

Dhirubhai Ambani University

(Formerly known as DA-IICT)

Topic: Introduction to Embedded System

Course: Programming Lab

Course Code- PC503

Dr. Ankit Vijayvargiya

Assistant Professor

Room No. 4205, Faculty Block 4

Email: [ankit_Vijayvargiya\[at\]dau.ac.in](mailto:ankit_Vijayvargiya[at]dau.ac.in)

Phone: 079-68261628(O), 7877709590(M)

Embedded System

What is an Embedded System?

An **embedded system** is a combination of **hardware** and **software** designed to perform a specific function within a larger system.

Main Components

1. Hardware

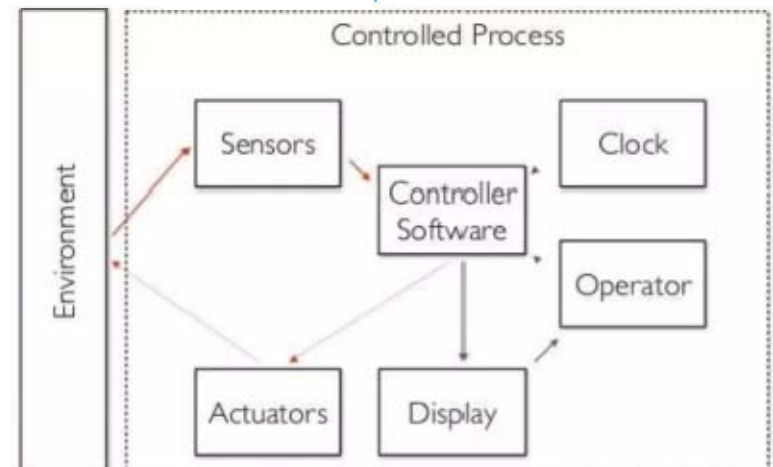
1. **Processor (Microcontroller or Microprocessor):** The brain that runs the control program.
2. **Memory:** For program storage (ROM/Flash) and temporary data (RAM).
3. **Input/Output Devices:** Sensors (inputs) and actuators or displays (outputs).
4. **Power Supply:** To run the system.

2. Software (Firmware)

1. Written in C, C++, or assembly.
2. Stored in ROM/Flash.
3. Executes continuously, usually with real-time constraints.

How it works?

- Sensors collect data from the environment.
- The processor (controller) runs software to make decisions.
- Actuators carry out physical actions based on those decisions.
- The system continuously monitors and reacts to changes, often in **real-time**.



Embedded System

Microprocessor (MPU)

- A microprocessor is the central processing unit (CPU) of a computer system, basically, the *brain* that performs calculations and runs programs.
- However, a microprocessor cannot work by itself. It needs external components like:
 - RAM (for temporary data)
 - ROM / Storage (for programs)
 - Input/Output (I/O) interfaces
 - Timers and communication ports
- So, a microprocessor-based system (like a PC) is made up of *many separate chips* working together.

Even though a microcontroller (like the ATmega328) already has CPU, memory, and I/O built in, sometimes we need **extra or specialized components** depending on the application.

Microcontroller (MCU)

- A microcontroller is a single integrated circuit (IC) that includes:
 - CPU (microprocessor core)
 - Memory (RAM + Flash/ROM)
 - Input/Output ports
 - Timers, ADCs, and peripherals
- It's a complete computer on a single chip, designed to control specific electronic systems.



Embedded System

Arduino

- Arduino is an open-source electronics platform based on easy-to-use hardware and software.
- An Arduino board can read inputs and convert them into outputs.

Why Arduino?

- Inexpensive
- Cross-platform- Windows/Linux/Mac
- Simple, clear programming environment
- Open source and extensible software

