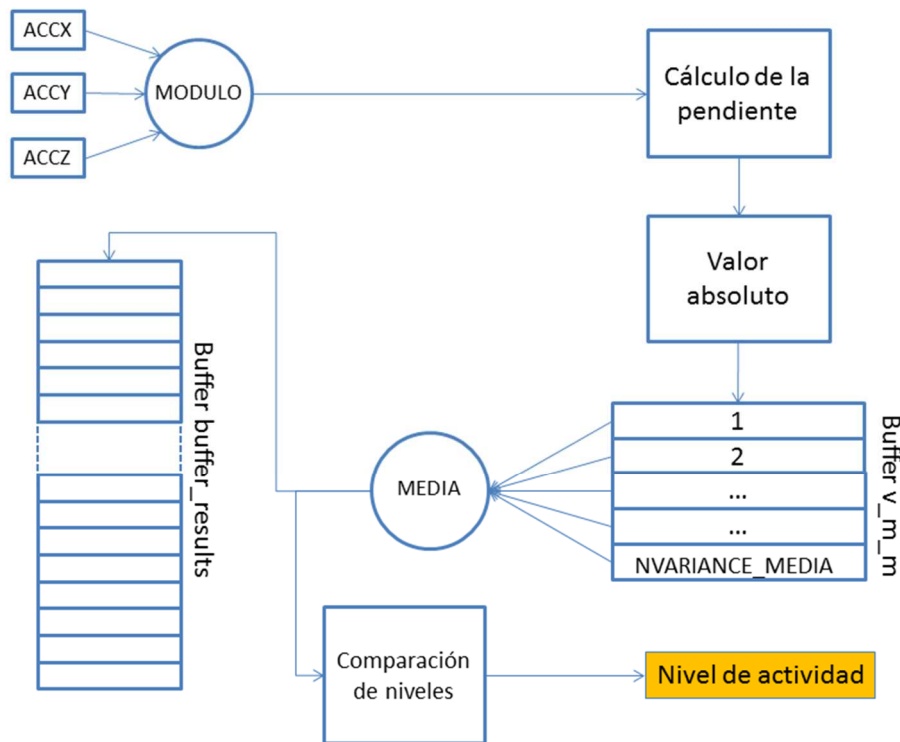


Figura 18 Diagrama del detector del cálculo de la pendiente

Primero de todo se calculará el módulo de la aceleración en los tres ejes del espacio: ACCX, ACCY, ACCZ.

Una vez calculado el módulo, se procederá a calcular la pendiente de ese módulo. Posteriormente se hará el valor absoluto, y se calculará la media de NVARIANCE\_MEDIA valores absolutos. Esa media nos servirá para saber el nivel de actividad de cada nodo. Estas medias se almacenarán en un buffer para posteriormente poder tratar esos datos.

El valor de las medias se comparará con los umbrales en el bloque “Comparación de niveles”. Este bloque proporcionará el nivel de actividad de cada nodo: nulo, bajo, medio o alto.

#### Almacenamiento de datos

Este último bloque será el encargado de almacenar en una base de datos (*datos estadísticos*<sup>4</sup>) los valores obtenidos por el bloque *Detección* para procesarlos después convenientemente.

#### 4.5.3 Validación con base de datos

Para validar los dos algoritmos de detección, se ha utilizado la base de datos obtenida de los “actores” que simularon diferentes grados de actividad<sup>5</sup>.

<sup>4</sup> Disponibles en el DVD del PFC: /basesdedatos/bateriademuestras.

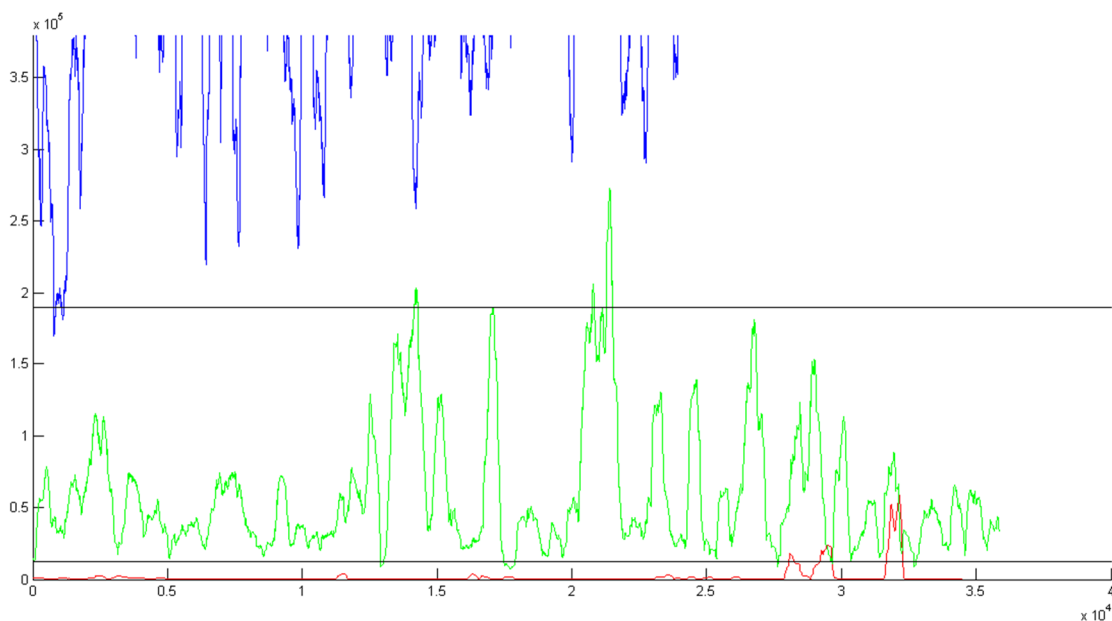
<sup>5</sup> El script utilizado para capturar datos es el que se ha contado en el apartado 4.3 Script en Matlab para la toma de datos de usuarios. Además, el estudio detallado de esa toma de datos se encuentra en el Anexo 7: Toma de datos.

