Stony Brook University

EEO335

Spring 2024

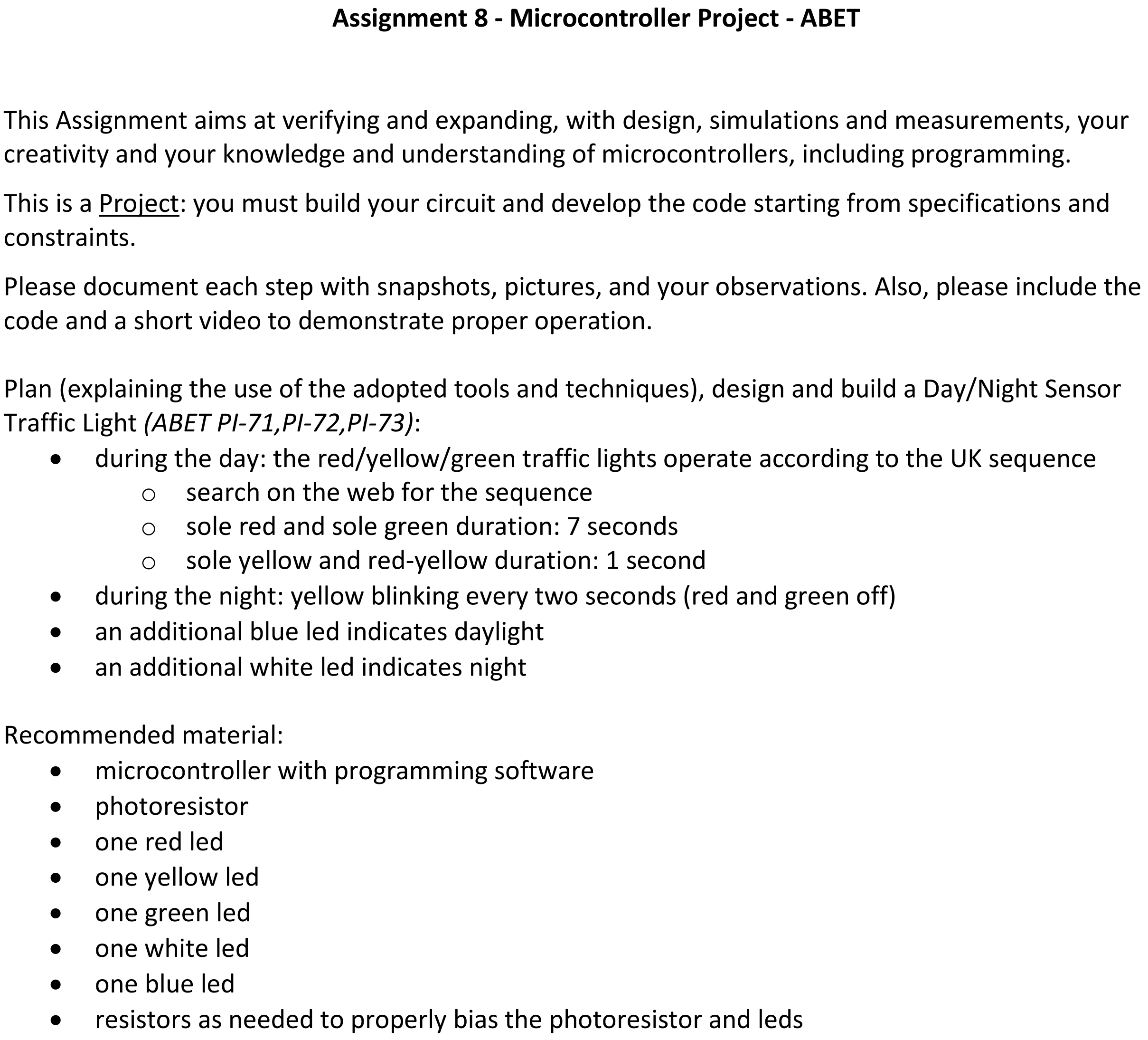
Microcontroller Project

Pete Mills
ID: 115009163

Professor
Gianluigi De Geronimo

April 20, 2024

Copy of Original Assignment



Assignment 8 - Microcontroller Project - ABET

This Assignment aims at verifying and expanding, with design, simulations and measurements, your creativity and your knowledge and understanding of microcontrollers, including programming.

