1 Introduction to Embedded Systems

LEARNING OBJECTIVES

- ✓ Learn what an Embedded System is
- ✓ Learn the difference between Embedded Systems and General Computing Systems
- ✓ Know the history of Embedded Systems
- ✓ Learn the classification of Embedded Systems based on performance, complexity and the era in which they evolved
- ✓ Know the domains and areas of applications of Embedded Systems
- ✓ Understand the different purposes of Embedded Systems
- ✓ Analysis of a real life example on the bonding of embedded technology with human life

Our day-to-day life is becoming more and more dependent on "embedded systems" and digital techniques. Embedded technologies are bonding into our daily activities even without our knowledge. Do you know the fact that the refrigerator, washing machine, microwave oven, air conditioner, television, DVD players, and music systems that we use in our home are built around an embedded system? You may be traveling by a 'Honda' or a 'Toyota' or a 'Ford' vehicle, but have you ever thought of the genius players working behind the special features and security systems offered by the vehicle to you? It is nothing but an intelligent embedded system. In your vehicle itself the presence of specialised embedded systems vary from intelligent head lamp controllers, engine controllers and ignition control systems to complex air bag control systems to protect you in case of a severe accident. People experience the power of embedded systems and enjoy the features and comfort provided by them. Most of us are totally unaware or ignorant of the intelligent embedded systems giving us so much comfort and security. Embedded systems are like reliable servants–they don't like to reveal their identity and neither they complain about their workloads to their owners or bosses. They are always sitting in a hidden place and are dedicated to their assigned task till their last breath. This book gives you an overview of embedded systems, the various steps involved in their design and development and the major domains where they are deployed.

1.1 WHAT IS AN EMBEDDED SYSTEM?

An embedded system is an electronic/electro-mechanical system designed to perform a specific function and is a combination of both hardware and firmware (software).

Every embedded system is unique, and the hardware as well as the firmware is highly specialised to the application domain. Embedded systems are becoming an inevitable part of any product or equipment in all fields including household appliances, telecommunications, medical equipment, industrial control, consumer products, etc.

1.2 EMBEDDED SYSTEMS vs. GENERAL COMPUTING SYSTEMS

The computing revolution began with the general purpose computing requirements. Later it was realised that the general computing requirements are not sufficient for the embedded computing requirements. The embedded computing requirements demand 'something special' in terms of response to stimuli, meeting the computational deadlines, power efficiency, limited memory availability, etc. Let's take the case of your personal computer, which may be either a desktop PC or a laptop PC or a palmtop PC. It is built around a general purpose processor like an Intel® Centrino or a Duo/Quad† core or an AMD Turion™ processor and is designed to support a set of multiple peripherals like multiple USB 2.0 ports, Wi-Fi, ethernet, video port, IEEE1394, SD/CF/MMC external interfaces, Bluetooth, etc and with additional interfaces like a CD read/writer, on-board Hard Disk Drive (HDD), gigabytes of RAM, etc. You can load any supported operating system (like Windows® XP/Vista/7, or Red Hat Linux/Ubuntu Linux, UNIX etc) into the hard disk of your PC. You can write or purchase a multitude of applications for your PC and can use your PC for running a large number of applications (like printing your dear's photo using a printer device connected to the PC and printer software, creating a document using Microsoft® Office Word tool, etc.) Now let us think about the DVD player you use for playing DVD movies. Is it possible for you to change the operating system of your DVD? Is it possible for you to write an application and download it to your DVD player for executing? Is it possible for you to add a printer software to your DVD player and connect a printer to your DVD player to take a printout? Is it possible for you to change the functioning of your DVD player to a television by changing the embedded software? The answers to all these questions are 'NO'. Can you see any general purpose interface like Bluetooth or Wi-Fi on your DVD player? Of course 'NO'. The only interface you can find out on the DVD player is the interface for connecting the DVD player with the display screen and one for controlling the DVD player through a remote (May be an IR or any other specific wireless interface). Indeed your DVD player is an embedded system designed specifically for decoding digital video and generating a video signal as output to your TV or any other display screen which supports the display interface supported by the DVD Player. Let us summarise our findings from the comparison of embedded system and general purpose computing system with the help of a table:

General Purpose Computing System	Embedded System
A system which is a combination of a generic hardware and a General Purpose Operating System for executing a variety of applications	A system which is a combination of special purpose hardware and embedded OS for executing a specific set of applications
Contains a General Purpose Operating System (GPOS)	May or may not contain an operating system for functioning

†The illustration given here is based on the processor details available till Dec 2008. Since processor technology is undergoing rapid changes, the processor names mentioned here may not be relevant in future.