Para validar el sistema se han realizado las siguientes acciones para cada algoritmo de detección:

- I. Se han juntado las muestras de las actividades de cada persona ordenándolas por el grado de actividad (actividad baja, media y alta).
- II. Se han introducido las muestras de cada actividad en el sistema explicado anteriormente, obteniendo así, los valores de los detectores de actividad para cada nodo.
- III. Se ha comprobado como en función del grado de actividad, el valor a la salida del detector ha cambiado, subiendo en el caso de un aumento de actividad, y bajando al disminuir la actividad.
- IV. Se han obtenido los umbrales óptimos para cada nivel de actividad y para cada nodo.

Para la obtención de los umbrales óptimos se ha programado un script en Matlab para que obtuviera ese valor<sup>6</sup>. Los umbrales óptimos serán los que den menor tasa de error a la hora de detectar el estado de actividad de cada uno de los nodos.

En la figura siguiente se puede ver una gráfica con muestras de algunos de los “actores”. Están representados en azul los valores a la salida del detector para muestras a una actividad alta. En verde para una actividad media, y el rojo para actividad baja. Está representado con dos líneas horizontales donde estarían los umbrales óptimos.

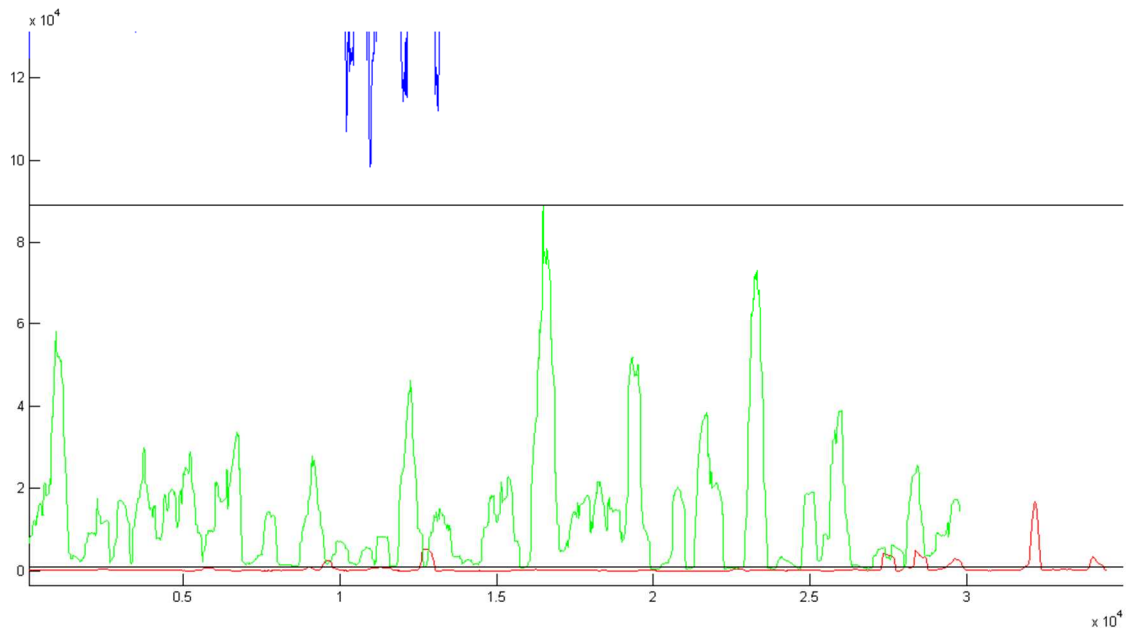


Gráfica 2 Umbral del detector de cálculo de la varianza. Muñeca

A continuación podemos ver la misma gráfica pero para el colgante:

<sup>6</sup> Disponible en el DVD del PFC: /scripts/prova.m.





**Gráfica 3 Umbrales del detector de cálculo de la varianza. Colgante**

Como se puede ver en la gráfica 2 y 3, los umbrales son diferentes para cada uno de los nodos. Los umbrales para la muñeca serán 189000 y 12300, mientras que para el colgante serán 89000 y 900. La tasa media de error es 2,15%.

