HW PLATFORM OF CPS

PREPARED BY,
S.BABITHA,AP,
IT,HITS.

DEFN

- Cyber-Physical Systems (CPS) integrate computation with physical processes.
- The hardware platform of a CPS is responsible for interfacing the physical world with the computational/control system.
- CPS hardware platforms typically include sensors, actuators, embedded processors, and communication components.

Key Components of CPS Hardware Platforms

Component	Description
Sensors	Collect data from the physical environment (e.g., temperature, pressure, motion, biomedical signals).
Actuators	Carry out physical actions based on control signals (e.g., motors, valves, robotic arms).
Embedded Processors	Microcontrollers or System-on-Chips (SoCs) that execute control algorithms and process sensor data.
Communication Interfaces	Provide wired/wireless connectivity (e.g., Ethernet, Wi-Fi, Zigbee, Bluetooth, 5G).
Power Supply Units	Deliver stable power (battery, solar, mains) to ensure reliable operation.
I/O Interfaces	Connect to external devices or subsystems (e.g., UART, SPI, I2C, GPIOs).

Types of CPS Hardware Platforms

Platform Type	Examples	Usage/Application
Microcontroller-Based Boards	Arduino, STM32, MSP430	Low-power embedded control for small-scale CPS (e.g., wearables, sensors).
SoC Platforms	Raspberry Pi, BeagleBone, Nvidia Jetson	Mid-to-high computing power for vision, AI, and real-time analytics.
Real-Time Embedded Systems	PXI, NI CompactRIO, Dspace	Industrial CPS where deterministic real-time response is crucial (e.g., automotive, robotics).
FPGA-Based Platforms	Xilinx Zynq, Intel FPGA	Custom hardware acceleration, low-latency signal processing.
Custom ASICs	Application-Specific Integrated Circuits	High-efficiency, domain-specific CPS (e.g., pacemakers, avionics).

Interaction Between Components

- **Sensors** → gather data from environment
- **Embedded Processor** → analyzes data and makes decisions
- **Actuators** → act on decisions to affect physical world
- **Communication Module** → transmits data/control signals among system components or to/from the cloud

Example: Smart Grid CPS Hardware Platform

- **Sensors**: Smart meters, current/voltage sensors
- **Processors**: ARM Cortex-M based microcontrollers
- **Actuators**: Circuit breakers, load control switches
- **Communication**: Zigbee, 5G, or Ethernet
- **Power**: Solar panels, backup batteries


Thank you

CPS NETWORK

PREPARED BY,
S.BABITHA,AP,
IT,HITS.

CPS Network (Communication in Cyber-Physical Systems)

- In **Cyber-Physical Systems (CPS)**, the **network** enables **communication between physical components (sensors, actuators)** and **cyber components (processors, controllers, cloud)**.
- It is essential for **real-time data exchange**, **coordination**, and **remote control** in distributed systems.

Role of the Network in CPS

- **Data Collection**: Transmit sensor data to processing units
- **Control Command Delivery**: Send control signals to actuators
- **Coordination**: Synchronize operations among distributed CPS nodes
- **Remote Access**: Enable monitoring and control via the internet/cloud
- **Feedback Loop**: Close the loop for real-time decision-making

Types of CPS Networks

- **Based on Communication Type**

Type	Description	Example
Wired Networks	Reliable and fast, used in stationary CPS	Ethernet, CAN bus, Modbus
Wireless Networks	Flexible, used in mobile or remote CPS	Wi-Fi, Zigbee, Bluetooth, 5G

Types of CPS Networks

Based on Range:

Range	Technologies
Short-range	Bluetooth, NFC, Zigbee
Medium-range	Wi-Fi, LoRa
Long-range	Cellular (4G/5G), Satellite

Common Network Technologies in CPS

Protocol/Tech	Description	Used In
CAN Bus	Real-time, low-latency protocol	Automotive CPS
Zigbee	Low-power, mesh network	Smart homes, industrial IoT
Wi-Fi	High bandwidth wireless	Smart buildings, health CPS
Bluetooth (BLE)	Low-energy, short range	Wearables, medical devices
5G	Ultra-low latency, high speed	Autonomous vehicles, smart factories
LoRaWAN	Long range, low power	Remote monitoring (agriculture, environment)
Ethernet	Stable, fast wired network	Industrial control systems
MQTT / CoAP	Lightweight IoT protocols	Cloud-connected CPS

Example: Smart Grid CPS Network Flow

- [Smart Meters] → Zigbee/Wi-Fi → [Edge Gateway] → Ethernet/5G → [Cloud/Control Center]

Challenges in CPS Networking

- **Real-Time Constraints**: Timely delivery of messages
- **Security & Privacy**: Protection from cyber threats
- **Reliability**: Fault-tolerant communication
- **Bandwidth & Latency**: Need for high throughput and low delay
- **Interoperability**: Multiple devices and protocols

Network in CPS Architecture Diagram (Simplified)

- +-----------+ Wireless/Wired +------------+ Control +------------+
- | Sensors | -----------------------> | Processor | ----------------> | Actuators |
- +-----------+ +------------+ +------------+
- ↑ ↓
- | Cloud/Edge Network |
- +------------------------------------+


THANK
YOU

CPS SOFTWARE STACK

PREPARED BY,
S.BABITHA,AP,
IT,HITS.

CPS Software Stack (SW Stack)

- 🧠 In **Cyber-Physical Systems (CPS)**, the **software stack** defines how various layers of software—from hardware interfaces to application logic—interact to manage both **cyber** (computing/control) and **physical** (sensor/actuator) components.

