Demonstration Paper: Model-Based Wireless Health System Design Tool

Sunit Verma
IMPACT LAB, ASU, Tempe
sverma14@asu.edu

Joseph Milazzo
IMPACT LAB, ASU, Tempe
Joseph.Milazzo@asu.edu

Yu Xie
IMPACT LAB, ASU, Tempe
yqxie@asu.edu

Priyanka Bagade
IMPACT LAB, ASU, Tempe
pbagade@asu.edu

Ayan Banerjee
IMPACT LAB, ASU, Tempe
abanerj3@asu.edu

Sandeep K.S. Gupta
IMPACT LAB, ASU, Tempe
sandeep.gupta@asu.edu

ABSTRACT

Well-being and fitness are major focuses pushing the need for a simple and effective method to monitor health. Researchers have pointed out safety, lifetime, and reliability as the key requirements of medical devices. Mismatch between requirements of wearable medical sensor and smart phone and their implementation is one of the major causes of failure. We demonstrate a Wireless Health System (WHS) design tool, which abstracts detail between model and implementation and generates sensor and smart phone code.

Keywords

AADL, Wireless Health System, Automatic code generation, Middleware, Smart phone, Android.

1. INTRODUCTION

In the recent years, researchers have shown considerable interest in WHS where a group of wearable wireless medical devices sense physiological signals and send the information to a smart phone base station. Traditionally, a model-based approach has been undertaken for design and verification of WHS as it abstracts system implementation and simplifies design. However, manual code generation may lead to errors causing violation of the requirements. Further, diversities in hardware, Operating Systems (OS) and programming languages slows manual development of WHS. Automated code generation can reduce human errors in manual implementation and quickens the development process. For rapid development of WHS, we demonstrate a system that takes high-level model of WHS, which is already verified against requirements, and automatically generates the code.

2. System Model

2.1 Overview

The system as shown in Figure 1 is composed of the following components: *Health-Dev* [1], user interface (UI), and *Smartphone Middleware*. *Health-Dev* is an automatic code generation tool that takes system requirements in Architectural Analysis and Design Language (AADL) form and generates sensor and smart phone code. It is written as an Open Source Architectural Analysis and Design Language Tool Environment (OSATE) plug-in and hence can be readily used on verified BSN models. We use *Ayushman* [2] as a test bed for *Health-Dev*.

Health-Dev generates code by describing a WHS model in terms of architectural components and the connections between them. Each component is parameterized with specific properties which describes the function. *Health-Dev* parses the components and generates platform-specific code.

The UI allows a user to easily specify system requirements without having knowledge of AADL. It consists of a series of windows that allows adding and manipulating components.

Smartphone Middleware is a set of APIs for Android phones to facilitate stable and extendable code; allowing *Health-Dev* to generate code for a multitude of platforms. It targets sensor data management and visualization. The middleware uses subscribe-publish to connect data from the network to Android applications (apps).

In addition, we have developed a Download Plug-in module that takes the generated code from *Health-Dev* and installs into the specific devices.

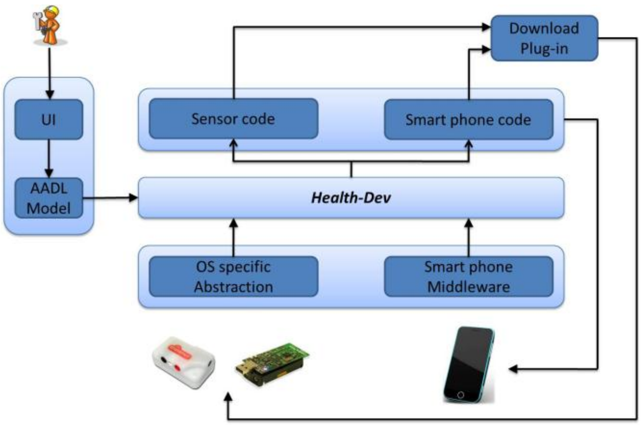

Figure 1. System model of WHS design tool

3. DEMONSTRATION

3.1 Use Case

We are demonstrating the use of *Health-Dev* by generating code for sensor and base station which can be used by separate application, *PETPeeves*.

PETPeeves is an Android mobile application developed to promote healthy exercising and sleeping habits particularly in children between 6 and 12 years of age. The goal is to promote

Wireless Health'12, October 22–25, 2012, San Diego, CA.
Copyright 2012 ACM 1-58113-000-0/00/0010…$10.00.

exercise and good sleeping practices in a fun way over extended period of time and establish an interactive connection with a virtualized pet. The virtual pet is taken care of by the user by performing healthy regimens of exercise and sleep habits every day.

The pet's main attribute, the mood, is updated based on the calories burned estimated from the user's heart rate during exercise.

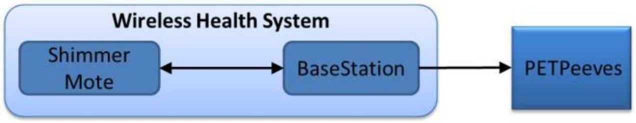

Figure 2. WHS for *PETPeeves*

Using *Health-Dev*, a WHS as shown in Figure 2 is developed which has one SHIMMER mote to sense user's ECG data and computes the heart rate. *Health-Dev* also generates an Android app which acts as a base station to handle data collection used by *PETPeeves*.

