Scalable ZigBee-Based Smart Authentication and Access Control System Design Using XMOS Programmable Chips

Wael Hosny Fouad Aly, Haytham Aboulabbas M., Moustafa H. Aly, Hossam Eldin Moustafa

Abstract: In this paper, an efficient, inexpensive, scalable, and ZigBee-based smart authentication and access control system is proposed. The system consists of a central node and remote nodes. The central node holds a database of authorized users and it is mesh-networked to a set of remote nodes which are spread throughout the premises of an enterprise. For each remote node, a radio frequency identification (RFID) reader is mounted which is used to communicate with the RFID tags. User identification is performed by reading the RFID tag number and authentication is done by means of a password entered by the user through a keypad that resides on the remote node. This data is then transmitted through the wireless 802.15.4 interface to the central node for verification. An event log file with date/time stamp is created for each user that is granted access to the system. It maintains footprints of all users' movement activities within the premises and it is stored on an SD card mounted to the central node. Moreover, the central node can be connected to a computer via the serial port, where the real-time event logs will be visualized on the screen along with the ability of automated parsing and storing the received information directly to Excel without the need of any additional programming requirements. This paper is based on a prototype development of the proposed system. It presents both hardware and firmware design aspects.

Index terms: remote node, Radio Frequency Identification (RFID), embedded system, bit banging, graphical user interface (GUI).

1 INTRODUCTION

The term access control is a very general concept is defined as: limiting access to information system resources only to authorized users, programs, processes, or other systems [1]. It is used in different areas like physical access, computer security, and telecommunications [2]. Physical access is the focus of this paper. It is about restricting the entrance to a property, a building, or a room to authorized persons. Several means are available to achieve this, e.g., a guard, a receptionist, locks, keys, or an electronic system of which the last one is the focus of the paper. The electronic system uses data carriers

There are two different electronic access control systems: online and offline systems. Both of them check if a person is authorized or not [3]. An online system, which is the case of the proposed system, is used when there are many people and few entrances in the enterprise. By means of a central computer (or node), a network and a database, it is possible to load data into the remote terminals (or nodes) situated beside the entrances [3]. One advantage of online systems is that it could easily protect areas in the enterprise that has special security requirements. Besides, it is crucial for those who need to keep track of their employees within the premises of the enterprise. Every terminal (or node) has a list of authorized keys which are compared with the entered key to find a match and decide whether the user is authorized to access or not [3].

The use of RFID for access control is becoming very popular because it can remove the manual aspect of entry involved with keys, keypads, and magnetic stripe cards while increasing security due to unique identification. Using RFID for entrance into a building will not only increase convenience without slowing down the organization's workflow, but also allows tracking of who has entered the premises at a given time. This will increase building security and thus the safety of the occupants.

As extensive as the possibilities of the RFID technology are, they are limited by the anticipated difficulties of implementation. For typical applications, such as deployment in a large-scale enterprise, multiple RFID readers are densely distributed throughout the premises with a single central node to control the network and manage the vast amount of data collected. Because of the number of readers transmitting their data to the central node through hardwired connections, installing a complete wired infrastructure is necessary to accommodate the data transmission. This increases the installation cost and the hassle of implementation, detracting from the appeal of RFID. To improve this situation, the major ambition of the proposed system is to enable a wireless communication between the central node and the distributed remote nodes. Having wireless remote nodes, which enhance security by

limiting access to restricted areas and help in tracking employee activities, would lower installation cost and increase the flexibility of the system.

The first step in allowing the central node and remote nodes to communicate wirelessly is to choose the protocol to be used. The relatively new ZigBee standard [4], specified by IEEE 802.15.4, is fitting because it is designed for low cost and low power applications in particular. Choosing a standard-based technology over a proprietary solution is beneficial because it offers more flexibility and universal functionality. Of the available wireless standards, ZigBee was determined to be the appropriate solution because Wi-Fi and bluetooth are more expensive and had higher power consumption due to the bandwidth and system resources offered.

