

Modeling and simulation of Wireless Sensor Network (WSN) with SpecC and SystemC

M. Rafiee, M. B. Ghaznavi-Ghouschi, S. Kheirh, B. Seyfe

Electrical engineering dept.

Shahed university

Tehran, IRAN

{rafiee, Ghaznavi, kheiri, seyfe}@shahed.ac.ir

Abstract— As the complexity of systems is rapidly growing, the designers are moving towards use System Level Design Languages (SLDL) such as SystemC and SpecC. In this paper we use SystemC and SpecC languages for modeling and simulation of Wireless Sensor Networks (WSN). First we start on simple ECO model and add modular sections as increases as the complexity. In the next step, we add data encryption using DES algorithm for security reasons. The future steps include modeling the idle/active/work phases for nodes and we add Attenuation-Transmission-power checking model for estimation power dissipation among nodes. We also include IEEE 802.15.4 protocol module and buffer insertion module. We have simulated the proposed framework with simulation of 3, 7 and 11 nodes in 3000 ms. The fidelity of simulations with SpecC and SytemC meet with 9.8% error and RMSD 4%. Our preliminary results from deployment of the functional-proven SystemC models into SystemC synthesizers, urges that it may be directly used in early stages of WSN nodes synthesis.

Keywords-component; *Wireless Sensor Network, System modeling, SystemC and SpecC languages*

I. INTRODUCTION

Networked sensor systems are seen by observers as an important technology that will experience major deployment in next few years for a plethora of applications. Typical applications include, but are not limited to, data collection, monitoring, surveillance, and medical telemetry [1]–[2]. Wireless Sensor Network consists of large number of sensor nodes. These tiny nodes consist of sensing, data processing, and communicating components. Sensor Network (SN) aims to provide an efficient and effective connection between the physical and computational worlds. In addition to algorithms, hardware and software architecture will significantly impact the effectiveness of SN. Furthermore, SN design methodologies will have an important impact on the cost and performance of SN. SN modeling and simulation is too important. In SN, the environment should be monitored entirely to obtain acceptable results. Different approaches have used for simulation and modeling of SN and WSN [4]. Traditional approaches consist of various simulation tools based on different languages such as C, C++, Java and partly matlab [4]. In the traditional techniques the main focus is based on analyzing the performance of

wired and wireless sensor networks with analytical methods, computer simulation, and physical measurement. However, many constraints imposed on sensor networks, such as energy limitation, decentralized collaboration, and fault tolerance necessitate the use of complex algorithms for sensor networks. It appears that simulation is currently the primary feasible approach to the quantitative analysis of sensor networks [4]. On the other hand there is a gap between simulation and synthesis in the traditional approaches. Therefore the tasks done in the system specification, performance evaluation are not used directly during the synthesis. A New approach using SpecC is reported in [5]. Some new tries started to use SystemC-AMS [15]. The two main reasons for focus on SystemC are: One is the existence of very suitable facilities for simulation of concurrent and parallel phenomena's-where is strongly observed in massive wireless sensor networks-in SystemC, and the second is the near-to-hardware nature in SystemC specifications which could lead to rapid prototyping and less time to market in designs.

In this paper we explain available WSN simulation techniques in section I. Traditional Approach on WSN Simulation System modeling is explained in section II and system is explained in section III. Modeling of ECO system with SystemC and SpecC languages are covered in sections IV and V. Modeling wireless sensor network and result are described in sections VI, VII and VIII. Finally we conclude the paper in section IX.

II. TRADITIONAL APPROACH ON WSN SIMULATION

There are different kinds of simulation tools specialized for WSN. GloMoSim [14] is a scalable simulation environment for wireless and wired network systems. It is being designed using the parallel discrete-event simulation capability provided by Parsec¹ [14]. Parsec is a C-based simulation language, developed by the Parallel Computing Laboratory at UCLA, for sequential and parallel execution of discrete-event simulation models. It can also be used as a parallel programming language. For developing protocols in GloMoSim, one should have some familiarity with Parsec,

¹ Parallel Simulation Environment for Complex systems

but not as an expert. Most protocol developers write pure C code with some Parsec functions for time management.

