Unit-V

IoT Applications

5.1 Home Automation: Smart lighting, Smart Appliances, Smoke/Gas Detector.

5.2 Smart Cities: Intelligent Traffic systems, Smart Parking, Smart water management

5.3 Energy: Smart Grids, Renewable Energy Systems.

5.4 Agriculture: Green House Control.

5.5 Health: Health & fitness monitoring, Wearable electronics.

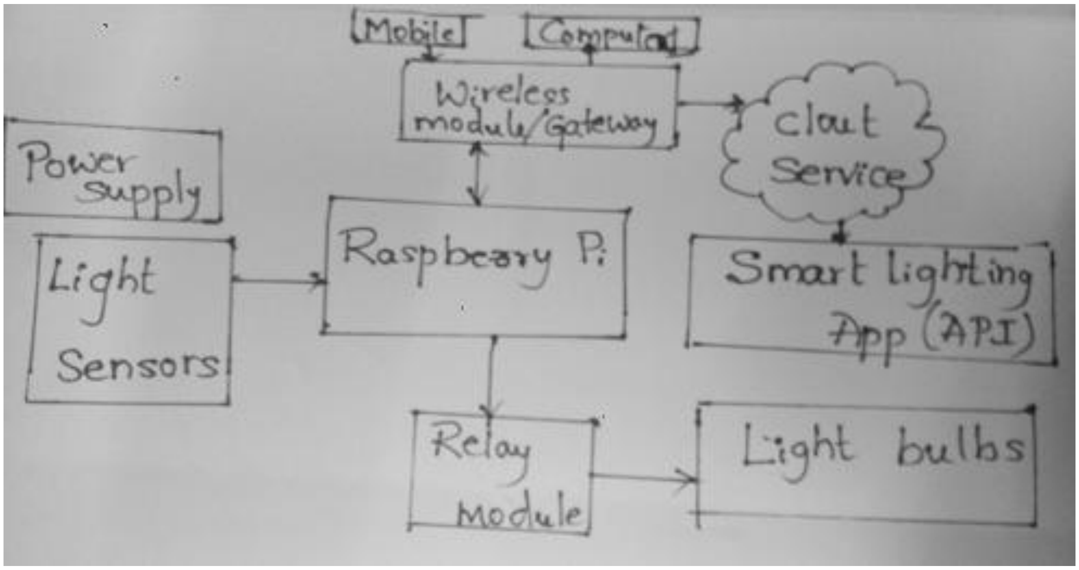
Home automation system:

Types of home automation.

- Smart Lighting control
- Smart appliances
- Smoke/Gas Detector
- HVAC,
- outdoor lawn irrigation
- Kitchen appliances and security systems.
- Automated Appliance Control System.
- Automated Door and Gate Access System.
- Automated Blinds System.
- Automated Irrigation Control Systems.

- ## Smart lighting

  An IoT-based smart lighting system for home automation typically consists of the following blocks:

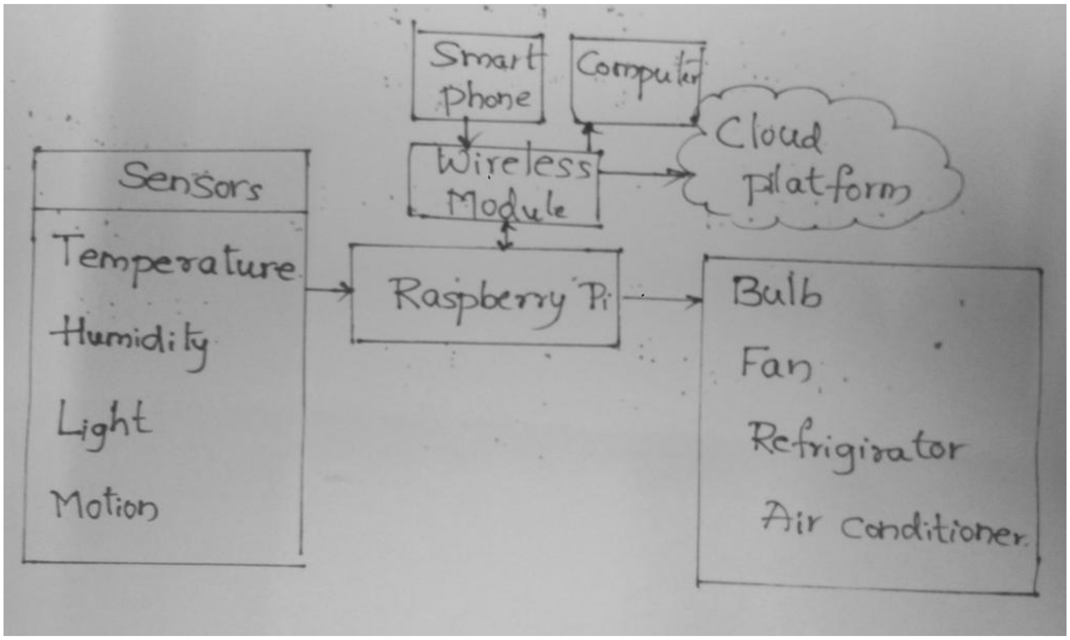


- **Light Sensors**: Light sensors are used to detect the amount of light in a room or area.
- **Raspberry Pi:** The Raspberry Pi is a small computer that acts as an IoT gateway and connects the light sensors to the internet.
- **Cloud Service**: The cloud service provides a centralized management platform for the smart lighting system. It is where all the data is stored, and the user can access it from anywhere with an internet connection.
- **Smart Lighting App**: The smart lighting app provides a user interface to monitor and control the smart lighting system. Users can view the real-time status of lights, control the brightness of lights, and create lighting schedules.
- **Python Programming Language**: Python is a programming language that can be used to write scripts for the Raspberry Pi to analyze the light sensor data, communicate with the cloud service, and control the lighting system.
- **LED Bulbs**: LED bulbs are used as the light source for the smart lighting system.
- **Relay Module**: A relay module can be used to control the on/off state of the LED bulbs.
- **Power Supply**: A power supply is required to power the Raspberry Pi, light sensors, LED bulbs, and relay module.