Overview of the CPS Software Stack

- +----------------------------+
- | Application Layer |
- +----------------------------+
- | Middleware Layer |
- +----------------------------+
- | Operating System (RTOS) |
- +----------------------------+
- | Hardware Abstraction |
- +----------------------------+
- | Device Drivers / Firmware|
- +----------------------------+
- | Hardware (Sensors/Actuators, MCU, etc.) |
- +----------------------------+

🔍 Layer-by-Layer Explanation

- ⚙️ **Hardware Layer**
- Physical components: Sensors, actuators, microcontrollers, SoCs
- Interfaces: GPIO, ADC/DAC, I2C, SPI, UART
- **2. 🔌 Device Drivers / Firmware**
- Interface code that directly communicates with hardware
- Converts raw signals to usable data for upper layers
- **3. 📃 Hardware Abstraction Layer (HAL)**
- Standardizes hardware access across different platforms
- Enables portability and reusability
- **4. ⏱️ Operating System Layer (RTOS)**
- Real-Time Operating System (RTOS) like FreeRTOS, VxWorks, Zephyr
- Handles:
  - Task scheduling
  - Interrupt management
  - Inter-process communication
  - Timing and real-time guarantees

Layer-by-Layer Explanation

- **5. 🛠️ Middleware Layer**
- Communication frameworks (e.g., DDS, MQTT, ROS)
- Data serialization, messaging, and network protocols
- Security services (authentication, encryption)
- Resource discovery and management
- **6. 📲 Application Layer**
- User-level logic and control algorithms
- Decision-making, analytics, AI/ML models
- Human-machine interface (HMI)
- Cloud/edge communication (e.g., dashboards, APIs)

Example: Smart Traffic CPS SW Stack

Layer	Components
Application	Traffic control algorithms, web dashboard
Middleware	DDS for sensor-car communication
RTOS	QNX or VxWorks on edge controller
HAL	Abstracts traffic light control logic
Drivers	Camera sensor drivers, GPIO drivers
Hardware	Cameras, actuators for lights, edge processor

Key Software Stack Features in CPS

- **Modularity**: Separation of concerns across layers
- **Portability**: HAL and middleware enable cross-platform deployment
- **Real-Time Capability**: Ensures timely response in critical systems
- **Scalability**: From embedded nodes to cloud-based systems
- **Security**: Encryption, authentication, secure updates

Thank you!

CYBER-PHYSICAL SYSTEMS (CPS) IN THE REAL WORLD

PREPARED BY,

S.BABITHA,AP,

IT,HITS.

WHAT IS CYBER-PHYSICAL SYSTEMS?

- Cyber-Physical Systems (CPS) integrate computational and physical components to create systems that interact with and control the physical world.
- They are found in various industries, including manufacturing, transportation, healthcare, energy, and smart cities, automating and optimizing processes through real-time data analysis and control.

KEY CONCEPTS

- **Integration of Cyber and Physical:**
- CPS seamlessly blend computational elements with physical processes, such as sensors, actuators, and control systems.
- **Real-Time Control:**
- CPS are designed to monitor and control physical processes in real-time, enabling timely adjustments and responses.
- **Data-Driven Decision-Making:**
- CPS utilize data from sensors and other sources to make informed decisions, optimize performance, and automate tasks.
- **Autonomous Actions:**
- CPS can be designed to perform actions autonomously, reducing human intervention and increasing efficiency.

EXAMPLES OF CPS IN ACTION:

- **Manufacturing:**
- Industrial control systems (ICS) and SCADA (Supervisory Control and Data Acquisition) systems are used to monitor and control machinery and processes on assembly lines, optimizing production and reducing waste.
- **Transportation:**
- Autonomous vehicles, smart traffic management systems, and advanced driver-assistance systems (ADAS) leverage CPS to enhance safety, reduce congestion, and improve transportation efficiency.
- **Healthcare:**
- Medical devices, such as pacemakers and insulin pumps, use CPS to monitor patients' health and deliver personalized therapies.

Energy:

- Smart grids, energy management systems, and renewable energy sources rely on CPS to optimize energy distribution, reduce energy consumption, and promote sustainability.
- **Smart Cities:**
- Smart buildings, smart traffic management systems, and intelligent infrastructure use CPS to optimize resource management, improve public safety, and enhance the quality of life for residents.
- **Agriculture:**
- Precision agriculture techniques, such as automated irrigation systems and crop monitoring, utilize CPS to optimize resource utilization and improve yields.

BENEFITS OF CPS

Increased Efficiency:

- CPS automate tasks, optimize processes, and reduce waste, leading to significant efficiency gains.
- **Enhanced Safety:**
- CPS can monitor and control physical processes in real-time, reducing the risk of accidents and improving safety.
- Improved Performance:
- CPS can optimize performance, reduce downtime, and improve overall productivity.
- Greater Automation:
- CPS enable automation of tasks, reducing the need for human intervention and increasing efficiency.
- Data-Driven Decision-Making:
- CPS provide real-time data and analytics, enabling data-driven decision-making and better informed choices.

What is an example of a cyber-physical systems?

- Few examples of cyber-physical systems are, smart manufacturing facilities with collaborative robots, autonomous vehicles utilizing sensors and AI for navigation, smart grids optimizing energy distribution, implantable medical devices like pacemakers providing automated therapy adjustments, and building automation.

For deeper insight on the capabilities and potential of CPS, here are some examples you may come across in a range of fields.

- Operational Technology. ...
- Industrial Internet of Things (IIoT) ...
- Industrial Control Systems (ICS) ...
- Building Management Systems (BMS) ...
- Smart Grids. ...
- Smart Buildings. ...
- Robotics. ...
- Smart Transportation Systems.

What is a cyber physical system in the real world?

- A cyber-physical system is a collection of computing devices communicating with one another and interacting with the physical world via sensors and actuators in a feedback loop.
- Examples include CPS includes self-driving cars, The STARMAC is a small quadrotor aircraft.


thank you

IoT

PREPARED BY,
S.BABITHA,AP,
IT,HITS

IOT

- The Internet of Things (IoT) refers to a vast network of physical objects, or "things," that are embedded with sensors, software, and other technologies, enabling them to connect and exchange data with other devices and systems over the internet.
- Essentially, it's the concept of connecting everyday objects to the internet and to each other, allowing them to collect and share data.

COMPARISON BETWEEN IoT AND CPS

Feature	IoT	CPS
Integration Level	Loosely coupled	Tightly integrated
Functionality	Data collection and communication	Monitoring, control, and decision-making
Real-Time Operation	Less critical	Essential
Complexity	Relatively simple	Highly complex
Examples	Smart thermostats, fitness trackers	Self-driving cars, smart factories

KEY CONCEPTS

- **Interconnected Devices:**
- IoT involves a wide range of devices, from simple sensors to complex machines, that are connected to the internet.
- **Data Collection and Exchange:**
- These devices are equipped with sensors that collect data about their surroundings or their own status, and they can then transmit this data to other devices or systems.
- **Automation and Control:**
- The data collected by IoT devices can be used to automate tasks, control processes, and provide valuable insights for decision-making.

