ASEE AMERICAN SOCIETY FOR ENGINEERING EDUCATION

Paper ID #40360

Emerging Innovations and Global Challenges on Curriculum Design: Case Study of Teaching Security in Embedded System Design

Dr. Mohammed Ferdjallah, Marshall University

Dr. Mohammed Ferdjallah is an Assistant Professor in the Department of Computer Science & Electrical Engineering at Marshall University. Dr. Mohammed Ferdjallah received his PhD degree in Electrical and Computer and MS degree in Biomedical Engineering from The University of Texas Austin. He also received his MD degree from the International University of the Health Sciences. He has a multidisciplinary expertise in image & signal processing, computational modeling, and statistical data analysis. As an electrical and biomedical engineering scientist, he conducted research in computer modeling of the brain, cranial electrical stimulation (CES), electrical impedance tomography, electrode design, and EMG and muscle action potentials and ions channels simulation & modeling. His technical research interests include digital systems, embedded, systems, computer architecture, adaptive and system identification, modeling and simulation, and signal and image processing. His clinical research interests include impacts of chronic diseases in elderly (such as Alzheimer's disease, cancer, and diabetes), innovative technology for drug addiction treatment and prevention, medical records, comparative outcomes research, and biomedical sciences. He has successfully published several peer-reviewed articles in biomedical sciences, physical medicine and rehabilitation, modeling and simulation of physiological signals, motion analysis, and engineering.

Asad Salem

Emerging Innovations and Global Challenges on Curriculum Design: Case Study of Teaching Security in Embedded System Design

Mohammed Ferdjallah[1], Asad Salem[2]
[1]Department of Computer Science and Electrical Engineering
[2]Department of Mechanical and Industrial Engineering
Marshall University
Huntington, WV 25705
ferdjallah@marshall.edu, asad.salem@marshall.edu

Hadjer Bouchareb
Department of Letters and Foreign Languages
University of Kasdi Merbah
Ouargla, Algeria 30000
hadbouchareb@gmail.com

Abstract

Emerging innovations and global challenges are not only overwhelming human health and the environment but also complicating the training of the next generations of engineers who will shape the future of the global economy. In particular, a topic of immediate concern is the security of embedded systems. Embedded systems are ubiquitous and are found in medical devices, nuclear plants, and chemical processing systems to name a few applications. The ever-increasing complexity of modern embedded systems has raised the potential for security breaches that may compromise the integrity and functions of these products. Although embedded systems design has been a core curriculum in most electrical and computer engineering programs, incorporating security-training skills at the undergraduate level remains challenging. Despite the enormity of security challenges, they are often ignored during the initial design and development period of an embedded system, thus leaving many devices vulnerable. We postulate that this design shortcoming is perhaps due, in part, to the teaching inadequacy of embedded security in early engineering education.

In this paper we propose a framework to design an undergraduate curriculum that will incorporate security concepts in embedded systems courses. The curriculum will create new learning materials and teaching strategies that will focus on security concepts in early phases of embedded systems design. Because it is almost impossible to add security to an existing embedded system, the significance of this paper is to enforce life-learning awareness of security concerns in embedded systems design in early education of undergraduate students. The undergraduate curriculum will integrate security concerns, challenges, and concepts in all aspects of embedded systems design. We propose new learning materials and teaching strategies that will focus on securing the hardware and software programming of the central processing unit and its peripherals at the low-level layer allowing the upper-level layer to implement security standards with a greater efficiency. We will also provide advanced concepts that mitigate security breaches either by low-level encryption solutions or dead-end program destruction. The long-term goal of this paper is to promote security awareness in embedded systems at the undergraduate level.

Key Words: Curriculum Design, Embedded Systems Design, Embedded Security, Global Challenges, Low-Level Applied Cryptology.

I. Introduction

According to Business Communications Company (BCC) Research Group publication, the world market for embedded software and hardware will grow to more than $250 billion by 2025[1]. Internet-based devices accelerated the progression of embedded systems into mobile devices, tablets, and wearables devices. As the needs of the customer are growing with the complexity of the network systems, the smaller systems with 4-bit, 8-bit and 16-bit CPUs are declined in favor of more secured large systems with 32-bit CPUs[2]. The smaller architectures have serious security drawbacks. As more functionalities are embedded in smaller devices, there is an alarming increase in security concerns due to the limited space for the basic security systems[3]. There have been several well-structured attacks on numerous embedded devices, especially those that have access to sensitive company data on many corporate networks[4]. Multiple layers of protection, including encryption, authentication, firewalls, security protocols, intrusion detection, and intrusion prevention systems usually guard enterprise networks[5]. However, most embedded systems are only secured using password protection and standard encryption protocols. Embedded systems do not have firewalls and are only protected by passwords in most cases[6].

