BITS Pilani

Pilani|Dubai|Goa|Hyderabad

Network Embedded Applications

Prof. Manoj S Kakade
Dept of EEE

innovate achieve lead

BITS Pilani

Pilani|Dubai|Goa|Hyderabad

Introduction

CS 1 : (Introduction to Networked Embedded Systems)

1	➢ Introduction to the course – evaluation components ➢ Introduction to various classes of Networked Embedded Applications ➢ Multi-processing, Distributed and Networked Systems ➢ IoT Vs NEA ➢ Buses Vs Networks	

Welcome Note

- Welcome to the course Network Embedded Applications.
- This course forms the basis of IoT, Smart Buildings, Industrial Network, Driver less cars etc.
- This course covers both networks and the embedded technology is considerable depth.

Course Objectives

CO1	To get an overview of various network embedded applications, architectures, available protocols
CO2	Understand the application of Wireless Sensor Networks in depth
CO3	Understand the application of Industrial Networks in depth
CO4	Understand the application of Automotive Networks in depth
CO5	Develop Deeply Embedded Systems and IoT Applications

Learning Outcomes

LO1	Design and Implementation of Wireless Sensor Networks
LO2	Design and Implementation of Industrial Networks
LO3	Design and Implementation of Automotive Networks
LO4	Design and Implementation of end-to-end IoT Networks

Books

Text Book

1. R.Zurawski, Network Embedded Systems, CRC press, 2009

Reference Books

1. G.Pottie, W.Kaiser, Principles of Embedded Networked System Design
2. IEEE Journals and Transactions.
3. IETF(Internet Engineering Task force) Drafts and RFCs (Request for Comments)
4. ACM (Association for computer Machinery) Transactions
5. Elsevier Journals

In This Course

- M1. Introduction To Networked Embedded Systems
- M2. Wireless Sensor Networks
- M3. Industrial Control Networks
- M4. Automotive Electronics

Evaluation Components

- EC1 - 30% (Assignment/Quiz(es))
  - Quiz 1 – 5%
  - Situated Learning Assignment – 10%
  - Assignment – 15%
- EC2 - 30% (Mid-semester Examination) – Closed Book
- EC3 - 40% (Comprehensive Examination) – Open Book

Introduction to Network Embedded Application

- Embedded System everywhere..

- **An embedded system is some combination of computer hardware and software, either fixed in capability or programmable, that is designed for a specific function or for specific functions within a larger system**

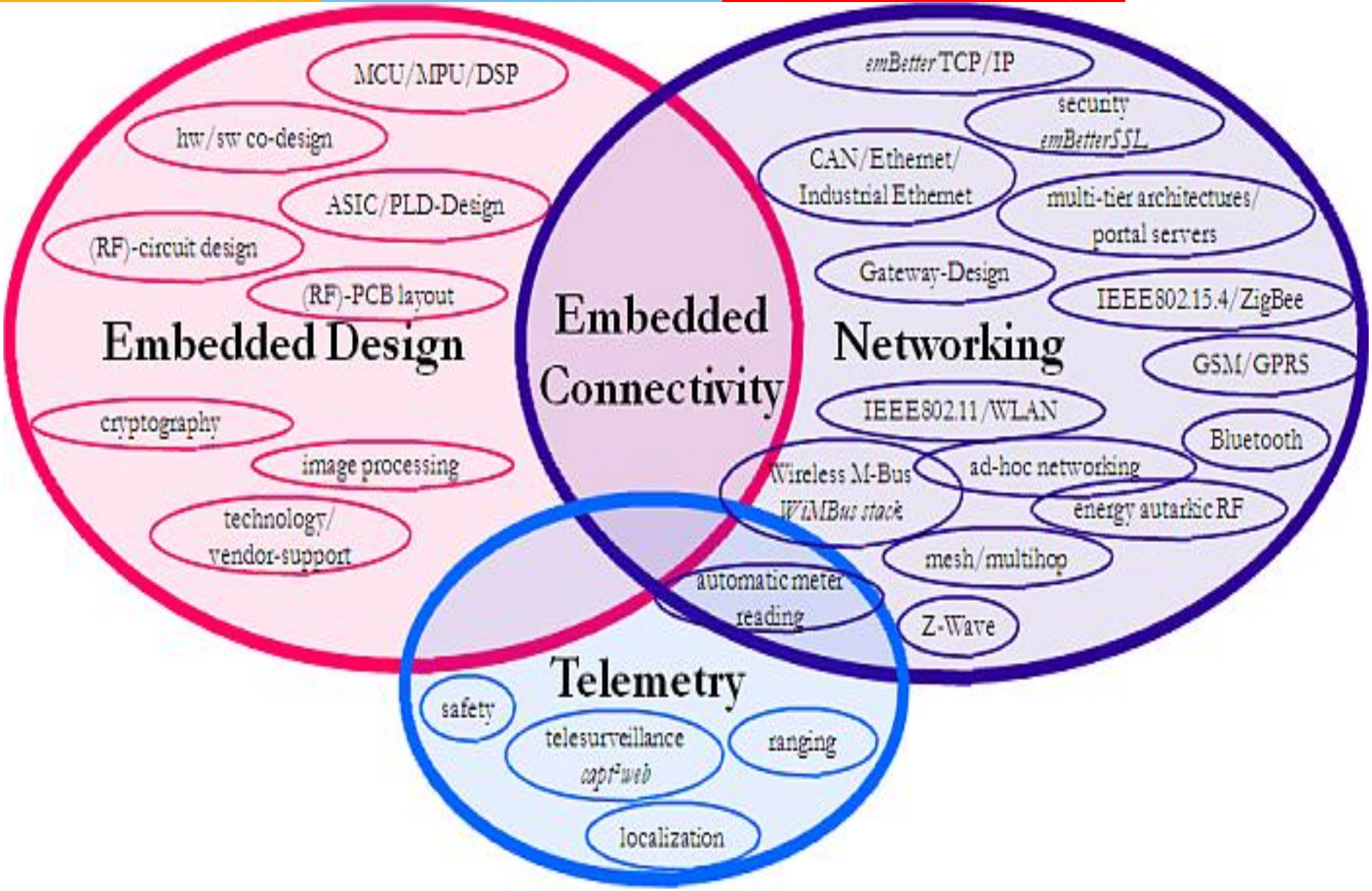
Introduction to Network Embedded Application

- Embedded System Advantages
  - Single Specific Function
  - Small Size
  - Reduce Cost
  - Portability
  - Low Power Requirement
  - Real Time Response etc
- Embedded System Disadvantages:
  - Difficult to Configure
  - Scalability
  - Limitation hardware
    - CD Player/Music System

Introduction to Network Embedded Application

- Future of Embedded Systems: Ubiquitous /Pervasive Computing:
- **Ubiquitous computing** (or "ubicomp") is a concept in software engineering and computer science where **computing** is made to appear **anytime and everywhere**.
- Ubiquitous Computing is the creation of smart products that are Connected, making Communication and the Exchange of data easier and Less obtrusive.
- Contributes to Smart Home / Smart Vehicle / Car, Smart City /Smart Office /Smart Hospital ….