HOW IT WORKS

- **1. Sensors and Devices:**
- IoT devices have sensors that gather data about their environment or their own state.
- **2.Connectivity:**
- This data is transmitted over the internet or other networks using various communication protocols.
- **3. Data Processing:**
- The data is then processed and analyzed, often in the cloud, to extract meaningful information.
- **4. User Interface:**
- The processed data is presented to users through interfaces, such as dashboards or mobile apps, allowing them to monitor and control the devices.

EXAMPLES OF IoT IN ACTION:

- **Smart Homes:**
- Smart thermostats, lighting systems, and security systems that can be controlled remotely.
- **Industrial IoT (IIoT):**
- Connecting machinery and equipment in factories to monitor performance, predict maintenance needs, and optimize production processes.
- **Wearable Technology:**
- Smartwatches and fitness trackers that monitor health metrics and activity levels.
- **Smart Cities:**
- Utilizing sensors to monitor traffic flow, manage energy consumption, and improve public safety.
- **Agriculture:**
- Using sensors to monitor soil conditions, weather patterns, and crop growth.

BENEFITS OF IOT

- ✓ **Increased Efficiency:**
- ✓ Automating tasks and optimizing processes can lead to significant efficiency gains.
- ✓ **Improved Decision-Making:**
- ✓ Access to real-time data provides valuable insights for informed decision-making.
- ✓ **Enhanced Productivity:**
- ✓ By automating tasks and streamlining workflows, IoT can boost productivity.
- ✓ **New Business Models:**
- ✓ IoT enables businesses to create innovative products and services and develop new revenue streams.

Thank you

KEY FEATURES OF CPS

PREPARED BY,
S.BABITHA,AP,
IT,HITS

KEY FEATURES OF CPS

- **Integration of Physical and Cyber Components:**
- CPS blurs the lines between the physical world and the digital realm by embedding computing and communication capabilities within physical objects and processes.
- **Real-time Operation:**
- CPS systems respond to changes in the physical environment with minimal delay, enabling time-critical applications.
- **Feedback Loops:**
- Sensors and actuators work together in a feedback loop, allowing for continuous monitoring and control of physical processes.

- **Network Connectivity:**
- CPS relies on networks for communication and data exchange between different components and systems, facilitating coordination and information flow.
- **Adaptability and Autonomy:**
- CPS can adapt to changing conditions, make decisions based on real-time data, and optimize their behavior over time.
- **Heterogeneity:**
- CPS systems often comprise diverse components like sensors, actuators, processors, and communication devices, requiring sophisticated integration and coordination.

- **Real-time Computation:**
- CPS possesses the ability to perform computations and make decisions based on real-time data, allowing for quick responses to physical changes.
- **Robustness and Reliability:**
- CPS must be reliable and robust to ensure safe and effective functioning, especially in safety-critical applications.
- **Safety-Critical Applications:**
- Certain CPS applications prioritize safety over performance, necessitating careful design and operation to prevent harm.

What are the key features of cyber-physical system?

- Cyber-physical systems consist of three interconnected parts: the physical processes they interact with, the computational elements that process data and make decisions, and the communication networks that facilitate real-time data exchange between these components.

