



MAINCRAFTS
TECHNOLOGY

- TASK – 1:
- 1. Smart temperature monitor
- 2. Smart lighting system
- 3. MOTION detection alarm

INTRODUCTION TO EMBEDDED SYSTEMS...

- An Embedded System is a special-purpose computer system designed to perform a specific task or set of tasks.
- It is embedded inside a larger system, unlike a general-purpose computer (like a PC or laptop).
- Embedded systems are the brains behind most smart devices today.
- They are specialized, efficient, and widely used in modern technology, from home appliances to industrial machines.

INTRODUCTION TO EMBEDDED SYSTEMS

CHARACTERISTICS OF EMBEDDED SYSTEMS

- ✓□ Dedicated function (only one task)
- ✓□ Real-time operation (fast response)
- ✓□ Small size & low power consumption
- ✓□ High reliability
- ✓□ Cost-effective
- ✓□ Embedded in other devices

COMPONENTS OF EMBEDDED SYSTEMS (mainly)

- Microcontroller/Microprocessor
- Memory (RAM, ROM/FLASH)
- Input devices (Sensor: light, pressure, etc...)
- Output devices (LED, LCD, Motors, etc...)
- Communication interfaces (UART, SPI, I2C, etc...)

SMART LIGHTING SYSTEM

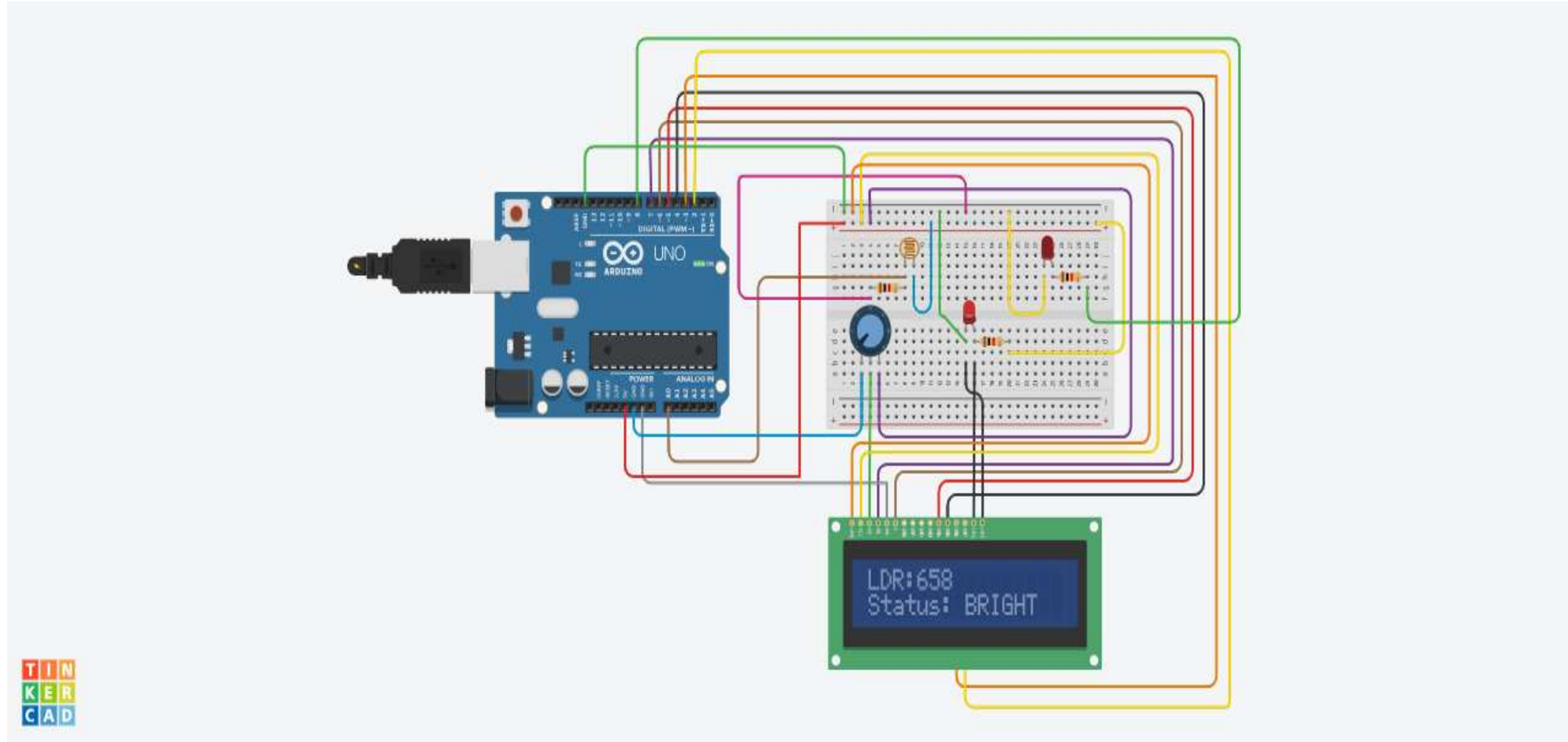
LED turns ON/OFF based on sensor input



TASK-1: SMART LIGHTING SYSTEM (CIRCUIT DIAGRAM)

- **Component List (Simplified)**

- ✓ Arduino Uno
- ✓ LDR sensor
- ✓ LED
- ✓ 220Ω resistor
- ✓ 10kΩ resistor
- ✓ Breadboard + wires
- ✓ Power supply (5V)
- ✓ LCD Display



TASK-1: SMART LIGHTING SYSTEM (EXPLANATION)

1. Sensor Detects the Environment

- The system uses a sensor such as **LDR, PIR, or IR**.
- The sensor continuously measures the surrounding conditions:
 - **LDR** → detects light intensity (dark or bright)
 - **PIR** → detects human motion
 - **IR** → detects obstacle or object presence

2. Sensor Sends Signal to Microcontroller

- The sensor converts the physical input into an electrical signal.
- This signal is sent to the microcontroller (Arduino/ESP32).
- Example:
 - **LDR** gives an analog voltage value depending on light intensity.
 - **PIR** gives a digital HIGH/LOW signal

3. Microcontroller Processes the Signal

- The microcontroller reads the sensor output continuously.
- It compares the sensor value with a **pre-defined threshold**.
- Example logic:
 - If light intensity is **low**, the system considers it **night time**.
 - If motion is detected, the system considers someone is present.

4. LED Turns ON/OFF Based on Condition