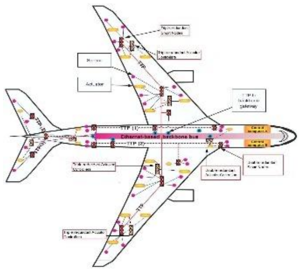
Introduction to Network Embedded Application



Networked Embedded Systems

- Embedded System that are connected to Network for the resources are **Networked Embedded System** (NES).
- Collection of spatially & functionally embedded nodes connected via wireline / wireless communication infrastructure / protocols interacting with the environment via
  - Sensors, Actuators
- Interact with each other
- Master node - overall coordination – working towards a single goal

Introduction to Networked Embedded Application



Automotive Networks

Intelligent Highways

Collaborative Robots

Wildlife Monitoring

- Large scale, ad hoc networks
- Heterogeneous node functionality: sensors, intelligent cameras, smart appliances
- Limited resources: CPU, memory, bandwidth, energy
- Wired/ Wireless Communication
- Volatile, possibly mobile

Networked Embedded Systems

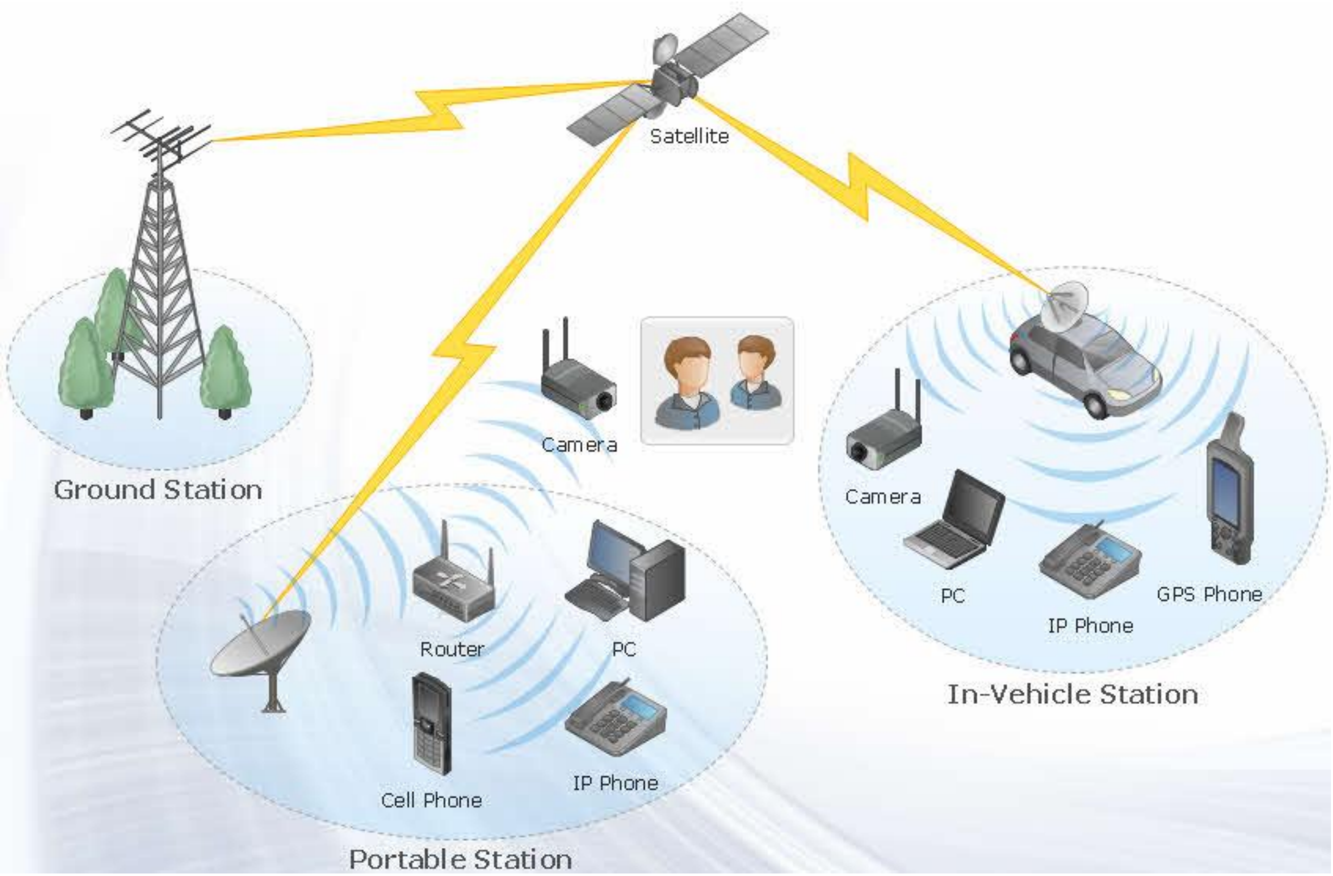
- Deeply Embedded Systems
- Sensor networks
- Distributed control applications
- Industrial Automation – Industry 4.0
- Automotive Electronics + ITS
- Ubiquitous computing environments
- IoT

Introduction to Networked Embedded Application

Example of NES :

1. Automobile: Safety Critical versus Telematics
2. Data Acquisition: Precision Agriculture and habitat Monitoring
3. Defense Application : Battle Space Surveillance
4. Biomedical Application
5. Disaster Management

Example :


Satellite
Ground Station
Camera
Router
PC
Cell Phone
IP Phone
Portable Station
Camera
PC
IP Phone
GPS Phone
In-Vehicle Station


ATM


ATM

Introduction to Networked Embedded Application

Characteristics of NES:

Functionality:

1. To Sense ….Temperature , Moisture, Movement, Light .etc
2. To Interact with Environment , User, Other Devices
3. To Communicate : Wireless or Wired

Constraints :

1. Small Size
2. Low energy requirement
3. Low Weight
4. Harsh working environment
5. Safety and Reliability
6. Low Cost

Introduction to Networked Embedded Application

Characteristics of NES:

2. Distributed Nature :

- Distributed Physically
- Distributed in function : NES components that perform **specific roles** and **work together with other NES components** to complete the system.
- Distributed Communication : local and global communication between the embedded systems distributed throughout the system.

