

# DISEÑO BIOINSPIRADO DE UN SISTEMA PARA EL CONTROL Y PROCESADO DE SEÑALES PROVENIENTES DE SENSORES INTELIGENTES

Trabajo de Fin del Master Oficial de Automática,  
Robótica y Telemática  
Curso 2006-2007



*ciber-bbn*



Alumno: Miguel Ángel Estudillo Valderrama  
Tutora: Laura María Roa Romero



# ÍNDICE GENERAL

Resumen .....	7
1.- Introducción .....	8
1.1.- Estado del Arte de los Sistemas de Teleasistencia.....	8
1.2.- Estado del Arte de la monitorización del movimiento y detección de caídas....	13
1.2.1.- Sistemas no basados en acelerometría .....	16
1.2.2.- Sistemas basados en acelerometría .....	17
1.3.- Nuevas tendencias de control. Evolución .....	21
2.- Descripción del Sistema .....	25
3.- Objetivos .....	27
4.- Resultados .....	28
4.1.- Arquitectura Hardware.....	28
4.1.1.- Diagrama de bloques de la arquitectura Hardware .....	28
4.1.2.- DSP .....	30
4.1.2.1.- Multiplexación de pines de los periféricos del DSP .....	33
4.1.2.2.- Esquemático del DSP .....	35
4.1.3.- Transceptor Zigbee.....	35
4.1.3.1.- Interfaz con el DSP .....	39
4.1.3.2.- Esquemático del módulo Zigbee .....	40
4.1.4.- Transceptor Bluetooth.....	40
4.1.4.1.- Interfaz con el DSP .....	42
4.1.4.2.- Esquemático del módulo transceptor .....	43
4.1.5.- Transceptor Infrarrojo .....	43
4.1.5.1.- Interfaz con el DSP .....	44
4.1.5.2.- Esquemático del transceptor infrarrojo .....	45
4.1.6.- Memoria Flash .....	45
4.1.6.1.- Interfaz con el DSP .....	47
4.1.6.2.- Esquemático de la memoria Flash.....	48
4.1.7.- Display LCD .....	48
4.1.7.1.- Interfaz con el DSP .....	51
4.1.7.2.- Esquemático del módulo LCD .....	51
4.1.8.- La Botonera.....	51
4.1.8.1.- Interfaz con el DSP .....	52
4.1.8.2.- Esquemático de la botonera .....	52
4.1.9.- Módulo Audio .....	53
4.1.10.- Módulo de Control y Supervisión .....	57
4.1.10.1.- MAX6374 de MAXIM .....	57
4.1.10.2.- BQ27200 de TI.....	58
4.1.10.3.- BQ24010 de TI.....	59
4.1.10.4.- Generación del reset del sistema .....	60
4.1.11.- Otros Aspectos del Diseño Hardware .....	61
4.1.11.1.- Colocación de los componentes en la placa.....	61
4.1.11.2.- Interferencias, masas y aspectos relacionados .....	63
4.1.11.3.- Estado actual del dispositivo. Pasos a seguir .....	64

---

4.2.- Arquitectura Software .....	66
4.2.1.- Materiales y Herramientas de Desarrollo Empleadas .....	67
4.2.1.1.- Kit de Desarrollo C6713DSK .....	67
4.2.1.2.- Code Composer Studio .....	69
4.2.1.3.- Toolboxes de Matlab.....	70
4.2.2.- Aspectos de Diseño .....	72
4.2.3.- Programación en hilos en el DSP.....	75
4.2.4.- Módulo de Ahorro Energético .....	78
4.2.5.- Módulo de Salvaguarda.....	80
4.2.6.- Módulo SPI .....	81
4.2.7.- Módulo I2C .....	85
4.2.8.- Módulo SelfTest.....	89
4.2.9.- Módulo Audio .....	91
4.2.10.- Otros Aspectos del Diseño Software.....	92
4.2.10.1.- Comunicación del DSK con Matlab 7.0 .....	92
4.2.10.2.- Estado actual del dispositivo. Pasos a seguir en el diseño software ....	93
4.3.- Modos de Funcionamiento y Consumo.....	94
4.3.1.- Modos de funcionamiento.....	94
4.3.2.- Consumo .....	95
4.3.3.- Conclusiones .....	97
5.- Conclusiones .....	98
6.- Publicaciones relacionadas con el trabajo.....	99
7.1.- Publicaciones en las que participa el autor.....	99
7.2.- Otras publicaciones del Grupo .....	99
7.- Bibliografía .....	101
Nomenclatura – Acrónimos.....	109

## ÍNDICE de FIGURAS

Figura 2.1.- Monitor de movimiento humano .....	25
Figura 4.1.- Diagrama de bloques del HW del PSE .....	28
Figura 4.2.- Esquemático del DSP .....	35
Figura 4.3.- Módulo de Evaluación del CC2430.....	37
Figura 4.4.- Configuración usual SPI .....	38
Figura 4.5.- Buffers circulares SPI .....	38
Figura 4.6.- Esquemático de los conectores del módulo Zigbee .....	40
Figura 4.7.- Módulo de evaluación del transceptor Bluetooth .....	42
Figura 4.8.- Esquemático de los conectores del módulo Bluetooth .....	43
Figura 4.9.- Esquemático del transceptor infrarrojo.....	45
Figura 4.10.- Esquemático de la memoria Flash .....	48
Figura 4.11.- Esquema i2c de un maestro. ....	50
Figura 4.12.- Esquemático del memoria módulo LCD .....	51
Figura 4.13.- Esquemático botón luminoso.....	53
Figura 4.14.- Esquemático botón no luminoso.....	53
Figura 4.15.- Generación Interrupción Botón Pulsado.....	53
Figura 4.16.- Esquemático del CODEC .....	55
Figura 4.17.- Esquemático del Amplificador de Audio .....	56
Figura 4.18.- Esquemático del Watchdog .....	58
Figura 4.19.- Esquemático del Gas Sauge.....	59
Figura 4.20.- Esquemático del Cargador de Batería.....	60
Figura 4.21.- Esquemático del Reset del Sistema .....	60
Figura 4.22.- Colocación de los componentes en la placa.....	62
Figura 4.23.- Placa PCB del PSE .....	64
Figura 4.24.- Kit de Desarrollo C6713DSK.....	68
Figura 4.25.-.- Diagrama de bloques del DSK6713 .....	69
Figura 4.26. Entorno de Desarrollo Code Composer Studio.....	70
Figura 4.27.- Paquetes de Matlab .....	71
Figura 4.28.- Diagrama de flujo principal .....	72
Figura 4.29.- Diagrama de flujo del módulo salvaguarda.....	73
Figura 4.30.- Funciones TSK involucradas en la gestión de las tareas TSK.....	77
Figura 4.31.- Configuración SPI del McBSP como maestro.....	82
Figura 4.32.- Transferencia SPI con CLKSTP = 10b.....	83
Figura 4.33.- Transferencia SPI con CLKSTP = 11b.....	83
Figura 4.34.- Capturas SPI a 6MHz .....	85
Figura 4.35.- Diagrama de bloques del módulo I2C .....	87
Figura 4.36.- Capturas I2C a 6.25KHz.....	89
Figura 4.37.- Dimensiones mecánicas de la batería Vizhuel.....	97

## ÍNDICE de TABLAS

Tabla 4.1.-Tabla de la verdad para SPI_CS#ZIGBEE .....	40
Tabla 4.2.-Tabla de la verdad para SPI_CS#BLUETOOTH.....	43
Tabla 4.3.-Tabla de la verdad para FLASH_CE# .....	47
Tabla 4.4.- Módulos software del PSE.....	74
Tabla 4.4.- Tabla de la verdad para .....	55
Tabla 4.5.- Comparativa de Hilos.....	77
Tabla 4.6.- Modos de Parada de Reloj SPI.....	82
Tabla 4.7.-: Valores de los Registros para una comunicación SPI a 6MHz.....	85
Tabla 4.8.- Consumo del DSP .....	96
Tabla 4.9.- Batería seleccionada.....	96

## Resumen

Se presenta la arquitectura hardware y software de un prototipo de servidor personal (PSE) para el control y procesado de señales provenientes de sensores inteligentes. Este dispositivo está diseñado en el contexto de un sistema de teleasistencia domiciliaria para personas mayores que pretende la monitorización del movimiento, estado postural y actividad física de humanos durante las 24 horas así como detectar posibles caídas. El PSE forma parte, junto con otros dispositivos, de una red de área personal inalámbrica (WPAN) que permite la completa movilidad del sujeto monitorizado tanto en su domicilio como fuera de él, y la comunicación con un centro de gestión que hace posible el seguimiento de numerosos pacientes.

Una de las características principales del dispositivo es su capacidad de personalización en línea al paciente mediante la generación de conocimiento adaptativo en tiempo real. En concreto para el caso de monitorización del movimiento humano, la captura y determinación automática y en tiempo real de parámetros cinéticos y posturales permite un mejor conocimiento y diagnóstico de las causas del deterioro motor y del aumento del porcentaje de caídas en el grupo poblacional de la tercera edad. A partir de dicho conocimiento se pueden diseñar tratamientos que analicen estas causas y que atiendan las necesidades de cada sujeto de forma personalizada en lo que denominaremos adaptación del dispositivo al usuario y al entorno.

Cabe señalarse sin embargo el carácter multimodal del monitor inteligente en la medida que va a ser capaz de gestionar y procesar señales heterogéneas de dispositivos de muy diversa naturaleza, desde las que proporcionan los sensores biomédicos clásicos (ECG, EEG, etc.) hasta sonidos vocales, como es el caso de un dispositivo protésico-terapéutico para la tartamudez basado en la alteración de la realimentación auditiva, de forma adaptativa, cuya propuesta podemos ver en [0].

En el siguiente capítulo se hará un breve análisis de las necesidades de monitorización en tiempo real centrándonos en el movimiento humano. En este contexto se introduce y describe como resultado el sistema desarrollado por el Grupo de Ingeniería Biomédica particularizando para el PSE, en el que se distinguirá un análisis tecnológico hardware y software de la solución propuesta. También se presenta la evaluación en términos de autonomía del dispositivo asociado a los distintos modos de funcionamiento establecidos.

## **1.- Introducción**

En este apartado se va a realizar un análisis de las necesidades de la monitorización en tiempo real de personas.

Empezaremos con un estudio del estado de arte de los sistemas de teleasistencia. A continuación presentaremos un análisis del estado de las cosas de la monitorización del movimiento humano y la detección de caídas. Finalmente haremos un pequeño repaso a las nuevas tendencias de control y su evolución, entre las que se enmarca este trabajo en tanto que se hace un control de una red de sensores inteligentes. Un ejemplo de esta red lo encontramos en el monitor de movimiento desarrollado y patentado por el Grupo de Ingeniería Biomédica (GIB) de la Universidad de Sevilla [1].

### **1.1.- Estado del Arte de los Sistemas de Teleasistencia**

#### **Motivación**

El interés por investigar las características de los cuidados a domicilio es grande desde hace tiempo y sigue en aumento, y se centra más en el análisis de los Cuidados de Enfermería a Domicilio y en los Servicios de Ayuda a Domicilio, que en otros aspectos como la asistencia médica, la terapia ocupacional o la fisioterapia en domicilio [2]. De entre los motivos que más interesan para su estudio destacan por su importancia el envejecimiento progresivo de la población [3,4], el cambio en la estructura de las familias [5,6] y la saturación de los hospitales [7].

España contrasta con otros países del entorno de la Unión Europea en que un 8 % de la población recibe asistencia a domicilio de enfermería y sólo un 0.18% de los Servicios Sociales (la más baja de los estudios) [8]. A pesar de la tendencia indudable de intentar dar asistencia a una mayor población podemos asegurar que en la actualidad los Servicios de Atención en Domicilio no están todavía suficientemente establecidos ni estructurados en nuestro país.

De entre los nuevos métodos que pretenden resolver estas deficiencias sanitarias destacan ante todo las aplicaciones telemáticas, en el sentido de que son capaces de recoger y enviar los datos del paciente desde el domicilio a un centro de control de datos biomédicos, lo cual contribuye indudablemente a la reducción de la carga asistencial del personal sanitario, ayudan a la eficacia y economía en la gestión de los centros

asistenciales, reduce la carga del cuidador, favorece la implicación y participación en los cuidados de las personas más cercanas, y todo ello en el entorno natural de la persona mayor: su casa, con la consiguiente mejora de calidad de vida.

Para alcanzar los objetivos anteriores es preciso acudir a soluciones tecnológicas que favorezcan la teleasistencia domiciliaria, mediante la sustitución de consultas presenciales por consultas virtuales, facilidades de videoconferencia y aplicaciones, en general, de las tecnologías de la información y comunicaciones.

A la vista de la situación descrita, la primera pregunta que cabe plantearse es si la teleasistencia domiciliaria es una solución eficaz en cuanto a los costes que supondría. Para abordar esta cuestión, es conveniente recordar que los costes sanitarios basados en la edad indican que las personas mayores son los principales consumidores, y que los gastos asociados a personas de 75 años o más superan en seis veces a los de la población media. Según estudios estadísticos realizados por el Instituto de Mayores y Servicios Sociales del 2004 [9], el número de habitantes en España es de 42.717.064 según datos del padrón municipal a 1 de Enero del 2003. En el año 2003 había contabilizadas en España 7.276.620 personas mayores de 65 años, con un crecimiento de más de 400.000 individuos en relación con los datos del 2002. Como se refleja en estos datos, las personas mayores de edad ya representan el 17,0% de toda la población, y ese porcentaje y sus cifras absolutas no van a dejar de crecer.

Para el caso andaluz se tiene estadísticas actualizadas [10] que indican que en nuestra comunidad autónoma las personas mayores representan ya el 15% de la población, pero el aumento progresivo de la esperanza de vida hace que las estimaciones apunten que en 2050 serán más del 30%. Tomando como cálculo el coste de las plazas en residencias y unidades de estancia diurna, el sector mueve cada año en Andalucía unos 680 millones de euros. Se justifica por tanto el gran interés suscitado por la investigación en el contexto de la teleasistencia sanitaria.

### **Evolución de los sistemas de teleasistencia**

Desde el punto de vista tecnológico cabe señalarse que se ha hablado de tres generaciones en la teleasistencia domiciliaria a personas mayores [11]. La primera generación se basó en la utilización de dispositivos que facilitaban la llamada telefónica a un centro de atención domiciliaria.

En cambio en los últimos años destacan de entre las soluciones que se están investigando aquellas que transmiten señales analógicas, digitales y vídeo en tiempo

real a través de Internet [12-13] utilizando normalmente el lenguaje HTML como interfaz de usuario médico. La gran ventaja de Internet es que ofrece una clara alternativa de bajo coste para la transferencia de datos [14] teniendo en cuenta que está universalmente extendido. Destacar en este ámbito la aportación que realizó el GIB [15] al presentar una solución eficiente y de bajo coste para la clasificación de pacientes susceptibles de tratamiento con cirugía menor ambulatoria en modalidad de acto único. Otro tipo de soluciones de sistemas de monitorización en el entorno del usuario teleasistido son los que emplean redes de comunicaciones basadas en líneas de potencia [16], inalámbricas [17,18] o bus de campo [19]. Son de interés también los que se basan en cámaras de vídeo controladas remotamente [20].

En la actualidad cabe señalarse que el modelo de interacción paciente-médico clásicamente representado por la teleconsulta y considerado el paradigma de la telemedicina durante la década anterior y finales de los '80, se está viendo superado por las necesidades del nuevo contexto sanitario. Los avances en técnicas diagnósticas y terapéuticas y las mejores condiciones socio-sanitarias de la población tratan de evitar la cada vez mayor incidencia de patologías crónicas dado el progresivo envejecimiento de la población que, en muchas ocasiones, puede padecer una o varias de estas patologías. Los beneficios que la telemedicina, y en general las Tecnologías de Información Sanitarias (TIS's), puede aportar en la atención a este tipo de pacientes pretende superar el modelo clásico centralizado [21]. Como consecuencia, este modelo se está desplazando hacia una interacción más compleja en la que la atención al paciente no recae en manos de un único facultativo y en la que además participan nuevos actores, como pueden ser enfermeros, cuidadores, etc. Esto dará lugar a la consideración de otras fuentes de información biológica relevante y conocimiento clínico general además de la historia clínica electrónica (HCE).

El Grupo de Ingeniería Biomédica ha mostrado recientemente [22] las ventajas de una metodología ligada al paradigma de teleasistencia basada en la generación de conocimiento, mediante la que las TIS son capaces de proporcionar conocimiento biomédico adaptativo y personalizado para un paciente de una forma que no puede llevarse a cabo por los modelos de telecuidado actuales, con el consiguiente aumento de su calidad de vida. La idea y principal innovación de este paradigma sanitario es la capacidad de crear conocimiento personalizado y en tiempo real como contrapartida a otros dispositivos de monitorización que realizan un procesamiento off-line. Esta

metodología aporta ventajas, por ejemplo, en la detección de caídas [23]. Se emplea una arquitectura de procesamiento multicapa y distribuida, donde la primera capa está definida por un conjunto de sensores inteligentes que envían los datos capturados y procesados hacia la segunda capa, la cual crea una imagen computacional del estado del paciente, centrada en el dominio biomédico deseado y que es procesada por un conjunto de subsistemas distribuidos.

### **Aspectos de diseño. Líneas de investigación**

Los principales aspectos de diseño de los sistemas de teleasistencia actuales se resumen en arquitecturas abiertas, lenguajes de programación modulares y orientados al objeto, con extensiones para ser ejecutados en tiempo real, soporte de diferentes interfaces de comunicaciones (TCP/IP, Ethernet, USB, etc.) e integración “plug-and-play” de diferentes componentes [24-26].

Uno de los aspectos clave en el diseño de cualquier implementación que permita la teleasistencia domiciliaria es que sea de fácil manejo. Dado que en la mayor parte de las ocasiones será empleado por personas mayores, se pretende minimizar el número de necesidades operacionales para aprender a manejar los equipos, y como consecuencia mejorar su sensación de comodidad y seguridad cuando los empleen en sus propios hogares [27].

De entre las líneas de investigación asociadas a los sistemas de teleasistencia caben señalarse por un lado el estudio e integración de las diversas tecnologías de red inalámbricas que den soporte a una red de área personal inalámbrica (WPAN), en el sentido de la convergencia que se debe producir durante la siguiente década entre dichas redes inalámbricas e Internet [28].

Por otro lado un aspecto fundamental sujeto a investigación en estos sistemas es el de la seguridad y protección de datos, que se traduce en el desarrollo de numerosos algoritmos de encriptación [29] o en la definición de arquitecturas de red nuevas con mejoras de seguridad basadas en recubrimientos virtuales [30].

También cabe citarse la gran labor de investigación realizada en relación con la estandarización de la información médica. Es de gran importancia establecer un modelo de información que se integre plenamente dentro del esfuerzo actual por conseguir la máxima interoperatividad de sistemas dada la vasta heterogeneidad de información médica disponible. La idea es que el modelo de información asociado al sistema de teleasistencia sea compatible con los estándares de HCE, fundamentales para la gestión

integral de la información del paciente, entre los que destaca la norma prEN13606 del TC251 del CEN en Europa [31] y HL7 [32] para los EEUU. Es de justicia señalar también la iniciativa no lucrativa de openEHR [33]. Para profundizar más en este aspecto es muy recomendable la referencia [34].

Por lo que respecta a los sensores biomédicos, la tendencia que se está siguiendo es la de monitorización mínimamente invasiva, con sensores integrados en el entorno de la persona mayor [35], que permitan disponer de los datos en el momento necesario, con el mínimo nivel de intrusión percibido posible [36-37]. Caben destacarse como ejemplos los monitores de frecuencia cardíaca, sensores de temperatura nasal/oral, electromiograma, electrocardiograma superficial, mecanismos de control respiratorio [38], y medida de glucosa mediante análisis de la saliva [39], entre otros. Con frecuencia los biosensores están adheridos a pulseras que se pueden transportar cómodamente para no provocar ansiedad a la persona mayor durante la captura de datos [40]. Hay casos en los que puede interesar la medida de signos vitales y fisiológicos sin que los sensores biológicos estén en contacto con el cuerpo durante la medida. Por ejemplo, en [41] se presenta un sistema para la medición de la frecuencia cardíaca y actividad respiratoria mediante las fluctuaciones del nivel del agua en reposo durante el baño. Otro ejemplo lo tenemos en [42] donde se describen sensores ECG con electrodos que se conectan a una bañera o electrodos textiles que sirven para medir el ECG en cama del paciente. En el caso de las personas mayores, uno de los problemas más graves son las caídas, como consecuencia de la pérdida de capacidad motora, que suponen un grave riesgo, sobre todo cuando se encuentran solas. Los detectores de caídas y dispositivos de monitorización del movimiento constituyen por tanto una importante ayuda para la determinación de estas situaciones de emergencia [43-44] y por ello se comentarán con más detalle en el siguiente apartado de estudio del estado del arte.

Ora línea de investigación trata de las técnicas de procesamiento digital de señales, y como ejemplo paradigmático el electrocardiograma (ECG). Éste es uno de los signos vitales de mayor importancia a transmitir en situaciones de emergencia [45], puesto que revela información diagnóstica muy importante sobre desórdenes cardiovasculares. Otro ejemplo, que es en el que se enmarca este trabajo, es el del procesado de señales acelerométricas correspondientes a un monitor de movimiento humano que permita la detección de caídas o el cálculo de índices de actividad metabólica que pueden ser

usados en conjunción con modelos fisiológicos u otros algoritmos, por ejemplo como ayuda a la prescripción de insulina en caso que el paciente sufra de diabetes.

Una última línea de investigación, no por ello menos importante, trata de las restricciones del procesamiento en tiempo real. Para el caso de implementación de un clasificador de movimiento para aplicaciones en tiempo real [23] dice que se deben hacer varias consideraciones para satisfacer las siguientes restricciones:

1. No existe conocimiento de los eventos futuros.
2. La cantidad de datos que puede ser almacenada es limitada (es suficiente con 1 segundo de muestras de datos).
3. La cantidad de tiempo disponible es limitada – debe ir al mismo ritmo que el de la tasa de adquisición de datos.

La primera restricción implica que los métodos de clasificación de movimiento que involucran futuros estados de movimiento no pueden desarrollarse. Por ejemplo, la clasificación de un transición “sentado-de pie” depende del conocimiento tanto del estado de “sentado” anterior tanto como del siguiente estado de “de pie”, para asegurar que se ha realizado una decisión precisa. De esta forma, el algoritmo embebido está inherentemente limitado en su habilidad para reconocer movimientos en los que estados de movimiento subsiguientes son esenciales para una clasificación precisa.

## ***1.2.- Estado del Arte de la monitorización del movimiento y detección de caídas***

Un interesante punto de partida para centrar la naturaleza del sistema que se desee desarrollar consiste en seguir la clasificación de sistemas para la detección del movimiento humano que propuso Mulder [46]. Por consiguiente, el dispositivo a implementar puede ser considerado un sistema “inside-in”, en la medida en que tanto sensor como fuente están en el cuerpo. Por el contrario los sistemas que se mencionarán a partir de ahora son “outside-in” (sensor externo y fuente en el cuerpo), dada su importancia en al campo de la predicción de caídas de las personas mayores.

Las caídas de personas en el sector de la tercera edad es un problema serio pero que aún no ha sido suficientemente estudiado, y como prueba tenemos el caso español [47]. Las caídas pueden producir desde leves deterioros funcionales, hasta tener consecuencias

más graves como las temidas fracturas de cadera o incluso fatales como la muerte. La reducción en la calidad de vida que dichas caídas provocan en las personas mayores, así como el coste sanitario asociado a su atención hospitalaria, motivan el estudio de los factores de riesgo y las causas que las provocan. Diversos estudios han analizado la casuística relacionada con la fenomenología de las caídas [47] [48][49][50][51][52][53] desde numerosos puntos de vista. A continuación enumeramos brevemente los aspectos más destacados:

- Lugar donde se producen las caídas.
- Frecuencia de las caídas.
- Dependencia con el sexo del sujeto.
- Factores de riesgo.
- Consecuencias de las caídas.
- Intervenciones para reducir el riesgo de caídas entre personas mayores.
- Miedo a caer entre personas mayores y factores psicológicos.

Los aspectos tratados dan muestra de la complejidad de la problemática de las caídas en tanto que existen factores de distinta naturaleza relacionados con la disminución de la calidad de vida producida por éstas. Por ello es esencial considerar en el diseño de un detector que vaya a ser portado por personas con mayor riesgo de sufrir caídas, y particularmente en aquellas que no quieren parecer desvalidas, el factor de *discreción* del aparato tanto en lo referente a su tamaño como a su posición en el cuerpo del sujeto monitorizado.

En [54] se establecen los riesgos y daños en las caídas basándose en otros estudios. Estos riesgos son enumerados como sigue:

1. Edad.
2. Sexo.
3. Movilidad.
4. Actividades de la vida diaria.
5. Historial de caídas previas.
6. Visión pobre.
7. Medicación.
8. Cognición y demencia.

Hacer notar como conclusión que no solo influyen factores fisiológicos en el aumento del riesgo de caídas sino que lo factores psicológicos juegan también un papel crucial.

La pérdida de confianza es determinante y puede incluso generar pánico a realizar diversos tipos de actividades por el miedo a una posible caída y a la falta de asistencia adecuada cuando esto ocurre.

Cabe señalarse que todavía no se ha realizado un estudio completo de las causas que dan origen a las caídas y de la relación que existe entre ellas. Se habla de factores de riesgo, como puede ser la pérdida de autonomía o el mismo miedo que las caídas producen pero no se ha descrito convenientemente la problemática desde un punto de vista etiológico. Brevemente vamos a comentar algunos aspectos desde esta perspectiva que sin duda dan forma a la definición de la monitorización de caídas a implementar.

Para empezar es conveniente señalar que aunque los aspectos del equilibrio humano que más se deterioran con la edad no están del todo identificados, está claro que al menos indirectamente interviene un deterioro del control motor relacionado con el equilibrio, i.e., no se trata solo de un problema de pérdida de capacidad muscular y flexibilidad. El estudio del equilibrio humano viene siendo objeto de estudio desde hace tiempo y es un aspecto importante en el campo de la quinesología<sup>1</sup>. [55] establece los elementos que intervienen en dicho equilibrio y su mantenimiento:

- Sistema músculo-esquelético.
- Coordinación motora.
- Percepción de la orientación.
- Organización sensorial.
- Control central predictivo.
- Adaptación al entorno.

Existen numerosas pruebas de medición para encontrar patrones de diferencias repetibles entre personas con desórdenes en el equilibrio y las que no lo tienen, destacándose dos modelos. El primero versa sobre los llamados análisis posturográficos, que miden parámetros relacionados al mantenimiento de ciertas posturas simples. El segundo se centra en el análisis de los modos de andar, más ligados en principio al reconocimiento de parámetros dinámicos. Entre los índices asociados con este segundo modelo, destacar [56] que evalúa la consecución de posturas fundamentales para el mantenimiento del equilibrio estable. Respecto al primer modelo, varios trabajos

---

<sup>1</sup> Estudio del movimiento humano.

señalan diferencias en los patrones cinemáticas entre las personas mayores que sufren caídas y las que no [57][58]. En relación con esto existen algunos patrones cinemáticas muy importantes correspondientes a los programas de coordinación motora centra para el mantenimiento del equilibrio, como son la estrategia del tobillo, la de la cadera y la del andado [59].

A continuación haremos un resumen de las soluciones tecnológicas aportadas en el contexto de la monitorización del movimiento y detección de caídas, distinguiendo entre sistemas basados o no en acelerometría.

### **1.2.1.- Sistemas no basados en acelerometría**

La mayor parte de estos sistemas se encuentran ya disponibles en el mercado al haber alcanzado la fase comercial. Sin embargo todos adolecen de una clara falta de interoperabilidad que origina importantes limitaciones en su funcionalidad.

Veamos una muestra de los sistemas más conocidos.

- Medallón o reloj de pulsera: con un botón que se activa por el sujeto en caso de accidente provocando una conexión telefónica con el centro de teleasistencia. Es el sistema más extendido en la actualidad. Sin duda se trata de un sistema portable que aumenta la autonomía del individuo. Sin embargo, presenta el inconveniente principal de que es un sistema de alarma no automático por lo que el individuo víctima de la caída debe estar consciente para alertar de la misma.
- Detector de actividad: Se pretende detectar la actividad inusual de la persona en algún lugar del hogar, y a partir de ella determinar la existencia de una posible caída, mediante detectores de presencia tales como sensores infrarrojos piroelectrónicos [60] o detectores de consumo eléctrico [61], es posible. No son sistemas autónomos ni portables.
- Detectores de impacto piezoeléctricos y detectores de inclinación [62][63]: El cálculo de la inclinación del sujeto de estudio es claramente el método más sencillo para determinar que se ha producido una caída. Estos sistemas se implementan con dispositivos electrónicos muy simples que lo único que detectan es si el individuo yace en posición horizontal o no. Esta información, junto con el conocimiento de las costumbres diarias del sujeto, permiten la estimación de posibles caídas. La fiabilidad del sistema puede mejorarse al añadir un sensor de impacto al de inclinación. Estos dispositivos pueden realizar

monitorización on-line del sujeto durante gran parte del día, siempre que éste se mantenga en entornos reducidos, como puede ser el hogar. Éste sin duda es su mayor inconveniente, al no permitir monitorización fuera de entornos conocidos. Son pequeños y pueden portarse en un cinturón.

- Análisis basados en frecuencias acústicas: Las caídas se detectan al analizar por las componentes frecuenciales de las vibraciones que causan el impacto. No es un sistema portable.
- Análisis basados en video [64]: se va a monitorizar los movimientos del paciente en tiempo real por medio de canales de vídeo. Presenta como grandes desventajas que no es un sistema autónomo, ya que necesita una infraestructura de video lo que limita la detección a zonas concretas y además no es un sistema portable.

### **1.2.2.- Sistemas basados en acelerometría**

De entre los sensores expuestos los acelerómetros son los que más ventajas presentan en tanto que responden a la frecuencia y a la intensidad del movimiento, algunos tipos se pueden usar para medir la inclinación y el movimiento del cuerpo, y el avance técnico en los sistemas MEMS ha hecho posible la existencia de acelerómetros comerciales en miniatura, baratos y fiables. La gran ventaja de los sistemas basados en acelerometría es que van a proporcionarnos datos sobre el gasto energético asociado a la actividad física del paciente.

