SUMMER INTERNSHIP Embedded C

- K GOUTHAMI
- BU21EECE0100498

TASK 12

Hands-on Activity-1

- ❑ Write a program to count no. of bits which are set in given binary pattern
- ❑ Write a program to set 5th and 12th bits in a 16-bit unsigned integer
- ❑ Write a program to clear 6th and 19th bits in a 32-bit unsigned integer
- ❑ Write a program to flip even positioned bits in a 16-bit unsigned integer
- ❑ An IP Address will be in the form of "a.b,c.d" format, where a,b,c,d will be in the range of 0-255. Given a,b,c,d values (or string format) pack them into 32-bit unsigned integer.
- ❑ Given an unsigned 32-bit integer holding packed IPv4 address, convert it into "a.b.c.d" format.
- ❑ Convert MAC address into 48-bit binary pattern
- ❑ Convert 48-bit binary pattern as MAC address
- ❑ Arduino examples using Bare metal code (Register level Bit Manipulations)
  - ➔ Blinky
  - ➔ LED controlling using PushButton

Q1)

```
#include <stdio.h> int
countSetBits(int n) {
int count = 0;     while
(n) {         count += n
& 1;         n >>= 1;
   }
   return count;
}
int main() {     int num;     prin ("Enter an integer: ");
scanf("%d", &num);     int setBits = countSetBits(num);
prin ("Number of set bits in %d is %d\n", num, setBits);
   return 0;
}
```

Q2)

```
#include <stdio.h>
int main()
{
   unsigned short int value = 0;     unsigned short int mask = (1 << 4) |
(1 << 11);     value |= mask;     prin ("The value a er se     ng the 5th and
12th bits is: %u\n", value);
   return 0;
}
```

Q3)

```
#include <stdio.h> unsigned int
clearBits(unsigned int num) {     unsigned int
mask = ~((1 << 5) | (1 << 18));      return
num & mask;
```

```
}
int main() {    unsigned int num;    prin ("Enter a 32-bit unsigned
integer: ");    scanf("%u", &num);    unsigned int result =
clearBits(num);    prin ("Result a er clearing the 6th and 19th bits:
%u\n", result);
   return 0;
}
```

Q4)

```
#include <stdio.h> unsigned short
flipEvenBits(unsigned short num) {    unsigned
short mask = 0x5555;    return num ^ mask;
}

int main() {    unsigned short num;    prin ("Enter a 16-bit unsigned
integer: ");    scanf("%hu", &num);    unsigned short result =
flipEvenBits(num);    prin ("Result a er flipping the even-posi oned
bits: %hu\n", result);
   return 0;
}
```

Q5)

```
#include <stdio.h> unsigned int packIP(unsigned char a, unsigned char b, unsigned char
c, unsigned char d) {    return (a << 24) | (b << 16) | (c << 8) | d;

}

int main() {    unsigned char a = 192;
unsigned char b = 168;    unsigned char c = 1;
unsigned char d = 100;    unsigned int packedIP
```

```
= packIP(a, b, c, d);     prin ("Packed IP address:
0x%X\n", packedIP);


   return 0;
}
```

Q6)

```
#include <stdio.h> int main() {     unsigned int packed_ip =
0xC0A80164;     unsigned char a = (packed_ip >> 24) & 0xFF;
unsigned char b = (packed_ip >> 16) & 0xFF;      unsigned char c
= (packed_ip >> 8) & 0xFF;       unsigned char d = packed_ip &
0xFF;             prin ("The unpacked IP address is:
%u.%u.%u.%u\n", a, b, c, d);     return 0;
}
```

Q7)

```
#include <stdio.h> #include <stdlib.h> unsigned long long
convertMACAddress(const char *mac) {
   unsigned int bytes[6];
   if (sscanf(mac, "%x:%x:%x:%x:%x:%x", &bytes[0], &bytes[1], &bytes[2], &bytes[3], &bytes[4],
&bytes[5]) != 6) {         fprin (stderr, "Invalid MAC
address format.\n");         exit(EXIT_FAILURE);
   }
   unsigned long long macBinary = 0;
   for (int i = 0; i < 6; ++i) {
      macBinary = (macBinary << 8) | (bytes[i] & 0xFF);
   }
   return macBinary;
}
int main() {     char macString[18];     prin ("Enter MAC address
in the format XX:XX:XX:XX:XX:XX: ");     if (scanf("%17s",
```

```
macString) != 1) {         fprin (stderr, "Failed to read MAC
address.\n");         return EXIT_FAILURE;
   }
   unsigned long long macBinary = convertMACAddress(macString);
prin ("MAC address in 48-bit binary pa ern: %012llx\n", macBinary);
   return 0;
}
```