An Embedded system is a special microcomputer-based system designed as an end product device[7]. This system is completely encapsulated in the device it controls. The hardware may be reduced to a circuit board attached to an electronic, mechanical, or chemical system. The software in particular, is either entirely hidden or reduced to a user interface. The user of the device cannot access or modify the software interface. Embedded systems vary in hardware and software complexity. Applications that process critical information require complex embedded systems with redundant control mechanisms. Such applications may include medical devices[8], nuclear plants, and chemical processing systems[9]. Breakdowns and malfunctioning are included in the design process of the embedded systems. The basic components of embedded systems include microcontrollers, Memory unit, Input/output interfaces, Input/Output devices, Sensors, and several buses. An embedded system is a highly integrated chip that contains all the components necessary to accomplish a very specific task.

The implementation of security measures is challenging and not easy due to the constraints on resources of embedded devices[10,11]. Modern devices and control systems are modular and very flexible. Individual elements of the system can be changed or upgraded independently from other elements. Modern embedded systems are compact, reliable, upgradeable, accessible to various buses, and have less power demand. Embedded systems are widely used in a variety of applications ranging from simple measuring devices to more complex and distributed control systems. Embedded systems applications include data acquisition, signal processing, networking, distributed system, sensors etc. It is this complexity of design and uses which makes security concerns challenging, and often only considered in the last phases of the design. Security issues dominantly come into play only when the system is connected to a network. A common mistake is to consider stand-alone units safer. Because it is almost impossible to add security to an existing product, it must be a prime design goal from the conception through production, deployment, and disposal of the embedded device.

Memory and embedded software are the parts of the embedded system, which are most vulnerable to security breaches[12]. A typical embedded system may have several megabytes of Flash ROM, which are almost never fully utilized. These memory patches provide mechanisms for security breaches. The hardware memory may contain firmware or data. Non-volatile memory chips are found in many hardware devices. These modern embedded systems increasingly use Flash memory, which can be re-written by software. Therefore, EEPROM chips remain a major security breach for storing software viruses. Modern embedded systems, increasingly, use reduced operating systems to manage complex codes. Software attacks against the operating system kernel increase the risks of security breaches because the kernel has full access to the system's address space. As software applications get more complicated, the likelihood of software viruses and security vulnerabilities increase. For user accessibility, more embedded systems are being connected to the Internet, which propagates security breaches. Many additional factors compound security concerns. These factors include hardware devices constraints, production costs, and security costs. Unfortunately, with the advances of technology embedded systems are becoming increasingly vulnerable to security breaches.

Despite these measures, security of embedded devices is often relaxed due to assumptions that embedded devices have sufficient security with encryption and authentication and thus are not vulnerable to cyberattacks[13]. The number of sophisticated embedded attacks has significantly increased, greater security measures are therefore required for embedded systems[14]. Unlike standard computers and network systems, embedded systems are designed to perform a designated set of tasks. These devices are typically designed to minimize the processing cycles and reduce memory usage, as there are no extra processing resources available. Most of the embedded devices do not support security solutions designed for large computers and network systems. Most embedded systems have several security challenges due to irregular security updates, attack replication, dependability, device life cycle, design protocols, and remote deployment. Embedded devices are deployed in the field, outside the network zone. These remote or mobile devices may be directly connected to the internet, without the security layers provided to secure networks.

With the growing trends in embedded systems design and security vulnerabilities, new security concepts must be investigated to make embedded devices more secure[15]. Currently, there are several software and hardware security methods that are implemented in embedded systems with varying degrees of success[16,17]. The software security methods include password protection and encryption protocols[18]. The hardware methods include interrupt-based methods, memory-based methods, PC-based methods, and backup-based methods, which are implemented with various design implementations. Despite the enormity of security challenges, they are often ignored during the initial design and development period of an embedded system, thus leaving many devices vulnerable. We postulate that this design shortcoming is perhaps due, in part, to the teaching inadequacy of embedded security in early engineering education.