Para el algoritmo del cálculo de la pendiente se ha realizado un análisis equivalente. Los umbrales obtenidos son 30 y 234 para la muñeca, y 10 y 116 para el colgante. La tasa media de error es de 3,25%, un poco peor que el algoritmo de cálculo de la varianza<sup>7</sup>.

Por tanto, se ha podido validar la efectividad de los algoritmos de detección de actividad. Con estos algoritmos se ha programado la aplicación final.

## 4.6 Desarrollo software del sistema de bio-feedback

### 4.6.1 Objetivo

El objetivo que nos proponemos es obtener una herramienta preparada para ser usada por psicólogos de la Asociación Atenciona para que puedan validar el sistema de detección de actividad con los sensores inerciales. La herramienta debe estar planteada para que la pueda usar un psicólogo dentro de las clases de ayuda al autocontrol con niños TDAH.

Por tanto, el psicólogo debe disponer:

1. Un menú de configuración del sistema donde pueda modificar diversas opciones:
  - a. Número de ECO Nodes trabajando simultáneamente (1 o 2).
  - b. Umbrales de detección de actividad.
  - c. Tiempo de cambio de actividad.
  - d. Activar o no el bio-feedback.
2. Un sistema de bio-feedback audio-visual.
3. Opción de parar o reiniciar el sistema cuando quiera.

<sup>7</sup> Más detalles del cálculo realizado en el Anexo 2

#### 4.6.2 Diagrama de flujo

En la figura de abajo podemos ver un diagrama de bloques del programa de recepción y detección.

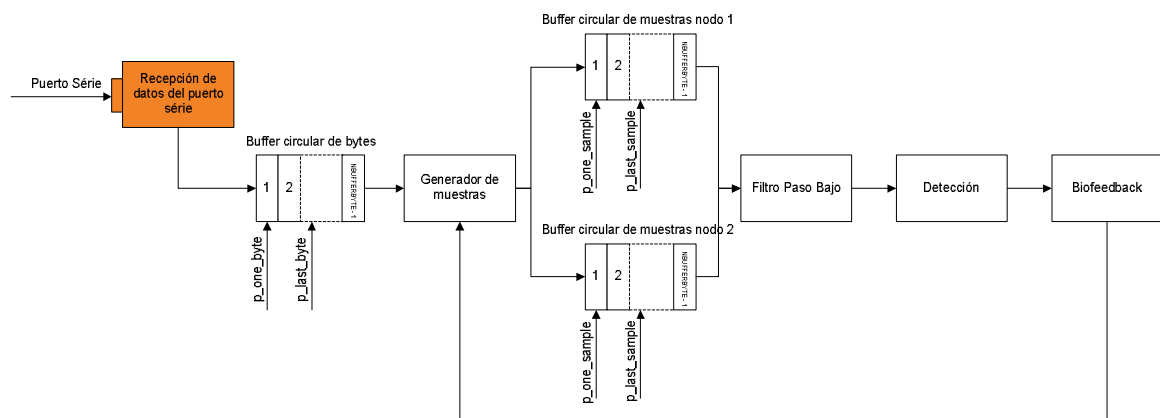


Figura 19 Diagrama de flujo del software del bio-feedback

El bucle principal del programa, primero de todo, convertirá los bytes en paquetes de muestras que contengan las aceleraciones, el número de nodo, el número de secuencia, y el momento en que se recibieron las muestras. Una vez traducidos los bytes a muestras, se almacenan las muestras en un buffer diferente para cada nodo. El bloque "Filtro paso bajo" adquirirá un paquete de muestras de cada nodo y le hará un filtrado paso bajo para eliminar las altas frecuencias. Una vez filtradas las aceleraciones, se procederá a analizarlas en el bloque Detección. Este bloque implementará alguno de los dos sistemas de detección que se ha programado. Una vez detectado el estado de actividad, se procederá a actualizar el bio-feedback.

En un principio se añadió un bloque antes del bloque Filtro Paso Bajo llamado Reconstrucción. Este bloque tenía la misión de reconstruir la señal cuando se produjeran desvanecimientos de señal en una o varias muestras. Posteriormente se ha visto que implementar un sistema de reconstrucción implica una elevada carga computacional y que no aporta grandes ventajas. Por todo ello, este bloque ha quedado descartado. De todas formas en el anexo 8 se encuentra la implementación de un filtrado adaptativo para la reconstrucción de la señal.

A continuación se va a explicar en detalle cómo se ha programado cada uno de los bloques del sistema.

