

Propuesta de una Nueva Arquitectura de Software para uso del Estándar ISO/IEEE 11073 en Dispositivos Médicos de Limitada Capacidad de Procesado y Memoria

M. Martínez-Espronedada¹, J. Escayola², S. Led¹, L. Serrano¹, I. Martínez², J. D. Trigo², J. García²

¹ Dep. Ingeniería Eléctrica y Electrónica (Univ. Pública Navarra) - Campus de Arrosadía s/n. 31006 - Pamplona, Spain

² Instituto de Investigación en Ingeniería de Aragón (Univ. Zaragoza) - c/ María de Luna, 3. 50018 - Zaragoza, Spain

{miguel.martinezdeespronedada, lserrano, sled}@unavarra.es, {jescajola, imr, jtrigo, jogarmo}@unizar.es

Resumen

Este trabajo analiza las características típicas encontradas en dispositivos médicos (Medical Devices, MD) llevables típicamente utilizados en unidades de cuidados intensivos (UCI), en sistemas de telemonitorización y en dispositivos de salud personal. Además, focalizando los esfuerzos en un estándar altamente aceptado para comunicaciones de MD como lo es la familia de normas ISO/IEEE 11073 (X73) y teniendo en cuenta las características de estos se propone una nueva metodología de implementación del estándar X73 para dispositivos de capacidades limitadas, con la cual, los requerimientos de memoria, potencia de procesamiento y consumo de potencia se mantienen extraordinariamente bajos. Además el trabajo introduce un método para testeo de conformidad para implementaciones basadas en la metodología propuesta.

1. Introducción

Los servicios de telemedicina mejoran considerablemente la calidad de vida del paciente y reducen las listas de espera ofreciendo un servicio mejor. El paradigma que propone, basado en Tecnologías de la Información y las Comunicaciones (TIC), está motivando la aparición de modelos de servicio de salud que apoderan al paciente [1]. Un amplio campo en telemedicina son los servicios de telemonitorización, que se focalizan mayoritariamente en personas ancianas, enfermos crónicos, pacientes bajo cuidados paliativos o pacientes que han sobrellevado una operación de cirugía. Puntos clave en estos sistemas son una alta seguridad de operación, la ergonomía y las capacidades *Plug&Play* (P&P), que deben de ser resueltas a través de la estandarización si se pretende ser efectivos. El estándar más maduro en este sentido es ISO/IEEE 11073 (X73) [2]-[5], el cual fue desarrollado inicialmente para Comunicaciones entre Dispositivos Médicos en el Punto de Cuidado (*Point of Care*, PoC) en Unidades de Cuidado Intensivo (UCI) y está siendo el centro de una continua sinergia de grupos de estandarización, universidades y fabricantes de equipamiento para desarrollarlo y adaptarlo a los requerimientos de los nuevos escenarios que han aparecido con la telemedicina y la salud personal. Uno de los desarrollos más importantes del estándar lo está llevando a cabo el grupo *Personal Health Devices* (PHD) [6],[7] que tiene como

objetivo definir un perfil más ligero que los existentes para dispositivos usados en salud personal y conforme a X73. Las desventajas del estándar X73 son: su complejidad inherente, el tiempo que necesita un desarrollados de dispositivos médicos (*Medical Devices*, MD) para entenderlo, los altos requerimientos del *hardware* necesario para ejecutarlo (que usualmente fija el tipo de plataforma emplear), la restricción a puerto serie (RS-232) o infrarrojos (IrDA), y la falta de tanto herramientas de testeo como de conformidad con X73. Varios esfuerzos han sido llevados a cabo para adaptar X73 a otras tecnologías de transporte [8]-[10], tales como Bluetooth [11], Ethernet, TCP/IP, WiFi [12], [13] y, en un futuro cercano, USB y ZigBee [14].

El estándar X73 define completamente los mensajes intercambiados y los diagramas de secuencia en una comunicación punto a punto con un MD y establece dos roles en la comunicación: el *agente*, que guarda los datos que van a ser transferidos, y el *manager*, que acepta y procesa los datos del *agente*. X73 También distingue tres actores en una UCI o en un sistema de telemonitorización: el dispositivo médico (MD), que adquiere datos del paciente, el elemento central de computación (*Compute Engine*, CE) que recibe datos de los MD y los post-procesa, y el servidor de monitorización (*Monitoring Server*, MS), que recibe todos los datos post-procesados del todos los CEs [15].

