

SPRAWOZDANIE					PROSZĘ PODAĆ NR GRUPY:						
				ZIISS1	3	5	1	2	IO		
IMIĘ	NAZWISKO	Temat ćwiczenia zgodny z wykazem tematów:			PONIŻEJ PROSZĘ PODAĆ TERMIN ZAJĘĆ:			ROK:			
MICHAŁ	WARSZAWSKI	Podstawy programowania Układy wejścia-wyjścia						2023 r.			
					PN	WT	SR	CZ	PT	SB	ND
					GODZINA ROZPOCZĘCIA ZAJĘĆ:						11 : 30

Wprowadzenie teoretyczne:

### Opisz rodzaje portów w mikrokontrolerze ATmega2560

Mikrokontroler ATmega2560, produkowany przez firmę Microchip, jest wyposażony w różne rodzaje portów, które pełnią różne funkcje.

- Porty ogólne (General Purpose I/O Ports) obejmują PORTA, PORTB, PORTC, PORTD, PORTE, PORTF, PORTG, PORTH, PORTJ i PORTK. Każdy z tych portów składa się z 8 pinów, które mogą być skonfigurowane jako wejścia lub wyjścia cyfrowe.
- Porty wejścia/wyjścia (I/O Ports) pozwalają na indywidualną konfigurację pinów jako wejścia lub wyjścia cyfrowe.
- Porty analogowe, takie jak A0, A1, ..., A15, umożliwiają korzystanie z przetworników analogowo-cyfrowych (ADC) do pomiaru napięć.
- Porty szeregowo (Serial Ports) obejmują interfejsy UART, SPI i I2C do komunikacji z innymi urządzeniami.
- Porty PWM (Pulse Width Modulation) umożliwiają generowanie sygnałów PWM do sterowania prędkością silników, jasnością diod LED, itp.
- Porty zegarowe, takie jak XTAL1 i XTAL2, służą do podłączania rezonatora kwarcowego lub oscylatora zegarowego.
- Porty wejścia przerwania (Interrupt Ports) pozwalają na natychmiastowe przerwanie programu w odpowiedzi na zmiany stanu na danym pinie.

## Zadanie 1

### Zapalenie wszystkich świateł w celach testowych

- Należy zapalić wszystkie światła zarówno dla pieszych jak i dla samochodów

### Włączenie światła zielonego dla samochodów i czerwonego dla pieszych

- Należy wyłączyć procedurę testową i załączyć światło zielone dla samochodów a czerwone dla pieszych
- Po ustalonym czasie należy zmienić zapalić światło żółte, następnie czerwone dla pojazdów i zielone dla pieszych.
- Po ustalonym czasie należy spowodować miganie światła zielonego a następnie zapalić światło czerwone dla pieszych i zapalić światła żółte i czerwone dla pojazdów a następnie światło zielone dla pojazdów.
- Po dwóch takich sekwencjach należy przejść w tryb „nocny” czyli wygasić wszystkie światła i włączyć pulsujące światła żółte dla pojazdów