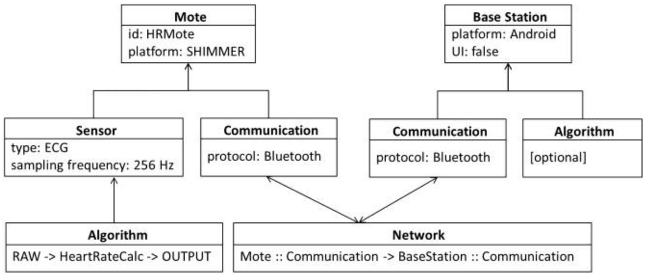

Figure 3. UML representation of WHS model for Use Case

Figure 3 gives a visual representation of the architectural components and connections of the WHS. There exist two main systems with one bridge between them. The first system is the Mote component, which defines an ID and the platform of SHIMMER. The mote consists of one Sensor component and a Communication component. The Communication component defines the communication protocol. The Sensor is defined to sense ECG with sampling frequency of 256 Hz. A Sensor can have an Algorithm component, which defines a sequence of physiological algorithms. In this case, raw data is passed to the "HeartRateCalc" algorithm and HeartRateCalc's output is sent to the Communication component.

The BaseStation component defines type of device used and UI design. The UI settings allow data to be either graphed in the Health-Dev app or instead buffer the data until an external app such as *PETPeeves* requests the data. An optional Algorithm component may be specified to process data before being displayed or sent to an external app. The BaseStation also contains a Communication component where Bluetooth is defined as the protocol. Lastly, a Network is defined to bridge the Communication components of the Mote and the BaseStation, allowing the two systems to communicate. This model acts as input to *Health-Dev, which* uses the *Smartphone Middleware* and a platform-specific code repository to generate sensor and phone code.

3.2 Demonstration Setup

We demonstrate the ability of model-based WHS tool to generate multi-platform code. *Health-Dev* also introduces energy optimizations in radio communication of motes by supporting various energy management techniques such as radio always on, always off, normal and hard duty cycling, and dynamic power management. User can easily integrate these techniques in the code by specifying required mote and energy management properties in AADL or selecting it from drop down in user interface of the tool.

The three applications to be demoed showcase the energy management strategies seamlessly integrated into *Health-Dev*, realistic health applications and multi-platform code generation. The first application demonstrates dynamic power management on a TelosB mote with ZigBee protocol. The second demonstrates the use case detailed in the paper, and lastly an Accelerometer sensing application for SHIMMER mote. The three applications are set up as seen in Figure 4, where a PC with *Health-Dev* generates code for the two Android applications and the TelosB application.

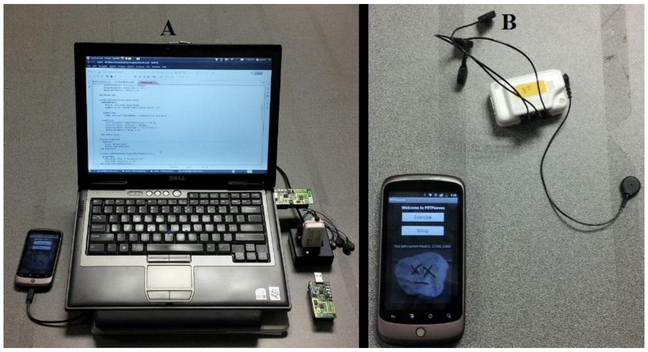

Figure 4. Screenshot of demonstration setup. A) Generating and downloading code using WHS design tool. B) *PETPeeves* application running using generated code.

4. CONCLUSION AND FUTURE WORK

In this work, we demonstrated a model-based WHS design tool that automatically generates sensor and smart phone code for a given high-level WHS specification. The proposed tool helps in rapid prototyping of WHS models. In future, we plan to add a more diverse set of wireless sensor platforms and expand *Smartphone Middleware* to the Windows platform.

5. ACKNOWLEDGMENTS

The authors thank Google for providing Nexus One smart phones. The research is funded in part by NSF grant CNS-0831544 and IIS-1116385.

REFERENCES

[1] Banerjee A., Verma S., Bagade P. and Gupta S.K.S. 2012. Health-Dev: Model Based development of Pervasive Health Monitoring Systems. In *Proceedings of the Ninth Annual International Conference on Wearable and Implantable Body Sensor Networks* (London, United Kingdom, May 9-12, 2012). BSN 2012. IEEE, Los Alamitos, CA, 45-50. DOI= http://ieeexplore.ieee.org/10.1109/BSN.2012.33.

[2] Venkatasubramanian K., Deng G., Mukherjee T., Quintero J., Annamalai V., and Gupta S.K.S. 2005. Ayushman: A Wireless Sensor Network Based Health Monitoring Infrastructure and Testbed. In *Proceedings of the First IEEE International Conference on Distributed Computing in Sensor Systems* (Marina del Rey, CA, USA, June 30 – July 1, 2005). DCOSS 2005. Springer, Berlin, Germany, 406-407. DOI= http://www.springerlink.com/10.1007/11502593_39