

# Using Zigbee to Integrate Medical Devices

Paul Frehill, Desmond Chambers, Cosmin Rotariu

**Abstract**— Wirelessly enabling Medical Devices such as Vital Signs Monitors, Ventilators and Infusion Pumps allows central data collection. This paper discusses how data from these types of devices can be integrated into hospital systems using wireless sensor networking technology. By integrating devices you are protecting investment and opening up the possibility of networking with similar devices.

In this context we present how Zigbee meets our requirements for bandwidth, power, security and mobility. We have examined the data throughputs for various medical devices, the requirement of data frequency, security of patient data and the logistics of moving patients while connected to devices.

The paper describes a new tested architecture that allows this data to be seamlessly integrated into a User Interface or Healthcare Information System (HIS). The design supports the dynamic addition of new medical devices to the system that were previously unsupported by the system. To achieve this, the hardware design is kept generic and the software interface for different types of medical devices is well defined. These devices can also share the wireless resources with other types of sensors being developed in conjunction on this project such as wireless ECG (Electrocardiogram) and Pulse-Oximetry sensors.

**Keywords**—Biomedical Telemetry, Medical Devices, Bioinformatics, Wireless Sensor Networks, Healthcare Information Systems.

## I. INTRODUCTION

MANy devices that exist today by the bedside in the hospital ward, intensive care unit or other clinical setting have data output features over serial ports and other types of interfaces such as USB. These devices are usually considered a significant investment and are usually purchased in an ad hoc fashion as required when finance becomes available. The consequence of this is that devices are often from different manufacturers that don't support any standard protocol. This can make integrating these devices into a single network difficult.

In the hospital ward Vital Signs monitors, Ventilators and Infusion Pumps of many different brands are usually portable and wheeled from patient to patient as required. By networking these devices the hospital gains all the advantages associated with storing patient data centrally in electronic records. By making the device part of a wireless sensor network such as a Zigbee [1] network there are several more

advantages including, cable replacement, mobility and location management. Once these devices are networked they can also use the infrastructure of other deployments of similar wireless sensor networks in the surrounding environment.

To achieve this type of solution each device must be fitted with a piece of hardware that will act as a serial to wireless bridge, a Medical Device Interface (MDI). This MDI will allow the device to receive and transmit data within the wireless sensor network. This inexpensive hardware will be generic to fit a wide range of medical devices. Similarly the firmware can be kept generic and any specific device communication protocols can be implemented on a server on the network backend.

The work described in this paper is part of a larger project, the goal of which is to provide a complete patient monitoring system. Other features of the overall system will be to provide ECG (Electrocardiogram) and Pulse-Oximetry data in a novel way over a wireless sensor network using expertise gained on prior projects [2].

## II. RELATED WORK

The concept of using wireless sensor networks for Medical Care and wireless patient monitoring has been explored by others but integrating data from other devices is generally not discussed. There is ongoing related work in patient monitoring using wireless sensors such as the "CodeBlue" project at Harvard [3]. Others have also proven successful with wireless sensor networks designs for medical sensors [4] and in the management of sensor data [5]. It has been identified that it is desirable to wirelessly enable existing medical devices that provide vital signs data using technologies such as Zigbee [6], [7]. The research described in this paper aspires to meet these requirements. The use of wireless sensor networks within the hospital has been extensively examined. Moreover, other wireless technologies within the same frequency band, such as IEEE 802.11 [8], have existed within the hospital for some time [9].

## III. REQUIREMENTS ANALYSIS

### A. Wireless Technologies

Established standards for wireless applications, such as Bluetooth [10] and IEEE 802.11, allow high transmission rates, but at the expense of power consumption, application complexity, and cost. Zigbee offers low cost, low power devices that can communicate with each other and the outside

This work is supported by Enterprise Ireland.  
Paul Frehill is with the Department of IT, National University of Ireland Galway, Ireland (phone: 353-91-492040; e-mail: p.frehill@nuigalway.ie).  
Des Chambers is with the Department of IT, National University of Ireland Galway, Ireland (e-mail: des.chambers@nuigalway.ie).