The next critical decision in the design process is how to implement the ZigBee standard, which saves the valuable implementation time. XBee-PRO modules from Digi International were chosen to enable the wireless communication between the central node and the spread remote nodes. These modules allow for the creation of complex mesh networks based on the XBee ZigBee mesh firmware. Point-to-point and multi-point networks are supported using these modules.

In the proposed system, each node employs XS1-G4 programmable chip from XMOS as its main controller which is responsible for interfacing and interconnection of the modules that reside in the node. Both central and remote nodes are built using a modular philosophy. This not only simplifies repairs and servicing of the system and shortens downtime, but it also provides customers with an exceptionally wide range of choices that enables the system to be tailored to the needs of individual customers. It also represents an important factor for further development of the system. Each module that was used in the system architecture can be further refined independently of other modules. As a result, improvements can be introduced continuously as soon as they have been thoroughly tested by little or no hardware changes along with minimum software modifications. This gives customers access to the latest execution.

2 RELATED WORK

In the evolution of RFID security [5], Charles Walton is claimed to be the first one building an access control system based on RFID. A tracking algorithm in RFID reader network [6] is a study from China about tracking persons or objects in a network consisting of RFID readers. The algorithm can track hundreds of transponders at the same time. Authors of [7] use RFID to evaluate the performance of the employees within the company by monitoring their movements in the facilities. With the advent of cheap low power commercial RF modules, fully automated management systems are being implemented using wireless technologies. For example, Ref. [8] presents a novel bus priority control system for the Advanced Public Transportation System (APTS) based on wireless sensor networks and ZigBee. Authors of [9] reported the use of ZigBee RF nodes for data packet transmission in an intra-car wireless environment.

3 XMOS TECHNOLOGY REVOLUTION

Designers for electronic products are challenged by requests for customized and differentiated complex products in short time frames. In this environment two factors become crucial: flexibility and simplicity. For many years Application Specific Integrated Circuits (ASICs) or Field-Programmable Gate Arrays (FPGAs) allowed engineers to meet very specific product design requirements. However, the design process involved is complex, time-consuming, and expensive. Technologies such as microcontrollers or Digital Signal Processors (DSPs) provide simplicity, but they lack the flexibility needed to meet rapidly changing market requirements.

XMOS, a fabless semiconductor company that was founded in 2005 by experienced semiconductor executives, has delivered a technology that provides the flexibility of FPGAs or ASICs with the design simplicity of processor-based designs [10]. XMOS has chosen an event-driven, multithreaded processor. The processor is a 32bit Xcore that has a fixed instruction set developed for fast real-time response and low silicon cost [10].

XMOS expects that not only its chips would be successful in areas where FPGAs have found success due to their configurability, such as communications base stations and switching fabrics, but also that XMOS silicon would get design wins in consumer electronics goods and even in such high volume devices, such as mobile phone handsets. XMOS describes its approach as "software-defined silicon" (SDS) and reckons it will provide consumer electronics

system designers with the unit cost advantage of the System-on-Chip (SoC) - but without the development time and cost - combined with the flexibility of the FPGA - but with far better silicon efficiency [11]. Figure 1 shows design challenges achieved using XMOS technology compared to other digital design platforms.

Following the previously mentioned merits of this newly emerging 32bit event-driven multithreaded processor, combined with its low cost and potential scalability, it is selected to be the main controller of the proposed system. The architecture of the chip, that combines a number of XCore processors, each with its own memory, on a single chip along with the direct support for concurrent processing (multi-threading) and 32 channel ends per XCore for scalable communication with other threads, on- or off-chip, serves the modularity of the system. This means that it is extremely simple to add, change, or even remove modules of the system to fit differentiated customer needs with each module of additional

Figure 1: Embedded system design using XMOS compared to other digital design platforms. Not surprisingly, the XMOS architecture has similarities to the transputer [11].

functionality is implemented as an independent module. Consequently, the system provides a high level of flexibility to anticipate future amendments and upgrades with almost little or no hardware changes.

