**INTERNET OF THINGS ASSIGNMENT RECORD**

**Subject code : BTCS-AMDS-009T**

|  |  |
| --- | --- |
| Name: | Rudra Prasanna Mishra |
| Registration Number: | FET-BAML-2022-2026-032 |
| Course: | Btech CSE AIML |
| Semester: | 5th |
| Faculty: | Mr. Biswajeeban Mishra &  Mr. Pritam Nanda |

|  |  |
| --- | --- |
| Remarks |  |
| Signature |  |



SRI SRI UNIVERSITY

Bidyadharpur, Cuttack, Odisha.

Index

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sl. No.** | **Date** | **Experiment/Case Study** | **Page No.** | **Remark** |
| **1** | 9-8-2024 | Prototyping and Arduino uno r3 |  |  |
| **2** | 9-8-2024 | Encoding formats. |  |  |
| **3** | 9-8-2024 | Basic Structure of an Arduino Program. |  |  |
| **4** |  |  |  |  |
| **5** |  |  |  |  |
| **6** |  |  |  |  |
| **7** |  |  |  |  |
| **8** |  |  |  |  |
| **9** |  |  |  |  |
| **10** |  |  |  |  |
| **11** |  |  |  |  |
| **12** |  |  |  |  |
| **13** |  |  |  |  |
| **14** |  |  |  |  |
| **15** |  |  |  |  |
| **16** |  |  |  |  |

***Assignment No.:1 Date: 9-8-2024***

**1. What is a Prototype?**

A [prototype](https://en.wikipedia.org/wiki/Prototype) is an early sample, model, or release of a product built to test a concept or process. It is a working model used to demonstrate and validate the design, functionality, and performance of a product before it is manufactured at scale. Prototyping is an essential phase in the product development cycle as it allows for the identification and rectification of design flaws and the gathering of user feedback.

**Open Source and Closed Source Prototype Platforms**

**Open Source Prototype Platforms:**

* These platforms offer designs and software that are publicly accessible, allowing users to view, modify, and distribute the code and design files.
* Examples include Arduino, Raspberry Pi, and BeagleBone.
* Opensource platforms encourage community collaboration and innovation.

**Closed Source Prototype Platforms:**

* These platforms have proprietary designs and software that are not freely accessible to the public. The source code and design files are controlled and protected by the company that developed them.
* Examples include platforms like Microsoft's Azure Sphere and some proprietary development boards from companies like Intel and TI.
* Closed source platforms often come with official support and documentation but limit customization and modifications.

**2. What is Arduino?**

[Arduino](https://en.wikipedia.org/wiki/Arduino) is an open-source electronics platform based on easy-to-use hardware and software. It is designed for anyone interested in creating interactive projects. Arduino boards are equipped with sets of digital and analog input/output (I/O) pins that can be interfaced with various expansion boards and other circuits. The platform uses the Arduino programming language, based on Wiring, and the Arduino Software (IDE), based on Processing, to write and upload code to the physical board.

**3. Arduino Uno R3 Key Specifications**

**Main Processor:**

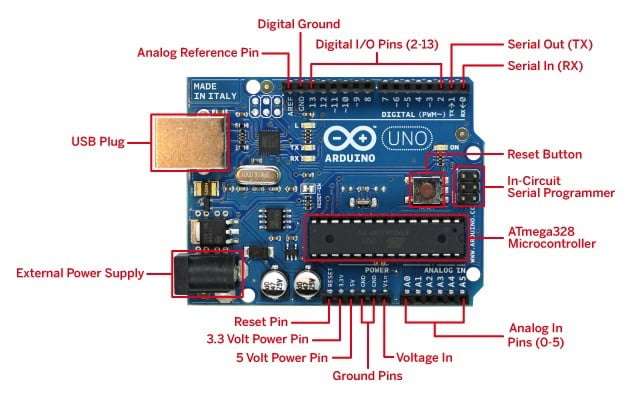
* **Microcontroller:** [ATmega328P](https://en.wikipedia.org/wiki/ATmega328) a  [8-bit](https://en.wikipedia.org/wiki/8-bit) [RISC](https://en.wikipedia.org/wiki/Reduced_instruction_set_computer) processor core.

**Memory:**

* **SRAM:** 2 KB (ATmega328P) [SRAM](https://en.wikipedia.org/wiki/Static_random-access_memory)
* **Flash Memory:** 32 KB (ATmega328P) of which 0.5 KB is used by the bootloader. [Flash memory](https://en.wikipedia.org/wiki/Flash_memory)
* **EEPROM:** 1 KB (ATmega328P) [EEPROM](https://en.wikipedia.org/wiki/EEPROM)

**I/O Pins:**

* **Digital I/O Pins:** 14 (of which 6 provide PWM output)
* **Analog Input Pins:** 6
* **PWM Pins:** 6
* **UART:** 1
* **SPI:** 1
* **I2C:** 1
* **External Interrupts:** 2 (pins 2 and 3)



***Assignment No.:2 Date: 9-8-2024***

**Q1: What is an encoding format?**

**A1:** An encoding format is a standardized method for converting data into a specific format that allows for efficient storage, transmission, and interpretation by computers.

