Towards Reconfigurable Embedded Medical Systems

Tammara Massey†, Foad Dabiri†, Roozbeh Jafari‡, Hyduke Noshadi†, Philip Brisk†, William Kaiser†, Majid Sarrafzadeh†
†University of California, Los Angeles, Los Angeles, CA 90032
‡University of Texas, Dallas, Dallas, TX 75083
{tmassey, dabiri, hyduke, philip, majid}@cs.ucla.edu, rjafari@utdallas.edu, kaisar@ee.ucla.edu

Abstract

Traditional embedded systems require flexibility and reconfigurable for effective operation. Devices used in medical systems that are life critical must be able to take the necessary action in response to an anomaly, such as the patient undergoing a heart attack or a stroke. Overly complicated operational requirements will be a clear barrier to their adoption, unless sound system-level design techniques can guarantee that these requirements will be met. The key to meeting these requirements is automated adaptability, meaning that the system must be able to reconfigure itself in response the environment. Unfortunately, dynamic reconfiguration itself is non-negligible in terms of both performance and power consumption. We present several reconfigurable design techniques for light-weight medical systems that demonstrate how one can make efficient use of limited resources and balance the tradeoffs between power consumption, memory consumption, and interoperability in an heterogeneous environments.

1. Introduction

Moore's Law has allowed processor performance to double approximately every 18 months due to continuous breakthroughs in transistor technology. Although processor performance is paramount to high-performance computing, the world of embedded systems has other priorities: namely the minimization of power and silicon area. Moreover, embedded systems often communicate via wireless networks with limited bandwidth and have limited communication and memory resources. The shrewd design of embedded systems, however, can make efficient use of limited resources through efficient and effective reconfiguration schemes that balance the tradeoffs between power consumption, memory consumption, and interoperability in heterogeneous environments.

Several of these projects are discussed here in the context of a reconfigurable fabric—literally, a wearable motherboard—as well as several customizable medical devices. Adaptive algorithms for communication throttling in response to dynamic environmental changes are also described. Lastly, we highlight the need to reprogram embedded systems following an extended period of time already employed in their respective environments. The architectures and algorithms presented demonstrate how well designed embedded systems can benefit from reconfigurability.

The physical limits alluded to in Moore's Law are encroaching quickly, and embedded system designers must respond with new ways to pack an increasing amount of processing technology and data onto devices. Our approach to these challenges is to focus on system-level design, emphasizing reconfigurable as a key aspect. Traditional embedded systems require flexibility and reconfigurable for effective operation. The key to meeting challenges in reconfigurability is ***automated adaptability***, where the system reconfigures itself in response the environment. Two levels of reconfigurability are explored in this paper.

- *Hardware reconfigurability* - the inclusion of new sensors to a system or the reconfiguration of the hardware resources upon external stimuli.
- *Software reconfigurability* – algorithms that adapt to the environment to make the best possible use of available resources. Reprogrammability allows the system to be updated via a wireless connection once it has been deployed in the field.

2. Reconfigurable Hardware

The *Plug-and-Play (PnP)* paradigm allows the addition of new peripheral without manual installation of additional software. Although it was originally proposed for desktop operating systems, PnP in the realm of embedded systems can also define methods for lightweight systems to automatically adapt to the addition (or removal) of sensors during runtime. Below three

0-7695-3081-8/07 $25.00 © 2007 IEEE
DOI 10.1109/HCMDSS-MDPnP.2007.7

hardware architectures that adapt and reconfigure themselves on lightweight embedded systems are discussed.

- The *Medical Motherboard* is inherently reconfigurable because it automatically detects sensors as they are added and a lightweight software layer allows the system to adapt to the new sensors.
- *CustoMed (Customizable Medical Devices)* is a novel, fully-wearable architecture that reduces the customization and reconfiguration time for light-weight embedded systems.
- *RFAB (Reconfigurable Fabric)* employs a multi-hop packet routing protocol and software-controlled electrical signal routers on electronic textiles that allow reconfiguration after the cloth have been damaged.