This is a Project: you must build your circuit and develop the code starting from specifications and constraints.

Please document each step with snapshots, pictures, and your observations. Also, please include the code and a short video to demonstrate proper operation.

Plan (explaining the use of the adopted tools and techniques), design and build a Day/Night Sensor Traffic Light *(ABET PI-71,PI-72,PI-73)*:

- during the day: the red/yellow/green traffic lights operate according to the UK sequence
    - search on the web for the sequence
    - sole red and sole green duration: 7 seconds
    - sole yellow and red-yellow duration: 1 second
- during the night: yellow blinking every two seconds (red and green off)
- an additional blue led indicates daylight
- an additional white led indicates night

Recommended material:

- microcontroller with programming software
- photoresistor
- one red led
- one yellow led
- one green led
- one white led
- one blue led
- resistors as needed to properly bias the photoresistor and leds

Overview

In this lab we design and built a Microcontroller based traffic light. We gained experience with microcontrollers, GPIO, ADCs, and programming.

1 Design

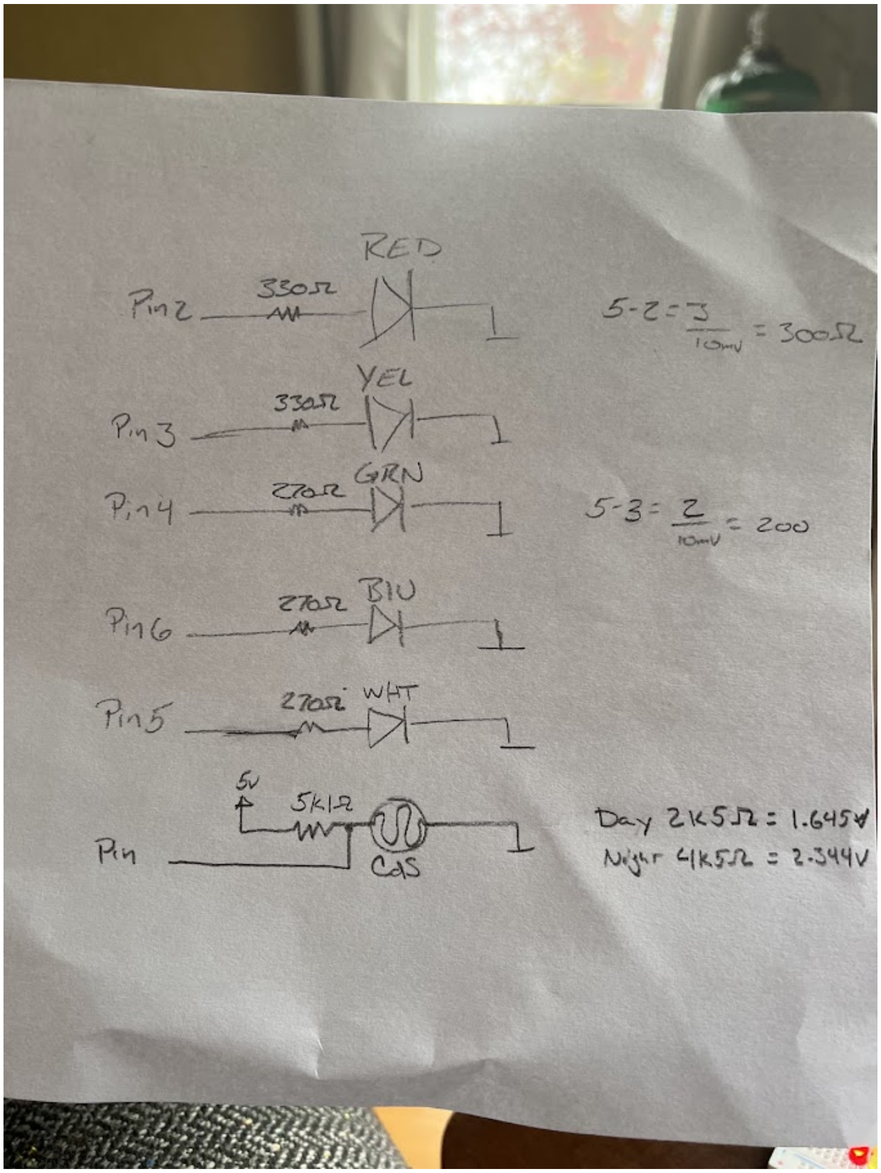


Figure 1: Schematic of circuit showing calculation of current limiting resistors for LEDs and voltage divider formed with a resistor and CdS photoresistor.