#### Recepción Puerto Serie

El puerto serie se ha configurado a una tasa de recepción de 115200 bits/segundo. El programa leerá de forma continua si hay nuevos bytes disponibles. En el caso de que así sea, los irá almacenando en un buffer interno hasta llegar a tener 50 bytes. Al llegar a esa cantidad, saltará la rutina de interrupción que almacenará los bytes en el buffer circular. El buffer circular dispone de dos punteros:  $p\_one\_byte$ , y  $p\_last\_byte$ .  $p\_one\_byte$ , apunta a la primera posición donde se encuentra un byte válido.  $p\_last\_byte$ , en cambio, apunta a la dirección posterior al último byte válido. Por tanto, los bytes almacenados serán los que vayan desde  $p\_one\_byte$  hasta  $p\_last\_byte-1$ . La rutina de interrupción del puerto serie, cada vez que

quiera añadir bytes al buffer, solo tendrá que aumentar (circularmente) el puntero *p\_last\_byte*.

**Generador de muestras (packet generator)**

Este bloque es el mismo que el explicado en el apartado 4.4.2.

**Filtro paso bajo**

Contado con detalle en el apartado 4.4.2.

**Detección**

Los algoritmos de detección se han descrito detalladamente en el apartado 4.5

**Bio-feedback**

Este bloque va a ser el encargado de adquirir los valores de actividad de cada nodo y concluir cual es el estado de actividad del usuario. Una vez obtenido el estado de actividad generarán el bio-feedback. A continuación se va a describir como se ha realizado el cálculo de la actividad global del usuario

**Cálculo del nivel de actividad global**

El bloque de detección dará un valor de actividad para cada nodo. Ese nivel de actividad va a ser el que se va a comparar con diferentes umbrales para obtener el estado de **actividad de cada nodo** (puede ser que los niveles de actividad de los dos nodos sean diferentes entre sí). Una vez sabemos el estado de actividad de cada uno de los nodos se puede concluir el estado de actividad del usuario. En la tabla siguiente se muestra el nivel de actividad del usuario en función de los niveles de cada uno de los nodos.

Tabla 1 Tabla de actividades de los nodos

		Actividad nodo muñeca			
		1	2	3	4
Actividad nodo colgante	1	2	3	4	5
	2	3	4	5	6
	3	4	5	6	7
	4	5	6	7	8

En el cuadro se muestran dos filas con fondo negro. Esas dos filas indican el nivel de actividad de cada nodo. Sumando la actividad de cada nodo se obtienen diferentes combinaciones. Si la actividad del colgante es 1 y el de la muñeca también, la suma será 2 (fondo blanco). Si en cambio la actividad del colgante es 4, y la de la muñeca es 3, la suma será 7 (fondo amarillo).

Se puede calcular fácilmente la actividad del usuario sumando las dos actividades de los dos nodos en funcionamiento.

Si el resultado es 2, querrá decir que los dos nodos están en reposo absoluto (nadie los lleva puestos) y que por tanto la actividad es absolutamente nula (fondo de la casilla blanco).

Si la suma es mayor a 2 y menor a 6 se tiene un estado de actividad bajo (color verde). Normalmente los dos nodos estarán en estado de actividad 2. Aunque en algunas ocasiones uno de los dos se pueda poner en estado 3 (actividad media), se considera que el estado es bajo (el algoritmo es más bien conservador).

Si los dos nodos pasan a un estado de actividad media, y por tanto la suma es mayor a 5 y menor a 8, se concluye que estamos en actividad media (color amarillo).

Por último, solo si los dos ECO Nodes pasan a una actividad alta, se considerará que se ha entrado en un estado hiperactivo (color rojo, suma total 8).

El sistema funcionaría de esta forma en el caso de disponer de dos nodos funcionando a la vez. En el caso de tener solo 1 nodo funcionando simultáneamente el grado de actividad global se calculará directamente del grado de actividad del nodo activo.

Para conseguir mayor fiabilidad y no ser afectados por movimientos bruscos, el usuario deberá mantenerse unos segundos (configurable por el usuario) en el mismo estado de actividad. Esta opción se implementará en el sistema final. Será configurable por los psicólogos que controlen el sistema.

#### Bio-feedback

Los profesionales de Atención, como primera aproximación al sistema de bio-feedback, sugirieron utilizar una imagen de un semáforo que va cambiando de color en función del estado de actividad del usuario. El color rojo representará el estado de actividad alto, el color amarillo actividad media, el color verde actividad baja, y por último si no hay color, será el estado de actividad nula.

Además, se ha desarrollado un feedback auditivo que se activa cuando se pasa a un estado de actividad alto.

#### 4.6.4 Menú de configuración

La siguiente imagen muestra el menú de configuración del sistema:

