**CSUA32183: Internet of Things**

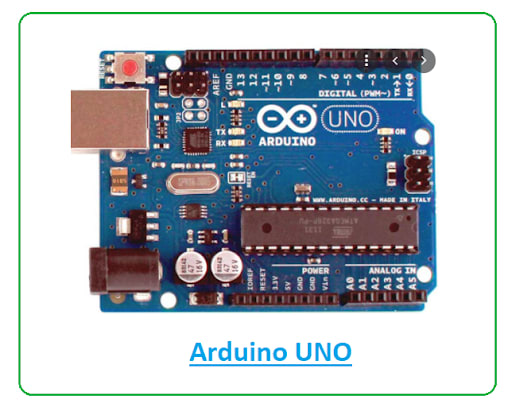
**Aim:** Study Write up on Arduino and Raspberry pi.

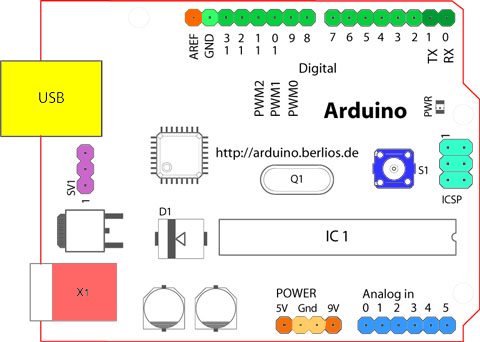
**Theory**:

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online.The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Teachers and students use it to build low cost scientific instruments, to prove chemistry and physics principles, or to get started with programming and robotics.

Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems:

* **Inexpensive** - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than $50
* **Cross-platform** - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
* **Simple, clear programming environment** - The **Arduino Software (IDE)** is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.
* **Open source and extensible software** - The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based.
* **Open source and extensible hardware** - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the breadboard version of the module in order to understand how it works and save money.





Arduino UNO features AVR microcontroller Atmega328, 6 analogue input pins, and 14 digital I/O pins out of which 6 are used as PWM output.

Starting clockwise from the top center:

* Analog Reference pin (orange)
* Digital Ground (light green)
* Digital Pins 2-13 (green)
* Digital Pins 0-1/Serial In/Out - TX/RX (dark green) - These pins cannot be used for digital i/o (digitalRead and digitalWrite) if you are also using serial communication (e.g. Serial.begin).
* Reset Button - S1 (dark blue)
* In-circuit Serial Programmer (blue-green)
* Analog In Pins 0-5 (light blue)
* Power and Ground Pins (power: orange, grounds: light orange)
* External Power Supply In (9-12VDC) - X1 (pink)
* Toggles External Power and USB Power (place jumper on two pins closest to desired supply) - SV1 (purple)
* USB (used for uploading sketches to the board and for serial communication between the board and the computer; can be used to power the board) (yellow)
* This board contains a USB interface i.e. USB cable is used to connect the board with the computer and Arduino IDE (Integrated Development Environment) software is used to program the board.
* The unit comes with **32KB flash memory** that is used to store the number of instructions while the **SRAM is 2KB** and **EEPROM is 1KB**.
* The operating voltage of the unit is **5V** which projects the microcontroller on the board and its associated circuitry operates at 5V while the input voltage ranges between 6V to 20V and the recommended input voltage ranges from 7V to 12V.

### **Arduino UNO Components**

The Arduino UNO board contains the following components and specifications:

* **ATmega328:** This is the brain of the board in which the program is stored.
* **Ground Pin:** there are several ground pins incorporated on the board.
* **PWM:** the board contains 6 PWM pins. PWM stands for Pulse Width Modulation, using this process we can control the speed of the servo motor, DC motor, and brightness of the LED.
* **Digital I/O Pins:** there are 14 digital (0-13) I/O pins available on the board that can be connected with external electronic components.
* **Analogue Pins:** there are 6 analogue pins integrated on the board. These pins can read the analogue sensor and can convert it into a digital signal.
* **AREF:**It is an Analog Reference Pin used to set an external reference voltage.
* **Reset Button:** This button will reset the code loaded into the board. This button is useful when the board hangs up, pressing this button will take the entire board into an initial state.
* **USB Interface:** This interface is used to connect the board with the computer and to upload the Arduino sketches (Arduino Program is called a Sketch)
* **DC Power Jack:** This is used to power up the board with a power supply.
* **Power LED:** This is a power LED that lights up when the board is connected with the power source.
* **Micro SD Card:** The UNO board supports a micro SD card that allows the board to store more information.
* **3.3V:** This pin is used to supply 3.3V power to your projects.
* **5V:** This pin is used to supply 5V power to your projects.
* **VIN:**It is the input voltage applied to the UNO board.
* **Voltage Regulator:** The voltage regulator controls the voltage that goes into the board.
* **SPI:** The SPI stands for Serial Peripheral Interface. Four Pins 10(SS), 11(MOSI), 12(MISO), 13(SCK) are used for this communication.
* **TX/RX:**Pins TX and RX are used for serial communication. The TX is a transmit pin used to transmit the serial data while RX is a receive pin used to receive serial data.