### Listing programu:

```
#define people_red A4
#define people_green A0
#define cars_red A15
#define cars_yellow A12
#define cars_green A9

void setup() {
    pinMode(cars_red, OUTPUT);
    pinMode(cars_yellow, OUTPUT);
    pinMode(cars_green, OUTPUT);
    pinMode(people_red, OUTPUT);
    pinMode(people_green, OUTPUT);
    // TEST
    // zapalenie wszystkich świateł
    digitalWrite(cars_red, HIGH);
    digitalWrite(cars_yellow, HIGH);
    digitalWrite(cars_green, HIGH);
    digitalWrite(people_red, HIGH);
    digitalWrite(people_green, HIGH);
    delay(1000);
    // zgaszenie wszystkich świateł
    digitalWrite(cars_red, LOW);
    digitalWrite(cars_yellow, LOW);
    digitalWrite(cars_green, LOW);
    digitalWrite(people_red, LOW);
    digitalWrite(people_green, LOW);
    delay(1000);
}

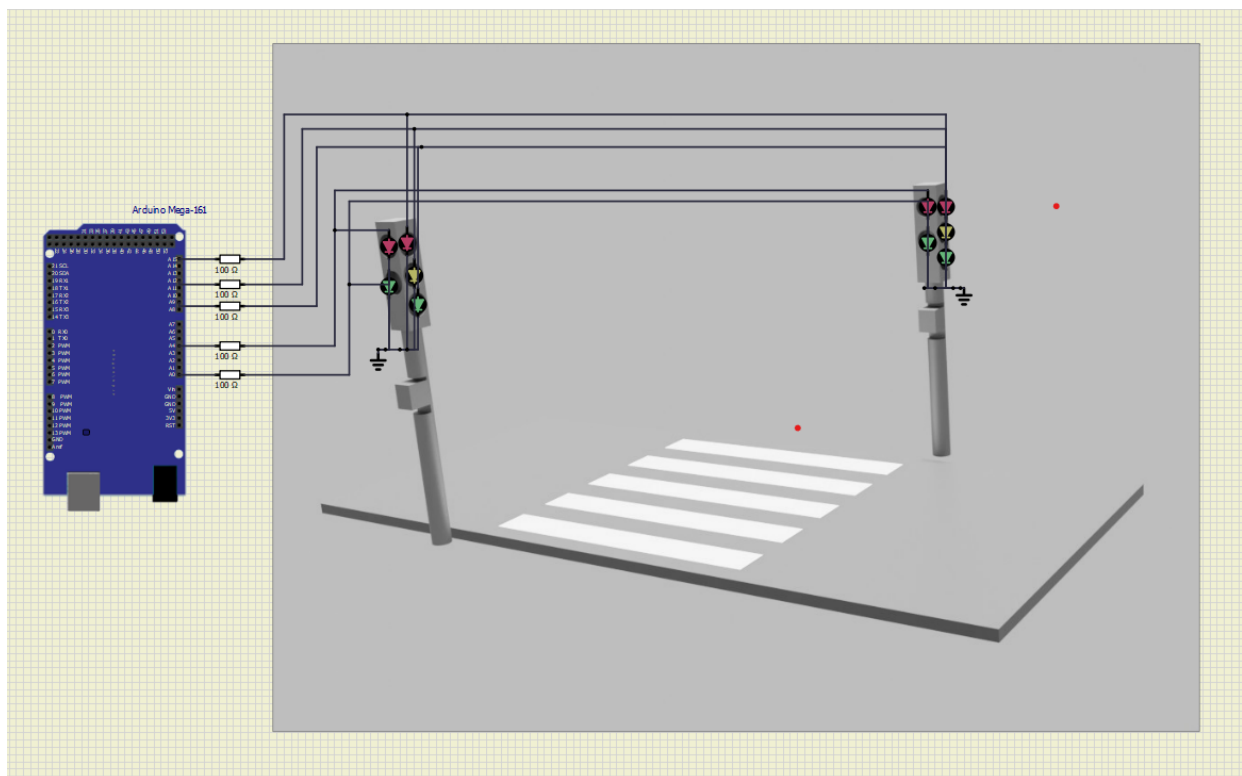
int Loops = 0;
void loop(){
    Loops++;
    if(Loops<=2){
        // TRYB DZIENNY
        digitalWrite(cars_green, HIGH);
        digitalWrite(people_red, HIGH);
        delay(10000);
        digitalWrite(cars_green, LOW);
        digitalWrite(cars_yellow, HIGH);
        delay(2000);
        digitalWrite(cars_yellow, LOW);
        digitalWrite(cars_red, HIGH);
        delay(4000);
        digitalWrite(people_red, LOW);
        digitalWrite(people_green, HIGH);
```

```

    delay(5000);
    for(int i=0; i<5; i++){
        digitalWrite(people_green, HIGH);
        delay(500);
        digitalWrite(people_green, LOW);
        delay(500);
    }
    digitalWrite(people_red, HIGH);
    delay(2000);
    digitalWrite(cars_red, LOW);
    digitalWrite(cars_yellow, HIGH);
    delay(1000);
    digitalWrite(cars_yellow, LOW);
    digitalWrite(people_red, LOW);
}
else{
    // TRYB NOCNY
    digitalWrite(cars_yellow, HIGH);
    delay(500);
    digitalWrite(cars_yellow, LOW);
    delay(500);
}
}
}