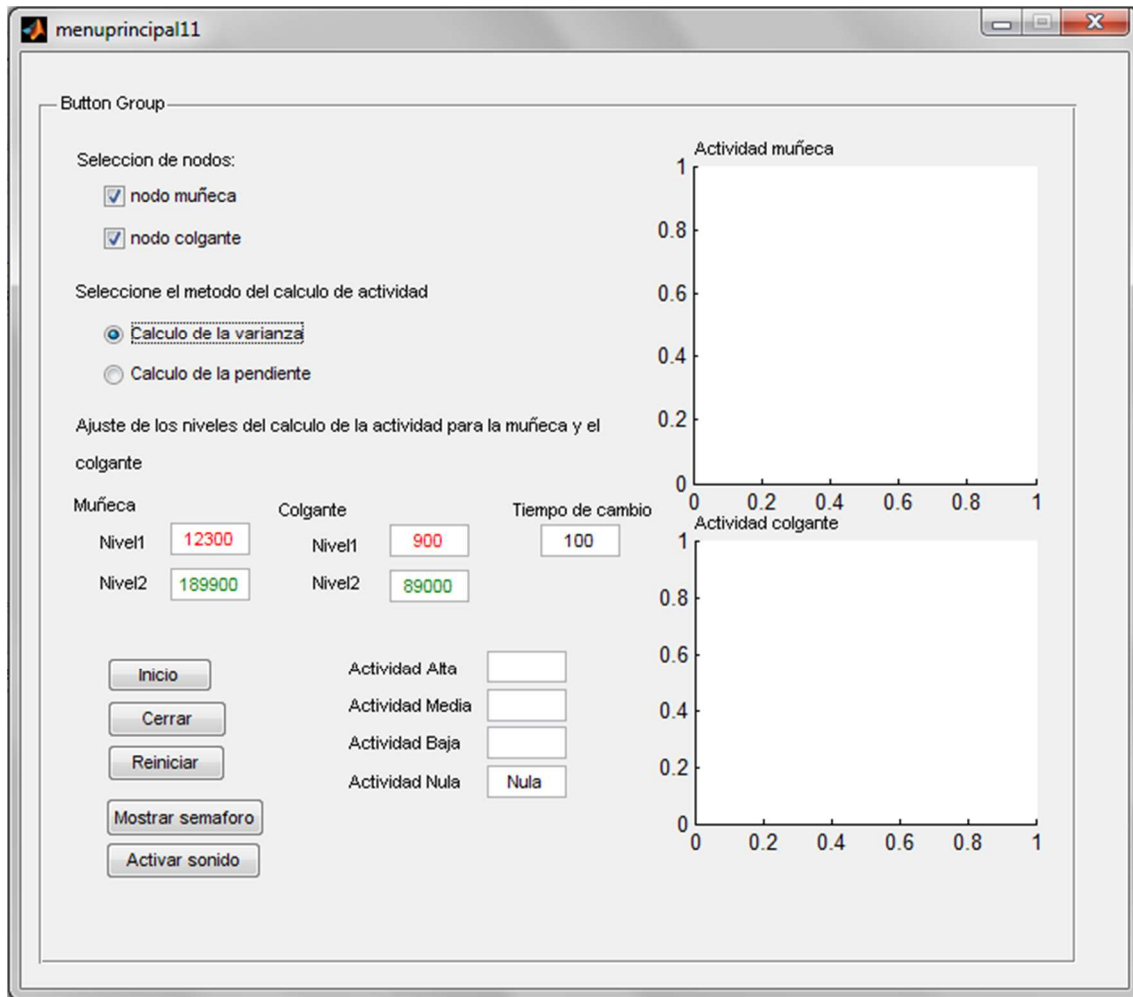


Figura 20 Menú de configuración

Como se puede ver, el programa permitirá seleccionar el número de nodos que va a usar el usuario (el sistema se ha optimizado para trabajar con 2 nodos). Después se seleccionará el sistema de detección (Cálculo de la varianza o Cálculo de la pendiente). Debajo están los umbrales configurables de los niveles de actividad del detector. Una vez escogido el sistema de detección se procederá a iniciar el sistema con el botón "Inicio". Junto con el botón inicio se encuentran los botones de cerrar, reiniciar, y los botones que activan el bio-feedback (Semáforo y Sonido). A la derecha de estos botones está un panel donde se podrá visualizar el estado de actividad de usuario. En la parte derecha del todo podemos ver unas gráficas que se activarán cuando se empiecen a recibir datos de los ECO Nodes. Cada gráfica representará la salida del detector para cada nodo (nodo muñeca y nodo colgante) junto con los umbrales que diferencian cada actividad (nula, baja, media y alta). La gráfica de arriba representará la actividad para el nodo de la muñeca, mientras que la gráfica de abajo representará la actividad para el nodo del colgante.



## Capítulo 5 Conclusiones y perspectivas

El objetivo principal del PFC se ha conseguido: desarrollar un sistema que permite valorar y validar la utilidad del bio-feedback basado en sensores inerciales para el tratamiento de niños con TDAH. Para alcanzar este objetivo se han cumplido una serie de objetivos parciales que se detallan a continuación:

1. Se ha puesto en marcha el ECO Kit con todos sus dispositivos y se ha programado el firmware consiguiendo un streaming de datos desde dos ECO Nodes hasta el PC a través de la Estación Base.
2. Se ha desarrollado un script en Matlab que permite capturar los datos de los ECO Nodes y almacenarlos para poderlos analizar y procesar con posterioridad.
3. Se han realizado pruebas con varios usuarios generando una base de datos de aceleraciones con las que se ha podido perfeccionar el protocolo de comunicación. Además se han probado y ajustado distintos algoritmos para detectar estados de actividad.
4. Se ha programado el software final con un sistema de detección de actividad ajustable para cada usuario. Está preparado para ser usado por psicólogos de la asociación Atenciona en niños con TDAH.

