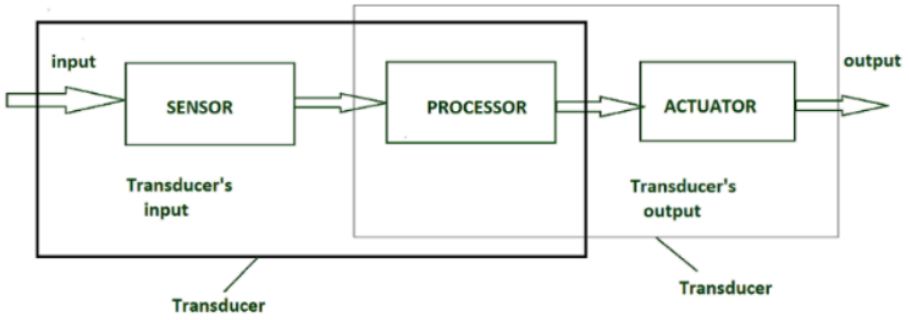
MODULE 4 - IOE

In the Internet of Everything (IoE) and Internet of Things (IoT), devices interact with the physical environment. This interaction requires:

- **Sensors** → to sense physical quantities.
- **Transducers** → to convert one form of energy into another.
- **Actuators** → to perform real-world actions.



Sensors

A sensor is a device that detects physical/chemical/biological changes and converts them into an electrical signal.

Working Principle:

1. A physical quantity (such as temperature, pressure, or light) changes a property of the sensor.
2. This change is converted into an electrical signal (voltage/current/resistance/frequency).
3. A controller (Arduino, Raspberry Pi) processes the signal.

Characteristics:

- **Sensitivity:** How much the output changes per unit change in input.
- **Accuracy:** How close the measurement is to the true value.
- **Range:** Minimum to maximum measurable value.
- **Response Time:** Time taken to respond to changes.

Types of Sensors: (acronym: LUT TRS)

- **Light Sensors:**
    a. Light sensors, also known as **photo sensors**, detect the intensity of light. A commonly used light sensor is the **Light Dependent Resistor (LDR).**
    b. *Example:* Automatic street lights.
- **Ultrasonic Sensors:**
    a. Ultrasonic sensors work similarly to **SONAR or RADAR**, using **high-frequency sound waves** to detect objects and measure distance.
    b. *Example:* parking sensors in cars

- **Temperature Sensors:**
    a. Temperature sensors measure heat and convert it into an **electrical signal**.
    b. *Example:* Smart thermostats.
- **Touch Sensor:**
    a. Touch sensors detect **physical contact,** such as a finger or stylus touch.
    b. *Example:* Smartphones, ATMs.
- **Range Sensors:**
    a. Range sensors measure the **distance between the sensor and an object**.
    b. **Short-range sensing:** Uses capacitance or inductance
    c. **Long-range sensing:** Uses **radio waves**
    d. Example: Parking assistance, obstacle avoidance in drones.
- **Speed Sensors:**
    a. Speed sensors detect the **rate of motion of an object**.
    b. *Example:* Wind speed sensors. Speedometer in vehicles.

Transducers

A transducer is a device that converts one form of energy into another. All sensors are transducers, but not all transducers are sensors.

- **Types of Transducers:**
    - **Input Transducers (Sensors):**
        - Convert physical quantity → electrical signal.
        - Examples: Microphone, ultrasonic.
    - **Output Transducers (Actuators):**
        - Convert electrical signal → physical action.
        - Examples: Motor, robot, printers.
    - **Active Transducers (Self-generating):**
        - Generate their own electrical signal without external power.
        - Examples: Thermocouple.
    - **Passive Transducers (Require external power):**
        - The output signal depends on the external excitation.
        - Examples: Resistance Temperature Detector Sensor, LDR

Actuators

An actuator converts an electrical signal into mechanical movement, heat, sound, or other physical action. It performs the **actual action** in IoE systems.

Working Principle:

1. The controller sends an electrical control signal.
2. The actuator converts the signal into a physical action.
3. This action changes the environment (fan ON, valve opens, etc.).

Characteristics:

- **Precision:** How accurately it performs the action.
- **Speed:** How fast it responds.
- **Control Method:** On/Off, analog, or digital control.

Types of Actuators:

- **Electrical Actuators:**
    - An **electric actuator** converts **electrical energy into mechanical motion**.
    - These include **servo motors and solenoids**.
    - *Example:* Robots, printers.