Q8)

```
#include <stdio.h> #include <stdlib.h>
void binaryToMac(const char* binary)
{
   unsigned int bytes[6] = {0};     for (int i = 0; i <
48; ++i) {         bytes[i / 8] = (bytes[i / 8] << 1) |
(binary[i] - '0');      }
   prin ("MAC Address: %02X:%02X:%02X:%02X:%02X:%02X\n",
         bytes[0], bytes[1], bytes[2], bytes[3], bytes[4], bytes[5]);
}
int main() {
   const char* binary_pa ern = "101010101011101111001100110111011110111111111111";
binaryToMac(binary_pa ern);
   return 0;
}
```

Task 14

1)bare metal blinky using arduino1

```
#define F_CPU 16000000UL
```

```
#include      <avr/io.h>
#include <u l/delay.h>
int main(void)
{
   // Set pin 7 (PD7) as an output
DDRD |= (1 << PD7);
   while (1)
   {
      PORTD |= (1 << PD7);
      _delay_ms(1000);
      PORTD &= ~(1 << PD7);
      _delay_ms(1000);
   }
   return 0;
}
```

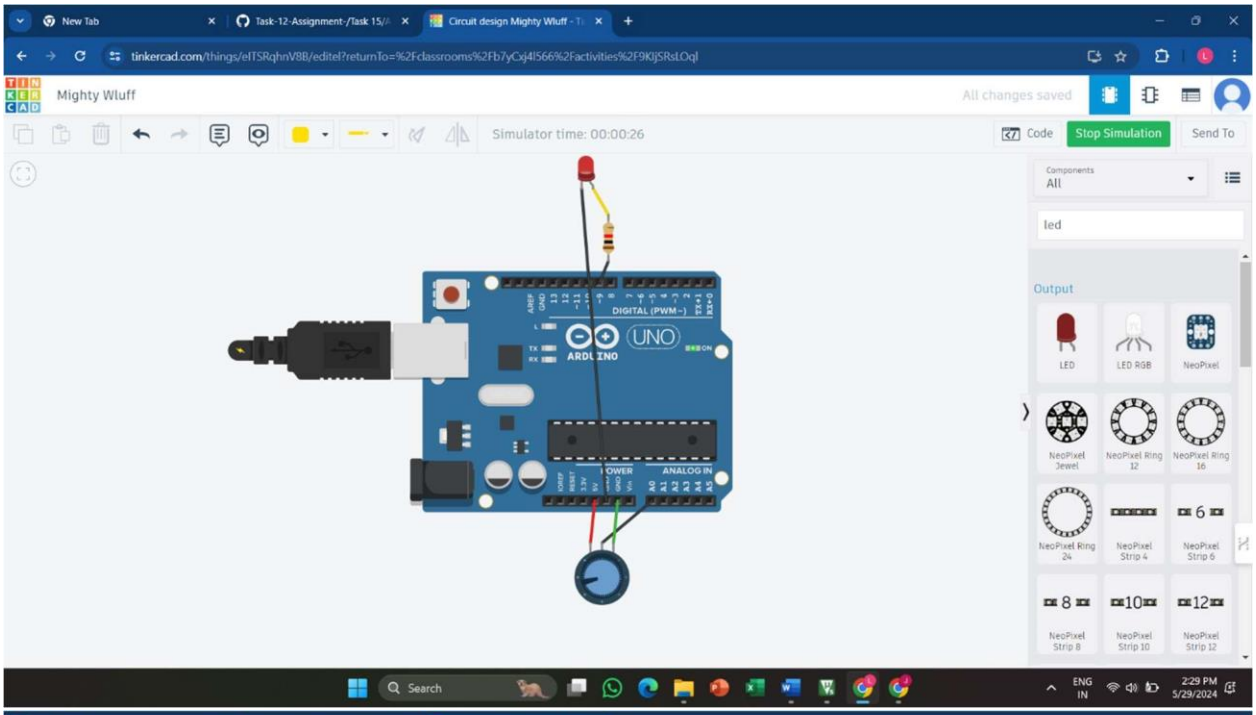
2)bare metal push bu on1

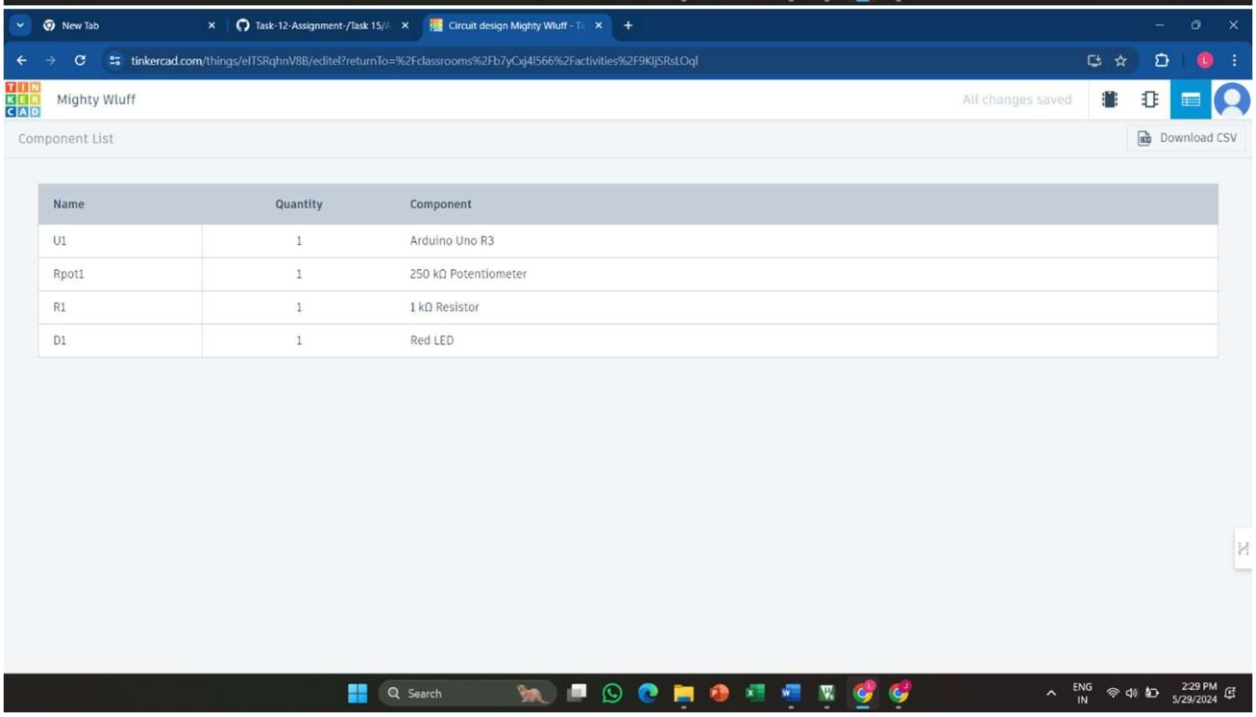
```
#define F_CPU 16000000UL
#include                  <avr/io.h>
#include <u l/delay.h>     const
uint8_t   bu   onPin   =   PD2;
const uint8_t ledPin = PB5;
uint8_t bu onState = 0; void
setup() {
   DDRD &= ~(1 << bu onPin);
   PORTD |= (1 << bu onPin);
   DDRB |= (1 << ledPin);
}
int main(void) {
```

```
   setup();     while (1) {         bu onState =
PIND & (1 << bu onPin);
       if (bu onState) {
          PORTB |= (1 << ledPin);
       } else {
          PORTB &= ~(1 << ledPin);
       }
       _delay_ms(10);
   }
   return 0;
}
```