Applications are alterable (programmable) by the user (It is possible for the end user to re-install the operating system, and also add or remove user applications)	The firmware of the embedded system is pre-programmed and it is non-alterable by the end-user (There may be exceptions for systems supporting OS kernel image flashing through special hardware settings)
Performance is the key deciding factor in the selection of the system. Always, 'Faster is Better'	Application-specific requirements (like performance, power requirements, memory usage, etc.) are the key deciding factors
Less/not at all tailored towards reduced operating power requirements, options for different levels of power management	Highly tailored to take advantage of the power saving modes supported by the hardware and the operating system
Response requirements are not time-critical	For certain category of embedded systems like mission critical systems, the response time requirement is highly critical
Need not be deterministic in execution behaviour	Execution behaviour is deterministic for certain types of embedded systems like 'Hard Real Time' systems

However, the demarcation between desktop systems and embedded systems in certain areas of embedded applications are shrinking in certain contexts. Smart phones are typical examples of this. Nowadays smart phones are available with RAM up to 256 MB and users can extend most of their desktop applications to the smart phones and it waives the clause "Embedded systems are designed for a specific application" from the characteristics of the embedded system for the mobile embedded device category. However, smart phones come with a built-in operating system and it is not modifiable by the end user. It makes the clause: "The firmware of the embedded system is unalterable by the end user", still a valid clause in the mobile embedded device category.

1.3 HISTORY OF EMBEDDED SYSTEMS

Embedded systems were in existence even before the IT revolution. In the olden days embedded systems were built around the old vacuum tube and transistor technologies and the embedded algorithm was developed in low level languages. Advances in semiconductor and nano-technology and IT revolution gave way to the development of miniature embedded systems. The first recognised modern embedded system is the Apollo Guidance Computer (AGC) developed by the MIT Instrumentation Laboratory for the lunar expedition. They ran the inertial guidance systems of both the Command Module (CM) and the Lunar Excursion Module (LEM). The Command Module was designed to encircle the moon while the Lunar Module and its crew were designed to go down to the moon surface and land there safely. The Lunar Module featured in total 18 engines. There were 16 reaction control thrusters, a descent engine and an ascent engine. The descent engine was 'designed to' provide thrust to the lunar module out of the lunar orbit and land it safely on the moon. MIT's original design was based on 4K words of fixed memory (Read Only Memory) and 256 words of erasable memory (Random Access Memory). By June 1963, the figures reached 10K of fixed and 1K of erasable memory. The final configuration was 36K words of fixed memory and 2K words of erasable memory. The clock frequency of the first microchip proto model used in AGC was 1.024 MHz and it was derived from a 2.048 MHz crystal clock. The computing unit of AGC consisted of approximately 11 instructions and 16 bit word logic. Around 5000 ICs (3-input NOR gates, RTL logic) supplied by Fairchild Semiconductor were used in this design. The user interface unit of AGC is known as DSKY (display/keyboard). DSKY looked like a calculator type keypad with an array of numerals. It was used for inputting the commands to the module numerically.

The first mass-produced embedded system was the guidance computer for the Minuteman-I missile in 1961. It was the '*Autonetics **D-17***' guidance computer, built using discrete transistor logic and a hard-disk for main memory. The first integrated circuit was produced in September 1958 but computers using them didn't begin to appear until 1963. Some of their early uses were in embedded systems, notably used by NASA for the Apollo Guidance Computer and by the US military in the Minuteman-II intercontinental ballistic missile.

1.4 CLASSIFICATION OF EMBEDDED SYSTEMS

It is possible to have a multitude of classifications for embedded systems, based on different criteria. Some of the criteria used in the classification of embedded systems are:

1. Based on generation
2. Complexity and performance requirements
3. Based on deterministic behaviour
4. Based on triggering.

