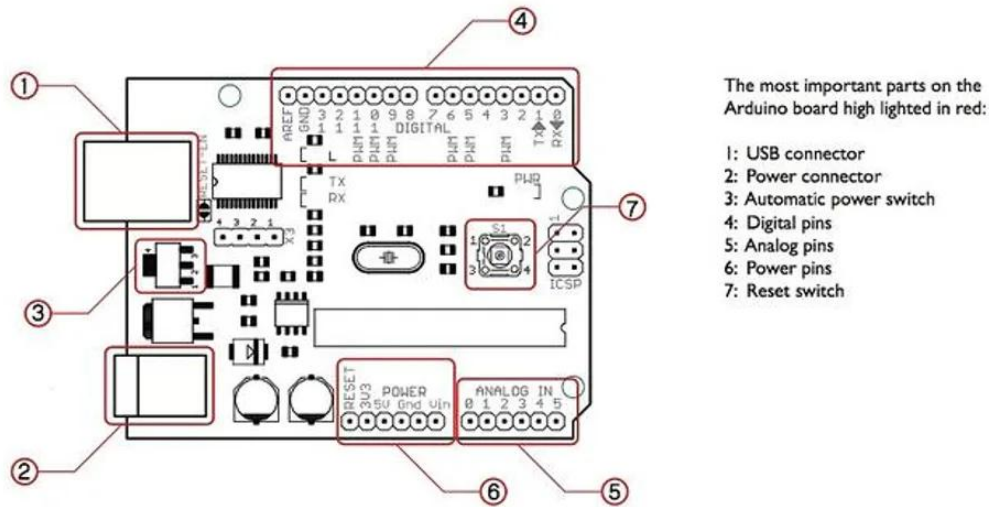# EXPERIMENT 1

AIM – To study about the ARDUINO and RASPBERRY-PI board architecture

THEORY-

ARDUINO

- **Analog Pins** – There are 6 pins connecting analogue input from A0 to A5. It receives input in the form of continuous waves sent by any external source.
- **Digital Pins** – Signifies serial input/output connections.
  - **D0, D1 -** It connects Arduino with other serial devices like computers or sensors.
  - **D2 to D13** – These are used for data transfer digital Input/Output.
- **USB Jack** – It is used to connect Arduino board to PC.
- **Microcontroller** – It is designed for specific tasks. It is the heart and brain of the Arduino board. To program the microcontroller, we have to provide coding from your computers.
- **Test LEDs** – Connected at D13 pin to check the Arduino board.
- **Receivers and Transmitters** – Rx blinks when the Arduino board receives information whereas the Tx blinks when the Arduino board sends information.
- **Power Jack** – Used for power supply.
- **ICSP (In-Circuit Serial Programming)** – Used to modify the Arduino firmware.

The latest Microcontroller used in the Arduino board is ATMEGA 328P.
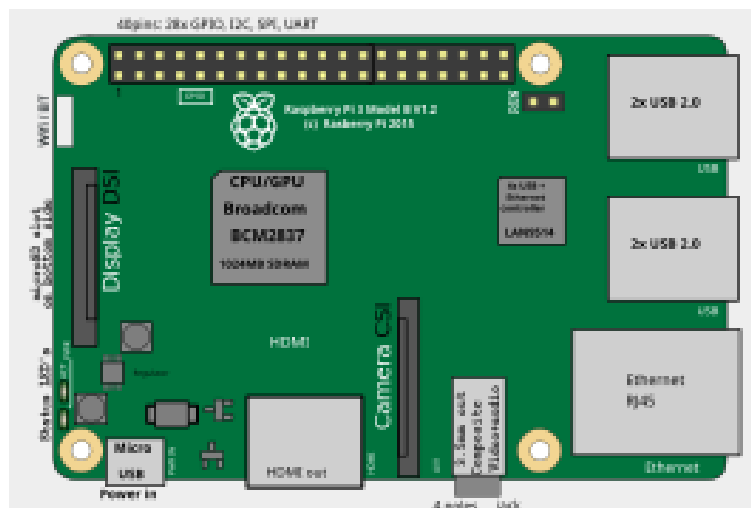
RASPBERRY PI BOARD ARCHITECTURE