Nuestro trabajo tiene por objetivo cumplir tanto con los requerimientos de X73 como con los requerimientos de los fabricantes proponiendo una arquitectura que permite realizar implementaciones compatibles con X73 en dispositivos con el rol de *agente*, requiriendo baja capacidad tanto de procesado como de memoria.

Este artículo está organizado de la siguiente manera. En la [Sección 2](#) se presenta el concepto básico de patrón y las características tanto *hardware* como *software* más generales de los MD. En la [Sección 3](#) se describe la arquitectura de la solución propuesta, mostrando sus características técnicas y detallando el proceso de desarrollo a seguir. Finalmente la [Sección 4](#) concluye el artículo.

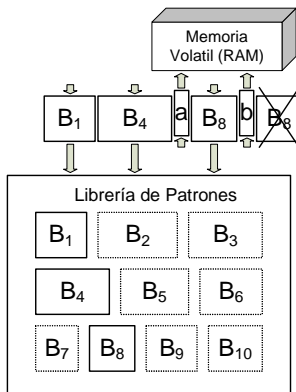
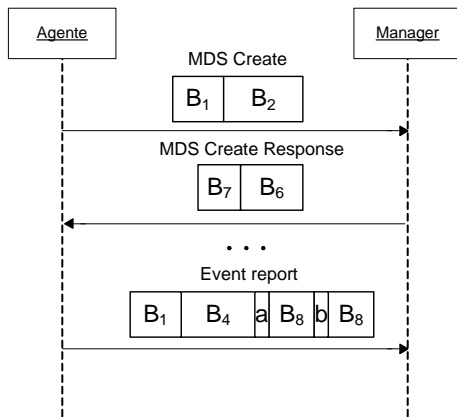
2. Metodología

La metodología propuesta en este trabajo se basa en la idea de patrón (Fig 1.a). Un patrón es un modelo o arquetipo que se utiliza para producir elementos de características similares a él. En nuestro caso, esta idea se aplica a los mensajes intercambiados en una transferencia basada en X73 para un agente específico y su manager.

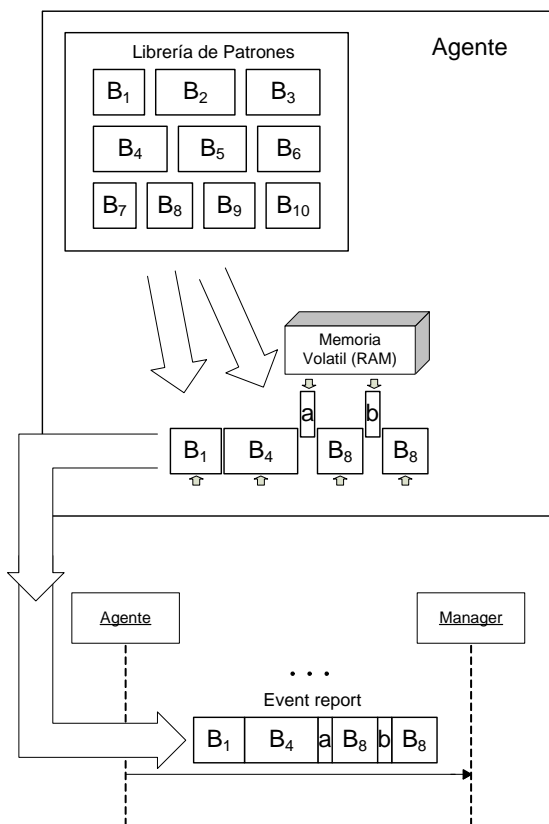
Realizando un estudio exhaustivo de todos estos posibles mensajes, se observa que hay un elevado número de mensajes que difieren únicamente en unos pocos bytes. Los mensajes en X73 se pueden representar por una estructura abstracta (*Abstract Syntax Notation One*, ASN1) codificada siguiendo las Reglas de Codificación Básicas (Basic Encoding Rules, BER) o las Reglas de Codificación para MDs (MDs Encoding Rules, MDER). Analizando dichas estructuras [4], [5], se comprueba que gran parte de los atributos que contienen permanecen constantes por lo que se pueden identificar bloques de bytes dentro de los mensajes X73 que permanecen invariantes. A dichos bloques denominaremos patrones.