Introduction to Networked Embedded Application

Characteristics of NES:

1. **Usability** :Entertainment ,Transport ,Communication .etc
2. **Availability**
   1. **Network (Access to System):** Mechanism through which the users can interface and interact with the components to investigate and rectify the problems.
   2. **Power Management** : NES do not have access to a renewable energy source and sometimes it is not possible to change the battery source either
3. **Dependability/ Reliability** : The system must guarantee a certain level of service that the user can depend on. Especially Avionics And Biomedical applications.

Introduction to Networked Embedded Application

Design Consideration for NES:

1. Deployments
2. Environment interaction
3. Life Expectancy of nodes
4. Communication between nodes
5. Reconfigurability
6. Security
7. Energy Constrained
8. Operating System

Introduction to Networked Embedded Application

1. Deployments : Physical distribution of the nodes

- The first concerns for deployment are safety, durability, and sturdiness;
- **Random Deployment:** Deploying NES nodes in an arbitrary fashion in the field. Random deployment is useful when the region being monitored is not accessible for precise placement of sensors.
- **Strategic Deployment:** Placing NES nodes at well-planned points so that the coverage is maximized or placing nodes strategically in a small field of concentration such that these nodes are not easily subjected to natural damage.

Introduction to Networked Embedded Application

2. Environment Interaction :

- Requirement of NES is an ability to work on their own and perhaps also has a feedback loop so that nodes can adapt to changes (failure of nodes, movement of objects) in the environment and continue functioning correctly.

Systems such as those used in

- Precision agriculture
- Chemical and hazardous gas monitoring

so on are designed to interact and react to changes in the system

Introduction to Networked Embedded Application

3. Life Expectancy of nodes:

- An essential requirement for nodes in a NES is a long life expectancy.
- This is because once deployed, it is very difficult to access and refurbish the batteries in the nodes.
- Also, these nodes must sustain environmental challenges such as
  - Inclement weather
  - Unexpected loss of nodes to animal interaction
  - Component failure

Introduction to Networked Embedded Application

4. Communication between nodes:

- Wired or Wireless links
- The nodes in the network may be **Stationary or Mobile**
- Mobile nodes bring in a whole range of issues related to **dynamic route and neighbor discovery, dynamic routing**, etc.
- NES should also be able to **reconfigure and adjust to tolerate loss of nodes** from a communication point of view.

Introduction to Networked Embedded Application

5. Reconfigurability :

- NES frequently require nodes **to reconfigure** after deployment.
- This may be to **add, remove, or change functionality or to adjust** parameters of the functionality.

6. Security :

- NES — particularly those that use wireless communication — are susceptible to malicious attack.
- Running security protocols is computationally expensive.

Introduction to Networked Embedded Application

7. Energy Constrained :

- Small form factor, low weight, and the deployment of NES nodes **in inaccessible and remote regions** implies that these nodes have access to a limited nonrenewable energy source.
- One major focus of the research community is to develop networking protocols, applications, operating systems, etc. (besides devices) that are energy efficient and utilize robust, high throughput but low power communication schemes.

Introduction to Networked Embedded Application

System Engineering and Engineering Trade off in NES

Hardware Constraints :

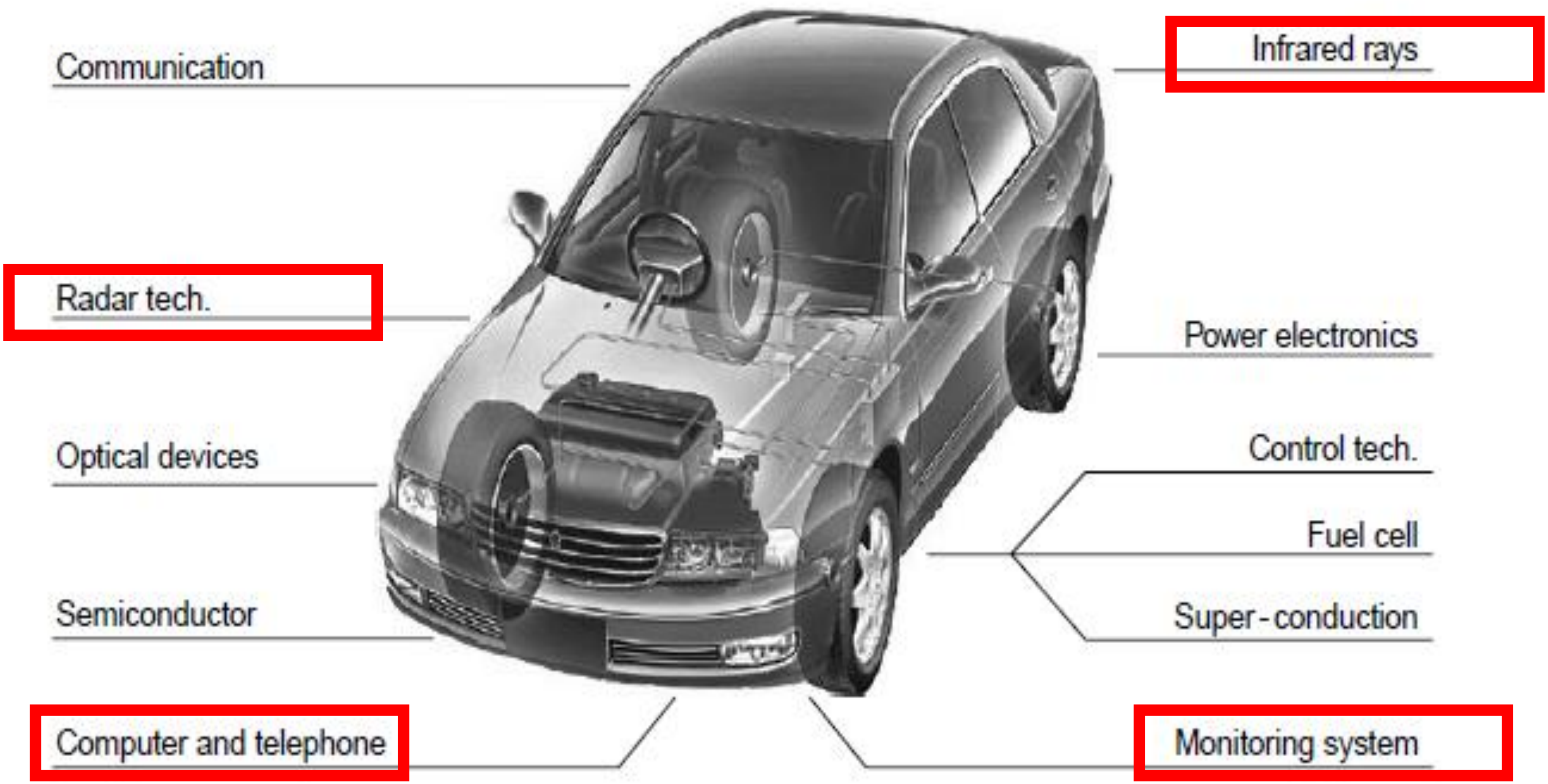
1. Area
2. Power
3. Size/Weights

Software Constraints :