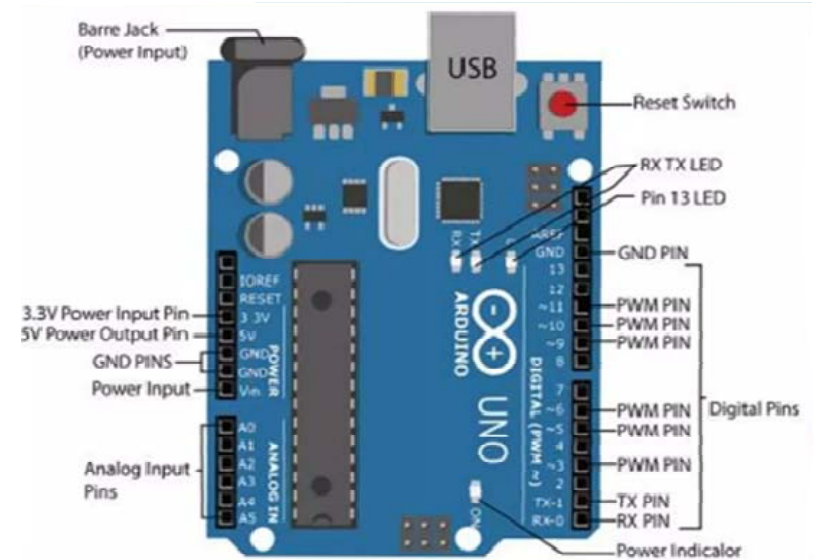


Embedded System

Different types of Arduino board



Arduino Uno Pins



Analog Signal



Digital Signal

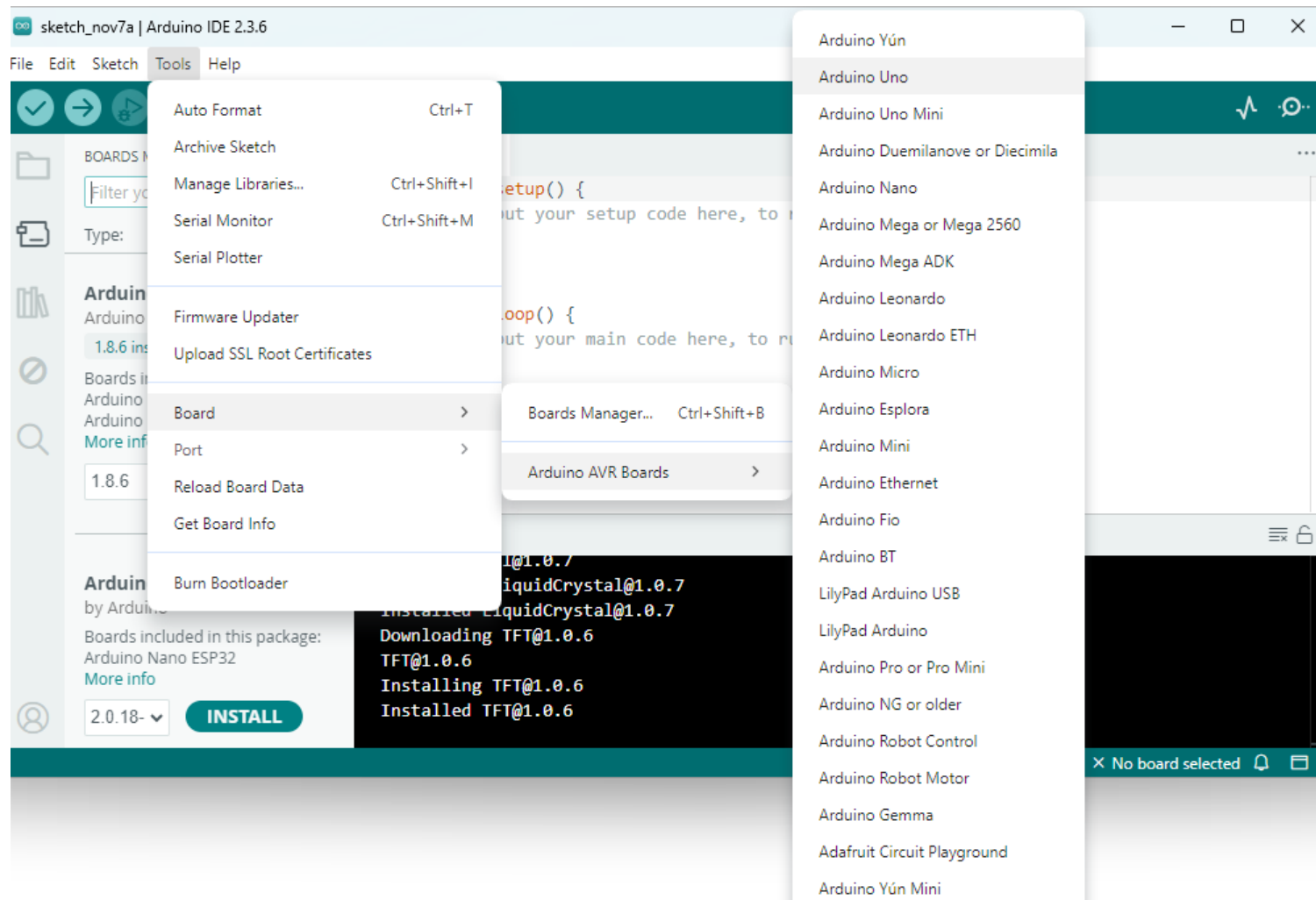


Embedded System

Arduino IDE



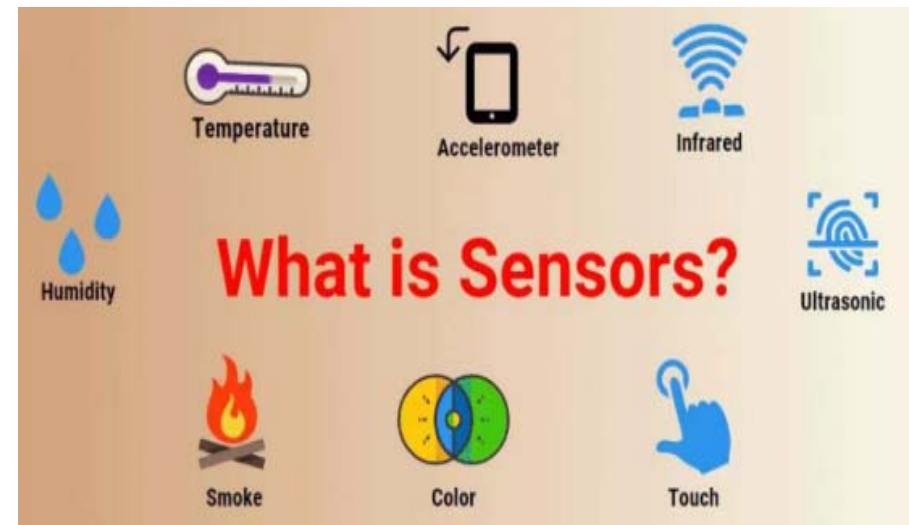
Embedded System



Embedded System

Sensors:

- A **sensor** is a device that **detects changes** in the physical environment and **converts them into electrical signals** that a system (like a microcontroller) can understand.
- Any physical quantity like pressure, force, strain, etc.
- These are classified as
 - Analog Sensor
 - Digital Sensor

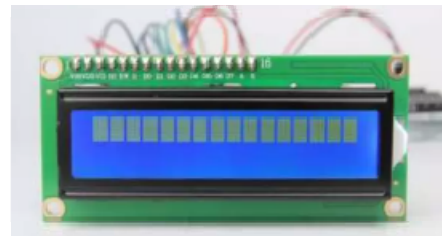


Embedded System

DA-IICT

Actuators:

- An **actuator** is a device that **converts electrical signals** (from a microcontroller or control system) **into physical action**, such as movement, rotation, or force.
- Examples:
 - Motor
 - LED
 - LCD