The term "motherboard" used in computer engineering reflects the idea of a platform that usually contains circuitry for at least one processor, interfaces to "plug in" peripherals such as a monitor and keyboard and slots for accepting additional devices, such as network cards, sound cards, and video cards. Our current research focuses on the development of a *Medical Motherboard*, a platform that connects sensors to a collection of relatively small, low-powered processing units *(*motes*)*, and is intended for use in medical embedded systems. The medical motherboard is inherently reconfigurable because it automatically detects sensors as they are added and a lightweight software layer allows the system to adapt to the new sensors. By fixing the number and communication structure between motes, the system remains flexible with respect to the inclusion of new sensors, but effectively minimizes the time required to reconfigure the system. At the same time, the system architecture is designed to tolerate the failure of one (or more) motes and/or potential loss of communication links between the motes. The *Medical Motherboard* also demonstrates how the wise use of reconfigurability in small embedded systems is an approach that is beneficial in heterogeneous medical systems.

In *CustoMed,* the most important component of is Med Nodes, stand-alone components consisting of a processing unit, external sensor boards and a battery. Med Nodes support various types of sensors for physiological reading from the human body, including pressure sensors, piezoelectric sensors, accelerometers, and flex sensors. As such, they are effective building blocks that can be assembled to create a flexible system whose fixed basic structure allows quick reconfiguration. Moreover, the system is scalable and can easily support quick reconfiguration of a large number of Med Nodes [3]. Researchers have prototyped and evaluated a wireless, *in vivo* MEMS sensor for continuous measurement of pressure in the upper urinary track [1]. Wearable Med Nodes also quantitatively gauge the range and degree of motion in stroke patients during rehabilitation for physical therapy [5].

Rips and tears in clothing will sever communication links between motes and sensors and could possibly isolate certain components from the rest of the system depending on the severity of the tear. In *RFab*, reconfigurability is primarily used for robustness and fault tolerance rather than to increase performance. The reconfiguration capability allows *RFab* to rebuild interconnection pathways in response to a tear, thereby allowing the system to continue to operate. Moreover, there is also redundancy in the sensing, and control units embedded on the fabric. Therefore, if one unit becomes completely isolated, than a redundant clone can perform the same task in a manner that is wholly transparent to the user. *RFab* employs an interconnect topology that is well-suited to specific tasks and effectively balances the computational load between motes. Specifically, the topology consists of wires organized into a mesh network. Redundancy was exploited in two approaches: passive wiring and reconfigurable wiring [2].

3. Reconfigurable Software

Reconfigurable software in heterogeneous embedded systems is essential for both usability and system performance. Software that can adapt to the environment in which it is deployed can wisely optimize the use of its resources. Optimization of resources is particularly important when it comes to issues related to power consumption and maximizing battery lifetime. Below, three algorithms that recognize new sensors in its environment or dynamically adapt a system's behavior to outside stimuli are described.

- *Dynamic sensor detection* provides software support for the addition of a new sensor without requiring installation of additional software in the context of PnP.
- *Adaptive electrocardiogram (ECG) analysis* is a distributed ECG that meets latency requirements while minimizing overall energy dissipation for the system
- A *Dynamic Security System* switches between four different levels of security either statically or dynamically depending on timing and/or power constraints imposed by the system.

In *dynamic sensor detection*, the Med Node used a simple feature extraction algorithm to dynamically detect the type of sensor by determining the unique features of pressure and flex sensors under differing circumstances. Feature extraction is a useful algorithm on embedded systems since extraction can accurately describe a large set of data with a small number of variables. In the algorithm, a malfunctioning sensor could be detected based on whether data remained at a static number for long periods of time. Error checking was performed at regular intervals to determine static sensor readings. If the same value was repeatedly sent for more than 100 consecutive intervals, then a malfunctioning sensor was detected.