1. Small Memory for Storage
2. Low Computational Complexity limits Size and Complexity of Software

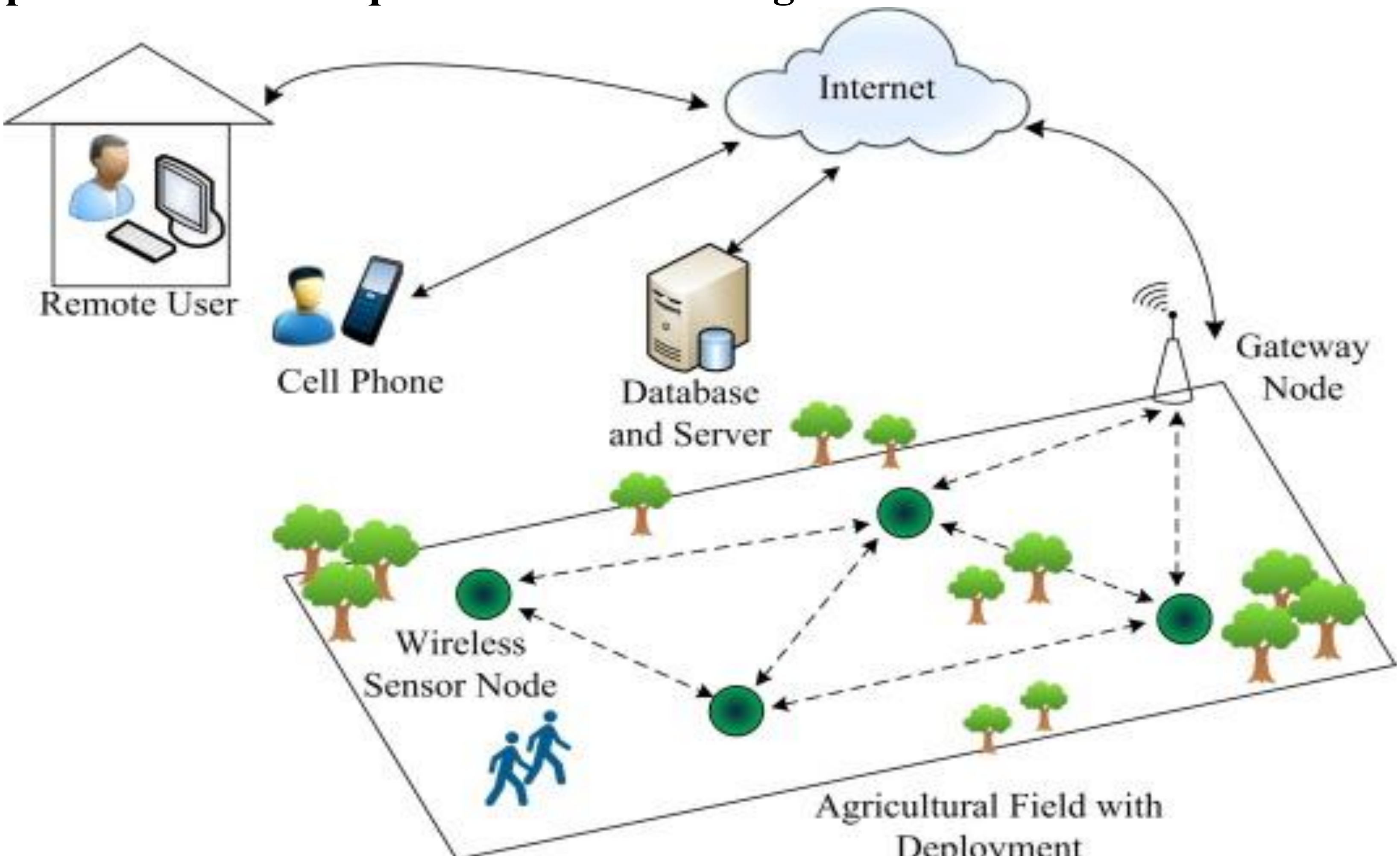
Introduction to Networked Embedded Application

Application 1 : Automobile



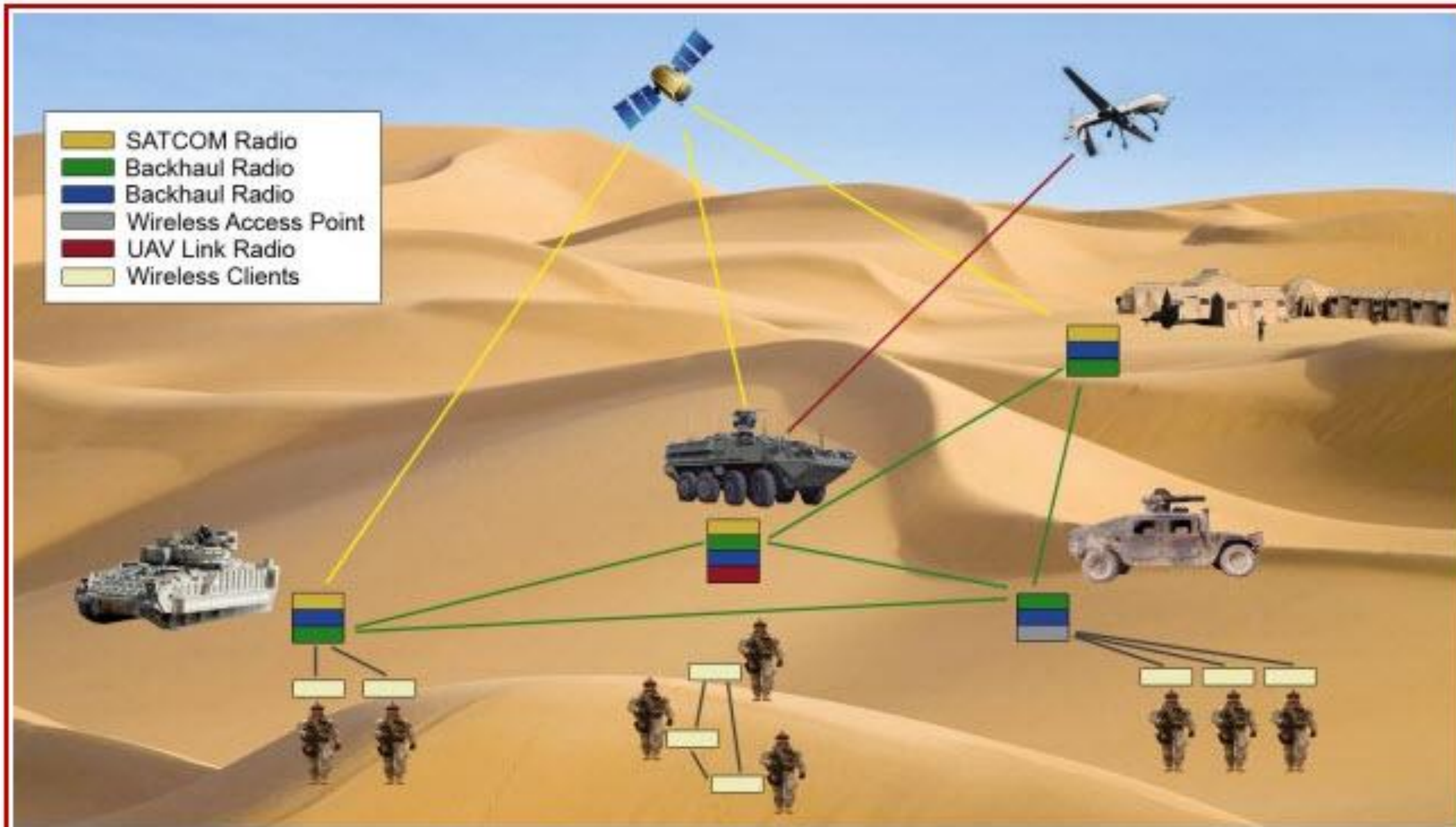
Introduction to Networked Embedded Application