The classification based on deterministic system behaviour is applicable for 'Real Time' systems. The application/task execution behaviour for an embedded system can be either deterministic or non-deterministic. Based on the execution behaviour, Real Time embedded systems are classified into *Hard* and *Soft*. We will discuss about hard and soft real time systems in a later chapter. Embedded Systems which are 'Reactive' in nature (Like process control systems in industrial control applications) can be classified based on the trigger. Reactive systems can be either *event triggered* or *time triggered*.

1.4.1 Classification Based on Generation

This classification is based on the order in which the embedded processing systems evolved from the first version to where they are today. As per this criterion, embedded systems can be classified into:

1.4.1.1 First Generation The early embedded systems were built around 8bit microprocessors like 8085 and Z80, and 4bit microcontrollers. Simple in hardware circuits with firmware developed in Assembly code. Digital telephone keypads, stepper motor control units etc. are examples of this.

1.4.1.2 Second Generation These are embedded systems built around 16bit microprocessors and 8 or 16 bit microcontrollers, following the first generation embedded systems. The instruction set for the second generation processors/controllers were much more complex and powerful than the first generation processors/controllers. Some of the second generation embedded systems contained embedded operating systems for their operation. Data Acquisition Systems, SCADA systems, etc. are examples of second generation embedded systems.

1.4.1.3 Third Generation With advances in processor technology, embedded system developers started making use of powerful 32bit processors and 16bit microcontrollers for their design. A new concept of application and domain specific processors/controllers like Digital Signal Processors (DSP) and Application Specific Integrated Circuits (ASICs) came into the picture. The instruction set of processors became more complex and powerful and the concept of instruction pipelining also evolved. The processor market was flooded with different types of processors from different vendors. Processors like Intel Pentium, Motorola 68K, etc. gained attention in high performance embedded requirements. Dedicated embedded real time and general purpose operating systems entered into the embedded market. Embedded systems spread its ground to areas like robotics, media, industrial process control, networking, etc.

1.4.1.4 Fourth Generation The advent of System on Chips (SoC), reconfigurable processors and multicore processors are bringing high performance, tight integration and miniaturisation into the embedded device market. The SoC technique implements a total system on a chip by integrating different functionalities with a processor core on an integrated circuit. We will discuss about SoCs in a later chapter. The fourth generation embedded systems are making use of high performance real time embedded operating systems for their functioning. Smart phone devices, mobile internet devices (MIDs), etc. are examples of fourth generation embedded systems.

1.4.1.5 What Next? The processor and embedded market is highly dynamic and demanding. So 'what will be the next smart move in the next embedded generation?' Let's wait and see.

1.4.2 Classification Based on Complexity and Performance

This classification is based on the complexity and system performance requirements. According to this classification, embedded systems can be grouped into:

1.4.2.1 Small-Scale Embedded Systems Embedded systems which are simple in application needs and where the performance requirements are not time critical fall under this category. An electronic toy is a typical example of a small-scale embedded system. Small-scale embedded systems are usually built around low performance and low cost 8 or 16 bit microprocessors/microcontrollers. A small-scale embedded system may or may not contain an operating system for its functioning.

1.4.2.2 Medium-Scale Embedded Systems Embedded systems which are slightly complex in hardware and firmware (software) requirements fall under this category. Medium-scale embedded systems are usually built around medium performance, low cost 16 or 32 bit microprocessors/microcontrollers or digital signal processors. They usually contain an embedded operating system (either general purpose or real time operating system) for functioning.

1.4.2.3 Large-Scale Embedded Systems/Complex Systems Embedded systems which involve highly complex hardware and firmware requirements fall under this category. They are employed in mission critical applications demanding high performance. Such systems are commonly built around high performance 32 or 64 bit RISC processors/controllers or Reconfigurable System on Chip (RSoC) or multi-core processors and programmable logic devices. They may contain multiple processors/controllers and co-units/hardware accelerators for offloading the processing requirements from the main processor of the system. Decoding/encoding of media, cryptographic function implementation, etc. are examples for processing requirements which can be implemented using a co-processor/hardware accelerator. Complex embedded systems usually contain a high performance Real Time Operating System (RTOS) for task scheduling, prioritization and management.