The electrocardiogram (ECG) is the record of variation of bioelectric potential with respect to time as the human heart beats. Implementation of ECG analysis algorithms become difficult on lightweight, mobile, embedded systems that must meet latency requirements while minimizing overall energy dissipation for the system [4]. *Adaptive ECG analysis* consists of preprocessing, pattern recognition, and classification. Reconfiguration can exploit new pattern recognition or feature extraction modules. A component-based software reconfiguration will enable systems to overcome runtime issues associated with competition for limited resources by parallel tasks running on the same platform. In the distributed algorithm, each detected feature was used in classification. The algorithm sought to have all simultaneously triggered features on the same processing unit to reduce transmission time between the distributed processors. This adaptable partitioning algorithm reconfigures how the features are distributed among the processing units to minimize communication. Based on various configurations (unique to each patient), modules can be added, modified or removed to reconfigure the system.

The *Dynamic Security System* (DYNASEC) is a software architecture to meet the security and power utilization needs of the next generation of networked embedded system [6]. The architecture allows an embedded system sufficient flexibility to adapt to changing system requirements and to reprogram an embedded device with new applications using the wireless network. DYNASEC is a software architecture that allows a centralized node to program other nodes with different levels of security. In particular, when the environment imposes stringent timing constraints on the system (for example, a patient suffers a heart attack and requires near-immediate drug delivery), then the strength of the cryptographic algorithm can be lowered to increase throughput. Adaptable security is necessary when the cost of security processing impedes the ability of the system to meet real-time constraints. A lightweight security protocol dynamically maximizes security settings to meet timing and/or power constraints. This optimal security protocol contains integrity to verify that the message has not changed in transit authentication information so that the receiver can confirm the identity of the sender. Encryption then ensures that only authorized nodes can read the information.

4. Conclusion

The architectures and algorithms presented demonstrate how well designed embedded systems benefit from reconfigurability. Limited resources can be optimized through efficient and effective reconfiguration schemes that balance the tradeoffs between power consumption, memory consumption, and interoperability in heterogeneous environments. Reconfigurable hardware and software have the desirable feature of allowing design flaws to be fixed after deployment, which is an essential feature in embedded medical systems.

5. References

[1]Lin, C., Jea, D. Dabiri, F. Massey, T., Tan, R., Sarrafzadeh, M., Srivastava, M., Schulam, P., Schmidt, J. and Montemagno, C. "The Development of an In-vivo Active Pressure Monitoring System." 4th International Workshop on Body Sensor Networks (BSN 2007). Aachen, Germany. Mar 2007.

[2] Jafari, R., Dabiri, F., Brisk, P., and Sarrafzadeh, M. Adaptive and Fault Tolerant Medical Vest for Life Critical Medical Monitoring". 20th ACM Symposium on Applied Computing (SAC 2005), Mar 2005, Santa Fe, NM.

[3] Jafari, R., Dabiri, F., Sarrafzadeh, M. "CustoMed: A Power Optimized Customizable and Mobile Medical Monitoring and Analysis System", ACM HCI Challenges in Health Assessment Workshop in conjunction with CHI 2005, Apr 2005, Portland, OR.

[4] Jafari, R., Noshadi, H., Ghiasi, S., Sarrafzadeh, M. "Adaptive Electrocardiogram Feature Extraction on Distributed Embedded Systems", IEEE Trans. Parallel and Distributed Systems (TPDS), vol. 17, no. 8, pp 1-11, Aug 2006.

[5] Jafari, R., Jindrich, D., Edgerton, V., Sarrafzadeh, M. CMAS: Clinical Movement Assessment System for Neuromotor Disorders, IEEE Biomedical Circuits and Systems Conference (BioCAS), Nov. 2006, London, UK.

[6] Massey, T., Brisk, P., Dabiri, F., Sarrafzadeh, M. "Delay Aware, Reconfigurable Security for Embedded Systems." Second International Conference on Body Area Networks (BodyNets 2007). Florence, Italy. June 2007.