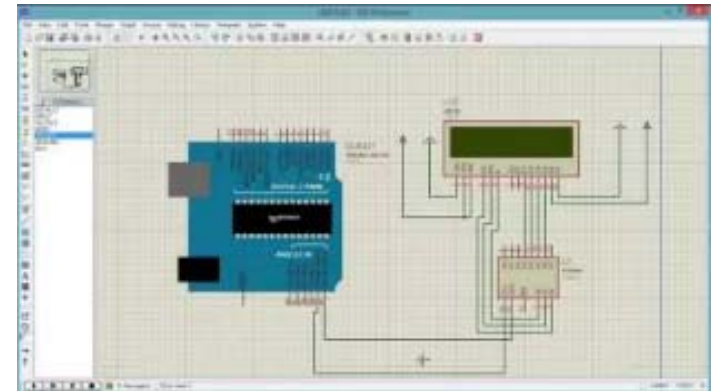


Embedded System

Simulation Software's



Tinkercad



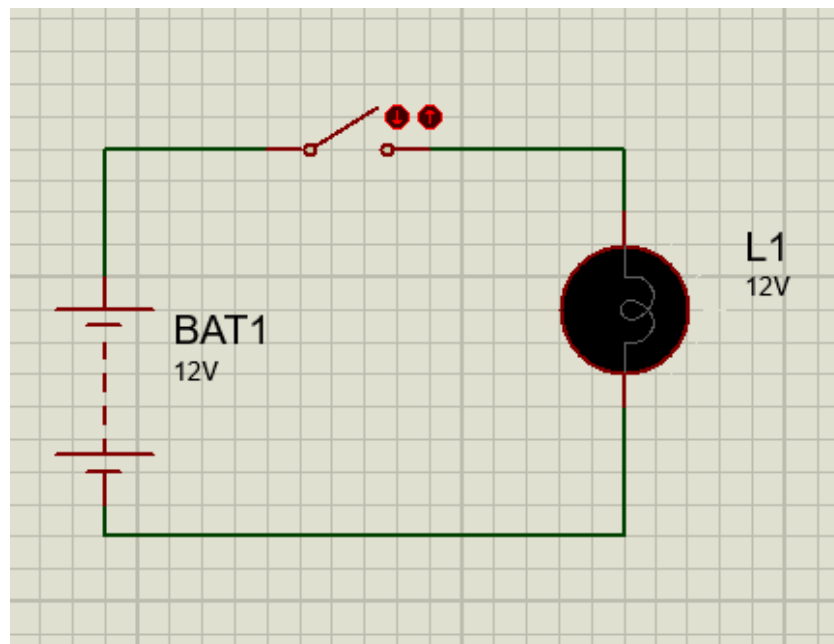
Proteus



Fritzing

Embedded System

Simple Circuit Simulation



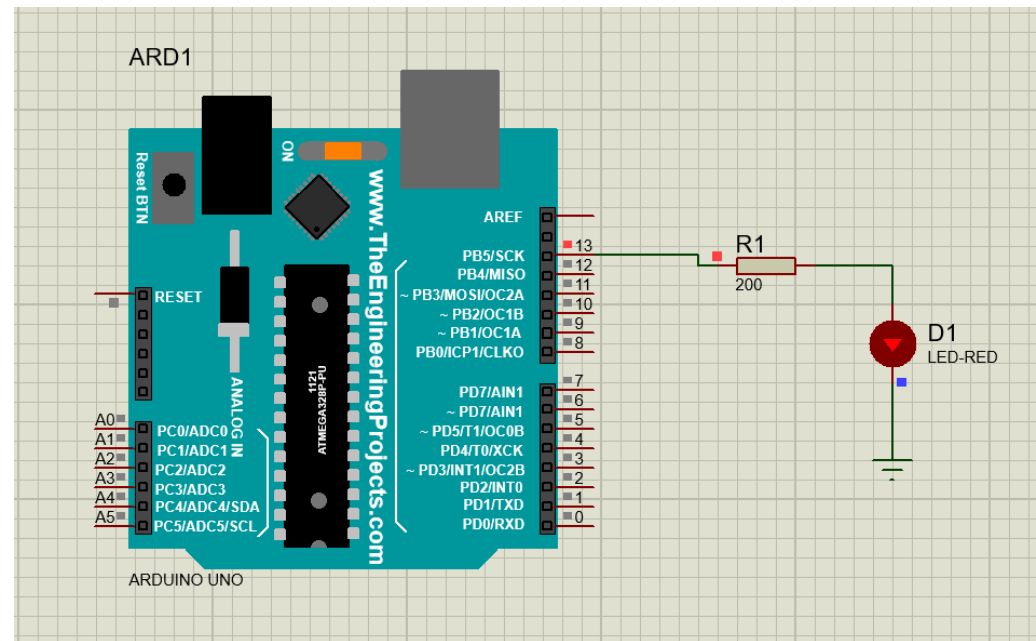
Embedded System

LED Blinking

```
int ledPin1 = 13;

void setup()
{
    pinMode(ledPin1, OUTPUT);
}

void loop() {
    digitalWrite(ledPin1, HIGH);
    delay(2000);
    digitalWrite(ledPin1, LOW);
    delay(2000);
}
```