En [65] se implementa un sistema de telemonitorización del comportamiento humano durante la vida diaria del paciente en su hogar con el propósito de almacenar los datos de la actividad normal para un posterior diagnóstico y/o exploración médica. Combinan un detector de caída con un conjunto de sensores de posición. Los datos adquiridos los transmiten a través de una red inalámbrica y son recogidos por un PC sobre el que corre LabView. Aunque se trata de un sistema capaz de monitorizar la actividad de un individuo y detectar caídas, sin duda, no proporciona monitorización on-line durante las 24 horas, esto es, no se trata de un sistema portable siendo este uno de los requisitos más importantes que buscamos para nuestra aplicación. En este caso tratamos simplemente con una herramienta de diagnóstico.

[66] tiene como objetivo principal evaluar los patrones de caminar justo antes de que se produzca una caída en pacientes de alto riesgo en situaciones de rehabilitación y en concreto se centra en pacientes con hemiplejia. En este estudio se usa un acelerómetro triaxial que es fijado a la cintura del sujeto. Los datos acelerométricos son muestreados a 200Hz y analizados usando la Transformada Wavelet Discreta (DWT). Esta herramienta matemática es poderosa y dota de grandes prestaciones al algoritmo de detección, sin embargo dificulta las posibilidades de monitorización on-line en tiempo real. De nuevo la monitorización es realizada off-line, por lo que nos volvemos a encontrar con una herramienta de diagnóstico en lugar de un sistema de teleasistencia.

Otro artículo que hace uso de transformada wavelet es en [67], donde se estudia la capacidad de esta herramienta para evaluar el riesgo de caídas. Otra herramienta que está empezando a usarse en nuestro campo son las máquinas de soporte vectorial para la detección, como se indica en [68].

En [69] se mantiene la línea de investigación seguida por el autor y ya presentada en [65]. La novedad radica en el la mejora de la eficiencia de la detección de caídas al interpretarlas como una sucesión de transiciones posturales. En esta ocasión se presenta un análisis de diversas situaciones y como el dispositivo es capaz de reconocer caídas. Así, partiendo de un estado estático, se analizan caídas frontales, de espalda y laterales. Y partiendo de un estado sentado se presentan caídas laterales y caídas de espalda.

Un estudio bastante interesante en esta línea es en [70], donde se presenta un método de monitorización de la actividad física capaz de detectar el estado postural del individuo (sentado, de pie y tumbado) así como periodos de caminar en personas mayores con únicamente un sensor cinemático colocado sobre el pecho. El sensor cinemático se compone de un giroscopio piezoeléctrico miniaturizado con un rango de medida de  $\pm 400^\circ/\text{seg}$  que permite medir la velocidad angular del tronco ( $g_s$ ) en el plano sagital y un acelerómetro en miniatura con un rango de  $\pm 2g$  que permite medir la aceleración vertical del tronco ( $a_v$ ).

En [71] se mejora el sistema anterior al incorporar otro acelerómetro al sensor cinemático, que nos permitirá medir la aceleración del tronco tanto vertical ( $a_{vs}$ ) como frontal ( $a_{fs}$ ) y de esta manera detectar el estado postural y sus transiciones. Para reconocer diferentes transiciones posturales y periodos de caminar durante la actividad física diaria se usa un análisis wavelet junto con un modelo cinemático simple.

[72] propone un sistema compuesto por un bloque detector de caídas portado en el cinturón, un bloque de procesamiento de señales y un bloque de telecomunicaciones. El bloque sensor está compuesto por tres acelerómetros biaxiales. Se establece que la caída solo puede comenzar desde la posición vertical del individuo, lo cual permite reducir el número de falsas alarmas. En el algoritmo se considera que, en el inicio de una caída debe ocurrir un cambio grande tanto en las componentes  $X$  e  $Y$  de la aceleración, que viene determinado por la superación de dos umbrales respectivamente durante un intervalo de tiempo  $t_{down}$ . Aunque el sistema es capaz de generar alarma en tiempo real, su portabilidad en cualquier situación del paciente no es posible. Por otro lado el uso de tres acelerómetros biaxiales implicaría tanto un espacio en placa mayor como un incremento en el coste del dispositivo.

En [73] se propone un sistema ambulatorio para la detección de caídas en tiempo real específico para personas mayores, compuesto por una parte sensora (acelerómetros triaxiales, sensor de inclinación y giroscopio) y un módulo de comunicaciones (módulo bluetooth y microcontrolador integrado para monitorización en tiempo real). También se presenta un algoritmo simple que permite la detección de caídas a partir de los datos acelerométricos capturados. Aunque el sistema permite la monitorización del individuo en tiempo real, de nuevo, existen situaciones de alto riesgo que no serían cubiertas como cuando la persona se está duchando.

En [74] se propone usar un pequeño dispositivo llevado sobre la muñeca y una red de sistemas fijos en el hogar del paciente. Juntos pretenden detectar eventos de caídas así como determinar la localización de la víctima. Para la detección se usan acelerómetros MEMS de baja potencia y bajo coste mientras que la localización se realiza por medio de una señal de RF. El sistema completo se puede considerar dividido en un subsistema fijo y un subsistema móvil, conformando de esta forma una infraestructura de red de sensores. El inconveniente más claro de este dispositivo es su imposibilidad de funcionar fuera del hogar, debido a la necesidad del conjunto de dispositivos para la localización del individuo, limitando entonces la capacidad de movimientos de este

En [75] se presenta un sistema de análisis ambulatorio, en el que se recogen la aceleración de un impacto, la dirección y el instante de la caída. El detector de caídas es implementado con un monitor portable y un computador para el análisis. El monitor

está compuesto de un acelerómetro triaxial piezorresistivo, un microcomputador y un datalogger. Como vemos, este sistema no permite detección de caídas en tiempo real, puesto que el procesamiento de los datos se hace offline (los datos son transferidos al PC vía un puerto RS232)

En [76] se desarrolla un detector de caídas basado en acelerometría situado en la cabeza a modo de audífono, así como un algoritmo capaz de distinguir entre actividades de la vida diaria y caídas simuladas. Según el autor, la alta sensibilidad y especificación del algoritmo muestra un mejor comportamiento que en otros detectores de caída llevados en la cintura o muñeca en la misma etapa de desarrollo. Los resultados expuestos en este estudio demuestran una alta sensibilidad, aunque el procesamiento lo realizan de forma off-line con lo que no resulta útil para la monitorización en tiempo real. Por otro lado, la posición del parche lo hace visible y por tanto podría disminuir su aceptación.

Del estudio del arte presentado se extrae que todavía no ha sido implementado ningún dispositivo capaz de cubrir todos los requisitos necesarios para una detección de caídas de forma eficaz y permitiendo una monitorización en tiempo real y en cualquier situación. En los pocos casos en los que se realiza un análisis en tiempo real, el diseño del dispositivo no permite la monitorización en las situaciones de alto riesgo, como son cuando el paciente va al baño o sale de su hogar. Además, muy pocos estudios tratan convenientemente o ni siquiera mencionan la necesidad de personalización y de adaptación al medio en el que el individuo se desenvuelve. Por último, en muchas ocasiones los resultados que se presentan no están realizados sobre individuos en situaciones cotidianas con lo que pueden no ser suficientemente fiables

El Grupo de Ingeniería Biomédica (GIB) ha presentado recientemente un estudio experimental al que ha sido sometido un prototipo de la IAU, en el que se ha demostrado la fiabilidad de su función de detección de impactos [22]. Esta unidad es capaz de detectar caídas, y forma parte de un sistema más amplio que denominamos monitor de movimiento [77], [78], [79], [80], [81], [82].

### **1.3.- Nuevas tendencias de control. Evolución**

#### **Introducción**

Muchos estudios han subrayado las direcciones futuras en control, como prueba la siguiente remesa bibliográfica [83.-86]. De ellos se entiende que los nuevos pasos en el campo del control son multidisciplinarios. Sírvese de ejemplo los vehículos de conducción autónoma, las redes de información robustas o las técnicas de mejoramiento para el sensado y la detección de agentes biológicos y químicos. Esta evolución está íntimamente relacionada con los rápidos avances en computación, comunicaciones, y tecnología de percepción, que ofrecen oportunidades sin precedentes para que el campo de control expanda sus contribuciones

Una mirada moderna del control nos muestra la realimentación como una herramienta para la gestión de lo que no tenemos certidumbre. Al medir la operación de un sistema, compararlo con una referencia, y ajustar variables de control disponibles, podemos provocar que el sistema responda apropiadamente incluso si su comportamiento dinámico no es conocido de forma precisa o si perturbaciones externas tienden a provocar en el una respuesta incorrecta. Es precisamente este aspecto de control como medida para asegurar robustez frente a incertidumbres lo que explica porqué los sistemas de control realimentados están por todas partes en nuestro mundo tecnológico moderno o en campos tan diversos como la biología o la medicina, donde los mecanismos de realimentación están siempre presentes [87.-89]

#### **Éxitos e impactos del control**

En los últimos 40 años, el advenimiento de la electrónica analógica y digital ha permitido a la tecnología de control extenderse más allá de sus aplicaciones iniciales y lo ha convertido en una tecnología útil en muchas aplicaciones. Podemos agrupar los éxitos más importantes que se han logrado en las siguientes categorías:

- Sistemas de guiado y de control de vehículos, misiles y satélites. Proporcionan estabilidad y seguimiento de trayectorias en presencia de entornos hostiles e incertidumbres
- Sistemas de control en la industria de la producción desde automóviles a circuitos integrados. Proporcionan posicionamiento y ensamblaje preciso.

- Sistemas de control de procesos industriales de hidrocarburos e industrias químicas. Mantienen alta calidad de producción al monitorizar una multitud de sensores heterogéneos.
- Control de sistemas de comunicación, desde sistemas de telefonía, teléfonos celulares e Internet. Regulan los niveles de potencia de señal en los transmisores y receptores, gestionan los buffers de paquetes en los equipos de enrutado, y proporcionan cancelación de ruido para responder a las características variantes de las líneas de transmisión.

### **Futuras oportunidades y retos**

Las oportunidades para nuevas aplicaciones que construirán avances en control se expanden increíblemente. La aparición de sistemas de sensores, así como de sistemas de comunicación y de computación distribuida ha empezado a crear un entorno en que hay que acceder a enormes cantidades de datos y tener la habilidad de procesarlos y comunicarlos en formas impensables hace 20 años [90]. El control debe ser un elemento esencial a la hora de construir estos sistemas interconectados, proporcionando alto rendimiento, alta confiabilidad y operación reconfigurable en presencia de incertidumbres.

Un reto de este campo es ir de la vista tradicional de los sistemas de control como un único proceso con un único controlador, hacia un paradigma más complejo en el que se presentan los sistemas de control como una colección heterogénea de sistemas de información y físicos, con interacciones e interconexiones complejas.

En [91] se trata específicamente de cinco grandes áreas donde debe centrarse los esfuerzos de aplicación de control. Estas áreas son la *aeroespacial y de transportes*, la *robótica y máquinas inteligentes*, la de *procesado y materiales*, la de *información y redes* y la de *medicina y biología*.

Por razones obvias nos centraremos en las dos últimas, que son los campos que toca la aplicación presentada en este trabajo, en el que se realiza un control de una red inalámbrica de sensores biomédicos inteligentes y se aplica a la monitorización del movimiento humano.

### **Información y redes**

El rápido crecimiento de las redes de comunicación proporciona numerosas oportunidades inmejorables y retos para el control. Podemos dividir esta área a su vez

en dos áreas principales, que se superponen al estar íntimamente relacionadas: control de redes y control sobre las redes.

El *control de redes* abarca un área extensa, abarcando temas como control de congestión, rutado y gestión de potencia.

El *control sobre redes* trata sobre las redes de sensores o actuadores con capacidades computacionales, conectadas inalámbricamente o alámbricamente, que pueden formar una orquesta que controla nuestro entorno físico. Ejemplos de ellos lo tenemos en casas, sistemas de fabricación, autopistas inteligentes o en los hogares y fuera de ellos con fines médicos, como el caso de nuestra aplicación. De esta forma, la siguiente fase en la revolución de la tecnología de la información es la convergencia de comunicaciones, computación y control.

Las nuevas tecnologías de sensores están creando nuevas oportunidades de control. Los sensores online – ultravioleta, infrarrojo,...- están haciéndose más robustos y menos caros y aparecen en más procesos de fabricación. Muchos de estos sensores están siendo ya usados por los sistemas de control de procesos actuales, pero se necesita un procesamiento de señal y unas técnicas de control más sofisticadas para usar de forma más efectiva la información en tiempo real que proveen estos sensores

A medida que las redes existentes continúan desarrollándose, la tecnología de redes se hace más barata y fiable que las conexiones punto a punto, incluso en sistemas de pequeña localización, operando cada vez más sistemas de control para gestionar estas redes. Se puede anticipar un escenario donde señales de sensores, actuadores, diagnóstico y de comando y coordinación viajan a través de redes de datos. Las funciones de control y estimación pueden distribuirse a través de múltiples procesadores, unidos también entre sí a través de redes de datos. Un ejemplo de esto lo tenemos en nuestro dispositivo, en el que un conjunto de sensores pequeños pueden desarrollar un procesamiento de la señal local antes de enviar la información relevante a través de una red inalámbrica.

### **Biología y medicina**

Una cuestión a destacar es la ciencia de la ingeniería de las redes de control biológicas. Hay una gran variedad de fenómenos biológicos que proporcionan una rica fuente de ejemplos de control, incluyendo regulación genética; mecanismos de realimentación cardiovascular, inmunológico y hormonal; locomoción y control muscular; atención y conciencia, etc.

Asimismo hace tiempo que se adivinan una serie de papeles potenciales para el control en la investigación biomédica. Estos incluyen salas de operación inteligentes en los hospitales, desde datos base hasta la toma de decisiones: terapia y cirugía guiada por imagen; integración de tejido y hardware; control del flujo de fluido para ensayos biológicos; y el desarrollo de prótesis físicas y neutrales. Muchas de estas áreas se superponen con la robótica [92]

Son de especial interés tres aspectos interrelacionados de los sistemas biológicos: biología molecular, biología integrativa e imágenes médicas. Estas áreas son representativas de una gran clase de sistemas biológicos y demuestra como los principios del control puedan usarse para entender la naturaleza.

### **Recomendaciones**

Por último destacar un par de recomendaciones interesantes que se hacen en [91] para la administración de EEUU pero que por extensión podrían poder aplicarse en nuestro país:

En lo que se refiere a la *integración de control, computación y comunicaciones* se recomienda un incremento substancial de la investigación destinada a la integración del control, ciencia de los ordenadores, comunicaciones y redes. Esto se debe a que los sensores baratos y ubicuos, comunicaciones y computación serán un gran activo para las nuevas aplicaciones de control a gran escala de sistemas complejos. La investigación sobre las redes y su control u el diseño de sistemas interconectados a gran escala y con gran seguridad van a generar nuevas temas de investigación y retos teóricos

También se recomienda *explorar aplicaciones de control de alto riesgo y alcance* hacia nuevos dominios como la nanotecnología, mecánica cuántica, electromagnetismo, biología y ciencia del medio ambiente. Para ello se aconseja que los expertos en control deben participar activamente en nuevas áreas de control fuera de las raíces tradicionales.

## 2.- Descripción del Sistema

El diseño del monitor portable de movimiento humano persigue una supervisión 24/7 del usuario tanto dentro como fuera del domicilio. Permite la monitorización del sujeto en situaciones de alto riesgo y elimina los componentes de aceleración relacionados con el movimiento relativo en la interfaz sensor-cuerpo humano. La arquitectura del monitor no restringe la localización del sensor sobre el cuerpo, permitiendo un acceso sencillo de su interfaz para el usuario.

El monitor está imbuido en una red personal inalámbrica (WPAN) y está compuesto por el servidor personal (PSE), y por una unidad de acelerometría inteligente (IAU) como observamos en la Figura 1.

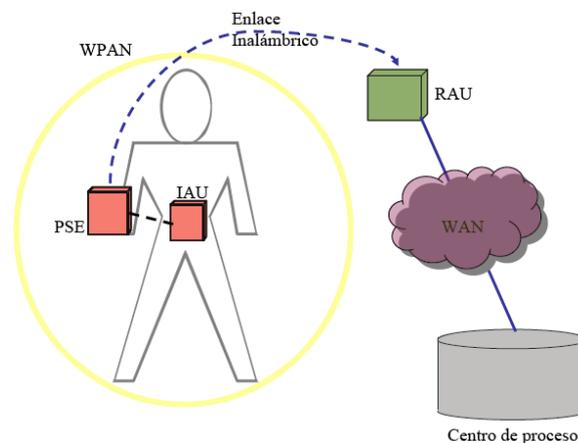


Figura 2.1.- Monitor de movimiento humano

El PSE es un pequeño dispositivo electrónico que será portado en la muñeca o en forma de colgante, y toma el papel de maestro en la WPAN además de procesar en tiempo real las medidas capturadas en la IAU, que actúa como esclavo en esta topología. El PSE gestiona la comunicación entre el monitor portable de movimiento y la unidad de acceso remoto (RAU), que es el punto de entrada al centro de proceso del sistema de teleasistencia. La interfaz del monitor está integrada en el PSE y de esta forma se separa de la IAU, que ha sido diseñada para llevarse como un parche impermeable adherido en la espalda del usuario a la altura del sacro. Las tecnologías inalámbricas empleadas en los enlaces IAU-PSE y PSE-URA son Zigbee y Bluetooth respectivamente, cuya selección es analizada en [93].

Las señales de aceleración son analizadas de forma distribuida entre la IAU y el PSE. La IAU [94] realiza un preprocesado de las señales muestreadas a una frecuencia mayor de la necesaria para estimar los parámetros cinemáticos y posturales. Este análisis

previo se realiza con el objetivo de detectar propiedades de la señal que sugieran la ocurrencia de eventos de caídas y de transiciones posturales. Los eventos y propiedades detectados junto con las señales de aceleración son enviados hacia el PSE. Este último realiza el procesamiento en tiempo real de las señales de aceleración, para adaptar el sistema al entorno en el que en cada instante se desenvuelve el individuo, permitiendo variar ciertos parámetros de la IAU.

El procesamiento distribuido de este sistema actúa emulando a los sistemas biológicos, presentando un preprocesado local que evita la sobrecarga del nodo central, y en caso necesario un procesado más profundo en el nodo central el cual está dotado de una capacidad de procesamiento mucho mayor. De ahí que denominamos al diseño como *bioinspirado*. Con esta metodología de procesamiento distribuida conseguimos disminuir la carga de procesado en el PSE y por tanto el flujo de datos entre dispositivos repercutiendo directamente en una disminución del consumo.

Debemos notar que la IAU determina por medio de un algoritmo implementado en el grupo [95-97], que se ha producido un evento de impacto pero es, en definitiva, el PSE el que establece si el impacto se corresponde con una caída y en tal caso gestiona las alarmas a enviar al centro de proceso.

### 3.- Objetivos

El objetivo principal de este trabajo es presentar la arquitectura hardware y software de un servidor personal para el control y procesado de señales provenientes de sensores inteligentes. Para el diseño del mismo se han tenido en cuenta los siguientes requisitos fundamentales:

- Mínimo consumo: la elección de los componentes ha sido de tal forma que permitan diversos modos de operación y bajo consumo, sin que esto implique una pérdida de seguridad para el paciente monitorizado.
- Tamaño reducido: perseguimos que el dispositivo sea discreto para que no haya problemas de aceptación por parte del paciente.
- Fiabilidad máxima: indispensable dado el contexto de detección de caídas.
- Autonomía máxima: directamente relacionado con el consumo. Queremos minimizar el número de recargas de la batería.
- Portabilidad: el dispositivo final será implementado para que el paciente pueda portarlo en todo momento en su hogar sin dificultar ninguna de sus actividades domésticas.
- Sencillez de manejo: indispensable, debido a que sus usuarios potenciales son personas mayores que pueden no tener conocimientos tecnológicos.

Desde un punto de vista de su funcionalidad en el sistema de monitorización en que se integra el PSE, tenemos los siguientes objetivos:

- Capacidad de detección de caídas y cambios bruscos en la movilidad.
- Capacidad de monitorización personalizada on-line adaptándose al entorno.
- Incorporación de sensores biomedicos de última generación sin perder la modularidad del sistema.

## 4.- Resultados

### 4.1.- Arquitectura Hardware

Como ya comentamos en el apartado de Descripción del Sistema, el monitor de movimiento está formado por una red inalámbrica de área personal (WPAN) en la que se encuentra imbuido el sujeto a monitorizar e integra, al menos, una unidad inteligente de acelerometría (IAU) y el servidor de dicha red (PSE). El PSE establece la comunicación inalámbrica entre la red y un sistema central de almacenamiento de datos y gestión de incidencias (URA).

Recordamos también que las funciones que se pretende que ejerza el PSE son las que enumeramos a continuación:

- Ejercer de servidor de red de la WPAN.
- Procesar aceleraciones en tiempo real.
- Gestionar las comunicaciones.
- Cumplir con los requisitos de autonomía, portabilidad, tamaño, discreción, coste y sencillez que se le exigen.

#### 4.1.1.- Diagrama de bloques de la arquitectura Hardware

El núcleo del PSE está constituido por un procesador digital de señal (DSP) encargado del procesamiento en tiempo real de las aceleraciones y la gestión de los módulos externos: módulo de comunicaciones, memoria, interfaz de usuario y módulo de control-supervisión (ver Figura 4.1).

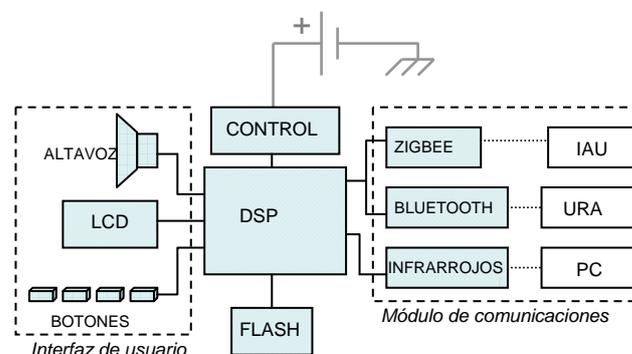


Figura 4.1.- Diagrama de bloques del HW del PSE

Sin lugar a dudas la parte más importante del dispositivo es la que aparece en la figura en la zona central no recuadrada. En ella podemos distinguir los siguientes elementos:

- DSP: será el núcleo o CPU del dispositivo y por tanto su importancia es vital.
- Memoria Flash: actuará tanto de memoria de datos como de programa.
- Módulo de Control y Supervisión: como su propio nombre indica velará por el correcto funcionamiento del Servidor Personal reiniciándolo en caso de operación incorrecta.

Por último en la parte derecha tenemos el bloque de comunicaciones, donde destacan:

- Transceptor Zigbee: que servirá para comunicarse con la IAU.
- Transceptor Bluetooth: dará cobertura al enlace PSE-URA.
- Transceptor Infrarrojo: servirá para reprogramar el dispositivo desde un PC externo, en caso de incorporación de nuevos sensores, actualización de los algoritmos matemáticos implementados, etc.

En la parte de la izquierda tenemos la interfaz de usuario. Esta será configurable atendiendo a la destreza del usuario y sus funcionalidades se adaptan conforme se incorporan sensores biomédicos a la WPAN, sirviéndose para ello de uno de los protocolos de comandos entre el PSE y un ordenador personal a modo de Host por medio del puerto infrarrojo. Consta de:

- Altavoz: para la reproducción de mensajes sonoros, tanto de alerta como para tranquilizar al paciente en caso de caída o percance grave.
- Pantalla LCD: en ella se mostrará información heterogénea, desde la relativa al estado del dispositivo y de configuración para el técnico encargado de su mantenimiento o de información general para el sujeto monitorizado.
- Botonera: constará de cuatro botones, donde uno de ellos actúa como botón de emergencia (luminoso) y el resto permite el desplazamiento por los menús que aparecen en pantalla y la selección de la opción deseada.

En los siguientes apartados analizaremos con mayor detenimiento las soluciones tecnológicas escogidas para cada uno de los bloques mencionados [98], con el objetivo de mostrar finalmente el prototipo preindustrial de PSE que actualmente tenemos implementado en PCB a falta de montar los componentes y poder realizar las pruebas convenientes. Señalar que a la hora de añadir los elementos resistivos capacitivos e

inductivos de los esquemáticos, se ha acudido a las recomendaciones de los datasheets de los dispositivos empleados.

#### **4.1.2.- DSP**

Como hemos comentado anteriormente, el corazón del PSE es un Procesador de Señales Digital o DSP. En este dispositivo es donde se llevará a cabo el procesamiento en tiempo real de las señales de aceleraciones procedentes de las unidades inteligentes de acelerometría IAU's. Es por tanto lógico que uno de los requisitos de partida más importantes del procesador es que sea capaz de procesar a una gran velocidad y, en segundo lugar, que disponga de utilidades que faciliten el cálculo de manera eficiente de las transformadas matemáticas necesarias para el tratamiento del tipo de señales con el que se trabaja. Para el caso de las señales procedentes de la IAU serán en última instancia valores de longitud un byte en Complemento a Dos (CA2) correspondientes a cada eje del acelerómetro embebido en la IAU.

Los DSPs operan tanto en punto fijo como en punto flotante [99], siendo los primeros los que más numerosos en el mercado. Las ventajas de los que trabajan en punto fijo se resumen en que son más baratos, consumen menos potencia y ocupan menos silicio. Por otro lado, los procesadores en punto flotante se caracterizan por manejar una cantidad enorme de variaciones en valores numéricos y aun así seguir proporcionando resultados extremadamente precisos. De todas formas es conveniente señalar que hoy en día las diferencias señaladas se están suavizando de manera significativa aun cuando no han desaparecido completamente. Actualmente, la característica crítica para los diseñadores a la hora de decidir entre uno y otro es la mayor precisión y flexibilidad matemática del formato, decantándose por en punto flotante. Para el manejo de datos que requieran aritmética real, gran precisión y un rango dinámico más amplio, los DSPs en punto flotante son los que ofrecen la mejor solución. Huelga decir que el procesamiento de señales que se va a realizar en el monitor demanda esos requisitos y por tanto también nos decidimos por un procesador en punto flotante.

Otros requisitos fundamentales a la hora de elegir el procesador más adecuado para nuestros objetivos son los que comentamos como sigue. Por un lado hay que recordar que el PSE será portado por el paciente, con lo cual su tamaño es una magnitud a minimizar. Por otro lado y muy relacionado con la anterior está la cuestión de la autonomía, es decir, el PSE tiene que minimizar su consumo prolongando el tiempo de operación sin interrupciones para una batería dada. Como nos dice el sentido común y

después veremos reflejado en el apartado de consumo, minimizar el consumo del PSE implica minimizar el del DSP. Además también se persigue minimizar en la medida de lo posible el precio del DSP, y del PSE en general, puesto que la aplicación desarrollada está destinada a un grupo poblacional, el de las personas mayores, que como señalamos anteriormente cada vez es más numeroso y por ello supone mayor gasto para las administraciones públicas en materia de salud. Esto se hace con vistas a que el dispositivo pueda ser financiado por la sanidad pública de forma que sus gastos a largo plazo se vean reducidos si realiza la inversión tras la comercialización del dispositivo, de ahí que el aspecto económico sea crucial. Una vez cumplidos los requisitos anteriores, buscaremos aquel DSP que ofrezca una periferia más completa, versátil y útil de cara a la aplicación que tenemos que desarrollar. No hay que olvidar que en nuestra aplicación el PSE se comunica de forma inalámbrica con las IAUs y con la URA, con lo que el DSP deberá tener periféricos con los que poder comunicarse con los respectivos transceptores. Por último y no menos importante en la elección, será el hecho de que el procesador escogido tenga herramientas de desarrollo tanto software como hardware que faciliten la implementación del servidor personal.

Tras realizar un estudio sobre las distintas posibilidades que ofrecía el mercado se estudiaron los DSP de la plataforma. TMS320C6000<sup>TM</sup> de Texas Instruments (TI). Esta plataforma [100] ofrece el rendimiento más alto tanto en punto fijo como en punto flotante y se recomienda para aplicaciones de video, imagen, infraestructura de banda ancha y aplicaciones de audio.

Dentro de esta plataforma vamos a tener procesadores optimizados para el tratamiento de video es el caso de las familias TMS320C64x<sup>TM</sup>. También se puede distinguir aquellos procesadores que operan en punto fijo TMS320C62x<sup>TM</sup> y TMS320C64x<sup>TM</sup>, y los que operan en punto flotante, la familia TMS320C67x<sup>TM</sup>. Obviamente por lo señalado con anterioridad será esta última familia la que nos interese. Comentamos resumidamente alguna de sus características para pasar a analizar el DSP escogido dentro de esa familia, el TMS320C6737-300 de TI.

**TMS320C67x<sup>TM</sup> floating point DSP:** operan en punto flotante, ofreciendo precisión, velocidad, ahorro de potencia y un rango dinámico operando entre 600 y 1800 MFLOPS. Dentro de ella se distingue la familia C671x y la C672x. Esta última fue la elegida al ser la más moderna y de mejores prestaciones, sobre todo en el núcleo de la CPU [101].

### **DSP TMS320C6727-300**

Para comenzar esta descripción [102, 103] podemos decir que a 300MHz este DSP tiene una potencia de cálculo máxima de 2400 MIPS y 1800 MFLOPS que se consiguen con la ejecución en paralelo de hasta 8 instrucciones en cada ciclo, seis de las cuales son en punto flotante.

Gracias a su eficiente sistema de memoria, el controlador mapea la memoria RAM de 256KByte y la ROM de 384KByte como una memoria de datos y programa unificada. El desarrollo se simplifica al no haber una división fija entre el tamaño de la memoria de datos y la de programa. Este controlador permite accesos a datos, en un solo ciclo, desde la CPU a la RAM y a la ROM.

La caché de programa de 32KByte consigue el volcado a una gran velocidad en la mayoría de aplicaciones. Esto evita la mayoría de conflictos de acceso a programa y/o datos en la memoria on-chip. También habilita la ejecución eficiente de código de programa desde una memoria externa al chip como puede ser una SDRAM.

Nos centramos ahora en la periferia del DSP. El **EMIF** (External Memory Interface) de 32 bits de ancho de bus de datos, soporta un banco simple de SDRAM y un banco simple de memoria asíncrona. Por tanto nuestro DSP puede conectarse a memoria SDRAM de hasta 256Mbits ó 512Mbits.