- **Hydraulic Actuators:**
    - A **hydraulic actuator** uses **pressurised hydraulic fluid** to perform mechanical operations. The actuator usually consists of a **cylinder or a hydraulic motor**. Depending on design, the actuator can produce **linear, rotary, or oscillatory motion**.
    - *Example:* aircraft brakes.

- **Pneumatic Actuators:**
    - A **pneumatic actuator** uses **compressed air or vacuum** to create **linear or rotary motion** through pistons or cylinders.
    - *Example:* Factory automation machines.

- **Thermal Actuators:**
    - These actuators are powered by **thermal expansion** or **magnetic properties** of materials.
    - *Example:* Bimetallic strips in thermostats.

- **Mechanical Actuators:**
    - Mechanical actuators use **gears** or **pulleys** to convert rotary motion into linear displacement.
    - *Example:* Robotic Joints.

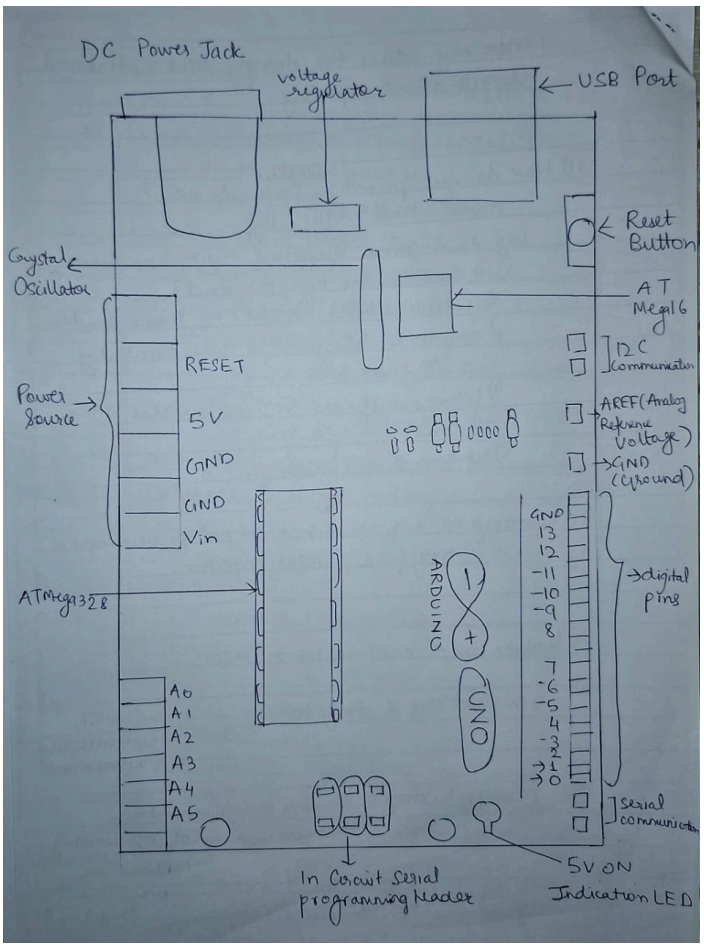
Example Flow in an IoE System

Smart Light Control

- **Sensor:** LDR (light sensor) detects darkness in the room.
- **Transducer:** Converts light level into an electrical signal.
- **Processor:** Microcontroller checks if the room is too dark.
- **Actuator:** Turns ON the light automatically.

Arduino Uno 3

Pin Diagram and Architecture



Device Features of Arduino / Pin Diagram

Simple Hardware Interface: Easily connects with sensors **(temperature sensors, light sensors, etc.), actuators (motors, LEDs, relays), and display modules (LCDs, 7-segment displays).** Standard pin layout for easy wiring or use of shields.

1. **DC Power Jack:** Connects external power supply (7–12V).
2. **Voltage Regulator:** Regulates the input voltage to a stable 5V or 3.3V.
3. **USB Port:**
    a. Built-in USB port for connecting to a PC and uploading compiled code (called “sketch”) and power supply.
    b. Acts as both **communication link** and **power source**.
    c. **Example:** Connect Arduino Uno to PC → Open Arduino IDE → Write code → Upload.
4. **Reset Button:** Restarts the program.
5. **Microcontroller (ATmega328 / ATmega16):** It runs the Arduino code. ATmega328 is the main chip on Arduino Uno
6. **I2C Pins (A4–SDA, A5–SCL):** For communication with sensors and other I2C devices.
7. **AREF (Analog Reference Voltage) Pin:** Used to provide a reference voltage for the analog inputs.
8. **GND Pins:** Ground connection for the circuit.