1.5 MAJOR APPLICATION AREAS OF EMBEDDED SYSTEMS

We are living in a world where embedded systems play a vital role in our day-to-day life, starting from home to the computer industry, where most of the people find their job for a livelihood. Embedded technology has acquired a new dimension from its first generation model, the Apollo guidance computer, to the latest radio navigation system combined with in-car entertainment technology and the microprocessor based "Smart" running shoes launched by Adidas in April 2005. The application areas and the products in the embedded domain are countless. A few of the important domains and products are listed below:

1. *Consumer electronics:* Camcorders, cameras, etc.
2. *Household appliances:* Television, DVD players, washing machine, fridge, microwave oven, etc.
3. *Home automation and security systems:* Air conditioners, sprinklers, intruder detection alarms, closed circuit television cameras, fire alarms, etc.
4. *Automotive industry:* Anti-lock breaking systems (ABS), engine control, ignition systems, automatic navigation systems, etc.
5. *Telecom:* Cellular telephones, telephone switches, handset multimedia applications, etc.
6. *Computer peripherals:* Printers, scanners, fax machines, etc.
7. *Computer networking systems:* Network routers, switches, hubs, firewalls, etc.
8. *Healthcare:* Different kinds of scanners, EEG, ECG machines etc.
9. *Measurement & Instrumentation:* Digital multi meters, digital CROs, logic analyzers PLC systems, etc.
10. *Banking & Retail:* Automatic teller machines (ATM) and currency counters, point of sales (POS)
11. *Card Readers:* Barcode, smart card readers, hand held devices, etc.

1.6 PURPOSE OF EMBEDDED SYSTEMS

As mentioned in the previous section, embedded systems are used in various domains like consumer electronics, home automation, telecommunications, automotive industry, healthcare, control & instrumentation, retail and banking applications, etc. Within the domain itself, according to the application usage context, they may have different functionalities. Each embedded system is designed to serve the purpose of any one or a combination of the following tasks:

1. Data collection/Storage/Representation
2. Data communication
3. Data (signal) processing
4. Monitoring
5. Control
6. Application specific user interface

1.6.1 Data Collection/Storage/Representation

Embedded systems designed for the purpose of data collection performs acquisition of data from the external world. Data collection is usually done for storage, analysis, manipulation and transmission. The term "data" refers all kinds of information, viz. text, voice, image, video, electrical signals and any other measurable quantities. Data can be either analog (continuous) or digital (discrete). Embedded systems with analog data capturing techniques collect data directly in the form of analog signals whereas embedded systems with digital data collection mechanism converts the analog signal to corresponding digital signal using analog to digital (A/D) converters and then collects the binary equivalent of the analog data. If the data is digital, it can be directly captured without any additional interface by digital embedded systems.

The collected data may be stored directly in the system or may be transmitted to some other systems or it may be processed by the system or it may be deleted instantly after giving a meaningful representation. These actions are purely dependent on the purpose for which the embedded system is designed. Embedded systems designed for pure measurement applications without storage, used in control and instrumentation domain, collects data and gives a meaningful representation of the collected data by means of graphical representation or quantity value and deletes the collected data when new data arrives at the data collection terminal. Analog and digital CROs without storage memory are typical examples of this. Any measuring equipment used in the medical domain for monitoring without storage functionality also comes under this category.

Some embedded systems store the collected data for processing and analysis. Such systems incorporate a built-in/plug-in storage memory for storing the captured data. Some of them give the user a meaningful representation of the collected data by visual (graphical/quantitative) or audible means using display units [Liquid Crystal Display (LCD), Light Emitting Diode (LED), etc.] buzzers, alarms, etc. Examples are: measuring instruments with storage memory and monitoring instruments with storage memory used in medical applications. Certain embedded systems store the data and will not give a representation of the same to the user, whereas the data is used for internal processing.

A digital camera is a typical example of an embedded system with data collection/storage/representation of data. Images are captured and the captured image may be stored within the memory of the camera. The captured image can also be presented to the user through a graphic LCD unit.

