

6. Explain about distributed embedded architecture. (6) [CO4-L1]



- A distributed embedded system can be organized in many different ways, but its basic units are the PE and the network as illustrated in Figure 4.4. A PE may be an instruction set processor such as a DSP, CPU, or microcontroller, as well as a nonprogrammable unit such as the ASICs used to implement PE 4. An I/O device such as PE 1 (which we call here a **sensor** or **actuator**, depending on whether it provides input or output) may also be a PE, so long as it can speak the network protocol to communicate with other PEs.
- The network in this case is a bus, but other network topologies are also

106

VII SEM

possible. It is also possible that the system can use more than one network, such as when relatively independent functions require relatively little communication among them. We often refer to the connection between PEs provided by the network as a **communication link**.

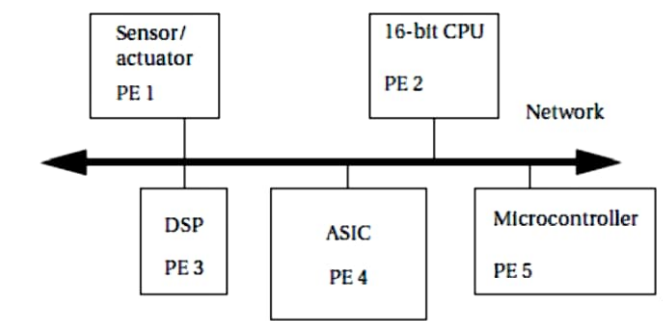


Fig An example of a distributed embedded system.

- The system of PEs and networks forms the **hardware platform** on which the application runs.

□ In particular, PEs do not fetch instructions over the network as they do on the microprocessor bus. We take advantage of this fact when analyzing network performance the speed at which PEs can communicate over the bus would be difficult if not impossible to predict if we allowed arbitrary instruction and data fetches as we do on microprocessor buses.

Distributed

- Building an embedded system with several PEs talking over a network is definitely more complicated than using a single large microprocessor to perform the same tasks. So why would anyone build a distributed embedded system? All the reasons for designing accelerator systems also apply to distributed embedded systems, and several more reasons are unique to distributed systems.
- In some cases, distributed systems are necessary because the devices that the PEs communicate with are physically separated. If the deadlines for processing the data are short, it may be more cost-effective to put the PEs where the data are located rather than build a higher-speed network to carry the data to a distant, fast PE.
- An important advantage of a distributed system with several CPUs is that one part of the system can be used to help diagnose problems in another part. Whether you are debugging a prototype or diagnosing a problem in the field, isolating the error to one part of the system can be difficult when everything is done on a single CPU. If you have several CPUs in the system, you can use

one to generate inputs for another and to watch its output.

Network Abstractions

- Networks are complex systems. Ideally, they provide high-level services while hiding many of the details of data transmission from the other components in the system. In order to help understand (and design) networks, the International Standards Organization has developed a seven-layer model for networks known as Open Systems Interconnection (OSI) models [Sta97A]. Understanding the OSI layers will help us to understand the details of real networks.
- The seven layers of the **OSI model**, shown in Figure 4.5, are intended to cover a broad spectrum of networks and their uses. Some networks may not need the services of one or more layers because the higher layers may be totally missing or an intermediate layer may not be necessary.
- However, any data network should fit into the OSI model. The OSI layers from lowest to highest level of abstraction are described below.
 - **Physical:** The physical layer defines the basic properties of the interface between systems, including the physical connections (plugs and wires), electrical properties, basic functions of the electrical and physical components, and the basic procedures for exchanging bits.
 - **Data link:** The primary purpose of this layer is error detection and control across a single link. However, if the network requires multiple hops over several data links, the data link layer does not define the mechanism for data integrity between hops, but only within a single hop.
 - **Network:** This layer defines the basic end-to-end data transmission service. The network layer is particularly important in multi hop networks.
 - **Transport:** The transport layer defines connection-oriented services that ensure that data are delivered in the proper order and without errors across multiple links. This layer may also try to optimize network resource utilization.

| | |
|--------------|----------------------------|
| Application | End-use interface |
| Presentation | Data format |
| Session | Application dialog control |
| Transport | Connections |
| Network | End-to-end service |
| Data link | Reliable data transport |
| Physical | Mechanical, electrical |

Fig The OSI model layers.

- Session: A session provides mechanisms for controlling the interaction of end user services across a network, such as data grouping and check pointing.
 - Presentation: This layer defines data exchange formats and provides transformation utilities to application programs.
 - Application: The application layer provides the application interface between the network and end-user programs.
- ☐ Although it may seem that embedded systems would be too simple to require use of the OSI model, the model is in fact quite useful.
- ☐ Even relatively simple embedded networks provide physical, data link, and network services.
- An increasing number of embedded systems provide Internet service that requires implementing the full range of functions in the OSI model.