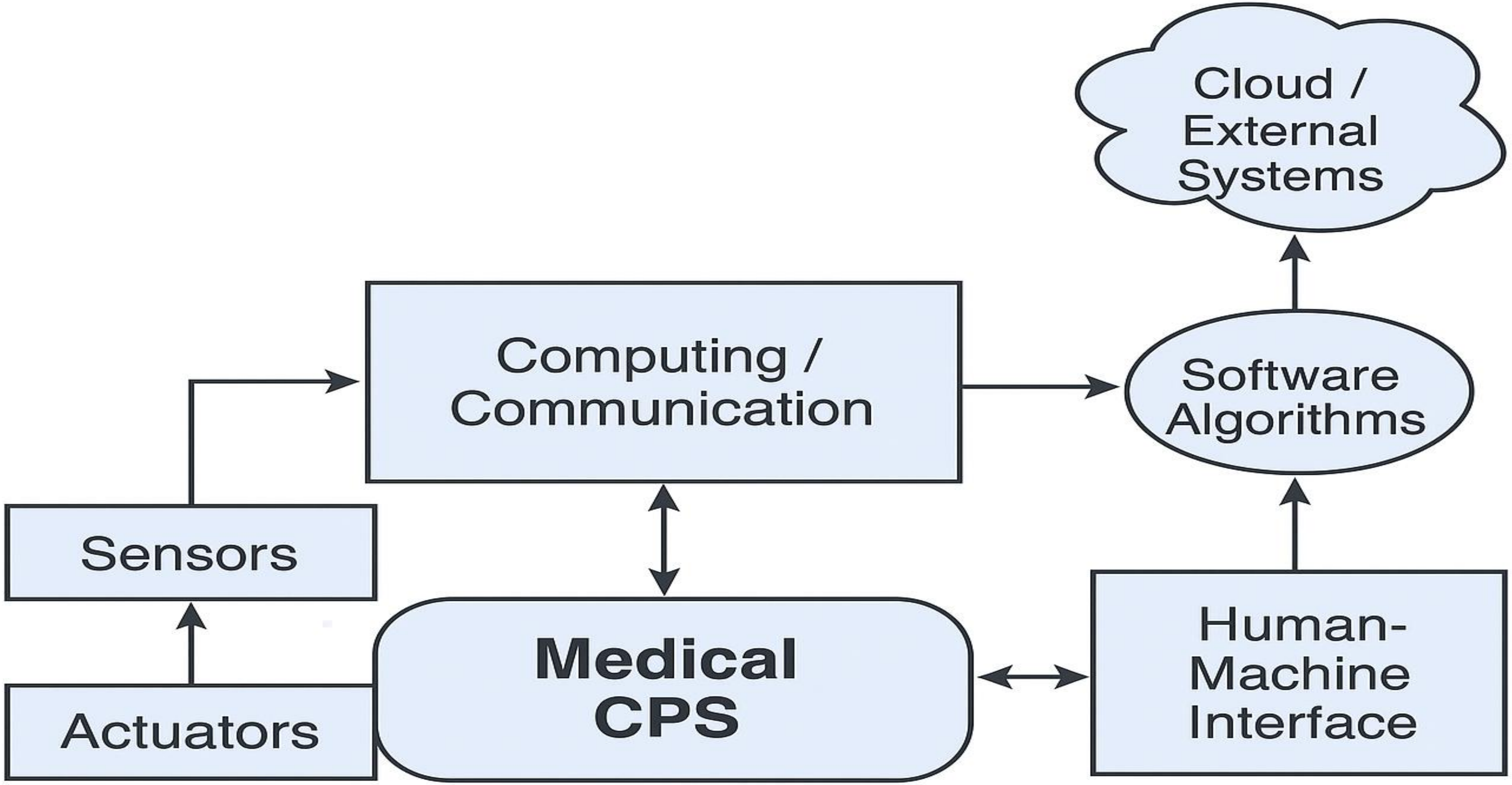

Thank you!

MEDICAL CPS

PREPARED BY,
S.BABITHA,AP,
IT,HITS.

DEFN

- **Medical Cyber-Physical Systems (Medical CPS)** are integrated systems that combine medical devices, sensors, actuators, communication networks, software, and computing platforms to monitor, diagnose, and treat patients in real time.
  ➞ They bridge the physical (biological) world and the cyber (computing + network) world for advanced healthcare delivery.


Cloud / External Systems
Computing / Communication
Software Algorithms
Sensors
Medical CPS
Human-Machine Interface
Actuators
Medical CPS

Key Components

- **Sensors:** Measure physiological parameters (e.g., ECG, blood pressure, glucose levels, oxygen saturation).
- **Actuators:** Administer treatment (e.g., insulin pumps, ventilators, drug delivery systems).
- **Communication networks:** Enable data exchange (e.g., wired, wireless, Bluetooth, Wi-Fi, 5G).
- **Embedded computing:** Process sensor data and make decisions (e.g., microcontrollers, edge computing devices).
- **Software algorithms:** Analyze data for diagnostics, predictive alerts, and control.

Applications

- **Implantable and wearable medical devices** — pacemakers, insulin pumps, smart prosthetics.
  ✅ **Remote patient monitoring** — continuous tracking of vital signs at home.
  ✅ **Surgical robots** — assist or perform precise procedures under doctor supervision.
  ✅ **Smart hospital rooms** — integrated monitoring, alarms, and environment control for patient safety.
  ✅ **Emergency response systems** — real-time data from ambulances to hospitals for rapid intervention.

Benefits

- **Improved patient safety and outcomes** — real-time monitoring and timely interventions.
- **Personalized treatment** — therapy adjusted based on individual patient data.
- **Reduced hospital visits** — via remote monitoring and telemedicine.
- **Efficient use of resources** — smarter allocation of healthcare staff and equipment.

Challenges

- **Cybersecurity risks** — medical CPS devices are vulnerable to hacking, which can endanger patients.
- **Interoperability issues** — integration of devices from different manufacturers can be complex.
- **Real-time reliability** — systems must function with high precision and minimal delay to ensure safety.
- **Regulatory compliance** — strict standards for safety, data privacy (e.g., HIPAA, GDPR).

Examples

- Artificial pancreas systems for diabetes management.
- Smart inhalers for asthma patients with usage tracking.
- Robotic surgical assistants like **da Vinci Surgical System**.
- Wireless vital signs monitors in ICUs.


Thank you!

PLATFORM COMPONENTS OF CPS

PREPARED BY,
S.BABITHA,AP,
IT,HITS.

DEFN

- The **CPS platform** provides the hardware and software foundation that enables sensing, computing, communication, and actuation.
- These components work together to ensure CPS can monitor, control, and interact with the physical environment in real-time.

Main Platform Components

- Sensing Components
- Actuation Components
- Computing / Processing Unit
- Communication Network
- Software / Middleware
- Human-Machine Interface

Sensing Components

- Devices that collect data from the physical world.
- Examples:
  → Temperature sensors
  → Pressure sensors
  → Cameras, LiDAR, GPS
  → Wearable biosensors (in Medical CPS)
- **Function:** Measure physical parameters and convert them into digital signals.

Actuation Components

- Devices that perform physical actions based on control commands.
- Examples:
  → Motors, robotic arms
  → Valves, relays
  → Drug delivery mechanisms (in Medical CPS)
- **Function:** Interact with or alter the physical environment.

Computing / Processing Unit

- Embedded systems or controllers that process sensor data and generate actuator commands.
- Examples:
  → Microcontrollers
  → FPGAs / DSPs
  → Edge devices
  → Cloud/edge hybrid platforms
- **Function:** Run control algorithms, real-time analytics, and decision-making logic.

Communication Network

- Provides connectivity between CPS components, often across different layers (local and remote).
- Examples:
  → Ethernet, Wi-Fi
  → Bluetooth, ZigBee, 5G
  → CAN bus (for automotive CPS)
  → Industrial IoT protocols (e.g., MQTT, OPC UA)
- **Function:** Enable data exchange between sensors, processors, actuators, and external systems.

Software / Middleware

- Provides the software environment to develop, deploy, and manage CPS applications.
- Features include:
  → Real-time operating systems (RTOS)
  → Middleware for device integration
  → Data analytics engines
  → Security modules
- **Function:** Abstract hardware details, manage data flow, ensure timing, and handle failures.

Human-Machine Interface (HMI)

- Provides interaction between human operators and the CPS.
- Examples:
  → Dashboards, control panels
  → Mobile apps
  → AR/VR interfaces for monitoring and control
- **Function:** Allow users to monitor system status, input commands, and visualize data.


Thank you

Processors in CPS Hardware Platforms

PREPARED BY,
S.BABITHA,AP,
IT,HITS

DEFN

- Processors are the **core computational units** in Cyber-Physical Systems (CPS).
- They perform **data acquisition**, **signal processing**, **control logic**, and **communication**.
- The choice of processor depends on the system's **real-time needs**, **power constraints**, and **application complexity**.