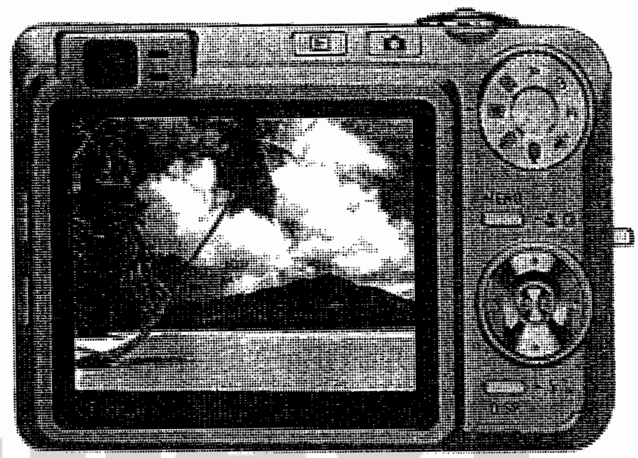

Fig. 1.1 A digital camera for image capturing/storage/display
(Photo courtesy of Casio-Model EXILIM ex-Z850 (www.casio.com))

1.6.2 Data Communication

Embedded data communication systems are deployed in applications ranging from complex satellite communication systems to simple home networking systems. As mentioned earlier in this chapter, the data collected by an embedded terminal may require transferring of the same to some other system located remotely. The transmission is achieved either by a wire-line medium or by a wireless medium. Wire-line medium was the most common choice in all olden days embedded systems. As technology is changing, wireless medium is becoming the de-facto standard for data communication in embedded systems. A wireless medium offers cheaper connectivity solutions and make the communication link free from the hassle of wire bundles. Data can either be transmitted by analog means or by digital means. Modern industry trends are settling towards digital communication.

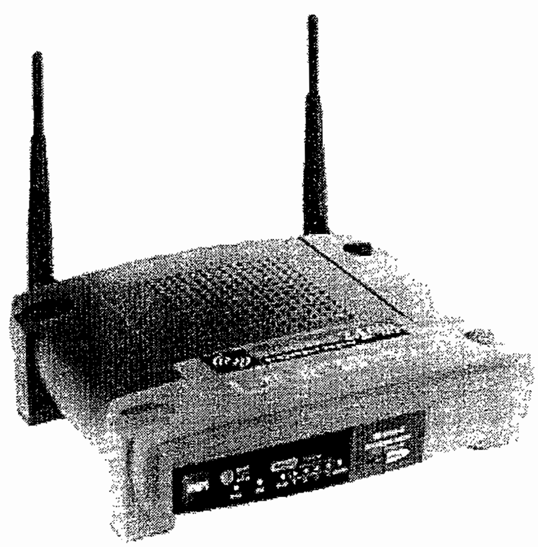

Fig. 1.2 A wireless network router for data communication
(Photo courtesy of Linksys (www.linksys.com). A division of CISCO system)

The data collecting embedded terminal itself can incorporate data communication units like wireless modules (Bluetooth, ZigBee, Wi-Fi, EDGE, GPRS, etc.) or wire-line modules (RS-232C, USB, TCP/IP, PS2, etc.). Certain embedded systems act as a dedicated transmission unit between the sending and receiving terminals, offering sophisticated functionalities like data packetizing, encrypting and decrypting. Network hubs, routers, switches, etc. are typical examples of dedicated data transmission embedded systems. They act as mediators in data communication and provide various features like data security, monitoring etc.

1.6.3 Data (Signal) Processing

As mentioned earlier, the data (voice, image, video, electrical signals and other measurable quantities) collected by embedded systems may be used for various kinds of data processing. Embedded systems with signal processing functionalities are employed in applications demanding signal processing like speech coding, synthesis, audio video codec, transmission applications, etc.

A digital hearing aid is a typical example of an embedded system employing data processing. Digital hearing aid improves the hearing capacity of hearing impaired persons.