El contenido de estos patrones son metadatos de X73 o, dicho de otro modo, información que se emplea para transferir un contexto para los datos relevantes como las mediciones de parámetros observadas. Dichos metadatos son el resultado de representar las estructuras codificadas en ASN1 a las codificaciones BER y MDER que realizan los elementos de asociación (*Association Control Service Element*, ACSE, y *Common Medical Device Information Service Element*, CMDISE, respectivamente). Los metadatos más comúnmente repetidos suelen portar información, por ejemplo, de identificadores de objetos y sus clases, identificadores de las unidades utilizadas al realizar la medición, tamaños de los campos de los objetos, etc. Esta metainformación es necesaria para dar capacidad P&P a los manager. Con ella, un manager genérico no debería necesitar configuración previa para acceder a un dispositivo conforme a X73 y, de esta forma, manipularlo. Una vez identificados los patrones mediante análisis de las estructuras de ASN1 definidas en X73, se pueden descartar todos los que están duplicados y agrupar todos los resultantes en una librería de patrones.

Determinados los patrones de una comunicación, cada uno de los mensajes intercambiados puede ser sintetizado a través de una permutación que parte de un conjunto concreto de patrones de la librería de patrones y una serie de variables como se muestra en la Fig 1.b. Puede intuirse que la capacidad de procesamiento requerida para la generación de cada uno de los mensajes va a ser muy reducida en comparación a una implementación no especializada [16], [17] y va a estar al alcance de cualquier MD del mercado. Algunos ejemplos de las variables empleadas en la generación de los mensajes son `InvokeId` usado para la identificación del procedimiento remoto o `ObservedValue` con el dato relevante (obtenido, por ejemplo de la última muestra capturada por el conversor analógico-digital, A/D). Los mensajes así sintetizados pueden transmitirse directamente o compararse con los recibidos. Tanto la transmisión como la comparación con un mensaje recibido conlleva



a) Concepción de librería de patrones



b) Síntesis de mensajes X73 a partir de sus patrones

Figura 1. Principio de funcionamiento

generalmente una serie de cambios internos en el MD como, por ejemplo, pasar de estado CONFIGURADO a CONECTADO, la inicialización de un conversor A/D, el cambio de estado en la máquina de estados finitos (*Finite State Machine*, FSM) o también modificaciones en alguno de los objetos del Modelo de Información de Dominio (*Domain Information Model*, DIM) y posiblemente sus atributos.

Un desarrollador de un determinado MD generalmente necesita desarrollar tanto el *hardware* como el *software*. La parte *hardware* típicamente incluye una placa con un microcontrolador, un chip de radiofrecuencia y un transductor. También es común encontrarse con dispositivos *System-on-Chip* (SoC) que incluyen los tres componentes en un único integrado. Una vez diseñado el *hardware* es necesario desarrollar el *software* que dote de inteligencia al dispositivo. Para este fin comúnmente se recurre a un entorno de desarrollo que permita un desarrollo acelerado del *software*, acortando así el tiempo de puesta en venta (*time-to-market*). Dicho entorno generalmente viene determinado por el *hardware* utilizado. Por ejemplo, en el caso de Bluetooth o ZigBee, el entorno de desarrollo viene determinado por la pila de comunicaciones, la cual incluye a su vez un Sistema Operativo en Tiempo Real (*Real Time Operating System*, RTOS) y una interfaz de programación de aplicaciones propietaria (*Application Programming Interface*, API).

