

# **MIKROPROSESOR & MIKROKONTROLLER**

## **UTS**



Mata Kuliah : Sistem Tertanam

Dosen : Nurseno Bayu Aji, S.Kom.,M.Kom.

Disusun oleh

Nama : Ilham Indra Atmaja

NIM : 4.33.23.2.16

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**JURUSAN TEKNIK ELEKTRO**  
**POLITEKNIK NEGERI SEMARANG**

- **Code**

**Kode Untuk C**

```
const int led1 = A2;
const int led2 = 3;
const int led3 = 8;
const int buttonPin = 7;

unsigned long prevMillis2 = 0, prevMillis3 = 0;
const long interval2 = 200, interval3 = 1000;

bool buttonPressed = false;

void setup() {
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(buttonPin, INPUT_PULLUP);
}

void loop() {
  if (digitalRead(buttonPin) == LOW) {
    delay(50); // Debounce
    buttonPressed = !buttonPressed;
    digitalWrite(led1, buttonPressed ? HIGH : LOW);
    if (!buttonPressed) { // Jika tombol dilepas, matikan semua LED
      digitalWrite(led2, LOW);
      digitalWrite(led3, LOW);
    }
    while (digitalRead(buttonPin) == LOW); // Tunggu tombol dilepas
  }

  if (buttonPressed) {
    unsigned long currMillis = millis();
    if (currMillis - prevMillis2 >= interval2) {
      prevMillis2 = currMillis;
      digitalWrite(led2, !digitalRead(led2)); // Toggle LED2
    }
    if (currMillis - prevMillis3 >= interval3) {
      prevMillis3 = currMillis;
      digitalWrite(led3, !digitalRead(led3)); // Toggle LED3
    }
  }
}
```

## Kode Untuk Assembly

```
1  .global _start
2
3  .section .text
4  _start:
5      // Set LED1 (A2/PORTC.2), LED2 (PD3/PORTD.3), LED3 (PB0/PORTB.0) sebagai OUTPUT
6      ldi r16, 0b00000100
7      out DDRC, r16          // LED1
8      ldi r16, 0b00001000
9      out DDRD, r16          // LED2
10     ldi r16, 0b00000001
11     out DDRB, r16          // LED3
12
13     // Set Button (PD7) sebagai INPUT_PULLUP
14     ldi r16, 0b01111111
15     out DDRD, r16          // Button
16     ldi r16, 0b10000000
17     out PORTD, r16         // Pull-up Button
18
19     main_loop:
20         in r17, PIND        // Baca tombol
21         sbrc r17, 7         // Jika tombol ditekan
22         rjmp button_pressed
23         rjmp main_loop     // Ulangi
24
25     button_pressed:
26         ldi r16, 0b00000100
27         out PORTC, r16      // Nyalakan LED1
28         call blink_led2    // Kedip LED2 cepat
29         call blink_led3    // Kedip LED3 lambat
30         rjmp main_loop     // Kembali ke loop
31
32     blink_led2:
33         ldi r18, 200
34         call delay          // Delay 200ms
35         in r17, PORTD
36         eor r17, 0b00001000 // Toggle LED2
37         out PORTD, r17
38         ret
39
```

```
32     blink_led2:
33         ldi r18, 200
34         call delay          // Delay 200ms
35         in r17, PORTD
36         eor r17, 0b00001000 // Toggle LED2
37         out PORTD, r17
38         ret
39
40     blink_led3:
41         ldi r18, 1000
42         call delay_long     // Delay 1000ms
43         in r17, PORTB
44         eor r17, 0b00000001 // Toggle LED3
45         out PORTB, r17
46         ret
47
48     delay:
49         push r18
50     delay_loop:
51         dec r18
52         brne delay_loop
53         pop r18
54         ret
55
56     delay_long:
57         push r24
58         push r25
59     long_delay_loop:
60         dec r25
61         brne long_delay_loop
62         dec r24
63         brne long_delay_loop
64         pop r25
65         pop r24
66         ret
67
```

**.global \_start**

**.section .text**

**\_start:**

**// Set LED1 (A2/PORTC.2), LED2 (PD3/PORTD.3), LED3 (PB0/PORTB.0)**

**sebagai OUTPUT**

**ldi r16, 0b00000100**

**out DDRC, r16        // LED1**

**ldi r16, 0b00001000**

**out DDRD, r16        // LED2**

**ldi r16, 0b00000001**

**out DDRB, r16        // LED3**

**// Set Button (PD7) sebagai INPUT\_PULLUP**

**ldi r16, 0b01111111**

**out DDRD, r16        // Button**

**ldi r16, 0b10000000**

**out PORTD, r16        // Pull-up Button**

**main\_loop:**

**in r17, PIND        // Baca tombol**

**sbrs r17, 7        // Jika tombol ditekan**

**rjmp button\_pressed**

**rjmp main\_loop        // Ulangi**

**button\_pressed:**

**ldi r16, 0b00000100**

**out PORTC, r16        // Nyalakan LED1**

**call blink\_led2        // Kedip LED2 cepat**

**call blink\_led3        // Kedip LED3 lambat**

**rjmp main\_loop        // Kembali ke loop**

**blink\_led2:**

**ldi r18, 200**

```
call delay          // Delay 200ms
in r17, PORTD
eor r17, 0b00001000 // Toggle LED2
out PORTD, r17
ret
```