OMNeT++ [16] is a discrete event simulation environment. Its primary application area is the simulation of communication networks, but because of its generic and flexible architecture, is successfully used in other areas like the simulation of complex IT systems, queuing networks or hardware architectures as well. OMNeT++ provides component architecture for models. Components (modules) are programmed in C++, and then assembled into larger components and models using a high-level language such NED. Reusability of models comes for free. OMNeT++ has extensive GUI support, and due to its modular architecture, the simulation kernel (and models) can be embedded easily into target applications.

III. SYSTEM MODELING LANGUAGES

System modeling in a large extent is a matter of handling abstract and possibly incomplete information and trying to evaluate different solutions based on the system model [5]. Ease modeling system depends on the semantics and syntax of the used language. At present, there is no complete language available for entire system modeling but there are some System-Level Design Language's (SLDL) being used extensively for following system-level design methodology .

Two major candidates for system modeling are SpecC and SystemC languages. SystemC is now based on IEEE-STD 1666-2005 [9]. SpecC and SystemC share many common features, such as dynamic sensitivity mechanism for dynamic scheduling of execution sequence. SpecC is better suited for the architecture exploration as compared to SystemC in terms of profiling and determination of execution sequence. The architecture refinement step involving allocation, partitioning and mapping is easier in SystemC compared to SpecC. SpecC and SystemC have similar capabilities in terms of support for transaction exploration, except for the determination of channel traffic which is much feasible with SpecC on account of its profiling capability. Both the SpecC and SystemC are equally capable for transaction refinement and the refinement process. SpecC and SystemC have similar capabilities in terms of support for communication exploration and equally capable for communication refinement [6].

The SpecC language was specifically developed for the specification and design of digital embedded systems, including hardware and software portions. Built on the top of the ANSI-C programming language, the SpecC language supports concepts essential for embedded systems design, including behavioral and structural hierarchy, concurrency, communication, synchronization, state transitions, exception handling, and timing. Since SpecC is a true superset of ANSI-C, so every C program is also a SpecC program [7].

SystemC is a C++ based modeling platform supporting design abstractions at the register-transfer, behavioral, and system levels. The SystemC classes add the necessary constructs to C++ for modeling systems and hardware at various levels of abstraction-from the abstract untimed models to cycle-accurate RTL models. The power of

SystemC is that it can be used as a common language by system designers, software engineers, and hardware designers. The language is an attempt at standardization of a C/C++ design methodology, and is supported by the Open SystemC Initiative (OSCI) [8]. IEEE-STD-1666-2005 is SystemC Language Reference Manual Standard. This standard provides a precise and complete definition of the SystemC class library [9].

Beside other features, SystemC models a system with logic threads running in parallel. However, its simulator does not take advantage of the parallelism. It also recently is used during direct model synthesis from system specification [18-20]. The above mentioned reasons and features are our motivation in choosing SystemC and SpecC for modeling and simulation. Our future experiments show that the synthesis is also very uniform using this approach.

IV. WSN MODELING

A sensor network is modeled as a set of heterogeneous entities. Sensor nodes deployed over the area of interest. They are triggered by a certain set of stimuli that eventually result in a sensor report that are transmitted to a remote base-station. Three main types of sensor nodes need to be created and supported: 1) target nodes that do stimulation of the sensors, 2) sensor nodes that monitor events and 3) user nodes that query the sensors and are the final destination of the target reports. We primarily used the approach proposed in [5] then we entirely start from scratch and write down a modular framework for simulation capable to include various protocols, topology, security and more features.

A. Node

Node performs a simple task: take an analog sample, and transmit the data over the wireless link to one or more receivers. Node module functionally consists of three concurrent modules, which are analogous to the three main components of the node, namely, Analog-To-Digital (ADC) Converter, Microcontroller, and Sender.

The ADC converts the analog signal into digital values. Sender transmits the data through the wireless channel only after forming a packet as shown in Fig. 1. Microcontroller controls the whole process by setting an interface between the two other sections.

In the transmitted packet, preamble marks the beginning of the package, ADDR is the address of the receiving node, PAYLOAD is the data and CRC is the cyclic redundancy check. ADDR is added by the microcontroller and preamble & CRC are added by the transceiver. All the nodes perform the same function but at different frequency, data resolutions and data domains. ADC converts the analog value into 10-bit digitals. Module μC takes the data from the ADC, adds the address of the receiver node and provides the data payload for the sender module. Sender is analogous to the transceiver; it forms the packet by adding CRC and preamble to the payload and transmits the packet to the Sink.

