**IoT & Automation Lab.**

**Assignment-1**

1. **What is a Prototype? What are Open source and closed source prototype platforms?**

A prototype is a basic, rough version of a product created to test ideas and show how it works. It helps understand if the design is right and gathers feedback before making the final version.

**Open-source software** is software whose code is open to everyone. Everybody can access, use, modify, or distribute it freely. This fosters community inputs and helps in team working. Some examples of these are Firefox, MySQL, and Arduino.

**Closed-source software** is just software whose code is kept private and owned by a particular company. The public cannot access or modify it, and more often than not, some license or subscription fees are paid for it. Examples include Skype, Microsoft Office, and Adobe Reader.

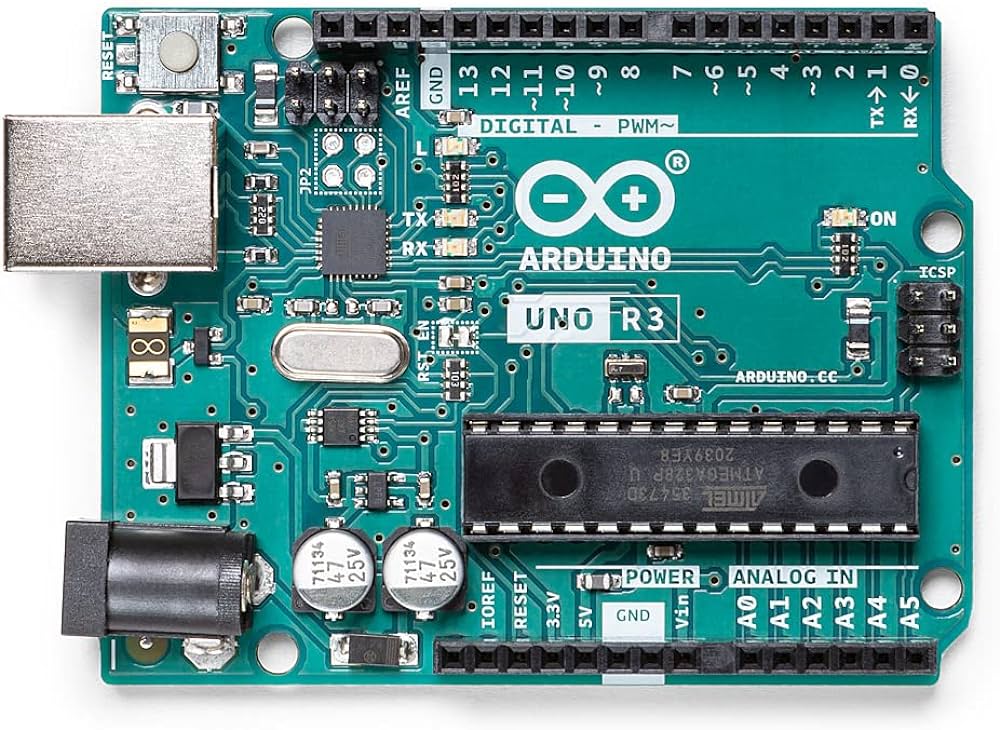
1. **What is Arduino?**

Arduino is an open-source electronic prototyping platform allowing easy creation and development of both hardware and software. This helps in making a project interactive by reading inputs example, press a button and making outputs, such as turning on a light. It has found wide application in the education sector, prototyping new ideas, and DIY projects, such as home automation and wearable technology.

Top of Form

Bottom of Form

1. **Write down Arduino Uno R3 Key Specifications:**



**Main Processor:**

* **Microcontroller:** ATmega328P (8-bit RISC processor)

**Memory:**

* **SRAM:** 2 KB (for temporary data)
* **Flash Memory:** 32 KB (for storing the application code; 0.5 KB used by the bootloader)
* **EEPROM:** 1 KB (non-volatile memory for storing data even after power-off)

**I/O Pins:**

* **Digital I/O Pins:** 14 (6 can provide PWM output)
* **Analog Input Pins:** 6
* **PWM Output Pins:** 6 (Pins 3, 5, 6, 9, 10, 11)
* **UART:** 1 (Serial communication on pins 0 (RX) and 1 (TX))
* **Built-in LED:** Pin 13

**Assignment-2**

**What is an Encoding format? List down encoding formats for various types of data (Text, Number, Photo, Audio, Video).**

Encoding format is a standardized way to convert data into a format that computers can read and process. It’s similar to translating human language into a language that computers can understand.

**Different Encoding Formats:**  
**Text Encoding:**  
ASCII: Stands for 128 characters, to be more precise letters, numbers, and other punctuation marks.  
Unicode: A general type of character set standard that characterize most of the commonly used language as well as the ASCII.  
UTF-8: A kind of variable encoded character code that is maintained between ASCII and the web\_standards.

**Number Encoding:**  
Binary: Functions only with 0s and 1s only.  
Decimal: The system that we us in our daily lives for instance, the base-10 system.  
Hexadecimal: Alphabetic characters represented by erotiske by 16 digits from 0 to 9 and A-F.  
Floating-point: Can be real numbers written in hindu numerals with a decimal point.

**Image Encoding:**  
JPEG: Some examples include lossy compression, which is best used for photos.  
PNG: It is a type of compression which is best used for images having clear cut edges and texts.  
GIF: Supports animation and transparency and is used for the simple images.  
BMP: A format with no compression which hence means that the size of the file is large.  
TIFF: There is no information loss when the images are compressed, it supports different image bit depths.

**Audio Encoding:**  
MP3: Lossy compression the most used format for audio, especially for music.  
AAC: Broadcasting and iTune’s uses lossy compression.  
WAV: Imperfect but will keep all the data as close to original: lossless compression, high-quality sound.  
FLAC: This is free of loss type of compression, it does not reduce on the quality of the audio.

**Video Encoding:**  
MP4: Regular type of a video file storage and sharing.  
AVI: Supports various codecs.  
MOV: Apple’s video format.  
WMV: Microsoft’s video format.

**Assignment-3**

**Explain Basic Structure of an Arduino Program.**

* There are two required parts or functions that enclose blocks of statements.
* **setup()** is the preparation, **loop()** is the execution.
* Both functions are required for the program to work.

void setup()

{

statements;

}

void loop()

{

statements;

}

**The setup() function is called once**, when the Arduino board is first turned on or reset. It is used to initialize the board and set up the hardware.

The setup function should follow the declaration of any variables at the very beginning of the program. It is **the first function to run in the program**, is run only once, and is used to set **pinMode** or initialize serial communication.

Setting pinMode: This tells the Arduino **whether a specific pin is going to be used for input** (reading data) or **output** (sending data).

For example, if you have an LED connected to pin 13, you would specify in the setup() function that pin 13 is an output pin.

**Initializing Serial Communication:** This is like **opening a communication channel** between your Arduino and your computer or another device. This is useful for **sending data back and forth**.

**The loop() function is called repeatedly,** until the Arduino board is turned off or reset. It is where the Arduino program does most of its work.

The loop function follows next and includes the code to be executed continuously – reading inputs, triggering outputs, etc.