4 BIT BANGING

Bit banging is a technique for serial communications using software instead of dedicated hardware. Software directly sets and samples the state of pins on the controller, and is responsible for all parameters of the signal: timing, levels, synchronization, etc. In contrast to bit-banging, dedicated hardware (such as UART, I²C, SPI … etc.) handles these parameters and provides a buffered data interface in other systems, so software is not required to perform signal demodulation. Bit-banging can be implemented at very low cost, and is widely used in embedded systems [12].

Although the high million-instruction-per-second (MIPS) possessed by XS1-G4, it hasn't any built-in connectivity protocols enabling it to be interfaced directly to other devices. However, deploying bit-banging technique for each desired interface protocol allows the same device to use different protocols with minimal or no hardware changes required. As a result, we can mix devices from different protocol families. This not only gives the system high level of flexibility but it also helps the potential scalability of the system to meet differentiated customer demands.

XMOS has developed the XC programming language [13-14]. The language was designed to exploit the XMOS processor architecture. XC extends the programming capabilities of the C language. Along with the sequential capabilities of C, XC provides explicit control of concurrency, communication, timing, and input-output. It supports deterministic concurrent programming [13]. Using XC language along with bit-banging technique, the system is currently utilizing different interfacing protocol libraries that enable the system to be interfaced to a variety of devices. An example on implementing UART interface protocol on XMOS using XC programming language and bit banging technique is explained in the following subsection:

4.1 Implementing UART on XMOS Using XC Programming Language

UART stands for Universal Asynchronous Receiver/ Transmitter. The UART protocol provides a simple way to transmit data over a serial link. Data is sent at a fixed baud rate, requiring no clock signal to be transmitted. The transmit procedure is illustrated in Figure 2. The quiescent state of the link is the high (1). A byte is sent by first asserting a start bit (0), then the data bits and finally the stop bit (1). Each of these values is asserted for an entire bit period.

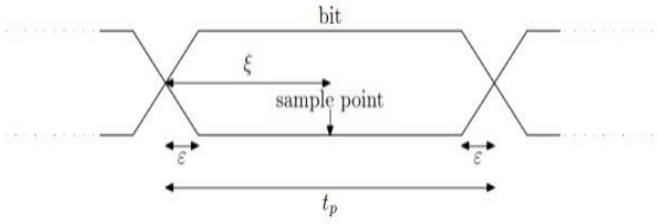

Figure 4: Bit cell with sampling point at offset ξ and transition-width ε.

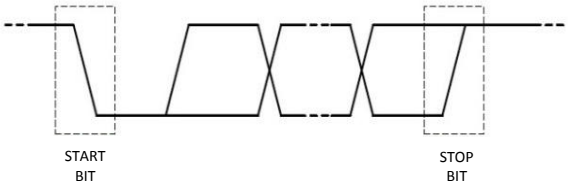

Figure 2: UART transmit procedure.

4.1.1 Sampling the Bits

The data is serialized into a bit-stream that can be seen as a continuous signal over time, s(t). In order to reconstruct the original (discrete) bit-stream it is necessary to sample the continuous signal. Formally, this is expressed by forming the inner product of the signal s(t) with an orthonormal set of functions $\varphi_n(t)$ representing the different bit cells, yielding to the orthogonal series coefficients s_n:

$$s_n = \int s(t)\varphi_n(t)dt \quad (1)$$

In the proposed model, it is assumed that the signal is binary, that is s_n takes one of the two distinct value ranges, θ_{high} or θ_{low}, representing the high or low state logic level respectively. Thus in order to obtain binary values, the resulting coefficients s_n need to be mapped to the bit values d_n according to the rule:

$$d_n = \begin{cases} 0 \; if \; s_n \in \theta_{low} \\ 1 \; if \; s_n \in \theta_{high} \end{cases} \quad (2)$$

In the case that s_n is neither in one of θ_{high} or θ_{low}, the value of d_n is not defined. When dealing with actual hardware, an undefined state results in a non-deterministic assignment of the values 0 or 1 to d_n.

