Exercise #1 – Familiarization with Arduino and Perform Necessary SoftwareInstallation

Aim:

To study Arduino UNO hardware and Arduino IDE software.

Apparatus Required:

Sign Number	Name of the Equipment	Quantity
1	Arduino UNO	1
2	Computer with Arduino IDE	1
3	USB Cable	1

Theory:

Internet of Things (IoT):

The Internet of Things - (IoT) describes physical objects (Or groups of such objects) that are embedded with sensors, processing ability, software, and other technologies, and that connect and exchange data with other devices and systems over the Internet or other communications networks.

The field has evolved due to the convergence of multiple technologies, including ubiquitous computing, commodity sensors, increasingly powerful embedded systems, and machine learning. Traditional fields of embedded systems, wireless sensor networks, control systems, automation (Including home and building automation), independently and collectively enable the Internet of things. In the consumer market, IoT technology is most synonymous with products pertaining to the concept of the "Smart Home", including devices and appliances (Such as lighting fixtures, thermostats, home security systems and cameras, and other home appliances) that support one or more common ecosystems, and can be controlled via devices associated with that ecosystem, such as smartphones and smart speakers. The IoT can also be used in healthcare systems.

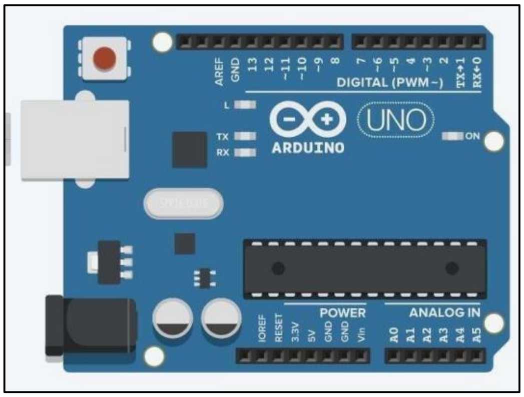
Microcontroller:

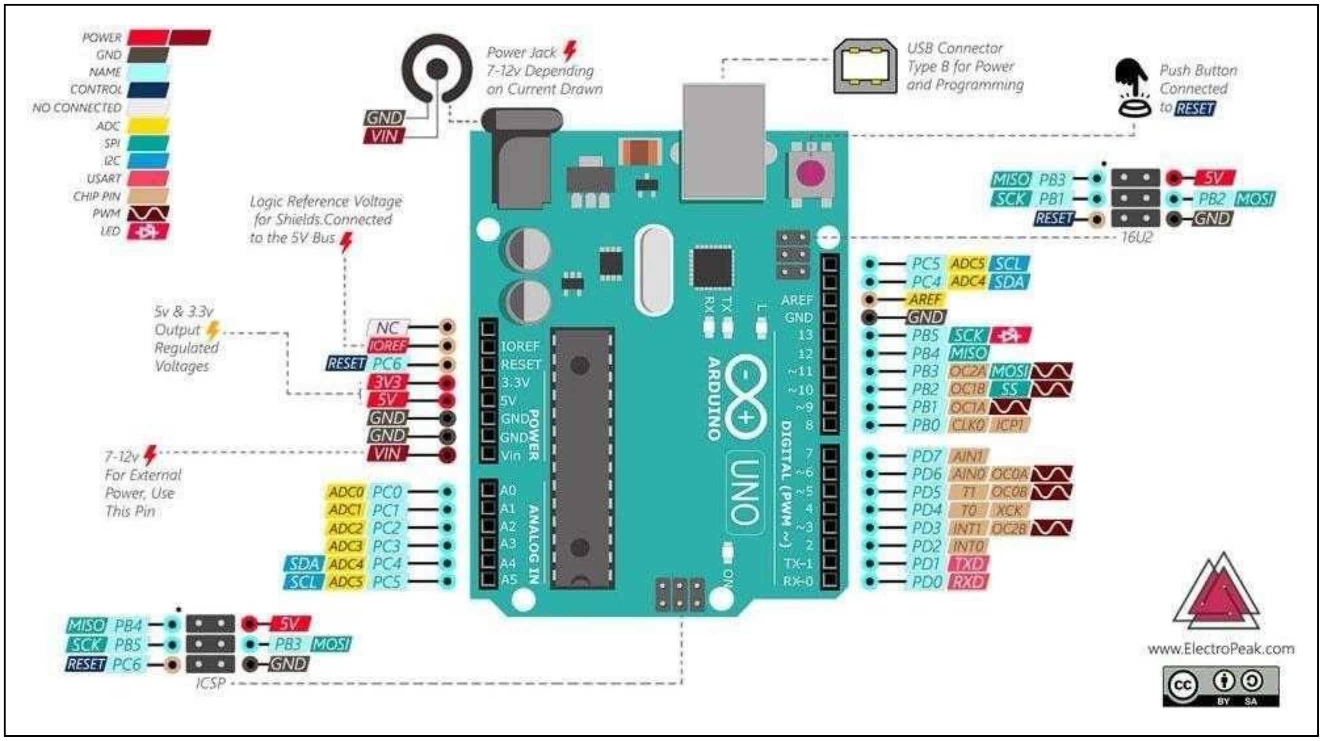
Basically, any product or device that interacts with its user has a microcontroller buried inside. Microcontrollers are "Special Purpose Computers”, that differs from the PC on:

1. Microcontrollers are "Embedded" inside some other device (Often a consumer product) so that they can control the features or actions of the product.
2. Microcontrollers are dedicated to one task and run one specific program. The program is stored in ROM (Read Only Memory) and generally does not change.
3. Microcontrollers are often low-power devices. A desktop computer is almost always plugged into a wall socket and might consume 50 watts of electricity. A battery-operated microcontroller might consume 50 milliwatts.

Arduino UNO:

An Arduino board consists of an Atmel 8-bit AVR microcontroller with complementary components that facilitate programming and incorporation into other circuits. It's an open-source physical computing platform based on a simple microcontroller board, and a development environment for writing software for the board.





General Pin Functions:

LED: There is a built-in LED driven by digital pin 13. When the pin is high value, the LED is ON, when the pin is low, itis OFF.

VIN: The input voltage to the Arduino board when it is using an external power source (As opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

5V: This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 20V), the USB connector (5V), or the VIN pin of the board (7 - 20V). Supplying voltage viathe 5V or 3.3V pins bypasses the regulator, and can damage the board.

3V3: A 3.3V supply generated by the on-board regulator. Maximum current draw is 50 mA.

GND: Ground pins.

Reset: Typically used to add a reset button to shields that block the one on the board.

Special Pin Functions:

Each of the 14 digital pins and 6 analog pins on the Uno can be used as an input or output, under software control (Using *pinMode()*, *digitalWrite()*, and *digitalRead()* functions). They operate at 5 volts. Each pin can provide or receive 20 mA as the recommended operating condition and has an internal pull-up resistor (Disconnected by default) of 20-50K ohm. A maximum of 40mA must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller. The Uno has 6 analog inputs, labeled A0 through A5; each provides 10 bits of resolution (That is, 1024 different values). By default, they measure from ground to 5 volts, though it is possible to change the upper end of the range using the AREF pin and the *analogReference()* function.

In addition, some pins have specialized functions:

Serial / UART: Pins 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connectedto the corresponding pins of the ATmega8U2 USB-to-TTL serial chip.

External interrupts: Pins 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or fallingedge, or a change in value.

PWM (Pulse Width Modulation): Pins 3, 5, 6, 9, 10, and 11. Can provide 8-bit PWM output with the *analogWrite()* function.

SPI (Serial Peripheral Interface): Pins 10 (SS), 11 (MOSI), 12 (MISO), and 13 (SCK). These pins support SPIcommunication using the SPI library.

I²C (Inter IC Communication): Pin SDA (A4) and pin SCL (A5). Support communication using the Wire library.