Application: Data Acquisition: Precision Agriculture and Habitat Monitoring



Introduction to Networked Embedded Application

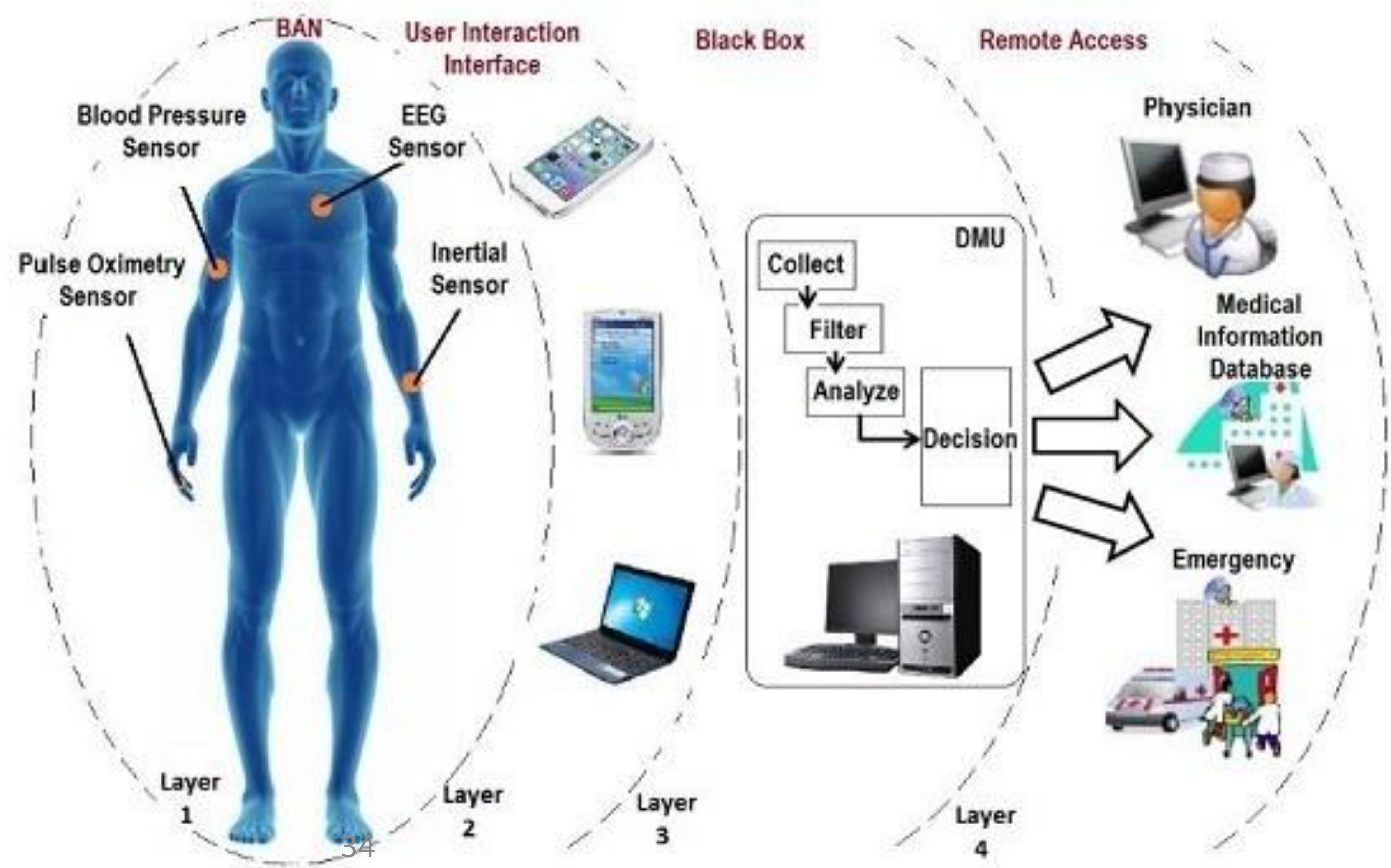
Defense Applications: Battle-Space Surveillance