Tras cumplir todos estos objetivos ha quedado a disposición de Tecnodiscap un sistema de bio-feedback basado en sensores inerciales que va a permitir valorar el concepto con usuarios reales en la asociación ATENCIONA. El sistema actual ha sido presentado a los psicólogos de la asociación Atenciona. Les ha parecido una herramienta ajustada a las necesidades expresadas inicialmente y con gran capacidad para el soporte en la enseñanza de los niños de su asociación al autocontrol. Por tanto, esta línea de investigación va a seguir avanzando en otros proyectos finales de carrera, o en trabajos desarrollados por Tecnodiscap.

Como posibles mejoras se plantea implementar el algoritmo de detección de actividad en el mismo nodo, disminuyendo de este modo el envío de información por RF (con el consiguiente ahorro de batería). Para ello sería necesario migrar la tecnología a una plataforma hardware que lo permitiera, como por ejemplo DSP (**D**igital **S**ignal **P**rocessing). Así mismo, correspondería a un estudio posterior (otros PFC) el desarrollo de un sistema más barato y escalable a múltiples usuarios. Para ello se debería implementar este sistema en dispositivos con mayor estandarización. Se ha pensado en dispositivos Zigbee. Esta tecnología permitiría también aumentar el número de usuarios simultáneos.

Este proyecto final de carrera ha durado más tiempo de lo que hubiera sido necesario en un principio (se ha desarrollado en más de un año en total). El principal motivo ha sido la necesidad de simultanear los estudios universitarios y el trabajo. Sin embargo, también otras causas han influido, como los problemas encontrados al trabajar con el Eco Kit. Por ejemplo, errores de programación en algunas librerías suministradas por el fabricante, la falta de soporte por parte del mismo para poner en funcionamiento el Eco Kit, o el encontrar nodos inalámbricos defectuosos. Estos problemas han hecho más tediosa toda la primera parte del desarrollo del proyecto que debería haber sido más ágil y han ocasionado errores que solo se han podido detectar y solventar a mitad del proyecto.

Este proyecto fin de carrera ha resultado ser una gran experiencia personal. El hecho de desarrollar una herramienta utilizable para un fin social (el tratamiento de una enfermedad como el TDAH) da una proyección nueva a los conocimientos obtenidos a lo largo de toda la carrera. Desde un punto de vista de conocimientos académicos adquiridos, debo destacar la experiencia en los siguientes aspectos:

1. Conocimientos de programación de microcontroladores usando el lenguaje de programación C con el programa Eclipse.
2. Conocimientos de programación en Matlab para capturar datos de la Estación Base que envían los sensores y para diseñar un entorno gráfico sencillo para realizar pruebas.
3. Experiencia conseguida en la metodología de redacción de la memoria de un proyecto final de carrera.
4. Experiencia conseguida en la metodología a usar en un trabajo científico.



## Bibliografía

1. **EMBEDDED PLATFORM LAB.** Eco Kit. [Online] [Disponible: 2 15, 2011.] <http://epl.cs.nthu.edu.tw/EcoKit/>.
2. **TECNODISCAP.** Tecnodiscap. [Online] [Disponible: 2 15, 2011.] <http://tecnodiscap.unizar.es/>.
3. **ASOCIACION ATENCIONA.** ATENCIONA. [Online] [Disponible: 01 24, 2011.] <http://www.atenciona.com>.
4. **ASOCIACIÓN ATENCIONA.** ATENCIONA. [Online] [Disponible: 2 15, 2011.] [http://atenciona.es/quienes\\_somos/](http://atenciona.es/quienes_somos/).
5. **FREESCALE SEMICONDUCTORS.** DEMO9S12NE64 Product Summary Page. [Online] [Disponible: 2 15, 2011.] [http://www.freescale.com/webapp/sps/site/prod\\_summary.jsp?code=DEMO9S12NE64](http://www.freescale.com/webapp/sps/site/prod_summary.jsp?code=DEMO9S12NE64).
6. **NORDIC SEMICONDUCTOR.** Datasheet nRF24E1. [Online] [Disponible: 2 15, 2011.] <http://www.nordicsemi.com/index.cfm?obj=product&act=display&pro=79>.
7. **Vega, Dr. Alejandro.** Manual del microcontrolador 8051. [Online] 1999. [Disponible: 2 15, 2011.] [http://www.epsem.upc.edu/~jesusv/uc8051\\_web/man\\_51\\_cast.pdf](http://www.epsem.upc.edu/~jesusv/uc8051_web/man_51_cast.pdf).
8. **Hitachi Metals, Ltd.** Hitachi Metals America, Ltd. [Online] [Disponible: 2 15, 2011.] <http://www.hitachimetals.com/>.
9. **THE ECLIPSE FOUNDATION.** Eclipse - The Eclipse Foundation open source community website. [Online] [Disponible: 2 15, 2011.] <http://www.eclipse.org/>.
10. **SPARKFUN ELECTRONICS.** Transceiver nRF24L01 + Module with RP-SMA - SparkFun Electronics. [Online] [Disponible: 2 15, 2011.] <http://www.sparkfun.com/products/705>.
11. **THE MATHWORKS, INC.** MATLAB - The Language Of Technical Computing. [Online] [Disponible: 2 15, 2011.] <http://www.mathworks.com/products/matlab/>.
12. **XSENS.** Xsens Xbus Kit: Measurement of human motion - Xsens. [Online] [Disponible: 2 15, 2011.] <http://www.xsens.com/en/general/xbus-kit>.
13. **INTERSENSE.** InterSense Incorporated | Precision Motion Tracking Solutions | Home. [Online] [Disponible: 2 15, 2011.] <http://www.intersense.com/>.
14. **ASCENSION TECHNOLOGY CORPORATION.** Ascension Technology Corporation. [Online] [Disponible: 2 15, 2011.] <http://www.ascension-tech.com/realtime/MotionSTARWirelessLITE.php>.
15. **MICROSTRAIN.** G-Link Extended Range Synchronized Wireless Accelerometer Node. [Online] [Disponible: 2 15, 2011.] <http://www.microstrain.com/g-link.aspx>.
16. **NEC-TOKIN.** NEC-TOKIN WEB SITE. [Online] [Disponible: 2 15, 2011.] <http://www.nec-tokin.com/english/>.