### **Arduino UNO Applications**

The Arduino boards can work as a stand-alone project and can be interfaced with other Arduino boards or Raspberry Pi boards. Arduino UNO board is used in the following applications.

* Weighing Machines
* Traffic Light Count Down Timer
* Parking Lot Counter
* Embedded systems
* Home Automation
* Industrial Automation
* Medical Instrument
* Emergency Light for Railways

**Raspberry Pi**

The Raspberry Pi was developed by Eben Upton at the University of Cambridge in the United Kingdom with the aim of teaching and improving programming skills of students in developing countries. While Arduino is a Microcontroller based development board, the Raspberry Pi is a Microprocessor (usually an ARM Cortex A Series) based board that acts as a computer.

You can connect several peripherals like a Monitor (through HDMI or AV Port), Mouse and Keyboard (through USB), connect to internet (through Ethernet or Wi-Fi), add a Camera (through the dedicated Camera Interface), just like we do to our desktop computer.

As Raspberry Pi is essentially a full computer, it can run an Operating System. The Raspberry Pi Foundation provides a Debian based Linux Distribution called the **Raspberry Pi OS** .

Another important thing about Raspberry Pi is, as it is a Linux based Computer, you can develop software using several Programming Languages like C, C++, Python, Java, etc.

Despite its original intentions, which is to promote programming in schools, the original Raspberry Pi SBC became extremely popular among DIY builders, hobbyists and enthusiasts for developing several applications like Robotics, Weather Stations, Camera based security systems etc.