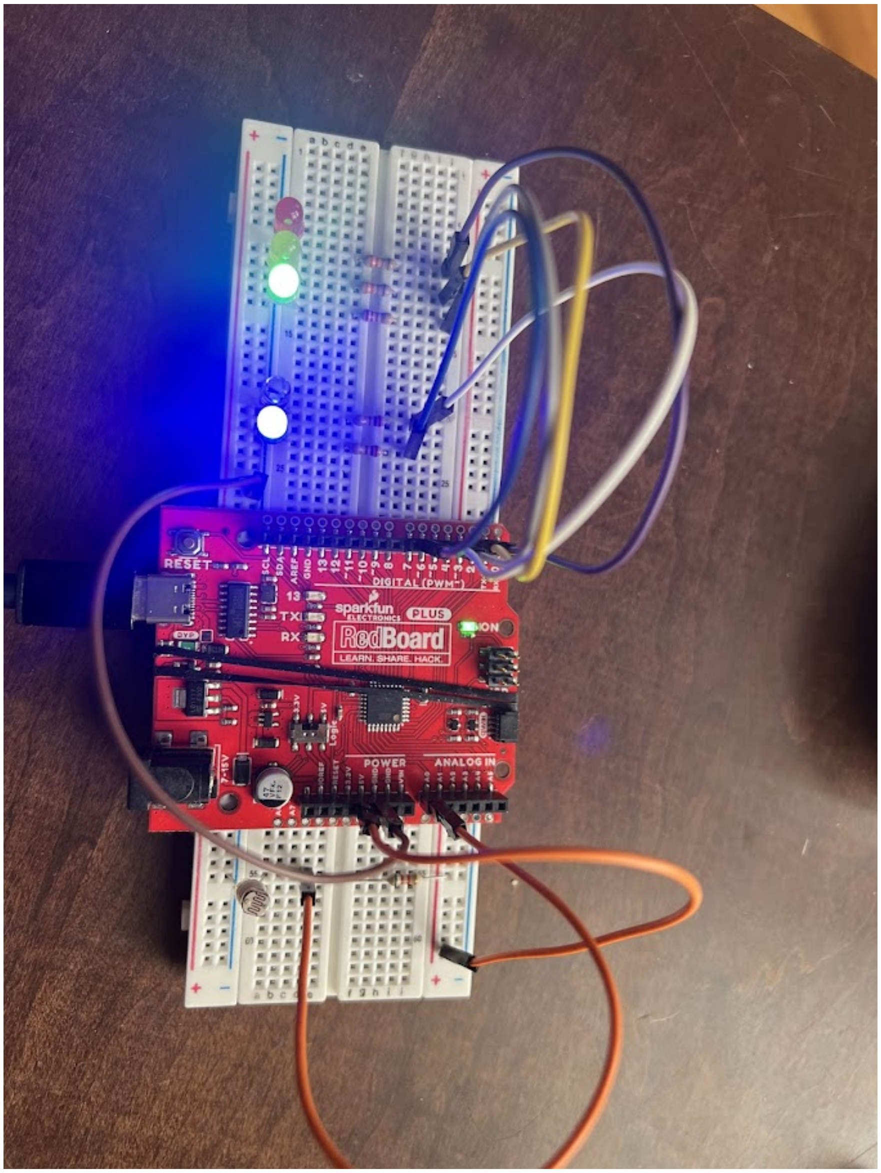