Types of Processors Used in CPS

Type	Description	Use Cases
Microcontrollers (MCUs)	Low-power, single-chip processors with built-in I/O, timers, and memory	IoT devices, sensors, medical implants
Digital Signal Processors (DSPs)	Specialized for fast signal processing tasks (FFT, filtering, etc.)	Audio/video processing, motor control
System-on-Chip (SoC)	Combines CPU, GPU, memory, and peripherals on one chip	Smart cameras, drones, edge AI
Field-Programmable Gate Arrays (FPGAs)	Reconfigurable logic devices for parallel processing	High-speed control, image/video processing
Application-Specific Integrated Circuits (ASICs)	Custom-designed chips optimized for specific tasks	Pacemakers, industrial controllers
Multi-core Processors	Multiple cores allow parallel tasks	Smart cars, autonomous robots

Processor Functions in CPS

- **Sensor Data Acquisition**
  - Read data from analog/digital sensors using ADC, I2C, SPI.
- **Real-Time Processing**
  - Apply filters, control algorithms (PID, AI-based), safety checks.
- **Decision Making & Actuation**
  - Translate control outputs into commands for actuators.
- **Communication**
  - Interface with wireless/wired networks (Wi-Fi, CAN, Zigbee).
- **Power & Resource Management**
  - Optimize energy usage, schedule tasks, manage peripherals.

Examples of CPS Processors

Processor	Features	Example System
ARM Cortex-M Series	Low-power, real-time	Wearables, medical devices
Raspberry Pi (Broadcom SoC)	Linux-based, powerful	Smart home hubs, automation
Nvidia Jetson	GPU + CPU for AI	Drones, autonomous vehicles
TI C2000 DSP	Fast control loops	Power grid, industrial automation
Intel Atom	High-performance x86	Industrial PCs, edge gateways

Thank you

RTOS (Real-Time Operating System) in Cyber-Physical Systems (CPS)

PREPARED BY,
S.BABITHA,AP,
IT,HITS.

DEFN

- A **Real-Time Operating System (RTOS)** is a type of operating system designed to **execute tasks within strict timing constraints**, making it a core component of many **Cyber-Physical Systems (CPS)** where **deterministic, predictable behavior** is crucial.

Key Characteristics of RTOS

Feature	Description
Determinism	Guarantees that high-priority tasks meet deadlines (real-time behavior)
Preemptive Multitasking	Allows switching between tasks based on priority
Low Latency	Fast context switching and interrupt handling
Task Prioritization	Ensures time-critical tasks are served first
Minimal Footprint	Lightweight and suitable for embedded hardware
Support for Synchronization	Semaphores, mutexes, message queues for safe task interaction

RTOS Architecture in CPS

```
+-------------------------+
|     Application Code    |
+-------------------------+
|     RTOS Services       |
| - Scheduler             |
| - Timers                |
| - IPC (Queues, Mutexes) |
| - Memory Mgmt           |
+-------------------------+
|   Hardware Abstraction  |
+-------------------------+
|   Device Drivers & HAL  |
+-------------------------+
|       Hardware          |
+-------------------------+
```

Common RTOS Used in CPS

RTOS	Features	Typical CPS Use
FreeRTOS	Open source, small footprint	IoT devices, wearables
VxWorks	Certified for safety-critical systems	Aerospace, automotive
Zephyr	Scalable, IoT-focused	Smart sensors, medical CPS
QNX	POSIX-compliant, fault-tolerant	Automotive infotainment, ADAS
RTEMS	Space-grade, real-time	Satellites, space probes
TI-RTOS	Optimized for TI microcontrollers	Industrial automation

APPLICATION OF RTOS IN CPS

CPS Domain	RTOS Role
Automotive	Real-time control of ABS, engine, and ADAS
Industrial Automation	Control of PLCs, motion systems, and robots
Medical Devices	Real-time monitoring (e.g., heart rate, insulin delivery)
Aerospace	Avionics, mission-critical flight systems
Smart Grids	Power control, fault detection systems

Why RTOS in CPS?

- Ensures **real-time guarantees** for physical control
- **Improves reliability** in mission-critical systems
- Enables **modular multitasking** in resource-constrained environments
- Supports **scalability** from microcontrollers to edge servers

Challenges in RTOS Development

- Hard real-time guarantees require **careful task scheduling**
- Requires **deep understanding of hardware constraints**
- Debugging timing and concurrency issues is complex
- Safety certifications (e.g., ISO 26262, DO-178C) can be demanding.


Thank You
So Much

SCHEDULING REAL-TIME CONTROL TASKS IN CPS

PREPARED BY.
S.BABITHA,AP,
IT,HITS.

DEFN

- In **Cyber-Physical Systems (CPS)**, **real-time scheduling** ensures that control tasks (like sensor reading, decision-making, and actuator control) execute **within strict timing constraints**.
- Failing to meet these deadlines can result in system instability or unsafe behavior — especially in **automotive**, **aerospace**, **industrial**, or **medical** CPS.

REAL-TIME TASK BASICS

Concept	Description
Task	A unit of work (e.g., read sensor, compute PID control, update actuator)
Period (T)	Time between successive task releases
Deadline (D)	Time by which the task must complete
Execution Time (C)	Time taken to complete the task
Response Time	Time from task release to completion

TYPES OF REAL-TIME TASKS

Task Type	Description	Example
Hard Real-Time	Must meet deadlines every time	Airbag deployment, pacemaker pulse
Firm Real-Time	Missing a deadline degrades performance	Video frame processing in ADAS
Soft Real-Time	Occasional deadline misses are acceptable	Smart lighting control

COMMON REAL-TIME SCHEDULING ALGORITHMS

Algorithm	Type	Key Idea	Best Use
Rate Monotonic Scheduling (RMS)	Fixed-priority	Shorter period = higher priority	Periodic, static systems
Earliest Deadline First (EDF)	Dynamic-priority	Task with closest deadline first	High processor utilization
Deadline Monotonic	Fixed-priority	Deadline dictates priority	When deadlines < periods
Round Robin	Time-sharing	Equal time slices for all tasks	Non-critical soft real-time tasks
Priority-Based Preemptive	Fixed/dynamic	High-priority tasks can interrupt low-priority ones	Mixed criticality systems

EXAMPLE: PERIODIC CONTROL TASKS SCHEDULING

Task	Period (T)	Execution Time (C)	Deadline (D)	Priority (RMS)
Sensor Reading	10 ms	1 ms	10 ms	High
Control Logic (PID)	20 ms	2 ms	20 ms	Medium
Data Logging	100 ms	5 ms	100 ms	Low


Thank you


THANK
YOU!