Overall, the IoT-based Smart Lighting System using Raspberry Pi can help to reduce energy consumption and provide a seamless lighting experience for users. By using real-time data and intelligent decision-making algorithms, the system can optimize the use of lights, reduce energy waste, and provide the right amount of lighting for each room or area. The Raspberry Pi and its peripherals can be programmed using Python, making it a cost-effective and flexible solution for building a smart lighting system.

- ## IoT based Smart Appliances:

The block diagram of an IoT based smart appliance system for home automation using Raspberry Pi can be divided into four main parts: Sensor Module, Raspberry Pi Board, Cloud Server, and Mobile Application.



- **Sensor Module**:This module consists of various sensors that are used to detect different environmental parameters such as temperature, humidity, light, and motion. When a change in any of these parameters is detected, the sensor sends a signal to the Raspberry Pi board.

- **Raspberry Pi Board**: The Raspberry Pi board is a small computer that runs the software required to control the smart appliances and send data to the cloud server. The Raspberry Pi board also contains an Ethernet port, which allows it to connect to the internet.

- **Cloud Server**: This server receives data from the Raspberry Pi board and stores it in a database. The server also runs software that analyzes the data and generates commands to control the smart appliances.

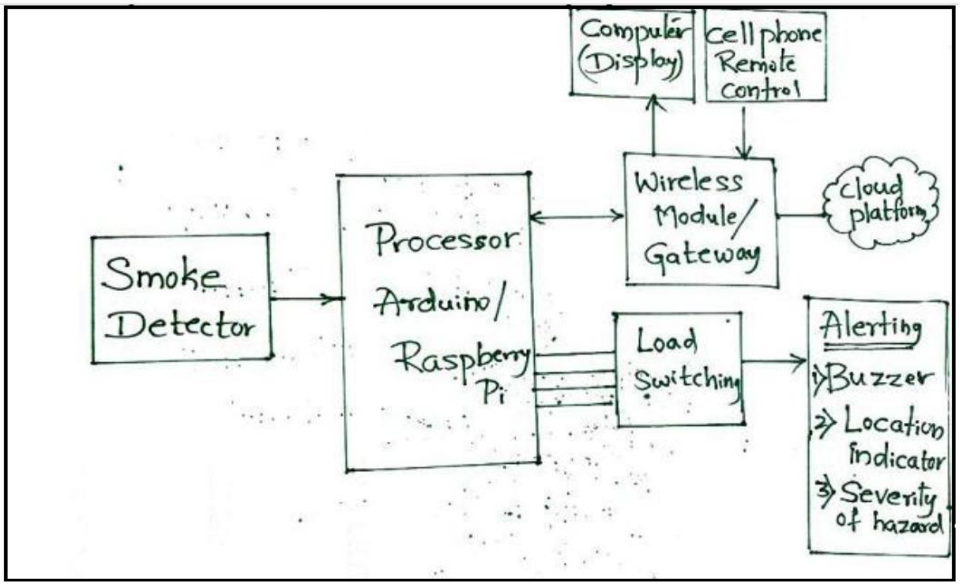
- **Mobile Application**: This application can be installed on a smart phone and is used to monitor and control the smart appliances remotely. The mobile application connects to the cloud server and sends commands to the server to control the appliances. The mobile application also displays the current status of the appliances

.

An IoT based smart appliance system for home automation using Raspberry Pi is made up of a sensor module that detects changes in environmental parameters, a Raspberry Pi board that processes the data and sends it to a cloud server, a cloud server that stores the data and generates commands to control the smart appliances, and a mobile application that allows users to monitor and control the appliances remotely.

- ## IoT Based smoke/Gas detector

An IoT based smoke detector is a device that uses sensors to detect smoke and sends notifications/alerts to connected devices or the cloud via the internet. Here is a block diagram of an IoT based smoke detector:



The block diagram of an IoT based smoke/gas detector using Raspberry Pi can be divided into four main parts: Sensor Module, Raspberry Pi Board, Cloud Server, and Mobile Application.

- **Sensor Module**: This module consists of a gas/smoke sensor, which is used to detect the presence of smoke or gas in the environment. When smoke or gas is detected, the sensor sends a signal to the Raspberry Pi board.

- **Raspberry Pi Board** :The Raspberry Pi board is a small computer that runs the software required to control the sensor module and send data to the cloud server. The Raspberry Pi board also contains an Ethernet port, which allows it to connect to the internet.

- **Cloud Server**: This server receives data from the Raspberry Pi board and stores it in a database. The server also runs software that analyzes the data and generates alerts if smoke or gas is detected.

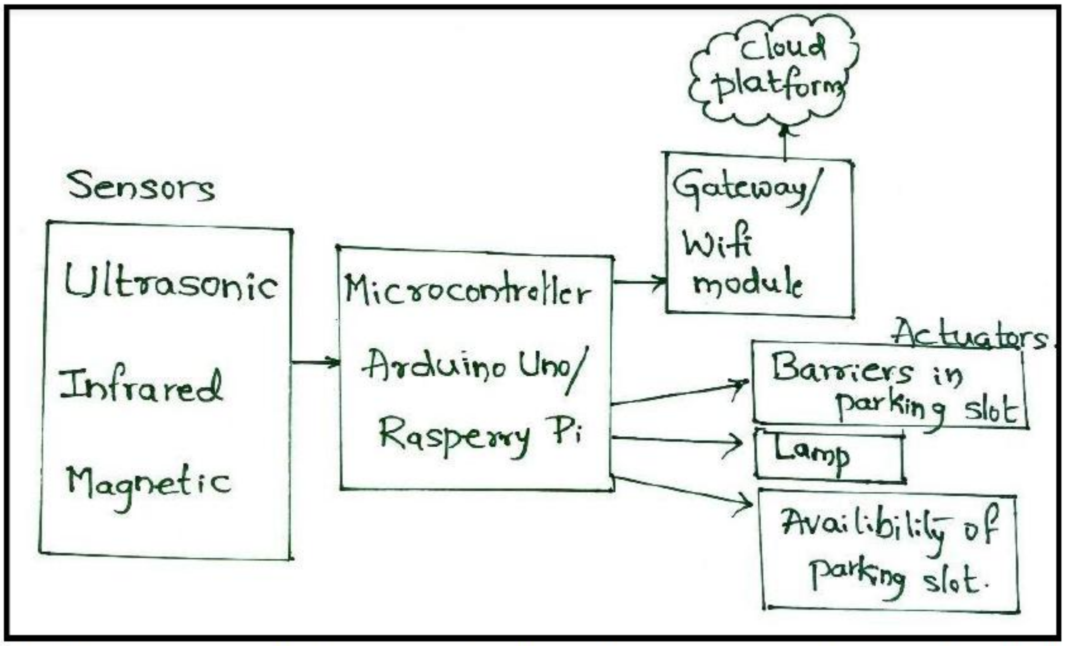
- **Mobile Application**: This application can be installed on a smartphone and is used to monitor the smoke or gas detector remotely. The mobile application connects to the cloud server and displays the current status of the smoke or gas detector.

