



# SUMMER INTERNSHIP Embedded C

-Lakshman B H  
-BU21EECE0100553

## 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 5<sup>th</sup> and 12<sup>th</sup> bits in a 16-bit unsigned integer
- ☐ Write a program to clear 6<sup>th</sup> and 19<sup>th</sup> 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 after setting 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 after 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 after flipping the even-positioned
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);
    printf("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');    }

    printf("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 <util/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)

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

Circuit design: Spectacular Esboo

tinkercad.com/things/K5gU673rP-spectacular-esboo/edit?tenant=circuits

Simulator time: 00:01:03

All changes saved

1 (Arduino Uno R3)

How the debugger works

1. Add breakpoints by clicking on the line numbers.
2. Hover over the variables while paused to see their value.
3. Use the buttons above to resume simulation or step one line at a time.

```

1 // Constants for pin assignments
2 const int potPin = A0; // Pot
3 const int ledPin = 9; // Pin
4 // Variable to store the poter
5 int potValue = 0;
6 void setup() {
7   // Init the LED pin as
8   pinMode(ledPin, OUTPUT);
9 }
10 void loop() {
11   // Read the value from the po
12   potValue = analogRead(potPin);
13
14   // Map the poten0 meter value to the PWM range (0-255)
15   int ledValue = map(potValue, 0, 1023, 0, 255);
16
17   // Set the brightness of the LED
18   analogWrite(ledPin, ledValue);
19
20   // Small delay to smooth out the reading
21   delay(10);
22 }