In this paper we propose a framework to design an undergraduate curriculum that will incorporate security concepts in embedded systems courses. The significance of this paper is to enforce life-learning awareness of security concerns in embedded systems design in early training of undergraduate students[19]. The long-term goal of this paper is to promote security awareness in embedded systems at the undergraduate level.

II. Curriculum Design Methodology

Traditionally, the embedded systems undergraduate curriculum is delivered through a sequence of courses that address the hardware and software of embedded systems. These courses are intensive and require great efforts from both the instructor and the students. We propose to redesign the undergraduate curriculum to integrate security concerns and concepts during early phases of embedded systems design. The goal of this new curriculum is to create new learning materials and teaching strategies, which will re-enforce life learning skills that incorporate security concepts in the design of embedded systems. The curriculum will include a research component to assess the student's learning skills. Through this curriculum, undergraduate students will learn the basics of integrated and collaborative research as well as research ethics. The key educational components that need to be addressed during the design and the delivery of the embedded systems design curriculum should include the following components:

- The curriculum should integrate new learning materials and teaching strategies, which integrates security concerns into embedded systems design.
- The educational activities of the curriculum should re-enforce life-learning strategies to incorporate security concepts in the early phases of embedded systems design.
- The curriculum should include a research component, which will offer a test bench for the objectives, learning materials, and teaching strategies of the new curriculum.
- The curriculum should include a component that will develop faculty expertise through workshops that emphasize security awareness and teaching concepts in embedded systems design.

Incorporating security concepts in the early design of embedded systems will provide a systematic assessment of robustness and safety of an embedded system. This unique curriculum design of embedded system curriculum provides a rich education to undergraduate students that incorporate security concerns in embedded systems design. It is anticipated that the design curriculum would forge a life learning that will re-enforce security concerns in future embedded systems. The proposed curriculum promotes techniques that are capable of reducing the impact of security breaches in embedded systems.

III. Curriculum Design Strategies

The following sections contain a work plan aimed at accomplishing the design of this curriculum. The proposal plan is divided into modules with specific descriptions of the tasks that will help the instructor to construct and organize the delivery of embedded systems with built-in security. Figure 1 illustrates the modules that describes the new curriculum.

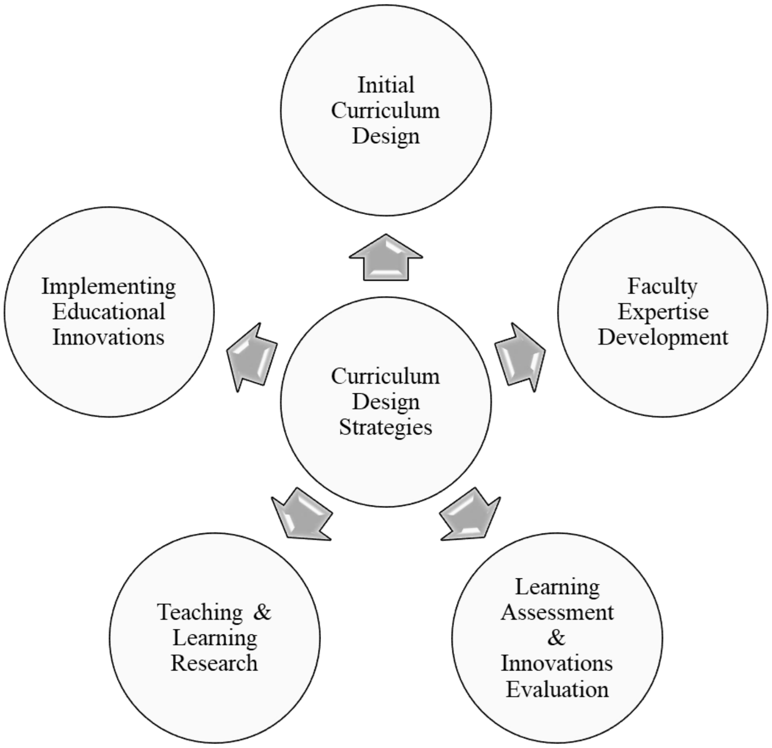
Initial Curriculum Design
Faculty Expertise Development
Learning Assessment & Innovations Evaluation
Teaching & Learning Research
Implementing Educational Innovations
Curriculum Design Strategies

Figure 1: Curriculum design modules of secure embedded systems

The major tasks of designing an embedded systems curriculum with security goals include:

- **Initial Curriculum Design:** This task will focus on the initial design of curriculum aimed at integrating security concepts in embedded systems design. We propose that the instructor identifies the current security concerns in the embedded system and constructs a set of research topics, which will help in defining the objectives and learning outcomes of the curriculum.

