TRENDS IN EMBEDDED SYSTEM TECHNOLOGY[1]

Sreenivasan R

Lecturer in Electronics Engineering, Government Polytechnic College, Palakkad.

ABSTRACT

Numerous innovative embedded system applications are currently being developed on single-board based technologies, such as platforms of embedded system, in today's advanced world. As a result, embedded technology can be utilized in a variety of settings. As a result, the most recent embedded system platforms and a number of their applications in a variety of fields are discussed in this paper. With the invention of microprocessors and microcontrollers, the embedded systems industry was born. Nowadays, various embedded system platforms like the Arduino, Raspberry Pi, and Beagle Bone Black are available on the market. These embedded platforms make it easier to connect IoT and SoC devices to communication protocols. Also provide ISP, software debugging tools for the system, and use these platforms to create new opportunities and challenges for developing advanced embedded system applications in the future.

Keywords: Embedded system; IoT (Internet on Thing); SoC (System on Chip); Arduino; Raspberry pi; ISP (In System Programming).

INTRODUCTION

An application-specific electronic subsystem used in a larger system like a vehicle, instrument, or appliance is an embedded system. Software, also known as embedded software, and hardware are typically the components of an embedded system. Cores and micro components make up the majority of hardware, while RTOSs, development languages, application development, system level design methodologies, application development tools, compilers, debuggers, and IDEs, ICEs, JTAG emulators, and so on makeup software. System-on-Chip (SOC) architecture makes it possible to combine a number of CPU subsystems for software execution with sophisticated interconnects and hardware subsystems on a single chip thanks to technological advancements. Moore's law has made it possible for inexpensive embedded systems to have a lot of memory and computing power. The Internet has opened up the possibility of inexpensive embedded device network connectivity. By designing a device as an intelligent, network-connected product, any device can now be made more flexible, useful, and frequently less expensive. As intelligent embedded products surround us in our homes, workplaces, automobiles, and on our person as we go about our lives, we are moving toward a world of pervasive computing and network connectivity. Computers of all sizes, from tiny wristwatch

[1] How to cite the article:
Sreenivasan R., Trends in Embedded System Technology, *International Journal of Advances in Engineering Research*, March 2012, Vol 3, Issue 3, 68-76

(IJAER) 2012, Vol. No. 3, Issue No. III, March **ISSN: 2231-5152**

cameras to telecommunication switches and a network infrastructure with thousands of nodes distributed worldwide, are now part of the expanding embedded computing universe. The complexity of developing embedded software applications increases with the size and complexity of microprocessor designs. Developers must use a variety of debugging strategies, including complex break points, occurrence counters, program traces, core dumps, and the increased sophistication of multi-processor designs, to diagnose challenging issues. In addition, developers must still concentrate on application size and speed as microprocessors get faster and cheaper. To find hot spots in code, analyze real-time performance, and debug optimized code, a variety of profiling methods may be required [1]. Since the early 1990s, the embedded market has changed. In an effort to provide embedded developers with a software solution for the development of their devices while minimizing the costs of developing their own operating systems or development tool chains, software providers could make use of the technical advantages that come with a strategic partnership [2].

Late advancement in the space of implanted framework exploded the rise of new stages. These platforms, which are based on computers and embedded systems, can control and monitor everything without human intervention. Nowadays, embedded systems are an essential part of everyday life. These systems—cell phones, smartcards, music players, routers, and automobile electronics—have been influencing and transforming modern lives like never before. An embedded system is a system that is made up of computer software, hardware, and additional mechanical or other technical parts that are made to do a specific job. The majority of embedded systems must satisfy requirements for real-time computing.[1,3]

The major building blocks of an embedded system are listed below:

Microcontrollers / digital signal processors (DSP)

- Integrated chips
- Real time operating system (RTOS) - including board support package and device drivers
- Industry-specific protocols and interfaces
- Printed circuit board assembly.