```

Zrzut ekranu ze schematem:



### Wnioski:

W kodzie definiowane są stałe dla konkretnych pinów, ułatwiając tym samym czytelność. Następnie w funkcji setup() ustawiane są piny jako wyjścia dla świateł samochodów i pieszych. Sekcja testowa sprawdza poprawność działania świateł. W pętli głównej programu loop(), jest zaimplementowana sekwencja świetlna dla trybu dziennego i nocnego. W trybie dziennym symulowana jest typowa sekwencję świateł na skrzyżowaniu, a także efekt migotania świateł dla pieszych. Po dwóch obiegach pętli, program przechodzi w tryb nocny, gdzie jedynie migają światła żółte samochodów.

## Zadanie 2

### Zapalenie wszystkich świateł w celach testowych

- Należy zapalić wszystkie światła zarówno dla pieszych jak i dla samochodów

### Włączenie światła zielonego dla samochodów i czerwonego dla pieszych

- Należy wyłączyć procedurę testową i załączyć światło zielone dla samochodów a czerwone dla pieszych
- Po ustalonym czasie należy zmienić zapalić światło żółte, następnie czerwone dla pojazdów i zielone dla pieszych.
- Po ustalonym czasie należy spowodować miganie światła zielonego a następnie zapalić światło czerwone dla pieszych i zapalić światła żółte i czerwone dla pojazdów a następnie światło zielone dla pojazdów.
- Po przyciśnięciu przycisku należy przejść w tryb nocny i utrzymywać go tak długo aż przycisk zostanie zwolniony. Tryb „nocny” czyli wygaszone wszystkie światła i włączone pulsujące światła żółte dla pojazdów.