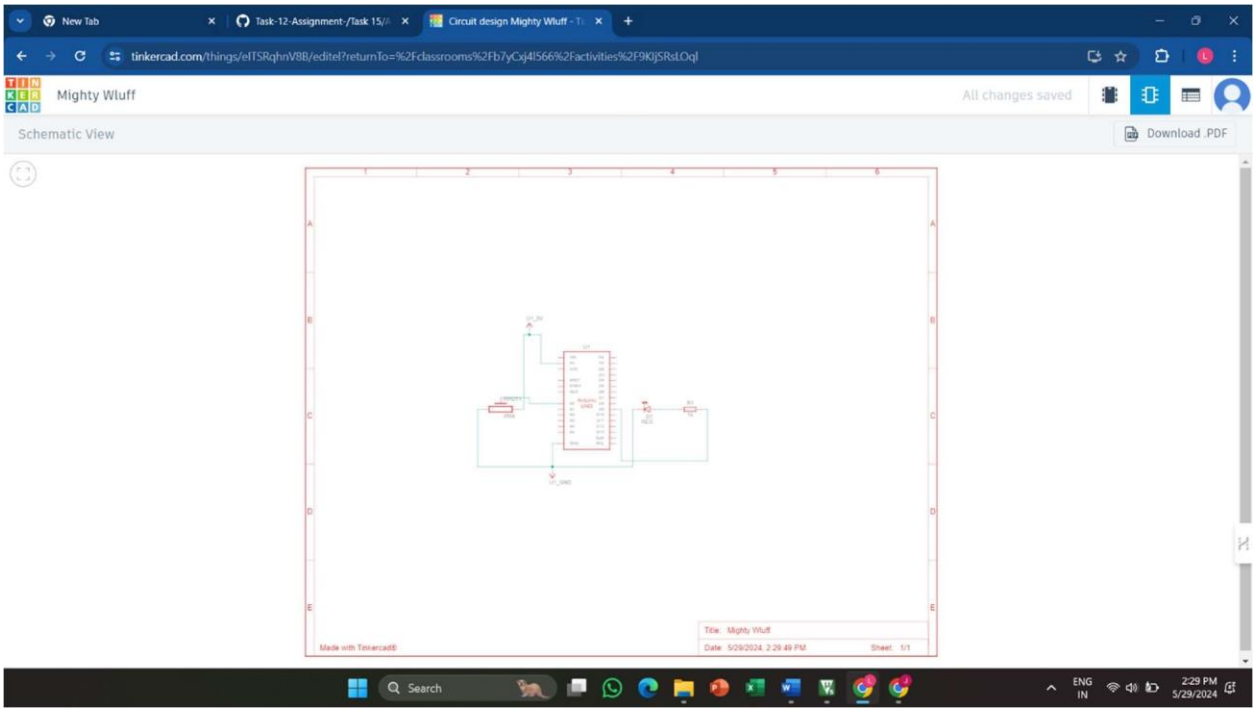
Task 15

Analog Read (Poten ometer)





Name	Quantity	Component
U1	1	Arduino Uno R3
Rpot1	1	250 kΩ Potentiometer
R1	1	1 kΩ Resistor
D1	1	Red LED



// Constants for pin assignments const int potPin = A0; // Pin where the poten ometer is connected const int ledPin = 9; // Pin where the LED is connected

// Variable to store the poten ometer value int potValue = 0;

void setup() {
  // Ini alize the LED pin as an output
pinMode(ledPin, OUTPUT);
}

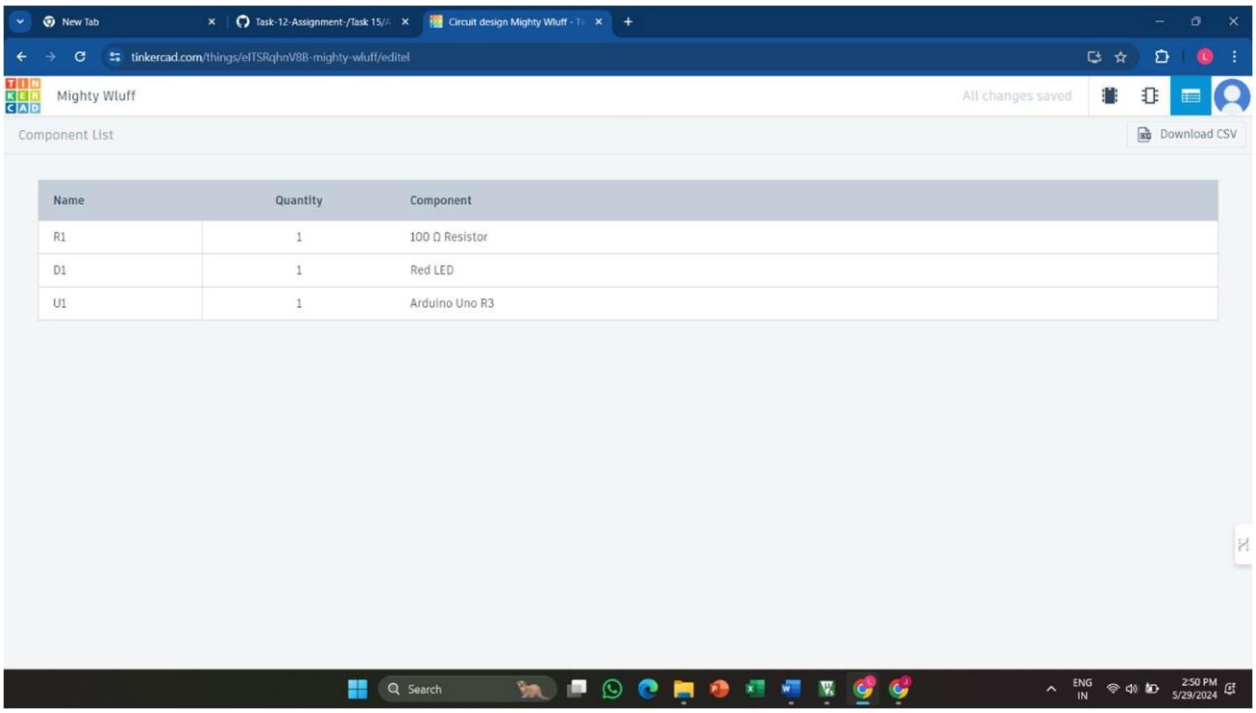
void loop() {
  // Read the value from the poten ometer potValue = analogRead(potPin);

