International Journal of Embedded and Wireless Communication Systems
Volume1, Issue 1 2025

A Study on The Technical Foundation Of Embedded Systems

Ms.Naila Fathima, Nahri Syed Hyder Hussain, Mohd Sayeed Ali Mudassir

1Assistent Professor,Department of CSE-AIML, Lords Institute of Engineering & Technology
2,3Student Department of CSE-AIML, Lords Institute of Engineering & Technology.

Research Article

Date of Submission: 13-05-2025 **Date of Acceptance: 11-05-2025** **Date of Publication: 09-07-2025**

Abstract

The term "embedded system" refers to a class of computer systems that are specifically designed to monitor and control physical items. An introduction to electronic systems (ESs), including their cybernetic-physical character and how they can be constructed to give the desired performance with a minimum amount of hardware, is presented in this book. " In addition, it covers a variety of design methods. Computer algorithms are implemented in hardware to the greatest extent possible. Models of finite state machines can be used to implement various parts of complicated ESs (FSMs). Different hardware accelerators in ESs are frequently implemented using field-programmable gate array (FPGA) technology. The book devotes a significant portion of its last chapter on programmable logic controllers (PLCs), which are widely utilised in industry.

Introduction

The core of embedded system engineering is the design of electronic systems that combine software programming techniques with hardware circuitry to provide solutions for practical applications and projects. The two ways that computer processes interact with the physical world (non-functional needs) provide additional challenges and restrictions for embedded system designers, who are already constrained by the platform and the environment. Note that the platform interaction includes all hardware and software elements, such as operating systems and communication networks, as well as platform-specific scheduling and communication techniques. [1] The development of a viable Embedded Systems Design Science is a significant potential and problem for computer science. Hardware and software components that are intended to carry out certain tasks make up embedded systems. They are used in industrial processes, consumer electronics, transportation, electrical appliances, and energy distribution, to name a few areas. When designing embedded systems, extra-functional requirements including resilience, autonomy, responsiveness, and effective use of time, memory, and energy must be taken into account. [2] Because of their speed, size, power, precision, reliability, and flexibility, embedded systems are increasingly widely used and convenient. Because of their speed and accuracy, they are a great option for high-precision operations in the medical and defense industries. Their tiny size makes them ideal for facilitating intelligence in mobile and portable devices. Another advantage of the intelligent system is its lower power usage. The capacity to react to real-time events, such time analysis and worst-case execution timings, is essential for embedded system design. Embedded systems must meet safety, availability, reliability, and dependability criteria. These systems need limited hardware capacity because to their small size, mobility requirements, and very cheap manufacturing costs. [3] The complexity of real-time embedded systems tends to raise the demand for engineering, early error detection, high-level design, integration, productivity, verification, and maintenance, which increases the importance of life cycle qualities such as maintainability, portability. [4]

Tools for developing embedded systems software in an integrated development environment

An integrated development environment offers all the necessary tools for developing embedded software. Writing software for an embedded device may need all of these tools. It's quite helpful to have all the tools you need in one place, from writing code to testing it. An IDE typically consists of a code editor, a compiler, and a debugger. Additionally, the IDE has an interface. Popular IDEs include Adobe Flash Builder, X code, Blue J, Code Blocks, Eclipse, and Android Studio.

Real time Operating System

All of the tools required to create embedded software are available in an integrated development environment. All of these tools

may be necessary when writing software for an embedded device. Having everything you need in one location, from developing code to testing it, is quite beneficial. A code editor, compiler, and debugger are the standard components of an IDE. The IDE also has an interface. Adobe Flash Builder, X Code, Blue J, Code Blocks, Eclipse, and Android Studio are a few of the well-known IDEs.

Finite State Machines

Finite State Machines (FSM) may be used to characterize the dynamic behavior of systems and its constituent parts. As a result,

FSMs developed in OO languages are usually hard to maintain. The State design described in [5], which is often used to construct FSMs in OO languages, is likewise impacted by these problems. To address this issue, we provide an alternative approach. A blackbox framework is also used to illustrate this method. A tool that is offered may also be used to automate the setting of the framework. The tool makes it simple for developers to create FSMs from a specification. [6]