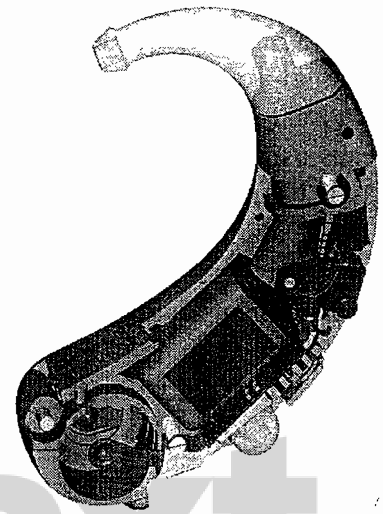

Fig. 1.3 **A digital hearing aid employing signal processing technique**
(Siemens TRIANO 3 Digital hearing aid; Siemens Audiology Copyright© 2005)

1.6.4 Monitoring

Embedded systems falling under this category are specifically designed for monitoring purpose. Almost all embedded products coming under the medical domain are with monitoring functions only. They are used for determining the state of some variables using input sensors. They cannot impose control over variables. A very good example is the electro cardiogram (ECG) machine for monitoring the heartbeat of a patient. The machine is intended to do the monitoring of the heartbeat. It cannot impose control over the heartbeat. The sensors used in ECG are the different electrodes connected to the patient's body.

Some other examples of embedded systems with monitoring function are measuring instruments like digital CRO, digital multimeters, logic analyzers, etc. used in Control & Instrumentation applications. They are used for knowing (monitoring) the status of some variables like current, voltage, etc. They cannot control the variables in turn.

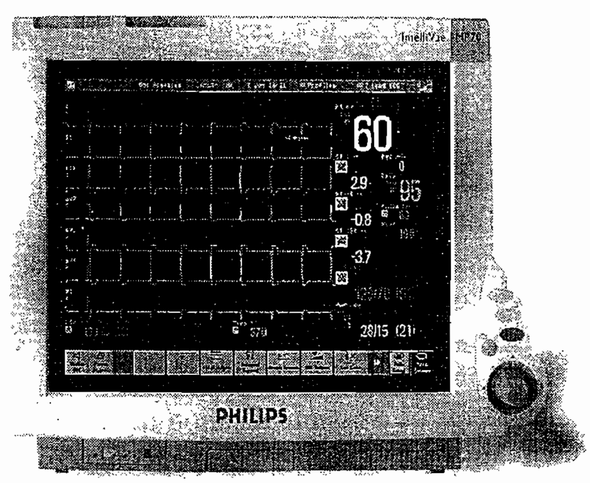

Fig. 1.4 **A patient monitoring system for monitoring heartbeat**
(Photo courtesy of Philips Medical Systems (www.medical.philips.com/))

1.6.5 Control

Embedded systems with control functionalities impose control over some variables according to the changes in input variables. A system with control functionality contains both sensors and actuators. Sensors are connected to the input port for capturing the changes in environmental variable or measuring variable. The actuators connected to the output port are controlled according to the changes in input variable to put an impact on the controlling variable to bring the controlled variable to the specified range.

Air conditioner system used in our home to control the room temperature to a specified limit is a typical example for embedded system for control purpose. An airconditioner contains a room temperature-sensing element (sensor) which may be a thermistor and a handheld unit for setting up (feeding) the desired temperature. The handheld unit may be connected to the central embedded unit residing inside the airconditioner through a wireless link or through a wired link. The air compressor unit acts as the actuator. The compressor is controlled according to the current room temperature and the desired temperature set by the end user.

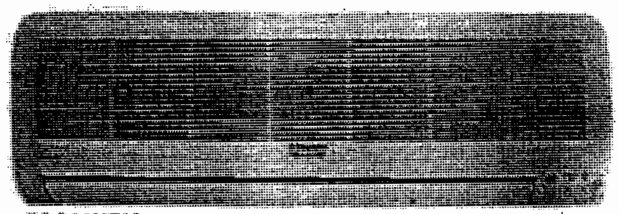

Fig. 1.5 **"An Airconditioner for controlling room temperature. Embedded System with Control functionality"**
(Photo courtesy of Electrolux Corporation (www.electrolux.com/au))

Here the input variable is the current room temperature and the controlled variable is also the room temperature. The controlling variable is cool air flow by the compressor unit. If the controlled variable and input variable are not at the same value, the controlling variable tries to equalise them through taking actions on the cool air flow.

1.6.6 Application Specific User Interface

These are embedded systems with application-specific user interfaces like buttons, switches, keypad, lights, bells, display units, etc. Mobile phone is an example for this. In mobile phone the user interface is provided through the keypad, graphic LCD module, system speaker, vibration alert, etc.