Thus an IoT based smoke/gas detector using Raspberry Pi is made up of a sensor module that detects the presence of smoke or gas, a Raspberry Pi board that processes the data and sends it to a cloud server, a cloud server that stores the data and generates alerts if necessary, and a mobile application that allows users to monitor the status of the detector remotely.

Smart Cities

• Smart parking systems

The block diagram of an IoT device for smart parking systems generally consists of the following components:



An IoT-based Smart Parking System using Raspberry Pi typically consists of the following blocks:

- Sensors : Ultrasonic, Infrared and Magnetic sensors are used to detect the presence of a vehicle in a parking spot.
- Raspberry Pi: The Raspberry Pi is a small computer that acts as an IoT gateway and connects the ultrasonic sensors to the internet.
- Cloud Service: The cloud service provides a centralized management platform for the smart parking system. It is where all the data is stored, and the user can access it from anywhere with an internet connection.
- Smart Parking App: The smart parking app provides a user interface to monitor and control the smart parking system. Users can view the real-time availability of parking spots, book parking spots in advance, and pay for parking.

- Python Programming Language: Python is a programming language that can be used to write scripts for the Raspberry Pi to analyze the ultrasonic sensor data, communicate with the cloud service, and control the parking system.

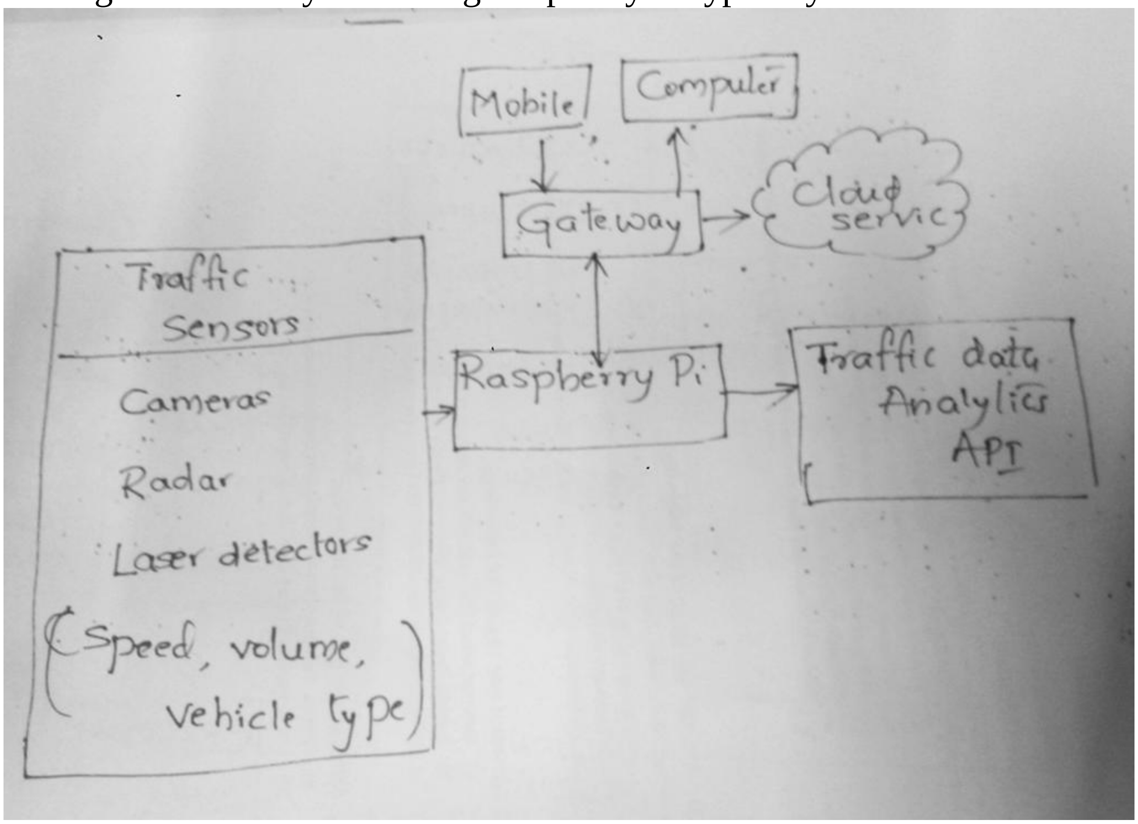
- LED Indicator: LED indicators can be used to indicate the availability of a parking spot. A green LED light can indicate that a spot is available, while a red LED light can indicate that a spot is occupied.

- Electric Lock: An electric lock can be used to prevent unauthorized access to a parking spot.

- Power Supply: A power supply is required to power the Raspberry Pi, ultrasonic sensors, LED indicators, and electric locks.

• Intelligent Traffic System

An IoT-based Intelligent Traffic System using Raspberry Pi typically consists of the following blocks:



- Traffic Sensors: Traffic sensors include cameras, radar, and laser detectors, which collect traffic data such as speed, volume, and vehicle type.

- Raspberry Pi: The Raspberry Pi is a small computer that acts as an IoT gateway and connects the traffic sensors to the internet.

- Cloud Service: The cloud service provides a centralized management platform for the intelligent traffic system. It is where all the data is stored, and the user can access it from anywhere with an internet connection.