The functions used for $\varphi_n(t)$ are usually pulse-shaped functions. In the case of UARTs, the sampling itself takes far less time compared to the duration of a bit period, therefore pulse-shapes of infinitesimal width as provided by Dirac's delta function will be assumed:

$$\varphi_n = \delta(nt_p + \xi - t) \quad (3)$$

This substitution in Eq. (1) results in:

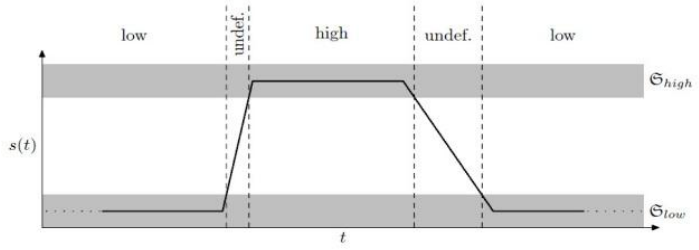

Figure 3: Binary data signal s(t) waveform, demonstrating periods of undefined state, resulting from transitions between the low- and high-state representing areas denoted as θ_{low} and θ_{high} respectively.

$$s_n = s(nt_p + \xi) \quad (4)$$

Thus, assigning s_n the value of the signal $s(t)$ at the time position $t = nt_p + \xi$.

In actual electronic circuits, the state transition between low and high logic state levels takes a certain amount of time, thus causing $s(t)$ (and therefore d_n) to be of undefined state for a certain amount of time as shown in Figure 3. In order to avoid undefined states, the signal is sampled at the midpoint of the interval.

Figure 4 shows a bit cell with a sample point at offset ξ from the start of the bit cell (for which $0 < \xi < t_p$ holds), ε representing the time at both ends of the bit cell at which the signal needs to stabilize (for which $0 < 2\varepsilon < t_p$ holds), and finally t_p being the bit length. In order to reliably sample a bit during its defined-state window, the following condition needs to hold [15]:

$$\varepsilon < \xi < t_p - \varepsilon \quad (5)$$

The following is a sample function that is written in XC language and implements byte transmission by XMOS over a serial link:

```
void txbyte (out port trans, chanend ch)
{
 int byte;
 unsigned time;
 timer t;
 while (1)
 {
  ch :> byte;
  t :> time;
  trans <: 0;
  time += BIT_TIME;
  t when timerafter (time) :> void;
  for (int i=0; i<8; i++)
  {
   trans <: >>byte;
   time += BIT_TIME;
   t when timerafter (time) :> void;
  }
  trans <: 1;
```

```
  time += BIT TIME;
  t when timerafter (time) :> void;
 }
}
```

In the previous function, as soon as the reception of a byte from the main program through the channel end (ch) is, the transmitter outputs the byte by first outputting a start bit, followed by a conditional input on a timer that waits for the bit time to elapse; the data bits and stop bit are output in the same way.

The function below receives a stream of bytes over the serial link:

```
void UART_rx_byte_x0 (chanend ch0)
                    {
                int byte;
              unsigned time;
                 timer t;
                 while (1)
                      {
 UART_rx_x0 when pinseq (0) :> void;
                 t :> time;
      time += UART_BIT_TIME_X0/2;
        for (int i=0; i<8; i++)
                      {
       time += UART_BIT_TIME_X0;
   t when timerafter (time) :> void;
         UART_rx_x0 :> >> byte;
                      }
       time += UART BIT TIME X0;
    t when timerafter (time) :> void;
          UART_rx_x0 :> void;
         byte = (byte >> 24);
             ch0 <: byte;
                      }
}
```

In the previous function, the receiver samples the incoming signal, waiting for a start bit. After receiving this bit, it waits for 1.5 times the bit time and then samples the wire at the midpoint of the first byte transmission, with subsequent bits being sampled at half of the bit time increments.