```

Serial Monitor

Circuit design: Shiny Kasi-Jarv

tinkercad.com/things/fmgm19Rfdw44-shiny-kasi-jarv/edit?tenant=circuits

Shiny Kasi-Jarv

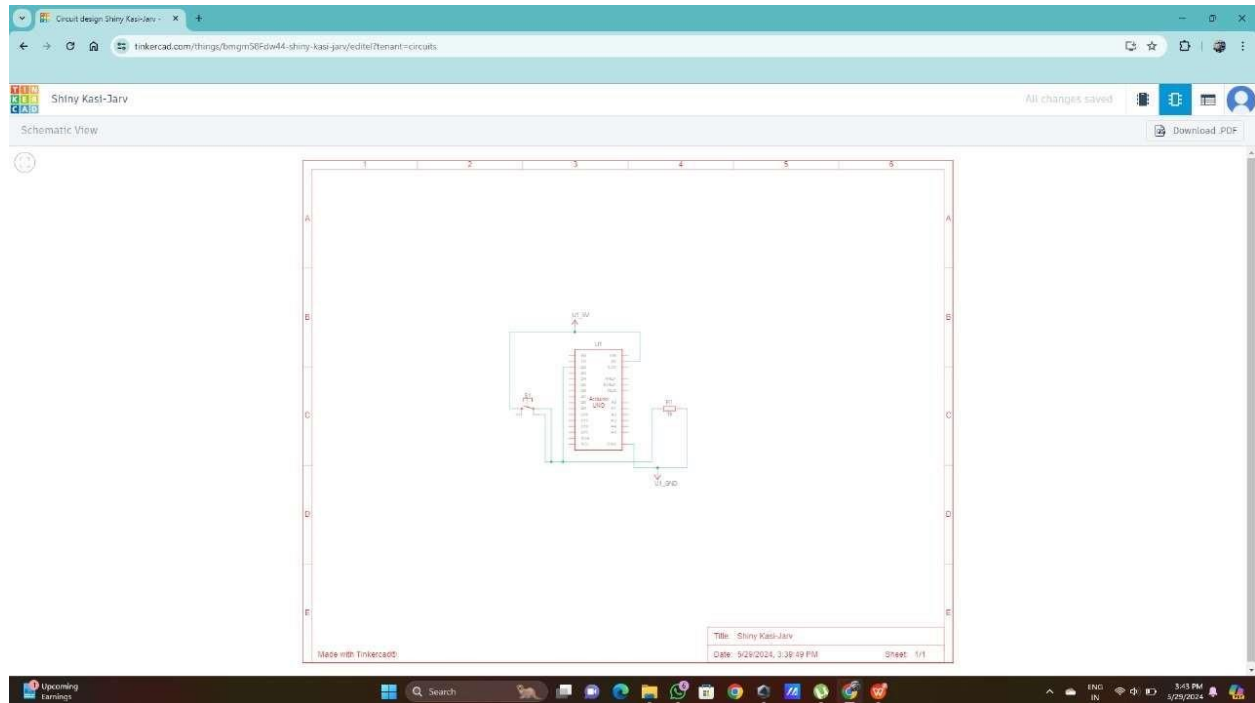
All changes saved

Download CSV

Component List

Name	Quantity	Component
U1:	1	Arduino Uno R3
S1:	1	Pushbutton
R1:	1	1 kΩ Resistor





```
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
}

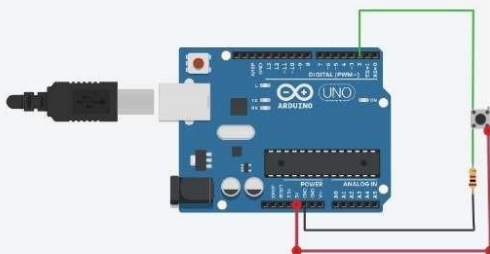
// Fade out from 100^6 to 0  for (int brightness = 100^6;
brightness >= 0; brightness--) {    analogWrite(ledPin,
brightness); // Set the brightness    delay(10); // Wait for
10 milliseconds
}
}
```

Circuit design: Shiny Kasi-Jarv

tinkecad.com/things/8mgnr9Rfdw44-shiny-kasi-jarv/edit#remant=circuits

All changes saved

Code Start Simulation Send To



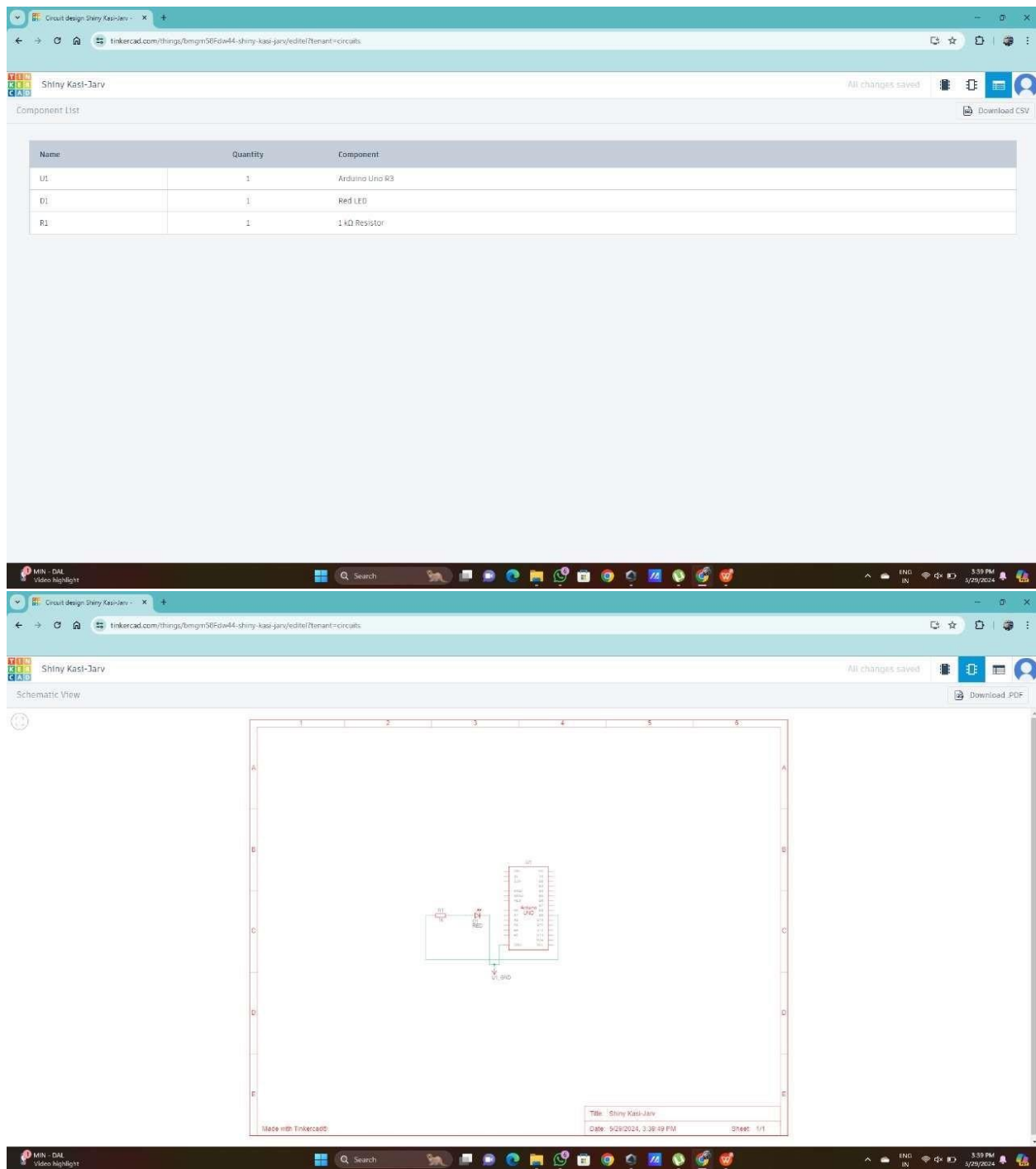
```
1 const int but onPin = 2; // Pin where the push but on is connected
2 void led on but onPressed = false; // Flag to indicate but on got
3 void setup() {
4   pinMode(but onPin, INPUT); // Set the but on pin as input
5   attachInterrupt(digitalPinToInterrupt(but onPin), but onPressISR,
6   edge
7   Serial.begin(9600); // Initilize serial communication on
8 }
9 void loop() {
10  if (but onPressed) {
11    Serial.println("But on Pressed!"); // Print message when but on is
12    but onPressed = false; // Reset the flag
13  }
14 }
15 void but onPressISR() {
16   but onPressed = true; // Set the flag to indicate but on press
17 }
```