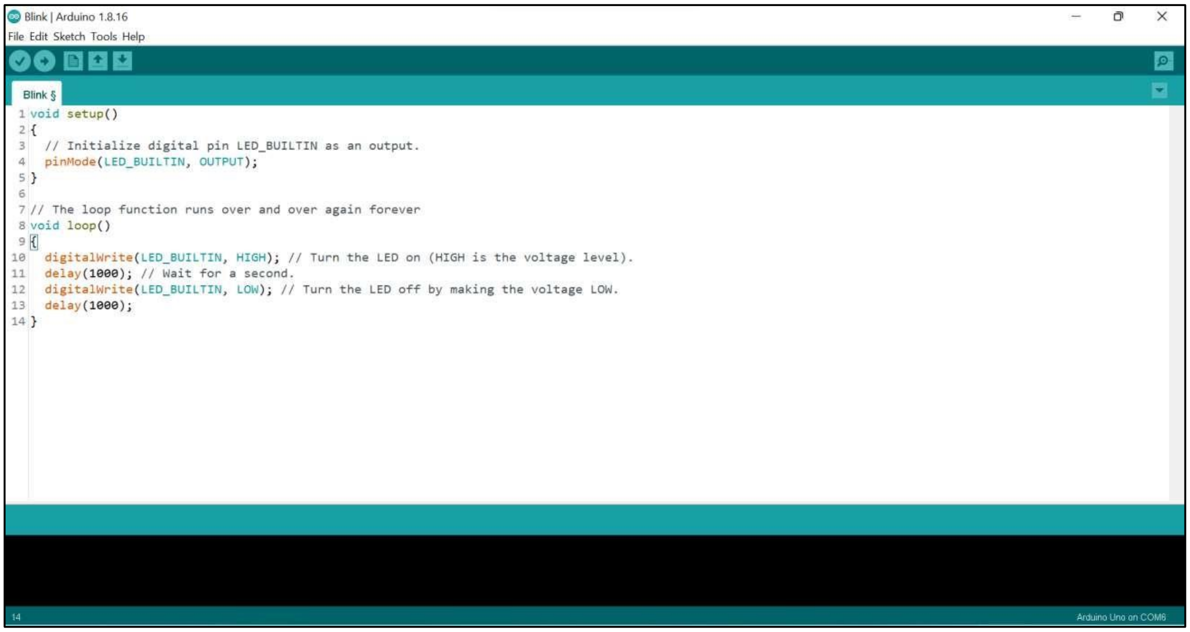
AREF (Analog Reference): Reference voltage for the analog inputs.

Arduino IDE Software:

The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in functions from C and C++.[3] It is used to write and upload programs to Arduinocompatible boards, but also, with the help of third-party cores, other vendor development boards.

Procedure to Install Arduino IDE software:

1. Visit: http://arduino.cc/en/Main/Software
2. Download & install the Arduino environment (IDE) for Windows, Mac, or Linux.
3. Extract the ZIP file. (The extracted folder will contain both the Arduino program itself and also the drivers that allow the Arduino to be connected to your computer by a USB cable).
4. Connect the board to your computer via the UBS cable.
5. The power light on the LED will light up and you may get a “Found New Hardware” message from Windows.
6. Ignore this message and cancel any attempts that Windows makes to try and install drivers automatically for you.
7. Open Device Manager
8. Under the section “Other Devices” you should see an icon for “Unknown Device”, write click on it and press update driver software.
9. Select the option: “Search Automatically for Drivers”.
10. You should be done by successfully installing the Arduino driver.
11. Launch the Arduino IDE.
12. Select your board (Tools -> Board ->UNO)



```
void setup()
{
  // Initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// The loop function runs over and over again forever
void loop()
{
  digitalWrite(LED_BUILTIN, HIGH); // Turn the LED on (HIGH is the voltage level).
  delay(1000); // Wait for a second.
  digitalWrite(LED_BUILTIN, LOW); // Turn the LED off by making the voltage LOW.
  delay(1000);
}
```

Result:

Thus, the Arduino UNO Hardware and Arduino IDE software were studied.