- Traffic Data Analytics: Traffic data analytics is a tool that analyzes the traffic data collected by the sensors to generate insights such as traffic volume, congestion, and accidents.

- Intelligent Traffic Management System: The intelligent traffic management system is a software system that uses the insights generated by the traffic data analytics to control traffic flow. It can control traffic lights, change speed limits, or reroute traffic to optimize traffic flow.

- Mobile App / Web App: The mobile app or web app provides a user interface to monitor and control the intelligent traffic system. Users can view real-time traffic data, receive alerts on traffic incidents, and change traffic flow parameters.

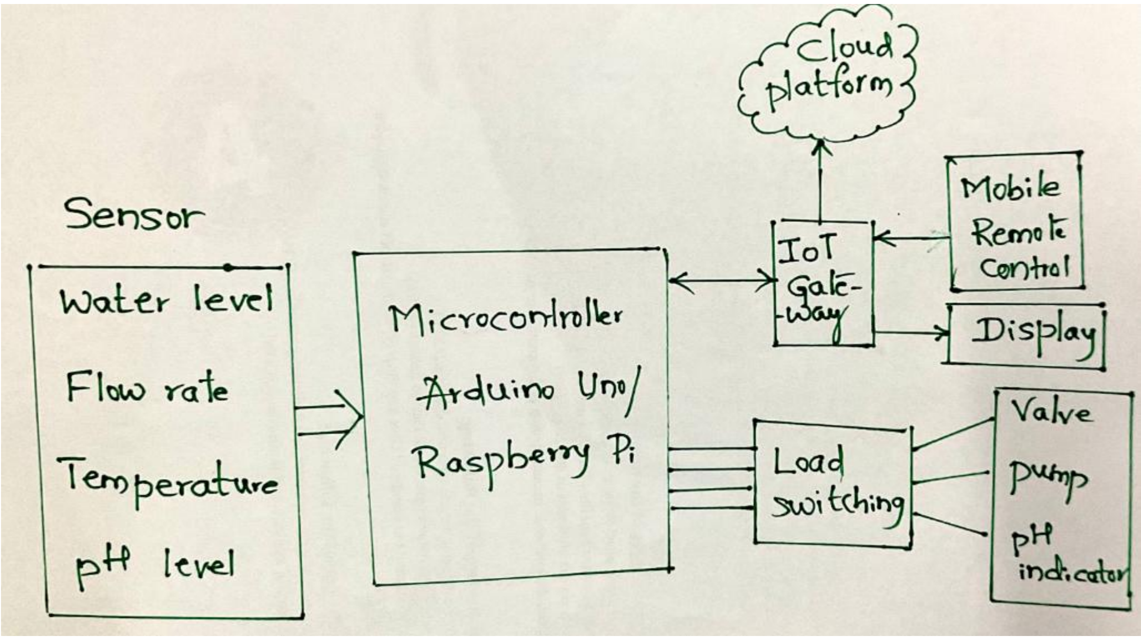
- Raspberry Pi Camera Module: The Raspberry Pi camera module is a camera that can be used to capture real-time images and videos of the traffic flow.

- OpenCV: OpenCV is an open-source computer vision library that can be used to analyze the images and videos captured by the Raspberry Pi camera module.

- Python Programming Language: Python is a programming language that can be used to write scripts for the Raspberry Pi to analyze the traffic data, control the traffic lights, and communicate with the cloud service.

- ## Smart water management:

The block diagram of IoT-based Smart water management system consists of various components that work together to ensure efficient and optimized water usage. The following are the components of the block diagram:



- Water Sensors: The system uses various types of sensors such as flow sensors, pressure sensors, pH sensor and water level sensors that collect data related to water usage, quality, and availability.

- Raspberry Pi: The Raspberry Pi acts as a hub that receives data from the sensors, processes it, and sends it to the cloud for further analysis.

- Connectivity: The system uses various connectivity options such as Wi-Fi, Bluetooth, or cellular networks to connect the sensors, Raspberry Pi, and cloud.

- Cloud Computing: The cloud analyzes the data collected from the sensors and provides insights on water usage, quality, and availability.

- User Interface: The user interface allows users to access the data and insights provided by the cloud through a web-based interface or a mobile application.

- Automation: The system can be automated based on the insights provided by the cloud. For example, the system can be programmed to automatically reduce the water flow in case of low water pressure to prevent wastage.

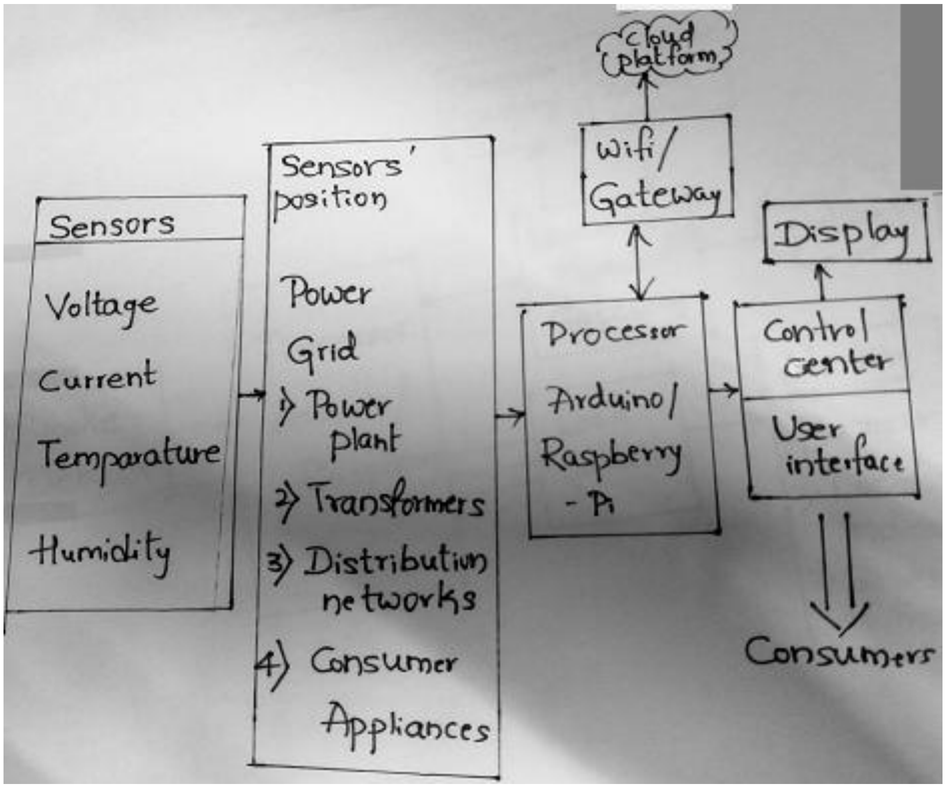
- Actuators: The system uses actuators such as valves and pumps to control the water flow and pressure based on the insights provided by the cloud.