5 SYSTEM MODEL

This section presents an overview of the system and introduces the functionality of each component individually.

5.1 Overview

Figure 5 depicts the system model. The goal of the system is to provide an online electronic access control system that will automate entry to the premises of an enterprise according to pre-assigned user credibility. It also monitors authorized users who have granted access to the system by keeping track of their movement activities within the premises.

The system consists of a central node and distributed remote nodes. Remote nodes are installed at the main entrance gates of the enterprise as well as at the office doors within the premises and wherever user tracking is required to be maintained by the system.

Once a user carrying the RFID tag is in the vicinity of the reader that reside in the remote node at the main entrance gate, the tag will be detected and its ID number is read by the RFID reader module and is then transferred over the IEEE 802.15.4 wireless link to the central node to verify whether access is granted to enter or not. Furthermore, the user is authenticated by means of a password entered using an alphanumeric keypad that resides in the remote node when prompted. Upon proper authentication by the central node, the programmable chip that resides in the corresponding remote node will trigger an electric door strike that will allow the user to open the main entrance gate. Then, a pre-determined set of premises, according to predefined user credibility, will be available for access for that specific user.

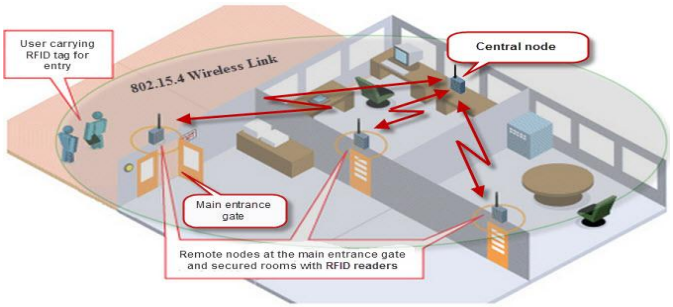

Figure 5: System model.

Once a user is granted access at the main entrance gate, a log file is created and maintained for that specific user on the SD card that resides on the central node. A date/time stamp of all user movement activities within the premises will then be recorded on his specific log file with the ability to restrict existence of specific users in pre-determined specific premises to specified amount of time, after which an audible and visual alarms will be activated at the remote node where he resides.

Moreover, the central node can be connected to a computer via the serial port where the real-time event logs will be

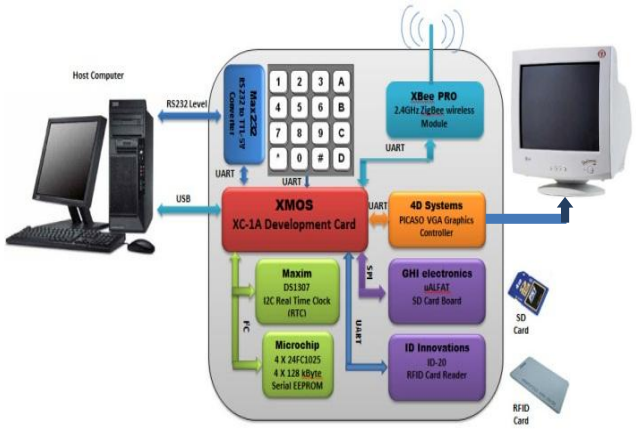

Figure 6: Central node architecture.

visualized on the screen along with the ability of automated parsing and storing the received information directly to Excel sheet without the need of any additional programming requirements.

5.2 Central Node

In the proposed system, a central node is a necessity and it represents the bane of the system. The general responsibilities of the central node are to authenticate users for access and keep track of their movement activities as well as providing commands to the remote nodes and maintain the ZigBee network. The central node continually evolves throughout the design process of the system. Its software is worked on and updated at each phase of the design process. In addition, the central node software, unlike the remote node program, features a graphical user interface (GUI). The software can be upgraded to understand additional commands as desired in later phases of development.