**blink\_led3:**

```
ldi r18, 1000
call delay_long      // Delay 1000ms
in r17, PORTB
eor r17, 0b00000001 // Toggle LED3
out PORTB, r17
ret
```

**delay:**

```
push r18
```

**delay\_loop:**

```
dec r18
brne delay_loop
pop r18
ret
```

**delay\_long:**

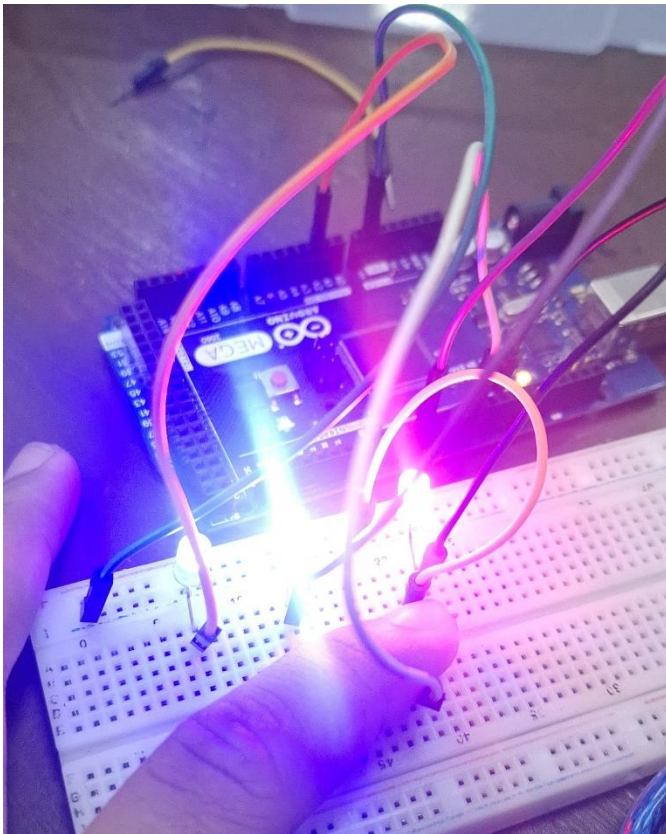
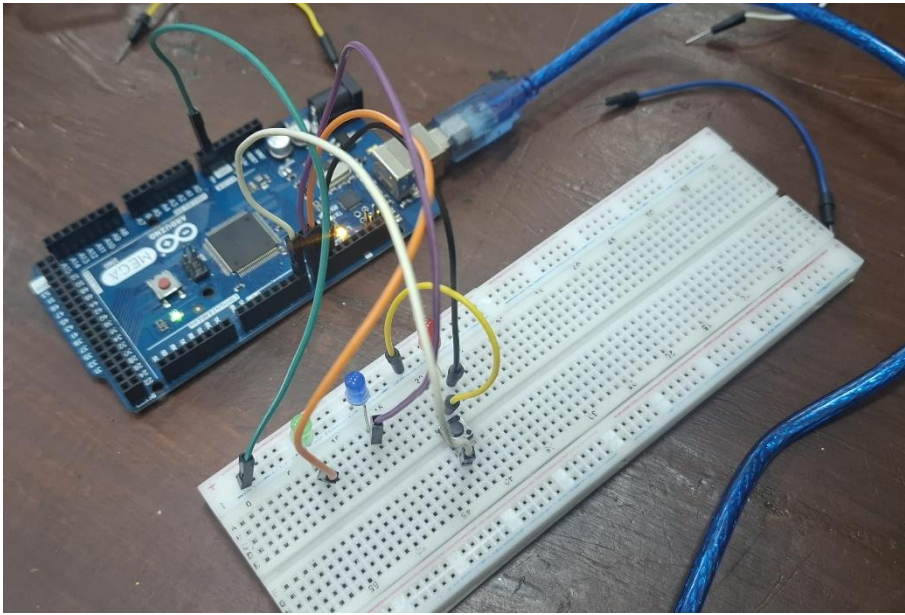
```
push r24
```

```
push r25
```

**long\_delay\_loop:**

```
dec r25
brne long_delay_loop
dec r24
brne long_delay_loop
pop r25
pop r24
ret
```

- **Rangkaian**



- **Kendala**

Kesulitan dalam pengerjaan uts adalah membuat kode assembly yang sesuai dan juga kode C untuk dapat Aduino IDE membaca kode assembly