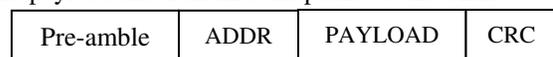


Figure 1. Package Format

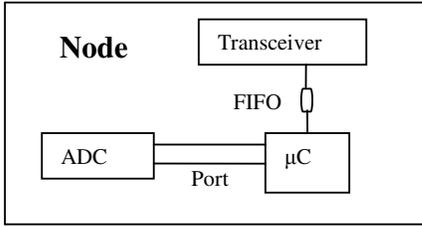


Figure 2. Node module

B. Sink

Sink consists of two components, Receivers, and CPU. Receiver receives data from concurrent node module and sends to CPU Module. CPU first waits for the initial startup commands from the monitor and then waits receiving data from Receiver Module. It checks address of the receiver, if address is right then deletes address from data and starts sending 20-bit digital value to monitor.

C. Monitor and Stimulus

Monitor sends initial startup commands to the sink. Then, the monitor starts waiting for the incoming data from the station and after receiving the data packets it plots the motion data of node. Stimulus basically provides the test bench with the sample input data and performs the function of the sensor.

V. OVERALL SYSTEM

The basic model we used for as primary simulation is based on ECO [5]. This model is depicted in Fig. 4. The test bench consists of ECO System, Stimulus and Monitor. The ECO System consists of two similar Nodes and one Sink module executing concurrently.

VI. COMPARISON ECO SYSTEM MODELING WITH SYSTEMC AND SPECC

Network is simulated with SystemC and SpecC languages. Simulation time for system model is 3000 ms. Only difference between two codes is that because SystemC language is using wait command in the Sender and Receiver modules because SystemC is without handshake channel command.

Fig. 5, shows comparison input and output received in the Monitor modules for the input sine wave from two networks.

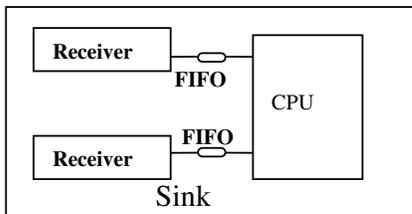


Figure 3. Sink module

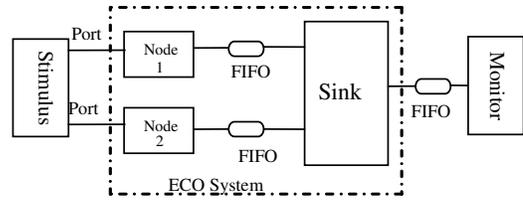


Figure 4. Overall ECO System

The above results show that ECO system is successfully modeled in SpecC and SystemC languages and the simulation results were consistent. Any time varying input signal can be easily recovered and monitored in the Monitor. The maximum absolute value of differences in SpecC and SystemC is about 9.8% and its RMSD² is about 4%.

VII. MODELING WIRELESS SENSOR NETWORK MODELING WITH SYSTEMC AND SPECC

Sensor node has two tasks in wireless sensor network: 1) receiving data from sensor module, 2) receiving data from other sensor node. Therefore for change ECO system to wireless sensor network, we add Receiver process to Sender module and produce transceiver module. In this method received data from sensor transmit via other node(s) to sink. Fig. 6, shows structure node and sink in wireless sensor network.

To increase security in wireless sensor network, we add DES [13] to our design. The algorithm is designed to encipher and decipher blocks of data. Inputs of this block are 64 bits data and 64 bits key. Deciphering must be accomplished by using the same key as for enciphering, but with the schedule of addressing the key bits altered so that the deciphering process is the reverse of the enciphering process.

Therefore, output data from Microcontroller enters to DES Encryption module, before sending to transceiver module, and finally transmitted to the Sink module. In Sink module after receiving data by receiver module, data decoded by decryption and processed by CPU. Fig. 7, shows structure of node and sink in wireless sensor network