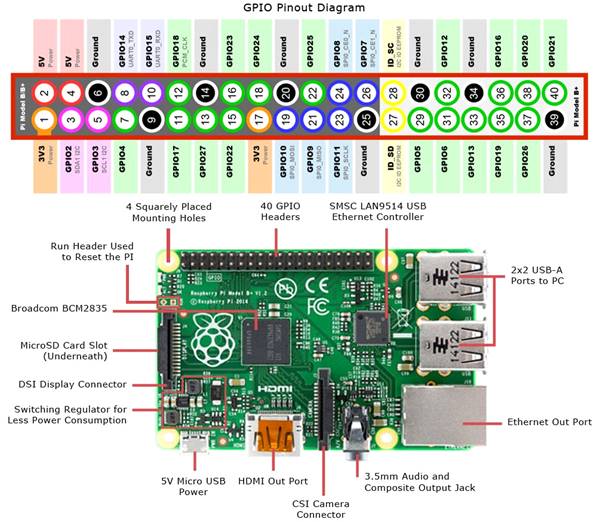
## Raspberry Pi Technology

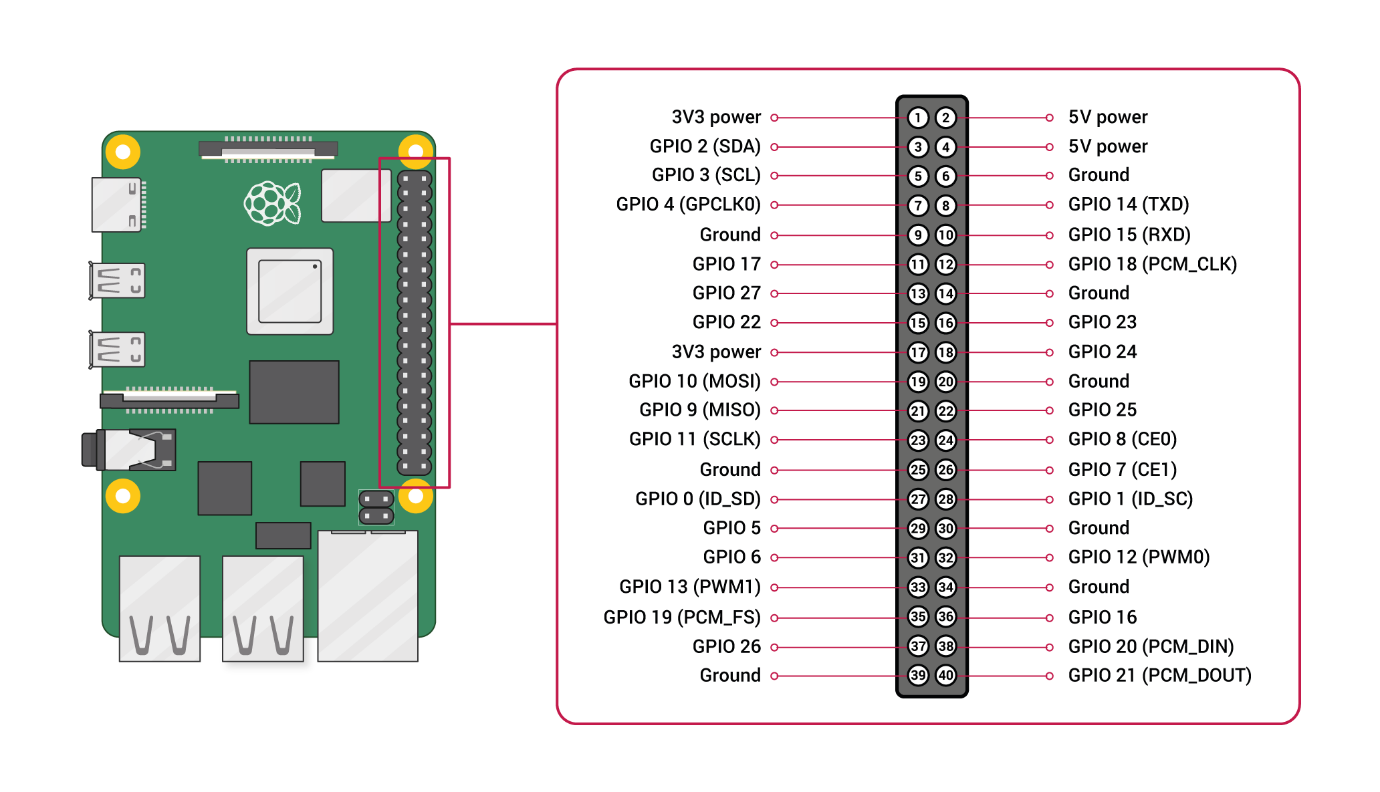
The raspberry pi comes in two models, they are model A and model B. The main difference between model A and model B is USB port. Model A board will consume less power and that does not include an Ethernet port. But, the model B board includes an Ethernet port and designed in china. The raspberry pi comes with a set of open source technologies, i.e. communication and multimedia web technologies

## Raspberry Pi Hardware Specifications

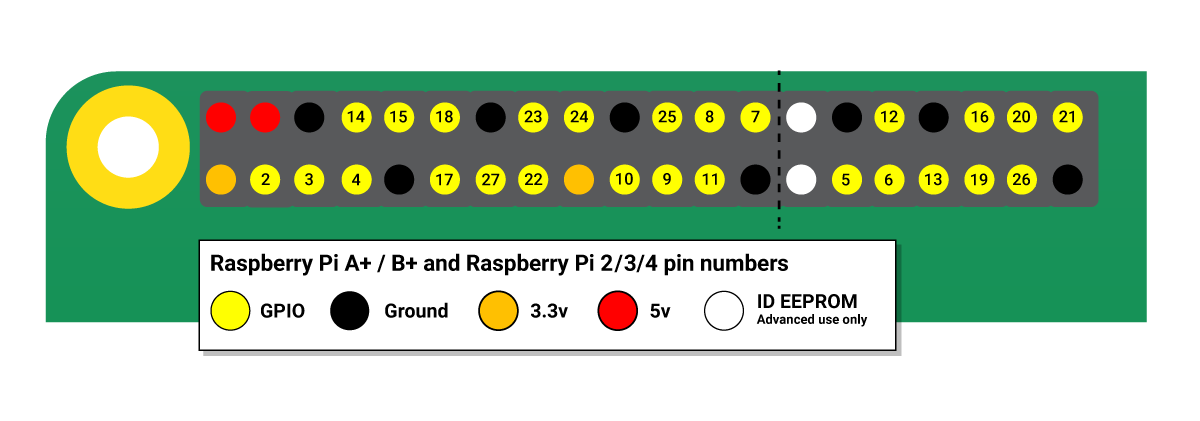
The raspberry pi board comprises a program memory (RAM), processor and graphics chip, CPU, GPU, Ethernet port, GPIO pins, Xbee socket, UART, power source connector. And various interfaces for other external devices. It also requires mass storage, for that we use an SD flash memory card. So that raspberry pi board  will boot from this SD card similarly as a PC boots up into windows from its hard disk.