Fig. 1.6 **An embedded system with an application-specific user interface**
(Photo courtesy of Nokia Mobile Handsets (www.nokia.com))

1.7 'SMART' RUNNING SHOES FROM ADIDAS—THE INNOVATIVE BONDING OF LIFESTYLE WITH EMBEDDED TECHNOLOGY

After three years of extensive research work, Adidas launched the "Smart" running shoes in the market in April 2005. The term "Smart Shoe" may sound gimmicky. But adaptive cushioning provided by the shoe makes sense, and the design engineering behind the shoes is very impressive. The shoe constantly adapts its shock-absorbing characteristics to customize its value to the individual runner, depending on the running style, pace, body weight, and running surface. The shoe uses a magnetic sensing system to measure cushioning level, which is adjusted via a digital signal processing unit that controls a motor-driven cable system.

A hall effect sensor is positioned at the top of the "cushioning element", and the magnet is placed at the bottom of the element. As the cushioning compresses on each impact, the sensor measures the distance from top to bottom of mid-sole (accurate to 0.1 mm). About 1000 readings per second are taken and relayed to the shoe's microprocessor. The Microprocessor (MPU) is positioned under the arch of the shoe. It runs an algorithm that compares the compression messages received from the sensor to a preset range of proper cushioning levels, so it understands if the shoe is too soft or too firm. Then the MPU sends a command to a micro motor, housed in the mid-foot. The micro motor turns a lead screw to lengthen or shorten a cable secured to the walls of a plastic-cushioning element. When the cable is shortened, the cushioning element is pulled taut and compresses very little. A longer cable allows for a more cushioned feel. A replaceable 3-V battery powers the motor and lasts for about 100 hours of running.

The Portland, Ore-based Adidas Innovation Team that developed the shoe was led by Christian DiBenedetto. It also included electromechanical engineer Mark Oleson, as well as a footwear developer and two industrial designers. Oleson explains that the team chose a magnetic sensor because it could measure the amount of compression in addition to the time it took to reach full compression. Gathering sensor data, he says, meant little without building a comparative "running context". So one of the first steps in developing the MPU algorithms was building this database. Runners wore test shoes that gathered information about various compression levels during a run. Then the runners were interviewed to learn their thoughts about the different cushion levels. "When the two matched up, that helped validate our sensor," says Oleson.

Adaptations in the cushioning element account for the change of running surface and pace of the runner, and they're made gradually over an average of four running steps. The goal is for the runner not to feel any sudden changes. Adaptations are made during the "swing" phase rather than the "stance" phase of the stride (i.e. when the foot is off the ground). If the shoe's owner prefers a more cushioned or a firmer "ride," adjustments can be made via "+" and "–" buttons that also activate the intelligent functions of the shoe.

LED indicators confirm when the electronics are turned on (The lights do not remain on when the shoes are in use). If the shoes aren't turned on, they operate like old-fashioned "manual" running shoes. The shoes turn off if their owner is either inactive or at a walking pace for 10 minutes.

Source Electronic Design
www.electronicdesign.com/Articles/Index.cfm?AD=1&ArticleID=10113
Re-printed with permission

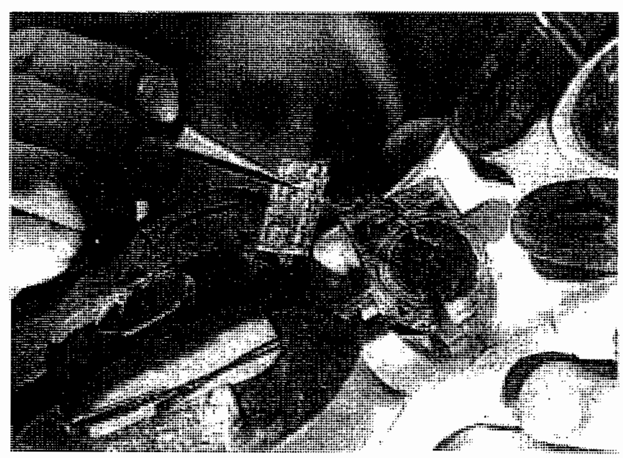

Fig. 1.7 **Electronics-enabled "Smart" running shoes from Adidas**
(Photo courtesy of Adidas – Salomon AG (www.adidas.com))