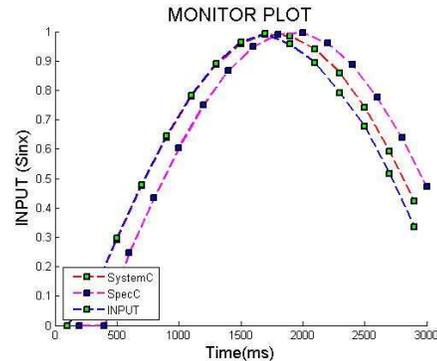


Figure 5. Comparison input and Output in SpecC and SystemC language

² Root Mean Square Deviation

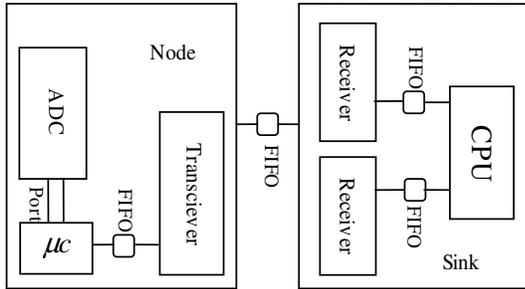


Figure 6. Structure of node and sink in WSN

Before sending data, the signal level in all those nodes is calculated according to the path loss formula $1/d^a$. The attenuation exponent (a) is used in the path loss formula $1/d^a$ where d is the distance between transmitting and receiving nodes. If the signal level in each node is larger than the receiver sensitivity, then it can be detected.

Transceiver can send data to next node or sink by using IEEE 802.15.4 standard [12] (CSMA/CA policy) surrogate use of Preamble and CRC.

IEEE 802.15.4 is a standard used for low rate personal area networks (PAN). It offers device level connectivity in applications with limited power and relaxed throughput requirements. Devices with IEEE 802.15.4 technology can be used in many potential applications, such as home networking, industry/environments monitoring, healthcare equipments, etc, due to its extremely low power features. The IEEE 802.15.4 offers device-level wireless connectivity at low cost. The low cost, here means lower manufacturing cost, lower installation cost and lower maintenance cost. The MAC protocol of 802.15.4 supports both contention-based medium access (Carrier Sense Multiple Access Collision Avoidance (CSMA-CA)) and scheduled-based medium access (Time Division Multiple Access (TDMA)) simultaneously [12].

In CSMA/CA algorithm, ack is sent to destination node and if support ack by destination node, source node starts to transmit data to destination node. Adding this algorithm to wireless sensor network increases delay and lost path.

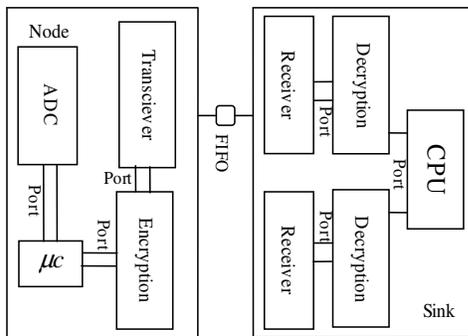


Figure 7. Structure node and sink in WSN

TABLE I. DELAY (MS) DUE MODELING SENSOR NETWORK WITH DIFFERENT TOPOLOGIES WITH SYSTEMC LANGUAGE

Language	SystemC		
Node_No	3	7	11
Topology			
Primary wireless sensor network	200	-	-
Wireless sensor network with DES security	400	700	1000
Wireless sensor network with DES security & Attenuation-Transmission-power	500	800	1100
Wireless sensor network with DES security & Attenuation-Transmission-power & IEEE 802.15.4	700	1200	1600

VIII. RESULT

These codes are simulated with SystemC and SpecC languages. Simulation time for system model is 3000 ms. the entire network is simulated with three, seven and eleven nodes and one sink. The simulation result of the target systems in various topologies are presented in this section.

The primary wireless sensor network is simulated with three nodes.

Table I shows the Delay due four different topologies that using different number of nodes in SystemC language.

Table II shows the Delay due four different topologies that using different number of nodes in SpecC language.

TABLE II. DELAY (MS) DUE MODELING SENSOR NETWORK WITH DIFFERENT TOPOLOGIES WITH SPEC C LANGUAGE

Language	SpecC		
Node_No	3	7	11
Topology			
Primary wireless sensor network	200	-	-
Wireless sensor network with DES security	400	600	800
Wireless sensor network with DES security & Attenuation-Transmission-power	500	700	900
Wireless sensor network with DES security & Attenuation-Transmission-power & IEEE 802.15.4	600	900	1200

IX. CONCLUSIONS

In this paper we have presented a framework for modeling and simulation of Wireless Sensor Networks with SystemC and SpecC language. SystemC and SpecC can be used for system modeling but they include limitation for system modeling. Simulation of the system-level scenario shows that the total elapsed time strongly depends on the size of the WSN. In the proposed approach the following features are implemented and tested successfully: *Simple ECO, Data encryption with DES algorithm, Attenuation-Transmission-power in sending modes of nodes, IEEE 802.15.4 protocol.* There are very wide range of other ideas may be stacked on the top of this framework including, *node unique Identification, path registration, broadcast (multicast).* In total, accurate modeling of SpecC is better than SystemC.