El soporte de memoria asíncrona se utiliza típicamente para arrancar desde un dispositivo de memoria NOR Flash no multiplexada que puede tener un ancho de bus de datos de 8, 16 ó 32 bits. Para arrancar desde dispositivos mayores que los soportados por las líneas de dirección dedicadas del periférico EMIF, se utilizan líneas de entrada salida de propósito general para las líneas superiores de dirección. También es posible configurar esta interfaz para la conexión de una memoria NAND-Flash de 8 ó 16 bits de ancho de bus de datos.

Nuestra aplicación emplea una memoria Flash que se conectará al periférico EMIF y desde la que arrancará el DSP. Por su tamaño se utilizan líneas adicionales para las líneas superiores de dirección como se explicaba en el párrafo anterior. La justificación del uso de una memoria no volátil externa se hará en el correspondiente capítulo.

Otro de los periféricos de este DSP es el **UHPI** (Universal Host Port Interface). Se trata de una interfaz paralela a través de la cual, una CPU externa al DSP puede acceder al DSP. Este periférico no se utiliza en nuestro primer diseño.

Los **McASPs** (Multichannel Audio Serial Ports) proporcionan una interfaz que no necesita lógica adicional para su conexión con CODEC, DACs, ADCs y otros dispositivos. Uno de los puertos McASP lo vamos a utilizar para transmitir información al paciente o al técnico encargado de la revisión y reprogramación del PSE, mediante la conexión de un DAC y unos altavoces.

Los periféricos **I<sup>2</sup>C** (son dos. Muchos dispositivos electrónicos poseen esta interfaz para su comunicación con un procesador o microcontrolador. En el PSE se empleará este periférico para comunicar al DSP con el display LCD así como el dispositivo que monitoriza el estado de la batería.

También disponemos de dos puertos **SPI** (Serial Peripheral Interface) uno de los cuales se configurará como maestro en la interfaz con los dos transceptores que actuarán como esclavos, conmutando apropiadamente entre uno y otro.

El DSP escogido también cuenta con un temporizador de interrupciones en tiempo real, **RTI** (Real-Time Interrupt Timer) que incluye: dos pares de contadores/preescaladores de 32 bits, dos capturadores de entradas, cuatro comparadores con capacidad de actualización automática y un sistema watchdog para mejorar la robustez del sistema.

El PSE utiliza el oscilador on-chip que incluye el DSP junto con un cristal de 24.576MHz externo.

El DSP elegido no posee pines para interrupciones de propósito general, no obstante, el pin dMAX se puede utilizar en combinación con una señal AMUTEIN del McASP para proporcionar esa capacidad de interrupción externa. Hay un multiplexor para cada McASP, controlado por los registros CFGMCASP0/1/2, que permiten que la entrada AMUTEIN de ese McASP sea una de siete pines de entrada/salida del DSP. Una vez que el pin se ha configurado como una fuente AMUTEIN, un pulso corto (de dos ciclos SYSCLK2 o más) generará un evento en el dMAX. Ese evento puede disparar el dMAX para generar una interrupción a la CPU programando el evento de entrada asociado.

#### **4.1.2.1.- Multiplexación de pines de los periféricos del DSP**

Los pines del DSP van a compartir distintas funcionalidades en la medida de que pueden ser asignados a distintos periféricos. Por lo general, cada periférico tiene un registro de control interno para determinar la función desempeñada por cada pin y si se trata de una entrada o de una salida. Este control software hace que la funcionalidad de los pines pueda cambiar durante la ejecución del programa.

Debido a lo comentado anteriormente y al hecho de que algunos de los periféricos mencionados requieren una dedicación exclusiva durante todo el funcionamiento del dispositivo, podemos indicar que la asignación de pines queda como sigue:

- Se comparte la funcionalidad de los periféricos SPI0, I2C0 e I2C1. El DSP utiliza el periférico I2C0 para comunicarse con el módulo LCD. De esta manera los pines asignados a este grupo quedaría distribuidos así:
  - SPI0\_SOMI/I2C0\_SDA → Habilitado como I2C0\_SDA.
  - SPI0\_SIMO → Disponible como pin GPIO.
  - SPI0\_CLK/I2C0\_SCL → Habilitado como I2C0\_SCL.
  - SPI0\_SCS / I2C1\_SCL → Habilitado como I2C1\_SCL.
  - SPI0\_ENA/ I2C1\_SDA → Habilitado como I2C1\_SDA.
  
- También se encuentran compartidos el SPI1, McASP0 y McASP1. Se utiliza el SPI1 como puerto de comunicación con los dos transceptores empleando cuatro líneas. Por tanto, los puertos de audio McASP0 y McASP1 quedan con un máximo de 12 y 4 pines de datos, respectivamente. Por ello los pines quedarían con la siguiente configuración:
  - AXR0[5] / SPI1\_SCS → Habilitado como SPI1\_SCS .
  - AXR0[6] / SPI1\_SCS → Configurado como AXR0[6], pin GPIO .
  - AXR0[7]/SPI1\_CLK → Habilitado como SPI1\_CLK.
  - AXR0[8]/AXR1[5]/SPI1\_SOMI → Habilitado como SPI1\_SOMI.
  - AXR0[9]/AXR1[4]/SPI1\_SIMO → Habilitado como SPI1\_SIMO.
  
- El periférico EMIF y el UHPI comparten pines. Nuestra aplicación utiliza una interfaz con memoria externa con un ancho de datos de 16 bits por lo que la funcionalidad de los pines queda con las líneas EM\_D[31:16]/UHPI\_H[15:0] configuradas como UHPI\_HA[15:0] y el periférico UHPI en el modo Non-Multiplexed Address/Data Mode.

Podemos observar esta multiplexación de pines en el siguiente esquemático.

### 4.1.2.2.- Esquemático del DSP

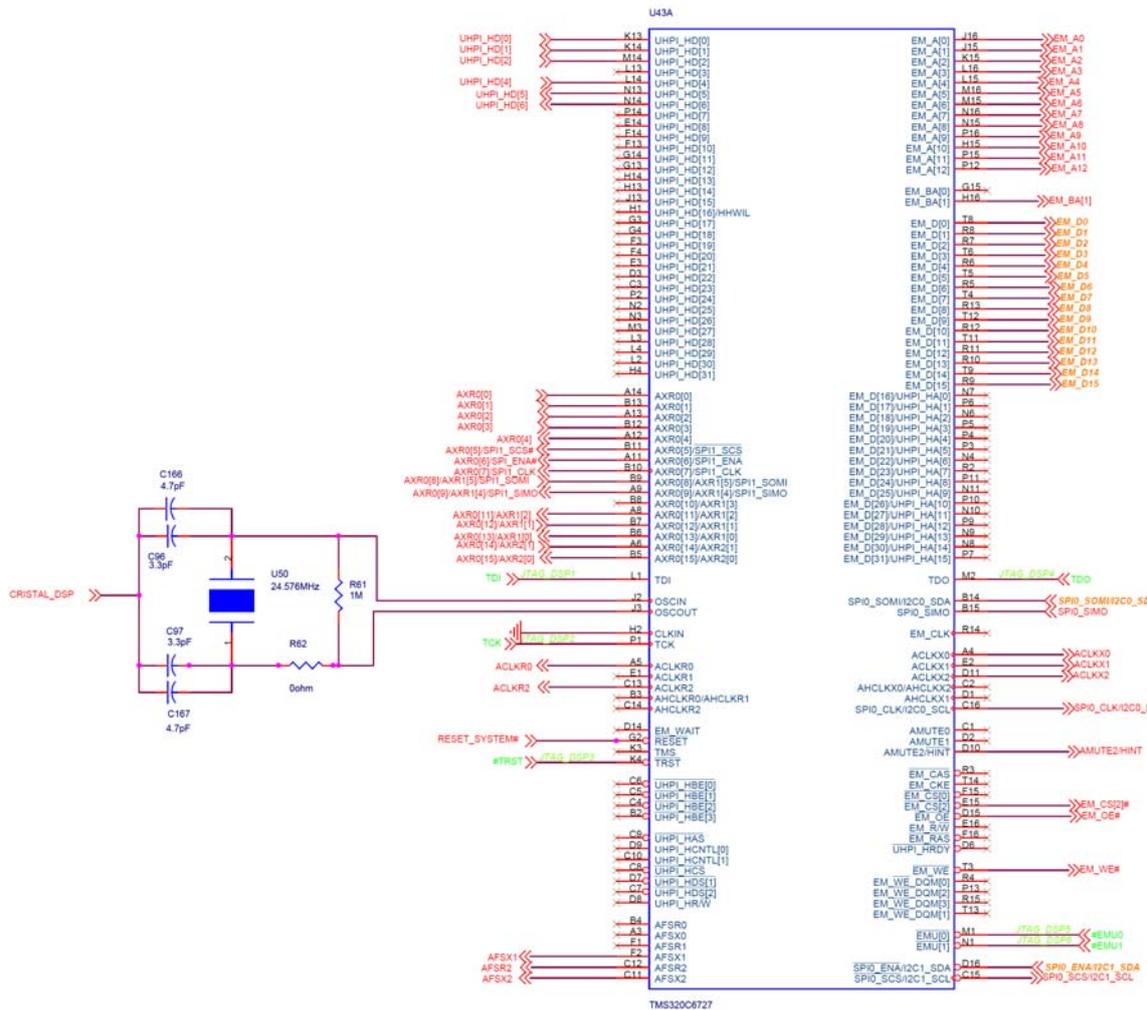


Figura 4.2.- Esquemático del DSP

### 4.1.3.- Transceptor Zigbee

El enlace inalámbrico establecido entre el PSE y la IAU es de muy corto alcance en la medida en que el sujeto porta la unidad sensora en la zona sacra de la espalda todo el tiempo y que el servidor, que en el diseño final será un dispositivo a modo de colgante o de pulsera, puede ser portado por el paciente monitorizado o encontrarse en la misma habitación si el paciente no lo lleva por él mismo (por ejemplo cuando el paciente se esté tomando una ducha o mientras duerme). En todo caso teniendo en cuenta esta primera premisa parece lógico pensar en un estándar inalámbrico WPAN con una cobertura aproximada de diez metros.

Otro requisito fundamental es que el consumo de potencia sea el menor posible, puesto que los transceptores suelen ser los elementos de este tipo de dispositivos con mayor consumo.

También hay que señalar que se empleará una topología de red maestro-esclavo en que el PSE actuará como maestro, iniciando la comunicación, y el resto de las unidades sensoras como esclavos. Otra cuestión importante del diseño mencionada numerosas veces es la de la escalabilidad: pretendemos que la WPAN sea abierta desde un punto de vista tecnológico tendiendo en cuenta que la adición de nuevos sensores no suponga una merma en su funcionamiento. Por último y muy relacionado con lo anterior, señalar que las distintas inalámbricas que pudieran convivir en el entorno del paciente deben hacerlo sin ningún tipo de problema distinguiendo además entre pacientes monitorizados próximos entre sí.

De acuerdo con esto último, se han implementado sensores inalámbricos recientes para las PANs que han sido implementados siguiendo una gran variedad de estándares como IrDA [104], Bluetooth (IEEE 802.15.1) [105], UWB (IEEE 802.15.3) [106], Zigbee (IEEE 802.15.4) [107], WiFi (IEEE 802.11X), GSM-GPRS [108] o incluso multiestándar. Por otro lado, está muy extendido emplear transceptores RF ya implementados para su uso comercial con modulaciones FSK, OOK o ASK [109][110]. Aunque el tiempo de desarrollo puede disminuirse en estos casos, la falta de interoperabilidad de las soluciones propietarias y su limitada escalabilidad son dos inconvenientes importantes. Teniendo en cuenta que las comunicaciones dentro de la WPAN son usualmente unidireccionales y dentro de la línea de vista, y que el ancho de banda requerido por la IAU es pequeño [111], Bluetooth y Zigbee destacan entre el resto de las alternativas. Comparado con Bluetooth, Zigbee soporta un mayor número de nodos, topologías de red diversas, conciencia de localización y un consumo de potencia muy bajo, al coste de una tasa de datos menor [112]. Es por estos motivos que hemos elegido a Zigbee como la interfaz inalámbrica para el enlace IAU-PSE.

A continuación comentaremos brevemente las características principales de la solución escogida.

### **CC2430 DE CHIPCON**

Como transceptor Zigbee se ha elegido el dispositivo CC2430 de Chipcon (TI) [113] para establecer la comunicación entre el PSE y la IAU siguiendo el estándar IEEE 802.15.4 a 250Kb/s. Se comunica a su vez con el DSP a través del periférico SPI (Serial

Peripheral Interface) a 6MHz siguiendo un protocolo comando–respuesta. En esta interfaz maestro–esclavo el DSP, y lógicamente el PSE, actúa como maestro y la IAU, y como consecuencia el transceptor, como esclavo. Toda trama recibida–transmitida se procesa en el microcontrolador del transceptor, puesto que se trata de un dispositivo SoC. Como consecuencia el DSP va a ser transparente a cuestiones como el entramado y sincronismo de la red inalámbrica IAU-PSE.

### **Módulo de Evaluación del transceptor CC2430**

Para la implementación del prototipo preindustrial, hemos hecho uso de un módulo de evaluación del transceptor CC2430 que adquirimos como parte de un entorno de desarrollo para Zigbee de Chipcon. Este módulo de evaluación (ver fotografía adjunta) será una placa PCB en la que estará embebido el transceptor, que se conectará a nuestro prototipo del PSE mediante un par de conectores de 2x10 pines.



Figura 4.3.- Módulo de Evaluación del CC2430

Cabe señalar que este módulo no irá en la versión definitiva del PSE, si no que en ella irá embebido el transceptor. Se emplea en el prototipo preindustrial con el objetivo de realizar pruebas y depurado del software implementado.

### **El Estándar SPI**

Serial Peripheral Bus o SPI es un enlace estándar de transmisión de datos serie asíncronos implementado por Motorola que opera en modo full duplex. Los dispositivos se comunican en modo maestro-esclavo, donde el dispositivo maestro inicial la transmisión de los datos. También se permiten múltiples esclavos con un a sola línea de selección de esclavo. La mayor parte de las ocasiones el bus SPI será un bus serie de “4 cables”, como podemos observar en la siguiente figura, aunque también existen configuraciones con 3, 2 e incluso con un solo cable.

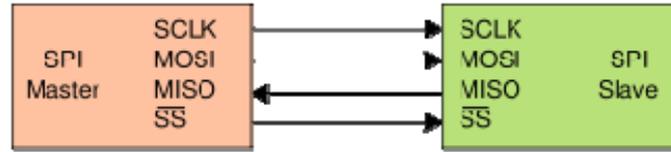


Figura 4.4.- Configuración usual SPI

### Interfaz

La interfaz consta de las siguientes señales lógicas:

- SCLK – Serial Clock, que es salida del maestro.
- MOSI/MISO- Master Output, Slave Input (salida del maestro).
- MISO/SOMI – Master Input, Slave Output (salida del esclavo)
- SS – Slave Select, activo a nivel bajo (salida del maestro)

### Transmisión de datos

Para empezar una comunicación, el maestro primero configura el reloj, empleando una frecuencia menor o igual a la máxima frecuencia que soporta el esclavo. Esas frecuencias están normalmente en el rango de 1 a 70 MHz.

El maestro entonces activa la señal de selección de esclavo. Es posible que para ello tenga que guardar un cierto periodo de espera, típico en dispositivos conversores CAD, antes de empezar a enviar los ciclos de reloj.

En cada ciclo de reloj SPI se produce una transmisión full-duplex (ver figura) de forma que:

- El maestro envía un bit a la línea MOSI; el esclavo lo lee de la misma línea.
- El esclavo envía un bit a la línea MISO; el maestro lo lee de la misma línea.

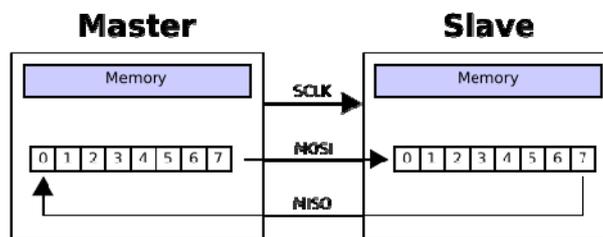


Figura 4.5.- Buffers circulares SPI

Las transmisiones normalmente involucran dos buffers circulares de un tamaño estándar, uno en el maestro y otro en el esclavo, conectados en forma de anillo. Normalmente, el primer bit que se saca es el más significativo, enviando el resto de los

bits de mas a menos significativos al registro de envío. Maestro y esclavo intercambian los valores de ambos buffers. Entonces cada dispositivo procesa los datos intercambiados y si se necesita intercambiar nuevos datos, los buffers se cargan con nuevos datos y el proceso se repite.

Las transmisiones pueden involucrar un número arbitrario de ciclos de reloj. En todo caso, cuando no hay más datos para transmitirse, el maestro para de conmutar su reloj y a continuación desactiva al esclavo poniendo la línea SSN a 1 lógico.

Normalmente los datos intercambiados son bytes de 8 bits, y el maestro puede iniciar su envío siempre que tenga a bien. Sin embargo en ocasiones se emplean otros tamaños como palabras de 16 bits para ciertos controladores a codecs de audio, o de 12 bits para una gran cantidad de convertidores ADC y DAC.

Cada esclavo en el bus que no haya sido activado por medio de su línea de selección de esclavo debe hacer caso omiso al reloj de entrada procedente del maestro y a la señal MOSI, así como no escribir en la línea MISO, puesto que el maestro solo selecciona un esclavo en cada momento.

#### **4.1.3.1.- Interfaz con el DSP**

Se resumen aquí las conexiones que se establecen entre el módulo Zigbee y el DSP como podemos observar en el apartado 5.1.3.2. En primer lugar se da el nombre del pin perteneciente al módulo de evaluación del transceptor y a continuación el nombre del pin correspondiente al DSP:

- **SCK** → *AXR0[7]/SPII\_CLK*.
- **MISO** → *AXR0[8]/AXR1[5]/SPII\_SOMI*.
- **MOSI** → *AXR0[9]/AXR1[4]/SPII\_SIMO*.
- **SSN** →  $\overline{SPI\_CS\_ZIGBEE}$  . Esta señal se va a crear a partir de tres pines del DSP, puesto que son tres los dispositivos que comparten la interfaz SPI y por tanto actuarán como esclavos en dicho estándar: transceptor Zigbee, transceptor Bluetooth y Codec del módulo Audio. Lo podemos ver en la siguiente tabla de verdad generación de señales SPI:

$\overline{\text{SPI\_SCS}}$	$\text{AXR0}[4]$	$\text{UHPI\_HD}[6]$	$\overline{\text{ZIGBEE}}$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	-	-	1

Tabla 4.1.-Tabla de la verdad para  $\overline{\text{SPI\_CS\_ZIGBEE}}$

En todo caso y por lo que nos afecta en estos momentos la señal de habilitación para el transceptor Zigbee sería  $\overline{\text{SPI\_CS\_ZIGBEE}} = \overline{\text{SPI\_SCS}} \text{ OR } \text{AXR0}[4] \text{ OR } \text{UHPI\_HD}[6]$ .

- **P0.0** →  $\text{AXR0}[0]$ . Esta línea adicional de E/S provoca una interrupción externa en el transceptor para regresar del modo de menor consumo al modo activo en el que se emitirá radiofrecuencia.

#### 4.1.3.2.- Esquemático del módulo Zigbee

En este esquemático podemos observar los conectores mencionados con anterioridad

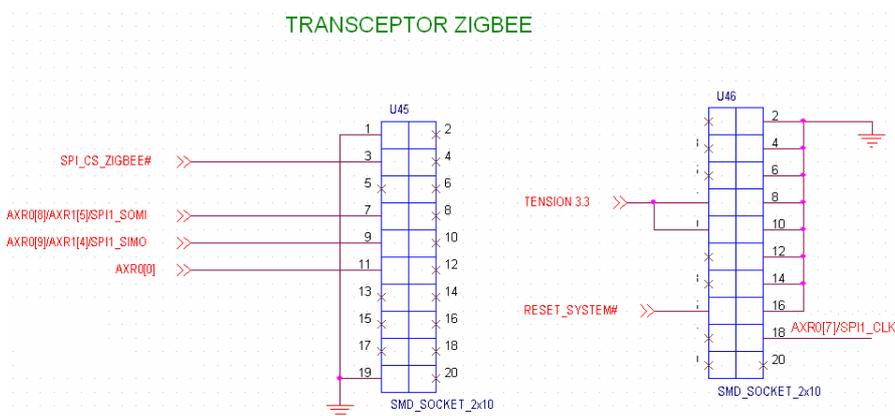


Figura 4.6.- Esquemático de los conectores del módulo Zigbee

#### 4.1.4.- Transceptor Bluetooth

El enlace inalámbrico que se establece entre el servidor personal PSE y la unidad de acceso remoto URA debe tener un mayor alcance que el enlace IAU-PSE, que recordamos estaba acotado a unos diez metros aproximadamente. Esto se debe a que la URA será un dispositivo que hará de interfaz entre el monitor de movimiento humano y

el centro de proceso a modo de módem para poder transmitir la información a través de una red de área extensa WAN. Por lo que podría tratarse perfectamente en la mayoría de los casos de un dispositivo situado en un emplazamiento fijo del hogar del usuario, con lo cual al desplazarse este por el entorno del hogar, también lo hará el PSE y como consecuencia esos diez metros de distancia se verán aumentados y el enlace PSE-URA debe superar esa distancia. Esta distancia se puede agrandar más aún en otros contextos como el de una residencia para personas mayores.

Es por esto que los estándares que se estudiaron para este enlace son del tipo WLAN, i.e., red de área local inalámbrica, que abarcan coberturas de entre 10 y 100 m. Dentro de ellos se encuentran soluciones tan conocidas y comerciales como Wi-Fi o Bluetooth, que se corresponden con los estándares IEEE 802.11 e IEEE 802.15.1, respectivamente. Finalmente se escogió Bluetooth para el enlace PSE-URA [114] porque presentaba características más óptimas en el contexto de nuestro monitor: consumo de potencia más eficiente, diversos modos de ahorro energético o un menor coste unitario de un single-chip Bluetooth respecto de uno Wi-Fi.

### **STLC2500C DE ST MICROELECTRONICS**

El dispositivo STLC2500C de ST Microelectronics [115] es el transceptor Bluetooth (BT) utilizado para comunicar al PSE con la URA mediante el estándar IEEE 802.15.1 a 2Mb/s. La comunicación con el DSP se realiza a 6MHz mediante SPI, interfaz en la que de nuevo el DSP, y por tanto el PSE, actúa como maestro y la URA como esclavo. Este dispositivo posee también un microprocesador que procesa todas las tramas descargando de esta tarea al DSP.

### **Módulo de evaluación del transceptor STLC2500C**

Tal y como sucedía para el caso Zigbee, se ha adquirido un kit de desarrollo para facilitar la tarea de la gestión del transceptor Bluetooth, en este caso se trata del Mini-kit STLC2541X4 de características similares al anterior. Si acaso destacar que es de mayor tamaño y que tiene incorporado un puerto USB para su conexión al Host.



Figura 4.7.- Módulo de evaluación del transceptor Bluetooth

#### 4.1.4.1.- Interfaz con el DSP

Nuevamente esta interfaz vendrá dada por el periférico SPI que es el que se empleará para comunicar al DSP con el transceptor Bluetooth y está formada por cinco señales: reloj, habilitación de esclavo, datos de entrada, de salida e interrupción. Veámoslas brevemente:

- **SPI\_CSN** → Se conecta a la señal  $\overline{SPI\_CS\_BLUETOOTH}$  y es generada como vimos en la tabla siguiente, i.e.:  $\overline{SPI\_CS\_BLUETOOTH} = \overline{SPI\_SCS}$  OR  $\overline{AXR0[4]}$  OR  $\overline{UHPI\_HD[6]}$ .
- **SPI\_CLK** → la señal de reloj debe estar activa durante un número de ciclos múltiplo de 8, y a nivel bajo cuando el SPI no se encuentre activo. Conectada a la línea  $AXR0[7]/SPI\_CLK$  del DSP.
- **SPI\_DO** → Cuando  $SPI\_SCN$  está inactiva,  $SPI\_DO$  aparece como tri-estado. Conectada a la línea  $AXR0[8]/AXR1[5]/SPI\_SOMI$  del DSP.
- **SPI\_DI** → Conectada a la línea  $AXR0[9]/AXR1[4]/SPI\_SIMO$  del DSP.
- **SPI\_INT** → Es la línea de interrupción del esclavo, que se usará cuando éste quiera solicitar una transferencia al maestro, y se activa a nivel alto. Se conecta al pin  $AHCLKR2$  del DSP y se asocia a la interrupción externa AMUTEIN2.

$\overline{SPI\_SCS}$	$AXR0[4]$	$UHPI\_HD[6]$	$\overline{BLUETOOTH}$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	-	-	1

Tabla 4.2.-Tabla de la verdad para SPI\_CS\_BLUETOOTH

### 5.1.4.2.- Esquemático del módulo transceptor

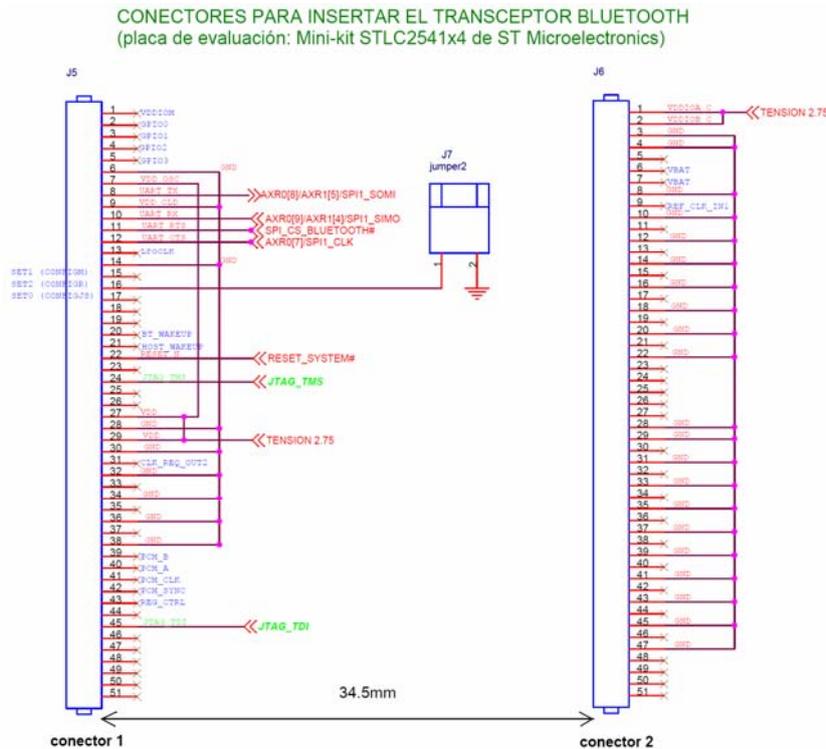


Figura 4.8.- Esquemático de los conectores del módulo Bluetooth

### 4.1.5.- Transceptor Infrarrojo

Un aspecto fundamental a tener en cuenta en el desarrollo del prototipo es la posibilidad de que pueda comunicarse con un Host o PC para que puedan realizarse distintas tareas como:

- Incorporación nuevo sensor biomédicos inteligentes a la red WPAN del monitor.
- Reconfiguración del/de los algoritmo/s matemático/s asociados a las señales procedentes de los sensores inteligentes. En nuestro contexto sería por ejemplo la petición de comprobación de calibración de los acelerómetros o el cambio de su frecuencia de muestreo y rango dinámico.
- Personalización de la interfaz de usuario de acuerdo a su destreza y adaptación de sus funcionalidad según el los distintos sensores de que consta la WPAN.
- Revisión del estado del dispositivo.

Evidentemente estas tareas serán realizadas por el personal especializado encargado de la gestión técnica del servidor personal.

Para implementar esta interfaz entre nuestro dispositivo y un Host se barajaron fundamentalmente dos opciones: emplear un puerto serie RS-232 o un transceptor infrarrojo. Dado las características similares de los protocolos RS-232 e IrDA 1.4 tanto en entramado, como en consumo [116], se eligió el infrarrojo para dotar de mayor portabilidad a la placa y reducir su tamaño, puesto que el conector de puerto serie es mucho mayor que el transceptor Infrarrojo. De esta forma se hace una apuesta total por las soluciones inalámbricas para la comunicación del servidor personal.

### **HSDL 3021 de AVAGO TECHNOLOGY**

El transceptor de infrarrojos es el HSDL 3021 de Avago Technology que permite acceder al PSE desde un PC a una tasa de hasta 4Mb/s. La comunicación entre el transceptor y el DSP se realiza mediante líneas de entrada-salida de propósito general (GPIO), en las que se emulará el protocolo IrDA1.4. Los motivos que nos llevaron a decantarnos por este transceptor, en lugar de otras soluciones de Avago o Vishay, fueron nuevamente el tamaño y consumo. Veamos las características técnicas principales de nuestra elección:

- Consumo LED: 65-150mA.
- Tensión LED: 3.3V.
- Alimentación: 2.4-3.6V.
- Consumo: 3.5mA, 2mA (idle), 1 uA max (shutdown).
- Alcance: 0.5 metros
- Tasa: 9.6Kbps – 4Mbps
- Tamaño: 2.5 x 8 x 3mm

#### **4.1.5.1.- Interfaz con el DSP**

Se trata del dispositivo HSDL-3021 de Avago Technology y las funciones de sus terminales son:

- TxD\_IR → Entrada de datos del transmisor IrDA. Conectada al DSP a través de la línea GPIO AXR0[11]/AXR1[2].
- RxD → datos de recepción IrDA. Conectado al DS a través de la línea GPIO AXR0[12]/AXR1[1].



EPROM porque la única forma de programarlas es mediante un programador externo que borre el contenido de la memoria mediante luz ultravioleta.

Las memorias EEPROM presentan la ventaja de ser borrables a nivel de byte y no a nivel de sector o chip completo como las Flash. En cambio las Flash tienen como mayor mejora el que presentan unos tiempos de programación inferiores a los de las EEPROM lo que las hace más adecuadas para su funcionamiento como memoria de datos. Este hecho nos hizo decantarnos por una Flash.