LATEST TRENDS IN EMBEDDED SYSTEM

Overview of the emerging technological trends and implications in the development of embedded systems.

System-on-Chip (SoC)

Advances in process technology have made it possible to integrate the primary components and subsystems of an electronic product onto a single chip or integrated chipset, bringing about a revolution in the design of integrated circuits (ICs). This improvement has been embraced by originators of mind

(IJAER) 2012, Vol. No. 3, Issue No. III, March **ISSN: 2231-5152**

boggling chips since it allows the most elevated conceivable degree of coordination, bringing about expanded execution, decreased power utilization, and benefits with regards to cost and measure. These are vital variables in the plan cycle, and the utilization of SoC is seemingly one of the critical choices in growing continuous implanted systems.[3] SoC can be characterized as a complex coordinated circuit, or incorporated chipset, that joins the really practical components or subsystems of a total final result in a solitary element. At least one programmable processor and frequently a combination of at least one RISC (reduced instruction set computing) control processor and one digital signal processor (DSP) are currently found in the most difficult SoC designs. Peripheral buses, on-chip communications structures, and occasionally a high-speed system bus are also included. An order of on-chip memory units, as well as connections to off-chip memory, is particularly significant for SoC processors. Higher performance and lower energy consumption are two benefits of using a hardware-based accelerating functional unit in the majority of signal processing applications. The design of the SoC includes a number of peripheral processing blocks with analog and digital interfaces for external communication (such as with system buses at the board or backplane level). Chemical processing or MEMS-based (micro electromechanical system) sensors and actuators may be included in future SoC designs.[3,4] All intriguing SoC designs include hardware and software components. Real-time operating systems, programmable processors, and other components of hardware-dependent software are examples of these. Therefore, system level design and engineering, hardware software tradeoffs and partitioning, and software architecture, design, and implementation are all part of the design and use of SoCs[2,5].

Internet of Things (IoT):

The Internet of Things, or IoT, is a network of interconnected computers, mechanical and digital machines, things, animals, or people with unique identifiers and the ability to transfer data over a network without human or computer interaction. Micro services, the internet, wireless technologies, and micro-electromechanical systems (MEMS) have all come together to form the Internet of Things (IoT). Unstructured machine-generated data can now be analyzed for insights that will drive improvements thanks to the convergence of operational technology and information technology.[9]

Wireless:

Most embedded devices operated as independent systems for a long time. However, the situation has altered since the introduction of wireless connectivity. Near field communications (NFC), short-range wireless protocols like Bluetooth and Zigbee, and long-range wireless protocols like WiMAX, LTE, cellular communications, and wireless local area network (WLAN) are all expected to see increased use in the near future. The Internet of Things (IoT) and system-on-chip (SoC) architecture, as well as the utilization of short range protocols and lower power consumption, are the most recent wireless trends for use in embedded systems.[3, 4, 11]

Multi-core Processors

With so many new features being added to embedded systems, high performance is a necessity. As a result, developers are increasingly choosing multicore processors for system design. Mechanicals and packaging are also developing into their own subspecialty, despite the fact that this range of new applications also requires low thermals in a small form factor setting. To meet the ever-increasing demands for performance, chip manufacturers traditionally developed faster single-core processors. The majority of gaming consoles and smart phones of today are multicore, and they are indeed becoming "smarter." [4,12] Multicore processors provide a solution to the requirement to mix new features with legacy code and combine multiple operating environments on the same system. In addition, there is ample opportunity for engineers to relearn about this new design space, architecture, design, programming, debugging, and testing so that they are well informed and aware of the optimal use of new power that a multicore offers. A highly integrated system can be built with real-time software components and human-directed elements running on separate cores in a single processing system, reducing system manufacturing and maintenance costs by eliminating redundant hardware, as opposed to traditional embedded systems with multiple subsystems. In order for the systems to truly have multi-processing capabilities, it may be necessary to migrate them to multi-core platforms in the near future. The widespread use of multi-core processors in embedded computing is still in its infancy. The rate at which the entire ecosystem responds to the standardization of technology in terms of debuggers, RTOSs, compilers, vendors of integrated development environments (IDEs), and programming methodologies will determine how quickly these processors are adopted.