3. Resultados

La gran variabilidad tanto de plataformas *hardware* como *software* impone una gran versatilidad a los entornos de desarrollo para estas. Nuestra propuesta, mostrada en la Fig. 2, es una arquitectura genérica para implementaciones multiplataforma. Está compuesta por un núcleo X73 mínimo y una librería de patrones. El núcleo X73 es un módulo (escrito en ANSI C [18]) que maneja todas las señales procedentes del resto de módulos y realiza todas las funciones relacionadas con las comunicaciones X73. Esto implica el procesado de la librería de patrones para obtener los mensajes pertinentes y, una vez obtenidos, compararlos con los recibidos o enviarlos a través de la capa de adaptación [19].

Este pequeño núcleo también se encarga de actualizar el estado de FSM, de modificar la información representada por el conjunto de objetos del DIM, y de atender algunas señales de sistema como envío y recepción de datos, conexión establecida, conexión perdida, señales de reloj para los *scanners* (tales como PeriCfgScanner), etc.

Tanto un núcleo X73 como su librería X73 solo pueden ser compartidos entre dispositivos que usan el mismo perfil de comunicación y la misma especialización de dispositivo (báscula, tensiómetro, glucómetro, etc.). Así, X73 define tres perfiles de comunicación [19]: por un lado, los clásicos de X73-PoC: perfil básico (*baseline*) y perfil modo sondeo (*polling*); y por otro, el nuevo perfil para dispositivos de salud personal (PHD). Además X73 define dos series de especializaciones de dispositivos: la serie 103xx para los perfiles clásicos y la serie 104xx para dispositivos que implementan el perfil PHD. De este modo, por ejemplo para una báscula, existirá una implementación del núcleo X73 y de la librería de patrones particular para cada uno de los tres perfiles.

Esta metodología permite reducir el consumo de recursos y de memoria al máximo. Lo que es más: el tipo de memoria empleado en esta aproximación es mayoritariamente de tipo no volátil, minimizándose el consumo de memoria volátil que normalmente es significativamente más escasa. Por consiguiente, como el consumo de memoria es muy bajo, la misma librería de patrones y el mismo núcleo (junto con su código) pueden ser empleados en un elevado número de MDs de forma que ambos pueden ser compartidos por un elevado número de fabricantes lo que beneficia la interoperabilidad ya que todos trabajan sobre el mismo código. Aunque un núcleo puede ser empleado en varios MDs, si estos utilizan la misma especialización y perfil de comunicación, todos los núcleos mantienen una buena base de código en común. Desde el punto de vista de un desarrollador novel, esta arquitectura, una vez desarrollada, ofrece un entorno de desarrollo rápido de aplicaciones (*Rapid Application Development environment*, RAD) para X73 que le permite el desarrollo de un dispositivo conforme a X73 sin necesidad de tener un conocimiento profundo sino con solo unos pocos conceptos básicos.