### **S29AL008D DE SPANSION**

La memoria no volátil Flash que se eligió vino condicionada por los requisitos ya habituales en nuestro diseño hardware como la minimización de su tamaño y consumo. Por otro lado pretendíamos que la memoria elegida fuera de la mayor capacidad posible. Pero sin lugar a dudas lo más importante fue comprobar que efectivamente que la CPU del DSP sea capaz de ejecutar código desde memoria externa. Para ello hará uso del periférico EMIF. En [117] se asevera que el núcleo C67x+ está mejorado respecto al C67x en lo que a tamaño de código se refiere y operaciones flotantes. Destaca de entre las mejoras la existencia de una memoria caché de instrucciones de 32Kbytes que soporta ejecuciones desde las memorias RAM y ROM on-chip, y también desde memoria externa a través de una interfaz de memoria externa (EMIF) basada en VBUSP.

Por tanto la búsqueda final de la memoria Flash se ajustó a las demandas tecnológicas de dicho periférico EMIF, y el resultado de la misma fue el modelo S29AL008D de Spansion que actúa como memoria de datos y de programa. Veamos algunas de sus características más importantes [118]:

- Capacidad de 512K x 16 bits.
- Tensión de alimentación: 2.7V a 3.6V.
- Consumo de corriente de:
  - 200nA en automatic sleep mode.
  - 200nA en standby mode.
  - 7mA en lectura.
  - 15mA en programación/borrado.
- Encapsulado de montaje superficial 48 ball FBGA de dimensiones: 8.15mm x 5.0mm de área y 1mm de altura.

- Presenta algoritmos embebidos para la reprogramación y borrado de todo el chip o de un sector, así como para la escritura y posterior verificación de datos.

#### 4.1.6.1.- Interfaz con el DSP

Como hemos señalado, el dispositivo que hará de interfaz entre el DSP y la memoria Flash será el periférico EMIF. Comentamos brevemente las conexiones:

- A\_FLASH0 – A\_FLASH18 → bus de direcciones. Conectado, respectivamente, a la líneas EM\_BA[1] , EM\_A[0] – EM\_A[12], ACLKR2, AFSR2, ACLKX2, AFSX2 y AMUTE2/HINT del DSP.
- D\_FLASH0 – D\_FLASH15 → bus de datos. Conectado, respectivamente, a la líneas EM\_D[0] – EM\_D[15] del DSP.
- $\overline{\text{WE}}$  → write enable. Conectado a la línea EM\_WE# del DSP.
- $\overline{\text{OE}}$  → output enable. Conectado a la línea EM\_OE# del DSP.
- $\overline{\text{CE}}$  → chip enable. Conectado a la línea FLASH\_CE# procedente de una puerta OR.
- BYTE → ancho del bus de datos. A nivel alto indica palabras de 16 bits.
- $\overline{\text{RY/BY}}$  → estado de la memoria: preparada (ready) u ocupada (busy). Conectado a la línea UHPI\_HD[5] del DSP.

En la siguiente tabla de verdad podemos ver la señal de generación para la habilitación de la flash FLASH\_CE. La señal de  $\overline{\text{RESET}}$  vendrá dada por  $\overline{\text{RESET\_SYSTEM}}$  , que será explicado en otro apartado.

$\overline{\text{EM\_CS[2]}}$	AXR0[15]/AXR2[0]	$\overline{\text{FLASH\_CE}}$
0	0	0
0	1	1
1	0	1
1	1	1

Tabla 4.3.-Tabla de la verdad para  $\overline{\text{FLASH\_CE}}$

#### 4.1.6.2.- Esquemático de la memoria Flash

Nuevamente en este esquemático aparece el socket o conector de la Flash, puesto que en el prototipo queremos tener la posibilidad de poder programar la memoria fuera de la placa. En el diseño final no aparecería dicho socket.

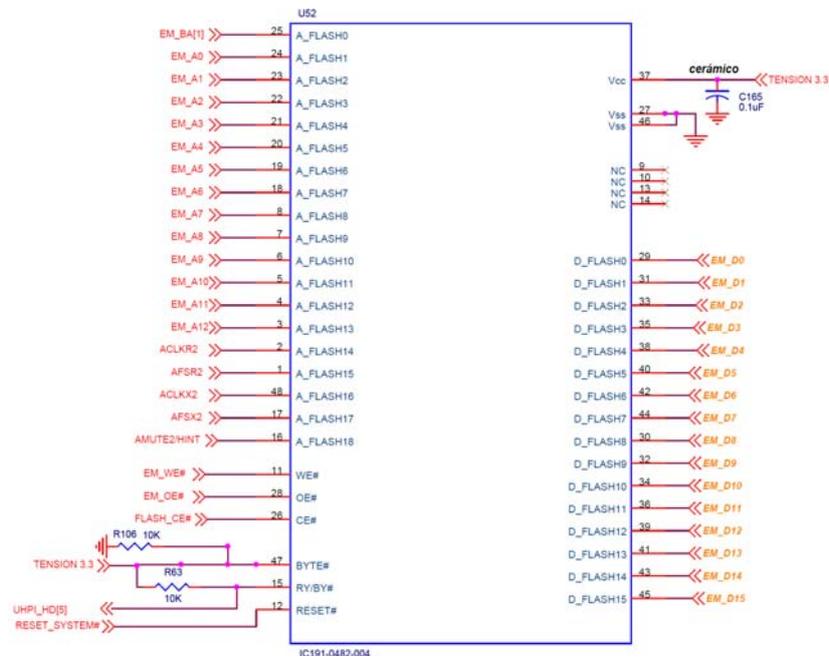


Figura 4.10.- Esquemático de la memoria Flash

#### 4.1.7.- Display LCD

Los objetivos a perseguir para la elección del display son nuevamente el bajo consumo, el reducido tamaño y que el dispositivo estuviese dotado de una interfaz compatible con el DSP. En este último aspecto se decidió que lo más flexible era utilizar una de las interfaces I2C para tener de esta forma disponible la otra para el dispositivo de monitorización de batería, sin necesidad de tener que conmutar entre una y otra vía hardware como hacíamos con los dispositivos que compartía el SPI1: los dos transceptores y el códec audio.

Otro requisito es que pretendemos que la interfaz gráfica sea de gran sencillez de manejo y lo más intuitiva posible, puesto que las personas a las que está destinado el

servidor personal pueden tener sus facultades mermadas y/o no tener grandes conocimientos tecnológicos.

### **EM6125 COG Module de EM Microelectronics**

Es un LCD del tipo Full graphic COG (Chip On Glass) con las siguientes características:

- 102 x 80 puntos.
- Tecnología empleada FSTN.
- Dimensiones externas: 37.5 x 38 mm.
- Tensión de alimentación de 1.8 a 5.5 V.
- Compatible I2C.

### **El Estándar I2C**

El bus I2C es un interfaz de doble cable serie soportada por Philips. La especificación soporta transmisión de datos de hasta 400Kbps. La interfaz I2C emplea un protocolo para la transmisión y recepción de datos gobernado por la comunicación maestro-esclavo. En esta comunicación, ambos pueden actuar como receptor o transmisor, pero el maestro determina qué modo está activado, para esto se cuenta con dos líneas, una correspondiente a la línea de datos (SDA) y otra para la línea del reloj (SCL). En definitiva:

- Maestro: nodo que gestiona el reloj y las direcciones de los esclavos.
- Esclavo: nodo que recibe el reloj y la dirección.

El bus es multimaestro, lo que significa que puede estar presente cualquier número de nodos maestro. Adicionalmente, el maestro y el esclavo intercambian papeles que pueden cambiar entre mensajes (después de enviar un bit de STOP).

El maestro está inicialmente en modo maestro enviando un bit de comienzo seguido de una dirección de esclavo consistente en 7 bits y correspondiente al dispositivo esclavo con el que quiere comunicarse. A continuación le sigue un bit que representa si el maestro quiere leer del esclavo o escribir en él.

Si el esclavo está disponible en el bus le responderá con un bit de ACK (asentimiento) para esa dirección. El maestro entonces continúa ya sea en modo de transmisión o recepción de acuerdo con el bit de lectura/escritura enviado anteriormente, y el esclavo continúa en el modo complementario (recepción o transmisión, respectivamente).

La dirección y los bytes de datos son enviados empezando por el MSB (Most Significant Bit). El bit de comienzo se indica con un flanco de bajada de SDA estando SCL en alto; el bit de Stop es indicado con un flanco de subida de SDA con SCL en alto.

Si el maestro desea escribir en el esclavo entonces envía repetidamente un byte siempre que el esclavo le responda con un bit ACK (En esta situación el maestro esta en modo transmisor y el esclavo en modo receptor).

Si el maestro desea leer del esclavo entonces recibe repetidamente un byte del esclavo y el maestro envía un ACK por cada byte salvo el ultimo (En esta situación el maestro esta en modo receptor y el esclavo en modo transmisor).

El maestro entonces finaliza la transmisión con un bit de Stop o puede enviar otro bit de Start si desea mantener el control para otra transferencia.

A nivel físico simplemente señalar que las líneas SCL y SDA son a drenador abierto, por lo que hay que usar resistencias de pull-up como observamos en la siguiente figura.

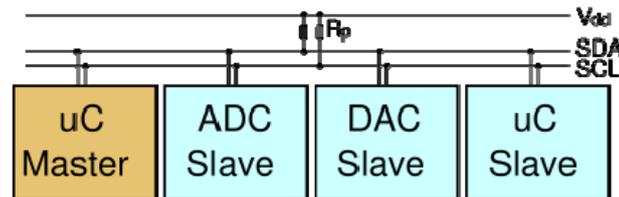


Figura 4.11.- Esquema i2c de un maestro (un microcontrolador) y tres esclavos (ADC, DAC y microcontrolador).

#### 4.1.7.1.- Interfaz con el DSP

Vendrá dada por el periférico I2C y los siguientes pines:

- SDA → datos I2C. Conectada a la línea SPI0\_SOMI/I2C0\_SDA del DSP.
- SCL → reloj I2C. Conectada a la línea SPI0\_CLK/I2C0\_SCL del DSP.
- I → entrada de selección de dispositivo. La habilitamos por defecto.
- CS → chip select. Un uno lógico habilita el dispositivo.

RES → reset.

#### 4.1.7.2.- Esquemático del módulo LCD

Para la conexión del módulo LCD a la placa PCB del prototipo del PSE se va a emplear un conector estilo SMT y vertical al que irá conectado el conector HSL (Heat Seal Connector) del módulo LCD.

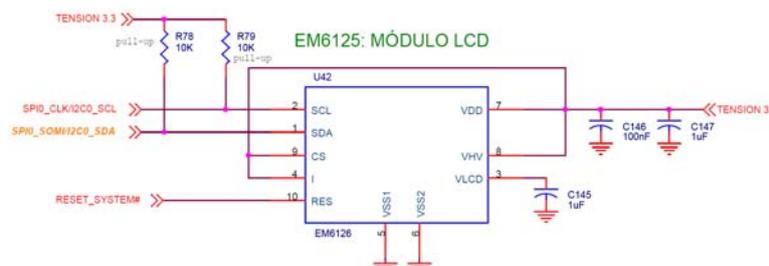


Figura 4.12.- Esquemático del módulo LCD

#### 4.1.8.- La Botonera

El diseño de nuestro prototipo consta de tres botones con los que el usuario pueda acceder y consultar el menú de aplicación del PSE. Dos de ellos serán para su desplazamiento por dicho menú, en un sentido y en el contrario, y el tercero de aceptación de la opción elegida.

Se añadirá un botón de reset manual del mismo tipo que los anteriores para poder reiniciar el dispositivo a un estado conocido.

La botonera también tendrá un botón de alarma para pulsarse en caso de emergencia, que se iluminará cuando sea pulsado para advertir al usuario de su activación y así, en caso de ser una falsa alarma o una pulsación por descuido, pueda desactivar el envío de

dicha alarma al centro pertinente. Se ha elegido el color rojo para este botón ya que instintiva y convencionalmente se le asocia a situaciones de peligro.

La otra parte de la interfaz de usuario es el módulo táctil que está formado por cuatro botones pertenecientes a la serie UB de Nikkai Switches. Uno actúa como botón de emergencia (luminoso) y el resto permite el desplazamiento por los menús que aparecen en pantalla y la selección de la opción deseada.

## **NIKKAI SWITCHES**

Éste ha sido el fabricante elegido para nuestra botonera y para ello se siguió nuevamente criterios de consumo y tamaño. Serán de tipo “push-to-break”, es decir, se hace contacto cuando se pulsa el botón y se deja de hacer contacto cuando no se pulsa el botón. En concreto se han elegido de la serie **UB25SKG03N-A** para los no sumisos y de la serie **UB25SKG035C-CC** para el botón luminoso, tratándose este último de un switch de tipo DPDT (Double Pole Double Throw).

### **4.1.8.1.- Interfaz con el DSP**

Ya mencionamos [102] que el DSP 6727 solo consta de tres interrupciones, por ello se ha optado por emplear una única interrupción para los cuatro botones que conforman la botonera. De forma que la interrupción que le llega al procesador es la OR de las cuatro salidas de los botones, de manera que siempre que se pulse un botón se requerirá la atención por parte del DSP que posteriormente y por inspección de las líneas de entrada salida de propósito general a las que están conectadas determinará cual de los cuatro botones fue pulsado y obrará en consecuencia. De las tres interrupciones externas disponibles esta interfaz utiliza AMUTEIN0 a través del pin *SPIO\_SIMO* perteneciente al periférico SPI0. Lo podemos ver en las figuras del siguiente apartado.

### **4.1.8.2.- Esquemático de la botonera**

Vemos el botón luminoso, uno de los tres no luminosos y la generación de la interrupción asociada a los cuatro.

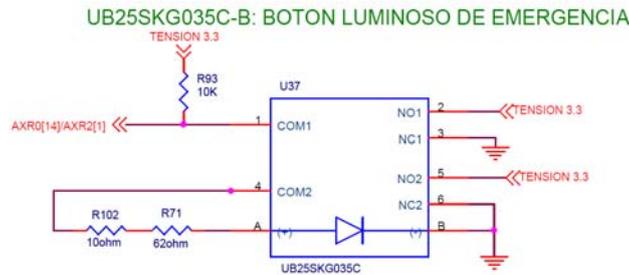


Figura 4.13.- Esquemático botón luminoso

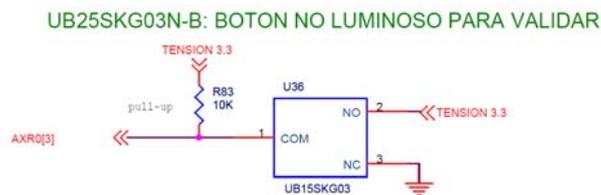


Figura 4.14.- Esquemático botón no luminoso

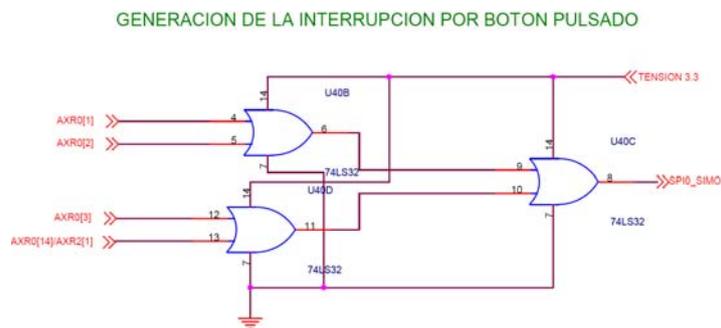


Figura 4.15.- Generación Interrupción Botón Pulsado

#### 4.1.9.- Módulo Audio

El módulo audio es el último bloque de la interfaz de usuario junto con la botonera y la pantalla LCD. Su función es reproducir mensajes sonoros para información (recordatorio toma de medicación, aviso de batería baja, etc.) y tranquilidad del paciente monitorizado, por ejemplo indicarle en caso de caída que no se preocupe puesto que una ambulancia se dirige a su hogar.

Está compuesto por altavoz, amplificador y codec de audio. La señal de sonido digitalizada llegará al codec a través del periférico McASP del DSP procedente de la memoria Flash, donde se encuentra almacenada en el formato correspondiente. La

configuración del codec se hará por medio del periférico SPI0 a 6MHz. Recordar que esta interfaz SPI era compartida también por ambos transceptores, por ello se adjuntará la tabla de verdad para el caso del codec

El periférico McASP [102, 119] funciona como un puerto serie de audio de propósito general, optimizado para las necesidades de aplicaciones de audio multicanal. Es útil tanto para protocolos IIS (Inter – Integrated Sound) como para DIT (intercomponent Digital audio Interface Transmisión). Recordar que en entre sus características principales destaca que permite la conexión directa sin lógica adicional (glueless) a convertidores analógico-digitales (ADC), convertidores digital analógicos (DAC) y CODEC entre otros dispositivos.

Pasamos ahora a comentar resumidamente los componentes que forman el módulo de audio.

### **CODEC TLV320AIC23B de TI**

El codec junto al amplificador se han elegido de TI debido a que es el que habitualmente emplea dicho fabricante en sus kits de desarrollo para los DSP de la familia 6000 y se ajustaba perfectamente a nuestras necesidades y requisitos para el servidor personal.

#### **Interfaz con el DSP**

Es el dispositivo TLV320AIC23B de Texas Instruments. Las funciones de los terminales más característicos son:

- BACK → reloj de bit de la interfaz I2S. Se conecta a la línea ACLKX1 del DSP perteneciente a un periférico McASP.
- DIN → entrada de datos serie al DAC estéreo perteneciente a la interfaz I2S. Conectada a la línea AXR0[13]/AXR1[0] del DSP perteneciente a un periférico McASP.
- LRCIN → reloj serie de palabra de la interfaz I2S. Conectado a la línea AFSX1 del DSP perteneciente a un periférico McASP.
- ROUT → Línea de salida del canal derecho. Conectada al amplificador de audio.
- SCLK → reloj serie de la interfaz SPI. Conectado a la línea AXR0[7]/SPI1\_CLK del DSP.
- SDIN → datos serie de la interfaz SPI. Conectado a la línea AXR0[9]/AXR1[4]/SPI1\_SIMO del DSP.

Esta sería la tabla de verdad para la activación de la señal de habilitación de esclavo para el CODEC del periférico SPI1.

SPI_SCS	AXR0[4]	UHPI_HD[6]	CODEC
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	-	-	1

Tabla 4.4.- Tabla de la verdad para SPI\_SCS\_CODEC

### Esquemático del CODEC

La salida ROUT del codec pasará a alimentar al amplificador.

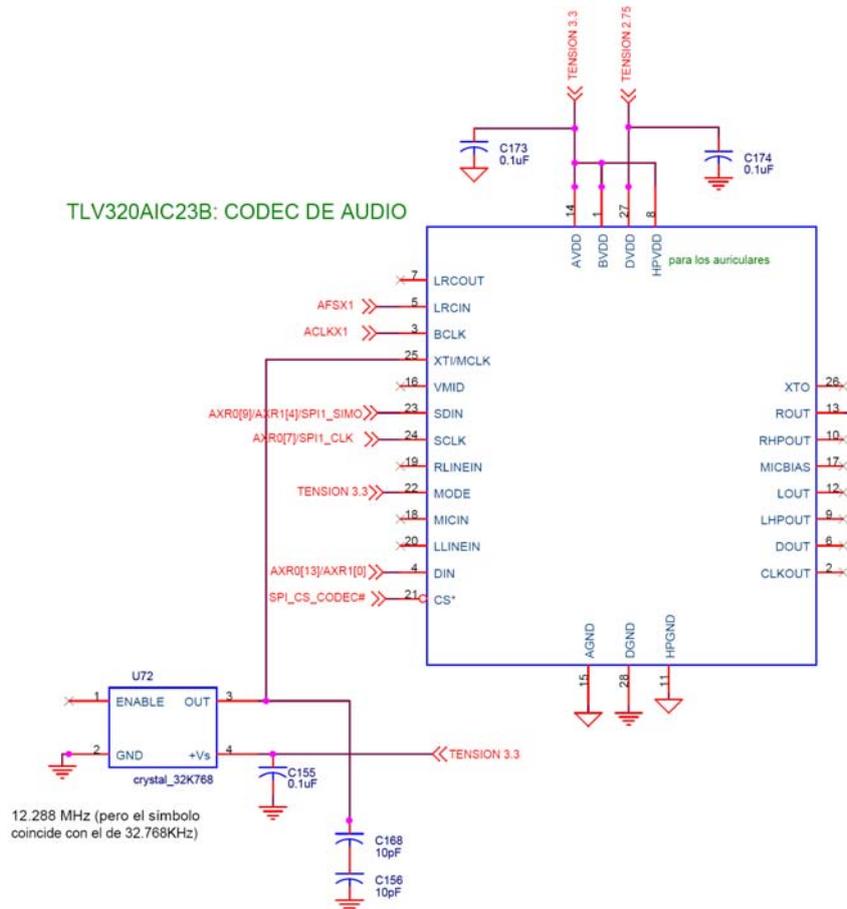


Figura 4.16.- Esquemático del CODEC

## AMPLIFICADOR DE AUDIO TPA302 DE TI

### Interfaz con el DSP

Los pines más importantes del amplificador son los siguientes:

- Vo1 → salida del amplificador que está conectada al altavoz.
- SHUTDOWN → debe mantenerse a nivel bajo durante la operación normal del amplificador. Está conectado a la línea ACLK0 del DSP.
- IN1 → entrada de la señal analógica de audio proveniente del codec.

El módulo de audio también forma parte de la interfaz de usuario y está compuesto por un altavoz de CUI INC, un amplificador (TPA302) y un codec de audio (TLV320AIC23B), ambos de TI. La configuración del codec se realiza a través de la interfaz SPI a 6MHz. Comparte esta interfaz con los transceptores Bluetooth y Zigbee, actuando también como esclavo. Se reproducen mensajes de audio que informan al paciente de alarmas, estados de baja batería, o le confirman que se ha alertado al personal sanitario correspondiente

### Esquemático del amplificador de Audio

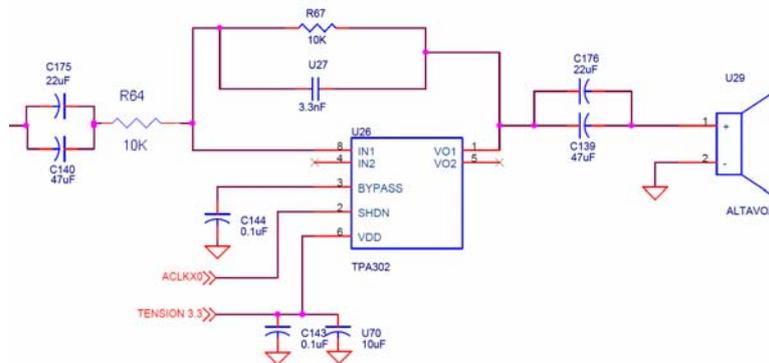


Figura 4.17.- Esquemático del Amplificador de Audio

### ALTAVOZ CUI INC CDMG 13008L-02

La oferta en cuanto a altavoces es muy variada. Nuevamente se ha acudido a requisitos de mínimo consumo y tamaño a la hora de seleccionar al candidato de CUI INC. Algunas de sus características tanto mecánicas como de otras índoles son:

- Dimensiones: 13mm de diámetro y 2mm de espesor.
- Consumo de potencia de 0.2W, como máximo 0.4W.
- Peso: 0.7g.

#### 4.1.10.- Módulo de Control y Supervisión

Dentro de este módulo vamos a distinguir a su vez el Watchdog o perro guardián de Maxim (MAX6374), que supervisa la actividad del microprocesador y reinicia el sistema en caso de operación incorrecta, los dispositivos de TI que monitorizan la batería (BQ27200) y controlan su proceso de carga (BQ24010), y los encargados del acondicionamiento de los distintos niveles de alimentación del PSE (TPS63000/63001). Empezamos con el módulo vigía de Maxim.

##### 4.1.10.1.- MAX6374 de MAXIM

Aparte de los módulos internos watchdog que poseen el DSP o el transceptor Zigbee, se ha considerado vital en el diseño tener uno externo a todos los componentes del PSE para poder monitorizar el estado de aquellos dispositivos que no dispongan de módulo watchdog propio, como por ejemplo el LCD. La función de este módulo es notificar errores software y detectar si el sistema se ha quedado bloqueado o “colgado”, procediendo a un reset general para que el servidor regrese a un estado de funcionamiento normal conocido

Se ha elegido un módulo comercial diseñado exclusivamente para este fin puesto que de entre las soluciones integradas y comerciales, es la que presentar un menor tamaño y consumo respecto a otras que requieren el uso de contadores, circuitos integrados y otros elementos de lógica digital.

El dispositivo seleccionado es el MAX6374, que configuramos para que nos proporcione un pulso de reset de 1ms, un tiempo de *timeout* de un segundo y un *tdelay* de 60 segundos, i.e., el dispositivo empieza a funcionar transcurridos 60 segundos.

##### Interfaz con el DSP

Veamos las funcionalidades de los terminales del módulo vigía:

- WDI → Entrada al módulo. Si no entra ningún pulso durante el periodo de timeout, la salida WDO dispara un pulso de 1ms y se reinicia el contador interno. Conectado al DSP mediante AXR0[6]/  $\overline{\text{SPI\_ENA}}$  .
- SET0,1,2 → Entradas lógicas para la selección del retraso de startup y el periodo de timeout del módulo.
- WDO → Salida del módulo, que junto con la proveniente del botón de reset externo genera la señal de reset del sistema que irá a los componentes del PSE.

## Esquemático del Watchdog

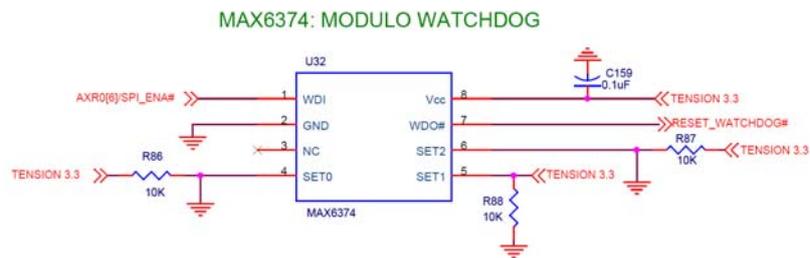


Figura 4.18.- Esquemático del Watchdog

### 4.1.10.2.- BQ27200 de TI

Permite que el sistema adquiera el estado de carga exacto de la batería. Para calcular la capacidad restante de una pila se puede medir únicamente su tensión, no obstante, el error cometido es elevado. La ventaja de estos circuitos es que permiten al usuario usar tanta carga de la batería como sea posible sin ser muy conservador porque la estimación de la capacidad es más precisa, como podemos observar en [120], donde se motiva su elección respecto de otras soluciones comerciales y su justifica su coste e integración en los packs de batería dada su extensivo uso en dispositivos portables. Se ha acudido de nuevo a Texas Instruments porque presenta una gran variedad de estos dispositivos con gran compatibilidad y disponibilidad, así como bajo precio. El DSP podrá acceder a dicho estado consultando los registros de una memoria EEPROM de que dispone el Gas Sauge, a través de un puerto I2C.

### Interfaz con el DSP

Se trata del dispositivo BQ27200 de TI. Las funciones de sus terminales son:

- BAT → entrada que testea la tensión de la batería.
- SCL → entrada para el reloj serie (I2C). Conectado a la línea SPI0\_SCS/I2C1\_SCL del DSP
- SDA → entrada de datos serie (I2C). Conectado a la línea SPI0\_ENA/I2C1\_SDA del DSP.
- SRN → entrada para medir corriente (negativa).
- SRP → entrada para medir corriente (positiva).

### Esquemático del Gas Sauge

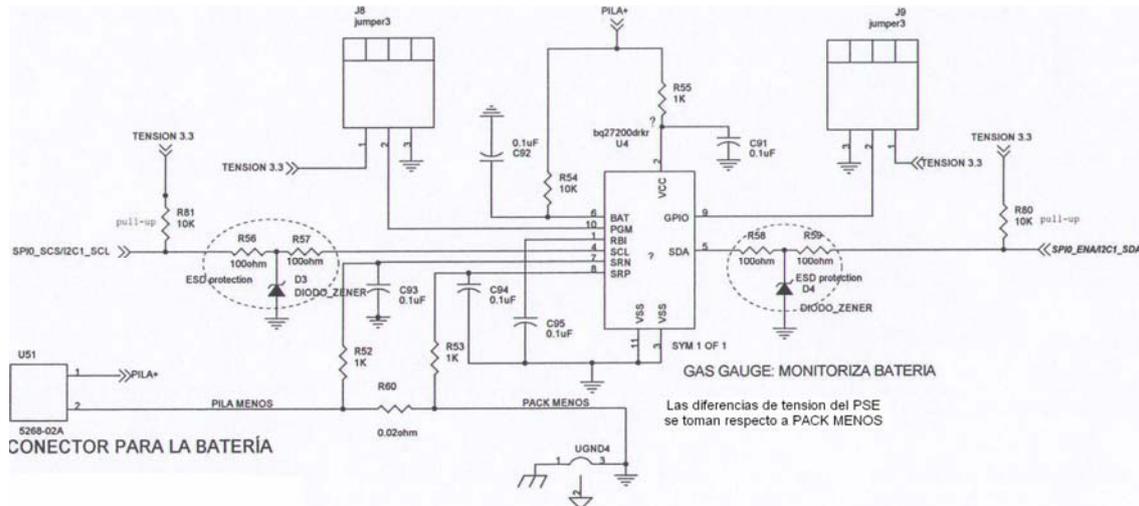


Figura 4.19.- Esquemático del Gas Saugue

#### 4.1.10.3.- BQ24010 de TI

Este componente va a encargarse de realizar la carga de la batería al tomar corriente de un adaptador AC/DC. Se vuelve a tratar de un dispositivo de Texas Instruments. En este caso se añaden tres LEDs de colores para indicar el progreso en el proceso de carga:

- Led Rojo → Al encenderse se indica que al adaptador está presente y conectado a nuestro prototipo mediante un conector de CLIFF CONNECTORS. En este caso el PSE pasa a alimentarse directamente desde el adaptador, mientras se realiza el proceso de carga de la batería.
- Led Verde → Al encenderse indica el comienzo del proceso de carga. Se apaga cuando finaliza.
- Led Naranja → Se enciende cuando finaliza el proceso de carga.

#### Interfaz con el DSP

Comentamos los terminales más importantes del dispositivo:

- BAT → Entrada para la tensión de la batería.
- IN → Tensión de carga de entrada que tiene que ser conectada a VCC.
- ISET → Punto de establecimiento de la corriente de carga.
- OUT → Salida de corriente de carga. Conectada a la batería.
- PG → Salida a colector abierto de estado power good. Indica cuándo está presente el adaptador. Conectado a la línea UHPI\_HD[2] del DSP (sirve para la decidir qué fuente deja pasar el multiplexor para alimentar el PSE: el adaptador o la batería) y al LED color rojo.