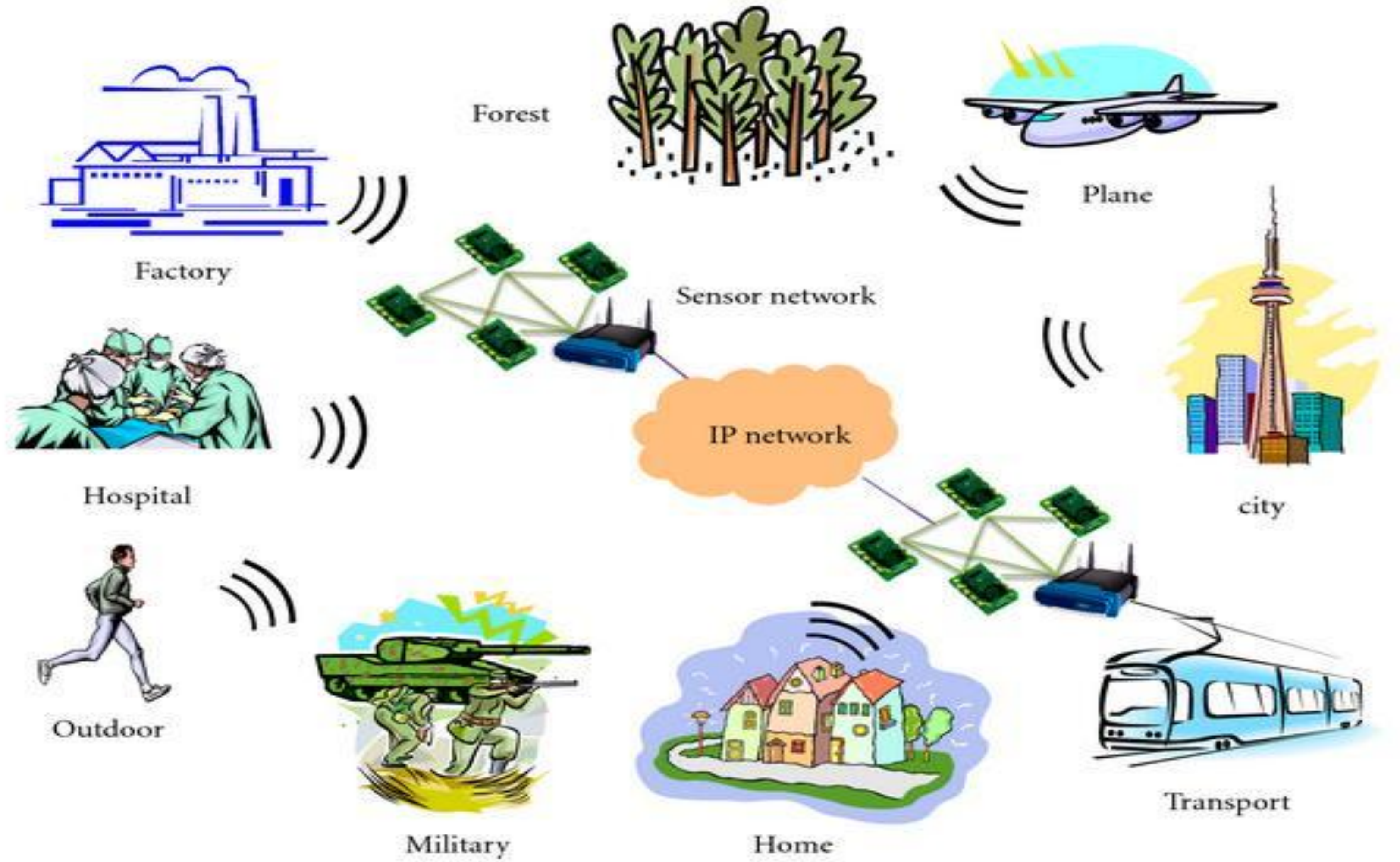
innovate achieve lead

Introduction to Networked Embedded Application

Applications: Medical



Introduction to Networked Embedded Application

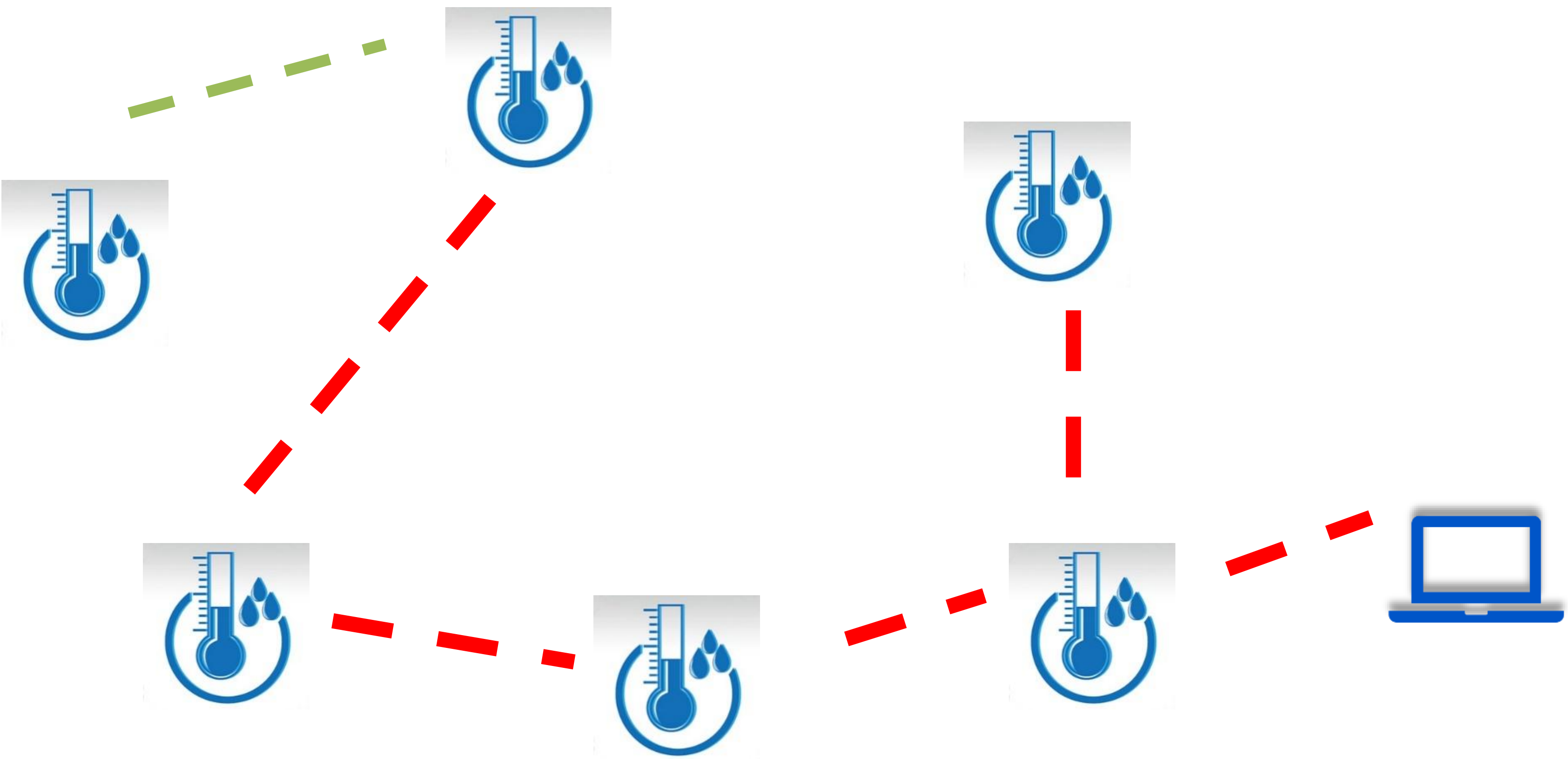


Introduction To Networked Embedded Systems

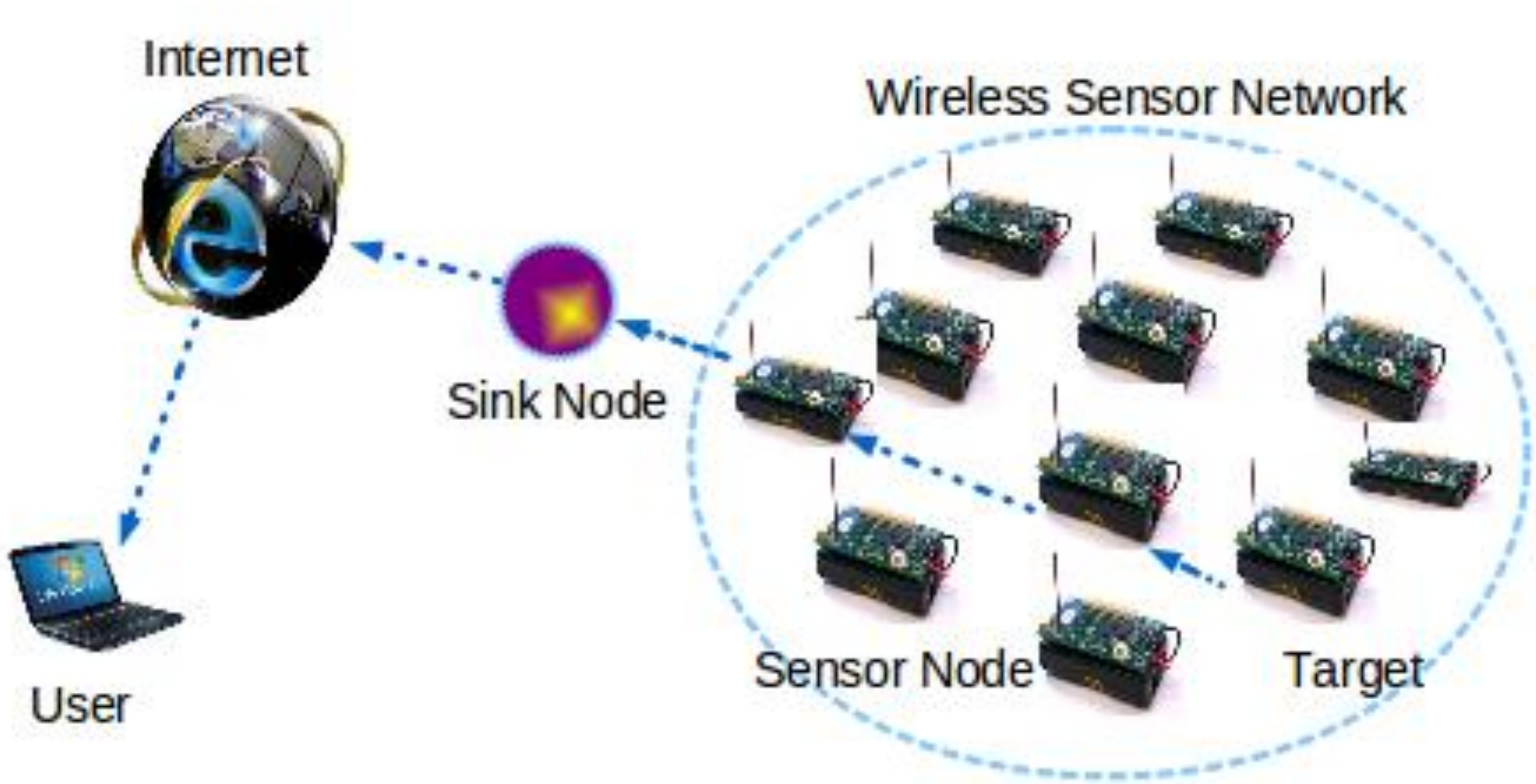
- IoT defined as :-
- "*Dynamic global internetwork with self configuring capabilities; of devices, mechanical and digital machines, objects, animals, people that have unique identities, physical attributes, virtual personalities – seamlessly integrated to transfer information associated with users and environments over the network without human interference*." Kranenburg 2008.
- What characterizes IoT is the 'end points'- the things. Simply put: Physical object + Controller/ Sensor/ Actuator + Internet = IoT .
- The scope of IoT is not limited to just connecting things (devices, applications, machines) to the internet. IoT allows these things to communicate and exchange data while executing applications that work towards a common goal.

IoT Vs NES

NES



IoT Vs NES
IoT



What is System ?

- Hardware
- Control
- Data

Distributed System

- Distributed H/w
  - Diff. systems provide diff. functions
- Distributed Control
  - Logical/Physical resources
  - Loosely Coupled
  - Tightly Coupled
- Distributed Data
  - Multiple copies of data at diff locations