  Overall, the block diagram of IoT-based Smart water management system shows how the components work together to ensure efficient water usage, prevent wastage, and promote sustainable practices.

Energy

- ## Smart Grids

The block diagram of an IoT-based Smart Grid system using Raspberry Pi is shown below:



IoT Smart Grid System using Raspberry Pi Block Diagram

The major components of this system are as follows:

- Power Generation: This component represents the power generation sources such as solar, wind, hydroelectric, or thermal power plants.

- Transmission System: The generated power is transmitted from the power plants to the substation using high voltage transmission lines.

- Substation: The substation transforms the high voltage power to low voltage power and distributes it to the distribution network.

- Distribution Network: This component represents the distribution network, which includes transformers, distribution lines, and other devices used to distribute the power to the end-users.

- IoT Devices: These are the devices that are used to collect and transmit data from the distribution network to the central server. These devices include smart meters, sensors, and other IoT devices that are used to monitor and control the power distribution system.

- Raspberry Pi: The Raspberry Pi serves as the central server in the Smart Grid system. It receives data from the IoT devices and processes it to optimize the energy distribution and consumption.

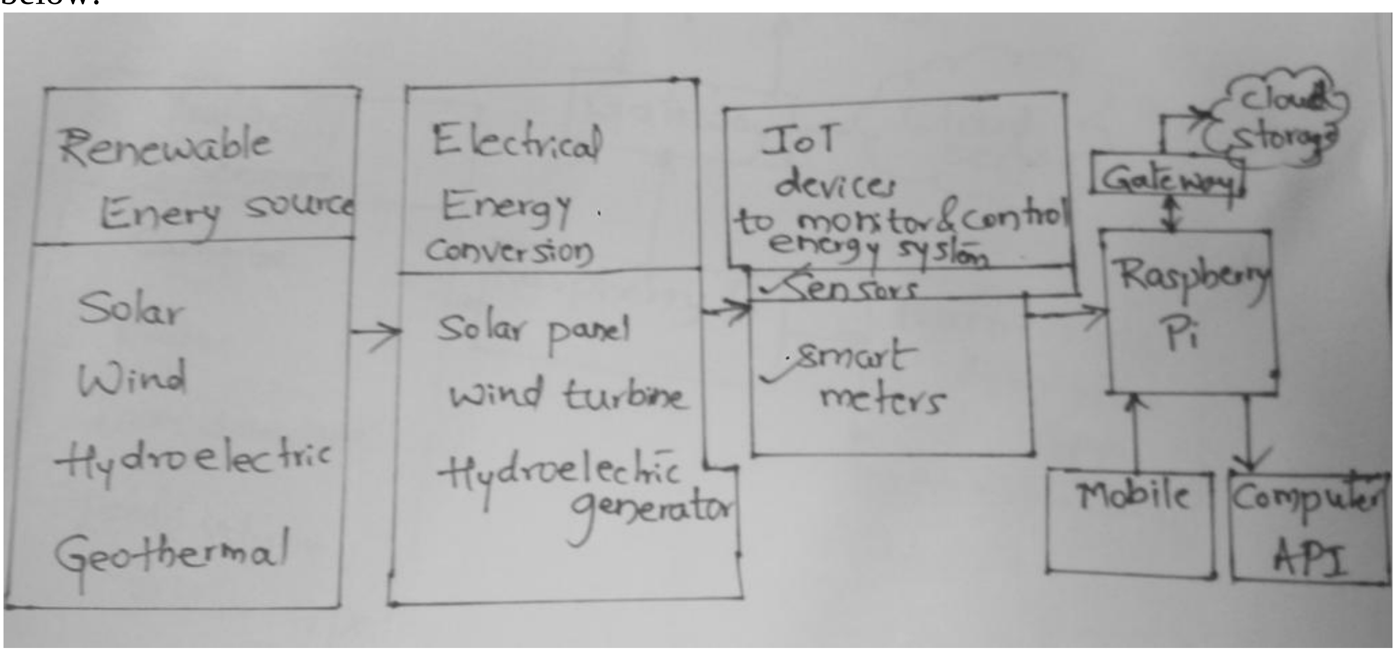
- IoT Network: This component represents the IoT network that provides connectivity between the IoT devices and the central server. It uses various wireless communication protocols such as Wi-Fi, Zigbee, or LoRaWAN to transmit data.

- Cloud Storage: The cloud storage is used to store the data collected from the IoT devices. It enables the data to be accessed remotely and provides a backup in case of data loss.

- User Interface: The user interface allows end-users to monitor their energy consumption and control their electrical appliances remotely using a smartphone or a web application.

- ## Renewable Energy Systems.

The block diagram of an IoT-based Renewable Energy System using Raspberry Pi is shown below:



The major components of this system are as follows:

- Renewable Energy Source: This component represents the renewable energy sources such as solar, wind, hydroelectric, or geothermal energy.

- Energy Conversion System: The renewable energy source is converted into electrical energy using an energy conversion system such as a solar panel, wind turbine, or hydroelectric generator.

- Power Conditioning System: The electrical energy generated is conditioned using a power conditioning system such as an inverter or rectifier to match the voltage and frequency of the grid.

- IoT Devices: These are the devices that are used to collect and transmit data from the renewable energy system to the central server. These devices include sensors, smart meters, and other IoT devices that are used to monitor and control the energy system.