Essential hardware specifications of raspberry pi board mainly include SD card containing Linux OS, US keyboard, monitor, power supply and video cable. Optional hardware specifications  include USB mouse, powered USB hub, case, internet connection, the Model A or B: USB WiFi adaptor is used and internet connection to  Model B is LAN cable.



A powerful feature of the Raspberry Pi is the row of GPIO (general-purpose input/output) pins along the top edge of the board. A 40-pin GPIO header is found on all current Raspberry Pi boards (unpopulated on Pi Zero and Pi Zero W). Prior to the Pi 1 Model B+ (2014), boards comprised a shorter 26-pin header.

Any of the GPIO pins can be designated (in software) as an input or output pin and used for a wide range of purposes.



**Voltages**

Two 5V pins and two 3V3 pins are present on the board, as well as a number of ground pins (0V), which are unconfigurable. The remaining pins are all general purpose 3V3 pins, meaning outputs are set to 3V3 and inputs are 3V3-tolerant.

**Outputs**

A GPIO pin designated as an output pin can be set to high (3V3) or low (0V).

**Inputs**

A GPIO pin designated as an input pin can be read as high (3V3) or low (0V). This is made easier with the use of internal pull-up or pull-down resistors. Pins GPIO2 and GPIO3 have fixed pull-up resistors, but for other pins this can be configured in software.

**More**

As well as simple input and output devices, the GPIO pins can be used with a variety of alternative functions, some are available on all pins, others on specific pins.

1. PWM (pulse-width modulation)

* Software PWM available on all pins
* Hardware PWM available on GPIO12, GPIO13, GPIO18, GPIO19

1. SPI

* SPI0: MOSI (GPIO10); MISO (GPIO9); SCLK (GPIO11); CE0 (GPIO8), CE1 (GPIO7)
* SPI1: MOSI (GPIO20); MISO (GPIO19); SCLK (GPIO21); CE0 (GPIO18); CE1 (GPIO17); CE2 (GPIO16)

1. I2C

* Data: (GPIO2); Clock (GPIO3)
* EEPROM Data: (GPIO0); EEPROM Clock (GPIO1)

1. Serial

* TX (GPIO14); RX (GPIO15)

#### **Applications of Raspberry Pi**

The raspberry pi boards are used in many applications like Media streamer, Arcade machine, Tablet computer, Home automation, Carputer, Internet radio, Controlling robots, Cosmic Computer, Hunting for meteorites, Coffee and also in raspberry pi based projects.