- **Developing Faculty Expertise:** The instructor may have many years of academic and industrial experience in the area of embedded systems design. We propose that the instructor should seek every opportunity to acquire new knowledge and expertise through workshops and partnerships with other educators in the areas of innovative security and embedded systems. The newly acquired material and teaching practices ensure revision and improvement of the curriculum in concordance with embedded system market.

- **Assessing Learning & Evaluating Innovations:** This task is always true for any curriculum design, but it is particularly important for embedded systems, which have real impacts on the learning and skills of future engineers. We propose and emphasize that the instructor constructs a set of learning skills, learning outcomes, and assessment tools, which he/she will use to assess and evaluate the learning skills of undergraduate students while learning embedded systems design and applications.

- **Conducting and Incorporating Research in Teaching & Learning:** In this paper, a set of initial research activities has been designed to conduct research to test, assess, and improve the objectives, learning materials, and teaching approaches of embedded system curriculum. The required development of embedded systems and design tools should be made available for the students to implement the new teaching materials.

- **Implementing Educational Innovations:** The instructor must recognize the challenges of security concerns in embedded systems design. Creating a unique curriculum on security concepts in early phases of embedded systems design will require collaborative efforts between the instructor and faculty members in the field of security and embedded systems.

IV. Learning Materials and Teaching Strategies

The traditional embedded systems design curriculum does not integrate a great deal of security concerns beyond proper function of the system. Recent research interests in security of embedded systems have sprouted graduate curricula that address some of the challenging security issues. Typical undergraduate course in embedded systems design may include introduction to embedded systems, hardware architecture, instruction set, instruction format, and assembler directives, software architecture and programming model, memory interfacing, I/O interfaces, interrupts and timers, and communication schemes (serial and parallel communications). And if time permits, a topic on applications such as A/D converters, signal processing techniques, closed-loop digital feedback control, interfacing power electronics and electromechanical systems, and accessing embedded system peripherals may be included. Some instructors have divided these topics into a sequence of courses. Many of these topics may be implicated in security vulnerabilities. Instructors have attempted to incorporate security concepts to invoke students' interests, however, the long topics and the shortcoming of students accumulated knowledge may not favor just superposing the additional knowledge to previous acquired knowledge. Furthermore, just adding security on exciting embedded systems is almost impossible. Therefore, there is a need for structural design of curriculum, which focuses security concepts on embedded systems design.

This curriculum recognizes several challenges for effective undergraduate education of security concepts in Embedded Systems. These challenges include laboratory existing tools, classroom, and laboratory activities, and teaching large numbers of students from diverse backgrounds. In fact, we, like many embedded systems instructors, have noticed an increase in students' interest in embedded systems design from various disciplines, especially those disciplines that require acquisition and analysis of data from distributed sources. We anticipate an even more increased interest in embedded security because of the impact of embedded systems in several fields. The central objective of this project is to devise teaching and research infrastructures, which will enable a successful design of an undergraduate curriculum on security concepts in early phases of embedded systems design. Figure 2 illustrates the design strategies of secure embedded systems.

We propose the following design strategies to be implemented during the curriculum design:

- **Identify Security Challenges in Embedded Systems:** The long-term goal of the curriculum on security concepts in early phases of embedded systems is to provide a state of art knowledge and valuable education, which will prepare undergraduate students not only for future challenges in the workplace, but also for the challenges of the global market. Therefore, the curriculum will be based on academic and industrial challenges of security in embedded systems to shape long-term learning behaviors and sound practices. We propose that the instructor should draw from his/her academic and industrial experiences to create a comprehensive set of challenges, which he/she will use to design performance criteria for learning.