- Raspberry Pi: The Raspberry Pi serves as the central server in the Renewable Energy System. It receives data from the IoT devices and processes it to optimize the energy generation and consumption.

- IoT Network: This component represents the IoT network that provides connectivity between the IoT devices and the central server. It uses various wireless communication protocols such as Wi-Fi, Zigbee, or LoRaWAN to transmit data.

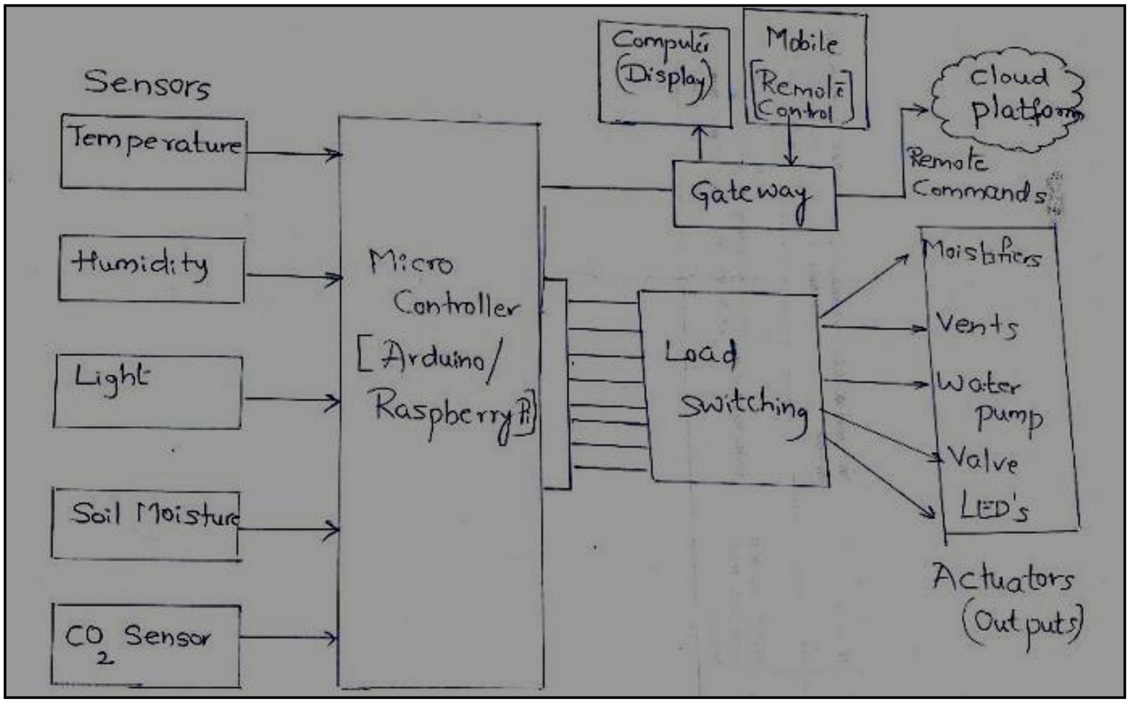
- Cloud Storage: The cloud storage is used to store the data collected from the IoT devices. It enables the data to be accessed remotely and provides a backup in case of data loss.

- User Interface: The user interface allows end-users to monitor their energy consumption and control their electrical appliances remotely using a smartphone or a web application.

Agriculture

- ## Green House Control.

The block diagram of an IoT-based Green House Control system using Raspberry Pi is shown below:



IoT Green House Control System using Raspberry Pi Block Diagram

The major components of this system are as follows:

- Sensors: This component represents the various sensors used to monitor the environmental conditions inside the greenhouse. These sensors can include temperature, humidity, light, soil moisture, and CO2 sensors.

- Actuators: The actuators are the devices used to control the environmental conditions inside the greenhouse. These can include devices such as fans, heaters, humidifiers, vents, pumps and lights.

- Raspberry Pi: The Raspberry Pi serves as the central server in the Green House Control system. It receives data from the sensors and sends commands to the actuators to maintain the desired environmental conditions inside the greenhouse.

- IoT Network: This component represents the IoT network that provides connectivity between the sensors, actuators, and the central server. It uses various wireless communication protocols such as Wi-Fi, Zigbee, or LoRaWAN to transmit data.

- Cloud Storage: The cloud storage is used to store the data collected from the sensors. It enables the data to be accessed remotely and provides a backup in case of data loss.

- User Interface: The user interface allows end-users to monitor and control the environmental conditions inside the greenhouse remotely using a smartphone or a web application.

In summary, an IoT-based Green House Control system using Raspberry Pi provides an efficient and sustainable solution for monitoring and controlling the environmental conditions inside the greenhouse, which can help to optimize plant growth, increase crop yield, and promote environmental sustainability.

IOT application used in fitness monitory system.

1. **Remote patient monitoring System -**
   - Remote patient monitoring is the most common application of IOT devices for healthcare. IOT devices can automatically collect health metrics like heart rate, blood pressure, temperature, and more from patients who are not physically present in a healthcare facility, eliminating the need for patients to travel to the providers, or for patients to collect it themselves.
   - When an IOT device collects patient data, it forwards the data to a software application where healthcare professionals and/or patients can view it. Algorithms may be used to analyse the data in order to recommend treatments or generate alerts.
     For example, an IOT sensor that detects a patient's unusually low heart rate may generate an alert so that healthcare professionals can intervene.
2. **Glucose monitoring-**
   - For the more than million diabetes, glucose monitoring has traditionally been difficult. Not only is it inconvenient to have to check glucose levels and manually record results, but doing so reports a patient's glucose levels only at the exact time the test is provided. If levels fluctuate widely, periodic testing may not be sufficient to detect a problem.
   - IOT devices can help address these challenges by providing continuous, automatic monitoring of glucose levels in patients. Glucose monitoring devices eliminate the need to keep records manually, and they can alert patients when glucose levels are problematic.
3. **Heart-rate monitoring**
   - Like glucose, monitoring heart rates can be challenging, even for patients who are present in healthcare facilities. Periodic heart rate checks don't guard against rapid fluctuations in heart rates, and conventional devices for continuous cardiac monitoring used in hospitals require patients to be attached to wired machines constantly, impairing their mobility.
   - Today, a variety of small IOT devices are available for heart rate monitoring, freeing patients to move around as they like while ensuring that their hearts are monitored continuously. Guaranteeing ultra-accurate results remains somewhat of a challenge, but most modern devices can deliver accuracy rates of about 90 percent or better.
4. **Hand hygiene monitoring**
   - Traditionally, there hasn't been a good way to ensure that providers and patients inside a healthcare facility washed their hands properly in order to minimize the risk of spreading contagion.

