The project involves controlling LEDs based on light intensity (lux value) and motion detection, with additional functionality to monitor power consumption.

**Components**

1. LEDs: Eight LEDs connected to digital pins 2 to 9.

2. Switch: Connected to pin 10, used to turn the entire circuit on or off.

3. PIR Sensor: Connected to pin 11, used to detect motion.

4. ACS712 Current Sensor: Connected to analog pin A1, used to measure current and calculate power consumption.

5. Serial Input: Used to input the lux value.

### Code Explanation

#### Global Variables and Constants

```cpp

const int leds[] = {2, 3, 4, 5, 6, 7, 8, 9}; // LED pins

const int switchPin = 10; // Switch pin

const int pirPin = 11; // PIR sensor connected to digital pin 11

const int currentSensorPin = A1; // ACS712 sensor connected to A1

float luxValue = -1.0; // Initialize lux value to a negative value

float powerWatts = 0.0;

bool motionDetected = false;

```

- `leds[]`: Array holding the pins to which the LEDs are connected.

- `switchPin`: Pin for the switch.

- `pirPin`: Pin for the PIR sensor.

- `currentSensorPin`: Pin for the ACS712 current sensor.

- `luxValue`: Stores the lux value read from the serial input.

- `powerWatts`: Stores the cumulative power consumption.

- `motionDetected`: Flag to indicate if motion is detected.

#### Setup Function

```cpp

void setup() {

for (int i = 0; i < 8; i++) {

pinMode(leds[i], OUTPUT);

}

pinMode(switchPin, INPUT\_PULLUP);

pinMode(pirPin, INPUT);

pinMode(currentSensorPin, INPUT);

Serial.begin(9600);

}

```

- Initializes the pins for the LEDs, switch, PIR sensor, and current sensor.

- Sets the switch pin to `INPUT\_PULLUP` to use the internal pull-up resistor.

- Begins serial communication at a baud rate of 9600.