Cosmin Rotariu is with the Department of IT, National University of Ireland Galway, Ireland (e-mail: crotariu@vega.it.nuigalway.ie).

world. ZigBee's self-forming and self-healing mesh-network architecture lets data and control messages pass from one node to another by multiple paths. This is particularly useful in a hospital environment where interference from walls, people and general obstacles is a major issue. Zigbee is based upon the IEEE standard 802.15.4 [11] for radio hardware and software specification.

### B. Mobility

Zigbee enabled devices support a sleep mode. An off-line node can connect to a network in about 30 ms. Waking up a sleeping node takes about 15 ms, as does accessing a channel and transmitting data. If the requirement is to collect data once a minute the device can be placed in a power saving mode saving significant amounts of energy and increasing the battery life. In sleep mode a zigbee chip can assume as little as 1.0uA [12]. This is particularly important in a medical setting where patients are often on the move while still attached to medical devices.

### C. Co-existence

Both Zigbee and IEEE802.11 operate in the license-free industrial scientific medical (ISM) 2.4GHz frequency band. IEEE802.11 is already in widespread use within hospitals which would encourage the adoption of Zigbee solutions in the same environment. However care has to be taken to avoid interference between these 2 neighbouring technologies as described in the paper entitled "Coexistence of IEEE802.15.4 with other systems in the 2.4GHz-ISM-band" [13]. By selecting an appropriate channel, after performing a simple site survey, these problems can be easily avoided.

### D. Device Parameters

Typical readings available on a ventilator are *Inspiratory Tidal Volume, Expiratory Tidal Volume, O<sub>2</sub> concentration, Respiratory Rate, Peak Pressure, Expired Minute Volume and Mean Airway Pressure*. The settings on the ventilator are also of interest to medical staff. The most typical settings we've chosen are *Inspiratory Tidal Volume, Minute Volume, O<sub>2</sub> Concentration, I:E Ratio, Breath Duration and Inspiratory Flow*.

Similarly we have chosen some common parameters for Vital Signs Monitors. These are *Respiratory Rate, Non Invasive Blood Pressure, SPO<sub>2</sub> and Temperature*.

The third device we selected parameters for is the Unfusion Pump. The common parameters we are most interested in here are *Volume, Time, Ramp and Occlusion Pressure*. Further parameters can be easily added to the system in the future.

### E. Bandwidth

For development purposes we analysed a Maquet Servo-I [14] which supports all the ventilator parameters described above. This ventilator works in a command response manner. When initial configuration has taken place 2 commands which are 7 bytes long each will produce 2 responses of 67 bytes each. Therefore even in a multi hop mesh network it is anticipated we would be able to support several of these devices plus other types of devices on the same 802.15.4

channel.

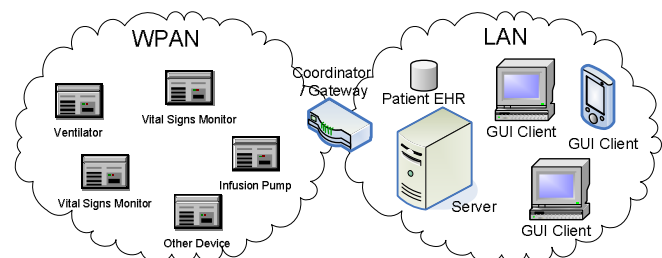
### F. Scalability

The ventilator, having the most parameters of the devices studied, requires the most bandwidth. Experiments carried out on a CSI Vital Signs Monitor [15] show that 44 bytes of data will produce all the information we are interested in. A Braun Infusion Pump [16] exports 24 bytes of data to produce the 4 parameters we need. For any of the medical devices we are concerned with, the readings are typically only required once a minute in a hospital environment. All these devices have their own alarm mechanisms built in; we are purely providing a means of exporting the data automatically. Theoretically a single Zigbee network could have above and beyond 600 Ventilators as each device only requires less than 1KB of bandwidth per minute. The frequency at which we capture the data is decided upon by the clinical staff themselves. A 1 minute interval is a typical value, however even if they were to require the data every few seconds it is clear the network could still support a large number of devices.