In the proposed system, a central node is a necessity and it represents the bane of the system. The general responsibilities of the central node are to authenticate users for access and keep track of their movement activities as well as providing commands to the remote nodes and maintain the ZigBee network. The central node continually evolves throughout the design process of the system. Its software is worked on and updated at each phase of the design process. In addition, the central node software, unlike the remote node program, features a graphical user interface (GUI). The software can be upgraded to understand additional commands as desired in later phases of development. Consequently, the system provides a high level of flexibility to anticipate future needs and developments. As a result, customers will receive exactly the right access control for their needs with the ability of future upgrade of the system at the same high level performance and minimum cost.

Moreover, the central node provides debugging and communication interfaces. It has the ability to be connected to a personal computing device (laptop or desktop) either through USB interface for further system development and code modification or RS232 interface to visualize the real-time access logs on the screen with the ability to automatically parsing and storing the received information directly to Excel. The hardware architecture of the central node is shown in Figure 6.

5.3 Remote Node

In the proposed system, remote nodes are installed near the main entrance gates of the enterprise, office rooms, and wherever user tracking is required to be maintained. Remote nodes are small simplified version of the central node with respect to hardware architecture and code complexity. A remote node comprises the same 2.4 GHz wireless transceiver and XC-1A development card of the central node. Besides, it contains relay control circuitry that unlocks an electric door strike if a specific user is authenticated by the central node. An alphanumeric keypad is used to enter password for user authentication, and visual and audible alarms that will be triggered in case of system faults, unauthorized user activities within the premises, or suspicion of system hack by intruders. Figure 7 shows the hardware architecture of the remote node.

6 SYSTEM OPERATION

Here, an overview of the ZigBee wireless protocol is presented followed by the communication involved in the system operation.

6.1 ZigBee Protocol Overview

The IEEE 802.15.4 wireless standard provides the Physical layer (PHY) and Medium Access Control layer (MAC) for the wireless communication while the ZigBee protocol,

working on top of it, would perform the Network layer (NWK) and Application layer (APL) tasks. The PHY, MAC and NWK layers would handle how the underlying wireless data transmission would be carried out and how the network of RF transceivers would be organized while the APL layer would handle the tasks associated with each autonomous device.

The ZigBee standard employs a suite of technologies to enable scalable, self-organizing, self-healing networks that can manage various data traffic patterns. ZigBee is a low-cost, low-power, wireless mesh networking standard. The low cost allows the technology to be widely deployed in wireless control and monitoring applications. The low power-usage allows longer life with smaller batteries, and the mesh networking provides high reliability and larger range. ZigBee has been developed to meet the growing demand for capable wireless networking between numerous low power devices. This new level of communication permits finely-tuned remote monitoring and manipulation [16-17].

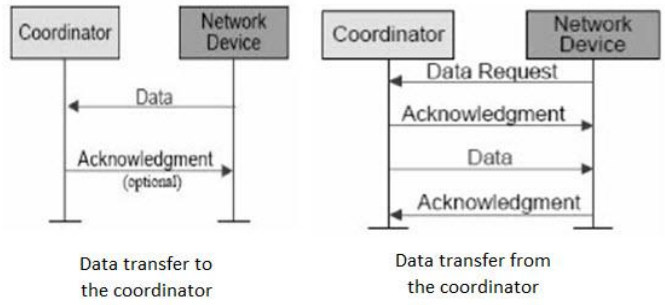

Figure 9: Data transfer on non-beacon network.

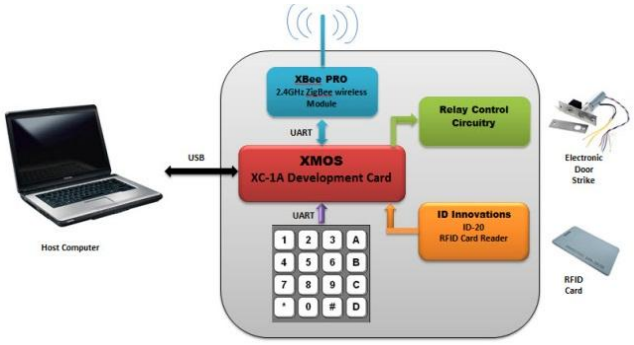

Figure 7: Remote node architecture.