- **HDMI Port**: Transmits high-definition video and audio signals to external displays, such as monitors or TVs. It supports both HDMI 1.3 and 1.4 standards for various resolutions and refresh rates.
- **USB Ports**: Allow connection to peripherals like keyboards, mice, or external storage devices. These can be USB 2.0 or 3.0, depending on the model, for data transfer.
- **RJ45 Ethernet Port**: Provides wired internet connectivity. It connects to a local network, enabling fast and reliable internet access via a network cable.
- **GPIO Pins**: General-purpose input/output pins used for controlling external devices (e.g., sensors, LEDs). They can be configured as input or output for various electronic projects.
- **Audio Jack**: A 3.5mm stereo output that allows you to connect headphones or speakers for audio playback from the Raspberry Pi.
- **CSI Camera Connector**: A ribbon cable interface used to connect a camera module for capturing images and videos, commonly used for projects requiring vision.
- **DSI Display Connector**: Used for connecting official Raspberry Pi displays. It supports high-speed data transfer for video output to LCD screens.
- **Power Input (Micro-USB / USB-C)**: Supplies power to the Raspberry Pi. Depending on the model, it can be a Micro-USB (older models) or USB-C (newer models) port.
- **SD Card Slot**: Holds the microSD card, which acts as the primary storage for the operating system, software, and files. It is essential for booting the Raspberry Pi.
- **Broadcom SoC (System on Chip)**: The main processing unit of the Raspberry Pi, integrating the CPU, GPU, RAM, and various peripherals into a single chip for efficient computing.
- **LED Indicators**: Small lights that provide status information, such as power, network activity, and SD card status, to help troubleshoot and monitor the device's operation.
- **Wi-Fi / Bluetooth**: Integrated wireless modules (in newer models) for connecting to local Wi-Fi networks or Bluetooth devices, enabling wireless communication without needing additional hardware.
- **Camera Interface (CSI)**: Allows for the connection of camera modules to capture images and video, supporting projects in surveillance, robotics, and computer vision.

EXPERIMENT 2

AIM – To interface LED/Buzzer with Arduino/Raspberry Pi and write a program to turn ON LED for 1 sec after every 2 seconds.
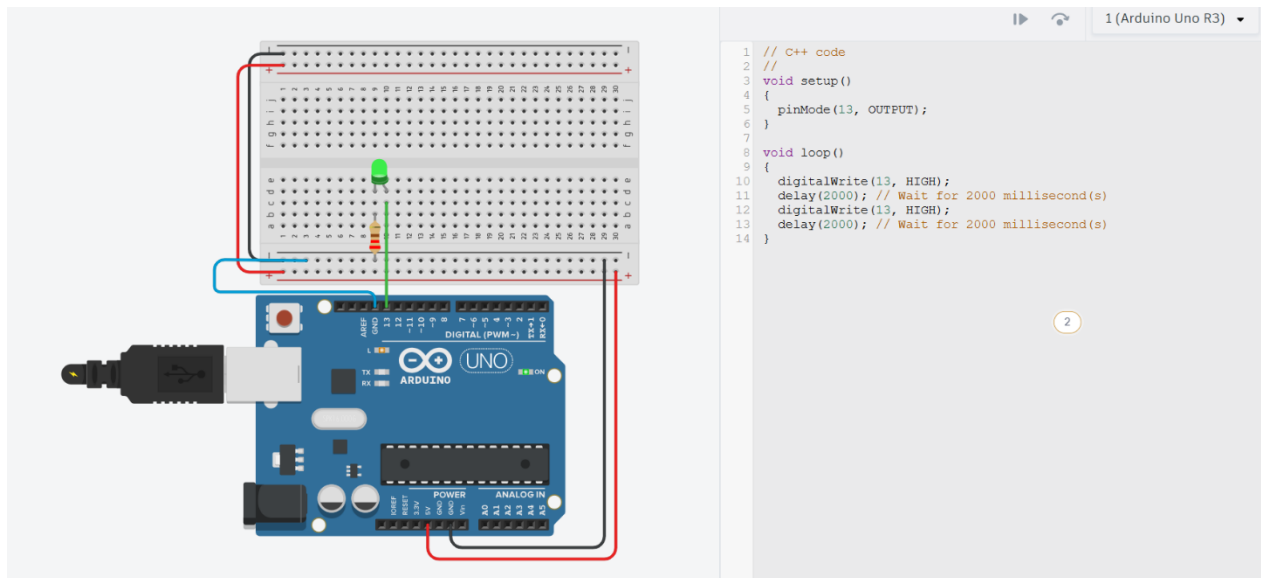
THEORY – The Arduino platform is widely used in electronics and programming due to its simplicity and versatility. The circuit setup involves connecting the LED to designated digital output pins on the Arduino board. This experiment is also a precursor to understanding event-driven programming and input/output (I/O) management in microcontroller applications.