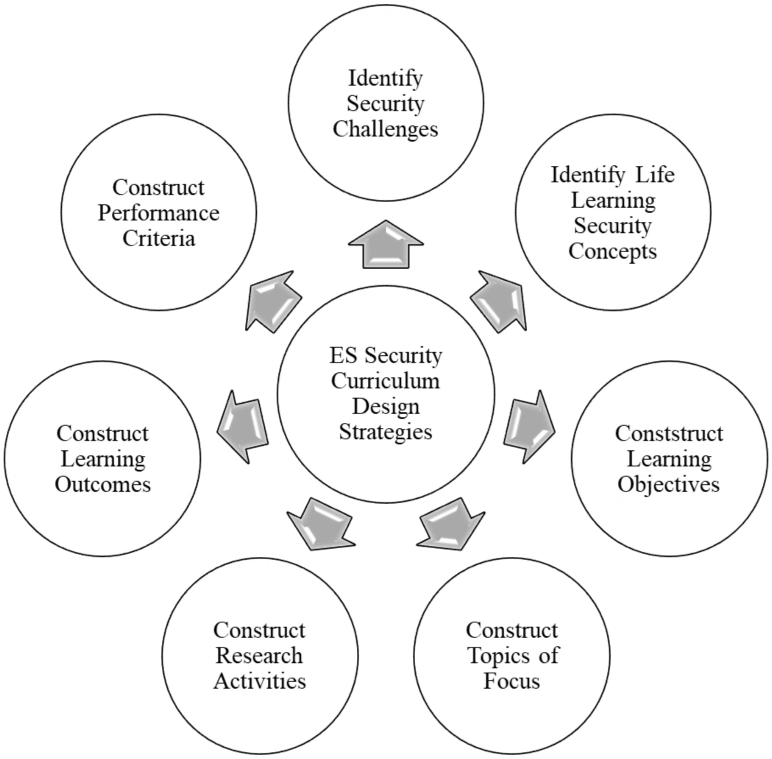
Identify Security Challenges
Identify Life Learning Security Concepts
Constsruct Learning Objectives
Construct Topics of Focus
Construct Research Activities
Construct Learning Outcomes
Construct Performance Criteria
ES Security Curriculum Design Strategies

Figure 2: Curriculum design strategies of secure embedded systems

- **Identify Life-Learning Security Concepts:** Despite the advances in computer technology, embedded systems will continue to be the target of increased security breaches. Security concerns will continue to be important criteria, which will shape the future of embedded systems. Therefore, incorporating security concepts in the design of embedded systems becomes a life-learning behavior, which will shape the professional behavior of future embedded systems engineers. We propose that the instructor identifies a set of life-learning behaviors based on the current security challenges.

- **Construct Curriculum Objectives:** We propose that the instructor constructs a set of objectives for designing a curriculum, which will focus on the application of security concepts in the early phase of embedded systems design. Based on Bloom’s Taxonomy, the curriculum should include levels such as recall of information, comprehension, application, and analysis.

- **Construct New Topics of Focus:** We propose that the instructor identifies new topics that will help focus the curriculum on the application of security concepts in the design of embedded systems. Some of the topics that should be considered include current security challenges, vulnerable hardware and software aspects of embedded systems, hardware security concerns, software security concerns, and current security measures.

- **Construct Potential Research Activities:** We propose that the instructor constructs a set of potential research activities, which will highlight the objectives of the proposed curriculum and challenge the students. The latest embedded development boards and software tools will be used to accomplish the proposed research activities. The primary objective of these research activities is to incorporate security concepts in early phases of embedded system design and provide a test bench to assess the effectiveness of the new learning materials and teaching strategies. Students should be encouraged to provide design modifications, which will create a cyclic model to improve students' activities and learning approaches.

- **Construct Research-Based Learning Outcomes:** The research projects will be designed to help undergraduate students integrate security concepts into prior acquired learning skills. The new learning outcomes are, therefore, targeted towards the learning skills of security concepts and their applications in embedded systems. The following learning outcomes should be monitored and analyzed to assess the learning skills of the undergraduate students:

  - ✓ Have a clear understanding of embedded systems,
  - ✓ Have a clear understanding of embedded hardware and software architectures,
  - ✓ Know how to use embedded design tools,
  - ✓ Understand the security vulnerabilities of embedded systems,
  - ✓ Understand how to apply the security concepts to embedded systems,

- **Construct Performance Criteria:** We propose that the instructor constructs a set of performance criteria, which will be used to assess and evaluate the new learning skills acquired by the undergraduate students. The performance criteria should emphasize how students managed to integrate the new skills into their prior learning skills. The following performance criteria should be considered:

  - ✓ Ability to assess security vulnerability of an embedded system,
  - ✓ Ability to determine security breaches of an embedded system,
  - ✓ Ability to find areas of an embedded system in need of security,
  - ✓ Ability to design and produce a secure embedded system,

V. Development of Faculty Expertise

No doubt embedded systems instructors have been involved in continuous education. They may have been attending short courses, workshops, and engineering education societies to strengthen their knowledge in embedded systems and to improve their teaching practices. The instructor should acquire new knowledge and teaching skills, which will enable him/her to revise his proposed curriculum and improve his/her teaching practices. The instructor should attend academic and industrial workshops and seminars, which focus on security concerns in embedded systems. Through this continuous education, the instructor should identify current industrial security needs, which help the instructor to design real world teaching activities. The instructor should recognize from his/her past experience that real world problems increase the learning interest of students and improve their abilities for self-assessment. The instructor should also establish new relationships with interested faculty members who share his/her teaching and research interests. In particular, the instructor should attempt to establish collaborations with faculty members. Through these interactions, the instructor attempts to identify common interests and teaching practices.

VI. Learning Assessment and Innovation Evaluation

The assessment of learning and evaluation of the educational innovations of this curriculum on embedded systems will assess the educational and research experiences of the undergraduate students. The instructor should design custom assessment tools, which will assess the learning skills and learning behaviors of the undergraduate students. The objective of the curriculum is to provide education for undergraduate students and test and validate the learning objectives of the curriculum. The instructor should continuously monitor the accomplishments and learning progress of the students by assessing their learning behaviors using the following set of criteria:

- Understanding of embedded systems,
- Understanding of embedded hardware and software architectures,
- Use of embedded design tools,
- Understanding of security vulnerabilities of embedded systems,
- Application of security concepts to embedded systems,

The integration of security concepts into the prior knowledge on embedded systems design should be measured. The following performance abilities of the students should be considered:

- Assess security vulnerability of an embedded system,
- Determine security breaches of an embedded system,
- Find areas of an embedded system in need of security,
- Design and produce a secure embedded system,

The research projects are designed based on our past experience and teaching practices. We include new teaching materials and teaching approaches, which are most likely not familiar to the undergraduate students. The instructor should monitor the transition and adjustment of the students to a research-based teaching environment. The following strategies should be used to assess and evaluate the research projects:

- ***Research Project Selection, Planning, and Modification:*** Although the objectives of projects are initially defined, undergraduate students are encouraged to integrate their educational experience and prior learning skills. The instructor should monitor and assess the progress of the students throughout the course period. Undergraduate students should be mentored to design realistic specifications that meet the goals of the project and time constraints of the project.

- ***Research Project Assessment and Implementation:*** The instructor should monitor the progress of the participating undergraduate students and document their educational activities, research activities, experience shortcomings, and potential social difficulties.

- ***Research Project Evaluation:*** Evaluation of the efforts of the students should be based on many criteria. These criteria include project planning and modification, project implementation, oral presentations, participation, and final written report.

VII. Research and Learning

The research projects are uniquely designed not only to highlight the educational experience of the students but also to challenge their creativity in real world applications. These projects are designed to address specific research challenges on security in embedded systems design. Students should be encouraged to work on complementary projects to promote student-student communication and work collaboration. Research projects are designed in modular fashion to encourage students to independently pursue other variations and applications of the projects. The following research projects will be available to instructors of embedded systems design to include in their curricula. Some projects are explicitly defined while others are implicitly defined and will be refined as students work together. The research projects should be constantly assessed and modified to meet the goals and time constraints of the research projects. Undergraduate students should have unique research experience and learn research ethics and technical writing skills. The following research projects are designed to address the latest research findings on security concerns in embedded systems design. The research projects are challenging activities with controlled difficulties, which encourage the students to strive for high performance standards and improve their learning skills. The technical core of these projects centers on assessing security vulnerability, determining potential security breaches, finding areas of an embedded system in need of security, and producing a secure embedded system. The following are some research activities that can be considered by embedded systems design instructors:

- ***Memory Operations and Array Processing:*** The objective of this activity is to review and strengthen the understanding of memory operations and array processing. The student will write programs (assembly and C) to perform simple memory operations and array processing routines using development embedded systems. The student will then identify potential security breaches and provide solutions to fix them.

