1

Embedded Systems —An Introduction

Learning Outcomes

- Basic Idea on System
- Definition of Embedded Systems
- Characteristics of Embedded Systems
- Challenges in Designing an Embedded System
- Categorization of Embedded Systems
- "Examples of Embedded Systems"
- Recap
- Quiz

Embedded Systems are available everywhere in this modern world. This chapter will touch on all basic aspects of understanding an Embedded System.

1.1 BASIC IDEA ON SYSTEM

What is a System? A system can be defined as simple as "It can accept some input, analyze and then it should give us the output that the system is meant for or it should drive the next piece of machine connected to it".

1.2 EMBEDDED SYSTEMS—DEFINITIONS

Definition of Embedded Systems can now be seen. Many people have given many definitions all over the world. If internet is surfed for definition of an embedded system one will get enormous amount of results getting displayed. Few of them are picked here to add clarity and finally embedded system can be defined in a smooth way.

According to Wayne Wolf,

"An Embedded System is a computing system other than desktop computers". This looks pretty simple and other definitions are as follows:

- An embedded system is the one that has computer hardware with software embedded in it as one of its most important components.
- It is a device that includes a programmable computer but is not itself intended to be a general purpose computer.

Embedded System can be well defined by taking couple of classical examples.

First, an air conditioner is taken for understanding. What does an air conditioner do? The temperature is set as per requirement, say 20°C. There may be variations in external temperature and that will also reflect in the room air conditioner is fitted. But however the external temperature varies, the AC machine facilitates user with cool atmosphere (i.e., 20 °C) inside the room as per requirement. What is the action taken?

Consider a second example of the pace maker. Its work is to trigger the heart beat if at all heart is getting into trouble. How is this done?

Answers for both the questions are the same.

When looking into the definition of an Embedded System, one can get answer for above quoted cases.

- An electronic controller built into the application, continuously monitors the process variables and ensures that the Process Variable (PV) does not change; in the event of a change the controller generates a counteracting signal applied to the application so that the deviated PV is brought to its normal operating value. This could define embedded systems clearly!

So here it is made very clear. In air conditioner, temperature is the process variable. A controller inside will keep on monitoring the process variable. If at all the room temperature changes due to variation in external temperature, controller will take a counter acting signal and PV (temperature) will be brought to required range.

Second case, controller inside a pace maker will keep monitoring the heart beat count. If it is getting low, immediately a counter acting action will be taken and it will boost up the heart.

Food for brain! Is laptop an embedded system?—This question will be answered shortly!

1.3 CHARACTERISTICS OF EMBEDDED SYSTEMS —AN OVERVIEW WITH EXAMPLES

When provided with a system, it should be identified, if it is an Embedded System. Certain common characteristics are there for all Embedded Systems. Having able to understand the characteristics, embedded systems can be spotted easily.

- Single Functioned

- Tightly Constraint
- Real Time and Reactive
- Complex Algorithms
- User Interface
- Multi Rate

Each of the above characteristics are discussed below in detail.

1. Single Functioned

An Embedded System can execute a specific function repeatedly i.e., dedicated function. As an example, Air conditioner will be cooling the room. Cooling is its dedicated functionality and it cannot be used for any other purposes. AC can't be used for making calls. Likewise mobile phone is an Embedded System that can be used to make and receive calls and it can't be used for controlling room temperature.

Consider the list of embedded systems that are being used every day.

1. Pager
2. Microwave oven
3. Mobile phone
4. ATMs
5. Car braking systems
6. Automobile cruise controllers
7. Pace makers
8. Modem
9. Network cards and many more

Fig. 1.1: Few applications of embedded systems (all single functioned)

From the examples quoted, it is understood about single functioned behaviour of Embedded Systems.

Is Laptop an Embedded System -> No, since it can be used for different purposes. It can play media players and at the same time, laptop can be used as a gaming machine. And the next day it can be used for typing data. So it is multifunctional and it can't be an Embedded System.

2. Tightly Constraint

Whatever system is being designed, they have constraints. Embedded Systems are also tightly constraint in many aspects. Few aspects are being analyzed here.

1. Manufacturing Cost
2. Performance
3. Size
4. Power

The above four parameters decide the success of Embedded System.

Consider buying a mobile phone as an example. If the mobile phone costs in lakhs, will it be bought? (Instead Landline phone would be preferred). Next scenario, the mobile phone that is bought, if it takes 1/2 an hour for making a call and if it also hangs frequently, will it be opted? (No Way!). Third point if the phone is weighing 3 kgs, will it be preferred? Finally coming to power criteria. All Embedded Systems are almost battery operated. And it is mobile as well! So it should be capable of retaining the charge for some reasonable amount of time. Else the battery will drain faster and one has to keep charger handy all the time. So it is very important to have this constraint in mind when designing an embedded system.

3. Real Time and Reactive

What is real time? —A nice question to start with!