**Comparison of Raspberry Pi vs Arduino**

|  |  |
| --- | --- |
| **Raspberry Pi** | **Arduino**  Arduino |
| Raspberry Pi is a Single Board Computer or SBC | Arduino is a Microcontroller based  development board |
| It is based on Broadcom SoC, an ARM Cortex A Series Microprocessor | It is based on Atmel Microcontrollers.  Arduino UNO uses ATmega328P  Microcontroller |
| **A Debian based Linux Distribution called Raspberry Pi OS is needed to boot the Raspberry Pi** | **As it is a Microcontroller, there is no**  **need for an operating system** |
| It uses very less RAM, 2 kB. | While Raspberry Pi requires more RAM, 1 GB. |
| Raspberry Pi SBC can perform multiple tasks simultaneously due to its powerful processor and Linux based OS | Arduino is usually used for running  a single task (or a very small no.  of simple tasks) repeatedly, over  and over again |
| All the necessary components like Processor, RAM, Storage, Connectors, GPIO Pins, etc. are situated on the Raspberry Pi Board itself | The Microcontroller on the Arduino  Board (like ATmega328P) contains  The Processor, RAM, ROM. The  board contains supporting hardware  (for power and data) and GPIO Pins |
| Both the hardware and firmware of Raspberry Pi are closed-source i.e., it is not available for general use | Arduino is developed as open-source  hardware and software from the  beginning. |
| Raspberry Pi SBC has several GPIO Pins (the famous 40-pin Raspberry Pi GPIO), using which you can connect different sensors, IO Devices, etc. | GPIO is an important peripheral of  any Microcontroller and Arduino  UNO is no exception. In Arduino  terminology, these pins are called  Digital IO (to connect LEDs and  Buttons) and Analog IN (to connect  analog devices) |
| Using the 40-pin GPIO Pins, you can add additional features / functionalities to Raspberry Pi with HAT (Hardware Attached on Top) expansion boards | A similar way to add extra features and  functionalities in Arduino is using  Arduino Shields (which are also  connected through the IO Pins) |
| As Raspberry Pi is essentially a computer, you have to properly shutdown after using it or before powering it down | As Arduino is a Microcontroller board, you can  plug and unplug the power as you want |
| **The main programming languages for developing application in Raspberry Pi are Python, Scratch, Ruby, C, C++** | **Arduino can be programmed using C or C++**  **Programming Languages** |
| The logic level of Raspberry Pi’s GPIO is 3.3V. So, be careful when connecting hardware to the GPIO Pins | Arduino’s logic level is 5V. As most  of the sensors and modules are designed for  Arduino, there won’t be any problem connecting  them to Arduino. But double check every  module and connection just to be on the safe side |
| Raspberry Pi must be powered using an USB Power Adapter as it requires 5V 2A or 5V 3A power | Arduino can be powered from a computer’s USB  Port (make sure the USB Port’s current limit is  not exceeded) |
| **You can easily connect to internet using Wi-Fi or Ethernet** | **For Arduino, you need additional**  **module or shields to connect to internet** |
| Raspberry Pi has the hardware for Bluetooth and Wi-Fi on board | There is no wireless connectivity in  case of Arduino (at least on board) |

**How to decide between Raspberry Pi and Arduino?**

So, to decide between the two, first you should know what you want to do in your project.

* Arduino is good for repetitive tasks such as opening the garage door, switching the lights on and off, reading from temperature sensors, control a motor as the user wants, etc.
* While Pi is good for performing multiple tasks, driving complicated robots, playing videos, connect to internet, interface cameras, etc.
* For example, if you want to develop an application where you want to monitor Humidity and Temperature from DHT11 Sensor and display the results on an LCD, then Arduino can be used to implement this.
* But if you want to monitor the Humidity and Temperature from DHT11 Sensor, send an e-mail with the results, check / compare the reading with a weather report from online and also display the results on an LCD, then Raspberry Pi is the right choice.
* In simple, Arduino is used for beginner’s projects and quick electronics prototyping while Raspberry Pi is used for and some complicated projects can be easily handled by pi.

**Conclusion**:

In this assignment we learnt about Arduino and Raspberry Pi, their diagrams and their comparison with each other as well as their respective application purposes.

**References:**

1. [https://www.watelectronics.com/know-all-about-raspberry-pi-board-technology/#:~:text=Raspberry%20Pi%20Hardware%20Specifications,interfaces%20for%20other%20external%20devices](https://www.watelectronics.com/know-all-about-raspberry-pi-board-technology/" \l ":~:text=Raspberry%20Pi%20Hardware%20Specifications,interfaces%20for%20other%20external%20devices).
2. <https://robu.in/arduino-pin-configuration/>
3. <https://en.wikipedia.org/wiki/Arduino>
4. <https://www.rs-online.com/designspark/what-is-arduino-uno-a-getting-started-guide>
5. <https://www.raspberrypi.com/documentation/computers/os.html>
6. <https://www.arduino.cc/en/reference/board>
7. <https://www.raspberrypi.com/documentation/computers/os.html#gpio-and-the-40-pin-header>
8. https://www.geeksforgeeks.org/difference-between-arduino-and-raspberry-pi/