Applications of LED/Buzzer Control Programs
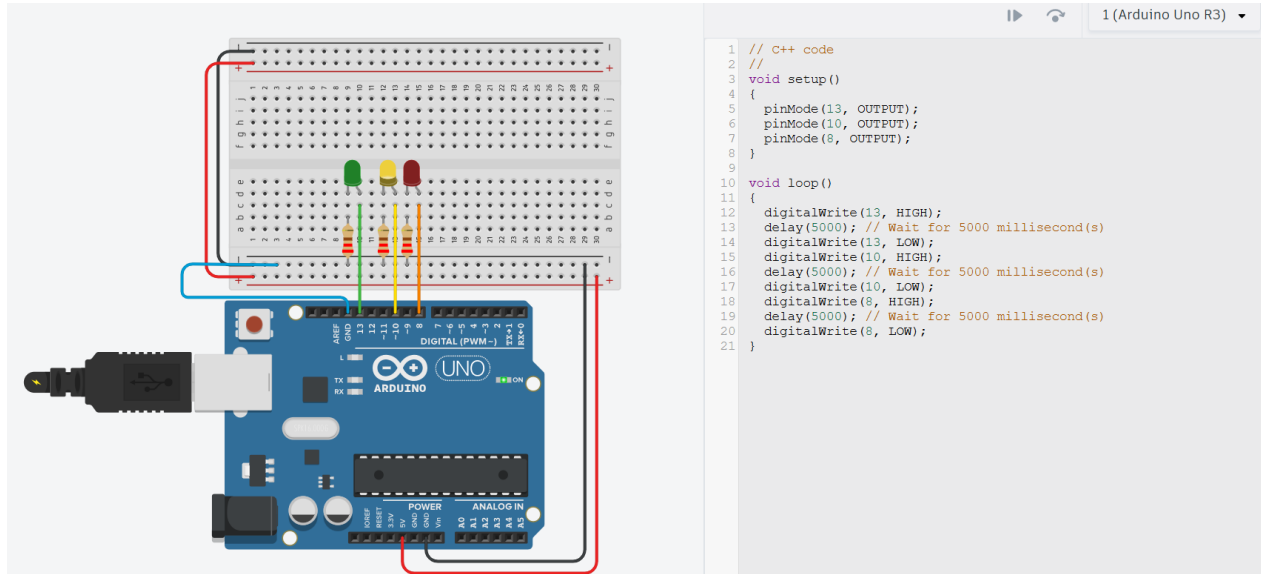
Programs that control LEDs and buzzers with Arduino have a wide range of practical applications across various fields:

1. Indicators and Alerts: LED indicators can signal the status of devices, while buzzers provide audible alerts for alarms, notifications, or warnings in system monitoring.

2. Robotics: In robotic systems, LEDs and buzzers can provide visual and auditory feedback, enhancing user interaction and communication between the robot and its environment.

3. Home Automation: In smart home setups, LED status lights can indicate device states (e.g., on/off) while buzzers can signal alarms or reminders for various tasks.

TO SIMULATE AN LED TO BLINK ON AND OFF AFTER EVERY 2 SECONDS


```cpp
// C++ code
//
void setup()
{
  pinMode(13, OUTPUT);
}

void loop()
{
  digitalWrite(13, HIGH);
  delay(2000); // Wait for 2000 millisecond(s)
  digitalWrite(13, HIGH);
  delay(2000); // Wait for 2000 millisecond(s)
}
```

TO SIMULATE A TRAFFIC SIGNAL THAT TURNS ON AND OFF AFTER EVERY 2 SECONDS


```cpp
// C++ code
//
void setup()
{
  pinMode(13, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(8, OUTPUT);
}

void loop()
{
  digitalWrite(13, HIGH);
  delay(5000); // Wait for 5000 millisecond(s)
  digitalWrite(13, LOW);
  digitalWrite(10, HIGH);
  delay(5000); // Wait for 5000 millisecond(s)
  digitalWrite(10, LOW);
  digitalWrite(8, HIGH);
  delay(5000); // Wait for 5000 millisecond(s)
  digitalWrite(8, LOW);
}
```

Functions Used in Code –

1. pinMode(pin, mode): Configures the specified pin to behave either as an input or an output. In this experiment, it sets the LED and buzzer pins as outputs.

2. digitalWrite(pin, value): Writes a HIGH or LOW value to a pin. Used to turn the LED and buzzer on (HIGH) or off (LOW).

3. delay(ms): Pauses the program for the duration specified in milliseconds. This function controls how long the LED stays on and when the loop iterates.

4. loop(): The main function that runs continuously after the setup() function. It is used for repeated actions like turning the LED on and off at specified intervals.