Embedded System

LED Blinking

```
int ledPin1 = 13;  
int ledPin2 = 12;
```

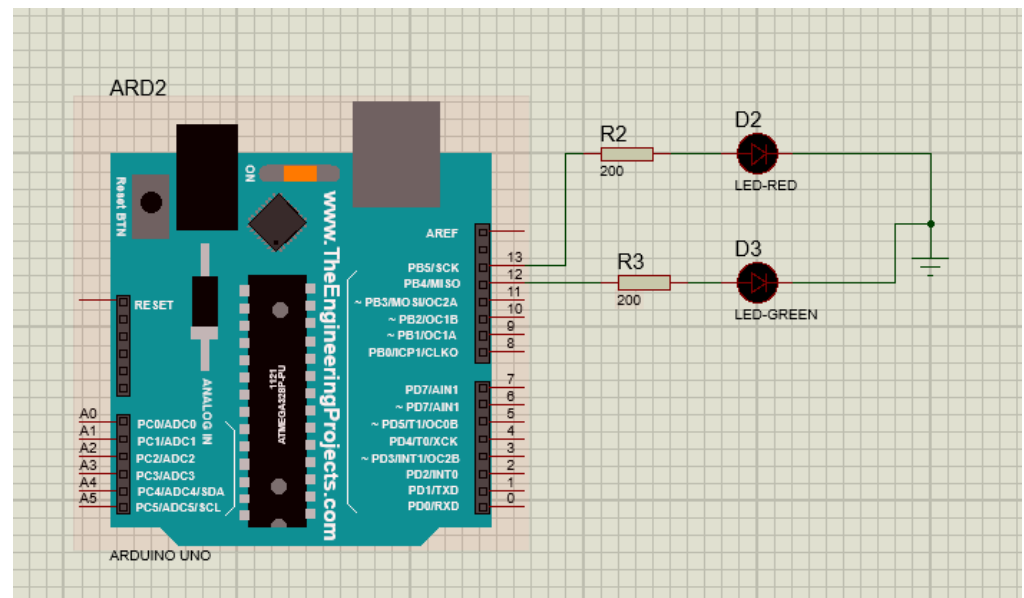
```
void setup()
```

```
{  
    pinMode(ledPin1, OUTPUT);  
    pinMode(ledPin2, OUTPUT);  
}
```

```
void loop() {
```

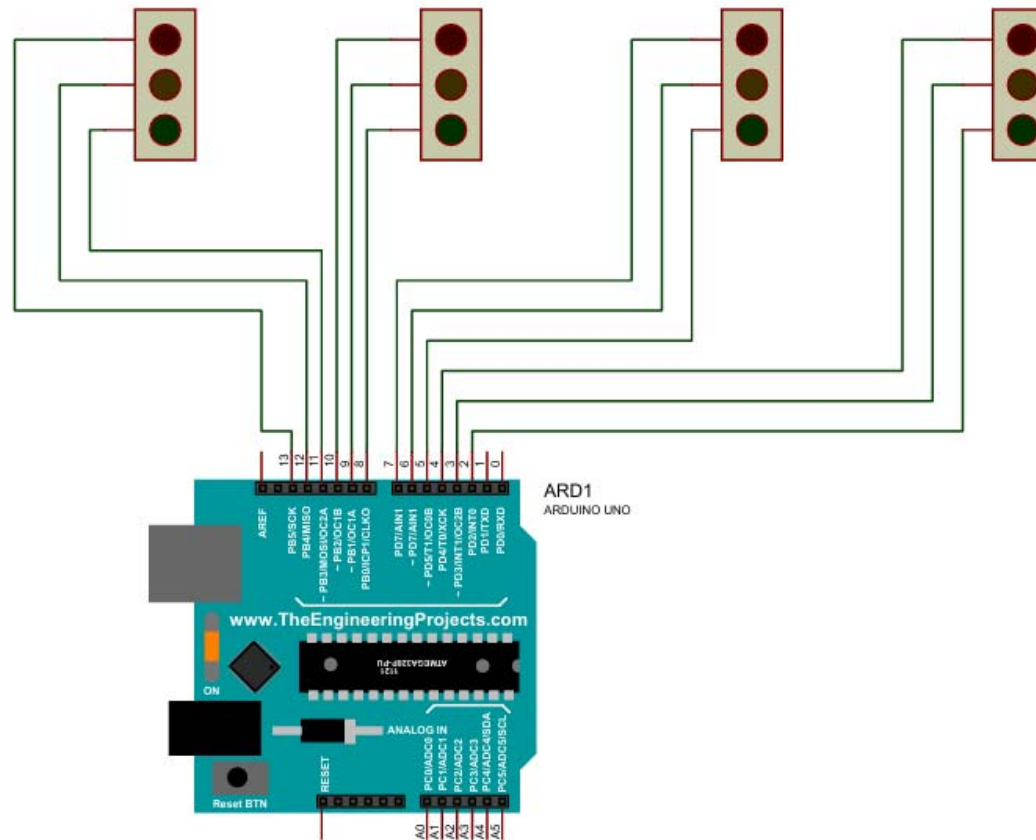
```
    digitalWrite(ledPin1, HIGH);  
    digitalWrite(ledPin2, LOW);  
    delay(2000);  
    digitalWrite(ledPin1, LOW);  
    digitalWrite(ledPin2, HIGH);  
    delay(2000);
```

```
}
```