17. **TEXAS INSTRUMENTS.** eZ430-Chronos. [Online] [Disponible: 2 15, 2011.]  
<http://www.ti.com/corp/docs/landing/mcu/index.htm?DCMP=MSP430&HQS=Tools+OT+ez430>.
18. **BIBLIOTECA NACIONAL DE MEDICINA DE LOS ESTADOS UNIDOS.** Medline Plus Información de salud para usted. [Online] [Disponible: 2 15, 2011.]  
<http://www.nlm.nih.gov/medlineplus/spanish/ency/article/001551.htm>.
19. **Mariño Acebal, José B. , et al., et al.** *TRATAMIENTO DIGITAL DE SEÑAL. Una introducción experimental.* Barcelona : Edicions UPC, 1999.
20. **FUNDACIÓN CON F.I.A.S.** Fundación Confias. Fundación para una infancia y adolescencia saludables. [Online] [Disponible: 2 15, 2011.] <http://www.fundacionconfias.org/>.
21. **Kleidaria, G., Paspali, A. and Papavasiliou, A.S.** Science Direct. [Online] [Disponible: 2 15, 2011.] <http://www.sciencedirect.com>.
22. **Kropotov, Jury D. , et al., et al.** Science Direct. [Online] [Disponible: 2 15, 2011.]  
<http://www.sciencedirect.com>.
23. **Monstra, Vicent J., Monstra, Donna M. and George, Susan.** The Effects of Stimulant Therapy, EEG Biofeedback, and Parenting Style on the Primary Symptoms of Attention-Deficit/Hyperactivity Disorder. [Online] <http://www.springerlink.com>.
24. **Alhambra, Marabella A. , Fowler, Timothy P. and Alhambra, Antonio A. .** EEG Biofeedback: A New Treatment Option For ADD/ADHD. [Online]  
<http://www.sharperprograms.com>.
25. **Lubar, Joel F., et al., et al.** Evaluation of the effectiveness of EEG neurofeedback training for ADHD in a clinical setting as measured by changes in T.O.V.A. scores, behavioral ratings, and WISC-R performance. [Online] [Disponible: 2 15, 2011.]  
<http://www.springerlink.com/content/l2213885102n16w6/>.

## Figuras

Figura 1 Diagrama de Gantt del desarrollo del proyecto.....	11
Figura 2 Croquis del sistema de bio-feedback.....	15
Figura 3 Esquema lógico del sistema .....	16
Figura 4 Diagrama de bloques del hardware del ECO Node .....	19
Figura 5 Diagrama de bloques del microprocesador nRF24E1.....	20
Figura 6 Diagrama de flujo del ECO Node .....	22
Figura 7 Diagrama de bloques del microcontrolador .....	24
Figura 8 Modulo Sparkfun WRL 00705 .....	24
Figura 9 Diagrama de flujo de la Estación Base .....	25
Figura 10 Trama de datos de la comunicación nodo-base.....	27
Figura 11 Diagrama del proceso de la comunicación entre dos nodos y la base .....	28
Figura 12 Diagrama de flujo del script para el estudio de errores .....	31
Figura 13 Diagrama de flujo para el script de toma de datos en personas.....	33
Figura 14 Diagrama de bloques del script de cálculo de los umbrales óptimos .....	34
Figura 15 Trama de datos Estación Base - PC.....	34
Figura 16 Modelo del filtro de Bessel .....	36
Figura 17 Diagrama de flujo del detector de cálculo de la varianza.....	37
Figura 18 Diagrama del detector del cálculo de la pendiente.....	39
Figura 19 Diagrama de flujo del software del bio-feedback .....	42
Figura 20 Menú de configuración.....	45
Figura 21 Xbus Kit .....	55
Figura 22 Imagen 3D producida por MotionSTAR Wireless LITE .....	56
Figura 23 MotionSTAR Wireless LITE .....	56
Figura 24 G-Link -mXRS Wireless Accelerometer Node.....	56
Figura 25 Dispositivo de Nec-tokin .....	56
Figura 26 eZ430 Chronos.....	57
Figura 27 Prof. Pai H. Chou.....	61
Figura 28 ECO Nodes.....	61
Figura 29 ECO Node con adaptador.....	61
Figura 30 Development Board.....	62
Figura 31 Ethernet Base Statio (Estación Base) .....	62
Figura 32 Multi-ECO Battery Charging Board .....	62
Figura 33 Adpatador de puerto serie.....	63
Figura 34 Development Board en modo Uploader.....	64
Figura 35 Development Board en modo Debugger .....	65
Figura 36 Configuración jumpers para la carga de nodos.....	66
Figura 37 Diagrama de flujo de ECO Node .....	71
Figura 38 Filtro adaptativo .....	149
Figura 39 Filtro predictivo .....	149
Figura 40 Expansión frecuencial en el muestreo .....	151