Review of Literature

"A model that can identify fire hazards and alert the closest fire station with its location was developed by Subbachary Yerroju and Ravi Kishore Kodali. The authors' circuit and sensor work flow was used to demonstrate the model creation process. [7]. The use of IOT in fire and safety monitoring was discussed by S.R. Vijayalakshmi and S. Muruganand. This article discusses the benefits and drawbacks of both wired and wireless security systems. [8] Pedersen S., Fountas S., and Blackmore S. described how combining conventional weeding, grass-cutting, and many other operations with a GPS-based system may increase efficiency and effectiveness, despite the system's higher cost. The usage of an MF-scamp robot for scouting, weeding, and harvesting is shown in a study. The authors state that since agricultural robotic systems are costly, researchers must identify a less priced substitute. [9] "Jian-Jie Dong, Suleman Shahid, Abdullah Al Mahmud, Catherine J. Stevens, and Omar Mubin investigated the employment of robots in education to accomplish the learning goal. An essay offers a thorough discussion of the many educational uses for robots. [10] "In the field of healthcare and medicine, researchers Allison M. Okamura, Maja J. Mataric C, and Henrik I. Christensen conducted a literature review on the use of robot" s. Hamshiri1, Cornelia Weltzien, Ian J. Yule, Tony E. Grift, Siva K. Balasundram, Lenka Pitonakova, Des Ahmad, and Girish Chowdhary investigated contemporary farming using robotic tractors and drones. According to an article, robots such as the bonirob, a sweeper, and others may be very useful in agriculture. [12]

Objectives

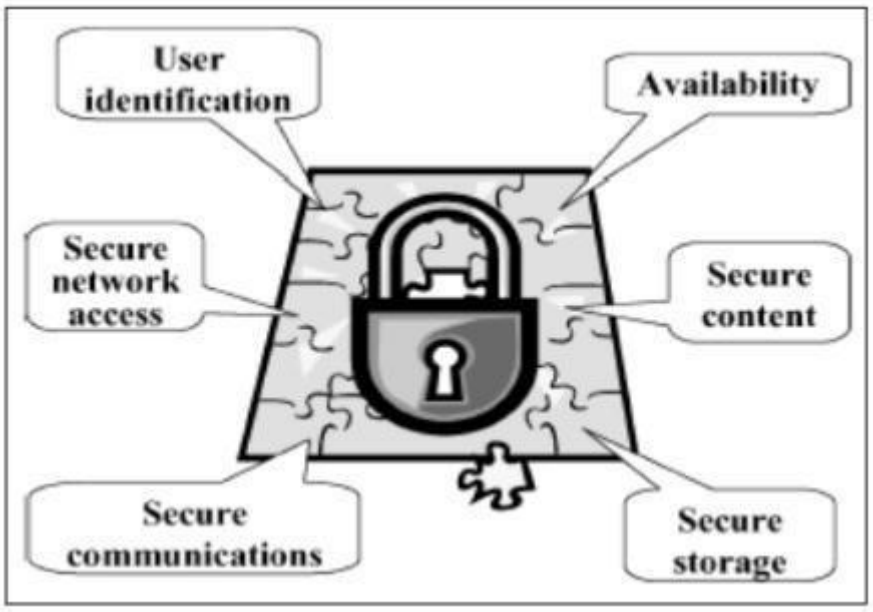
- In order to learn about the embedded systems.
- To learn more about embedded systems' integrated development environments
- To learn about real-world operating systems
- In order to learn more about finite state machines.

Research Methodology

In scientific terminology, "research" might be described as a systematic and scientific search for pertinent data on a scientific problem. The data collected for this study comes from a number of previously published sources, making it secondary in nature. The Internet was one of the many sources from which the data for this study was collected.