- STAT1 → Salida a colector abierto de estado de carga 1. Se conecta a la línea UHPL\_HD[0] del DSP y al LED color verde.
- STAT2 → salida a colector abierto de estado de carga 2. Se conecta a la línea UHPL\_HD[1] del DSP y al LED color naranja

### Esquemático del Cargador de Batería

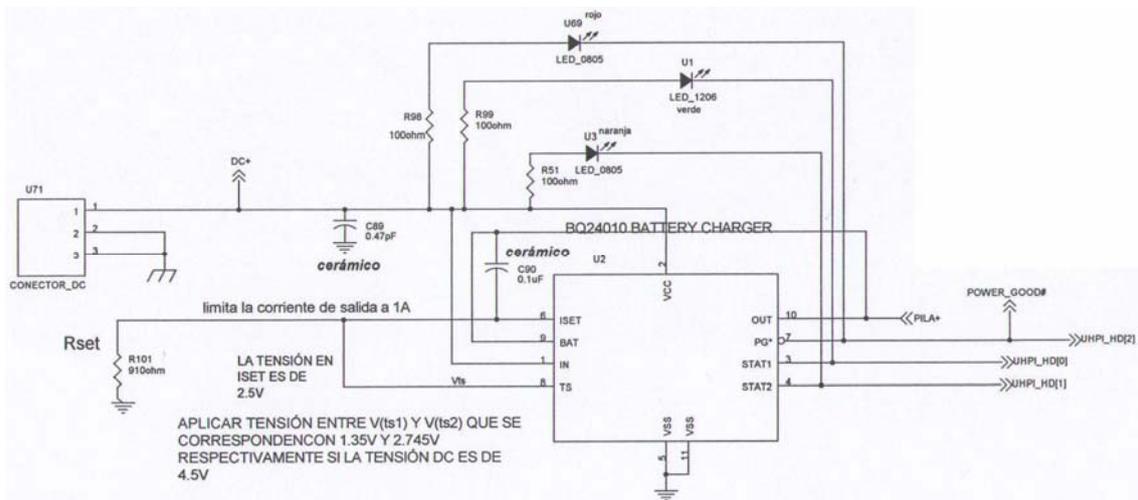


Figura 4.20.- Esquemático del Cargador de Batería

#### 4.1.10.4.- Generación del reset del sistema

Merece mención aparte el circuito de generación del reset general del sistema dada su importancia en el módulo de control y supervisión.

UB25SKG03N-A: BOTON NO LUMINOSO PARA RESET MANUAL

GENERACION DE LA SEÑAL DE RESET GENERAL DEL SISTEMA

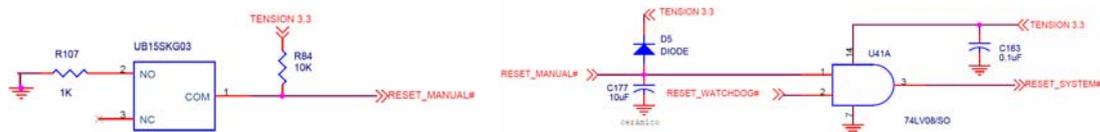


Figura 4.21.- Esquemático del Reset del Sistema

El objetivo del condensador de 10µF que se conecta a la señal de entrada RESET\_MANUAL es formar, junto a la resistencia de 1KOhm del pin 2 del botón, un circuito RC que garantice que no esté demasiado tiempo a nivel bajo el condensador. Evitamos así un reset general excesivamente largo que paralizaría el funcionamiento del sistema.

Por otro lado la finalidad del diodo D5 es que es provocar un reset rápido sin tener que esperar a que lo haga el reset del watchdog, que recordemos tiene un tdelay de 60

segundos, en caso de que por accidente se desconecte la batería del PSE. En este caso la señal TENSION 3.3 se iría a cero voltios, el diodo conduciría y el condensador se descargaría rápidamente activando la señal  $\overline{\text{RESET\_MANUAL}}$  y provocando el reset general del sistema.

#### **4.1.11.- Otros Aspectos del Diseño Hardware**

Primeramente señalar que por motivos de extensión de la memoria no se han comentado ciertos elementos que no tienen la mismas peculiaridades que los anteriores, si bien son vitales para el correcto funcionamiento del servidor personal en un caso (elementos reguladores de tensión de continua, multiplexor de potencia) y para su depuración y testeado de errores en el otro (conectores y jumpers de observación).

Una vez comentado esto, pasamos a comentar detalles conviene señalar otras consideraciones que se tuvieron en cuenta en el desarrollo del prototipo preindustrial del servidor personal.

##### **4.1.11.1.- Colocación de los componentes en la placa**

La gran mayoría de los dispositivos portables comerciales actualmente en el mercado, como por ejemplo PDAs o teléfonos móviles, tiene una disposición de los componentes bastante similar, que además sigue en buena parte el sentido común. A fin de cuentas de lo que se trata es de agrupar elementos por su funcionalidad, siempre evitando problemas de interferencias, acoplos y otros efectos adversos.

En concreto, como se puede ver en la siguiente figura adjunta, se distinguen las siguientes partes principales:

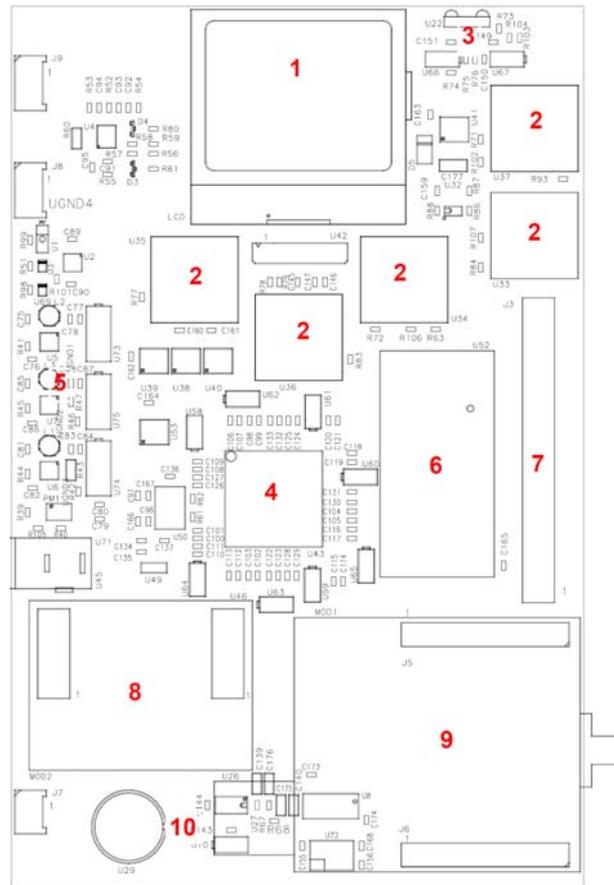


Figura 4.22.- Colocación de los componentes en la placa

- En la parte superior tenemos la pantalla LCD (1) y la botonera (2). En el extremo superior derecho está el transceptor infrarrojo (3)
- En la parte central (4) está el DSP junto con toda su circuitería de desacoplo asociada.
- A la izquierda del DSP (5), en la parte central, tenemos los componentes asociados al sistema de alimentación del servidor personal.
- A la derecha del DSP, en la parte central, tenemos el conector para la memoria Flash (6) y el conector de observación para monitorizar las señales más importantes del PSE (7).
- Debajo del DSP tenemos a la izquierda los conectores para el módulo Zigbee (8) y a la derecha los del módulo de evaluación Bluetooth (9).
- Finalmente en la parte inferior izquierda podemos ver los integrantes del módulo audio: codec, amplificador y altavoz (10).

Se ha buscado en todo momento optimizar el tamaño del prototipo preindustrial para que se acercase lo más fielmente posible a las dimensiones del dispositivo final. Actualmente dicho tamaño es de 156x100 mm.

Por otro lado cabe señalarse con relación a lo anterior, que uno de los componentes fundamentales del servidor personal, como es la batería y su correspondiente conector, se han dispuesto del otro lado de la placa para también optimizar el tamaño del PCB del PSE. La batería quedaría por tanto en la otra cara a la misma altura que la interfaz de usuario, para que se sitúe lo más cerca posible de la etapa de alimentación y las pistas hacia ella sean lo más cortas posibles con el objetivo de evitar efectos de EMIs.

#### **4.1.11.2.- Interferencias, masas y aspectos relacionados**

Sin lugar a dudas el problema de las interferencias y la colocación de las tierras en dispositivos electrónicos no es un asunto trivial, y buena cuenta de esto da la numerosa bibliografía relacionada con el tema [121]. Merece la pena detenerse el tiempo necesario para hacer una planificación correcta que evite estos efectos perniciosos antes de dar por finalizado nuestro diseño, y como veremos también va a estar muy relacionado con la colocación de los dispositivos de que consta el PSE.

Dada la distinta naturaleza del ruido analógico y digital (el digital es más pernicioso) hay que emplear planos de tierra separados para los componentes analógicos y los componentes digitales en nuestra futura placa PCB [122]. A ser posible deben situarse estos planos por debajo de dichos componentes, que por otra parte deberían estar agrupados según su naturaleza en la placa (en nuestro caso la parte analógica sería el módulo audio en la parte inferior izquierda; la parte digital sería el resto del circuito). Se recomienda también rellenar los huecos entre pistas del PCB con tierra y en caso de no poderse realizar planos de tierra, emplear pistas cortas y anchas en su lugar.

Nuestra solución ha sido:

- Aislar componentes analógicos con su propio plano de masas. Lo llamamos plano AGND y lo representamos con el símbolo 
- Aislar componentes digitales con su propio plano de masas. Lo llamamos plano DGND y lo representamos con el símbolo 
- Aislar componentes de potencia con su propio plano de masas. Lo llamamos PGND y lo representamos con el símbolo 

- Los pines de tierra analógicos y digitales del convertidor CDA deben estar conectados a la tierra analógica AGND, puesto que el acoplo de señales digitales en circuitos analógicos en este componente es crítico.
- En el caso de los reguladores, su datasheet indica que el pin DGND debe estar unido al PGND cerca del encapsulado. Creo que lo conveniente es que este punto de unión sea el de todas las tierras como indico a continuación.
- Unir las tres tierras (AGND, DGND, PGND) únicamente en un solo punto, esto se hace idealmente en la tierra común para la batería del circuito, punto de referencia desde el que se toman las diferencias de tensiones del PSE. En nuestro caso sería el punto PACK\_MENOS de la figura 4.19.

Otras recomendaciones que se han seguido hablan de que las etapas amplificadoras de los circuitos de audio estén cerca del altavoz, o que para los circuitos de regulación de tensión conmutados, que pueden ser fuentes de interferencias, se empleen pistas cortas y anchas.

#### **4.1.11.3.- Estado actual del dispositivo. Pasos a seguir.**

Actualmente tenemos la placa PCB del dispositivo PSE como podemos observar en la fotografía, en posición horizontal.

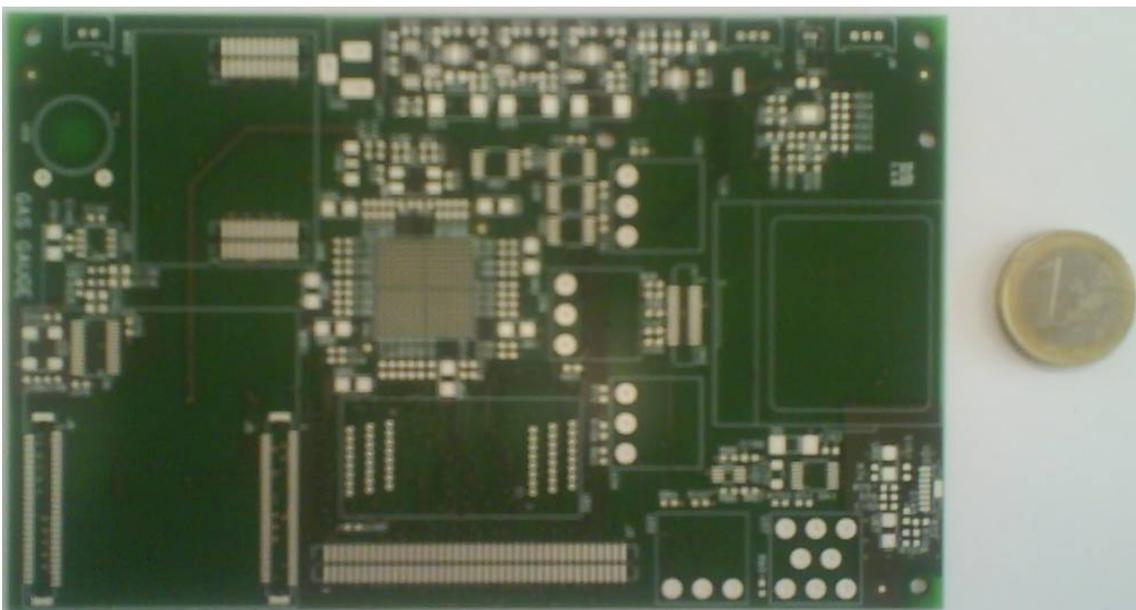


Figura 4.23.- Placa PCB del PSE

Cabe señalar que el PCB es de cuatro capas debido sobre todo a la complejidad del encapsulado del DSP 6727, que es del tipo Plastic Ball Grid Array (PBGA) de 256 terminales.

El siguiente paso, que es en el que actualmente estamos inmersos, consiste en realizar el montaje de los componentes SMD en la placa, así como aquellos que poseen sujeción mecánica, como es el caso del altavoz y la batería.

Una vez realizado esto se procederá a la etapa de pruebas y depurado del prototipo tanto a nivel hardware como la evaluación de las rutinas implementadas a nivel software.

Finalizada esta etapa podemos decir que nos encontraríamos dispuestos a realizar el diseño final del servidor personal, en el que podemos retirar los componentes de observación (jumpers y conector); quitar los módulos de evaluación de los transceptores e insertar en su lugar los transceptores y su lógica digital asociada una vez que se determine la antena que irá finalmente en el PSE; eliminar el socket de la Flash para insertarla directamente en la placa PCB; e incluso la búsqueda de una batería de un tamaño algo menor si es posible.

De esta manera tendríamos un dispositivo electrónico viable desde un punto de vista comercial. Sería mucho más pequeño, y podría utilizarse a modo de colgante o pulsera

## **4.2.- Arquitectura Software**

En este apartado se presentan los aspectos metodológicos y de diseño en el desarrollo del software del servidor personal para el control y procesado de señales provenientes de sensores biomédicos inteligentes para su funcionamiento 24/7 [123]. Una característica fundamental que se ha tenido en cuenta en su implementación es la capacidad de personalización en línea al paciente mediante la generación de conocimiento adaptativo en tiempo real. En ese sentido se ha perseguido una solución modular y abierta, que dé lugar a una fácil configuración del dispositivo para que se adapte a la incorporación y procesamiento de señales heterogéneas de sensores de muy diversa naturaleza, en lo que denominamos carácter multimodal del servidor personal.

Por otro lado, dado que el sistema operativo del PSE va ser ejecutado en un DSP de la familia 6000 de Texas Instruments, hemos debido ceñirnos a las prestaciones que este procesador nos proporciona para el desarrollo del software de control, que a todos los efectos podemos denominar SO de control o ejecutivo de control.

Las funciones principales que debe cumplir el sistema operativo del servidor personal son:

- Gestionar la comunicación con los periféricos: debe ser capaz de atender las exigencias de los distintos periféricos, principalmente en lo que respecta a los estándares de comunicación interdispositivo (I2C, SPI, etc.).
- Gestionar el funcionamiento interno del PSE: debe encargarse de tareas tales como la reprogramación del dispositivo y actualización del modelo matemático; revisión del estado del dispositivo y diagnóstico; gestión de la incorporación de nuevos sensores a la WPAN o la revisión del estado tanto del PSE en sí mismo como de los sensores. Esta función emplea la interfaz infrarroja para comunicarse con Host.
- Procesar en tiempo real las bioseñales procedentes de los sensores y obrar en consecuencia. Como ejemplo, y para el caso particular de la función de detección de caídas, este procesamiento permitirá la adaptación de los parámetros del algoritmo implementado en la IAU al usuario y contexto.

Para el desarrollo del software del PSE se ha seguido un diseño modular en el que las distintas aplicaciones se integran en un conjunto de hilos que se corresponden con las funcionalidades principales del servidor personal.

#### **4.2.1.- Materiales y Herramientas de Desarrollo Empleadas**

Con el objetivo de satisfacer las funciones mencionadas en el apartado anterior se han implementado unos módulos software empleando un conjunto de entornos de desarrollo [124]. Por un lado la herramienta Code Composer Studio v3.1 junto con un Kit de Desarrollo de TI (TMS320C6713 DSK) nos permite programar el DSP tanto en ensamblador como en lenguaje de alto nivel (C, C++) además de proporcionar varias utilidades de gestión interna del procesador y de los periféricos embebidos en el Kit mediante la utilidad DSP/BIOS. Por otro lado nos estamos sirviendo de Matlab y más concretamente de los paquetes Embedded Tools for TI C6000 DSP™ y Real Time Workshop™, que nos hacen posible compilar y ejecutar código C optimizado en el DSP para el desarrollo del hilo de procesamiento de las señales acelerométricas. Además, el paquete Link for Code Composer Studio™, nos permite comunicarnos con la placa de desarrollo del DSP y realizar un procesamiento paralelo de los datos que gestiona el DSP en Matlab.

Pasamos a comentar estas herramientas con algo más de detalle.

##### **4.2.1.1.- Kit de Desarrollo C6713DSK**

El kit de desarrollo C6713 DSK es una plataforma de desarrollo independiente de bajo coste que permite a los usuarios evaluar y desarrollar aplicaciones para la familia de DSP TI C67xx. El DSK viene con un conjunto de dispositivos en placa que permite una gran variedad de contextos de aplicaciones. Sus prestaciones clave son:

- Un DSP TMS320C6713 de TI operando a 225 MHz.
- Un codec estéreo AIC23.
- 16 Mbytes de memoria SDRAM.
- 512 Kbytes de memoria Flash (256 Kbytes en la configuración por defecto).
- Configuración del software en placa a través de registros implementados en CPLD.
- Opciones de arranque (boot) configurables.
- Conectores de expansión estándar para usar daughter cards.

- Emulación del protocolo JTAG a través de interfaz USB o por medio de un emulador externo.
- Adaptador de continua (+5V).

Que vemos en la siguiente fotografía:



Figura 4.24.- Kit de Desarrollo C6713DSK

Desde un punto de vista funcional, el DSP del DSK accede a los periféricos a los periféricos de que consta la placa a través del periférico embebido EMIF de 32 bits. La SDRAM, Flash y el CPLD están conectados al bus. Las señales EMIF también están conectadas a los conectores de expansión, que sirven para monitorizar la placa y conectar otras tarjetas de evaluación.

El DSP accede a señales audio analógicas a través de un codec AIC23 y tres conectores jack audio (entrada mic, entrada de línea, salida de línea y salida de auriculares). La salida analógica es conducida tanto a la línea de salida como al auricular. Empleamos McBSP0 para enviar comandos a la interfaz de control del codec mientras que McBSP se emplea para los datos de audio digitales. Tanto McBSP0 como McBSP1 pueden redirigirse hacia los conectores de expansión mediante software.

Un dispositivo lógico programable denominado CPLD se usa para implementar la lógica que mantiene conectados los componentes en la placa. El CPLD tiene un registro basado en la interfaz de usuario que permite al usuario configurar la placa al leer y escribir de estos registros.

El DSK incluye 4 LEDs y un switch DIP de 4 posiciones como interfaz de usuario. Ambos son accesibles al leer y escribir en los registros del CPLD.

El adaptador de continua de 5 voltios permite alimentar al circuito. Los reguladores de tensión conmutados proporcionan los +1.26V para el núcleo del DSP y los +3.3V para la E/S. La tarjeta se mantiene en estado de reset hasta que se alcanzan estos niveles de tensión.

Code Composer se comunica con el DSK a través de un emulador JTAG embebido en una interfaz USB con el Host. En el DSK también puede emplearse un emulador externo a través de un conector JTAG.

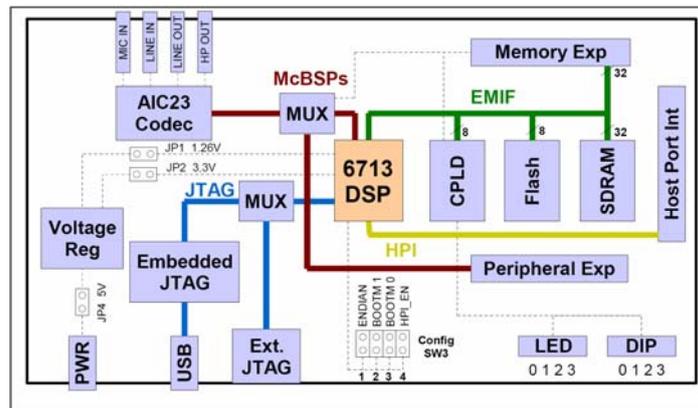


Figura 4.25.-.- Diagrama de bloques del DSK6713

#### 4.2.1.2.- Code Composer Studio

Code Composer Studio (CCStudio) es un entorno de desarrollo integrado (IDE) completo vía software que soporta las plataformas de DSPs de Texas Instruments y que forma parte de la estrategia de herramientas de desarrollo eXpressDSP Software de TI. El Code Composer Studio IDE v3.1 es el primer entorno de desarrollo inteligente que permite el desarrollo de aplicaciones multiprocesador y multiusuario para las familias TMS320C2000, TMS320C5000 y TMS320C6000. CCStudio integra todas las herramientas host y target en un entorno unificado, de forma que proporciona herramientas de generación de código, depuración e intercambio de datos en tiempo real. En suma es una aplicación (un entorno IDE) que corre en un host y que permite enlazar con la tarjeta emuladora o de evaluación, la cual tiene un TMS320, mediante un puerto JTAG. Es abierto, de forma que aparte de las aplicaciones que trae, permite la adición de plug-ins.

Veamos algunas de sus prestaciones más interesantes:

- Herramientas de generación de código: compilador C/C++, ensamblador y linkado.

- Gestor de proyecto visual integrado con interfaz de control de fuente, capaz de gestionar 1000 ficheros de proyecto.
- Drivers para emulación XDS510 y XDS560.
- Editor completamente integrado.
- Soporte de gestión de componentes para múltiples versiones de DSP/BIOS y herramientas de generación de código dentro de la IDE.
- Debugger de código fuente de interfaz común tanto para el simulador como para el emulador.
- Soporte para la herramienta DSP/BIOS™.
- RTDX™ para la transferencia de datos en tiempo real entre host y target.

En la siguiente figura se puede observar una captura de pantalla de la IDE.

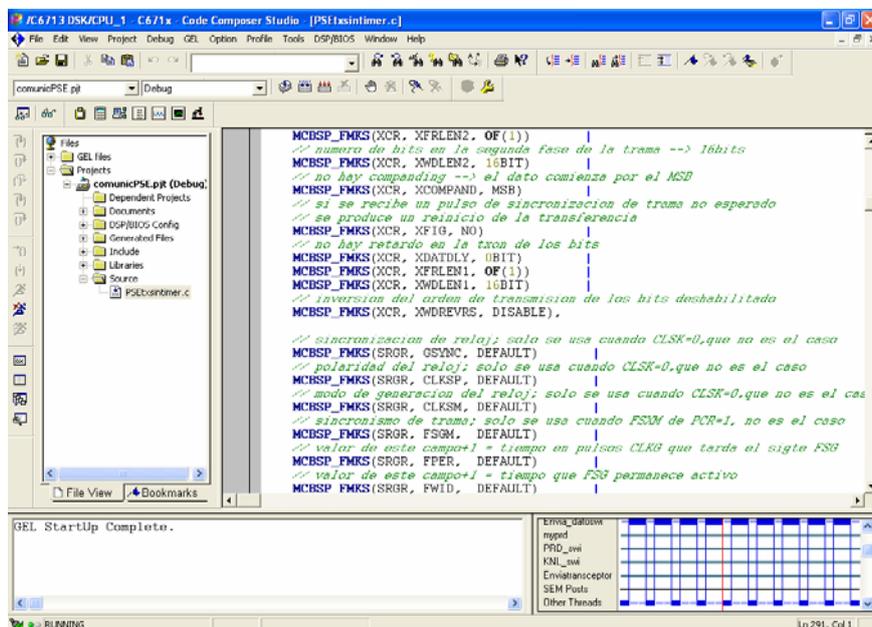


Figura 4.26. Entorno de Desarrollo Code Composer Studio

#### 4.2.1.3.- Paquetes de Matlab

Target for TI C6000™ v3.3 integra Matlab y Simulink con las herramientas de eXpressDSP y los procesadores C6000. Juntos, estos productos permiten desarrollar y validar algoritmos de procesamiento de señal desarrollando conceptos tales como generación de código automática, prototipado y empleo de sistemas embebidos en los procesadores TI C6000.

Al usar Target for TI C6000 y las herramientas de desarrollo de TI junto con el Link for Composer Studio, Real-Time Workshop y Real-Time Workshop Embedded Coder, se

puede generar una implementación en lenguaje C y en tiempo real del modelo Simulink. Target for TI C6000 soporta los DSPs en punto flotante de la familia T C67x, incluyendo los periféricos on-chip y on-board, además de numerosos kits de evaluación, como el C6713DSK. Juntas proporcionan una plataforma integrada para el diseño, simulación, implementación y verificación de sistemas de procesamiento de señal embebidos en procesadores TIC6000 estándar. Nos permiten implementar y validar diseños de sistemas de procesamiento de señal y algoritmos a través de código. Se puede generar lenguaje C, implementación en tiempo real del modelo o de simulink e invocar a code composer estudio para crear y descargar un ejecutable en el dsp para la evaluación en tiempo real.

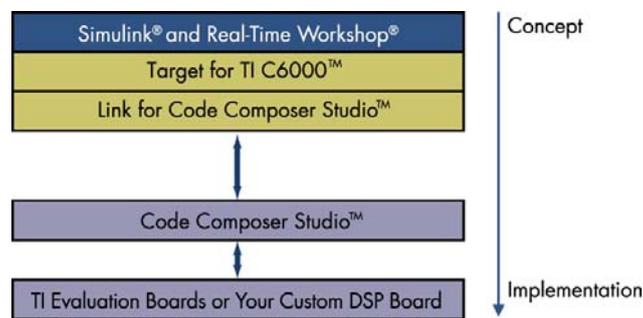


Figura 4.27.- Paquetes de Matlab

Sus prestaciones clave son:

- Automatizar la ejecución, testeo, verificación y validación del código generado en los procesadores TI C6000.
- Permitir la evaluación en tiempo real de los diseños de los sistemas en los kits de evaluación.
- Soporta integración con el software DSP/BIOS.
- Soporta prestaciones en modo externo para modelado en tiempo real y monitorización de aplicaciones embebidas.
- Soporta el perfilado (profiling) de código para identificar regiones de código generado disponible para la optimización manual.
- Proporciona acceso a nivel de bloques de los periféricos on-chip, como el ADC, DAC y la memoria del target
- Proporciona librerías especializadas para la comunicación de target y Host vía Ethernet.
- Proporciona código ensamblador optimizado para generación de código y simulación en las librerías 62x y 64x.

- Proporciona bloques para la implementación de esquemas de planificación de ejecución en tiempo real para tareas sincrónicas y asíncronas.

#### 4.2.2.- Aspectos de Diseño

Se ha realizado un diseño de las aplicaciones concurrente, mediante hilos, de manera que se maximice su robustez y que pueda liberarse al máximo de carga de procesamiento al DSP, que en funcionamiento normal debe estar el mayor tiempo posible disponible para ejecutar el hilo de procesamiento en tiempo real de los datos de los sensores.

Se ejecutan 3 hilos principales que se corresponden prácticamente con las funcionalidades anteriormente descritas:

- **Hilo de Gestión de los Periféricos.**
- **Hilo de Gestión del PSE.**
- **Hilo de procesamiento de señales.**

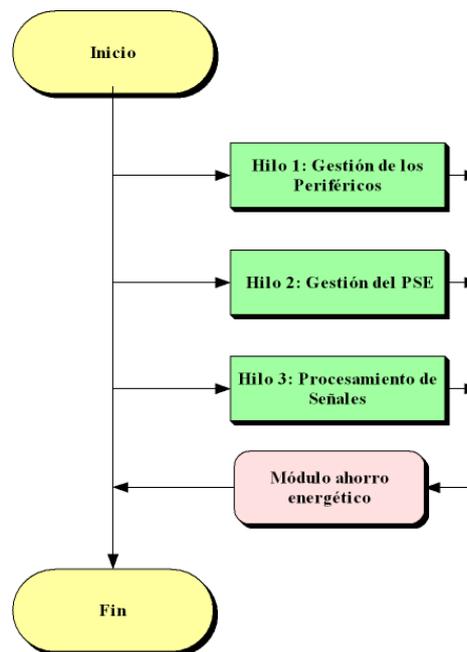


Figura 4.28.- Diagrama de flujo de principal

Cuando ninguno de los hilos esté en ejecución o termine el hilo de procesamiento, que no es indefinido ya que finaliza cuando lo haga el correspondiente procesamiento, se ejecuta el módulo de ahorro energético, que da lugar a una disminución de la frecuencia

de conmutación de la lógica de estado de la circuitería CMOS del DSP y a un estado de inactividad de la CPU, a espera de una interrupción que lo saque de dicho estado [125]. La disponibilidad del sistema las 24 horas del día es fundamental, lo que obliga a supervisar el estado de la batería. Con este fin se ha desarrollado un módulo específico que, en condiciones de batería crítica, hace una salvaguarda del instante, evento, últimos datos capturados y otros datos de estado e identificadores del sistema con el objetivo de que el servidor personal retome la actividad en el mismo estado en cuanto recupere energía. Esta salvaguarda junto con el hilo de procesamiento de señal se sitúa en la memoria Flash siguiendo un protocolo de almacenamiento óptimo conforme al diseño del hilo de procesamiento. En la siguiente figura podemos observar el diagrama de flujo de dicho módulo.