### Listing programu:

```
#define people_red A4
#define people_green A0
#define cars_red A15
#define cars_yellow A12
#define cars_green A9

void setup(){
    pinMode(cars_red, OUTPUT);
    pinMode(cars_yellow, OUTPUT);
    pinMode(cars_green, OUTPUT);
    pinMode(people_red, OUTPUT);
    pinMode(people_green, OUTPUT);
    // TEST
    turnOnOffAll(HIGH);
    delay(1000);
    turnOnOffAll(LOW);
    delay(1000);
    // PRZERWANIE
    pinMode(13, INPUT_PULLUP);
    attachInterrupt(digitalPinToInterrupt(13), nightMode, FALLING);
    interrupts();
}

void loop(){
    if(digitalRead(13)==1){
        // TRYB DZIENNY
        digitalWrite(cars_green, HIGH);
        digitalWrite(people_red, HIGH);
        delay(10000);
        digitalWrite(cars_green, LOW);
        digitalWrite(cars_yellow, HIGH);
        delay(2000);
        digitalWrite(cars_yellow, LOW);
        digitalWrite(cars_red, HIGH);
        delay(4000);
        digitalWrite(people_red, LOW);
        digitalWrite(people_green, HIGH);
        delay(5000);
        for(int i=0; i<5; i++){
            digitalWrite(A0, HIGH);
            delay(500);
            digitalWrite(A0, LOW);
            delay(500);
        }
    }
}
```

```

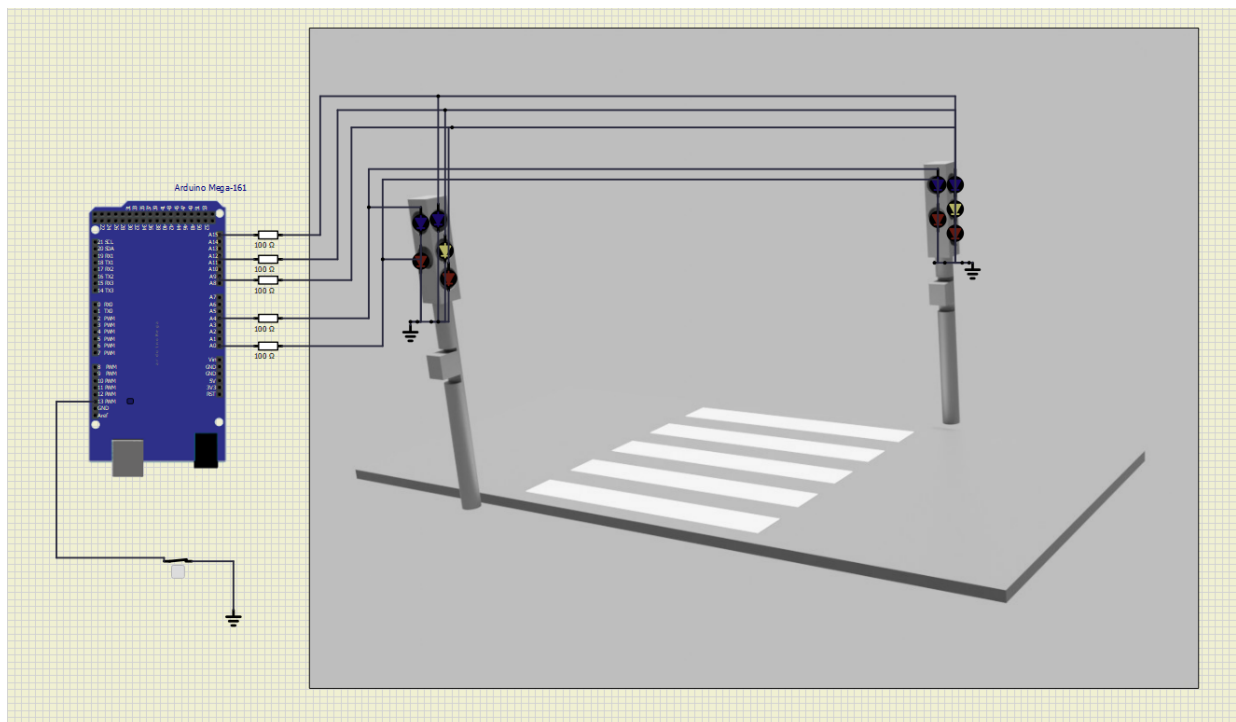
        digitalWrite(people_red, HIGH);
        delay(2000);
        digitalWrite(cars_red, LOW);
        digitalWrite(cars_yellow, HIGH);
        delay(1000);
        digitalWrite(cars_yellow, LOW);
        digitalWrite(people_red, LOW);
    }
    else{
        // TRYB NOCNY
        digitalWrite(cars_yellow, HIGH);
        delay(500);
        digitalWrite(cars_yellow, LOW);
        delay(500);
    }
}

void nightMode(){
    return 0;
}

void turnOnOffAll(int state){
    digitalWrite(cars_red, state);
    digitalWrite(cars_yellow, state);
    digitalWrite(cars_green, state);
    digitalWrite(people_red, state);
    digitalWrite(people_green, state);
}

```

#### Zrzut ekranu ze schematem:



#### Wnioski:

W przedstawionym kodzie mikrokontrolera dodatkowo skonfigurowano przerwanie na pinie 13, aby obsłużyć tryb nocny. W pętli `loop()` sprawdzany jest stan przycisku (pin 13). Sekwencja dla trybu dziennego jest wykonywana dopóki nie zostanie włączony przycisk. Wtedy aktywowana jest sekwencja nocna i trwa dopóki przycisk nie zostanie wyłączony.