**Q2: What are some common encoding formats for text?**

**A2:** Common text encoding formats include:

* [**ASCII (American Standard Code for Information Interchange)**](https://en.wikipedia.org/wiki/ASCII)
* [**UTF-8 (Unicode Transformation Format - 8-bit)**](https://en.wikipedia.org/wiki/UTF-8)
* [**UTF-16 (Unicode Transformation Format - 16-bit)**](https://en.wikipedia.org/wiki/UTF-16)
* [**UTF-32 (Unicode Transformation Format - 32-bit)**](https://en.wikipedia.org/wiki/UTF-32)
* [**ISO 8859-1 (Latin-1)**](https://en.wikipedia.org/wiki/ISO/IEC_8859-1)

**Q3: What are some encoding formats used for numbers?**

**A3:** Common number encoding formats include**:**

* [**Binary (Base-2)**](https://en.wikipedia.org/wiki/Binary_number)
* [**Decimal (Base-10)**](https://en.wikipedia.org/wiki/Decimal)
* [**Hexadecimal (Base-16)**](https://en.wikipedia.org/wiki/Hexadecimal)
* [**IEEE 754**](https://en.wikipedia.org/wiki/IEEE_754)
* [**BCD (Binary-Coded Decimal)**](https://en.wikipedia.org/wiki/Binary-coded_decimal)

**Q4: What are some encoding formats used for photos or images?**

**A4:** Common photo/image encoding formats include:

* [**JPEG (Joint Photographic Experts Group)**](https://en.wikipedia.org/wiki/JPEG)
* [**PNG (Portable Network Graphics)**](https://en.wikipedia.org/wiki/Portable_Network_Graphics)
* [**GIF (Graphics Interchange Format)**](https://en.wikipedia.org/wiki/GIF)
* [**BMP (Bitmap)**](https://en.wikipedia.org/wiki/BMP_file_format)
* [**TIFF (Tagged Image File Format)**](https://en.wikipedia.org/wiki/TIFF)

**Q5: What are some common audio encoding formats?**

**A5:** Common audio encoding formats include:

* [**MP3 (MPEG-1 Audio Layer III)**](https://en.wikipedia.org/wiki/MP3)
* [**WAV (Waveform Audio File Format)**](https://en.wikipedia.org/wiki/WAV)
* [**AAC (Advanced Audio Coding)**](https://en.wikipedia.org/wiki/Advanced_Audio_Coding)
* [**FLAC (Free Lossless Audio Codec)**](https://en.wikipedia.org/wiki/FLAC)
* [**OGG (Ogg Vorbis)**](https://en.wikipedia.org/wiki/Vorbis)

**Q6: What are some common video encoding formats?**

**A6:** Common video encoding formats include:

* [**MP4 (MPEG-4 Part 14)**](https://en.wikipedia.org/wiki/MPEG-4_Part_14)
* [**AVI (Audio Video Interleave)**](https://en.wikipedia.org/wiki/Audio_Video_Interleave)
* [**MKV (Matroska Video)**](https://en.wikipedia.org/wiki/Matroska)
* [**MOV (QuickTime Movie)**](https://en.wikipedia.org/wiki/QuickTime_File_Format)
* [**WMV (Windows Media Video)**](https://en.wikipedia.org/wiki/Windows_Media_Video)

***Assignment No.:3 Date: 9-8-2024***

**Basic Structure of an Arduino Program**

1. **Include Libraries (Optional):**
   * At the top of the sketch, you can include libraries that extend the functionality of your Arduino code. This is done using the #include directive.
   * Example: #include <DTH11.h> // Include DTH11 library for controlling sensor.
2. **Global Variables and Constants:**
   * You can define global variables and constants that will be used throughout the program. These are declared outside of any functions, usually right after the include statements.
   * Example: int ledPin = 13; // Define a variable for the LED pin
3. **setup() Function:**
   * The setup() function is called once when the Arduino board is powered on or reset. It is used to initialize variables, pin modes, start using libraries, and perform any other setup tasks.
   * Example: void setup() {   
     pinMode(ledPin, OUTPUT); // Set the LED pin as an output  
     }
4. **loop() Function:**
   * The loop() function runs continuously after the setup() function has completed. This is where the main logic of your program goes. The Arduino executes the code inside loop() over and over again until the board is powered off or reset.
   * Example:  
      void loop() {  
     digitalWrite(ledPin, HIGH); // Turn the LED on  
     delay(1000); // Wait for 1 second  
     digitalWrite(ledPin, LOW); // Turn the LED off  
     delay(1000); // Wait for 1 second(1000 ms)  
     }

**Example of a Complete Arduino Program:**

// Include Libraries (if needed)

#include <DTH11.h>

// Global Variables and Constants

int ledPin = 13; // Define the LED pin

// setup() function: Runs once at the start

void setup() {

pinMode(ledPin, OUTPUT); // Set the LED pin as an output

}

// loop() function: Runs repeatedly after setup() finishes

void loop() {

digitalWrite(ledPin, HIGH); // Turn the LED on (HIGH is the voltage level)

delay(1000); // Wait for 1 second (1000 milliseconds)

digitalWrite(ledPin, LOW); // Turn the LED off by making the voltage LOW

delay(1000); // Wait for 1 second

}

**Key Points:**

* setup() Function: Used to initialize settings; runs only once.
* loop() Function: Contains the main logic; runs continuously.
* Global Variables: Declared outside of any function, accessible throughout the sketch.
* Comments: Use // for single-line comments or /\* \*/ for multi-line comments.

This basic structure allows you to build increasingly complex Arduino programs by adding more code to the setup() and loop() functions, as well as including libraries and defining global variables as needed

**References:**

* [Arduino](https://en.wikipedia.org/wiki/Arduino): Overview of the Arduino platform, its history, and its uses.
* [Arduino IDE](https://en.wikipedia.org/wiki/Arduino_IDE): Information on the Integrated Development Environment used to write and upload sketches to the Arduino board.