Embedded System

Traffic Light



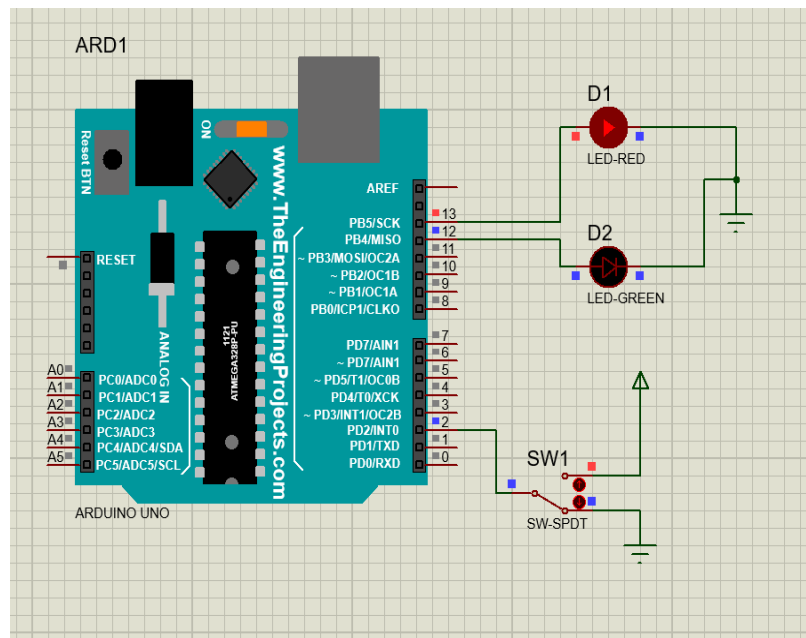
Embedded System

LED Blinking with Switch

```
int ledPin1 = 13;
int ledPin2 = 12;
int inputPin = 2;

int reading;

void setup() {
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);
  pinMode(inputPin, INPUT);
}
```



```
void loop() {
  reading = digitalRead(inputPin);

  if (reading == LOW)
  {
    digitalWrite(ledPin1, HIGH);
    digitalWrite(ledPin2, LOW);
  }
  else
  {
    digitalWrite(ledPin1, LOW);
    digitalWrite(ledPin2, HIGH);
  }
  delay(2000);
}
```

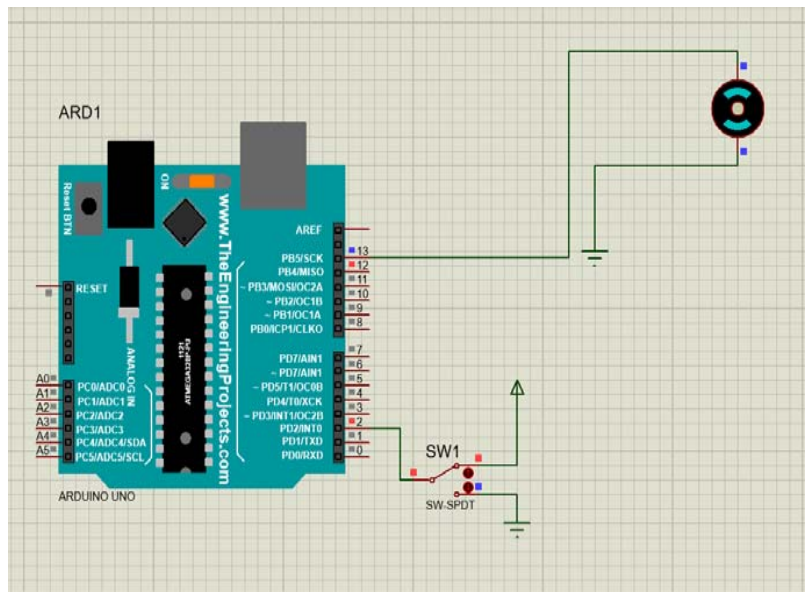
Embedded System

LED Blinking with Switch

```
int ledPin1 = 13;
int ledPin2 = 12;
int inputPin = 2;

int reading;

void setup() {
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);
  pinMode(inputPin, INPUT);
}
```



```
void loop() {
  reading = digitalRead(inputPin);

  if (reading == LOW)
  {
    digitalWrite(ledPin1, HIGH);
    digitalWrite(ledPin2, LOW);
  }
  else
  {
    digitalWrite(ledPin1, LOW);
    digitalWrite(ledPin2, HIGH);
  }
  delay(2000);
}
```

299