A definition can be given through an example here. Take an instance of travel in BMW car. (Great feel it would be). (The braking system is an embedded system). And unfortunately a lorry is coming opposite to the car... The driver is applying brake there!. What would be the action required? It should immediately stop the car right. This is a real time and reactive behaviour. The brake may be applied at any point in time. And the vehicle should be stopped immediately at the instance of applying brake. It is never known when brake has to be applied, so the system should be ready to accept the input at any time and should be ready to process it.

So keeping above example in mind and defining Real Time, **it is logical correctness of the operation in deterministic deadline** (The vehicle should be stopped immediately, which means as logical correctness of the operation in deterministic deadline).

Few examples can be spotted for Real time and Reactive behaviour of an Embedded System:

(*a*) Pace Maker's action

(*b*) Flights Landing Gear Control

(*c*) ECG Machines output

And so on … Many examples could be cited here!

4. Complex Algorithms

The processor inside the embedded system should perform operations that are complex in nature. An example is digital camera. It is used to take colour photographs, motion pictures, black and white pictures, etc. It needs to pull in lots of complex algorithms for performing all the above mentioned operations. So as a point to conclude, every Embedded System will have lots of complex algorithms running inside it.

5. User Interface

Here too with an example the concept can be explained. NOKIA mobile phones are very big hit in market right, Why? What is the reason? Is that because other mobile did not perform well? No, is the answer. Nokia had excellent and simple user interface. Calls can be made and received very easily. Typing SMS is also easier… So it has been read by the people very well.

So designing system with easier and comfortable interface is most important. Also it should have options required for the operation of the device. Example is **ATM machine**; it has got comfortable interfaces and options. Keep it in mind and design the system.

6. Multi Rate

Embedded Systems need to control and drive certain operations at one rate and certain other operations at different rate. Example can be Digital Camera. It is used to take pictures which are still. Also it is capable of shooting video. So it has to be capable of driving the first operation from a speed different than the second one.

A Small Recap: Please do not forget the definition of Real Time; down the line it will be needed.

1.4 CHALLENGES IN DESIGNING AN EMBEDDED SYSTEM

First and foremost problem in designing an Embedded System is "Very Less Availability of Tools and Debuggers".

Other than the point quoted, there are several other challenges.

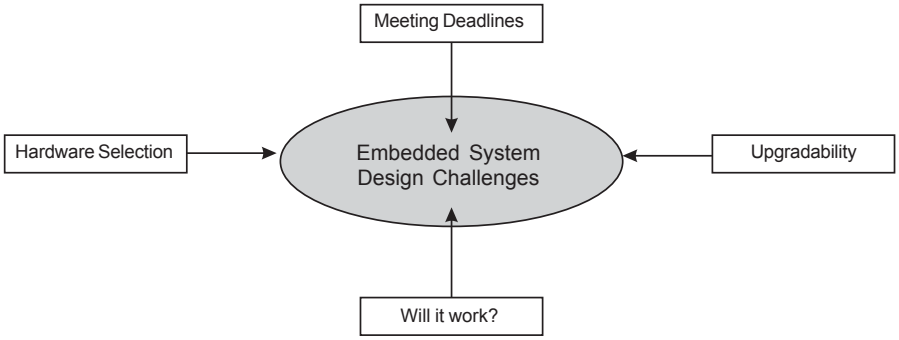

Fig. 1.2: Challenges in embedded system design

Figure 1.2 diagrammatically represents the challenges. Reader will be exposed to all these challenges with some relevant examples.

1. Meeting Deadlines

How can the deadline be met that is meant for the product? Meeting deadline accurately will need high speed hardware. Increasing hardware components with quality would increase the cost of the product. This is the first challenge in front of designers.

2. Hardware Selection

Embedded Systems never had a luxury of having much hardware. Taking memory into consideration first, Embedded Systems will have very little inbuilt memory. Adding more memory of smaller size will increase cost factor. So keep memory only as much as needed. It can have an expansion slot for the system, if user is willing to expand memory, who bothers, let user expand.

Coming to processor selection, if a very high speed processor is selected, it would end up in draining the battery at the earliest. But it can't be compromised with speed also. So select a processor that perfectly fits in with requirement. Too high speed processor would cost more and can drain battery also.

3. Is it upgradable and maintainable?

Assume that a mobile phone has been designed and it is released in the market. But after reaching to the people, the product was found with problems in one or

two aspects. The developer would know that problem and it can be fixed. But how will it reach the phone that had already reached to the public?

So it must be supporting with upgradation of versions of software for it. Keep this in mind that the product should be upgradable with the same hardware!

Secondly, when develop software for embedded systems, it should be kept in mind on maintainability. The code should not be just written in such a way that only developer who developed it can understand. It should be understandable for other engineers also. Other engineers should also be able to understand and fix bugs in the code if any, if need be.

4. Will it work?

Nice Question. Isn't it? Yeah. Please ensure if the system that has been designed is really working fine. How can it be ensured? Through rigorous testing it is possible; it needs to be proceeded with testing in many ways. First can be Unit Testing, next stage is Sanity Testing and the third stage can be Regression testing. Also even if the product has entered, it has to be constantly monitored. If any customer complaint rises, that bug has to be looked into and has also to be fixed. And more importantly, the bug that is fixed should not introduce any new bugs.