9. **Digital Pins (0–13):** Used as input/output.
    a. Can be used as digital input or output pins.
    b. Some pins (like 0 and 1) are used for serial communication (RX and TX).
    c. Pins 3, 5, 6, 9, 10, and 11 provide PWM (Pulse Width Modulation) output controlling brightness or motor speed
10. **Analog Pins (A0–A5):** Used to read analog signals from sensors, converting voltage into a digital value.
11. **Power Pins:**
    a. **Vin:** External input (**7 - 12V)**
    b. **5V, 3.3V:** Regulated outputs of 5V and 3.3V respectively
    c. Has protection against overvoltage.
12. **Crystal Oscillator:** Provides a clock signal to the microcontroller. (16 MHz)
13. **ICSP (In Circuit Serial Programming) Header:** Used for programming the microcontroller directly.
14. **Power LED:** Glows when the board is powered.
15. **TX/RX LEDs:** Indicate serial communication activity.

Platform Features of Arduino / Advantages (write the same ans for both)

1. Arduino IDE

- A simple easy-to-use software platform for writing, compiling, and uploading code to the Arduino.
- Provides:
    - Text editor to write Arduino “sketches” (source code).
    - Compiler converts sketches to machine code.
    - Upload tool to send code to the Arduino board.

2. Open-Source Hardware & Software

- Both hardware design and IDE are **open-source**.
- **Benefits:** Wide range of shields, libraries & projects available

3. Strong Community Support

- Large global user community.
- Provides tutorials, examples, and libraries.
- **Example:** Servo.h → controls motors, LiquidCrystal.h → controls LCDs

4. Cross-Platform Support: Code works and can be transferred on **Windows, macOS, Linux**.

5. Cost-Effective: Affordable and accessible to students, hobbyists and professionals.

Types of Arduino Boards

Feature	Arduino Nano	Arduino UNO R3	Arduino Pro Mini	Arduino Micro	Arduino Leonardo	Arduino Nano Every	Arduino Mega2560 Rev3
Microcontroller	ATmega328	ATmega328P	ATmega328	ATmega32U4	ATmega32U4	ATmega4809	ATmega2560
Form Factor	Small, breadboard-friendly	Standard-sized development board	Very compact, no USB port	Small, breadboard-friendly	Standard-sized	Small, next-gen version of Nano	Large-sized
USB Port	Mini USB	USB B	No USB (uses external USB-UART)	Micro USB	Micro USB	Micro USB	USB B
Power Supply	5V regulated	5V regulated	3.3V or 5V versions available	5V regulated	5V regulated	5V regulated	5V regulated
PWM Outputs	6 pins (~3, 5, 6, 9, 10, 11)	6 pins (~3, 5, 6, 9, 10, 11)	6 pins (~3, 5, 6, 9, 10, 11)	7 pins (~3, 5, 6, 9, 10, 11, 13)	7 pins (~3, 5, 6, 9, 10, 11, 13)	6 pins	15 pins
Analog Inputs	8 channels (10-bit ADC)	6 channels (10-bit ADC)	6 channels (10-bit ADC)	12 channels (10-bit ADC)	12 channels (10-bit ADC)	8 channels (10-bit ADC)	16 channels (10-bit ADC)
Communication Interfaces	UART, SPI, I2C	UART, SPI, I2C	UART, SPI, I2C	UART, SPI, I2C	UART, SPI, I2C	UART, SPI, I2C	4 x UART, SPI, I2C
Onboard USB-to-UART Bridge	FT232 Chip	ATmega16U2	None (external module required)	Built-in (ATmega32U4)	Built-in (ATmega32U4)	Built-in (ATmega4809)	ATmega16U2
Program Memory (Flash)	32 KB	32 KB	32 KB	32 KB	32 KB	More (depends on ATMega4809 spec)	256 KB
SRAM	2 KB	2 KB	2 KB	2.5 KB	2.5 KB	6 KB	8 KB
EEPROM	1 KB	1 KB	1 KB	1 KB	1 KB	256 Bytes	4 KB
Special Features	Small size, breadboard compatible	Best for beginners, wide support	Ultra compact, low-cost solution	Acts as HID (Keyboard/Mouse)	HID and Virtual COM port support	200% more RAM and 50% more Flash than Nano	Suitable for PLC, CNC, 3D printers
Price	Low	Moderate	Low	Moderate	Moderate	Low	Higher than others

Raspberry Pi

- The **Raspberry Pi** is a **low-cost, credit-card-sized single-board computer (SBC)** developed by the **Raspberry Pi Foundation (UK)** in **2012**.