Power consumption

Ultra-low power consumption is another important factor that differentiates products from one another. Applications based on Zigbee require a battery life of more than two years. For this situation, shrewd planning of transmission and gathering will just serve partially. Manufacturers of devices are responsible for reducing power consumption, particularly during radio communication. The gadget ought to stay in rest mode the remainder of the time. During a radio interface, the typical consumption of current is between 30 and 35 mA.

OPERATING SYSTEMS IN EMBEDDED SYSTEM:

Traditionally, embedded systems did not have an operating system (OS) because they only had lightweight control programs that provided limited input/output and memory services. However, as the systems became more complex, it became inevitable to have an OS that provided all traditional functionality, including memory protection, error checking, and transparent inter-process communication, as well as low latency real-time response, a small footprint in both time and space, and low latency real-time response. These applications include communications, consumer electronics, industry controls, automotive electronics, aerospace Emerging multicore also requires multimission, multithread, multiprocess, multiprocessor, and multiboard debugging and must run on open source tool chains like eclipse, among others. The majority of new designs today are moving away from proprietary operating systems and tool chains and are increasingly choosing open source platforms for both development and deployment because cost is their primary competitive advantage. Android is used by many new handhelds and smart phones today.[4,5]

TODAY'S MICROCONTROLLERS AND MICROPROCESSORS

ARM

A family of computer processor-specific reduced instruction set computing (RISC) architectures known as ARM—originally Acorn RISC Machine and later Advanced RISC Machine—is designed to function in a variety of settings. Systems-on-chips (SoCs) with memory, interfaces, and radios are examples of ARM products that use one of these architectures. Cores that carry out this instruction set are also designed by it. The ARM architecture adds support for a 64-bit address space and 64-bit arithmetic, while ARM supports a 32-bit address space. ARM Holdings' core instructions are 32-bit fixed-length, but later architecture versions also support a variable-length instruction set with both 32-bit and 16-bit instructions for increased code density. In terms of production, ARM is the instruction set architecture that is used the most frequently. Currently, there are specialized, older "classic and widely used Cortex cores. [2] The ARM engineering has developed a ton from its most memorable form ARM1 to the most recent ARM11 Processor center. Since its introduction to the market, ARM has produced the product families ARM1, ARM2, ARM3, ARM4 and 5, ARM6, ARM7, ARM8, Strong ARM, ARM9, ARM10, and ARM11. [3] In comparison to typical CISC x86 processors found in the majority of personal computers, processors designed using RISC require fewer transistors. This approach diminishes expenses, intensity and power use. For battery-powered, lightweight devices like smart phones, laptops, tablet computers, and other embedded systems, these characteristics are desirable. ARM could also be a power-efficient option for supercomputers, which use a lot of electricity. [2,3]

AVR

A single-chip, 8-bit RISC microcontroller with a modified Harvard architecture known as an AVR (Advance Virtual RISC). In contrast to the OTP ROM, EPROM, or EEPROM that other microcontroller families used at the time, the AVR was one of the first families to use on-chip flash memory for program storage. A type of Atmel device known as an AVR microcontroller offers distinct advantages over other common chips. The intelligence, memories in RAM and EEPROM, and interfaces to the rest of the system, such as serial ports, disk drives, and display interfaces, are provided by the AVR. There are various packages for AVR microcontrollers, some of which are designed for surface mounting and others for through-hole mounting. From 8 to 100 pins, AVRs are available, but anything above 64 pins must be surface mounted. The majority of individuals begin with a DIL (Dual In Line) 28-pin chip like the ATmega328 or a 40-pin chip like the ATmega16 or ATmega32. [3]

Various processors for different fields are also available in the market,

- DSP Processors
- Media Processor
- Graphic Processor Unit
- Multiprocessor Systems using GPPs
- System-on-a-Chip (SoC).
- Application Specific System Processor (ASSP)
- Application Specific Instruction Processors (ASIPs)
- GPP core(s) or ASIP core(s) on either an Application Specific Integrated Circuit (ASIC) or a Very Large
- Scale Integration (VLSI) circuit.