## IV. SYSTEM ARCHITECTURE

### A. High-Level Architecture

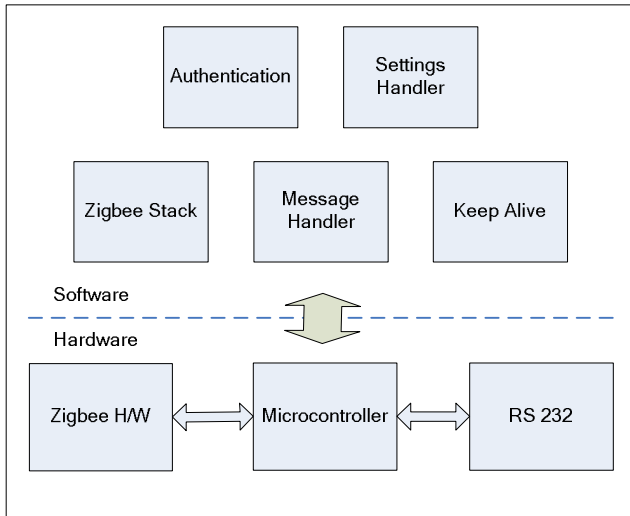
The overall System Architecture consists of a Wireless Personal Area Network (WPAN) and a Local Area Network. The WPAN implemented as a Zigbee network communicates with the LAN via a gateway. This gateway also serves as the WPAN coordinator which is responsible for forming the network. Each medical device has a Zigbee node attached (MDI) which enables data to be transmitted wirelessly to the Gateway and then onto a Server existing on the LAN. See Fig. 1 below for a graphical representation of this. When an MDI is powered on it automatically joins the network and makes itself known to the Server. A user can then associate this device with a patient using a GUI client. Once an association has been completed the MDI will be notified to begin transmitting data. Data received by the server will be stored in the Electronic Health Record (EHR) for that patient and displayed on any GUI Client that is subscribed.



**Figure 1. High-Level Architecture**

### B. Medical Device Interface

The diagram in Fig. 2 below shows the key components of the MDI. The hardware comprises of a Zigbee module, a microcontroller and an RS 232 Interface. The microcontroller is responsible for interfacing with both the RS232 Interface and the Zigbee module.

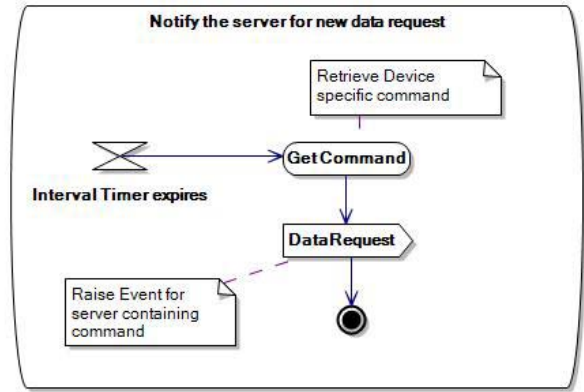


**Figure 2. MDI Block Diagram**

When the MDI is powered on, the Zigbee stack will automatically join a Zigbee network within range. Next the MDI will announce an ID which is also visible on the external surface of the device. This is done using a protocol we designed for this project. The protocol supports these types of status messages in addition to supporting the actual real data we are interested in. At this point it is possible to make an association with the MDI. To achieve this, the administrator selects the ID from an automatically generated list on screen, a patient demographic and a type of medical device which is supported in the system. This process results in the server sending the correct RS232 settings to the MDI for the medical device that it is connected to. Now that the system can communicate directly with the medical device the server will send any necessary commands to initiate a data stream from the device.

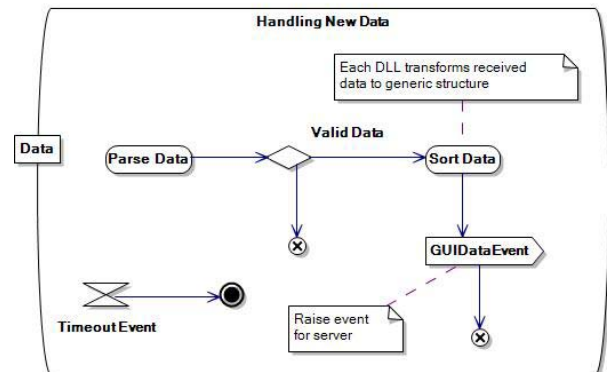
### C. Server Functionality

The server is responsible for decoding specific medical device data. This functionality is implemented in a DLL (Dynamic Link Library) that is run on the server. There is one DLL for each type of medical device the system supports which allows for future medical devices to be supported without upgrading the server software. Any future device can easily be supported within the DLL framework by simply inheriting from the appropriate class for that particular type of device. These DLLs are loaded at run-time and have a standard interface that each designer must adhere to in order to interoperate with the system. A designer must also complete an XML file from a template to indicate which features the new medical device supports. The DLL only handles device specific information; the main server application decodes this information from our project protocol.