Features

- **Processor (CPU):** ARM-based, ranging from single-core (Pi 1) to powerful quad-core (Pi 4).
- **Operating System:** Linux-based, commonly **Raspberry Pi OS (formerly Raspbian)**, but can also run Ubuntu, Windows IoT, etc.
- **Connectivity:** Supports **USB, HDMI, Ethernet, Wi-Fi, Bluetooth**.
- **GPIO Pins (General Purpose Input Output):** Allows direct hardware interfacing with sensors, LEDs, motors, and other electronics.
- **Main Storage:** Uses a **microSD card.**
- **Affordable:** Ranges from ultra-cheap **Pi Zero (~$5)** to high-performance **Pi 4 (~$35–$75 depending on RAM)**.

Applications

- **Education:** Teaching programming (Python, C, Java).
- **IoT & Robotics:** Acts as a controller for smart devices, automation, and robots.
- **DIY Projects:** Home automation, weather stations, security systems, etc

On-board Components of Raspberry Pi (Short Notes)

(acronym for first 7 components - "**S**ome **R**eally **P**retty **G**irls **H**ug **U** and **E**veryone")

1. System on Chip (SoC):

- Heart of Raspberry Pi
- Main chip with **CPU + GPU + Memory controller**.
- Example: Pi 4 → BCM2711

2. RAM:

- Built-in memory (not upgradable) for storage.
- Size: **256 MB** → **8 GB** (model-based).

3. Power Input:

- **Micro-USB (old)** or **USB-C (Pi 4)**.
- Input: **5V, 2–3A**.

4. GPIO Pins:

- **26 (Pi 1) or 40 (Pi 2 and newer) pins** to connect sensors, motors, etc.
- Supports **UART, I²C, SPI, PWM**.

5. HDMI / Display Ports:

- Connects to monitor or TV.

- **Pi 4:** 2× Micro-HDMI (dual 4K).

6. USB Ports:

- For connecting keyboard, mouse, pen drive, camera,
- **Pi 4:** 2× USB 2.0 + 2× USB 3.0.

7. Ethernet Port:

- For wired internet connection.
- **Pi 4:** True Gigabit Ethernet.