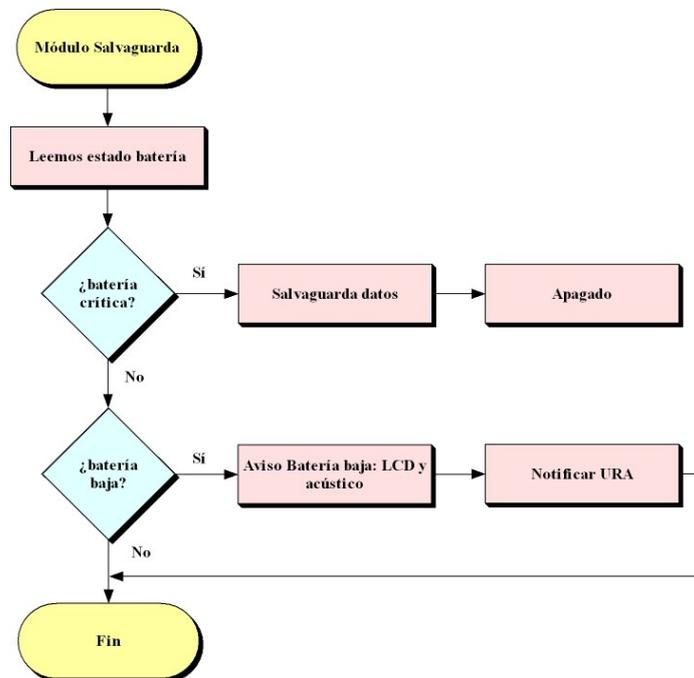


Figura 4.29.- Diagrama de flujo del módulo salvaguarda

La ocupación de memoria ante este evento para el caso de un único sensor acelerométrico controlado es de 30 bytes asociados al último dato capturado por la IAU (10 muestras de 1 byte en Complemento a Dos por cada eje de aceleración). La información de estado propia del PSE a almacenar no supera las decenas de bytes. En base a estos números, la memoria Flash utilizada en el diseño no introduce ninguna limitación en el número de sensores gestionados por el PSE.

La mayor fuente de ocupación de datos viene dada por la posibilidad de grabar voz en la memoria Flash y que después pueda reproducirse a modo de mensajes de advertencia e información. En una primera propuesta cinco mensajes de seis segundos de duración, a una tasa de muestreo típica del codec de 8 KHz ocuparían 240000 bytes, lo que supone un 22.8% de la memoria Flash. De este modo la memoria puede ser utilizada también para la ejecución del hilo de procesamiento.

Por otra parte, el DSP dispone de una memoria ROM desde la que arranca la aplicación principal y que alberga los módulos de gestión básicos no modificables, que se encargan principalmente de la activación y configuración inicial de los periféricos y de lanzar los hilos. El tamaño actual del código, todavía en fase de depurado y optimización es de un 35 % de los 384 Kb de tamaño de memoria ROM.

Dada la cualidad de monitorización y supervisión del servidor personal, existen numerosas aplicaciones que deben ejecutarse con una determinada periodicidad. Esta frecuencia de operación es modificable y se adapta a las necesidades de la WPAN, como en el caso de las proximidades temporales a una detección de impacto, en la que la frecuencia de comunicación con la IAU (Zigbee) se incrementa pasando de 1Hz al doble\*. En la siguiente tabla podemos ver algunas de ellas:

<b>Módulo</b>	<b>Frecuencia de Ejecución</b>	<b>Comunicación</b>
Zigbee	1 Hz*	SPI
Bluetooth	1 Hz	SPI
Flash	Continua	GPIO
LCD	Determinada por el usuario	I2C
Infrarrojo	Determinada por el usuario	IdDA 1.4 emulado en GPIO
Botonera	Determinada por el usuario	Interrupción en GPIO
Batería	10 veces al día	I2C
SelfTest	1 vez al día	Varios

Tabla 4.4.- Módulos software del PSE

Donde en la columna de comunicación se especifican los estándares interdispositivo empleados, ya sean emulados a través de líneas de E/S de propósito general (prefijo GPIO) o mediante periféricos específicos embebidos del DSP, como es el caso del I2C o SPI.

A continuación comentaremos con más detalle los aspectos más importantes del software implementado.

### **4.2.3.- Programación en hilos en el DSP**

Muchas aplicaciones de los DSP en tiempo real deben realizar un número de funciones parecidas pero no relacionadas en el mismo tiempo, ya sea en respuesta a eventos externos como la disponibilidad de los datos o en presencia de una señal de control. Es tan importante la función en sí misma como cuándo se ejecuta. Este tipo de funciones se denominan hilos.

Para poder implementar hilos, vamos a hacer uso de la utilidad DSP/BIOS [126], que es un kernel escalable en tiempo real disponible para los DSP TMS320C. Se diseñó para usarse por aplicaciones que requieren planificación y sincronización, comunicación host-to-target o instrumentación en tiempo real. DSP/BIOS nos proporciona programación concurrente multi-hilo con reemplazo, abstracción hardware, análisis en tiempo real y herramientas de configuración.

Dentro del kernel DSP/BIOS, se definen los hilos de una forma amplia en el sentido de incluir cualquier conjunto de instrucciones independientes ejecutadas por el DSP. Un hilo es por tanto un punto único de control que contiene una subrutina, una rutina de servicio de interrupciones (ISR) o una llamada a una función. DSP/BIOS permite a nuestras aplicaciones estructurarse como un conjunto de hilos, cada uno de los cuales desempeña una determinada función. Los programas multihilos se ejecutan en un único procesador al permitir a hilos de alta prioridad reemplazar hilos de menor prioridad y permitiendo varios tipos de interacción entre hilos, incluyendo bloqueos, comunicación y sincronización.

La gran ventaja de organizar las aplicaciones en tiempo real de esta forma, como contrapartida al esquema clásico de un único bucle centralizado con mecanismo de polling o sondeo, es que son más fáciles de diseñar, implementar y mantener.

DSP/BIOS proporciona soporte para numerosos tipos de hilos de programas con diferentes propiedades. Los tipos de hilo (de más alta a más baja prioridad) son:

- Interrupciones Hardware (HWI): se disparan en respuesta a un evento externo asíncrono ocurrido en la periferia del DSP. La duración de su ejecución (*deadline*) varía entre 2 y 200 us. Las HWI sólo pueden ser reemplazadas por otra interrupción a menos que las interrupciones sean desactivadas. En las HWI

se incluyen las denominadas funciones CLK. Estas funciones se ejecutarán en cada periodo de tick, con lo que su carga de procesamiento deber ser mínima.

- **Interrupciones Software (SWI):** se usan para planificar eventos con un deadline mayor a los 200 us, permitiendo a las HWI ceder procesado menos crítico a un hilo de menor prioridad. Las SWI tienen menos prioridad que las HWI. Existen 14 niveles de prioridad para las SWI. Las interrupciones Software pueden reemplazarse por otra SWI de mayor prioridad o por una HWI. No pueden bloquearse. En las SWI se incluyen las denominadas funciones PRD, de ejecución periódica con periodo un múltiplo entero del periodo de tick. Ambos periodos son configurables.
- **Tareas (TSK):** difieren de las SWI en que puede esperar (bloquearse) durante su ejecución hasta que los recursos necesarios estén nuevamente disponibles. DSP/BIOS proporciona un número de estructuras que puede usarse para comunicación entre tareas, sincronización y exclusión mutua para el acceso a memoria compartida, como son las colas, semáforos, buzones (mailboxes) y cerraduras (locks). Las tareas tienen menos prioridad que las interrupciones software. Hay 15 niveles de prioridad. Las tareas pueden reemplazarse por una HWI o una SWI. Pueden bloquearse en espera de disponibilidad de recursos y de hilos de menor prioridad. DSP/BIOS reemplaza inmediatamente la tarea en curso si otra tarea de mayor prioridad pasa a estar disponible para ejecutarse.
- **Hilos en background (IDL):** son los que se ejecutan cuando no lo hace ninguno de los anteriores, con lo que sólo deben procesar funciones de pequeño deadline. El hilo de background es el de menor prioridad de todos. Solo se ejecuta en un bucle cuando la CPU no está ocupada ejecutando otro hilo.

En la siguiente tabla se resumen las características principales de los hilos:

Características	HWI	SWI	TSK	IDL
Prioridad	Mayor	2ª Mayor	2ª Menor	Menor
Nº de niveles de prioridad	Depende del DSP	15	16	1
Puede dar paso y bloquearse	No, se ejecuta hasta el final salvo reemplazo	No, se ejecuta hasta el final salvo reemplazo	Sí	No
Estados de	Inactivo, preparado,	Inactivo, preparado,	Preparado, en	Preparado, en

<b>ejecución</b>	en ejecución	en ejecución	ejecución, bloqueado, terminado	ejecución
<b>Pila</b>	Pila del sistema (única)	Pila del sistema (única)	Pila de la tarea (una por tarea)	Pila del sistema (única)
<b>Comparte datos con hilos mediante</b>	Cadenas, colas, tuberías, variables globales	Cadenas, colas, tuberías, variables globales	Cadenas, colas, tuberías, cerraduras, mailboxes, variables globales	Cadenas, colas, tuberías, variables globales
<b>Sincronismo con hilos mediante...</b>	No aplicable	SWI mailbox	Semáforos, mailboxes	No aplicable

Tabla 4.5.- Comparativa de Hilos

Pese a que la terminología empleada por los desarrolladores de estas opciones software las denomina a todas con el mismo nombre, el concepto tradicional de hilo es el que representa la tarea TSK en el sentido de que son más complejas que el resto de las opciones. Comparadas con las SWI, solo las tareas TSK pueden esperar un evento para su ejecución (en lo que denominamos bloqueo), como la disponibilidad de un recurso concreto. En cambio, todas las entradas que necesite un SWI deben estar disponibles cuando el programa principal ejecuta una SWI. En la siguiente figura se indica los estados en los que puede estar una tarea TSK, y las funciones que se involucran en su gestión.

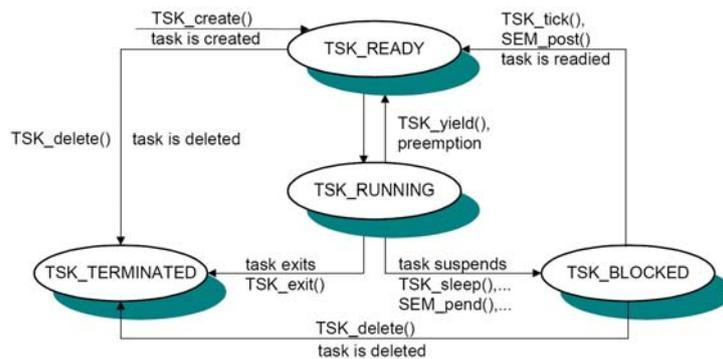


Figura 4.30.- Funciones TSK involucradas en la gestión de las tareas TSK

Para lo que a nuestra aplicación afecta, se han empleado todas las prestaciones que nos ofrece DSP/BIOS. Esto es, se han empleado tareas HWI para atender a interrupciones tales como las asociadas al módulo I2C. Se han usado tareas SWI que se disparan para descargar la rutina HWI, como en el caso de la interrupción en el puerto McBSP. Por

otro lado, las funciones PRD se han utilizado para la ejecución de los módulos periódicos de los que hablábamos en la tabla 4.4.

Por último comentar que para los tres hilos principales expuestos en el apartado anterior, se ha establecido la misma prioridad. Como consecuencia su planificación vendrá dada en base a la filosofía de “primero en llegar, primero en servirse”. De todas formas estas prioridades se pueden modificar durante la ejecución, por ejemplo se puede establecer una mayor prioridad para el hilo de procesamiento si se quiere realizar un estudio más profundo en lo que a carga computacional se refiere, para que no sus resultados no se vean excesivamente retardados por la ejecución de los otros dos hilos.

#### **4.2.4.- Módulo de Ahorro Energético**

Como comentamos anteriormente, este módulo entra en funcionamiento a modo de función en ensamblador, cuando ninguno de los otros módulos que integran los hilos esté ejecutándose, con el objetivo de que el DSP no consuma potencia si no realiza operaciones de interés.

La mayor parte de la energía de operación de la lógica CMOS [125] se disipa durante la conmutación de circuitos de un estado lógico a otro. Previniendo la conmutación de algunos de los dispositivos lógicos, los modos power-down (bajo consumo) se usan para conseguir ahorros energéticos significantes sin perder ningún dato del contexto de operación. Estos tres modos de operación, PD1, PD2 y PD3, disponibles en los DSP de la familia 6000, desarrollan esta funcionalidad

El modo Power-down PD1 bloque las entradas de reloj internas de la periferia de la CPU, impidiendo que la mayor parte de su lógica conmute. PD1 efectivamente “apaga” la CPU. Durante PD1, las transacciones DMA/EDMA pueden proceder entre periféricos y memoria interna.

En cambio, durante PD2, la estructura de reloj on-chip entera (incluyendo buffers múltiples) se detiene en la salida del PLL. PD3 es como PD2 pero también desconecta la entrada de reloj externa (CLKIN) hacia el PLL. Despertar de PD3 toma mas tiempo que de PD2 debido a que el PLL necesita ser reestablecido, tal y como hace cuando se “enciende” el dispositivo. Los modos PD2 y PD3 solo son abortados por un reset del dispositivo.

Es por este último detalle que de entre los power-downs nos interesará el PD1, ya que podemos programar una interrupción periódica y configurable, como la del timer, para

poder sacar a nuestro dispositivo del modo de bajo consumo sin necesidad de tener que forzar un reset externo que quedaría a manos del usuario.

El PD1 toma afecto 8 o 9 ciclos después de la instrucción que motivó el power down (al programar los bits PWRD del registro CSR). Para entrar en el modo power down empleamos el siguiente código

### Código

```
; ===== _halt =====
; ahorro energetico mediante Power Down PD

N                .set                4400h        ; despertamos PD1 por interrupt habilitada o no

_halt:
                MVC    CSR,B0            ; B0 <-- CSR
                MVK    N,B1
                OR     B0,B1,B0          ; B0 <-- B0 + N

                B      NextInst          ; ocultamos escritura en CSR con los NOP
                NOP

                MVC    B0,CSR            ; establecemos modo PD1

                NOP
                NOP
                NOP
NextInst:       NOP
                NOP                    5

                .end                    ; proseguimos ejecución normal
```

Además de los modos de ahorro energético power-down descritos en este documento, la instrucción IDLE proporciona menor consumo energético al ejecutar numerosos NOPs. La instrucción IDLE termina solo al servirse una interrupción.

```
; ahorro energetico mediante bucle inoperante IDLE
_halt:

L1:
                IDLE

                NOP
                NOP
                NOP
                NOP
                NOP
                NOP

                NOP    5
                B      L1

                .end
```

A pesar de conseguir un supuesto menor consumo, nos hemos decantado por la opción PD1 ya que nos parece más fiable emplear un modo de ahorro energético diseñado a tal

efecto que dejar inoperante la CPU mediante instrucciones IDLE que dotan de menor estabilidad a la aplicación software.

#### 4.2.5.- Módulo de Salvaguarda

Este módulo es de gran importancia en el dispositivo PSE permitiendo hacer una salvaguarda en la Flash de los datos más importantes en caso de estado de batería crítica. Estos datos pueden ser:

- Identificador de red.
- Identificador de sensor/es.
- Últimos datos de evaluación procesados.
- Identificador de evento batería crítica.
- Información de estado del PSE

En todo caso y como ya hemos mencionado, no supondría un tamaño mayor a las decenas de bytes: una estimación de 100 bytes es procedente. Teniendo en cuenta que cada escritura en memoria son dos accesos de 70ns, i.e., 140 ns, y que en el peor de los casos almacenaremos los datos byte a byte, tendríamos un tiempo total de duración para salvaguarda de datos de 14 us.

A continuación vemos parte del código implementado, tanto empleando C como lenguaje ensamblador para que sea todavía más óptimo en su ejecución temporal.

#### Código

```
FLASH_START .equ 0x90000400 ;direccion de comienzo de la flash donde se copiarian los datos
CODE_START .equ 0x00000400 ;direccion de memoria interna (después del BOOT_SIZE)
CODE_SIZE .equ 0x00003000 ;tamaño del codigo a copiar
.....
_boot_start:
    mvgl FLASH_START,B4 ;pasamos al registro B4 la direccion de comienzo de memoria interna
    mvkh FLASH_START,B4

    mvgl CODE_START,A4 ;pasamos al registro A4 la direccion de Flash donde se va a copiar
    mvkh CODE_START,A4
    zero A1 ;ponemos a 0 nuestro contador interno
.....

_boot_loop1:
    ldb *B4++,B5 ;lectura de la memoria interna → B5
    mvgl CODE_SIZE-4,B6 ;B6 = BOOT_SIZE -1024
```

```
    add 1,A1,A1          ;A1+=1,incrementamos contador
||   mvkh CODE_SIZE-4,B6
    cmplt A1,B6,B0
    nop
    stb B5,*A4++        ;almacenamos en la Flash
    [B0] b  _boot_loop1
....
    .end
```

En lenguaje C podemos emplear la siguiente función:

```
void *memcpy(void *dst, const void *src, size_t len)
```

Donde *dst* y *src* serian las direcciones destino de copia origen de copia del tamaño de código *size\_t*, que es un tipo de dato que nos permite como máximo una copia de memoria de  $2^{16}$  bits, que será más que suficiente para nuestros intereses.

#### 4.2.6.- Módulo SPI

Este módulo nos permitirá comunicar el DSP con los transceptores inalámbricos Bluetooth y Zigbee, y con el codec de audio a una frecuencia de 6 MHz. Como hemos mencionado varias veces, para lo que respecta a los transceptores, el DSP es transparente a cuestiones como el entramado y sincronización propias de los protocolos inalámbricos y se descarga esta tarea en los microprocesadores de los transceptores. Para lo que nos concierne, la aplicación de comunicación con los transceptores va a tener una rutina que envíe cada 1 segundo una petición de datos al transceptor Zigbee en forma de un byte especial predefinido de petición. Como respuesta procedente de la IAU, el PSE tiene otra rutina capaz de recibir 10 muestras obtenidas durante el último segundo. Cada muestra estará formada por 3 bytes (uno por cada eje de aceleración), donde cada byte está codificado en CA2 y en punto fijo. Para el enlace con la URA mediante el protocolo Bluetooth también se establece una frecuencia de comunicación de 1 Hz.

Para poder implementar este protocolo interdispositivo se ha empleado el periférico McBSP [127] del Kit de Desarrollo DSK6713. Recordamos que un sistema compatible con el protocolo SPI debe tener una configuración maestro-esclavo. El protocolo SPI es una interfaz de 4 cables compuesta de datos serie de entrada (“master in slave out” o MISO), datos de salida serie (“master out slave in” o MOSI), reloj (SCK) y una señal de

activación a nivel bajo (SS). La comunicación entre el maestro y el esclavo está determinada por la presencia o ausencia del reloj de maestro. La transferencia de datos es iniciada mediante la detección del reloj de maestro y es terminada por la ausencia de éste. El esclavo debe ser activado durante este periodo de transferencia. Cuando el McBSP es el maestro, la activación del esclavo se deriva del pulso de sincronización de trama transmitida FSX. Podemos observarlo en la Figura 4.31.

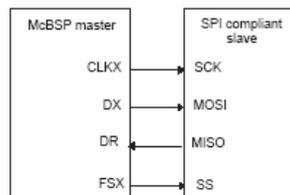
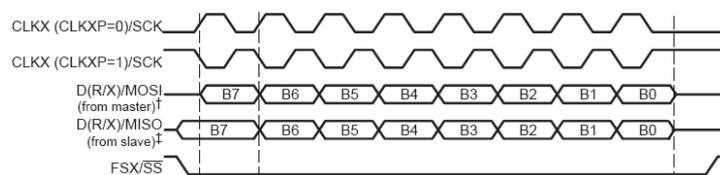


Figura 4.31.- Configuración SPI del McBSP como maestro

El modo de parada de reloj (CLKSTP) del McBSP proporciona la compatibilidad con el protocolo SPI. El McBSP soporta dos formatos de transferencia SPI que están especificados por medio del campo CLKSTP del registro SPCR (Registro de control de los puertos serie). Este campo junto con el bit CLKXP del registro PCR (Registro de control de pines), que determina el flanco activo del reloj CLKX, permite que los relojes serie se paren entre las distintas transferencias de datos usando alguna de las cuatro variaciones de tiempo posibles, como muestra la Tabla 1. Las Figuras 4.32 y 4.33 muestran los diagramas de tiempo de los dos formatos de transferencia SPI y las cuatro variaciones de tiempo.

CLKSTP	CLKXP	Esquema de reloj
0X	X	Modo de parada de reloj deshabilitado. Reloj habilitado para modo de funcionamiento no-SPI.
10	0	Estado inactivo bajo sin retraso. El McBSP transmite datos en el flanco de subida de CLKX y recibe datos en el flanco de bajada de CLKX.
11	0	Estado inactivo bajo con retraso. El McBSP transmite datos ½ ciclo más tarde del flanco de subida de CLKX y los recibe en el flanco de subida de CLKX.
10	1	Estado inactivo alto con retraso. El McBSP transmite datos en el flanco de bajada de CLKX y recibe datos en el flanco de subida de CLKX.
11	1	Estado inactivo alto con retraso. El McBSP transmite datos ½ ciclo más tarde del flanco de bajada de CLKX y los recibe en el flanco de bajada de CLKX.

Tabla 4.6.- Modos de Parada de Reloj SPI



† If the McBSP is the SPI master (CLKXM = 1), MOSI=DX. If the McBSP is the SPI slave (CLKXM = 0), MOSI = DR.  
‡ If the McBSP is the SPI master (CLKXM = 1), MISO=DR. If the McBSP is the SPI slave (CLKXM = 0), MISO = DX.

Figura 4.32.- Transferencia SPI con CLKSTP = 10b

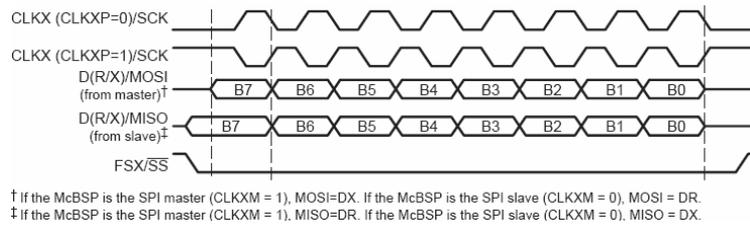


Figura 4.33.- Transferencia SPI con CLKSTP = 11b

### Operación del McBSP como Maestro SPI

Cuando el McBSP es el maestro SPI, que es el caso que nos interesa, genera el reloj de maestro CLKX y el habilitador de esclavo FSX. Por tanto, CLKX debe configurarse como una salida (CLKXM = 1) y FSX debería configurarse como una salida que pudiera conectarse a la entrada de habilitación de esclavo (SS) del dispositivo esclavo (FSXM =1). La transferencia DXR-hacia-XSR de cada elemento genera la habilitación de esclavo FSX (FSGM=0 en SRGR (*Simple Rate Generator Register*)). Por tanto para recibir un elemento en el modo maestro, el McBSP debe también transmitir simultáneamente un elemento (escrito en DXR) para generar la necesaria habilitación de esclavo FSX. Como consecuencia FSX necesita ser desactivado para habilitar al esclavo antes de que el McBSP empiece a sacar datos en el pin DX (mirar Figuras 2 y 3). Además, XDATDLY y RDATDLY deben ponerse a '1' para evitar funcionamientos erróneos del dispositivo.

Como maestro SPI, el McBSP genera CLKX y FSX a través de un generador interno de tasas de muestras (SRG). El bit CLKSM en el SRGR debe activarse para especificar al reloj CPU o a la entrada de reloj externo (CLKS) como fuente de reloj para el generador. EL CLKGDV (clock divide ratio) en SRGR debe programarse para generar CLKX a la tasa de datos SPI requerida. El McBSP genera un reloj continuo (CLKX) internamente y detiene el reloj que va hacia la interfaz externa cuando las transferencias han acabado. El reloj de recepción del McBSP es proporcionado a través del reloj interno que está corriendo continuamente. La selección del los modos de parada del reloj anulan campos de bits del *frame generator* (FPER y FWID) del SRGR.

### Configuración de la aplicación del McBSP para el estándar SPI

Para la operación de McBSP como maestro en el modo SPI, se deben seguir los siguientes pasos para una correcta inicialización, en los que iremos incluyendo parte del código implementado:

1. Configuración del registro CPLD del DSK para habilitar los puertos McBSP como E/S genéricas a través del DaughterCard del DSK.

```
DSK6713_rset(DSK6713_MISC, MCBSP2SEL);
```

2. Poner XRST=RRST=0 en el SPCR.

```
MCBSP_FMKS(SPCR, XRST, NO)
MCBSP_FMKS(SPCR, RRST, NO)
```

3. Programar los registros de configuración necesarios (y no los registros de datos) de que dispone el McBSP (ver tabla 3 para un ejemplo de un transmisión SPI a 6MHz) como es necesario cuando el puerto serie está en el estado de reset (XRST=RRST=0). Escribir los valores deseados en el campo CLKSTP del SPCR. La Tabla anterior mostraba los distintos tipos de modos CLKSTP.

```
MCBSP_XCR_RMK(...),
MCBSP_RCR_RMK(...),
MCBSP_SRGR_RMK(...),
MCBSP_PCR_RMK(...),
MCBSP_SPCR_RMK(...),
```

4. Poner GRST=1 en SPCR para poner al generador SRG fuera del reset.

```
MCBSP_enableSrg(hMcbSP1);
```

5. Esperar dos ciclos de reloj para que el McBSP se reinicialice.
6. Poner XRST=RRST=1 para habilitar el puerto serie. Hay que resaltar que el valor escrito en el SPCR en este momento debería tener sólo los bits de reset cambiados a '1' y los campos de bits restantes deberían tener los mismos valores del paso 2 y 4 anteriores

```
MCBSP_SPCR_RMK(
    ....
    MCBSP_SPCR_XRST_DEFAULT,
    ....
    MCBSP_SPCR_RRST_DEFAULT
),
```

7. Esperar dos ciclos de bit para que el transmisor y el receptor se activen
8. Escribir/leer los datos en los registros de datos correspondientes.

```
MCBSP_write(hMcbSP1,0xA0000000);
data=MCBSP_read(hMcbSP1);
```

Registro	Valor en Hex.	Descripción
RCR	0x000100A0	Fase simple, un elemento de 32 bits por trama, un ciclo de bit de retraso
XCR	0x000100A0	Fase simple, un elemento de 32 bits por trama, un ciclo de bit de retraso
SRGR	0x20000012	<ul style="list-style-type: none"> <li>• CLKX generado por el reloj interno (CLKSM = 1).</li> <li>• FSX generado por la transferencia DXR-hacia-XSR (FSGM=0)</li> <li>• CLKGDV = 0x12 para 6MHz.</li> </ul>
PCR	0x0000A0C	<ul style="list-style-type: none"> <li>• FSX es una salida (FSXM = 1) activa a nivel bajo (FSXP = 1).</li> <li>• FSR es una entrada (FSRM = 0) activa a nivel bajo (FSRP = 1).</li> </ul>

		<ul style="list-style-type: none"> <li>• CLKX es una salida (CLKXM = 1) y comienza con un flanco de subida (CLXP =0).</li> </ul>
<b>SPCR</b>	0x00001800	CLKSTP = 11b. Como CLKXP = 0, se refiere a los datos transmitidos en el flanco de subida y recibidos en el flanco de bajada de CLKX por el maestro.

Tabla 4.7.-: Valores de los Registros para una comunicación SPI a 6MHz

A continuación vemos una captura de osciloscopio de una señal SPI de 32 bits consistente en '1' y '0' alternándose.

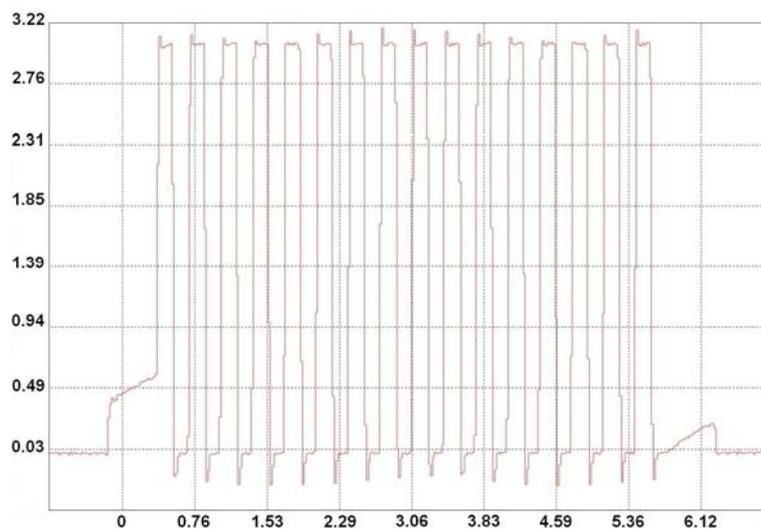


Figura 4.34.- Capturas SPI a 6MHz (ordenadas en voltios, abscisas en us y ms respectivamente)

#### 4.2.7.- Módulo I2C

Este módulo se emplea para comunicar el DSP con los dispositivos LCD y Gas Sauge a 400 Kbps. La frecuencia de ejecución de este módulo para el LCD vendrá determinada por el usuario, cuando éste quiera hacer uso de la interfaz de usuario del PSE. En cambio, se ha establecido que la monitorización del estado de la batería asociada al Gas Sauge se haga 10 veces al día, aunque este puede configurarse a gusto del consumidor.

El DSP tiene un módulo I2C [128], que permite transmitir y recibir hasta 8 bits a dispositivos compatibles I2C. Se trata de una interfaz serie síncrona a dos cables, línea de datos y línea de reloj, que funciona con salidas a drenador abierto que forman puertas AND cableadas. Comentamos brevemente otras prestaciones del módulo:

- Compatible con las especificaciones del bus I2C de Philips Semiconductors.

- Un evento de lectura EDMA y un evento de escritura EDMA, que pueden usarse por el controlador EDMA (Enhanced Direct Memory Access). Como alternativa al EDMA tenemos:
- Interrupción que puede usarse por la CPU y que se genera como resultado de una de estas condiciones: transmisión de datos preparada, recepción de datos preparada, acceso a registro preparado, no-asentimiento recibido, pérdida de arbitraje en el bus.
- Capacidad de habilitar y deshabilitar el módulo.
- Formato de datos libre.