## Gráficos

Gráfica 1 Respuesta frecuencial del filtro .....	32
Gráfica 2 Umbrales del detector de cálculo de la varianza. Muñeca.....	40
Gráfica 3 Umbrales del detector de cálculo de la varianza. Colgante .....	41
Gráfica 4 Pérdidas relativas en función de la distancia para 1 ECO Node .....	129
Gráfica 5 Grafica del nivel de actividad.....	140
Gráfica 6 Tasa de error en función del umbral.....	141
Gráfica 7 Detección por cálculo de la varianza. Muñeca umbral 1.....	142
Gráfica 8 Detección por cálculo de la varianza. Muñeca umbral 2.....	142
Gráfica 9 Detección por cálculo de la varianza. Colgante umbral 1 .....	143
Gráfica 10 Detección por cálculo de la varianza. Colgante umbral 2 .....	143
Gráfica 11 Detección por cálculo de la pendiente. Muñeca umbral 1.....	144
Gráfica 12 Detección por cálculo de la pendiente. Muñeca umbral 2.....	144
Gráfica 13 Detección por cálculo de pendiente. Colgante umbral 1 .....	145
Gráfica 14 Detección por cálculo de pendiente. Colgante umbral 2 .....	145
Gráfica 15 N° de errores en ráfagas de errores en recepción.....	147
Gráfica 16 Ejemplo de reconstrucción de señal con interpolación lineal .....	148
Gráfica 17 Ejemplo filtro predictivo para una actividad baja .....	150

## Tablas

Tabla 1 Tabla de actividades de los nodos .....	43
Tabla 2 Pérdidas relativas en función de la distancia para 1 ECO Node a 40Hz.....	127
Tabla 3 Pérdidas relativas en función de la distancia para 1 ECO Node 20Hz .....	128
Tabla 4 Pérdidas relativas en función de la distancia para 1 ECO Node 10Hz .....	128
Tabla 5 Pérdidas relativas en transmisión simultánea de 2 ECO Nodes sin control de acceso al medio. Nodo 4.....	129
Tabla 6 Pérdidas relativas en transmisión simultánea de 2 ECO Nodes sin control de acceso al medio. Nodo 5 .....	130
Tabla 7 Pérdidas relativas en transmisión no simultánea de 2 ECO Nodes con control de acceso al medio en diferente canal. Nodo 4 .....	131
Tabla 8 Pérdidas relativas en transmisión no simultánea de 2 ECO Nodes con control de acceso al medio en diferente canal. Nodo 5 .....	131
Tabla 9 Pérdidas relativas en transmisión no simultánea de 2 ECO Nodes con control de acceso al medio en tres canales diferentes. Nodo 4.....	132
Tabla 10 Pérdidas relativas en transmisión no simultánea de 2 ECO Nodes con control de acceso al medio en tres canales diferentes. Nodo 5 .....	133
Tabla 11 Pérdidas relativas en transmisión no simultánea de 2 ECO Nodes con control de acceso al medio en un canal. Nodo 4.....	134
Tabla 12 Pérdidas relativas en transmisión no simultánea de 2 ECO Nodes con control de acceso al medio en un canal. Nodo 5.....	134
Tabla 13 Pérdidas relativas en transmisión no simultánea de 3 ECO Nodes con control de acceso al medio en un canal. Nodo 1.....	135

Tabla 14 Pérdidas relativas en transmisión no simultánea de 3 ECO Nodes con control de acceso al medio en un canal. Nodo 4.....	135
Tabla 15 Pérdidas relativas en transmisión no simultánea de 3 ECO Nodes con control de acceso al medio en un canal. Nodo 5.....	135
Tabla 16 Datos de los voluntarios.....	137
Tabla 17 Datos de los acelerómetros.....	139