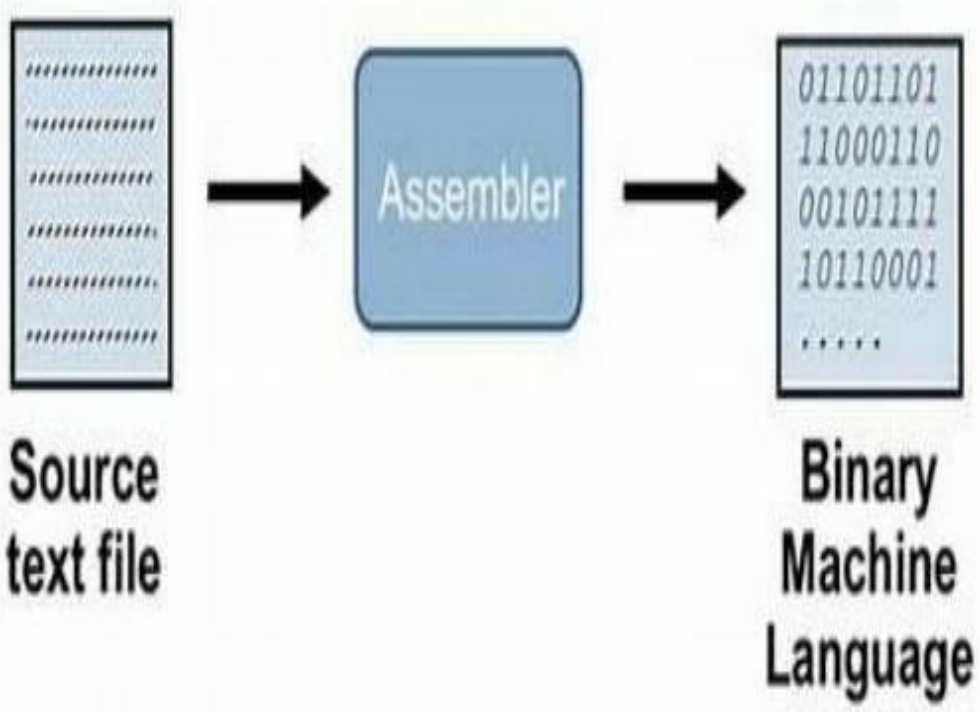
Result and Discussion

Any malicious actor has the ability to damage the embedded system's essential functionalities. There are many people involved in the design, production, and usage chain of an embedded system. The security requirements for embedded systems, like thermostats, and mobile devices are different from those for internet thermostats. [13]



Security requirements for embedded systems are depicted in Fig. 1.

Fig. 1 illustrates the security needs for embedded systems. User identification is the process of confirming an individual's identity prior to granting them access to a system or network. The device gains access to the network or service after it has been authorized. Assembler is the most popular software development tool for embedded systems. Its main purpose is to translate assembly



language code into bits, op-codes, and machine code.

Fig. 2 Working of Assembler

For the embedded system, one microcontroller is proposed in conjunction with other sensors, such as a flame sensor and other gas sensors. Figure 3 shows the suggested system in action. [14]

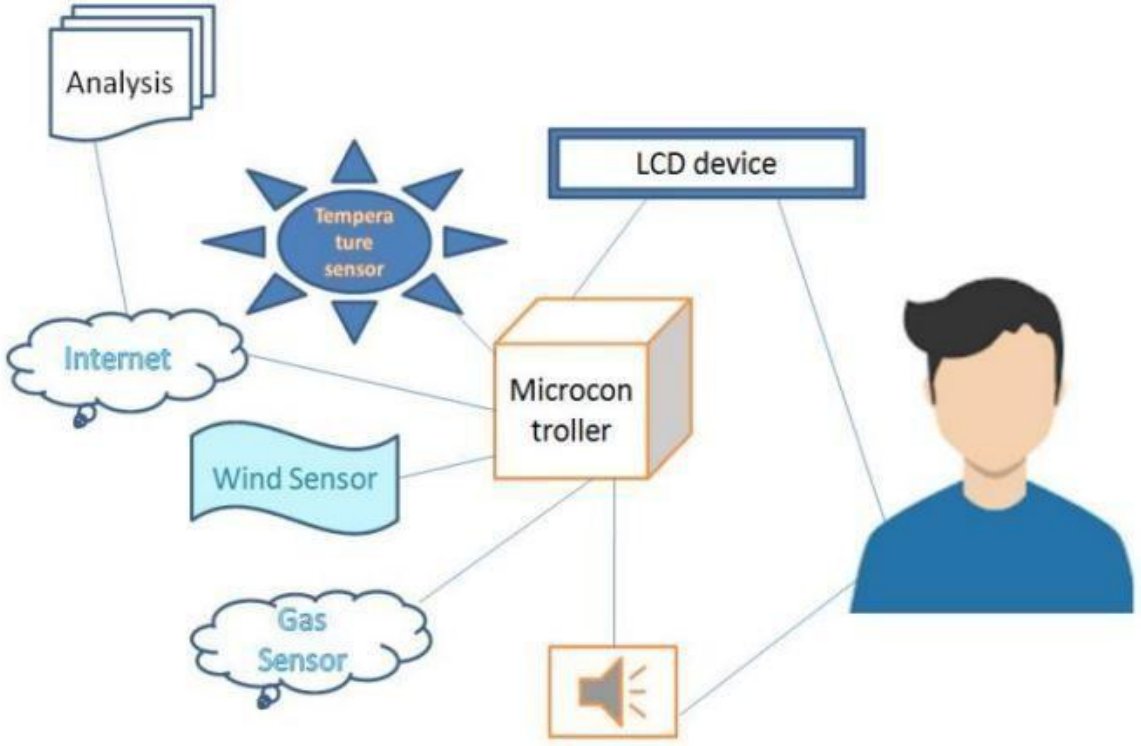


Fig. 3 proposed system

This FSM's schematic is shown in fig 4. This FSM has a single goal: to help you. After every 80 characters, it adds a newline to the text. Three states represent a single line of text, therefore it can do this. At this point, the FSM is in a "Empty State," which is also the default state in this FSM. [15]