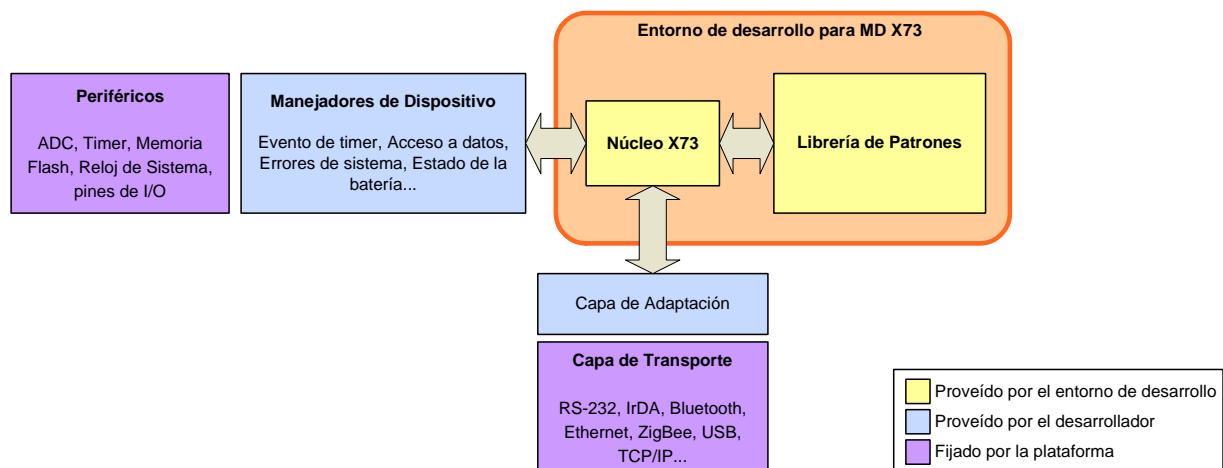


Figura 2. Arquitectura del entorno de desarrollo

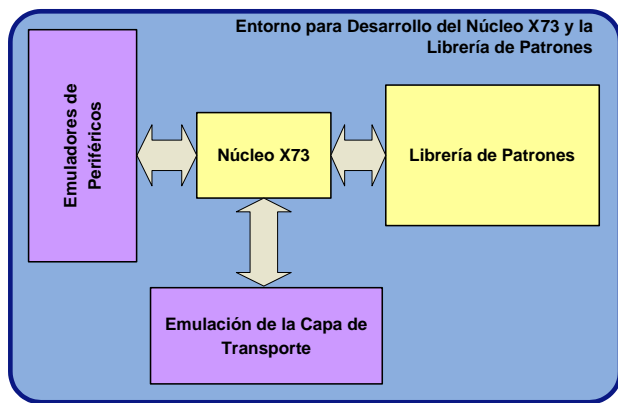


Figura 3. Herramienta de testeo para el núcleo X73 y la librería de patrones

Los módulos que deben de ser desarrollados son, por un lado, la capa de adaptación y, por otro, los gestores de dispositivo. La primera traduce las peticiones (*requests*) de envío de datos a las capas inferiores y proporciona servicios de señalización e interrupciones (*responses* y *indications*) de la capa de transporte de vuelta a las capas superiores; los segundos generan señales de indicación para el núcleo X73 tales como la expiración de un *timer*, y proveen servicios tales como acceso a datos. Por ejemplo en el caso de un pulsímetro, los gestores de dispositivo deben proporcionar al núcleo X73 de una señal de forma que le informe de cuándo una nueva muestra está lista además de métodos para acceder a está.

Además de las ventajas ya descritas de bajo consumo de memoria y procesado que ofrece esta arquitectura, otra que merece mención es la posibilidad de utilizar un entorno emulado para depuración y testeo de conformidad como muestra la Fig 3. En este caso, tanto la capa de adaptación como los gestores de dispositivos, se pueden sustituir por emuladores que trabajen directamente sobre el mismo ordenador personal en el que se carga la aplicación de testeo y conformidad. De esta forma, se puede probar la conformidad del núcleo X73 y de la plataforma de desarrollo sobre entornos más potentes y también más accesibles puesto que el único requerimiento es que el compilador sea compatible con ANSI C.

4. Conclusiones

Este artículo puede ser empleado como un punto de partida para desarrolladores que desean implementar el estándar X73 en un dispositivo con capacidades de procesamiento y memoria muy reducidas. En este artículo se ha presentado una arquitectura especialmente diseñada para MDs fruto de la experiencia basada en la idea de patrones. La idea principal en que se fundamenta es la alta redundancia del estándar y la aparición de bloques constantes en los mensajes intercambiados. Se espera que una vez desarrollada esta propuesta de arquitectura, se reduzca considerablemente el tiempo necesario para implementar un dispositivo basado en X73 y con ello facilite la adopción del estándar por parte del fabricante.

Agradecimientos

Este trabajo ha sido parcialmente subvencionado por los proyectos TSI2005-07068-C02-01 y TSI2007-65219-C02-01 de la Comisión Interministerial de Ciencia y Tecnología (CICYT) y Fondos Europeos para el Desarrollo Regional (FEDER), PET2006-0579 del Programa de Estímulo de Transferencia de Resultados de Investigación (PETRI), una beca FPI a M. Martínez-Esproncada (res. 1342/2006 de la Universidad Pública de Navarra), y una beca a Jesús Trigo (ref. IT7/08) de Diputación General de Aragón (DGA), Consejo Asesor de Investigación y Desarrollo (CONAID) y Caja de Ahorros de la Inmaculada (CAI).