  - Single copy distributed over multiple locations
  - Replication/ Partitioning of data

Attributes of a Distributed System

- Arbitrary No. of system and application process (logical resources)
- Modular Physical Architecture
- Communication by message passing using shared commn. systems
- System wide control

Schroeder's Definition

- A list of symptoms of a distributed system
  - Multiple processing elements (PEs)
  - Interconnection hardware
  - PEs fail independently
  - Shared states

Enslow's Definition

- Distributed system = distributed hardware + distributed control + distributed data

Hardware

- A single CPU with one control unit
- A single CPU – multiple ALUs – single control unit
- Separate Specialized functional units, such as one CPU with one floating pt. co-processor
- Multi-processors with multiple CPUs but only one single I/O system and on global memory
- Multiple computers with multiple CPUs, multiple I/O systems and local memories

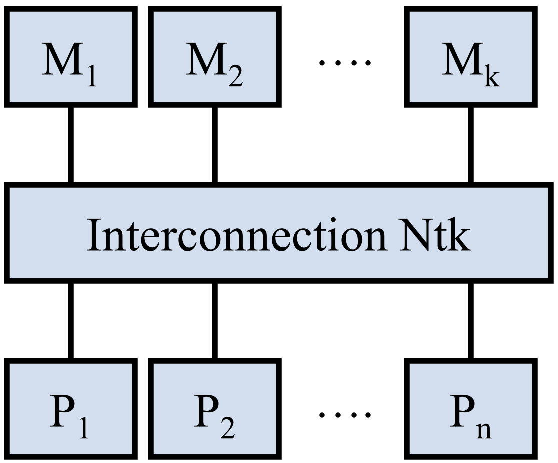
Control

- A Single Fixed Control pt. – Physically the system may or may not have multiple CPUs
- Single Dynamic Control point
- A fixed master - slave structure
- Dynamic master – slave structure
- Multiple Homogeneous control points – copies of the same controller are used
- Multiple Heterogeneous control points – different controllers are used