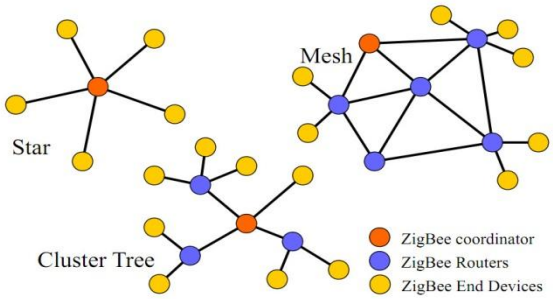

Figure 8: ZigBee topologies

Figure 8 introduces the concept of the ZigBee network topology. Several topologies are supported by ZigBee, including star, mesh, and cluster tree.

6.2 Nodes Communication

In the proposed system, remote nodes take the role of a ZigBee end devices while the central node takes the role of ZigBee coordinator. Different ZigBee devices implement different device profiles are clear under the ZigBee protocol stack to suit the application in which they are being used. The ZigBee alliance has defined several device profiles for typical applications intended for ZigBee devices, such as home and building automation, industrial control, etc. [18]. The specification has also provided flexibility to include custom device profiles to suit customized applications [19]. In the proposed system, customized device profiles have been defined to suit our application.

In the proposed system, central and remote nodes communicate using non-beacon-enabled networks. Figure 9 shows data transfer using non-beacon network. In this type of network, ZigBee module will have its receiver continuously active, requiring a more robust power supply.

However, this allows for heterogeneous network in which some devices receive continuously, while others only transmit when an external stimulus is detected. This ZigBee module may receive constantly, since it is connected to the mains supply, while a probably battery-powered remote node would remain asleep until an RFID tag is detected at the vicinity of the RFID tag reader. The remote node then wakes up, sends both the tag ID and the

password entered by the user to the central node, and waits to receive an acknowledgment from the central node indicating whether the user is authenticated for access or not, and then returns to sleep.

7 CONCLUSIONS

Although the principles of access control and security are simple, the large number of options for each of the component parts, the need to select the right components for the operating conditions and application, the requirement for compatibility between components, and the demand of minimizing cost while meeting differentiated customers' requirements for potential competitiveness on the market, present a real challenge to the system designer.

In this paper, we have presented a novel ZigBee-based smart authentication and access control system. We have provided a detailed description of our system referring to the prototype developed. A custom ZigBee device profile was developed to suit the proposed application and was implemented in the ZigBee protocol stack defined by the ZigBee specification.

Thanks to XMOS technology, yet the system has the potential for more advanced features to be implemented and integrated to it that could meet the ultimate perspective security requirements, access options, and network topology. Moreover, the modular philosophy the system is built up with allows the system to be tailored in order to meet the ever differentiated customer needs in short time frames with efficient use of hardware components and minimized software development budget. As a result, each customer would be guaranteed to get his tailored-made cost-effective system that should be just good enough to cover his current requirements with the ability of future upgrades.

REFERENCES

[1] Committee on National Security Systems: *National Information Assurance (IA) Glossary*, CNSS Instruction No. 4009, http://www.cnss.gov/Assets/pdf/cnssi_4009.pdf, viewed June 2011.

[2] "*Access Control*," in Wikipedia: The Free Encyclopedia, Wikimedia Foundation Inc., http://en.wikipedia.org/wiki/Access_control, viewed June 2011.

[3] Finkenzeller, K.; Waddington, R.; *RFID Handbook: Radio-Frequency Identification Fundamentals and Applications*. John Wiley Sons, January 2003.

[4] *ZigBee Alliance*; an open, non-profit association of members that created the growing family of ZigBee standards, http://www.zigbee.org/Home.aspx, viewed June 2011.