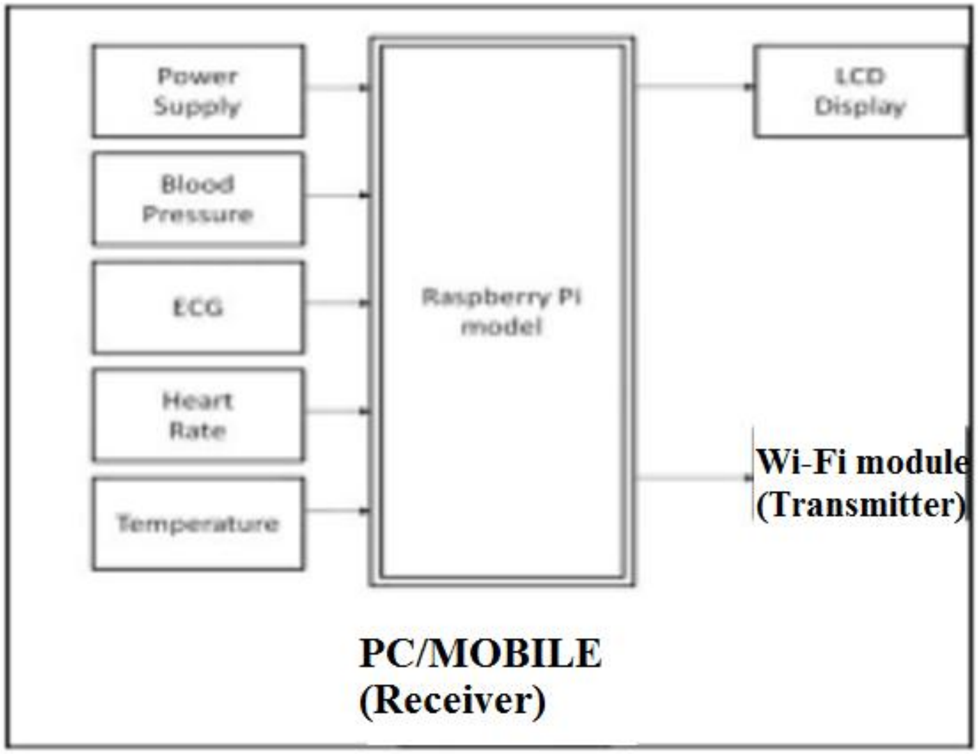
- Today, many hospitals and other health care operations use IOT devices to remind people to sanitize their hands when they enter hospital rooms. The devices can even give instructions on how best to sanitize to mitigate a particular risk for a particular patient.
- A major shortcoming is that these devices can only remind people to clean theirhands; they can’t do it for them. Still, research suggests that these devices can reduceinfection rates by more than 60 percent in hospitals.

5. **Depression and mood monitoring**
    - Information about depression symptoms and patients’ general mood is another type of data that has traditionally been difficult to collect continuously. Healthcare providers might periodically ask patients how they are feeling, but were unable to anticipate sudden mood swings. And, often, patients don’t accurately report their feelings.
    - “Mood-aware” IOT devices can address these challenges. By collecting and analysing data such as heart rate and blood pressure, devices can infer information about a patient’s mental state. Advanced IOT devices for mood monitoring can even track data such as the movement of a patient’s eyes.
    - The key challenge here is that metrics like these can’t predict depression symptoms or other causes for concern with complete accuracy. But neither can a traditional in-person mental assessment.
6. **Parkinson’s disease monitoring**
    - In order to treat Parkinson’s patients most effectively, healthcare providers must be able to assess how the severity of their symptoms fluctuate through the day.
    - IOT sensors promise to make this task much easier by continuously collecting data about Parkinson’s symptoms. At the same time, the devices give patients the freedom to go about their lives in their own homes, instead of having to spend extended periods in a hospital for observation.