- ***Simple and Complex Arithmetic Operations:*** The objective of this activity is to build a strong foundation for embedded mathematical operations. The student will implement several arithmetic operations such as 16-bit addition, 32-bit addition, 16-bit multiplication, and 16-bit division using 16-bit microcontrollers. The student will identify and fix potential software bugs that will jeopardize the integrity and security of the embedded system.

- ***Interrupts and Real Time Interrupts:*** The objective of this activity is to highlight the significance of the interrupt system of an embedded system. The student will implement several interrupts operations including real time interrupts. The student will then identify and assess potential security breaches through the interrupts system and provide solutions.

- ***Asynchronous Serial I/O Interfacing and Security Issues:*** The objective of this activity is to review the principles of asynchronous serial I/O interfacing. This activity highlights the security issues related to serial interfacing. The student will write a program to interface two microcontrollers using the built-in asynchronous serial I/O interfaces. The goal of the activity is to read from or write to the memory of both embedded systems. The student will identify and provide solutions for the potential security breaches by unauthorized entity.

- ***Synchronous Serial I/O Interfacing and Security Issues:*** The objective of this activity is to review the principles of synchronous serial I/O interfacing. The student will write a program to interface two microcontrollers using the built-in synchronous serial I/O interfaces. The goal of the activity is to read from or write to the memory of both embedded systems. The student will identify and provide solutions for the potential security breaches by unauthorized entity.

- ***Program Destruction:*** Destroying a program in memory is not a cost-effective alternative. But if needed, it can be implemented in stand-alone devices. A program will be written to detect whether an entity is attempting to breach memory. If an attempt is made, the protective program will delete the memory. The device can only be active again after inserting the proper program. The student will use and assess the effectiveness of this security concept.

- ***Security by Adding Codeless Lines:*** This activity focuses on security by adding meaningless lines in memory between authentic codes. The main program will be written to ignore the meaningless lines. If a memory breach occurred, the unauthorized entity may not be able to identify the proper operation of the main program. The student will use and assess the effectiveness of this security concept.

- ***Maintaining Program Integrity by Checksum:*** The primary objective of this activity is to maintain integrity of data in memory. The program will be written to detect the checksum of its data. Changing any data inside the memory will create a mismatch in the data checksum. The program detects the mismatch and stops operating. The student will use and assess the effectiveness of this security concept.

- ***Protection by Encryption:*** This activity is especially applicable for distributed systems. Data and program will be encrypted, which make it very hard (if not impossible) to decipher the data is a memory breach occurred. The problem, of course, is to design an encryption method that will allow the embedded system to use the program and data. The student will provide solutions on how to use encryption techniques with limited cost burden on products.

VIII. Conclusion

In this paper, we presented a case study of curriculum design, while addressing emerging technological innovations and urgent global challenges. These challenges range from security concerns that can affect banking systems and power grid networks to climate changes and world global economy. In particular, we investigated the redesign of embedded systems curriculum. Embedded systems design is a fundamental core skill of electrical and computer engineering curriculum. We proposed to include and mitigate security concerns in every topic of the embedded system design course to enlist the concepts of security awareness in the design. We recognize that just adding the security concepts as a separate topic did not seem to enhance and safeguard many embedded devices that are still under constant threats of breaches. We emphasized the continuous education and development of embedded system educators to constantly expose them to new emerging technology and global challenges that will face the future work force both locally and globally. We also proposed including research in the curriculum as a means to encourage and prepare learners using real and practical problems. Finally, through this case study, we created a sense of awareness when designing courses in all aspects of academic curricula to address the emerging innovations and the new global challenges that will likely continue to grow due to climate changes, world population expansion, limited resources, and regional conflicts.

Bibliography

Mohammed Ferdjallah: Dr. Mohammed Ferdjallah is an Assistant Professor in the Department of Computer Science & Electrical Engineering at Marshall University. He received his PhD degree in Electrical and Computer and MS degree in Biomedical Engineering from The University of Texas Austin. He also received his MD degree from the International University of the Health Sciences. He has multidisciplinary expertise in image & signal processing, computational modeling, and statistical data analysis. As an electrical and biomedical engineering scientist, he conducted research in computer modeling of the brain, cranial electrical stimulation (CES), electrical impedance tomography, electrode design, and EMG and muscle action potentials and ions channels simulation & modeling. His technical research interests include digital systems, embedded, systems, computer architecture, adaptive and system identification, modeling and simulation, and signal and image processing. His clinical research interests include impacts of chronic diseases in elderly (such as Alzheimer's disease, cancer, and diabetes), innovative technology for drug addiction treatment and prevention, medical records, comparative outcomes research, and biomedical sciences. He has successfully published several peer-reviewed articles in biomedical sciences, physical medicine and rehabilitation, modeling and simulation of physiological signals, motion analysis, and engineering.