LATEST PLATFORM IN EMBEDDED SYSTEM

Arduino

The Ivrea Interaction Design Institute was the birthplace of Arduino, a simple tool for quick prototyping aimed at students without prior experience in programming or electronics. From basic 8-bit boards to products for IoT applications, wearables, 3D printing, and embedded environments, the Arduino board began to adapt to new requirements and challenges as soon as it reached a wider audience. Since all Arduino boards are open-source, users can build them on their own and eventually modify them to meet their specific requirements. The software is open-source as well, and users all over the world are contributing to its development. The open-source electronics platform Arduino is built on software and hardware that are simple to use. Arduino has been the brain of thousands of projects over the years, ranging from common household items to intricate scientific instruments. This open-source platform has attracted a global community of makers, including students, hobbyists, artists, programmers, and professionals. Their contributions have resulted in an incredible amount of knowledge that can be of great assistance to novices as well as experts. It is a microcontroller with a single board that was designed to simplify the operation of interactive environments or objects. An open-source hardware board based on an 8-bit Atmel AVR or 32-bit Atmel ARM microcontroller makes up the hardware. The current model has a USB interface, an 8-bit Atmel AVR microcontroller, and 14 digital I/O pins that let you attach multiple extension boards. [8] For physical computing, numerous additional microcontroller platforms and microcontrollers are available. Parallax Fundamental Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, and numerous others offer comparative usefulness. The tangled complexities of microcontroller programming are encapsulated in a user-friendly package by each of these tools. Additionally, working with microcontrollers is made easier with Arduino.[3,8] Arduino has some advantages over other systems for educators, students, and interested amateurs:

• Affordable: When compared to other microcontroller platforms, Arduino boards are relatively inexpensive.
• Cross-platform: The Arduino Software (IDE) is compatible with Windows, Mac OS X, and Linux. The majority of microcontroller systems only support Windows.
• A clear, simple programming environment: The Arduino Software (IDE) is simple enough for novices to use, but it is also flexible enough for more experienced users to take advantage of. Because it is based on the Processing programming environment, it is easy for teachers to use, and students learning to program in Processing will be familiar with how the Arduino IDE works.
• Software that is open source and can be changed: The Arduino software is made available as open source tools, and experienced programmers can change it. C++ libraries can be used to extend the language, and people who want to learn the technical jargon can switch from Arduino to the AVR C programming language.
• Hardware that is open source and can be expanded upon: Because the plans for the Arduino boards are made available under a Creative Commons license, seasoned circuit designers are able to develop their very own version of the module by expanding upon it and making it better. [3,8]

Beagle Bone Black:

The world of low-power, high-performance Beagles, which are small, affordable, and open-source. Putting Android, Ubuntu and other Linux flavors readily available, the Beagle family fires up as high as 1GHz with adaptable fringe interfaces and a demonstrated environment of component rich "Cape" module sheets. The Beagle Bone Black is relatively new to the world of simple microprocessor breakouts, but what it lacks in time to market, it more than makes up for in capability. The current version of the Beagle Bone Black comes from a long line of Beagle Board products; a product with a small form factor, a lot of power, and a lot of room for expansion that lets builders, makers, artists, and engineers make truly innovative projects. In the beginning, the purpose of the Beagle Board family was to provide hobbyists with a relatively low-cost development platform on which to test the powerful new system-on-a-chip (SOC) devices, which were basically able to perform all of the functions of a computer on a single chip. The single-board, credit card-sized Beagle Bone Black computer is reasonably priced. It has a full Linux operating system, a powerful ARM Cortex A8 CPU, and easy access to external sensors. It is a free platform aimed at hobbyists and students looking for a hardware-focused alternative to the Raspberry PI[8].