SENSORS

PREPARED BY,
S.BABITHA,AP,
IT,HITS.

DEFN

- **Sensors** are critical hardware components in CPS that collect real-world data and convert it into electrical signals that processors can understand.
- This enables **perception**, **monitoring**, and **control** in CPS environments.

Classification of Sensors in CPS

Sensor Type	Measures	Examples	Applications
Temperature Sensors	Heat	LM35, DS18B20	HVAC, smart agriculture
Pressure Sensors	Pressure/force	BMP180, MPX5010	Automotive, aerospace
Motion Sensors	Movement/acceleration	Accelerometer (MPU6050), gyroscope	Wearables, robotics, drones
Proximity Sensors	Distance to objects	Ultrasonic, IR, LIDAR	Collision avoidance, automation
Light Sensors	Light intensity	LDR, TSL2561	Smart lighting, agriculture
Gas/Chemical Sensors	Gas concentration	MQ series (MQ2, MQ135)	Environment monitoring, safety
Sound Sensors	Sound levels	Microphone modules	Voice control, security systems
Image Sensors	Visual data	CMOS/CCD camera modules	Surveillance, medical imaging
Position Sensors	Location/position	GPS, encoders	Smart vehicles, robotics
Touch Sensors	Physical touch/pressure	Capacitive sensors	Touch screens, wearables
Humidity Sensors	Moisture content	DHT11, DHT22	Greenhouses, climate control

Sensor Workflow in CPS

- **Physical Phenomenon** → e.g., temperature, motion, sound
- **Sensor Measures It** → converts to electrical signal
- **Signal Conditioning** → filtering, amplification
- **Analog-to-Digital Conversion (ADC)** → if needed
- **Processor Reads & Analyzes** → control or report results
- **Actuators Respond or Data is Transmitted**

Example: Smart Healthcare CPS Sensors

Sensor	Role
ECG Sensor	Monitor heart activity
SpO2 Sensor	Measure blood oxygen level
Accelerometer	Detect falls or movement
Temperature Sensor	Monitor body temperature

Integration Considerations

- **Sensor Accuracy & Precision**
- **Power Consumption**
- **Interface Type**: Analog, I2C, SPI, UART
- **Sampling Rate**: Real-time responsiveness
- **Calibration & Drift**

THANK YOU

RTOS (Real-Time Operating System) in Cyber-Physical Systems (CPS)

PREPARED BY,
S.BABITHA,AP,
IT,HITS.

DEFN

- A **Real-Time Operating System (RTOS)** is a type of operating system designed to **execute tasks within strict timing constraints**, making it a core component of many **Cyber-Physical Systems (CPS)** where **deterministic, predictable behavior** is crucial.

Key Characteristics of RTOS

Feature	Description
Determinism	Guarantees that high-priority tasks meet deadlines (real-time behavior)
Preemptive Multitasking	Allows switching between tasks based on priority
Low Latency	Fast context switching and interrupt handling
Task Prioritization	Ensures time-critical tasks are served first
Minimal Footprint	Lightweight and suitable for embedded hardware
Support for Synchronization	Semaphores, mutexes, message queues for safe task interaction

RTOS Architecture in CPS

- +-------------------------+
- | Application Code |
- +-------------------------+
- | RTOS Services |
- | - Scheduler |
- | - Timers |
- | - IPC (Queues, Mutexes) |
- | - Memory Mgmt |
- +-------------------------+
- | Hardware Abstraction |
- +-------------------------+
- | Device Drivers & HAL |
- +-------------------------+
- | Hardware |
- +-------------------------+

Common RTOS Used in CPS

RTOS	Features	Typical CPS Use
FreeRTOS	Open source, small footprint	IoT devices, wearables
VxWorks	Certified for safety-critical systems	Aerospace, automotive
Zephyr	Scalable, IoT-focused	Smart sensors, medical CPS
QNX	POSIX-compliant, fault-tolerant	Automotive infotainment, ADAS
RTEMS	Space-grade, real-time	Satellites, space probes
TI-RTOS	Optimized for TI microcontrollers	Industrial automation

APPLICATION OF RTOS IN CPS

CPS Domain	RTOS Role
Automotive	Real-time control of ABS, engine, and ADAS
Industrial Automation	Control of PLCs, motion systems, and robots
Medical Devices	Real-time monitoring (e.g., heart rate, insulin delivery)
Aerospace	Avionics, mission-critical flight systems
Smart Grids	Power control, fault detection systems

Why RTOS in CPS?

- Ensures **real-time guarantees** for physical control
- **Improves reliability** in mission-critical systems
- Enables **modular multitasking** in resource-constrained environments
- Supports **scalability** from microcontrollers to edge servers

Challenges in RTOS Development

- Hard real-time guarantees require **careful task scheduling**
- Requires **deep understanding of hardware constraints**
- Debugging timing and concurrency issues is complex
- Safety certifications (e.g., ISO 26262, DO-178C) can be demanding.


Thank You
So Much

WIRELESS HART

PREPARED BY,
S.BABITHA,AP,
IT,HITS.

DEFN

- **WirelessHART** (Wireless Highway Addressable Remote Transducer Protocol) is a **wireless communication protocol** specifically designed for **industrial CPS applications**, especially in **process automation and control systems**.
- It extends the **wired HART protocol** by adding wireless capabilities while maintaining compatibility with existing HART devices.

🔑 Key Features of WirelessHART

Feature	Description
Based on IEEE 802.15.4	Operates in the 2.4 GHz ISM band, using 802.15.4 physical layer (same as Zigbee).
Mesh Networking	Supports self-organizing and self-healing mesh topology for high reliability.
Time-Synchronized	Uses TDMA (Time Division Multiple Access) to schedule communications and avoid collisions.
Secure	End-to-end encryption, authentication, and message integrity using AES-128.
Backwards-Compatible	Works with legacy HART devices via gateways.
Deterministic Communication	Supports real-time data transmission with bounded latency.