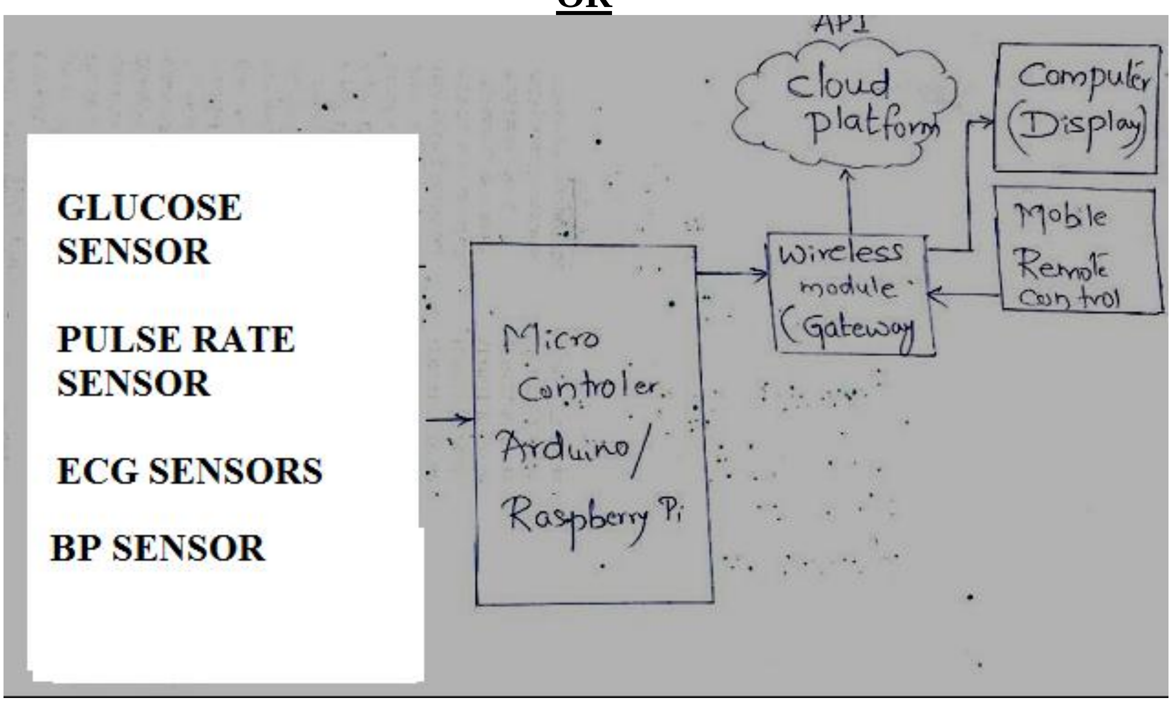
Health

- ## Health & fitness monitoring

  The major components of this system are as follows:



OR



The block diagram of health and fitness monitoring using Raspberry Pi can be divided into the following components:

Wearable electronics: These are the devices worn by the user, such as smartwatches, fitness trackers, and other sensor-based devices that collect health and fitness-related data, including physical activity levels, sleep quality, and heart rate.

Sensors: The wearable devices contain sensors that collect data about the user's health and fitness status. The sensors could be embedded in the device or connected to it externally.

Raspberry Pi: A small, affordable, credit card-sized computer that collects data from wearable devices using sensors. It can be configured with various sensors, such as a heart rate sensor, accelerometer, or gyroscope, to collect data about the user's health and fitness status.

Connectivity: The Raspberry Pi can be connected to the internet using wired or wireless connectivity. This enables the collected data to be transmitted to a central data repository for storage and analysis.

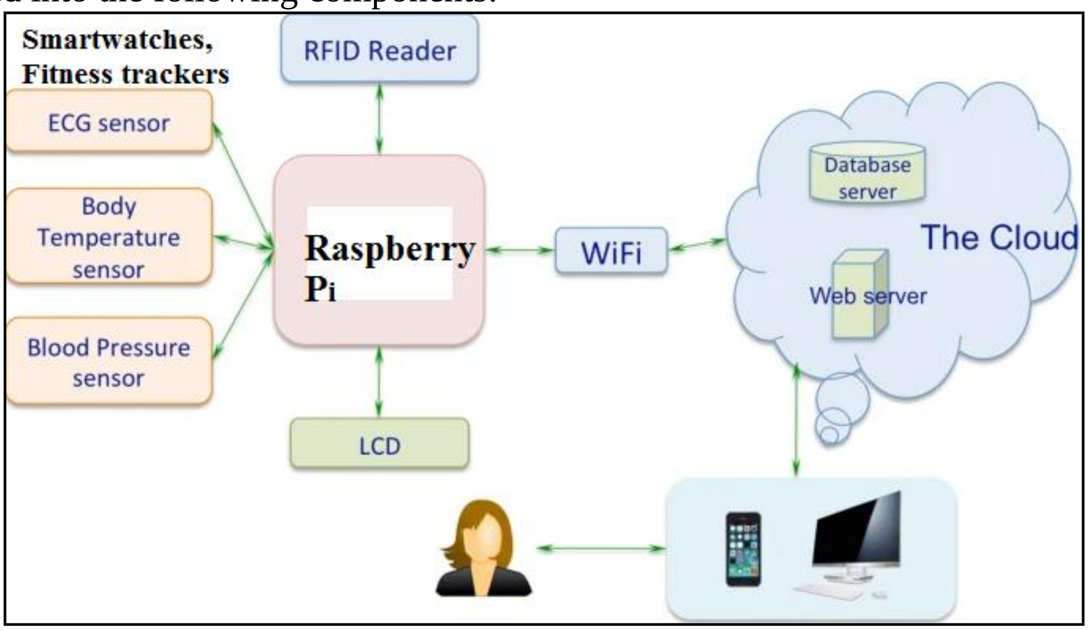
Cloud-based platform or mobile app: The collected data is transmitted to a cloud-based platform or mobile app for analysis and processing. This data can be used to track physical activity levels, monitor sleep quality, and provide personalized recommendations for improving fitness and overall health.

Data analytics: The collected data can be analyzed using advanced data analytics techniques, such as machine learning algorithms, to provide insights into physical activity patterns, sleep quality, and overall health and fitness status.

User interface: The user can access the health and fitness data through a user interface, such as a mobile app or web portal. This allows them to monitor their progress towards fitness goals and receive personalized recommendations for improving their health and fitness.

wearable electronics

The block diagram of IoT-based health monitoring using wearable electronics and Raspberry Pi can be divided into the following components:



- Wearable electronics: These are the devices worn by the user, such as smartwatches, fitness trackers, and other sensor-based devices that collect health-related data, including vital signs such as heart rate, blood pressure, oxygen levels, and physical activity levels.

- Sensors: The wearable devices contain sensors that collect data about the user's health status. The sensors could be embedded in the device or connected to it externally.

- Raspberry Pi: A small, affordable, credit card-sized computer that collects data from wearable devices using sensors. It can be configured with various sensors, such as a heart rate sensor, temperature sensor, blood pressure sensor, or even a camera, to collect data about the user's health status.

- Connectivity: The Raspberry Pi can be connected to the internet using wired or wireless connectivity. This enables the collected data to be transmitted to a central data repository for storage and analysis.

- Cloud-based platform or mobile app: The collected data is transmitted to a cloud-based platform or mobile app for analysis and processing. This data can be used to track health conditions, detect potential health problems, and provide personalized recommendations to improve health and wellness.

- Data analytics: The collected data can be analyzed using advanced data analytics techniques, such as machine learning algorithms, to predict health outcomes, such as the risk of developing a chronic disease or the likelihood of hospital readmission.

In conclusion, the block diagram of IoT-based health monitoring using wearable electronics and Raspberry Pi involves the collection of health-related data from wearable devices using sensors, transmitting the data to the Raspberry Pi for processing and analysis, and transmitting the results to a cloud-based platform or mobile app for personalized recommendations and data analytics.