Summary

- ✓ An embedded system is an Electronic/Electro-mechanical system designed to perform a specific function and is a combination of both hardware and firmware (Software).
- ✓ A general purpose computing system is a combination of generic hardware and general purpose operating system for executing a variety of applications, whereas an embedded system is a combination of special purpose hardware and embedded OS/firmware for executing a specific set of applications.
- ✓ Apollo Guidance Computer (AGC) is the first recognised modern embedded system and 'Autonetics D-17', the guidance computer for the Minuteman-I missile, was the first mass-produced embedded system.
- ✓ Based on the complexity and performance requirements, embedded systems are classified into small-scale, medium-scale and large-scale/complex.
- ✓ The presence of embedded systems vary from simple electronic toys to complex flight and missile control systems.
- ✓ Embedded systems are designed to serve the purpose of any one or a combination of data collection/storage/representation, data communication, data (signal) processing, monitoring, control or application specific user interface.

Keywords

Embedded system : An electronic/electro-mechanical system which is designed to perform a specific function and is a combination of both hardware and firmware

Microprocessor : A silicon chip representing a Central Processing Unit (CPU)

Microcontroller : A highly integrated chip that contains a CPU, scratchpad RAM, special and general purpose register arrays and integrated peripherals

DSP : Digital Signal Processor is a powerful special purpose 8/16/32 bit microprocessor designed specifically to meet the computational demands and power constraints

ASIC : Application Specific Integrated Circuit is a microchip designed to perform a specific or unique application

Sensor : A transducer device that converts energy from one form to another for any measurement or control purpose

Actuator : A form of transducer device (mechanical or electrical) which converts signals to corresponding physical action (motion)

LED : Light Emitting Diode. An output device producing visual indication in the form of light in accordance with current flow

Buzzer : A piezo-electric device for generating audio indication. It contains a piezo-electric diaphragm which produces audible sound in response to the voltage applied to it

Operating system : A piece of software designed to manage and allocate system resources and execute other pieces of software

Electro Cardiogram (ECG) : An embedded device for heartbeat monitoring

SCADA : Supervisory Control and Data Acquisition System. A data acquisition system used in industrial control applications

RAM : Random Access memory. Volatile memory

ADC : Analog to Digital Converter. An integrated circuit which converts analog signals to digital form

Bluetooth : A low cost, low power, short range wireless technology for data and voice communication
Wi-Fi : Wireless Fidelity is the popular wireless communication technique for networked communication of devices

Objective Questions

1. Embedded systems are
   (a) General purpose (b) Special purpose
2. Embedded system is
   (a) An electronic system (b) A pure mechanical system
   (c) An electro-mechanical system (d) (a) or (c)
3. Which of the following is not true about embedded systems?
   (a) Built around specialised hardware (b) Always contain an operating system
   (c) Execution behaviour may be deterministic (d) All of these
   (e) None of these
4. Which of the following is not an example of a 'Small-scale Embedded System'?
   (a) Electronic Barbie doll (b) Simple calculator
   (c) Cell phone (d) Electronic toy car
5. The first recognised modern embedded system is
   (a) Apple Computer (b) Apollo Guidance Computer (AGC)
   (c) Calculator (d) Radio Navigation System
6. The first mass produced embedded system is
   (a) Minuteman-I (b) Minuteman-II
   (c) Autonetics D-17 (d) Apollo Guidance Computer (AGC)
7. Which of the following is (are) an intended purpose(s) of embedded systems?
   (a) Data collection (b) Data processing (c) Data communication
   (d) All of these (e) None of these
8. Which of the following is an (are) example(s) of embedded system for data communication?
   (a) USB Mass storage device (b) Network router
   (c) Digital camera (d) Music player
   (e) All of these (f) None of these
9. A digital multi meter is an example of an embedded system for
   (a) Data communication (b) Monitoring (c) Control (d) All of these
   (e) None of these
10. Which of the following is an (are) example(s) of an embedded system for signal processing?
   (a) Apple iPOD (media player device) (b) SanDisk USB mass storage device
   (c) Both (a) and (b) (d) None of these

Review Questions

1. What is an embedded system? Explain the different applications of embedded systems.
2. Explain the various purposes of embedded systems in detail with illustrative examples.
3. Explain the different classifications of embedded systems. Give an example for each.