Serial Monitor

Upcoming Earnings

Search

3:45 PM 3/25/2024



## 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`

```

    attachInterrupt(digitalPinToInterrupt(bu onPin), bu onPressISR, RISING); // Attach interrupt on rising edge
    Serial.begin(9600);           // Initialize serial communication
}

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
}

```

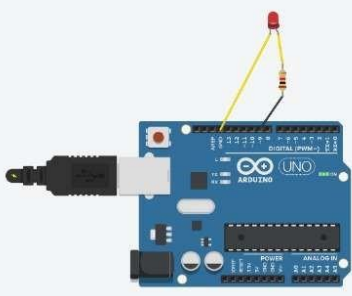
Circuit design Shiny Kasi-Jarv

tinkecad.com/things/fmgm9f6d44-shiny-kasi-jarv/edit#t=components

All changes saved

Simulator time: 00:00:04

1 (Arduino Uno R3)



```
1 const int ledPin = 9; // Pin where the LED is connected
2 void setup() {
3   // This allows the LED pin as an output
4   pinMode(ledPin, OUTPUT);
5 }
6 void loop() {
7   // Fade in from 0 to 100%
8   for (int brightness = 0; brightness <= 100; brightness++) {
9     analogWrite(ledPin, brightness); // Set the brightness
10    delay(10); // Wait for 10 milliseconds
11  }
12  // Fade out from 100% to 0
13  for (int brightness = 100; brightness >= 0; brightness--) {
14    analogWrite(ledPin, brightness); // Set the brightness
15    delay(10); // Wait for 10 milliseconds
16  }
17 }
18 }
```

Circuit design Spectacular Esboo

tinkecad.com/things/K5gU67E3P-spectacular-esboo/edit#t=components

All changes saved

Download CSV

Component List

Name	Quantity	Component
U1:	1	Arduino Uno R3
R1:	1	1kΩ Resistor
D1:	1	Red LED

Circuit design Spectacular Esboo

tinkercad.com/things/K5gjU67E3rP-spectacular-esboo/edit?tenant=circuits

Spectacular Esboo

All changes saved

Download PDF

Schematic View

5V  
10k  
LED  
7805  
GND

1 2 3 4 5  
A B C D E

Made with Tinkercad

Title:	Spectacular Esboo
Date:	5/29/2024, 3:32:39 PM
Sheet:	1/1

Search

ENG IN

3:33 PM 5/29/2024