[5] Rieback, M.R.; Crispo, B.; Tanenbaum, A.S.; *The Evolution of RFID Security*, Pervasive Computing, IEEE, Vol.5 (1), Page(s): 62-69, DOI 10.1109/MPRV.2006.17, March 2006.

[6] Wei J.; Dan Y.; Yan M.; *A Tracking Algorithm in RFID Reader Network*, Frontier of Computer Science and Technology, 2006. FCST '06. Japan-China Joint Workshop, Fukushima, ISBN: 0-7695-2721-3, Page(s): 164-171, DOI 10.1109/FCST.2006.7, November 2006.

[7] Iman Morsi, Yasser Elsherief, Amr El Zawawi; *A Security System and Employees Performance Evaluation Using RFID Sensors and Fuzzy Logic*, Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, 2009. COMPUTATIONWORLD '09, Page(s): 597-602, DOI 10.1109/ComputationWorld.2009.112, 2009.

[8] Z. Wu; H. Chu; Y. Pan; X. Yang; *Bus priority control system based on wireless sensor network (WSN) and zigbee*, in Vehicular Electronics and Safety, 2006. ICVES 2006. IEEE International Conference on, Dec. 2006, Page(s): 148–151.

[9] H. M. Tsai, C. Saraydar, T. Talty, M. Ames, A. Macdonald, O. K. Tonguz; *Zigbee-based intra-car wireless sensor network*, in Communications, 2007. ICC '07. IEEE International Conference on, June 2007, Page(s): 3965–3971.

[10] XMOS Ltd, 2010. *XMOS: A Programmable Revolution, A Compelling Alternative to Low Cost FPGAs*, Retrieved May 25, 2010, from https://www.xmos.com/download/public/XM-000162-WP-1.pdf.

[11] Peter Clarke, 2007. *XMOS fields software-defined silicon*, Retrieved (n.d.) from http://www.eetasia.com.

[12] "*Bit banging*," in Wikipedia: The Free Encyclopedia, Wikimedia Foundation Inc., http://en.wikipedia.org/wiki/Bit-banging, viewed June 2011.

[13] XMOS Ltd, "*XC Language*", http://www.xmos.com/technology/xc, viewed June 2011.

[14] Douglas Watt, *Programming XC on XMOS Devices*, XMOS Ltd, 2009, Last viewed May 25, 2011, from www.xmos.com/published/programming-xc-xmos-devices.
[15] Herbert Valerio Riedel, *Universal Asynchronous Receiver/Transmitters: A Software Implementation Approach*, Institute of Computer Aided Automation, Research Group Industrial Software, 2009.
[16] S.S.Riaz Ahamed; *The Role of ZigBee Technology in Future Data Communication System*, Journal of Theoretical and Applied Information Technology, Vol.5, No.2, 2009.
[17] S. D. Dissanayake, P. P. C. R. Karunasekara, D. D. Lakmanaarachchi, A. J. D. Rathnayaka, A. T. L. K. Samarasinghe, *Zigbee Wireless Vehicular Identification and Authentication System*, 4th International Conference on Information and Automation for Sustainability, 2008. ICIAFS 2008. Page(s): 257-260, DOI 10.1109/ICIAFS.2008.4783998, 2008.
[18] A. Wheeler, *Commercial applications of wireless sensor networks using zigbee*, in Communications Magazine, IEEE, vol. 45, no. 4, Toronto, Ont., Canada, Apr. 2007, Page(s): 70–77.
[19] Zigbee Specification, ZigBee Document 053474r06 Version 1.0, Zigbee Alliance Std., Dec. 2004.

Wael Hosny Fouad Aly, Haytham Aboulabbas M., Moustafa H. Aly
College of Engineering and Technology, Arab Academy for Science, Technology & Maritime Transport, Alexandria, Egypt
drwaelaly@gmail.com, haythamaboulabbas@hotmail.com, drmosaly@gmail.com

Hossam Eldin Moustafa
Faculty of Engineering, University of Alexandria, Alexandria, Egypt
ahossam@cs.ucf.edu