Let's now get to know about the categorization of the embedded systems!

1.5 CATEGORIZATION OF EMBEDDED SYSTEMS

Embedded Systems can be categorized based on the complexity in building, cost factors, purpose of the system, tools and other related environment availability, etc. Keeping these points, Table 1.1 has been framed and it has dealt with categories. Broadly one can classify embedded system into any of these,

1. Small Scale Embedded Systems,
2. Medium Scale Embedded Systems, and
3. Sophisticated Embedded Systems.

Table 1.1: Categorization of Embedded Systems

	Small Scale Embedded System	Medium Scale Embedded System	Sophisticated Embedded System
Processor	Here it can be 8 bit or 16 bit processor. (it can't do computationally intensive process)	It can be a 16 bit or 32 bit processor here. (think of little complex and intensive process with this processor)	PLA, PAL, FPGA and ASIC will fall in this category. (These are high end design elements that can be used to do wonders)

Contd...

Hardware Complexity	Very little complexity will be visualized here.	We will have to face more complexity in terms of peripheral additions, interfaces etc.	Highly Complex. (designers need enormous expertise to do proceed with this)
Software Complexity	No complexity will be there in this coding. Because the device is not meant for performing complex functionalities.	There will be complexity added up. This will have few functions and code might be lengthy.	Yeah. Most Complex. Designer needs to be a master to work on the code.
Power	Battery operated	Battery operated	Can be Battery or Live Source based on the application.
Tools Availabiltiy	Can be programmed in simple environment. So no much research on tools is required here.	Here we will have to use Debugger, Compiler, IDE etc., as the task goes slightly cumbersome.	Designer needs sophisticated environment here.
Examples	Calculator can be the simplest example. Stepper motor controller can be added to the list.	Washing Machine, Microwave Oven, and Vending machine.	Flight Landing Gear Systems, Car Braking Systems, Military Applications, Robots.

1.6 EXAMPLES OF EMBEDDED SYSTEMS

Coming to examples, everything around us can be taken! Yeah, each and everything these days are embedded systems. Let's take this way.

From the time of getting up in the morning,

1. Digital Water Heater! (After bath need to breakfast!). Here temperature is the process variable and it is the input to be set by user. Controller will take care on the controlling action and will take care of the heating process.
2. Microwave Oven (coffee after breakfast). Here again temperature will be the process variable. Same controlling action will be taken.
3. Braking System, Music Player, Tire Pressure Monitoring System, Airbags, Power Windows and GPS (Many more embedded systems are there inside … only 1% is quoted, workplace has been reached now. Car braking system can be an instance; it shows the real time and reactive behaviour. Brake would be applied at any point in time, but still it would stop the car.)
4. All gym equipments from treadmill to cycling equipments are embedded systems.

5. Video games, all digital gaming machines, I pod, MP3/MP4 players, Xbox and what not!

POINTS TO REMEMBER

1. An Embedded System is single functioned and AC can be remembered as a simple example.
2. Embedded Systems are Real time systems which is reactive in nature.
3. Many design challenges are associated with making an embedded system, including cost, power etc.
4. Embedded Systems are classified into three major divisions—Small scale, medium scale and large scale Embedded Systems.

Review Questions

1. What is an Embedded System? Give an example.
2. Embedded Systems are quoted as single functioned systems. Justification is required in this case.
3. Define Real time.
4. Throw an example for real time and reactive Embedded System.
5. What are the major categories of the Embedded Systems? Give an example for each division.
6. Is LCD projector an Embedded System? Please justify.

1.7 QUIZ

1. Pick odd one out! (Embedded is the clue)
   (*a*) Laptop (*b*) Projector
   (*c*) Mobile phone (*d*) MP3 player
2. Some of the important characteristics expected in consumer electronics products are .....
   (*a*) Recovering from failures
   (*b*) Low cost
   (*c*) Performance
   (*d*) Low unit cost, low power consumption and smaller product size
3. One of the most important characteristics of Embedded Systems for automotive industry (with respect to the inmates of the vehicle) is .....
   (*a*) Recovering from failures

(*b*) Low cost

(*c*) Safety

(*d*) Low unit cost, low power consumption and smaller product size

4. ..... manages resources like CPU time and memory of an Embedded System.

   (This is a general question, try to answer this.)

5. Embedded System can do multiprocessing — True or False.
6. What is Real time? Please give an example.
7. Give an example of multi rate characteristic of an Embedded System.
8. Embedded Systems are almost battery operated—True or False.
9. Mobile phone is not an Embedded System—True or False.

Answers for Quiz

1. Laptop
2. Low unit cost, low power consumption and smaller product size
3. Safety
4. Operating system
5. False
6. Braking, Pace maker's behaviour can be a good example here.
7. A digital camera can work on a black and white image, colour image, video and audio. Processing is different in all these which is referred as multi rate.
8. True
9. False (May be in future, Mobile phones would do multiple things and can break this question).