# CAB202 Assignment:

Student Name: William Ee

Student Number: N10494448

TinkerCAD Link: https://www.tinkercad.com/things/26QCszVqaqE-assignment/editel?sharecode=5iWBRDcEPTHEnZfQfwAoseQPxBCOtNrOJV\_Ifo566lY

Video Demonstration Link: https://youtu.be/p9GtenfXYtM

## Introduction

The application created is a nightlight in the form of a bedside lamp which is primarily used for providing comfort for children whilst sleeping, but can also be used to reveal the layout of the room such as stairs, hallways or bathroom to avoid tripping at night. This nightlight will have additional utilities such as enabling the user to select the time they wish the nightlight to remain on for which can be used to conserve electricity. Additional functionality added into the application would be the automatic setting of the brightness of the night light depending on the surrounding brightness. This increases convenience and adds automation to the nightlight.

|  |  |
| --- | --- |
| Serial I/O – UART | UART is used for serial input and output. Serial output is used to display instructions or feedback user via the console such as ‘enter the amount of time’. Serial input allows the user to input the amount of time they wish the nightlight to be on, enclosed in quotation marks e.g. “10”, via the console. |
| Digital I/O - Switch | After the user has entered the desired time, the user will need to press the button switch to turn the nightlight on. The user may turn off the nightlight manually by pressing the switch button again. |
| Digital I/O – Debouncing | Debouncing is used to accurately recognise a button click whereby the switch is pressed then released; preventing the recognition of multiple button clicks caused by bouncing. |
| Digital I/O – LED (lightbulb) | Primary light source of the application, hence proving the main functionality of a nightlight. |
| Analog Output – PWM | PWM is used to gradually dim the nightlight over a period of time. This is done by repetitively lowering the compare value which will increase the amount of time the PWM output PIN is set to low (lower the duty cycle), making the light appear dimmer. |
| Analog Input – ADC | ADC is used to detect the brightness level from an ambient light sensor and convert this analogue value into a digital value upon button press. This digital value is then used to determine the initial duty cycle/brightness of the lightbulb, where a high value results in a low brightness of the lightbulb and vice versa. |
| LCD | The LCD is used to display the countdown timer for the nightlight as well as its current brightness. It also displays instructions e.g. ‘enter time’ or ‘press button’. |
| Timers (other than debouncing or PWM) | The timer has been used to cause an interrupt every second which will update the countdown time and current brightness of nightlight. This will display on the LCD as well as serial output the updated countdown time. |
| Advanced Functionality  (LED matrix) | The LED matrix is a screen display for the user to look at. In this scenario it is in the shape of a star which is targeted towards children. This is implemented by multiplexing using a timer overflow interrupt. Though only one column of LEDs is on at a time, as they are turned on and off in quick succession, to the human eye the entire matrix appears on. |

## Diagram, schematic Description automatically generatedSchematic

­

## Wiring Instructions

**Breadboard Components**

Connect (S1) Pushbutton: place the pushbutton on the left-hand side of the breadboard. Connect pin 13 (PB5) to terminal 1b of the pushbutton. Connect terminal 2a of the pushbutton to the 5v power supply. Connect terminal 1a to one end of a 220-ohm resistor (R3). Connect the other end of the resistor to ground.

Connect (Q1) Ambient Light Sensor: place the ambient light sensor to the right of the pushbutton on the breadboard. Connect the collector of the ambient light sensor to 5v power supply. Connect the ADC pin A0 (PC0) to the emitter of the ambient light sensor. Also connect one end of a 2 k-ohm resistor (R1) to the emitter of the ambient light sensor, perpendicular of the A0 pin connection to the emitter. Connect the other end of the resistor to ground, parallel of the A0 pin connection to the emitter.

Connect (L1) Lightbulb: place the light bulb to the right of the ambient light sensor on the breadboard. Connect the PWM output pin 11 (PB3) to Terminal 1 of the lightbulb. Connect Terminal 2 of the lightbulb to one end of a 2 k-ohm resistor (R3), then connect the other end of the resistor to ground.

**LCD** (U2)

Connect LCD data pins: The LCD is connected in 4-pin mode. Connect data pin 4 (DB4) of the LCD to pin 4 (PD4) of the microcontroller; connect data pin 5 (DB5) of the LCD to pin 5 (PD5) of the microcontroller; connect data pin 6 (DB6) of the LCD to pin 6 of the microcontroller; and connect data pin 7 (DB7) of the LCD to pin 7 of the microcontroller.

Connect other pins: Connect the Register Select (RS) pin on the LCD to pin 9 (PB2) on the microcontroller. Connect the Enable pin (E) pin on the LCD to pin 8 (PB0) on the microcontroller. Connect the Power (VCC) pin on the LCD to the 5v power supply on the microcontroller. Connect the Read/Write pin (R/W) on the LCD to ground on the microcontroller. Also connect this pin to the Contrast (VO) pin on the LCD. Subsequently, connect the VO pin to the Ground (GND) pin on the LCD. Connect the LED anode pin (LED) to the end of a 220-ohm resistor (R4), then connect the other end of the resistor to the 5v power supply from the microcontroller. Then connect the LED cathode pin (LED) which is to the right of the LED anode pin to ground on the microcontroller.

**LED Matrix**

Assembling LED matrix: assemble 20 Yellow LEDs (D1-D20) into four rows and five columns from bottom to top, right to left (starting from bottom right (D1), ending top left (D20)). This project uses a common cathode LED matrix, meaning that the columns of the matrix are connected by the LED’s anodes, and the rows are connected by the LED’s cathodes. Therefore, to turn a particular LED on in the matrix, the pin connected to the LED’s anode/column would be set to high and the pin connected to the particular row that the LED is on would be set to low/grounded.

Connecting columns: To connect the first column (the rightmost column); connect output pin A1 (PC1) on the microcontroller to D1’s anode located at the bottom right of the LED matrix; connect D1’s anode to the next LED’s (D2) anode in the same column; connect D2’s anode to D3’s anode; connect D3’s anode to D4’s anode. At this point the first column of the LED matrix has been connected.  
Connect the remaining output pins to the bottom LED’s anodes on their corresponding columns as follows: output pin A2 (PC2) to column 2 (second rightmost column) by connecting it to D5’s anode, output pin A3 (PC3) to column 3 (middle column), output pin A4(PC4) to column 4 (second leftmost column) and output pin A5 (PC5) to column 5 (leftmost column). Then for each column, proceed to connect each LED’s anode to next LED’s anode above it in the column, and repeat this until reaching the top LED.

Connecting rows: To connect the rows, connect each LED’s cathode in each row to the next LED’s cathode to the left. For example, in the bottom row, you would connect D1’s cathode to D5’s cathode, then D5’s cathode to D9’s cathode, and D9’s cathode to D13’s cathode, then D13’s cathode to D17’s cathode. Repeat this process for the other LEDs in each row.  
 After connecting the rows of LEDs, connect each row’s leftmost LED’s cathode to the end of a 100-ohm resistor, ensuring that each row is connected to one resistor. Afterwards, connect the other end of the bottom row’s resistor to output pin 12 (PB4); connect the other end of the second bottom-most resistor to output pin 10 (PB2); connect the other end of the second top most row’s resistor to output pin 3 (PB3); and connect the end of top row’s resistor to output pin 2 (PB2). With this the LED matrix should be connected properly.