**Figure 3. New Data Request**

When the DLL is loaded by the server application it will receive a value to represent how frequently the server wants GUI data. This value is used to create the interval timer represented in the above UML diagram. When this timer expires the DLL will check the current state it has for the ventilator. Fig. 3 above shows an activity diagram representing a sequence of events surrounding this timer expiration in a DLL for a ventilator. When the timer expires the DLL retrieves the command in the form of a byte array of ASCII characters. Next the DLL raises an event containing the byte array. The server application accepts this event through its event handler, encodes it in our protocol and sends it to the medical device.



**Figure 4. Handling New Data**

When the medical device returns a response to the server application this data is passed to the DLL as shown in Fig. 4 above. If the DLL does not receive any data within a specified timeout period the DLL will re-initiate communication with the medical device. Data that is received by the DLL is checked to see if it is consistent with the format expected and that the checksum is valid if applicable. Once the data is proven to be valid, a generic data structure that is shared between the server application and clients is populated. Finally the DLL will raise another event, this time to indicate that GUI data is available. The server then multicasts the new data to any client that is subscribed to that address.

The Server could be extended to integrate with existing

hospital systems using HL7 [17] or another uniform interface that is designed specifically for wireless sensor networks could be used such as that developed by DERI [18].

## V. LABORATORY RESULTS

The architecture described here has been implemented and tested successfully in a laboratory environment. For this purpose we connected the MDI to a PC simulator designed to act as a Maquet Servo-i ventilator. The Maquet ventilator can return breath readings and settings. To capture these, the SDADB and SDADS Maquet commands are sent to specify which breath readings and settings we wish to retrieve. Then by sending the RADAB and RADAS commands periodically we can capture up to date information from the ventilator. Fig. 6 shows the simulator handling these commands at runtime.

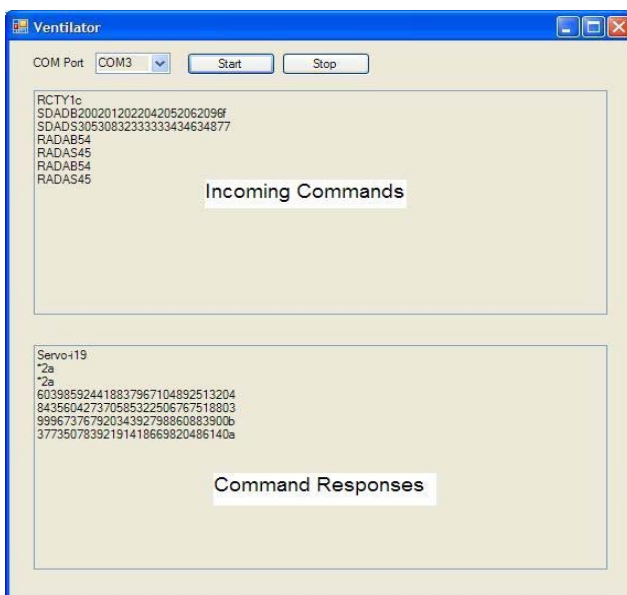


Figure 6. Ventilator Simulator

As described in the Architecture the Server DLL maintains a state for each end device so it knows what information to expect in return. In our experiments we successfully retrieved ventilator data at 5 second intervals. We have designed a GUI client which we successfully subscribed to receive this ventilator data. We are collaborating with a local hospital that use the Maquet ventilator and their requirement is data collection at 1 minute intervals. These results are very positive prior to usability tests in the hospital with the actual ventilator. We also performed a limited amount of range testing in the laboratory. We achieved a range of 20m within the confines of the laboratory which would equate to the maximum distance between an end device and a router in the hospital. In addition we carried out some mobility testing by moving the MDI during operation which did not result in any packet loss. Initial results are positive but further extensive testing is needed which will be performed in the hospital environment.