#### Loop Function

```cpp

void loop() {

int pirState = digitalRead(pirPin);

// Read current sensor and update power consumption

int analogValue = analogRead(currentSensorPin);

float current = (analogValue - 512) / 102.4; // ACS712 sensitivity for 20A model

powerWatts += abs(current) 5; // Assuming a 5V supply voltage

if (Serial.available() > 0) {

luxValue = Serial.parseFloat();

}

if (pirState == HIGH) {

// Motion detected, make LED at pin 2 brighter

analogWrite(leds[0], 255); // Full brightness

motionDetected = true;

} else {

motionDetected = false;

}

if (digitalRead(switchPin) == LOW) {

// Switch is off, turn off all LEDs

for (int i = 0; i < 8; i++) {

digitalWrite(leds[i], LOW);

Serial.print("LED ");

Serial.print(i + 2); // Adding 2 to i to print the actual pin number

Serial.println(" is OFF");

}

} else {

// Switch is on

if (luxValue < 100) {

// Lux value is less than 100, turn on all LEDs except for pin 2

for (int i = 0; i < 8; i++) {

if (i != 0) {

analogWrite(leds[i], 255); // Full brightness

Serial.print("LED ");

Serial.print(i + 2); // Adding 2 to i to print the actual pin number

Serial.println(" is ON at full brightness");

} else {

if (motionDetected) {

analogWrite(leds[i], 255); // Full brightness if motion detected

Serial.println("LED 2 is ON at full brightness due to motion detection");

} else {

digitalWrite(leds[i], LOW); // Pin 2 off

Serial.println("LED 2 is OFF");

}

}

}

} else if (luxValue >= 100 && luxValue <= 10000) {

if (luxValue >= 300 && luxValue <= 500) {

// Lux value between 300 and 500, set brightness to 70%

for (int i = 0; i < 8; i++) {

if (i != 0) {

analogWrite(leds[i], 179); // 70% brightness

Serial.print("LED ");

Serial.print(i + 2); // Adding 2 to i to print the actual pin number

Serial.println(" is ON at 70% brightness");

} else {

if (motionDetected) {

analogWrite(leds[i], 255); // Full brightness if motion detected

Serial.println("LED 2 is ON at full brightness due to motion detection");

} else {

digitalWrite(leds[i], LOW); // Pin 2 off

Serial.println("LED 2 is OFF");

}

}

}

} else if (luxValue >= 900 && luxValue <= 1100) {

// Lux value approximately 1000, set brightness to 50%

for (int i = 0; i < 8; i++) {

if (i != 0) {

analogWrite(leds[i], 128); // 50% brightness

Serial.print("LED ");

Serial.print(i + 2); // Adding 2 to i to print the actual pin number

Serial.println(" is ON at 50% brightness");

} else {

if (motionDetected) {

analogWrite(leds[i], 255); // Full brightness if motion detected

Serial.println("LED 2 is ON at full brightness due to motion detection");

} else {

digitalWrite(leds[i], LOW); // Pin 2 off

Serial.println("LED 2 is OFF");

}

}

}

} else {

// Lux value in the range (100, 300) or (500, 10000), turn off all LEDs except for pin 2

for (int i = 0; i < 8; i++) {

if (i != 0) {

digitalWrite(leds[i], LOW);

Serial.print("LED ");

Serial.print(i + 2); // Adding 2 to i to print the actual pin number

Serial.println(" is OFF");

} else {

if (motionDetected) {

analogWrite(leds[i], 255); // Full brightness if motion detected

Serial.println("LED 2 is ON at full brightness due to motion detection");

} else {

digitalWrite(leds[i], LOW); // Pin 2 off

Serial.println("LED 2 is OFF");

}

}

}

}

} else {

// Lux value greater than 10,000, turn off all LEDs

for (int i = 0; i < 8; i++) {

digitalWrite(leds[i], LOW);

Serial.print("LED ");

Serial.print(i + 2); // Adding 2 to i to print the actual pin number

Serial.println(" is OFF");

}

}

}

// Print lux value, PIR state, and cumulative power consumption to Serial Monitor

Serial.print("Lux Value: ");

Serial.println(luxValue);

Serial.print("PIR State: ");

Serial.println(pirState);

Serial.print("Cumulative Power Consumption (Watts): ");

Serial.println(powerWatts);

delay(1000); // Adjust the delay as needed

}

```

**Functionality**

1. PIR Sensor:

- Reads the state of the PIR sensor. If motion is detected (`pirState == HIGH`), the LED at pin 2 (leds[0]) is set to full brightness (255). Otherwise, it is dimmed or turned off depending on the lux value.

- Sets the `motionDetected` flag to true if motion is detected, otherwise sets it to false.

2. Current Sensor:

- Reads the analog value from the current sensor to calculate the current and updates the cumulative power consumption (`powerWatts`).

3. Serial Input:

- Reads the lux value from the serial input if available.

4. Switch:

- Checks the state of the switch. If the switch is off (`digitalRead(switchPin) == LOW`), all LEDs are turned off.

- If the switch is on, the LEDs are controlled based on the lux value and motion detection.

5. LED Control:

- Depending on the lux value, sets the brightness of the LEDs (except pin 2) as follows:

- If `luxValue < 100`: All LEDs except pin 2 are set to full brightness.

- If `luxValue` is between 100 and 10000:

- If between 300 and 500: LEDs are set to 70% brightness.

- If between 900 and 1100: LEDs are set to 50% brightness.

- Otherwise

: All LEDs except pin 2 are turned off.

- If `luxValue > 10000`: All LEDs are turned off.

- The LED at pin 2 is handled separately, turning it to full brightness if motion is detected.

6. Serial Output:

- Prints the current lux value, PIR state, and cumulative power consumption to the serial monitor for debugging and monitoring.

This project effectively integrates light intensity control, motion detection, and power monitoring into a single system. The code ensures that the LEDs operate correctly based on the input conditions and provides feedback through the serial monitor for easy debugging and verification.