

C4-EMBEDDED SYSTEM DESIGN

INTRODUCTION TO EMBEDDED SYSTEM Embedded systems overview An embedded system is nearly any computing system other than a desktop computer. An embedded system is a dedicated system which performs the desired function upon power up, repeatedly. Embedded systems are found in a variety of common electronic devices such as consumer electronics ex. Cell phones, pagers, digital cameras, VCD players, portable Video games, calculators, etc., Embedded systems are found in a variety of common electronic devices, such as: (a) consumer electronics -- cell phones, pagers, digital cameras, camcorders, videocassette recorders, portable video games, calculators, and personal digital assistants; (b) home appliances -- microwave ovens, answering machines, thermostat, home security, washing machines, and lighting systems; (c) office automation -- fax machines, copiers, printers, and scanners; (d) business equipment -- cash registers, curbside check-in, alarm systems, card readers, product scanners, and automated teller machines; (e) automobiles -- transmission control, cruise control, fuel injection, anti-lock brakes, and active suspension

Classifications of Embedded systems

- 1. Small Scale Embedded Systems:** These systems are designed with a single 8- or 16-bit microcontroller; they have little hardware and software complexities and involve board level design. They may even be battery operated. When developing embedded software for these, an editor, assembler and cross assembler, specific to the microcontroller or processor used, are the main programming tools. Usually, `_C` is used for developing these systems. `_C` program compilation is done into the assembly, and executable codes are then appropriately located in the system memory. The software has to fit within the memory available and keep in view the need to limit power dissipation when system is running continuously.
- 2. Medium Scale Embedded Systems:** These systems are usually designed with a single or few 16- or 32-bit microcontrollers or DSPs or Reduced Instruction Set Computers (RISCs). These have both hardware and software complexities. For complex software design, there are the following programming tools: RTOS, Source code engineering tool, Simulator, Debugger and Integrated Development Environment (IDE). Software tools also provide the solutions to the hardware complexities. An assembler is of little use as a programming tool. These systems may also employ the readily available ASSPs and IPs (explained later) for the various functions—for example, for the bus interfacing, encrypting, deciphering, discrete cosine transformation and inverse transformation, TCP/IP protocol stacking and network connecting functions.
- 3. Sophisticated Embedded Systems:** Sophisticated embedded systems have enormous hardware and software complexities and may need scalable processors or configurable processors and programmable logic arrays. They are used for cutting edge applications that need hardware and software co-design and integration in the final system; however, they are constrained by the processing speeds available in their hardware units. Certain software functions such as encryption and deciphering algorithms, discrete cosine transformation and inverse transformation algorithms, TCP/IP protocol stacking and network driver functions are implemented in the hardware to obtain additional speeds by saving time. Some of the functions of the hardware resources in the system are also implemented by the software. Development tools for these systems may not be readily available at a reasonable cost or may not be available at all. In some cases, a compiler or retargetable compiler might have to be developed for these. The processing units of the embedded system

1. Processor in an Embedded System A processor is an important unit in the embedded system hardware. A microcontroller is an integrated chip that has

the processor, memory and several other hardware units in it; these form the microcomputer part of the embedded system. An embedded processor is a processor with special features that allow it to be embedded into a system. A digital signal processor (DSP) is a processor meant for applications that process digital signals. 2. Commonly used microprocessors, microcontrollers and DSPs in the small-, medium- and large scale embedded systems 3. A recently introduced technology that additionally incorporates the application-specific system processors (ASSPs) in the embedded systems. 4. Multiple processors in a system. Embedded systems are a combination of hardware and software as well as other components that we bring together into products such as cell phones, music player, a network router, or an aircraft guidance system. They are a system within another system.

A simple embedded system Building an embedded system we embed 3 basic kinds of computing engines into our systems: microprocessor, microcomputer and microcontrollers. The microcomputer and other hardware are connected via a system bus. A system bus is a single computer bus that connects the major components of a computer system. The technique was developed to reduce costs and improve modularity. It combines the functions of a data bus to carry information, an address bus to determine where it should be sent, and a control bus to determine its operation. The system bus is further classified into address, data and control bus. The microprocessor controls the whole system by executing a set of instructions called firmware that is stored in ROM. An instruction set, or instruction set architecture (ISA), is the part of the computer architecture related to programming, including the native data types, instructions, registers, addressing modes, memory architecture, interrupt and exception handling, and external I/O. An ISA includes a specification of the set of opcodes (machine language), and the native commands implemented by a particular processor. To run the application, when power is first turned ON, the microprocessor addresses a predefined location and fetches, decodes, and executes the instruction one after the other. The implementation of a microprocessor based embedded system combines the individual pieces into an integrated whole which represents the architecture for a typical embedded system and identifies the minimal set of necessary components.

Embedded system life cycle The traditional design approach has been to traverse the two sides of the accompanying diagram separately, that is, Design the hardware components Design the software components. Bring the two together. Spend time testing and debugging the system. The major areas of the design process are Ensuring a sound software and hardware specification. Formulating the architecture for the system to be designed. Partitioning the h/w and s/w. Providing an iterative approach to the design of h/w and s/w. The important steps in developing an embedded system are Requirement definition. System specification. Functional design Architectural design Prototyping. The major aspects in the development of embedded applications are Digital hardware and software architecture Formal design, development, and optimization process. Safety and reliability. Digital hardware and software/firmware design. The interface to physical world analog and digital signals

Figure 1.4: Interfacing to the outside world Embedded applications are intended to work with the physical world, sensing various analog and digital signals while controlling, manipulating or responding to others. The study of the interface to the external world extends the I/O portion of the von-Neumann machine as shown in figure 1.4 with a study of buses, their constituents and their timing considerations. Exemplary applications of each type of embedded system Embedded systems have very diversified applications. A

few select application areas of embedded systems are Telecom, Smart Cards, Missiles and Satellites, Computer Networking, Digital Consumer Electronics, and Automotive