## VI. CONCLUSION

In this paper we have shown that a Wireless Sensor Network is a suitable means for capturing data from a medical device. We have discussed how Zigbee meets our requirements in terms of data throughput, power and mobility. Moreover, using this technology we can develop a low cost, scalable solution for a wide range of medical devices. In addition we have an infrastructure that allows us to easily support new devices within the system as needed. This architecture facilitates moving the device data to third party systems and to our own User Interface. We hope that our field trials will result in positive feedback from the clinical staff.

## REFERENCES

- [1] Zigbee Alliance, "Zigbee Specification", <http://www.zigbee.org>
- [2] C Rotariu, et al. "Network Enable PnP ECG Sensor for monitoring of Biomedical Signals". In proceeding of 5<sup>th</sup> RoEDuNet IEEE International Conference, June 2006, Pages 17-22.
- [3] K Lorincz, et al. "Sensor Networks for Emergency Response: Challenges and Opportunities". In *IEEE Pervasive Computing*, Special Issue on Pervasive Computing for First Response, Oct-Dec 2004.
- [4] A Milenkovic, C Otto, E Jovanov. "Wireless Sensor Networks for Personal Health Monitoring: Issues and Implementation". *Computer Communications*, Volume 29, Issues 13-14, 21 August 2006, Pages 2521-2533
- [5] J Herbert, et al. "Mobile Agent Architecture Integration for a Wireless Sensor Medical Application". *Web Intelligence and International Agent Technology Workshops*, 2006. WI-IAT 2006 Workshops. 2006 IEEE/WIC/ACM International Conference on Dec. 2006 Page(s):235 – 238
- [6] M Paksunemi, H Sorvoja, E Alasaarela, R Myllylä. "Wireless sensor and data transmission needs and technologies for patient monitoring in the operating room and intensive care unit". *Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, September 1-4, 2005
- [7] Medical Design Staff. "ZigBee Networks Open the Door to More Wireless Medical Devices". Available at: <http://www.medicaldesign.com/articles/ID/12338>
- [8] IEEE 802.11 Wireless LAN Standard. Available at: <http://standards.ieee.org/getieee802/802.11.html>
- [9] Frost & Sullivan. "Wireless LAN in the European Hospital Environment". B098-56, 2002.
- [10] Bluetooth Specification. Available at: <http://www.bluetooth.com/Bluetooth/Learn/Technology/Specifications/>
- [11] IEEE 802.15.4 standard. Available at: [www.ieee802.org/15/pub/TG4.html](http://www.ieee802.org/15/pub/TG4.html)
- [12] Ember Corporation. EM250 Single-chip Zigbee/802.15.4 Solution. Available at: [http://www.ember.com/pdf/EM250/120-0082-000H\\_EM250\\_Datasheet.pdf](http://www.ember.com/pdf/EM250/120-0082-000H_EM250_Datasheet.pdf)
- [13] A Sikora, V.F. Groza. "Coexistence of IEEE802.15.4 with other Systems in the 2.4 GHz-ISM-Band". *Instrumentation and Measurement Technology Conference, 2005. IMTC 2005. Proceedings of the IEEE 16-19 May 2005 Volume: 3*, On page(s): 1786- 1791
- [14] Maquet GmbH & Co. KG. Servo-I Universal Ventilator. Available: <http://www.maquet.com>
- [15] Criticare Systems Inc. 506N3 Vital Signs Monitor. Available: <http://www.csiusa.com/products.html>
- [16] B. Braun Medical Inc. Horizon NXT Infusion Pump. Available: <http://www.bbraunusa.com/products.html>
- [17] Health Level Seven, <http://www.hl7.org>
- [18] Shu Lei, Hui Xu, Wu Xiaoling, Zhang Lin, Jinsung Cho, Sungyoung Lee, "VIP Bridge: Integrating Several Sensor Networks into One Virtual Sensor Network," *icisp*, p. 2, International Conference on Internet Surveillance and Protection (ICISP'06), 2006