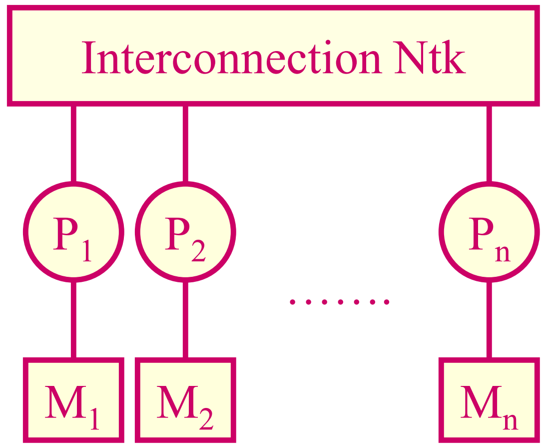
Data

- Centralized Data bases with a single copy of both files and dir
- Distributed files with a single centralized dir and no local dir
- Replicated database with a copy of files and dir at each site
- Partitioned data base with a master that keeps a complete duplicate copy of all files
- Partitioned database with a master that keeps only a complete dir
- Partitioned database with no master file/dir

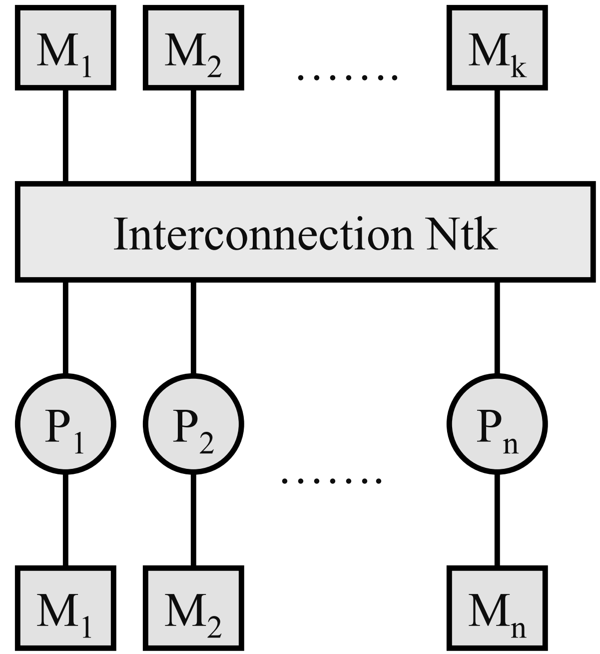
Multi-processor Architectures



Shared memory Architecture

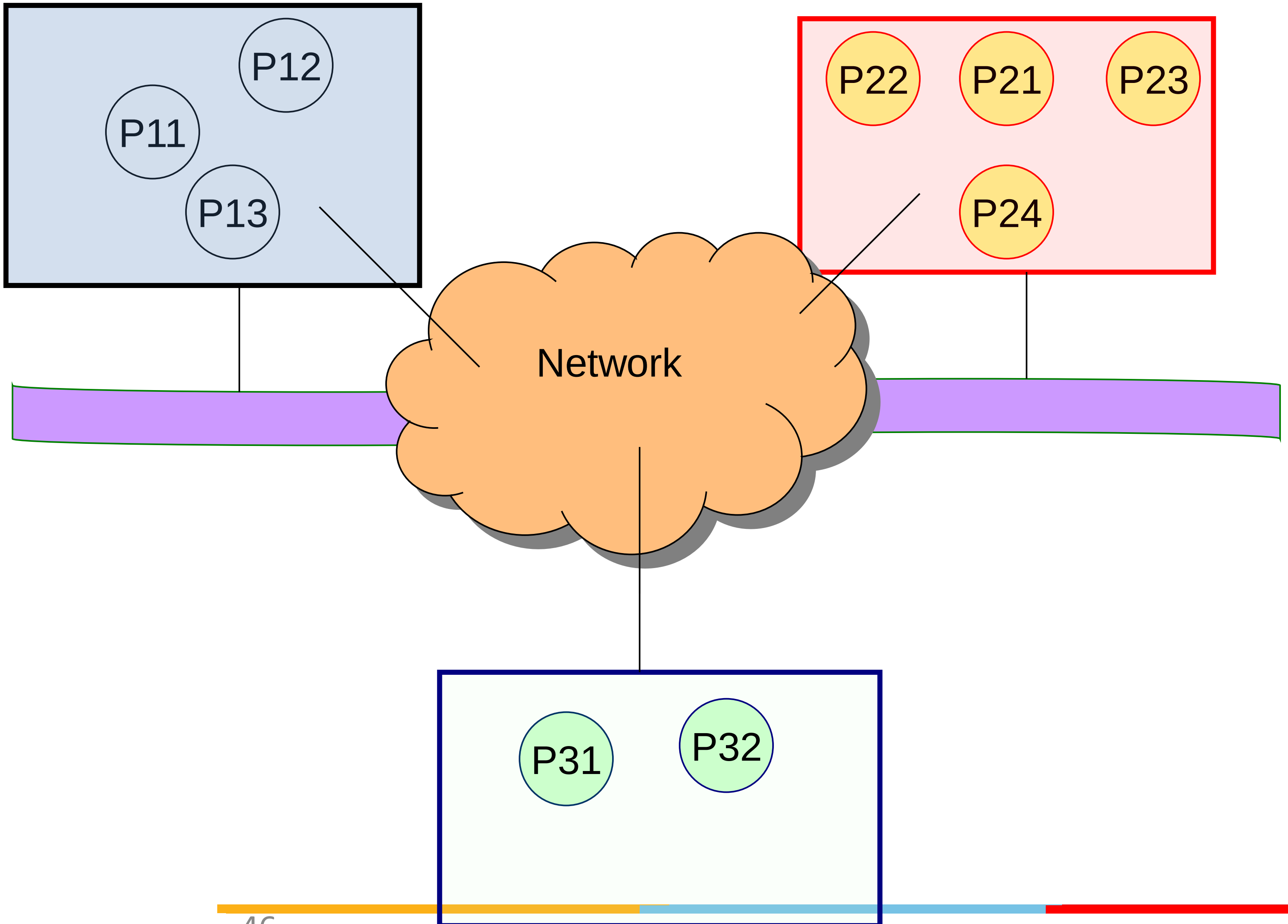


UMA



NUMA

Distributed memory Arch



P12
P11
P13
P22
P21
P23
P24
Network
P31
P32

Buses Vs Networks

- This collection of wires, that transfers data between components inside a computer, or between computers. This expression covers all related hardware components (wire, optical fiber, etc.) and software, including communication protocols. can be called a *bus* or a *network*.
- The term “bus” is usually used in reference to a set of wires connecting digital components within the enclosure of a computer device, and “network” is for something that is physically more widespread.
- In recent years, however, the word “bus” has gained popularity in describing networks that specialize in interconnecting discrete instrumentation sensors over long distances (“Fieldbus” and “Profibus” are two examples).
- In either case, we are making reference to the means by which two or more digital devices are connected together so that data can be communicated between them.

Prof. Manoj S Kakade

Email id: manoj.kakade@pilani.bits-pilani.ac.in

THANK YOU!!!!!