Typical Architecture of a WirelessHART Network

- [Field Devices] ↔ [Routers/Repeater Nodes] ↔ [WirelessHART Gateway] ↔ [Host System / Controller]
- **Field Devices**: Wireless sensors/actuators (temperature, pressure, flow meters)
- **Routers**: Relay data to extend range and improve reliability
- **Gateway**: Converts WirelessHART to standard protocols (Modbus, Ethernet, etc.)
- **Host System**: SCADA, DCS, or cloud controller

Applications in CPS

Application Area	Examples
Process Automation	Oil refineries, chemical plants, gas pipelines
Industrial Monitoring	Pressure, flow, temperature sensing in hazardous or hard-to-reach areas
Smart Manufacturing	Asset tracking, machine health diagnostics
Environmental Monitoring	Air quality and emissions data collection

Advantages of WirelessHART in CPS

- No need for extensive cabling – reduces cost and complexity
- High reliability in noisy industrial environments
- Supports both monitoring and control
- Energy-efficient design for battery-operated devices
- Scalable and easily expandable network

Limitations

- Limited bandwidth (suitable for low-data-rate applications)
- Operates in the crowded 2.4 GHz band (possible interference)
- Latency not suitable for ultra-time-critical systems


Thank
You

ACTUATORS

PREPARED BY,

S.BABITHA,AP,

IT,HITS.

DEFINITION

- **Actuators** are devices that convert control signals (usually electrical) into physical action (movement, heat, light, etc.).
- In CPS, they **bridge the gap between the cyber world and the physical world**, enabling systems to **influence their environment** based on sensor inputs and computational decisions

Role of Actuators in CPS

- Receive commands from the processor/controller
- Translate electrical signals into physical actions
- Execute control decisions (e.g., turning a valve, moving a robot arm)

Types of Actuators in CPS

Actuator Type	Function	Examples	CPS Applications
Electric Motors	Produce rotation or linear motion	DC motors, stepper motors, servo motors	Robotics, drones, vehicles
Solenoids	Create linear motion using electromagnetic force	Door locks, valves	Smart locks, fluid control
Hydraulic Actuators	Use fluid pressure to produce force/motion	Cylinders, motors	Heavy industrial machinery
Pneumatic Actuators	Use compressed air to drive motion	Air cylinders	Automation, manufacturing
Thermal Actuators	Generate heat or motion using temperature changes	Thermostats, shape memory alloys	HVAC, biomedical systems
Piezoelectric Actuators	Precise micro-movements via electric charge	Piezo motors	Precision positioning, optics
Light/LED Actuators	Emit light based on signals	LEDs, laser modules	Visual feedback, smart lighting
Sound Actuators	Generate sound or alerts	Buzzers, speakers	Alarms, voice notifications

Example: Smart Home CPS Actuators

Device	Actuator Used	Function
Smart Door Lock	Solenoid	Lock/unlock the door remotely
Smart Thermostat	Thermal actuator	Regulate temperature
Smart Light	LED	Turn light on/off or change brightness
Smart Fan	Electric motor	Adjust fan speed

Key Characteristics of Actuators

- **Precision**: How accurately it performs the action
- **Speed**: Response time to control signals
- **Power**: Required electrical or mechanical power
- **Interface**: PWM, analog, digital, or communication bus (I2C, CAN)
- **Durability**: Depends on material and use case

Control Flow Example in CPS

- [Sensors] → [Processor/Controller] → [Actuators]
- ↑ Decision Logic ↓
- Environment (Software) Physical Action


thank you

AUTOMOTIVE ETHERNET IN CYBER-PHYSICAL SYSTEMS (CPS)

PREPARED BY,
S.BABITHA,AP,
IT,HITS.

DEFN- **AUTOMOTIVE ETHERNET**

- **Automotive Ethernet** is a high-speed, cost-effective, and scalable in-vehicle networking technology that has become a key enabler for modern **automotive CPS** like **ADAS (Advanced Driver-Assistance Systems)**, **infotainment**, **autonomous driving**, and **vehicle-to-everything (V2X)** communication.
- Automotive Ethernet is a customized version of standard Ethernet (IEEE 802.3) adapted for the **automotive environment**:
- Operates reliably under **harsh conditions** (vibration, temperature, EMI)
- Supports **real-time and time-sensitive communication**
- Uses **single-pair twisted cables** to reduce weight and cost

🔑 Key Features

Feature	Description
High Bandwidth	Speeds from 100 Mbps to 10 Gbps (100BASE-T1, 1000BASE-T1, etc.)
Full-Duplex Communication	Simultaneous send/receive
Reduced Cabling	Single twisted-pair wiring (vs. traditional multi-wire harnesses)
Time-Sensitive Networking (TSN)	Ensures real-time, deterministic communication for safety-critical systems
Interoperability	Based on open IEEE standards
Power over Data Lines (PoDL)	Delivers power and data over a single pair

Automotive Ethernet Architecture in CPS

```
• Camera] ─┐
•         ├───> [Switch] ───> [Central ECU] ───> [Cloud Gateway]
• [LIDAR] ──┘        │
• [Infotainment] ─────────┘
```

- **End devices**: Sensors (camera, radar, lidar), ECUs, infotainment
- **Ethernet Switches**: Central nodes that manage and route data
- **Gateway ECU**: Bridges between CAN/FlexRay/LIN and Ethernet networks

Applications in Automotive CPS

System	Function
ADAS	Real-time sensor fusion from radar, lidar, cameras
Infotainment	High-speed video and audio streaming
Vehicle Diagnostics	OTA (over-the-air) updates and system health monitoring
Autonomous Driving	Connects AI compute units, sensor data, and actuators
V2X Communication	Enables vehicle-to-vehicle/infrastructure data exchange

🔄 Comparison: CAN vs. Automotive Ethernet

Feature	CAN	Automotive Ethernet
Speed	Up to 1 Mbps	Up to 10 Gbps
Payload	8 bytes	1500+ bytes
Topology	Bus	Star/switch-based
Use	Control systems	Data-intensive systems (ADAS, infotainment)
Real-Time Support	Yes (via priority)	Yes (via TSN)

Advantages of Automotive Ethernet

- Scales easily with growing data demands
- Reduces cable weight and cost
- Supports both **safety-critical** and **high-bandwidth** functions
- Standardized and interoperable

Challenges

- Requires EMI shielding and robust connectors
- Complex integration with legacy networks (CAN, LIN, FlexRay)
- New diagnostic and security considerations


Thank
You

CAN (CONTROLLER AREA NETWORK) IN CYBER-PHYSICAL SYSTEMS (CPS)

PREPARED BY,
S.BABITHA,AP,
IT,HITS.