8. Wireless Module:

- (From Pi 3 onwards)
- Built-in **Wi-Fi + Bluetooth.**

9. CSI (Camera Serial Interface) Port: Connects **Camera Module** for photo/video using small ribbon cable.

10. DSI (Display Serial Interface) Port: Connects **official touchscreen display** using ribbon cable.

11. Audio Jack (3.5 mm): Pi 4 combines **audio + video output**.

12. MicroSD Slot: Works as **primary storage and boot device**. Holds the **Raspberry Pi OS (Linux)** and user files.

13. LED Indicators:

- **PWR (Red):** Power on
- **ACT (Green):** SD card activity

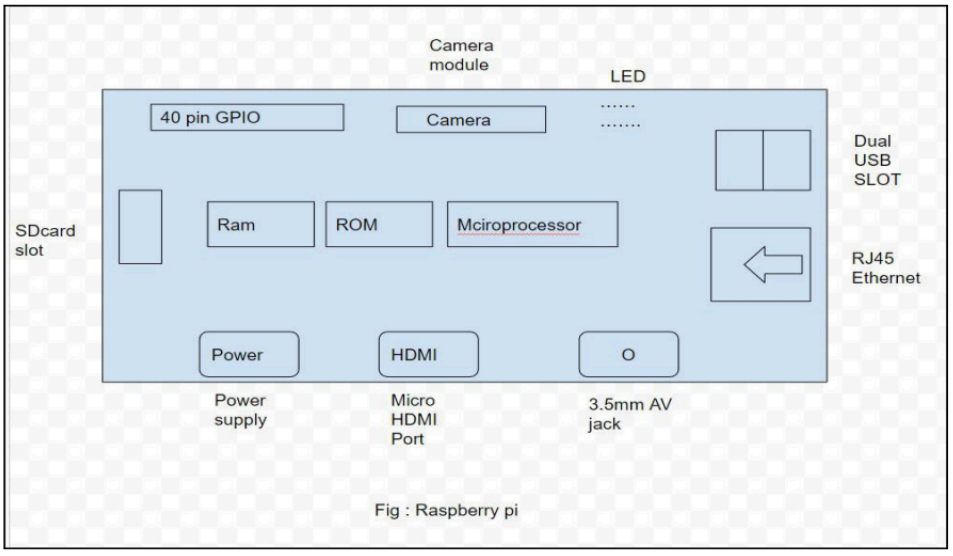
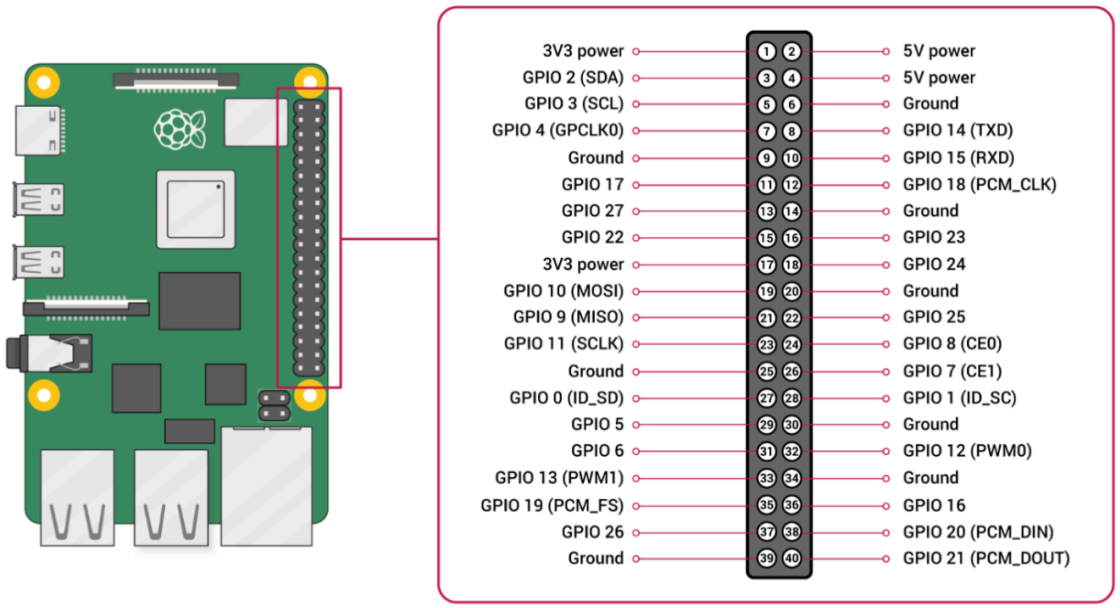


Fig : Raspberry pi

Pin Diagram and Architecture (RATTOFY THE DIAGRAM.)



(Steps for learning the diagram : First learn the Power Pins and Ground Pins from below then ratofy and other pins.)

Raspberry Pi 40-Pin GPIO – Pin Explanation

Power Pins (1, 2, 4, 17)

- **Pin 1 (3V3)**: Provides **3.3V power**. Safe voltage for sensors, modules, and ICs.
- **Pin 2 (5V)**: Provides **5V power** (direct from USB power supply).
- **Pin 4 (5V)**: Another **5V supply**.
- **Pin 17 (3V3)**: Additional **3.3V supply**.

Ground Pins (GND)

- **Pins 6, 9, 14, 20, 25, 30, 34, 39**: All are **Ground (0V)** connections. Used as reference voltage for circuits.

General Purpose Input/Output (GPIO) Pins

I²C Communication (used to connect sensors like temperature, accelerometers, etc.)

- **GPIO2 (Pin 3)** → SDA (data line)
- **GPIO3 (Pin 5)** → SCL (clock line)

Basic General Purpose Pins (used a lot in projects for LEDs, buttons, relays)

(rato method) Normal Pin - Pin 7,11,13,15 and 29,31 and 36,37

(rato method) GPIO Pin - Pin 4,17,27,22 and 5,6 and 16,26

- **GPIO4 (Pin 7)**
- **GPIO17 (Pin 11)**
- **GPIO27 (Pin 13)**
- **GPIO22 (Pin 15)**
- **GPIO5 (Pin 29)**
- **GPIO6 (Pin 31)**
- **GPIO16 (Pin 36)**
- **GPIO26 (Pin 37)**
- These can be programmed as **input (read sensors)** or **output (switch on LEDs/motors)**.

🎬 **SPI Communication (**It is a **communication protocol** used to connect microcontrollers (like Raspberry Pi) with external devices such as sensors, displays, SD cards, and other chips).