REFERENCES

- [1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey", *Computer Networks*, 38(4), pp. 393-422, March 2002.
- [2] D. Estrin, L. Girod, G. Pottie, and M. Srivastava, "Incrementing the World with Wireless Sensor Networks", *IEEE ICASSP 2001*, 4, pp. 2033-2036, 2001.
- [3] J. Feng, F. Koushanfar, M. Potkonjak. "Sensor Network Architecture", Book chapter, in: 'Handbook of Sensor Networks', I. Mahgoub and M. Ilyas (eds.), CRC press, Section III, no. 12, 2004.
- [4] G. Chen, J. Branch, M. Pflug, L. Zhu, and B. Szymanski, "SENSE: A Wireless Sensor Network Simulator", *Advances in Pervasive Computing and Networking*, Springer, New York, pp. 249-267, 2004.
- [5] G. Sachdeva, R. Doemer, and P. Chou, "System Modeling: A Case Study on A Wireless Sensor Network", Technical Report CECS-TR-05-12, University of California, June 15, 2005.
- [6] L. Cai, S. Verma and D. D. Gajski, "Comparison of SpecC and SystemC Languages for System Design", Technical Report CECS-03-11, University of California, Irvine, May 15, 2003.
- [7] D. D. Gajski, J. Zhu, R. Dömer, A. Gerstlauer, and S. Zhao, "SpecC: Specification Language and Design Methodology", Kluwer Academic Publishers, 2000.
- [8] P. R. Panda, "SystemC a modeling platform supporting multiple design abstractions," in *International Symposium on System Synthesis*, pp. 75-80, Sept. 2001.
- [9] "IEEE STD 1666 - 2005 IEEE Standard SystemC Language Reference Manual", IEEE, pp. 1- 423, 2006.
- [10] M. S. Obaidat, D. B. Green, "On Modeling Networks of Wireless Microsensors", Kluwer Academic Publishers, pp. 115 - 153, 2003.
- [11] H. Karl, A. Willig, "Protocols and Architectures for Wireless Sensor Networks", John Wiley & Sons Ltd, 2005.
- [12] L. Cheng, "IEEE 802.15.4 MAC Protocol Study and Improvement", A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy, Georgia State University, 2000.
- [13] R. G. Kammer, "DATA ENCRYPTION STANDARD (DES)", Federal Information Processing Standards Publication, 1999 October 25.
- [14] L. Bajaj, M. Takai, R. Ahuja, K. Tang, R. Bagrodia, M. Gerla, "GloMoSim: A Scalable Network Simulation Environment", University Technical Report 990027, UCLA Computer Science Department, Los Angeles, USA, 1999.
- [15] M. Vasilevski, F. Pecheux, H. Aboushady, L. de Lamarre, "Modeling heterogeneous systems using SystemC-AMS case study: A Wireless Sensor Network Node", *IEEE International Behavioral Modeling and Simulation Conference, BMAS'07*, San Jose, CA, USA, pp. 11-16 ,20-21 Sept. 2007.
- [16] S. M. Bruschi, R. H. C. Santana, M. J. Santana, T. S. Aiza, "An automatic distributed simulation environment", *Proceedings of the 36th conference on Winter simulation*, Volume 1, pp. 378 - 385, 5-8 Dec. 2004.
- [17] IEEE, IEEE 802.15.4 Standard Specification for Low-Rate Wireless Personal Area Networks (LR-WPANs), May 2003.
- [18] "Agility Compiler manual", For Agility 1.2, Available: <http://www.celoxica.com>.
- [19] "SystemCrafter SC User Manual", Version 3.0.0, Available: <http://www.SystemCrafter.com>.
- [20] "Getting Started with CoCentric SystemC Studio", Version 2002.05-SP1, synopsys, U.S.A, September 2002.