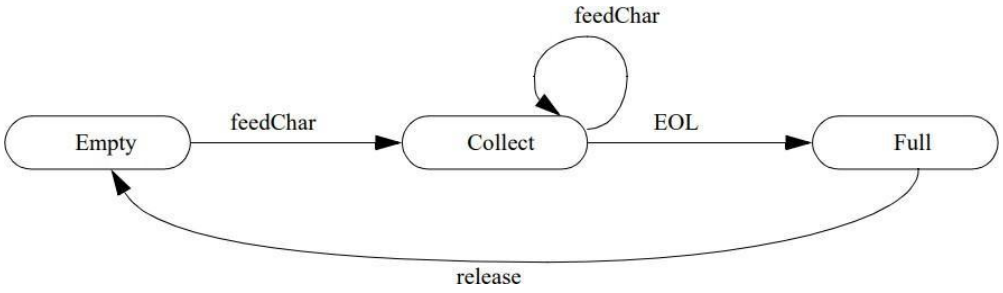


Fig. 4 WrapAText: a FSM for wrapping text

In the Collect State, it waits for more characters to be received (feedChar). The EOL (end of line) event is used to transition to the Full State if at least 80 characters have been received. After the line is printed, the FSM returns to the Empty State (release event) in preparation for the next line of text.

Conclusion

Applications for embedded systems may be found in many different fields, including computer networks, smart cards, satellites, digital electronics, and telecommunications. Depending on the application, an embedded system may be either programmable or non-programmable and perform one or more tasks. Because embedded system technology reduces circuit complexity, the final product is smaller and less expensive. Several computer programs were used to make this design.

References

1. S. Ravi and A. Raghunathan, "Security in Embedded Systems: Design Challenges", ACM Transactions on Embedded Computing Systems, Vol. 3, No. 3, August 2004, Pages 461–491.

2. T.A. Hen zinger, B. Horowitz, and C.M. Kirsch," Giotto: A time-triggered Language for embedded programming", International Workshop on Embedded Software EMSOFT 2001: Embedded Software pp 166-184.

3. T. A. Henzinger, "The Embedded Systems Design Challenge", 4th international conference on Formal Methods, Canada, 2006

4. Noorbasha Johnsaida, Lakkisetty V.Rahul, Tadi Shalini, “IOT Based Smart Fire Emergency Response System”, International Journal of Advance Research and Development, volume 3 issue 2.

5. Abhishek Sharma, Sudha Pal, Vaibhav Jain, “IOT based smart security system for prevention of industrial hazard”, International Research Journal of Engineering and Technology, volume 5 issue 3.

6. MARK E. CAMBRON :- "Using the Arduino in Freshmen Design". Sixth Annual (FYEE) First Year Engineering.., 2014

7. Ravi Kishore Kodali and Subbachary Yerroju, “IoT based smart emergency response system for fire hazards”, Publisher: IEEEConference Location: Tumkur, India.

8. R.Vijayalakshmi , S.Muruganand, ”Internet of Things technology for fire monitoring system”, International Research Journal of Engineering and Technology volume 4 , issue 6.

9. Pedersen S., Fountas S and Blackmore S, “Agricultural Robots –Applications andEconomic Perspectives” Springer, Part of the Progress in Precision Agriculture book series

10. Omar Mubin,Catherine J. Stevens,Suleman Shahid,Abdullah Al Mahmud,and Jian-Jie Dong, “A Review ofthe Applicabilityof Robots in Education”, Technology for Education and Learning, 2013

11. Allison M. Okamura,Maja J. Mataric C, and Henrik I. Christensen,”Medical and health care robotics achievements and opportunities”, IEEE Robotics & Automation Magazine September 2010.

12. Shamshiri1, Cornelia Weltzien, Ibrahim A. Hameed, Ian J. Yule, Tony E. Grift, Siva K. Balasundram,Lenka Pitonakova, Desa Ahmad, Girish Chowdhary, ”Research and development in agricultural robotics:A perspective of digital farming ”, International Journal of Agricultural and Biological Engineering, volume 11,no 1.

13. ELLIOTT, C., VIJAYAKUMAR, V., ZINK, W., & HANSEN, R. (2007).:- "National Instruments LabVIEW: A Programming Environment for Laboratory Automation and Measurement". Journal of the Association for Laboratory Automation, 12(1), 17–24.

14. Fengjie Wu and Tangmei He:- "Application of Proteus in Microcontroller Comprehensive Design Projects.

15. Ivica Cmvkovic, Component- based Engineering For Embedded Systems, Malardalen University, Department of Computer and Electronics Sweden, ICSE’05.