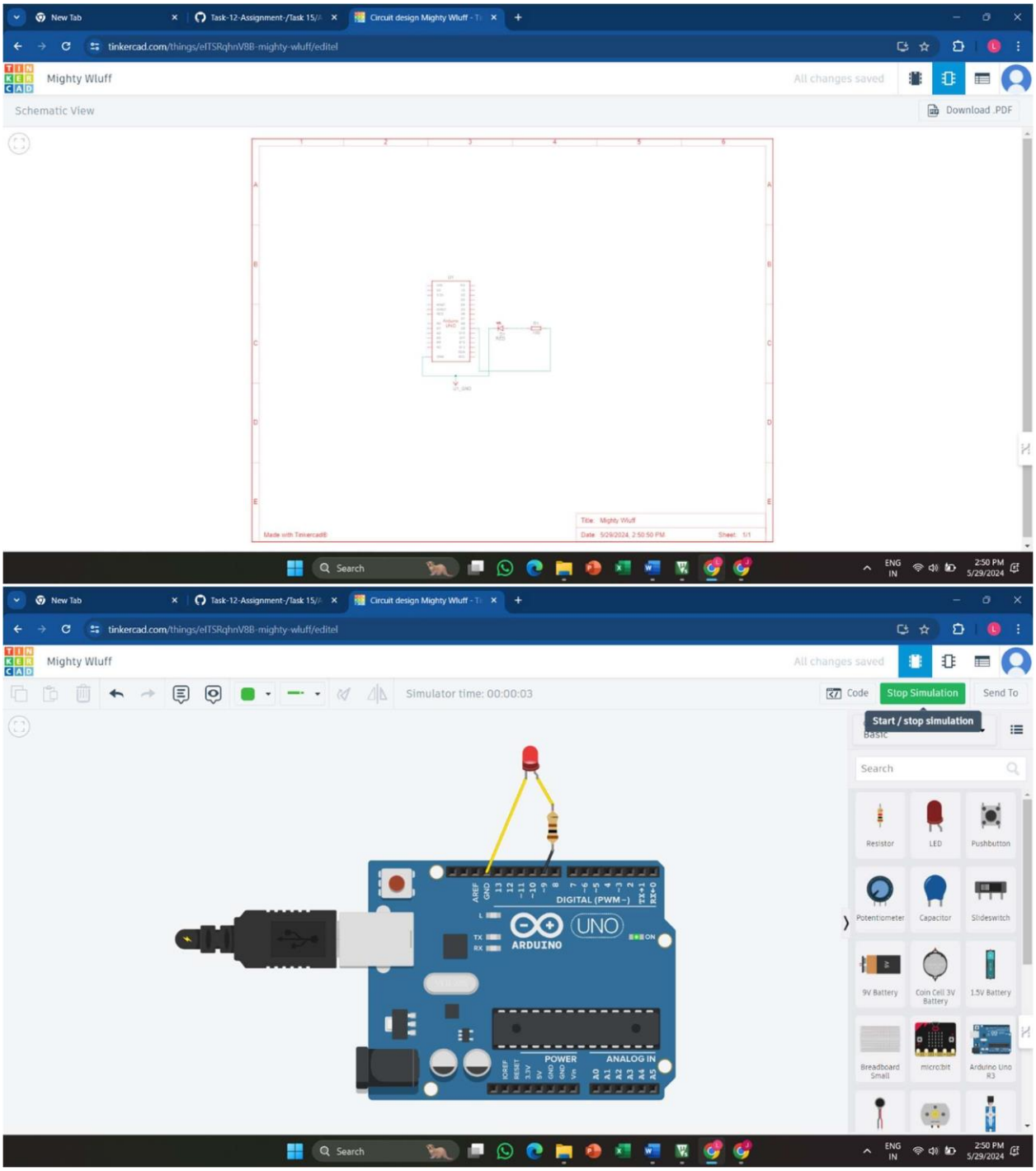
```
  // Map the poten ometer value to the PWM range (0-255)   int
ledValue = map(potValue, 0, 1023, 0, 255);

  // Set the brightness of the LED
analogWrite(ledPin, ledValue);

  // Small delay to smooth out the reading
delay(10);
}
```