### Zadanie 3

#### Zapalenie wszystkich świateł w celach testowych

- Należy zapalić wszystkie światła zarówno dla pieszych jak i dla samochodów

#### Włączenie światła zielonego dla samochodów i czerwonego dla pieszych

- Należy wyłączyć procedurę testową i załączyć światło zielone dla samochodów a czerwone dla pieszych
- Po przyciśnięciu przycisku znajdującego się na dowolnym z sygnalizatorów należy zapalić światło żółte, następnie czerwone dla pojazdów i zielone dla pieszych.
- Po ustalonym czasie należy spowodować miganie światła zielonego a następnie zapalić światło czerwone dla pieszych i zapalić światła żółte i czerwone dla pojazdów a następnie światło zielone dla pojazdów.
- Po przyciśnięciu przycisku należy przejść w tryb nocny i utrzymywać go tak długo aż przycisk zostanie zwolniony. Tryb „nocny” czyli wygaszone wszystkie światła i włączone pulsujące światła żółte dla pojazdów.

**UWAGA ! Jeżeli przycisk wywołania światła zielonego zostanie naciśnięty kolejny raz przed upływem 30 sekund należy opóźnić wywołanie włączenia światła zielonych dla pieszych o 5 sekund.**

#### Listing programu:

```
#define PEOPLE_RED A4
#define PEOPLE_GREEN A0
#define CARS_RED A15
#define CARS_YELLOW A12
#define CARS_GREEN A9
#define NIGHT_ON_PIN 13
#define BUTTON_PIN 1
#define LED_PIN 6

void setup() {
    pinMode(CARS_RED, OUTPUT);
    pinMode(CARS_YELLOW, OUTPUT);
    pinMode(CARS_GREEN, OUTPUT);
    pinMode(PEOPLE_RED, OUTPUT);
    pinMode(PEOPLE_GREEN, OUTPUT);
    pinMode(LED_PIN, OUTPUT);

    // TEST
    turnOnOffAll(HIGH);
    delay(1000);
    turnOnOffAll(LOW);
    delay(1000);

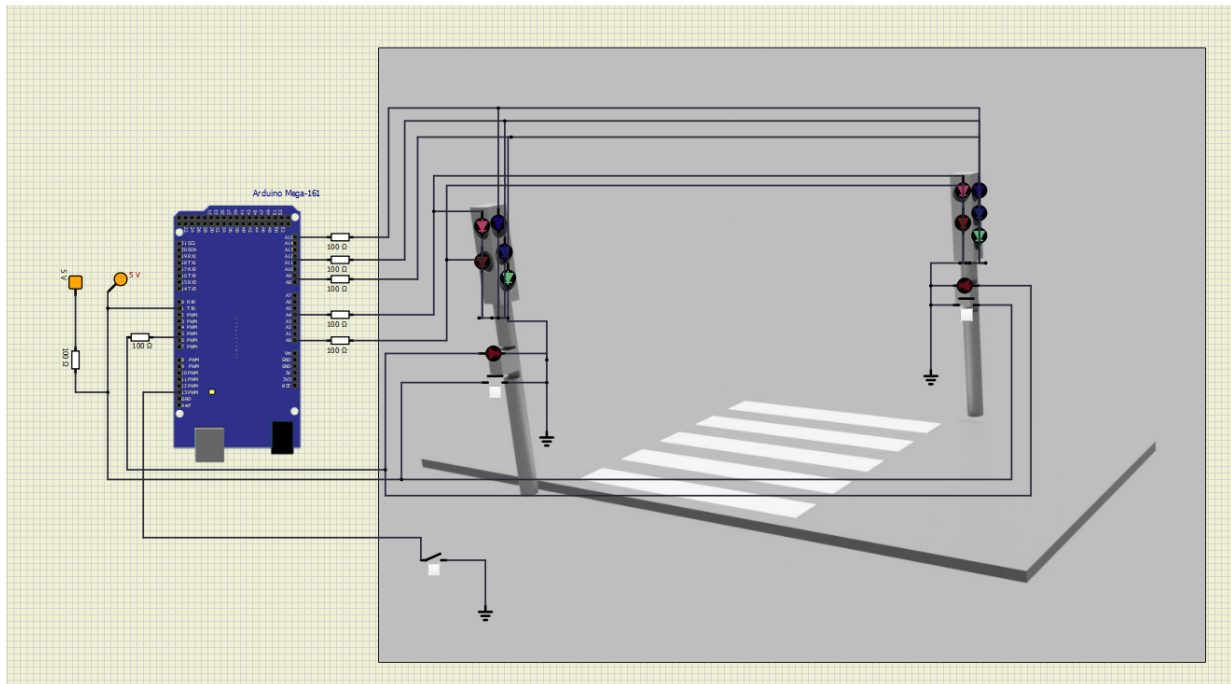
    // PRZERWANIA
    pinMode(NIGHT_ON_PIN, INPUT_PULLUP);
    attachInterrupt(digitalPinToInterrupt(NIGHT_ON_PIN), przerwanie, FALLING);
    pinMode(BUTTON_PIN, INPUT);
    attachInterrupt(digitalPinToInterrupt(BUTTON_PIN), przerwanie, FALLING);
    interrupts();
}

void loop() {
    // jeżeli tryb nocny ON
    if(digitalRead(NIGHT_ON_PIN) == LOW){
        digitalWrite(CARS_GREEN, LOW);
        digitalWrite(PEOPLE_RED, LOW);
        digitalWrite(CARS_YELLOW, HIGH);
        delay(500);
        digitalWrite(CARS_YELLOW, LOW);