- When the sensor output crosses the threshold:
 - The microcontroller sends a signal to the LED pin.
 - LED turns **ON** (HIGH).
- When the condition is not met:
 - LED remains **OFF** (LOW)

TASK-1: SMART LIGHTING SYSTEM (APPLICATIONS)

- Automatic street lights
- Home lighting automation
- Security lights (motion-based)
- Office building lighting control
- Garden/pathway lights
- Parking lot lights
- Solar-powered lighting systems

SOURCE CODE WITH

COMMENTS

sketch_jan19d.ino

```
1  #include <LiquidCrystal.h>           // Include the LiquidCrystal library to control the LCD display
2  LiquidCrystal lcd(7, 6, 5, 4, 3, 2); // Initialize LCD with pin connections LCD pins: RS, EN, D4, D5, D6, D7
3  int ldrPin = A0;                     // LDR sensor is connected to analog pin A0
4  int ledPin = 9;                     // LED is connected to digital pin 9 (PWM pin)
5  int ldrValue;                       // Variable to store LDR analog value
6  void setup()
7  {
8      pinMode(ledPin, OUTPUT);        // Set LED pin as output to control the LED
9      lcd.begin(16, 2);               // Initialize the LCD with 16 columns and 2 rows
10     lcd.print("Object Detect");      // Display "Object Detect" on LCD
11     delay(2000);                     // Wait for 2 seconds so user can read the message
12     lcd.clear();                     // Clear the LCD screen
13 }
14 void loop(){
15     ldrValue = analogRead(ldrPin);   // Read analog value from LDR (0 to 1023)
16     lcd.setCursor(0, 0);             // Set cursor to first row, first column
17     lcd.print("LDR:");               // Display text "LDR:"
18     lcd.print(ldrValue);              // Display the current LDR value
19     lcd.print("  ");                 // Add spaces to clear leftover digits
20     lcd.setCursor(0, 1);             // Set cursor to second row, first column
21     if (ldrValue < 500){              // If LDR value is less than 500 (light blocked)
22         digitalWrite(ledPin, HIGH);  // Turn ON the LED
23         lcd.print("Status:  DARK");  // Display message "DARK"
24     }
25     else{                             // If LDR value is greater than or equal to 500
26         digitalWrite(ledPin, LOW);   // Turn OFF the LED
27         lcd.print("Status: BRIGHT"); // Display message "BRIGHT"
28     }
29     delay(500);                       // Wait for 0.5 seconds before next reading
30 }
```


THANK YOU