El módulo I2C consta de lo siguientes bloques principales:

- Interfaz serie: un pin de datos (SDA) y un pin de reloj (SCL).
- Registros de datos para “mantener” el dato recibido y el “transmitido” entre el pin SDA y la CPU o el controlador EDMA.
- Registros de estado y control.
- Interfaz de bus de datos periférico para habilitar la CPU y el controlador EDMA para acceder a los registros del módulo I2C.
- Un reloj de sincronización para sincronizar el reloj de entrada I2C (procedente del generador de reloj del DSP) y el reloj en el pin SCL, y para sincronizar las transferencias de datos entre maestros de diferentes frecuencias de reloj.
- Un preescalador para disminuir el reloj de entrada controlado por el módulo I2C.
- Un filtro de ruidos en cada uno de los dos pines, SDA y SCL.
- Un árbitro para gestionar el arbitraje entre el módulo I2C (cuando es maestro) y otro maestro.
- Lógica de generación de interrupción, para que la interrupción pueda enviarse a la CPU.
- Lógica de generación de evento EDMA, para que la actividad en el controlador EDMA pueda sincronizarse con la recepción de datos y su transmisión en el módulo I2C.

La siguiente figura muestra los cuatro registros principales para la transmisión y la recepción. La CPU o el controlador EDMA escribe los datos para la transmisión en el I2CDXR y lee los datos recibidos del I2CDRR. Cuando el módulo I2C se configura como transmisor, los datos escritos en I2CDXR son copiados en I2CXSR y sacados en

el pin SDA de uno en uno. Cuando el módulo I2C se configura como receptor, los datos recibidos se sacan del I2CRSR y entonces copiados en el I2CDRR, de donde se leen.

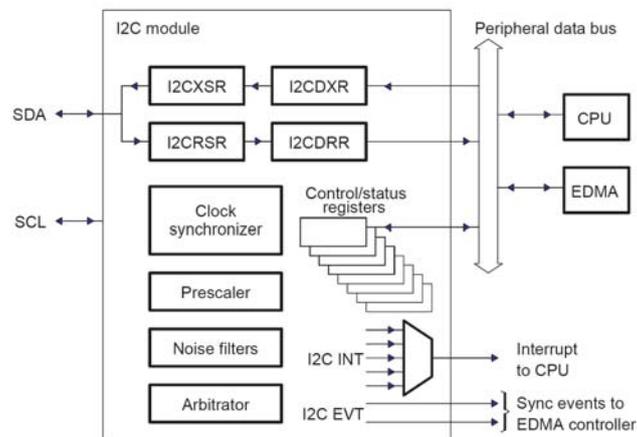


Figura 4.35.- Diagrama de bloques del módulo I2C

Pasamos a continuación a comentar el código de este módulo.

## Código

Pone todos los registros a sus valores por defecto:

```
I2C_RSET(I2COAR0,I2C_I2COAR_DEFAULT); // Own Address Register
I2C_RSET(I2CIMR0,I2C_I2CIMR_DEFAULT); // Interrupt Mask Register
I2C_RSET(I2CCLKL0,I2C_I2CCLKL_DEFAULT); // Clock Divider Registers
I2C_RSET(I2CCLKH0,I2C_I2CCLKH_DEFAULT); // Clock Divider Registers
I2C_RSET(I2CCNT0,I2C_I2CCNT_DEFAULT); // Data Count Register
I2C_RSET(I2CSAR0,I2C_I2CSAR_DEFAULT); // Slave Address Register
I2C_RSET(I2CMDR0,I2C_I2CMDR_DEFAULT); // Mode Register
I2C_RSET(I2CPSC0,I2C_I2CPSC_DEFAULT); // Prescaler Register
```

## Iniciamos el módulo I2C

```
// Desactiva todas las interrupciones
I2C_RSET(I2CIMR0,I2C_I2CIMR_RMK(I2C_I2CIMR_ICXRDY_MSK,
....
I2C_I2CIMR_AL_MSK));

//Establece frecuencia SCL
I2C_RSET(I2CCLKL0,I2C_I2CCLKL_RMK(I2C_I2CCLKL_ICCL_OF(100-6)));
I2C_RSET(I2CCLKH0,I2C_I2CCLKH_RMK(I2C_I2CCLKH_ICCH_OF(100-6)));
I2CMDR:

// El modulo es el maestro
I2C_RSET(I2CMDR0,I2C_I2CMDR_RMK( ..., I2C_I2CMDR_MST_MASTER,...))

// Modo de direccionamiento de 7 bits.
I2C_RSET(I2CMDR0,I2C_I2CMDR_RMK( ..., I2C_I2CMDR_XA_7BIT, ...))

// 8 bits por data word
I2C_RSET(I2CMDR0,I2C_I2CMDR_RMK(...,I2C_I2CMDR_BC_BIT8FDF,...))

// Configuración reloj SCL (debe estar en reset)
I2C_RSET(I2CPSC0,I2C_I2CPSC_RMK(I2C_I2CPSC_IPSC_OF(20-1)));

// Configuramos dirección esclavo a "address"
I2C_RSET(I2CSAR0,I2C_I2CSAR_RMK(I2C_I2CSAR_A_OF(address)));
```

### Configuramos módulo como transmisor o receptor

```
...
// Espera a que el bus deje de estar ocupado
while((I2C_RGET(I2CSTR0)&_I2C_I2CSTR_BB_MASK)!=0);
...
```

### Procedemos al envío de datos

```
...
while(len>0) // mientras que haya datos que transmitir
{
// Si I2CDXR está preparado (los datos han sido copiados desde I2CDXR a I2CXSR)
if ((I2C_RGET(I2CSTR0)&_I2C_I2CSTR_ICXRDY_MASK)!=0)
{
// Escribo el dato en I2CDXR (como mucho de 8 bits)
I2C_RSET(I2CDXR0,I2C_I2CDXR_RMK(I2C_I2CDXR_OF(*source++)));
len--;
}
}

// Espera a que el bus deje de estar ocupado (el bus esta ocupado cuando el
// modulo I2C ha recibido o transmitido un bit de START en el bus)
while((I2C_RGET(I2CSTR0)&_I2C_I2CSTR_BB_MASK)!=0);
...
```

### Y a su recepción, si procede:

```
...
while(len>0)
{
// Si I2CDRR está listo (Los datos han sido copiados desde I2CRSR a I2CDRR.
if((I2C_RGET(I2CSTR0)&_I2C_I2CSTR_ICRRDY_MASK)!=0)
{
// Guarda el dato recibido en source
*source++=I2C_RGET(I2CDRR0);
len--;
}
}

// Espera a que el bus deje de estar ocupado (el bus esta ocupado cuando el
// modulo I2C ha recibido o transmitido un bit de START en el bus)
while((I2C_RGET(I2CSTR0)&_I2C_I2CSTR_BB_MASK)!=0); //
...
```

A continuación vemos una captura de osciloscopio de una señal I2C de 8 bits a 6.25 Kbps en la que el maestro (DSP) queda a la espera de confirmación (ACK) del esclavo.

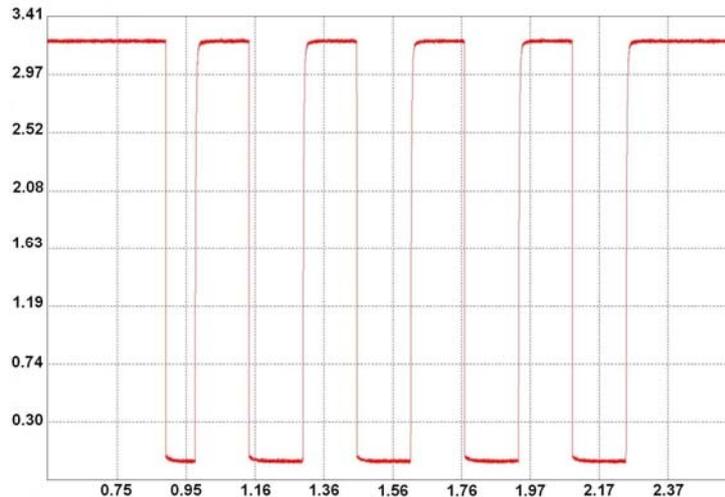


Figura 4.36.- Capturas I2C a 6.25KHz (ordenadas en voltios, abscisas en us y ms respectivamente)

#### 4.2.8.- Módulo SelfTest

Este módulo emplea la librería BSL del Development Kit para realizar test básicos en el DSP y sus periféricos embebidos. Está programado en la memoria Flash de forma que se ejecuta tanto en el arranque (boot) del DSP como a una frecuencia determinada por el usuario, que en un principio se ha establecido en una vez cada 24 horas.

Las pruebas más importantes realizadas son para los siguientes periféricos, en los que iremos incluyendo parte del código:

##### 1. Memoria interna.

Vamos a escribir con un patrón de datos específico en la memoria interna del DSP y a continuación vamos a comprobar que lo que se ha escrito efectivamente es lo correcto. Tanto a nivel de palabra de 32 bits:

```
// Chequeamos la memoria interna desde byte 0x10000 a byte 0x40000
...
// Rellenamos el rango de la memoria con un valor
for (i = start; i < end; i+=4)
{
    *((Uint32 *)i) = val;
}
...
// Verificamos la escritura
for (i = start; i < end; i+=4)
{
    if (*((Uint32 *)i) != val)
        return 1;
}
return 0;
...
```

Comparamos bit a bit dentro de cada palabra:

```
...
// Leemos 1's y 0's
```

```

mask = 1;
for (i = 0; i < 32; i++)
{
    // Probamos '1' en la posicion i
    *pdata = mask;
    if (*pdata != mask)
        return 1;

    // Probamos '0' en la posicion i
    *pdata = ~mask;
    if (*pdata != (~mask))
        return 1;

    mask = mask << 1;
}
return 0;
...

```

## 2. Memoria Flash externa.

```

// Reseteamos la Flash
*((volatile UInt8 *)DSK6713_FLASH_BASE) = 0xf0;

// Leemos los ID del Fabricante de la Flash
MfgId = *((UInt8 *)DSK6713_FLASH_BASE) & 0xff;
DevId = *((UInt8 *)DSK6713_FLASH_BASE + 1) & 0xff;

// Resetemos la Flash
*((UInt16 *)DSK6713_FLASH_BASE) = 0xf0;

...
// Comprobamos los ID
if ((MfgId != 0x01) || (DevId != 0xb9))
    return 1;
...
return 0;

```

## 3. McBSP en modo lazo cerrado

```

// Configuramos McBSP en modo lazo cerrado
MCBSP_config(hMcbbsp, &mcbbspCfg_loopback);

...
// Bucle de transferencia de datos
for (i = 0; i < (N + delayed); i++) {
    ....
    // Escribimos datos de 16 bits en DXR
    MCBSP_write(hMcbbsp,src[i]);
    ....
    // Leemos los valores de 16 bits de DRR Read 16 bit value from DRR */
    receivedata = MCBSP_read(hMcbbsp);
    ....
    // Comprobamos los datos para asegurarnos que la transferencia fue correcta
    for (i = 0; i < N; i++)
        if (dst[i] != src[i])
            return 1;
}
...
return 0;

```

## 4. Tranferencia DMA interna: muy parecido al test McBSP, esto es:

```

// Establecemos varlores origen y destino de la transferncia Set src values and clear destination */
for (i = 0; i < N; i++) {
    src[i] = i;
    dst[i] = 0;
}

// Configuramos EDMA
hEdmaCha8 = EDMA_open(8, EDMA_OPEN_RESET);
EDMA_config(hEdmaCha8, &edmaCfg8);

```

```
// Iniciamos la transferencia EDMA Start CPU initiated EDMA transfer by setting bit 24 in ESR */
EDMA_RSET(ESR, 0x00000100);

// Sondemoas la interrupcion DMA para verificar que se ha realizado la transferencia
while (!EDMA_intTest(15))
{
    ;
}

...
// comprobamos los datos
for (i = 0; i < N; i++)
    if (dst[i] != src[i])
        return 1;
...
return 0;
```

## 5. Codec

```
// Generamos onda sinusoidal de 1KHz y de 1 segundo de duracion
for (i = 0; i < 1000; i++)
{
    for (j = 0; j < SINE_TABLE_SIZE; j++)
    {
        ...
        // Enviamos la onda al codec para poder escucharla
        while (!DSK6713_AIC23_write(hCodec, sinetable[j]));
        ...
    }
}

...
return 0;
```

Este módulo muestra el índice del test que se ejecuta en los LEDs del DSK6713. Si se detecta algún error mientras se ejecuta el test, parpadeará el LED correspondiente al índice de dicho test. Si por el contrario se completa al test satisfactoriamente, los cuatro LEDs parpadearán tres veces y se quedarán encendidos.

### 4.2.9.- Módulo Audio

En este caso se trata de que el PSE reproduzca mensajes sonoros almacenados en la Flash de variado contenido, ya sean de advertencia, para tranquilidad del paciente, información del estado del dispositivo, etc. Se va a disponer de una tabla de mensajes a configurar por el médico y modificable mediante comando, de forma que estos no se encuentran limitados en cuanto a lo que se diga.

Este módulo debe hacer uso intensivo del periférico McASP, así como el SPI para la configuración del codec audio.

En el código anexo se hace uso de un botón del interfaz DIP del DSK6713 (concretamente el número 3) para proceder a la grabación, que se hará a través de un micrófono conectado a la entrada MIC-IN de la placa. A continuación se reproduce

dicho mensaje sonoro encendiéndose para ello el LED 0. Previamente a todo ello se habrá reservado una zona de la Flash para su almacenamiento.

### Código

```
while(1)
{
//Si apretamos el boton 3
if(DSK6713_DIP_get(3) == 0)
{
....
//Encendemos el LED correspondiente
DSK6713_LED_on(3);
...
for (i = 0; i<N; i++)
// grabamos datos
buffer[i] = input_sample();

//Apagamos LED 3 al terminar
DSK6713_LED_off(3);
break;
}
};
....
while(1)
{
if((DSK6713_DIP_get(0)==0)&&(var==0)) //if SW#0 pressed and var=0
{
....
//encendemos LED 0
DSK6713_LED_on(0);
for (i = 0; i<N; i++)
....
//reproducimos datos guardados
output_sample(buffer[i]*10);

//Apagamos LED 0 al terminar
DSK6713_LED_off(0);
}
....
};
...
return(0);
```

## 4.2.10.- Otros Aspectos del Diseño Software

Nuevamente señalar que por motivos de extensión de la memoria se ha centrado en los módulos software que el autor considera más relevantes, y por ello no se han comentado todo los módulos implementados o en proceso de implementación. Ejemplos de estos últimos tendríamos en el módulo timer para el watchdog en el primer caso, o en la emulación del protocolo IrDA1.4 en pines GPIO para el módulo infrarrojo, en el segundo caso.

Pasamos ahora a comentar otros detalles relevantes.

### 4.2.10.1.- Comunicación del DSK con Matlab

Las librerías de la herramienta Real-Time Data Exchange (RTDX™) permiten crear un canal virtual para transferir datos entre el Host y el dispositivo DSK (target) a través de

un puerto JTAG sin interferir en la aplicación que se está ejecutando en el target. RXTDX consiste tanto en componentes de host como de target. Una pequeña librería software correrá en la aplicación target, que realiza llamadas a funciones de esta librería API para enviarle datos u obtenerlos. Esta librería hace uso de un emulador para mover datos hacia o desde el Host por medio de una interfaz JTAG. La transferencia de datos en el host ocurre en tiempo real mientras la aplicación target esté en ejecución.

La gran utilidad de RTDX la vamos a tener en el procesamiento de las señales que se realizan en el PSE procedentes de los sensores biomédicos inteligentes, puesto que gracias a ella vamos a ser capaces de realizar un procesamiento en paralelo en Matlab de los datos que toma el PSE. En concreto se ha realizado una aplicación que ha permitido la recepción de información en forma de vector de valores del DSK procedente del Host. Una vez que el DSP ha recogido el vector de datos, los procesa internamente (filtrado, escalado, ...) y los devuelve al Host, donde se estará ejecutando paralelamente la aplicación .m en Matlab correspondiente.

#### **4.2.10.2.- Estado actual del dispositivo. Pasos a seguir en el diseño software.**

Evidentemente, y como se ha mostrado durante todo el trabajo, el diseño software está profundamente ligado al diseño hardware del servidor personal. Es por este motivo que las aplicaciones realizadas deben pasar por una necesaria fase de depurado en el prototipo preindustrial para llevar al diseño definitivo. En ella han de tenerse en cuenta la adaptación de código del DSP 6713 al DSP 6727, que en todo caso no deber ser demasiado problemática puesto que son procesadores de la familia 6000 de gran compatibilidad, y de hecho a nivel de ensamblador ambos dispositivos solo difieren en un par de instrucciones de operaciones de multiplicación añadidas de mayor precisión.

Otro paso a seguir, que escapa al objetivo del presente trabajo, es el del desarrollo de los algoritmos de procesamiento de señales que irán ejecutándose en el PSE en tiempo real. Las aplicaciones RTDX comentadas en el apartado anterior, así como el del juego de bloques Simulink del paquete del paquete *Embedded Targets for TI C6000* de Matlab, que citamos en el apartado 5.2.1, deben servirnos de herramienta de apoyo para la implementación de procesamiento de señales asignada al PSE.

### **4.3.- Modos de Funcionamiento y Consumo**

#### **4.3.1.- Modos de funcionamiento**

En nuestro DSP 6727 se van a distinguir cuatro modos de operación, en función de la carga de trabajo de la CPU y los periféricos embebidos que se encuentran activos:

- Operación normal: todos los periféricos del DSP, utilizados por el PSE, están funcionando. La carga de la CPU se estima en un 60%.
- Operación básica: todos los periféricos están dormidos salvo la memoria Flash donde se almacenan las rutinas a ejecutar. Además, del periférico EMIF, también se encontrará activo el periférico SPI que sirve para comunicar con los transceptores de comunicación y el I2C que se comunica con el dispositivo que monitoriza el estado de la batería. La carga de la CPU es de un 60%.
- Configuración de usuario: si el PSE está siendo configurado por el usuario a través del puerto de infrarrojos estarán activos tanto el periférico GPIO a través del cual se comunica con el DSP, así como el módulo LCD que se comunica con el DSP a través del periférico I2C0. También estará activo el periférico I2C1 para monitorizar la batería. La memoria Flash también estará activa porque puede que haya que guardar algún parámetro de configuración pero su utilización se verá sensiblemente reducida. Se supone que no se estará llevando a cabo ninguna comunicación con los transceptores ni ejecutando en el DSP algoritmos de procesado de señal. La carga de la CPU se reduce al 35% porque se limita a la ejecución de rutinas de control, no de procesado de señal.
- Ahorro energético: el DSP cesa toda comunicación con los periféricos y en la CPU sólo se ejecutan rutinas de control con lo que su carga es del 25%.

El tranceptor Zigbee recibe 10 muestras de aceleraciones por segundo desde la IAU e intercambia con ella eventos–alarmas y parámetros de calibrado–sincronización. Presenta cuatro modo de operación: está dormido por defecto (mín. consumo) y cada segundo se despierta y pasa al modo activo. Se pueden distinguir a su vez dos estados según se encuentre activa la parte de radiofrecuencia (máx. consumo) o no.

El transceptor BT permite cada segundo el intercambio de información entre el PSE y la URA. Dispone de distintos modos de operación en función del consumo y, de manera análoga a Zigbee, está dormido por defecto (mín. consumo) y pasa al modo activo cuando se establece una comunicación con la URA. En este modo se vuelven a

distinguir dos estados dependiendo de si la parte de radiofrecuencia se encuentra activa (máx. consumo) o no.

El transceptor de infrarrojos está por defecto en modo de ahorro energético (mín. consumo) y pasa a modo activo durante el periodo de tiempo en el que el PSE es configurado desde un PC.

La memoria Flash opera de manera normal cuando el 75% del tiempo se utiliza como memoria de programa (acceso en lectura) y el resto del tiempo como memoria de datos. Sin embargo, cuando el PSE está siendo configurado desde un PC, la memoria funciona sólo como memoria de datos. El consumo varía con el tipo de operación que se realiza: lectura, escritura o borrado. Durante los periodos de inactividad (líneas de datos y direcciones no varían) la memoria está en modo de bajo consumo.

El módulo LCD, por defecto, no muestra información por pantalla (mín. consumo) y se enciende cuando el usuario pulsa los botones para interactuar con él (máx. consumo) y cuando se configura el PSE desde un PC (consumo reducido).

El módulo de audio está por defecto en modo de bajo consumo y se activa para reproducir mensajes de audio.

El módulo de control-supervisión funciona sin parar.

### 4.3.2.- Consumo

La mayor parte del consumo del PSE se debe al DSP. TI proporciona en su página web una aplicación Excel que estima el consumo del DSP a partir de diferentes parámetros como frecuencias de operación, de la CPU y de los periféricos embebidos, o porcentajes de utilización de los mismos [129,130]. Los resultados obtenidos en función del modo de operación del DSP aparecen en la siguiente tabla:

<b>Consumo</b>	<b>Núcleo</b>	<b>Módulo E/S</b>	<b>Total</b>
<b>Normal</b>	650 mA	47.57 mA	696.57 mA
<b>Básica</b>	636.66 mA	33.03 mA	669.69 mA
<b>Configuración</b>	524.16 mA	32.72 mA	556.89 mA
<b>Ahorro</b>	480 mA	5.15 mA	485.15 mA

Tabla 4.8.- Consumo del DSP

Para el resto de componentes que forman parte del PSE se han usado los datos de consumo de sus hojas de características y se ha descrito matemáticamente el consumo del PSE en función del número de horas  $n$ , que la batería puede hacer frente a esa demanda.

La elección de la batería ha de maximizar la autonomía de nuestro prototipo, porque el PSE es un dispositivo portable, y minimizar el tamaño para que sea discreto y se preserve la condición médica del paciente. Se realizó una evaluación de baterías de polímero de Litio con capacidad nominal superior a 1000mAh y pequeñas dimensiones [98], y el resultado de dicho estudio ha sido la batería AE385067P de Vizhuel, cuyas características eléctricas y mecánicas podemos ver a continuación.

Modelo	Fabricante	Capacidad Nominal (mAh)	Vol (mm <sup>3</sup> )	n(h)
AE385067P	Vizhuel	1300	13477.77	4

Tabla 4.9.- Batería seleccionada

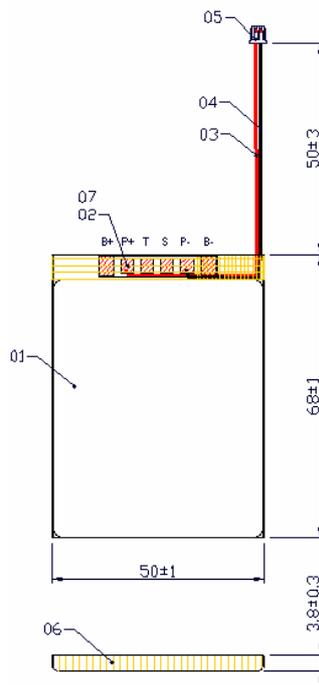


Figura 4.37.- Dimensiones mecánicas de la batería Vizhuel

### 4.3.3.- Conclusiones

Las dimensiones de las baterías consideradas han hecho que minimizar el tamaño tenga prioridad frente a maximizar la autonomía. La autonomía de la batería seleccionada es

de 4 horas para el modo de operación descrito en el apartado anterior. A partir de este resultado el objetivo es extender la operación del PSE a las 24 horas diaria.

Con este fin podemos realizar técnicas de reducción del consumo consistentes en la óptima conmutación entre los modos de operación citados, y sobre todo desde un punto de vista software la correcta gestión y ejecución del módulo de ahorro energético e incluso la disminución de la frecuencia de de la CPU si procede.

A lo anterior se superponen técnicas de otra índole. Por ejemplo no hay por qué esperar a una descarga total de la batería para comenzar el proceso de carga, pudiendo cargar la batería cuando el individuo se encuentre comiendo. Para el correcto funcionamiento del monitor de movimiento el PSE no ha de ser portado por el sujeto en todo momento, basta con que ambos se encuentren en la misma habitación. Además, la posibilidad de utilizar una batería supletoria que cubra a la batería principal durante su carga, duplica el espacio temporal de funcionamiento del PSE.

## 5.- Conclusiones

En este estudio se han presentado las características funcionales y de diseño de la arquitectura hardware y software de un servidor personal para la monitorización de sensores inteligentes, que aporta como principal novedad su capacidad de procesamiento y generación de conocimiento personalizado en tiempo real por medio de una arquitectura distribuida, y que se engloba dentro del problema de control de las redes de sensores inalámbricas.

Se ha presentado un diseño tecnológico viable del PSE de un monitor de movimiento humano que cumple los requisitos funcionales, de autonomía, portabilidad, tamaño, discreción, coste y sencillez que se le exigían.

El diseño y resultados obtenidos hasta el momento sugieren que el PSE puede cumplir las especificaciones funcionales definidas, y por tanto su viabilidad dentro de la capa de sensores inteligentes citada.

### Futuros avances

Como futuros avances en la investigación destacamos los siguientes:

- El cierre y validación del ejecutivo de control sobre el prototipo preindustrial del PSE que actualmente está en fase de montaje de los componentes.
- El cierre y validación de los algoritmos de procesamiento de señal que se ejecutarán en el PSE.
- La incorporación de nuevas tecnologías de comunicación inalámbricas que nos permitan la incorporación efectiva de una gran variedad de sensores, como es el caso de Wibree/BT [0].
- La extensión de la autonomía de la batería al día completo mediante las técnicas citadas en el apartado 4.3.3, que incluso pueden facilitarnos la elección de una batería de menor tamaño.
- Una vez conseguidos los anteriores, realizar un diseño definitivo del PSE a modo de reloj o pulsera, que se conseguirá al eliminar conectores, jumpers y sockets, y embebiendo los transceptores junto con sus respectivas antenas.

## **6.- Publicaciones relacionadas con el trabajo**

### **6.1.- Publicaciones en las que participa el autor**

1. M.A. Estudillo Valderrama, M. Prado, L. Fernández del Rosal, “Diseño software de un servidor personal (PSE) para el control y procesado de señales provenientes de sensores inteligentes”, Actas del XXV Congreso Anual de la Sociedad Española de Ingeniería Biomédica, 2007, pp. 60-63.
2. L. Fernández del Rosal, M. Prado Velasco, M. A. Estudillo Valderrama, “Diseño hardware y autonomía del servidor personal inalámbrico (PSE) de un monitor de movimiento humano”. Actas de XXV Congreso Anual de la Sociedad Española de Ingeniería Biomédica CASEIB 2007.

Destacar que en estos momentos se está estudiando la posibilidad de enviar el menos dos publicaciones a sendas revistas internacionales de alto índice de impacto.

### **6.2.- Otras publicaciones del Grupo**

Patentes de la estructura del monitor en desarrollo → cosas que no se han podido publicar.

1. Prado, M.; Roa, L.M.; Reina, J.L.; Palma, A.; Milán, J.A., "Virtual Center for Renal Support: Technological Approach to Patient Physiological Image", IEEE Transactions on Biomedical Engineering, 49, 12, 2002, p. 1420-1430.
2. Sánchez Fernández, G.; Reina Tosina, J.L.; Roa, L.M.; Prado, M.; Díaz Caballero, A., "Design and evaluation of communications protocols for a home health telecare system for elderly people", Xth Mediterranean Conference on Medical and Biological Engineering and 2nd Health Telematic conference, 2004, p. 1128-1131.
3. Prado, M; Roa, L.M; Reina Tosina, J.L., “A Knowledge-Based Telehealthcare System Aimed At A Patient's Personalized Assistance”, Xth Mediterranean Conference on Medical and Biological Engineering and 2nd Health Telematic conference. MEDICON and HEALTH TELEMATICS 2004.
4. Díaz Caballero, A.; Prado, M.; Roa, L.M.; Reina Tosina, J.L.; Sánchez Fernández, G., “Accelerometry Based Device For Human Movement Monitoring”, Xth Mediterranean Conference on Medical and Biological Engineering and 2nd Health Telematic conference. MEDICON and HEALTH TELEMATICS 2004.
5. Prado, M; Roa, L.M; Reina Tosina, J.L., “Distributed Intelligent Architecture For Falling Detection And Physical Activity Analysis In The Elderly”, Proceeding Of The Second Joint Embs/Bmes Conference, 2002.
6. Prado, M; Roa, L.M; Reina Tosina, J.L, Palma Alvarez A., Milan Martin J. Antonio, “Virtual Center For Renal Support: Definicion Of A Novel Knowledge-Based Telemedicine System”, Proceedings 23rd Annual International Conference Of The Ieee-Embs, 2001.
7. Díaz Caballero, A.; Prado, M.; Roa, L.M.; Reina Tosina, J.L.; Sánchez Fernández, G., “Preliminary Evaluation Of A Full-Time Falling Monitor For The Elderly, Proceedings Of The 26th International Conference Ieee Engineering In Medicine And Biology Society, 2004.

8. Sánchez, J.G.; Reina-Tosina, L.J.; Roa, L.M.; Prado, M., “Infraestructura de comunicaciones de bajo coste para un sistema de teleasistencia domiciliaria de personas mayores”, Actas del XXI Congreso Anual de la Sociedad Española de Ingeniería Biomédica, 2003.
9. Sánchez, J.G.; Reina, L.J.; Roa, L.M.; Prado, M.; Díaz, A.J., “Diseño y evaluación de protocolos de comunicaciones para un sistema de teleasistencia domiciliaria para personas mayores”, Actas del XXI Congreso Anual de la Sociedad Española de Ingeniería Biomédica, 2003.
10. Díaz, A.; Prado, M.; Roa, L.M.; Reina, J.; Sánchez, G., “Evaluación de una unidad inteligente de acelerometría para la monitorización del movimiento”, Actas del XXI Congreso Anual de la Sociedad Española de Ingeniería Biomédica, 2003.
11. Prado, M.; Roa, L.M.; Reina Tosina, J.L., “Evaluación De La Función De Detección De Caídas De Un Monitor Portable De Movimiento Humano, Actas del XXII Congreso Anual de la Sociedad Española de Ingeniería Biomédica, 2004.
12. Prado, M.; Roa, L.M.; Reina, L.J.; Rovayo, M., “Wearable human movement monitoring device aimed at providing personalized support in healthcare”, Microtechnologies For The New Millenium 2005.
13. Prado, M.; Roa, L.M.; Reina, L.J, “Viability study of a personalized and adaptive knowledge-generation telehealthcare system for nephrology (NEFROTEL)”, International Journal of Medical Informatics, 2006.
14. Reina, L.J; Roa, L.M.; Prado, M., “Design of antenas for a wearable sensor for homecare movment monitoring”, Proceedings of the 28th IEEE EMBS Annual International Conference, 2006.
15. Prado, M.; Roa, L.M, “Wearable Real-time and Adaptive Feedback Device to Face the Stuttering: a Knowledge-based Telehealthcare Proposal”, Medicon 2007-11 Mediterranean Conference On Medical And Biological Engineering And Computing, 2007.
16. Fabiola Fernández Gutiérrez, Prado, M, Roa, L.M, “Sistema de supervisión en un proyecto de teleasistencia: metodología y herramientas de desarrollo”, Actas Del XXV Congreso Anual De La Sociedad Española De Ingeniería Biomédica, 2007.
17. J. G. Jiménez, L. M. Roa, M. Prado, “Diseño de una unidad inteligente de acelerometría de segunda generación para la monitorización del movimiento humano”, Actas Del XXV Congreso Anual De La Sociedad Española De Ingeniería Biomédica, 2007.