    }
}
```

```
        delay(500);
    }
    // jeżeli tryb nocny OFF
    else{
        // kliknięto przycisk pieszych
        if((digitalRead(BUTTON_PIN) == LOW)){
            digitalWrite(LED_PIN, HIGH);
            delay(2000);
            digitalWrite(CARS_GREEN, LOW);
            digitalWrite(CARS_YELLOW, HIGH);
            delay(2000);
            digitalWrite(CARS_YELLOW, LOW);
            digitalWrite(CARS_RED, HIGH);
            delay(2000);
            digitalWrite(PEOPLE_RED, LOW);
            digitalWrite(PEOPLE_GREEN, HIGH);
            digitalWrite(LED_PIN, LOW);
            delay(5000);
            for(int i=0; i<5; i++){
                digitalWrite(PEOPLE_GREEN, HIGH);
                delay(500);
                digitalWrite(PEOPLE_GREEN, LOW);
                delay(500);
            }
            digitalWrite(PEOPLE_RED, HIGH);
            delay(2000);
            digitalWrite(CARS_RED, LOW);
            digitalWrite(CARS_YELLOW, HIGH);
            delay(1000);
            digitalWrite(CARS_YELLOW, LOW);
        }
        // przycisk nie aktywny
        else{
            digitalWrite(CARS_YELLOW, LOW);
            digitalWrite(CARS_GREEN, HIGH);
            digitalWrite(PEOPLE_RED, HIGH);
        }
    }
}

void przerwanie() {
    return;
}

void turnOnOffALL(int state) {
    digitalWrite(CARS_RED, state);
    digitalWrite(CARS_YELLOW, state);
    digitalWrite(CARS_GREEN, state);
    digitalWrite(PEOPLE_RED, state);
    digitalWrite(PEOPLE_GREEN, state);
}
```

**Zrzut ekranu ze schematem:****Wnioski:**

W przedstawionym kodzie mikrokontrolera dodatkowo skonfigurowano przerwanie na pinie 1, aby obsłużyć wymuszanie świateł dla pieszych. W pętli `loop()` sprawdzany jest stan przycisków (pin 13 i pin 1). Światła zielone dla samochodów i czerwone dla pieszych są zapalone dopóki nie zostanie kliknięty przycisk pieszych. Wtedy aktywowana jest sekwencja dla pieszych a po jej zakończeniu ponownie zielone światło otrzymują samochody. Równoległe aktywna jest obsługa przycisku dla trybu nocnego, który ma pierwszeństwo nad oba rodzajami schematów dziennych. Przyciski dla pieszych w trybie nocnym są nieaktywne.