School: IQ Academy

Topic: Embedded Systems

Level: Junior and Senior

Time: 1hour 30mins

Aims: At the end of the training, students should be able to:

- Define embedded systems
- Different types of embedded systems

Introduction to Embedded Systems

A combination of computer hardware and software where software is embedded into hardware is called an embedded system. It is an electronic system that can be programmed based on our needs. There are a set of protocols where we can organize the way of working and controlling the system. The hardware where the software is embedded is built with the help of a microcontroller. The hardware is comprised of a user interface, which includes the display, input, and/or output interfaces and memory of the system. When we consider types of Embedded Systems, power supply, processor, and communication ports are needed for the system to function properly.
Types of Embedded Systems

We can classify embedded systems based on performance and functional requirements and based on the performance of the microcontroller. The microcontroller is important for an embedded system as hardware working is fully dependent on the microcontroller. It takes care of the communication and the processors needed for the system. The requirements of the user depend on the function of the embedded system. Let us have a look at the types.

Based on performance and functional requirements:

1. Stand-alone Embedded Systems

This type of embedded systems, as the name suggests, does not require a host system like a computer or a processor as it works by itself and displays data on the connected device or make necessary changes on the device. Input data is taken from the ports as analog or digital signals, and processing is done in the port itself. The result after proper calculation and conversion is displayed through a connected device. This result may either control or drive or display the device depending on the data. These systems offer flexibility and efficiency even though they work alone. Various examples are washing machines, mobile phones, or any systems that work alone without a computer's help.
2. Real-time Embedded Systems

When an output is required at a particular time, real-time embedded systems can be used. When a task has to be completed on time or a deadline for a project, this system comes as a savior. An

external environment is controlled with the help of computer systems and connected through sensors or any other output/input interfaces. We can schedule the output either through a static or dynamic manner. There are two types under this category. They are soft and hard real-time embedded systems.

Soft Embedded Systems: These systems consider processes as the main task and manage the task completely. Deadlines are not considered as a priority, and even if the deadlines are missed, missing the processes should not happen in this system. In a computer system, the sound system is an example of a soft system. But it should be noted that deadlines should not always be missed as it results in the system's degradation.

Hard Embedded Systems: These systems consider timelines as the deadline, and they should not be missed in any case. Also, even if there are any delays, they should also be time-bound in the system. There is no permanent memory, and hence the processes should be done properly for the first time itself. The best example of a hard embedded system is an aircraft control system. The timeline should not be missed as well as the processes.

3. Network Embedded Systems

When a program is running inside another device, a network is formed. This is called network embedded systems, where a microprocessor or a controller controls the running program. A network is related to this system, and they can be either LAN or WAN. It is not necessary that the connection should be wired or wireless. This category can be considered the fastest growing in the embedded systems due to flexibility and connection. Also, there is a web server where the connection is based upon the web browser. All the network is controlled and accessed with the help of a web browser. Security systems in any office or tech park are examples of network embedded systems where all the connections are made through a common network and controlled under one umbrella.

4. Mobile Embedded Systems

All the devices that are portable and working with an embedded system is a mobile embedded system. Though there is a limitation of memory and functionality, its portability and handy systems are useful for all people. The best example that we can connect easily is mobile phones, laptops, and calculators.

Based on the performance of the microcontroller:

Small Scale Embedded Systems: 8 bit or 16-bit microcontrollers are used to design these and work with the help of batter in the system. Several programming tools are used to develop small scale embedded systems. The hardware is very small, and the processor is slow. The memory is also less. The codes for developing these embedded systems can be written with the help of any IDE.

Medium Scale Embedded Systems: 16 bit or 32-bit microcontrollers are used to develop medium systems. Also, these can be developed with DSPs or RISCs. Hardware and software

functionalities are complex, and several coding languages can be used as programming tools. As small-scale systems, an IDE is required for medium scale systems also. We can use medium-scale systems in high-end applications with large memory and processing data.

Sophisticated Embedded Systems: The most complex embedded system with all the difficult complexities of hardware and software that makes the system useful for all is called sophisticated embedded systems. These systems require registers of huge memory, scalable processors, and IPs to work well in any environment. They are used in systems with graphical screens, touchpads, and cutting-edge options where software and hardware are equally needed for performance.

Conclusion

It is understood that embedded systems are known for their flexibility and reliability of the system. And without embedded systems, we cannot imagine the real world now. Understanding the types will help us gain more insight into the embedded systems and real-time applications in the world.