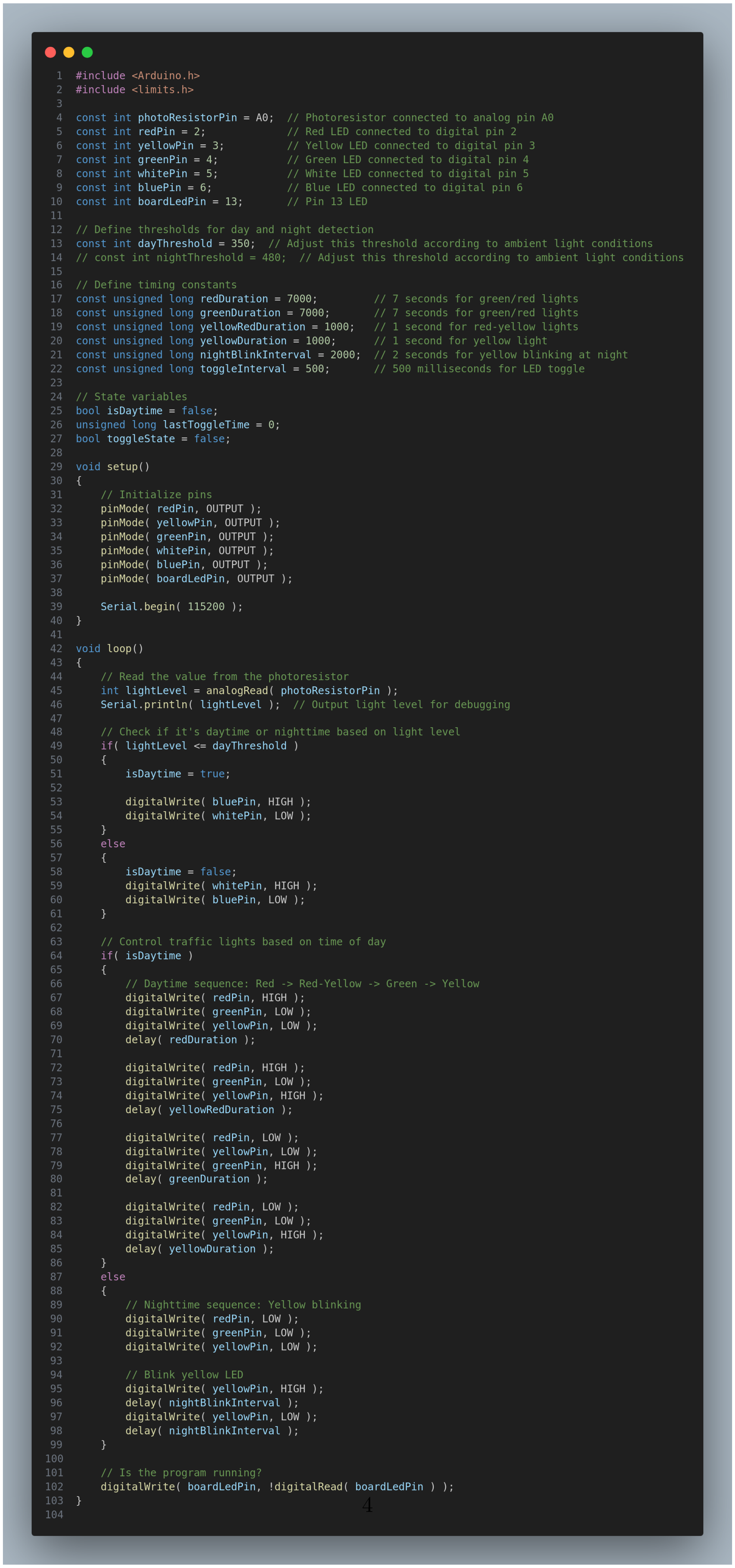