## 7.- Bibliografía

0. Prado, M.; Roa, L.M, “Wearable Real-time and Adaptive Feedback Device to Face the Stuttering: a Knowledge-based Telehealthcare Proposal”, Medicon 2007-11 Mediterranean Conference On Medical And Biological Engineering And Computing, 2007.
1. L. Roa Romero, M. Prado Velasco, J. Reina-Tosina “Sistema portable para la monitorización del movimiento, estado postural, y actividad física de humanos durante las 24 horas del día”. Patente. Solicitud: 2002-07-18 00:00:00, Nr: P200201710.
2. F. Lesemann, C. Martin, "Home-based Care, the Elderly, the Family and the Welfare State: An International Comparison", University of Ottawa Press, 1993.
3. D. Ribera Domene, A. Reig Ferrer, A. Mojos, y otros, "Condiciones sociales de salud y competencia funcional de los ancianos muy viejos. Un modelo sociosanitario de atención domiciliaria", Revista Española de Geriátria y Gerontología, Vol. 30, No. 2, pp. 85-94, 1995.
4. "La Tercera Edad en España, aspectos cuantitativos, proyecciones de la población española de 60 y más años para el periodo 1986-2010", Ministerio de Asuntos Sociales, 1991.
5. M. A. Durán, "El cuidado de la salud", Ministerio de Cultura, Instituto de la Mujer, pp. 83-104, Madrid, 1988.
6. G. R. Parkerson, J. LL. Michener, y otros, "Associations among family stress and personal functional health status", *J. Clin. Epidemiol.*, Vol. 42, pp. 217-229, 1989.
7. R. M. López Pisa, N. Pijuán Maymó, "Atención Domiciliaria", Doyma, 1991.
8. OCDE, “L'aide aux personnes âgées dépendantes. Chapitre de synthese du rapport final”, París, 1993.
9. Antonio Abellán García, M<sup>a</sup>. Dolores Puga Gonzáles, Lourdes Pérez Ortiz, Mayte Sancho Castiello; “Tomo 1: Informe 2004. Las Personas Mayores en España, Datos Estadísticos Estatales y por Comunidades Autónomas”. Ministerio de Trabajo y Asunto Sociales Secretaría de Estado de Servicios Sociales, Familias y Discapacidad Instituto de Mayores y Servicios Sociales (IMSERSO), 2004.
10. Diario El País, Sección Andalucía, 2-12-2007.
11. K. Doughty, K. Cameron, P. Garner, “Three Generation of Telecare of the Elderly”, *Journal of Telemedicine and Telecare*, Vol. 2, No. 2, pp. 71-80, 1996.
12. N. Lovell, B. Celler, F. Magravi, et al, “Web-Enabling Technologies in Home Telecare Provision”, *Proc. World Congress on Med. Physics and Biomed. Eng.*, 2000.
13. R. Kitney, S. Claesen, C. Toumazou, “Automatic Physiological Monitoring in the Home Using a Web-Based Electronic Patient Record (EPR)”, *Proc. World Congress on Med. Physics and Biomed. Eng.*, 2000.
14. S. Singh, H. Hsiao, “Internet Based Infant Monitoring System”, "*Proc. First Joint BMES/EMBS Conf.*", pp. 674, 1999.
15. Reina, L.J.; Roa, L.M.; Estudillo, M.A.; Prado, M.; Gómez Cía, T.; Amaya, M.N., “Plataforma de teleconsulta de alta resolución para cirugía menor ambulatoria”, XXIII Congreso Anual de la Sociedad Española de Ingeniería Biomédica. CASEIB 2005.

16. M. Nambu, K. Nakajima, A. Kawarada, T. Tamura, "A System to Monitor Elderly People Remotely Using the Power Line Network", Proc. World Congress on Med. Physics and Biomed. Eng., 2000.
17. B. Tran, "Wireless Physiological Monitoring: Applications and Implementation for Home Health Care", Proc. World Congress on Med. Physics and Biomed. Eng., 2000.
18. M. van Gils, I. Korhonen, T. Tuomisto, et al, "Wireless Monitoring System for Personal Weight Control", Proc. World Congress on Med. Physics and Biomed. Eng., 2000.
19. Z. Benyó, B. Benyó, P. Várady, "Patient Monitoring on Industry Standard Field Bus", Proc. First Joint BMES/EMBS Conf., pp. 705, 1999.
20. E. Zahedi, M. Mohd Ali, M. Gangeh, "Design of a Web-Based Wireless Mobile Teleconsultation System with a Remote Control Camera", Proc. World Congress on Med. Physics and Biomed. Eng., 2000.
21. Carl May, Maggie Mort, Frances Mair, and Tracy Finch. Telemedicine and the 'Future Patient'? Risk, Governance and Innovation. Technical Report Project ref: L21825 2067, Economic & Social Research Council, 2005.
22. Prado M., Roa L., Reina L.J. Viability study of a personalized and adaptative knowledge-generation telehealthcare system for nephrology (NEFROTEL). International Journal of Medical Informatics, 2006.
23. D. M. Karantonis, M. R. Narayanan, M. Mathie, N. H. Lovell, B. G. Celler. Implementation of a Real-Time Human Movement Classifier Using a Triaxial Accelerometer for Ambulatory Monitoring. IEEE Trans. Information Tech. Biomed., vol. 10, no. 1, pp. 156-157, 2006.
24. S. Warren, R. L. Craft, "Designing Smart Health Care Technology into the Home of the Future", Proc. First Joint BMES/EMBS Conf., pp. 677, 1999.
25. P. Várady, "Design Aspects of Modern Open Architecture Patient Monitoring Systems", Proc. First Joint BMES/EMBS Conf., pp. 705, 1999.
26. S. Warren, R. L. Craft, R. C. Parks, et al, "A Proposed Information Architecture for Telehealth System Interoperability", Proc. First Joint BMES/EMBS Conf., pp. 702, 1999.
27. A. Kinsella, "Current Issues in Design of Telecare Technologies", Proc. First Joint BMES/EMBS Conf., pp. 690, 1999.
28. D. Goodman, "The Wireless Internet: Promises and Challenges", Computer, Vol. 33, No. 7, pp. 36-41, 2000.
29. D. Clark, "Encryption Advances to Meet Internet Challenges", Computer, Vol. 33, No. 8, pp. 20-24, 2000.
30. K. P. Birman, "The Next-Generation Internet: Unsafe at Any Speed?", Computer, Vol. 33, No. 8, pp. 54-60, 2000.
31. CEN ENV 13606. 2000. Medical Informatics { Electronic healthcare record communication. European Prestandard ENV 13606, European Committee for Standardization, Brussels, Belgium.
32. HL7. Health Level 7. <http://www.hl7.org/>
33. openEHR. Open Electronic Health Record. <http://www.openehr.org/>
34. Marco Eichelberg, Thomas Aden, Riesmeier Jörg, Asuman Dogac, and Gokce B. Laleci. A survey and analysis of Electronic Healthcare Record standards. ACM Comput. Surv., 37(4):277-315, 2005.
35. J. M. Winters, M. Rosen, "The Rehabilitation Engineering Research Center on Telerehabilitation : Mission and Approaches", Proc. First Joint BMES/EMBS Conf., pp. 676, 1999.

36. G. Williams, K. Doughty, D. A. Bradley, "Distributed Intelligent Nodes as Information Filters in Advanced Telecare Systems", Proc. First Joint BMES/EMBS Conf., pp. 703, 1999.
37. J. Carter, M. Rosen, "Unobtrusive Sensing of Activities of Daily Living: A Preliminary Report", Proc. First Joint BMES/EMBS Conf., pp. 678, 1999.
38. E. Gottlieb, J. Vu, J. Winters, "Home Healthcare Wireless Physiological Monitoring: Systems Integration and Human Factors Evaluation", Proc. First Joint BMES/EMBS Conf., pp. 688, 1999.
39. M. Yamaguchi, M. Mitsumori, Y. Kano, "Noninvasively Measuring Blood Glucose Using Saliva", IEEE Eng. Med. Biol. Mag., Vol. 17, No. 3, pp. 59-63, 1998.
40. M. Arimoto, Y. Yozenawa, "A Wrist-Mounted Activity and Pulse Recording System", Proc. First Joint BMES/EMBS Conf., pp. 695, 1999.
41. K. Nakajima, "A Method for Evaluating Water Surface Fluctuation During Bathing", Proc. First Joint BMES/EMBS Conf., pp. 695, 1999.
42. A. Kawarada, A. Tsukada, K. Sasaki, et al, "Automated Monitoring System for Home Health Care", Proc. First Joint BMES/EMBS Conf., pp. 694, 1999.
43. G. Williams, K. Doughty, K. Cameron, D. Bradley, "A smart fall & activity monitor for telecare applications", Proc. 20th Ann. Int. Conf. IEEE Eng. In Med & Biol. Soc., Hong Kong, 1998.
44. K. Doughty, R. Isak, P. J. King, P. Smith, G. Williams, "MIDAS – Miniature Intelligent Domiciliary Alarm System – A Practical Application of Telecare", Proc. First Joint BMES/EMBS Conf., pp. 691, 1999.
45. K. Shimuzu, "Telemedicine by Mobile Communication", IEEE Eng. Med. Biol. Mag., Vol. 18, No. 4, pp. 32-44, 1999.
46. Mulder, "Hand Centered Studies of Human Movement Project", Technical Report 94-1: Human Movement Tracking Technology, 1994.
47. Padilla\_ruiz F. et al., "Frequency, characteristics and consequences of falls in a cohort of institutionalized elderly patients", Atencion Primaria, 1998.
48. Dresner\_Pollak et al., "Characteristics of falls in 70 year olds in Jerusalem.", Israel Journal of Medical Sciences, 1996.
49. Downton J.H., "Prevalence, characteristics and factors associated with falls among the elderly living at home", 1991.
50. Tinetti M.E. et al., "A multifactorial intervention to reduce the risk of falling among elderly people living in the community.", >New England Journal of Medicine, 1994.
51. Tinetti, M.E., "A multifactorial approach to reducing injurious falls.", Clinics in Geriatric Medicine, 1996.
52. Donald I.P, Bulpitt C.J., "The prognosis of falls in elderly people living at home.", 1999.
53. Tennsted S. et al., "A randomized, controlled trial of a group intervention to reduce fear of falling and associated activity restriction in older adults.", Journals of Gerontology, Psychological Sciences and Social Sciences, 1998.
54. K. Cameron, K. Hughes, K. Doughty; "Reducing Fall Incidence In Community Elders by Telecare Using Predictive Systems" Proceedings – 19<sup>th</sup> International Conference – IEEE/EMBS, 1997.
55. Horak F.B., "Foundation for Physical Therapy. Contemporary Management of Motor Control Problems. Assumptions underlying motor control for neurological rehabilitation", Proceedings of the II-STEP Conference, 1991.

56. Tinetti E., "Performance-oriented assessment of mobility problems in elderly patients", *J. Am. Geriatrics Soc.*, 1986.
57. Lee L.W., Kerrigan D.C. "Identification of kinetic differences between fallers and nonfallers in the elderly.", *American Journal of Physical Medicine and Rehabilitation*, 1999.
58. Lee L.W., Kerrigan D.C. et al. ">Kinetic alterations independent of walking speed in elderly fallers.", *Archives of Physical Medicine and Rehabilitation*, 2000.
59. Horak F.B., Nashner L.M., "Central programming of postural movements: adaptation to altered support-surface configurations.", *Journal of Neurophysiology*, 1986.
60. Gael LeBellego, Norbert Noury, Gilles Virone, Mireille Mousseau, and Jacques Demongeot, "A Model for the Measurement of Patient Activity in a Hospital Suite". *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, no. 1, 2006.
61. Suhuai Luo, Qingmao Hu, "A Dynamic Motion Pattern Analysis Approach to Fall Detectgion". *IEEE International Workchop on Biomedical Circuits & Systems*, 2004.
62. G. Williams, K. Doughty, K. Cameron, D.A. Bradley, "A Smart Fall & Activity Monitor for Telecare Applicarions". *Proceedings of the 20th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Vol. 20, No 3, 1998.
63. Toshiyo Tamura, Takumi Yoshimura, Fumitaka Horiuchi, Yuji Higashi, Toshiro Fujimoto, "An Ambulatory Fall Monitor for the Elderly". *Proceedings of the 22nd Annual EMBS International Conference*, 2000.
64. Derek Anderson, James M. Keller, Marjorie Skubic, Xi Chen, Zhihai He, "Recognizing Falls from Silhouettes" *Proceedings of the 28th IEEE EMBS Annual International Conference*, 2006.
65. Norbert Noury, Thierry Hervé, Vicent Rialle, Gilles Virone, Eric Mercier, Gilles Morey, Aldo Moro, Thierry Porcheron; "Monitoring Behaviour in Home Using a Smart Fall Sensor and Position Sensors". *1st Annual International IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine & Biology*, 2000.
66. Horiuchi, F.; Kadoya, R.; Higasi, Y.; Fujimoto, T.; Sekine, M.; Tamura, T.; "Evaluation by accelerometry of walking pattern before falls in hemiplegic patients" *Engineering in Medicine and Biology Society. Proceedings of the 23rd Annual International Conference of the IEEE Volume 2*, 2001.
67. Ahsan H. Khandoker, Rezaul K Begg, Marimuthu Palaniswami, "Estimating Falls Risk in the Elderly: a Wavelet Based Multiscale Analysis", *4<sup>th</sup> International Conference on Electrical and Computer Engineering ICECE 2006*.
68. Guangyi Shi, Cheung-Shing Chan, Yilun Luo, Guanglie Zhang, Wen J. Lii, Philip H. W. Leong, Kwok-Sui Leung, "Development of a Human Airbag System for Fall Protection Using MEMS Motion Sensing Technology", *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006.
69. Norbert Noury, "A Smart Sensor for the Remote Follow Up of Activity and Fall Detection of the Elderly" *2nd Annual International IEEE-EMBS Special Topic Conference on Mictotechnologies in Medicine & Biology*, 2002.
70. B. Najafi, K. Aminian, F. Loew, Y. Blanc, Ph. Robert, "An Ambulatory System for Physical Activity Momtoring in Elderly", *1st Annual Intemational IEEE-*

- EMBS Special Topic Conference on Microtechnologies in Medicine & Biology, 2000.
71. Bijan Najafi, Kamiar Aminian, Anisoara Paraschiv-Ionescu, François Loew, Christophe J. Büla, Philippe Robert, “Ambulatory System for Human Motion Analysis Using a Kinematic Sensor: Monitoring of Daily Physical Activity in the Elderly”. *IEEE Transactions on Biomedical Engineering*, Vol. 50, No. 6, 2003.
  72. Suhuai Luo, Qingmao Hu, “A Dynamic Motion Pattern Analysis Approach to Fall Detectgion”. *IEEE International Workchop on Biomedical Circuits & Systems*, 2004.
  73. J.Y. Hwang, J.M. Kang, Y.W. Jang , H. C. Kim, “Development of Novel Algorithm and Real-time Monitoring Ambulatory System Using Bluetooth Module for Fall Detection in the Elderly” *Proceedings of the 26th Annual International Conference of the IEEE EMBS*, 2004.
  74. Jay Chen, Karric Kwong, Dennis Chang, Jerry Luk, Ruzena Bajcsy, “Wearable Sensors for Reliable Fall Detection” *Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, 2005.
  75. Toshiyo Tamura, “Wearable accelerometer in clinical use” *IEEE Engineering in Medicine and Biology 27th Annual Conference Shanghai*, 2005.
  76. U. Lindemann, A. Hock, M. Stuber, W. Keck, C. Becker; “Evaluation of a fall detector based on accelerometers: a pilot study” *Medical & Biological Engineering & Computing* 2005.
  77. M. Prado, J. Reina-Tosina, L. Roa, “Distributed Intelligent Architecture for Falling Detection and Physical Activity Analysis in the Elderly”. *Proceeding of the Second Joint EMBS/BMES Conference*. 2002.
  78. A. Díaz, M. Prado, L.M. Roa, J. Reina-Tosina, G. Sánchez, “Preliminary evaluation of a full-time falling monitor for the elderly” *Proceedings of the 26th Annual International Conference of the IEEE EMBS San Francisco,CA, USA • September 1-5, 2004*.
  79. Manuel Prado, Laura M. Roa, Javier Reina, “Evaluación de la función de detección de caídas de un monitor portable del movimiento humano”. *Actas de XXII Congreso Anual de la Sociedad Española de Ingeniería Biomédica CASEIB 2004*.
  80. Manuel Prado, Laura M. Roa, Javier Reina-Tosina, Manuel Rovayo, “Wearable human movement monitoring device aimed at providing personalized support in healthcare”. *Smart Sensors, Actuators, and MEMS II, Proceedings of SPIE*, Vol. 5836, 2005.
  81. Guillermo Sánchez, Javier Reina-Tosina, Laura M. Roa, Manuel Prado, Alberto Díaz; “Diseño y Evaluación de Protocolos de Comunicaciones para un Sistema de Teleasistencia Domiciliaria de Personas Mayores”, *XXI Congreso Anual de la Sociedad Española de Ingeniería Biomédica*, 2003.
  82. Prado, M.; Roa, L.; Reina-Tosina, J. “Metabolic activity monitoring in a telehealthcare system for chronic patients” *Enterprise Networking and Computing in Healthcare Industry, Proceedings. 5th International Workshop on* 2003.
  83. M. Athans, J. J. S. Sentieiro, and M. A. Dahleh. *Workshop on future directions in systems and control theory. Cascais, Portugal, June 2000*.
  84. H. Nijmeijer and J. M. Schumacher. *Four decades of mathematical system theory. In J. W. Polderman and H. L. Trentelman, editors, The Mathematics of*

- Systems and Control: From Intelligent Control to Behavioral Systems, pages 73–83. Univ. of Groningen, 1999.
85. R. W. Brockett. New issues in the mathematics of control. In B. Engquist and W. Schmid an W. Schmid, editors, *Mathematics Unlimited—2001 and Beyond*, pages 189–220. Springer Verlag, 2000.
  86. P. R. Kumar. New technological vistas for systems and control: The example of wireless networks. *Control Systems Magazine*, 21(1):24–37, 2001.
  87. Roa, L.M.; Trujillo, F.; Gómez Cía, P.T., "A model of the plasma glucosa control using the system dynamic approach", *Modelling, Simulation & Control*, C, AMSE Press, 5, 3, 1986, p. 15-30.
  88. Roa L.M., "Modelos matemáticos del control respiratorio", *Neurobiología de las Funciones Vegetativas*, 1998, p. 231-255.
  89. Gentilini, A.; Frei, Ch.W.; Glattfedler, A.H.; Morari, M.; Sieber, Th.J.; Wymann, R.; Schnider, Th.W.; Zbinden, A.M., "Multitasked Closed-Loop Control in Anesthesia", *IEEE Engineering in Medicine and Biology*, 2001.
  90. National Research Council. *Embedded, Everywhere: A Research Agenda for Networked Systems of Embedded Computers*. National Academy Press, 2001
  91. Richard M. Murray, Karl J. Aström, Stephen P. Boyd, Woger W. Brockett, Guner Stein, "Future Directions in Control in an Information-Rich World", *IEEE Control Systems Magazine*, 2003.
  92. R. Taylor, D. Stoianovici, "Medical Robotics in Computer-Integrated Surgery", *IEEE Transactions on Robotics and Automation*, Vol. 19, Num. 5, pp. 765-781, 2003.
  93. Reina L.J., Roa L., Prado M. "Design of antennas for a wearable sensor for homecare movement monitoring" *Proceedings of the 28th IEEE EMBS Annual International Conference*, pp. 5972-5976, 2006.
  94. J. G. Jiménez, L. M. Roa, M. Prado, "Diseño de una unidad inteligente de acelerometría de segunda generación para la monitorización del movimiento humano", *Actas Del Xxv Congreso Anual De La Sociedad Española De Ingeniería Biomédica*, 2007.
  95. A. Díaz, M. Prado, L.M. Roa, J. Reina-Tosina, G. Sánchez, "Preliminary evaluation of a full-time falling monitor for the elderly" *Proceedings of the 26th Annual International Conference of the IEEE EMBS San Francisco, CA, USA*, 2004.
  96. Manuel Prado, Laura M. Roa, Javier Reina, "Evaluación de la función de detección de caídas de un monitor portable del movimiento humano". *Actas de XXII Congreso Anual de la Sociedad Española de Ingeniería Biomédica CASEIB 2004*.
  97. Manuel Prado, Laura M. Roa, Javier Reina-Tosina, Manuel Rovayo, "Wearable human movement monitoring device aimed at providing personlized support in healthcare". *Smart Sensors, Actuators, and MEMS II, Proceedings of SPIE*, Vol. 5836, 2005.
  98. L. Fernández del Rosal, M. Prado Velasco, M. A. Estudillo Valderrama, "Diseño hardware y autonomía del servidor personal inalámbrico (PSE) de un monitor de movimiento humano". *Actas de XXV Congreso Anual de la Sociedad Española de Ingeniería Biomédica CASEIB 2007*.
  99. Texas Instruments. G. Frantz, R. Simar; "Comparing Fixed- and Floating Point DSPs". White Paper. Disponible: <http://focus.ti.com/lit/ml/spry061/spry061.pdf>

100. Texas Instruments.. Disponible:  
<http://focus.ti.com/dsp/docs/dspplatformscontenthttp.jsp?sectionId=2&familyId=132&tabId=489>
101. Texas Instruments. B. Cobb, A. Seely, N. Zoran; “TMS320C6713 to TMS320C672x migration Guide”. Technical Document.. Disponible:  
<http://focus.ti.com/lit/an/spraa78/spraa78.pdf>
102. Texas Instruments. “TMS320C6727, TMS320C6726, TMS320C6722 Floating- Point Digital Signal Processors”. Data Sheet. Disponible:  
<http://focus.ti.com/lit/ds/symlink/tms320c6727.pdf>
103. Texas Instruments. “TMS320C672x DSP Peripherals Overview Reference Guide”. User Guide. Disponible:  
<http://focus.ti.com/lit/ug/spru723b/spru723b.pdf>
104. E. Sakka, A. Prentza, I. E. Lamprinos, L. Leondaridis, and D. Koutsuris, “Integration of Monitoring Devices in the e-Vital Service,” Proc. 26th Annual Intl. Conf. of the IEEE EMBS, pp. 3097–3100, San Francisco, Sep. 1-5, 2004.
105. S.-H. Choi, B.-K. Kim, J. Park, C.-H. Kang, and D.-S. Eom, “An implementation of wireless sensor network for security system using Bluetooth,” IEEE Trans. Consumer Electronics, vol. 50, no. 1, pp.236–244, Feb. 2004.
106. B. Radunovic, H. L. Truong, and M. Weisenhorn, “Receiver Architectures for UWB-Based Transmit-Only Sensor Networks ,” 2005 IEEE Intl. Conf. on Ultra-Wideband, pp. 379–384, Sep. 5-8 2005.
107. T.-H. Lin, W. J. Kaiser, and G. J. Pottie, “Integrated low-power communication system design for wireless sensor networks,” IEEE Communications Magazine, vol. 42, no. 12, pp. 142–150, Dec. 2004.
108. F. Lamberti and C. Demartini, “Low-cost home monitoring using a Java-based embedded computer,” Proc. 4th Special Topic Conf. Information Technology Applications in Biomedicine, pp. 342–345, Apr. 2003.
109. R. Isais, K. Nguyen, G. Perez, R. Rubio, and H. Nazeran, “A lowcost microcontroller-based wireless ECB-blood pressure telemonitor for home care,” Proc. 25th Annual Intl. Conf. of the IEEE EMBS, pp. 3157–3160, Cancun, Mexico, Sep. 17-21, 2003.
110. C. Lynch and F. O’Reilly, “PIC-based TinyOS Implementation,” Proc. 2nd European Workshop on Wireless Sensor Networks, pp. 378–385, Istanbul, Jan. 31 - Feb. 2, 2005.
111. M. Prado, L. M. Roa, J. Reina-Tosina, M. Rovayo, “Wearable human movement monitoring device aimed at providing personalized support in healthcare,” Microtechnologies for the New Millenium. Smart Sensors, Actuators, and MEMS II. Proceeding of SPIE, vol. 5836, pp. 570–578, Seville, May 9-11, 2005.
112. J. A. Gutierrez, M. Naeve, E. Callaway, M. Bourgeois, V. Mitter, and B. Heile, “IEEE 802.15.4: a developing standard for low-power lowcost wireless personal area networks,” IEEE Network, vol. 15, no. 5, pp. 12–19, Sep./Oct. 2001.
113. Texas Instruments. “CC2430 A True System-on-Chip solution for 2.4 GHz IEEE 802.15.4 / ZigBee™”. Data Sheet. Disponible:  
<http://focus.ti.com/lit/ds/swrs036e/swrs036e.pdf>
114. ”Bluetooth Specification Version 2.0 + EDR”. Disponible:  
<http://bluetooth.com/Bluetooth/Learn/Technology/Specifications/>
115. Datasheet del STLC 2500C  
<http://www.st.com/stonline/products/literature/bd/12019/stlc2500c.pdf>

116. "Infrared Data Communication According to IrDA Standard", Vishay Semiconductors, Septiembre 2006.
117. Texas Instruments. "TSM320C67x/C67x+ DSP CPU and Instruction Set Reference Guide". User Guide. Disponible: <http://focus.ti.com/lit/ug/spru733a/spru733a.pdf>
118. Spansion. "S29AL008De8 Megabit (1 M x 8-Bit/512 K x 16-Bit) CMOS 3.0 Volt-only Boot Sector Flash Memory". Data Sheet. Disponible: [http://www.spansion.com/datasheets/s29al008d\\_00\\_a7\\_e.pdf](http://www.spansion.com/datasheets/s29al008d_00_a7_e.pdf)
119. Texas Instruments. "TMS320C672X DSP Multichannel Audio Serial Port (McASP) Reference Guide". User Guide. Disponible: <http://focus.ti.com/lit/ug/spru878a/spru878a.pdf>
120. "Battery Products & Technology. Solutions for OEM Design Engineers, Integrators & Specifiers of Power Management Products", Michael Wang, Reprinted from the March 2004 issue of Battery Power Products & Technology magazine.
121. "Interferencias electromagnéticas en sistemas electrónicos", Joseph Balcells et al, Ed. MARcombo-Boixerau, 1992.
122. "Analog-To-Digital Converter Grounding Practices Affect System Performance", Application Bulletin, Texas Instruments
123. M.A. Estudillo Valderrama, M. Prado, L. Fernández del Rosal, "Diseño software de un servidor personal (PSE) para el control y procesamiento de señales provenientes de sensores inteligentes". Actas de XXV Congreso Anual de la Sociedad Española de Ingeniería Biomédica CASEIB 2007.
124. Rulph Chassaing. Digital Signal Processing and Applications with the C6713 and C6416DSK. J Wiley & Sons, 2005 (ISBN: 0-471-69007-4).
125. Texas Instruments "TMS320C DSP Power-Down Logic and Modes Reference Guide". <http://focus.ti.com/lit/ug/spru728c/spru728c.pdf>
126. Texas Instruments, "TMS320 DSP/BIOS User's Guide" Disponible en: <http://focus.ti.com/lit/ug/spru423f/spru423f.pdf>
127. Texas Instruments, "TMS320C6000 Peripherals Reference Guide" Disponible en: <http://focus.ti.com/lit/ug/spru190d/spru190d.pdf>
128. Texas Instruments, "TMS320C6000 DSP Inter-Integrated Circuit (I2C) Module Reference Guide" Disponible en: <http://focus.ti.com/lit/ug/spru175d/spru175d.pdf>
129. Cobb B. TMS320C672x Power Consumption Summary. <http://www.ti.com/litv/pdf/spraaa4b>
130. Aplicación Excel descrita en [129] disponible en <http://www.ti.com/litv/zip/spraaa4b>

Nomenclatura – Acrónimos

ACK	Acknowledge
ASK	Asentimiento
BT	Bluetooth
CAD	Covertidor Analógico-Digital
CA2	Complemento a Dos
CEN	Comité Europeo de Normalización
COG	Chip On Glass
DAC	Convertidor Digital-Analógico
DSP	Procesador de Señales Digitales
EDMA	Enhanced Direct Memory Access
ECG	Electrocardiograma
EMI	Interferencia ElectroMagnética
EMIF	External Memory Interface
E/S	Entrada / Salida
FSK	Modulacion por Desplazamiento en Frecuencia
GIB	Grupo de Ingeniería Biomédica de la Universidad de Sevilla.
HSL	Heat Seal Connector
HL7	Health Level 7
IAU	Unidad de Acelerometría Inteligente
IDE	Entorno de Desarrollo
irDA	Estandar Infrarrojo
I2C	Inter-Integrated Circuits
MFLOPS	Millones de Operaciones en Punto Flotante por Segundo
MIPS	Millones de Instrucciones por Segu

MSB	Most Significant Bit
openEHR	open Electronic Health Record
PBGA	Plastic Ball Grid Array
PSE	Personal Server: Servidor Personal
ROM	Read Only Memory
SMT	Surface- Mount Technology
SoC	System on Chip
SPI	Serial Peripheral Interface
TI	Texas Instruments
TIS	Tecnologías de Información Sanitarias
WAN	Red de Área Extensa
WLAN	Red de Área Local Inalámbrica
WPAN	Wireless Personal Area Network: Red de Área Personal Inalámbrica
24/7	24 horas al día, 7 días a la semana.