(IJAER) 2012, Vol. No. 3, Issue No. III, March **ISSN: 2231-5152**

MAJOR APPLICATION AREAS OF EMBEDDED SYSTEMS

• Controls for industries: Controllers with special functions and smart sensors
• Electronics for automobiles: networking, process controls, chassis, body electronics, security, power train, in-vehicle entertainment, and infotainment system electronic control units
• Aerospace / Military: Systems for satellites, radar, sonar, navigation, weather, flight control, and aircraft management
• Telecom: cellular phones, smart devices, networking gateways, switches, routers, and bridges
• Robotics: Multi-function peripherals, printers, scanners, fax machines, copier, storage devices, and smart cards are all examples.
• Electronics for the home: Music players, computerized cameras, blue ray players, set-top boxes, PDAs, videogames, GPS collectors, home machines.
• Medical Technology: Imaging, diagnostic tools, surgical systems, patient monitoring, and electronic stethoscopes
• Automation from afar: Home automation, utility meters, and heating, ventilation, and air conditioning (HVAC) automation are examples of building automation.

CHALLENGES IN EMBEDDED SYSTEMS DESIGN

The design of embedded systems is a crucial component that is rapidly evolving; However, there are a number of obstacles that must be overcome, such as security and safety concerns, the need to update the hardware and software of the system, power consumption, seamless integration, and verification and testing, which is essential to enhancing the system's performance. It is essential to avoid unexpected behavior that could put users in danger when developing an embedded system. It ought to be constructed in such a way that there are no issues with the functionality that saves lives in critical settings. Mobile applications are usually used to control embedded devices, so it's important to make sure there are no chances of data theft or breach.

Manufacturers are now heavily relying on the usage of embedded devices in everything from automobiles to security systems, consumer electronics to smart home solutions, and others. Embedded technologies will continue to grow. Certainly, the embedded system may now be the most significant driver of advancements in device cognition and performance.

Secured embedded systems, software development, and FPGA design services are all made possible by Softnautics' use of the most appropriate technology stacks and the best design practices. Platform enablement, firmware and driver development, OS porting and bootloader optimization, Middleware Integration, and other services across a variety of platforms help businesses construct next-generation systems, solutions, and products.

(IJAER) 2012, Vol. No. 3, Issue No. III, March **ISSN: 2231-5152**

CONCLUSION

Embedded platforms have gradually introduced a sea of technological changes into our day-to-day lives, which, in turn, contribute to making our lives easier and more comfortable through a variety of cutting-edge applications and technologies. To connect real-world devices to embedded system platforms, we must develop standard protocols and communication standards. We can only wish for better embedded system platforms in the future. In a nutshell, the current state of China's science and technology has seen the widespread application of some of the most cutting-edge real-world technology, such as embedded software technology, the most prevalent form of computer time application. The use of embedded software and systems technology will be able to meet the actual application requirements to a large extent because, in the era of rapid development of information technology, a simple computer machine does not meet the demands of people's lives and jobs.

REFERENCES

[1]. Muhammad Ali Mazidi; *The 8051 microcontroller and embedded systems*; 2nd Edition.
[2]. R. Zurawski, The Industrial Communication Systems, Special Issue. IEEE, 93 .
[3]. http://www.raspberrypi.org/
[4]. http://www.beagleboard.org/BLACK
[5]. http://en.wikipedia.org/wiki/Arduino
[6]. http://en.wikipedia.org
[7]. http://en.wikipedia.org/wiki/Atmel_AVR
[8]. www.embeddedcraft.org/ES%20Trends.pdf
[9]. http://en.wikipedia.org/wiki/PIC