Figure 2: Photo of circuit in operation.



```cpp
#include <Arduino.h>
#include <limits.h>

const int photoResistorPin = A0;  // Photoresistor connected to analog pin A0
const int redPin = 2;             // Red LED connected to digital pin 2
const int yellowPin = 3;          // Yellow LED connected to digital pin 3
const int greenPin = 4;           // Green LED connected to digital pin 4
const int whitePin = 5;           // White LED connected to digital pin 5
const int bluePin = 6;            // Blue LED connected to digital pin 6
const int boardLedPin = 13;       // Pin 13 LED

// Define thresholds for day and night detection
const int dayThreshold = 350;  // Adjust this threshold according to ambient light conditions
// const int nightThreshold = 480;  // Adjust this threshold according to ambient light conditions

// Define timing constants
const unsigned long redDuration = 7000;         // 7 seconds for green/red lights
const unsigned long greenDuration = 7000;       // 7 seconds for green/red lights
const unsigned long yellowRedDuration = 1000;   // 1 second for red-yellow lights
const unsigned long yellowDuration = 1000;      // 1 second for yellow light
const unsigned long nightBlinkInterval = 2000;  // 2 seconds for yellow blinking at night
const unsigned long toggleInterval = 500;       // 500 milliseconds for LED toggle

// State variables
bool isDaytime = false;
unsigned long lastToggleTime = 0;
bool toggleState = false;

void setup()
{
    // Initialize pins
    pinMode( redPin, OUTPUT );
    pinMode( yellowPin, OUTPUT );
    pinMode( greenPin, OUTPUT );
    pinMode( whitePin, OUTPUT );
    pinMode( bluePin, OUTPUT );
    pinMode( boardLedPin, OUTPUT );

    Serial.begin( 115200 );
}

void loop()
{
    // Read the value from the photoresistor
    int lightLevel = analogRead( photoResistorPin );
    Serial.println( lightLevel );  // Output light level for debugging

    // Check if it's daytime or nighttime based on light level
    if( lightLevel <= dayThreshold )
    {
        isDaytime = true;

        digitalWrite( bluePin, HIGH );
        digitalWrite( whitePin, LOW );
    }
    else
    {
        isDaytime = false;
        digitalWrite( whitePin, HIGH );
        digitalWrite( bluePin, LOW );
    }

    // Control traffic lights based on time of day
    if( isDaytime )
    {
        // Daytime sequence: Red -> Red-Yellow -> Green -> Yellow
        digitalWrite( redPin, HIGH );
        digitalWrite( greenPin, LOW );
        digitalWrite( yellowPin, LOW );
        delay( redDuration );

        digitalWrite( redPin, HIGH );
        digitalWrite( greenPin, LOW );
        digitalWrite( yellowPin, HIGH );
        delay( yellowRedDuration );

        digitalWrite( redPin, LOW );
        digitalWrite( yellowPin, LOW );
        digitalWrite( greenPin, HIGH );
        delay( greenDuration );

        digitalWrite( redPin, LOW );
        digitalWrite( greenPin, LOW );
        digitalWrite( yellowPin, HIGH );
        delay( yellowDuration );
    }
    else
    {
        // Nighttime sequence: Yellow blinking
        digitalWrite( redPin, LOW );
        digitalWrite( greenPin, LOW );
        digitalWrite( yellowPin, LOW );

        // Blink yellow LED
        digitalWrite( yellowPin, HIGH );
        delay( nightBlinkInterval );
        digitalWrite( yellowPin, LOW );
        delay( nightBlinkInterval );
    }

    // Is the program running?
    digitalWrite( boardLedPin, !digitalRead( boardLedPin ) );
}
```

Figure 3: Code with comments describing the operation of the firmware to meet the requirements of the lab.