Referencias

- [1] J. L. Monteagudo, O. Moreno, "e-Health for patient empowerment in Europe," *Instituto de Salud Carlos III, Ministerio de Sanidad y Consumo*, España, 2007
- [2] Comité Europeo de Normalización (CEN). <http://www.cen.eu>. Technical Committee 251 – Health informatics (CEN/TC251). <http://www.cen251.org>.
- [3] Asociación Española de Normalización (AENOR/CTN139) <http://www.aenor.es/desarrollo/inicio/home/home.asp>
- [4] ISO/IEEE11073 Point-of-Care Medical Device Communication standard (X73-PoC). Health informatics. [Part 1. Medical Device Data Language (MDDL)] [Part 2. Medical Device Application Profiles (MDAP)] [Part 3. Transport and Physical Layers]. <http://www.ieee1073.org>. See also the previous standards: IEEE13734-VITAL and ENV13735-INTERMED of CEN/TC251, <http://www.medicaltech.org>.
- [5] ISO/IEEE11073 - Personal Health Devices standard (X73-PHD). Health informatics. [P11073-00103. Technical report - Overview] [P11073-104xx. Device specializations] [P11073-20601. Application profile - Optimized exchange protocol]. IEEE Standards Association webpage: <http://standards.ieee.org/>.
- [6] M. Clarke *et al.*, "Developing a standard for personal health devices based on 11073," *Int Conf IEEE Eng in Medicine and Biology Society (EMBS)*, pp. 6174-6176, 2007.
- [7] L. Schmitt, T. Falck, F. Wartena and D. Simons, "Novel ISO/IEEE 11073 standards for personal telehealth systems interoperability," in *2007 Joint Workshop on High Confidence Medical Devices, Software, and Systems and Medical Device Plug-and-Play Interoperability - HCMDSS-MD PnP '07*, 2008, pp. 146-8.
- [8] S. Warren, J. Lebak y J. Yao, "Lessons learned from applying interoperability and information exchange standards to a wearable point-of-care system," *Transdisciplinary Conference on Distributed Diagnosis and Home Healthcare*, pp. 101-104, 2006.
- [9] J. Yao y S. Warren, "Applying the ISO/IEEE 11073 standards to wearable home health monitoring systems," *Journal of Clinical Monitoring and Computing*, vol. 19, pp. 427-436, 2005.
- [10] M. Martínez-Esproncada *et al.*, "Implementación de la norma ISO/IEEE 11073 en una báscula dotada de Bluetooth para la telemonitorización de peso", *Congreso SEIB*, pp.415-418, 2007.
- [11] Bluetooth SIG. "The Official Bluetooth Wireless Info Site," <http://www.bluetooth.com>,
- [12] The Working Group for Wireless Local Area Networks standards, <http://www.ieee802.org/11/>, Last access: 06/08.
- [13] B. P. Crow, I. Widjaja, J. G. Kim y P. T. Sakai, "IEEE 802.11 wireless local area networks," *IEEE Communications Magazine*, vol. 35, pp. 116-126, 1997.
- [14] P. Kinney, "ZigBee technology: Wireless control that simply works," *ZigBee technology: wireless control that simply works*, 2003.
- [15] M. Galarraga, L. Serrano, I. Martínez y P. de Toledo, "Standards for medical device communication: X73 PoC-MDC," *Medical and Care Computetics 3*, IOS Press, pp. 242-256, 2006.
- [16] I. Martínez *et al.*, "Implementation of an end-to-end standard-based patient monitoring solution," *IET Commun.*, vol. 2, pp. 181-191, 2008.
- [17] M. Galarraga *et al.*, "Proposal of an ISO/IEEE11073 Platform for Healthcare Telemonitoring: Plug-and-Play Solution with new USE Cases", *Int Conf IEEE Eng in Medicine and Biology Society (EMBS)*, pp 6709 – 6712, 2007.
- [18] JTC 1/SC 22, "ISO/IEC 9899:1999", *ISO Standards*, 2005
- [19] M. Martínez-Esproncada *et al.*, "Propuesta de una interfaz de transporte para permitir el uso de ISO/IEEE 11073 con diferentes tecnologías de transmisión", *Congreso SEIB*, pp.411-414, 2007.