Analout Output(fading)

```
const int ledPin = 9;      // Pin where the LED is connected

void setup() {
  // Ini alize the LED pin as an output
pinMode(ledPin, OUTPUT);
}

void loop() {   // Fade in from 0 to 100^6   for (int brightness
= 0; brightness <= 100^6; brightness++) {
analogWrite(ledPin, brightness);  // Set the brightness
delay(10);  // Wait for 10 milliseconds
  }

  // Fade out from 100^6 to 0   for (int brightness = 100^6;
brightness >= 0; brightness--) {     analogWrite(ledPin,
brightness);  // Set the brightness     delay(10);  // Wait for
10 milliseconds

  }
```

}


New Tab
Task-12-Assignment-/Task 15/
Circuit design Mighty Wluff -
tinkercad.com/things/elTSRqhnV8B-mighty-wluff/editel
Mighty Wluff
All changes saved
Component List
Download CSV
Name
Quantity
Component
R1
1
100 Ω Resistor
D1
1
Red LED
U1
1
Arduino Uno R3
Search
ENG
IN
2:50 PM
5/29/2024



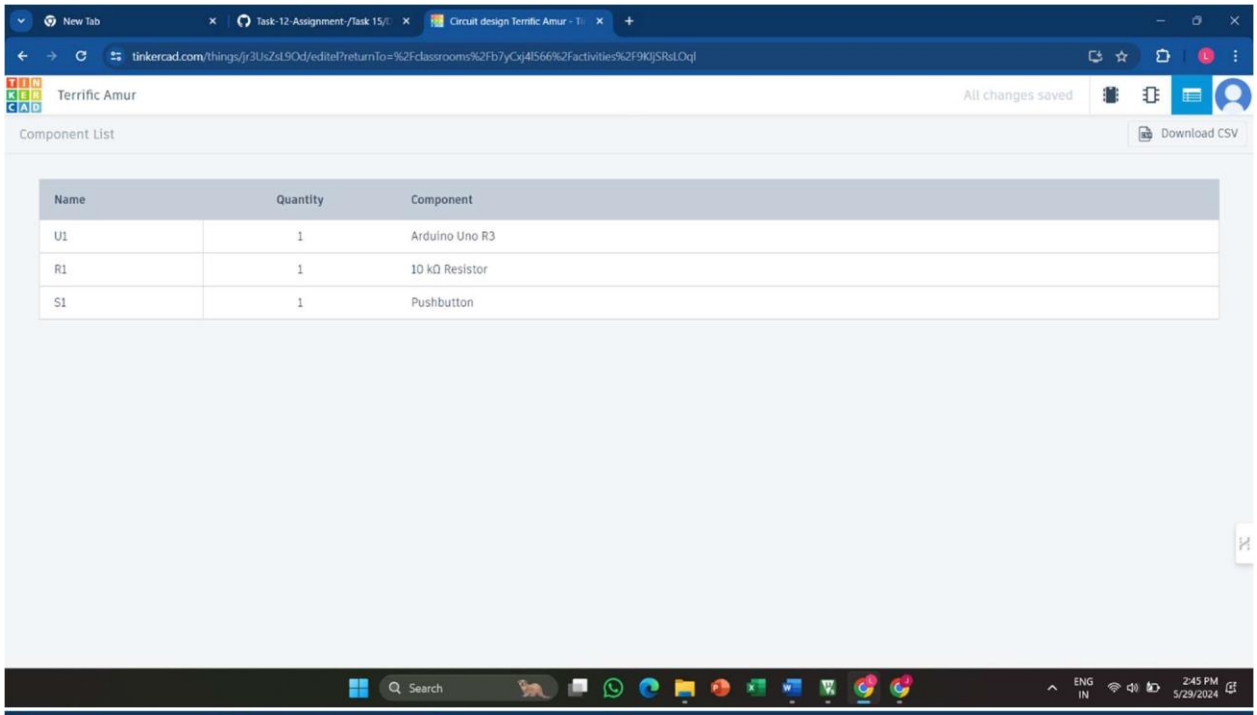
Digital Input using Interrupt

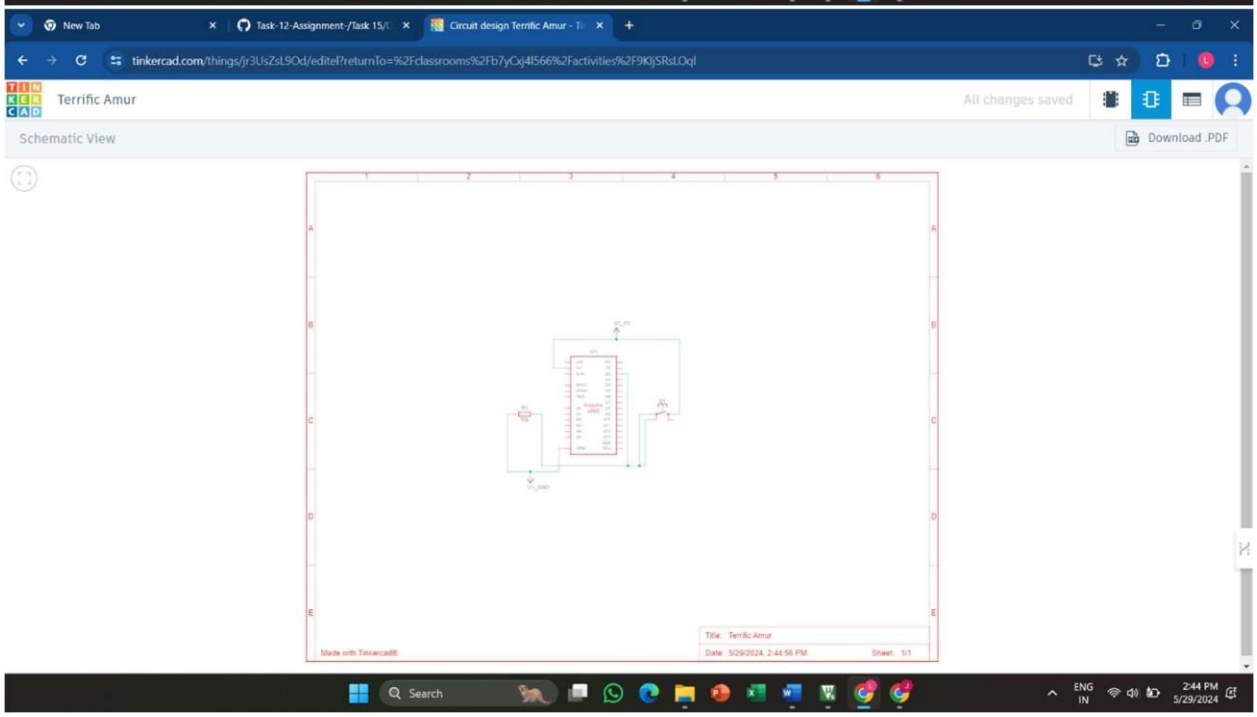
const int bu onPin = 2; // Pin where the push bu on is connected vola le