(rato method) Normal Pin - Pin 19, 21, 23, 24, 26 and 35, 38, 40

(rato method) GPIO Pin - Pin 10, 9, 11, 8, 7 and 19, 20, 21)

- **GPIO10 (Pin 19)** → MOSI (sends data from Pi to device)
- **GPIO9 (Pin 21)** → MISO (receives data from device to Pi)
- **GPIO11 (Pin 23)** → SCLK (clock signal)
- **GPIO8 (Pin 24)** → CE0 (Chip Select 0 – chooses device 1)
- **GPIO7 (Pin 26)** → CE1 (Chip Select 1 – chooses device 2)
- **GPIO19 (Pin 35), GPIO20 (Pin 38), GPIO21 (Pin 40)** → Can also act as SPI in some cases

🎬 **Special Identification Pins (for Raspberry Pi HATs)**

- **GPIO0 (Pin 27)** → ID_SD (used internally)
- **GPIO1 (Pin 28)** → ID_SC (used internally)

🎬 **PWM (Pulse Width Modulation – used for dimming LEDs, controlling motor speed)**

- **GPIO12 (Pin 32)**
- **GPIO13 (Pin 33)**
- **GPIO19 (Pin 35)**

UART (Serial Communication)

- **GPIO14 (Pin 8)** – TXD (Transmit Data).
- **GPIO15 (Pin 10)** – RXD (Receive Data).

Advantages

- **Low cost + High functionality**
- **Compact and portable** (fits in your palm)
- **Strong community support** (millions of users worldwide)
- **Flexible use cases** – from learning programming to building AI/ML and IoT projects

Comparison of Various Raspberry Pi Models

Model	Release Year	Processor (CPU)	RAM	USB Ports	Networking	Wireless	GPIO Pins	Special Features / Notes
Raspberry Pi 1 (Model B)	2012	700 MHz ARM11 (single-core)	256 MB (later 512 MB)	2 × USB 2.0	10/100 Ethernet	No	26	First Pi, very basic, HDMI + RCA output
Raspberry Pi 2 (Model B)	2015	900 MHz Quad-core ARM Cortex-A7	1 GB	4 × USB 2.0	10/100 Ethernet	No	40	Big upgrade in CPU, introduced 40-pin GPIO
Raspberry Pi 3 (Model B)	2016	1.2 GHz Quad-core ARM Cortex-A53 (64-bit)	1 GB	4 × USB 2.0	10/100 Ethernet	Wi-Fi 802.11n, Bluetooth 4.1	40	First Pi with built-in wireless
Raspberry Pi 3 (Model B+)	2018	1.4 GHz Quad-core Cortex-A53	1 GB	4 × USB 2.0	Gigabit Ethernet (over USB 2.0, max 300 Mbps)	Wi-Fi 802.11ac, BT 4.2	40	Faster CPU, better network, improved wireless
Raspberry Pi 4 (Model B)	2019	1.5 GHz Quad-core Cortex-A72 (64-bit)	2 GB, 4 GB, 8 GB options	2 × USB 2.0, 2 × USB 3.0	True Gigabit Ethernet	Wi-Fi 802.11ac, BT 5.0	40	Dual micro-HDMI ports (supports dual 4K displays), USB-C power
Raspberry Pi Zero	2015	1 GHz ARM11 (single-core)	512 MB	1 × micro USB OTG	No	No	40 (unpopulated header)	Ultra-small, cheapest Pi
Raspberry Pi Zero W	2017	1 GHz ARM11	512 MB	1 × micro USB OTG	No	Wi-Fi, Bluetooth 4.1	40	Tiny, with wireless
Raspberry Pi Zero 2 W	2021	1 GHz Quad-core Cortex-A53	512 MB	1 × micro USB OTG	No	Wi-Fi, Bluetooth 4.2	40	Much faster than Zero W, compact size

Arduino vs Raspberry Pi – Simplified Comparison Table (Only Your Selected Points)

S.No.	Arduino	Raspberry Pi
1. Control Unit	Uses **ATmega** microcontroller family	Uses **ARM-based** processor family
2. Based on	**Microcontroller**	**Microprocessor**
3. Hardware & Software Structure	Simple and easy	More complex
4. CPU Architecture	**8-bit**	**64-bit**
5. RAM Usage	Very little RAM (around **2 kB**)	Needs more RAM (around **1 GB**)
6. Processing Speed	About **16 MHz**	About **1.4 GHz**
7. Cost	Cheaper	Expensive
8. Logic Level	**5V**	**3V**
9. Internet Support	No inbuilt Internet	Has **Wi-Fi and Ethernet**
10. Applications (easy ones)	Traffic light system, simple sensors	Small computer projects, camera projects