DEFN

- **CAN** is a **robust, real-time, and reliable communication protocol** designed for embedded control systems.
- Originally developed by **Bosch** for automotive applications, CAN is now widely used in **industrial CPS**, **robotics**, **medical devices**, and **automation systems**.

🔑 KEY FEATURES OF CAN

Feature	Description
Multi-Master Bus	Any node can initiate communication if the bus is free
Broadcast Communication	Messages are broadcast to all devices; receivers decide relevance
Message-Based Protocol	Communication is based on messages, not node addresses
Real-Time Support	Message priority via identifiers ensures time-critical data gets through first
Error Detection & Handling	Built-in CRC checks, automatic retransmission, error confinement
Deterministic	Ensures predictable communication for real-time control

CAN BUS PHYSICAL AND DATA LINK LAYER

- **Medium**: Twisted-pair cable (differential signaling)
- **Speed**: Up to 1 Mbps (standard CAN); CAN FD supports higher speeds
- **Message Frame**: Includes identifier, control, data (0–8 bytes for classic CAN), and CRC

TYPICAL CAN NETWORK ARCHITECTURE IN CPS

```
[Node A] ---+
            |
[Node B] ---+--- CAN Bus (Twisted Pair)
            |
[Node C] ---+
```

Each **node** contains:

Microcontroller with built-in or external **CAN controller**

Transceiver for differential signaling

APPLICATIONS OF CAN IN CPS

CPS Domain	Application Examples
Automotive CPS	Engine control units (ECU), airbags, ABS, infotainment
Robotics	Real-time coordination between motor controllers and sensors
Industrial Automation	PLC communication, motor drives, factory floor devices
Medical CPS	Medical imaging, ventilators, infusion pumps
Aerospace	Avionics systems and flight control computers

ADVANTAGES OF CAN IN CPS

- Highly reliable and fault-tolerant
- Real-time message prioritization
- Low-cost cabling and hardware
- Supports up to 8 bytes per message (CAN FD supports more)

LIMITATIONS

- ✓ Limited data payload (classic CAN: 8 bytes/message)
- ✓ Not suitable for high-bandwidth multimedia data
- ✓ Limited to short-range (typically < 40m at high speeds)

VARIANTS OF CAN

Variant	Description
CAN 2.0A/B	Standard/extended frame format (11/29-bit identifiers)
CAN FD (Flexible Data Rate)	Allows longer payloads (up to 64 bytes) and higher speed
CANopen / DeviceNet	Higher-layer protocols for industrial use, built on CAN


Thank
You

CPS

PREPARED BY,

S.BABITHA,AP,

IT,HITS.

Cyber-Physical Systems (CPS) — Overview

- **Cyber-Physical Systems (CPS)** are systems that tightly integrate **computing (cyber)** components with **physical processes**.
  ➡ In CPS, sensors monitor physical conditions, controllers compute actions, and actuators influence the physical world — all working together in a feedback loop.
- 👉 CPS connects the physical world (machines, devices, environment) and the cyber world (software, computation, communication) in real time.

Main Components

- **Sensors:** Collect data from the physical environment (e.g., temperature, pressure, motion).
- **Actuators:** Carry out actions (e.g., move a robotic arm, adjust a valve).
- **Embedded systems / controllers:** Process data, make decisions, and control actuators.
- **Communication network:** Connect components (e.g., wired, wireless, IoT protocols).
- **Software:** Implements control logic, data analytics, and human-machine interfaces.

Key Features

- ✅ Tight integration of computation and physical processes
  ✅ Real-time monitoring and control
  ✅ Networked and distributed system architecture
  ✅ Adaptability and autonomy in decision-making
  ✅ Safety-critical and high-reliability requirements

Applications

- **Smart grids:** Real-time monitoring and control of electricity supply and demand.
- **Autonomous vehicles:** Self-driving cars that sense, compute, and act in real-time.
- **Medical CPS:** Implanted devices, robotic surgery, remote monitoring.
- **Industrial automation / IIoT:** Smart factories, predictive maintenance.
- **Smart buildings / cities:** Automated energy, water, traffic, and security systems.

Challenges

- **Cybersecurity threats:** Vulnerability to attacks that could cause physical harm.
- **Real-time performance:** Strict timing requirements to ensure safety and reliability.
- **Complexity of integration:** Hardware, software, and network components must work seamlessly.
- **Interoperability:** Devices and systems from different vendors need to communicate effectively.
- **Regulatory and safety compliance:** Especially in critical sectors (healthcare, transport, energy).

Examples

- Drones that autonomously navigate and perform tasks
- Smart transportation systems (e.g., adaptive traffic signals)
- Industrial robots with vision systems
- Automated irrigation systems in agriculture

Aspect	Cyber-Physical Systems (CPS)	Internet of Things (IoT)
Focus	Tight integration of computation, communication, and control with physical processes	Connecting devices to the internet for data exchange
Control loop	Emphasizes real-time feedback loops between cyber and physical components	Primarily focused on data collection, sharing, and remote monitoring
Latency requirements	Often requires real-time or near-real-time response (e.g., industrial robots, medical CPS)	Many applications tolerate higher latency (e.g., smart home devices, fitness trackers)
Autonomy	High — CPS systems often operate autonomously with decision-making ability	Low to medium — IoT devices often collect and send data, decisions made at cloud or server level
Examples	Autonomous vehicles, industrial robots, smart grid control systems, medical CPS	Smart appliances, wearable health trackers, smart meters, connected thermostats
Typical architecture	Closed-loop system: sensors → compute/control → actuators → physical process → sensors	Open-loop or loosely coupled systems: devices → cloud/server → user interface
Main objective	Control and coordination of physical processes	Monitoring, data collection, communication
Criticality	Often safety-critical (failure may cause harm or damage)	Generally less safety-critical (though exceptions exist, e.g., smart grid IoT)


THANK YOU!
SO MUCH