Asad Salem: Dr. Asad Salem is a professor in the Department of Mechanical Engineering at Marshall University. Some of his research interests include renewable and sustainable energy sources such as wind and solar, thermal science, computational fluid dynamics, finite elements, co-generation, and energy storage. Dr. Salem has taught several courses on solar and wind energy and fuel cells. He has extensive experience advising undergraduate and graduate students on several projects. At the current time Dr. Salem is involved with GE Renewable Energy and Aker Offshore Wind in several projects and prospects related to offshore wind farms at the USA East and West Coasts.

Hadjer Bouchareb: Hadjer Bouchareb is a K-12 instructor of English and Basic Computer Science Technology. She received her BA degree in Translation and Interpretation from the University if Kasdi Merbah, Algeria. She also received her Associate Degree in Computer Science Technology from the Ghardaia Technology Center, Algeria.

References

1. *Embedded Systems: Technologies and Markets,*. August 2020,. *BCC Publishing*.
2. Toulson R, Wilmshurst T. Embedded Systems, Microcontrollers and ARM. 2012:3-16.
3. Koopman P. Embedded system security. *Computer*. 2004;37(7):95-97. doi:10.1109/MC.2004.52
4. Bond M, Anderson R. API-level attacks on embedded systems. *Computer*. 11/01 2001;34:67-75. doi:10.1109/2.955101
5. Gopalan G, Srivatsava A. Big data analysis for implementation of enterprise data security. *IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS)*. 01/01 2012;2:742-746.
6. Melicher W, Kurilova D, Segreti SM, et al. Usability and Security of Text Passwords on Mobile Devices. ACM;
7. Patterson D, Hennessy J. *Computer organization and design - the hardware / software interface (3. ed.)*. 2007.
8. Paolo D, Maria Chiara C, Antonella B, Arianna M. Micro-systems in biomedical applications. *Journal of Micromechanics and Microengineering*. 2000/06/01 2000;10(2):235. doi:10.1088/0960-1317/10/2/322
9. Malinowski A, Yu H. Comparison of Embedded System Design for Industrial Applications. *IEEE Transactions on Industrial Informatics*. 2011;7(2):244-254. doi:10.1109/TII.2011.2124466
10. Ravi S, Raghunathan A, Chakradhar ST. Tamper resistance mechanisms for secure embedded systems. *17th International Conference on VLSI Design Proceedings*. 2004:605-611.
11. Kopetz H. The Complexity Challenge in Embedded System Design. 2008:3-12.
12. Sharma A. Advanced Semiconductor Memories: Architectures, Designs, and Applications. 2009:
13. Anderson R, Bond M, Clulow J, Skorobogatov S. Cryptographic Processors-A Survey. *Proceedings of the IEEE*. 03/01 2006;94:357-369. doi:10.1109/JPROC.2005.862423
14. Yoon MK, Mohan S, Choi J, Kim JE, Sha L. SecureCore: A multicore-based intrusion detection architecture for real-time embedded systems. 2013:21-32.
15. Kocher P, Lee R, McGraw G, Raghunathan A, Ravi S. Security as a new dimension in embedded system design. ACM;
16. Patel K, Parameswaran S, Shee SL. Ensuring secure program execution in multiprocessor embedded systems: A case study. 2007:57-62.
17. Ragel R, Parameswaran S, Kia S. *Micro embedded monitoring for security in application specific instruction-set processors*. 2005:304-314.
18. Ferdjallah M, Redmon C. Embedded System Security Using Cryptographic Techniques. *Integrated Design and Process Technology, IDPT-2010*. June 2010:1-4.
19. Pereira G, Vieira A, Machado R, Ferreira B, Dias L, Oliveira J. Activity based simulation – A modeling tool for new teaching-learning strategies. *Journal of Simulation*. 02/12 2022:1-10. doi:10.1080/17477778.2022.2032431