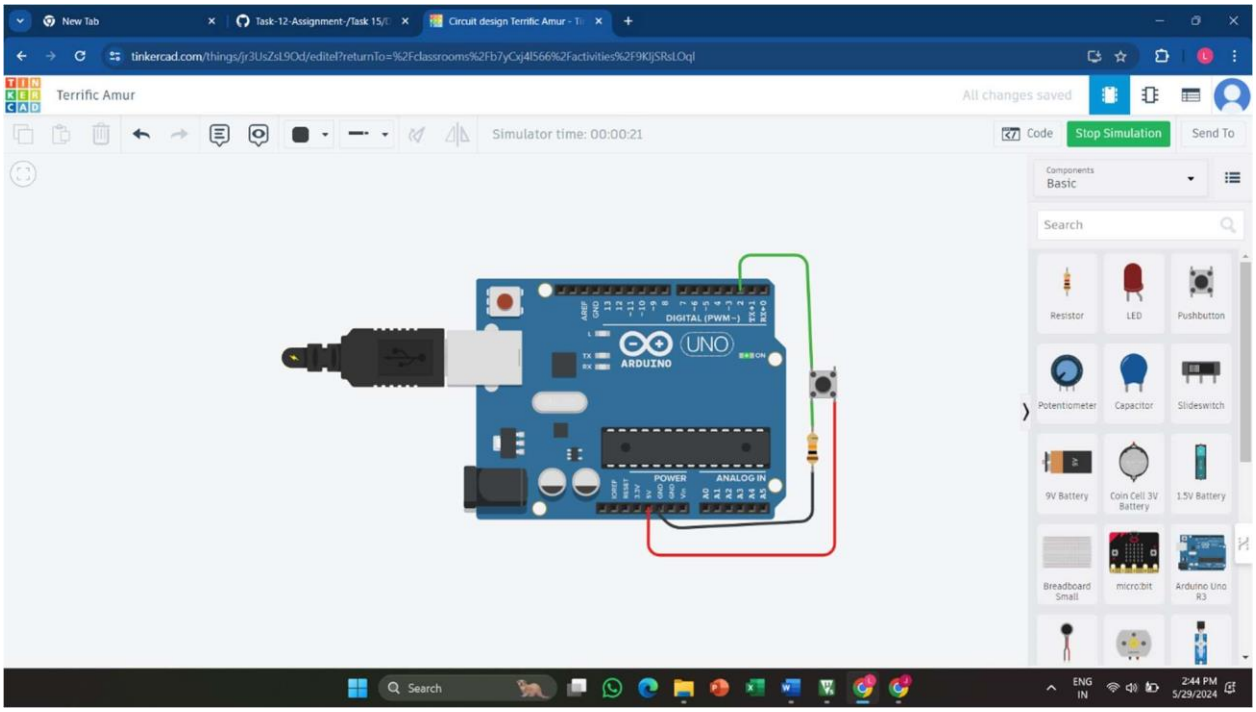
bool bu onPressed = false; // Flag to indicate bu on press void setup() {

pinMode(bu onPin, INPUT); // Set the bu on pin as input

```
  a achInterrupt(digitalPinToInterrupt(bu onPin), bu onPressISR, RISING); // A ach interrupt on rising
edge
  Serial.begin(9600);                 // Ini alize serial communica on
}

void loop() {   if (bu
onPressed) {
    Serial.println("Bu on Pressed!");  // Print message when bu on is pressed
    bu onPressed = false;               // Reset the flag
  }
}

void bu onPressISR() {   bu onPressed = true; // Set the flag to
indicate bu on press
}
```


New Tab
Task-12-Assignment-/Task 15/
Circuit design Terrific Amur - T
tinkercad.com/things/jr3UsZsL9Od/editel?returnTo=%2Fclassrooms%2Fb7yCxj4l566%2Factivities%2F9KljSRsLOql
Terrific Amur
All changes saved
Component List
Download CSV
Name
Quantity
Component
U1
1
Arduino Uno R3
R1
1
10 kΩ Resistor
S1
1
Pushbutton
Search
ENG
IN
2:45 PM
5/29/2024


New Tab
Task-12-Assignment-/Task 15/
Circuit design Terrific Amur - T
tinkercad.com/things/jr3UsZsL9Od/editel?returnTo=%2Fclassrooms%2Fb7yCxj4l566%2Factivities%2F9KljSRsLOql
Terrific Amur
All changes saved
Schematic View
Download .PDF
Made with Tinkercad
Title: Terrific Amur
Date: 5/29/2024, 2:44:56 PM
Sheet: 1/1
Search
ENG
IN
2:44 PM
5/29/2024


New Tab
Task-12-Assignment-/Task 15/
Circuit design Terrific Amur - T
tinkercad.com/things/jr3UsZsL9Od/editel?returnTo=%2Fclassrooms%2Fb7yCxj4l566%2Factivities%2F9KIjSRsLOqJ
Terrific Amur
All changes saved
Simulator time: 00:00:21
Code
Stop Simulation
Send To
Components
Basic
Search
Resistor
LED
Pushbutton
Potentiometer
Capacitor
Slideswitch
9V Battery
Coin Cell 3V Battery
1.5V Battery
Breadboard Small
micro:bit
Arduino Uno R3
DIGITAL (PWM~)
ARDUINO
UNO
POWER
ANALOG IN
Search
ENG
IN
2:44 PM
5/29/2024