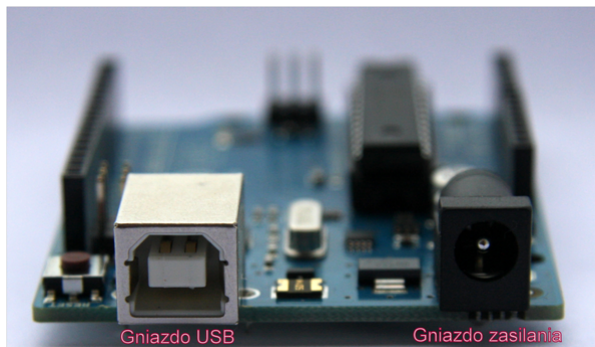


Nazwa implementacji: Wprowadzenie do środowiska mikrokontrolera - b

Autor: Krzysztof Bytow

Opis implementacji: Zastosowanie modułu-interfejsu Arduino oraz obsługa interaktywnego terminala Arduino IDE, służącego do programowania mikrokontrolera. Prezentacja i wyjaśnienie sposobu zestawiania połączeń na podstawie dokumentacji ilustrującej montaż układów ćwiczeniowych. Podłączenie i sterowanie diodą led z wykorzystaniem czujnika nachylenia i buttona.

Wprowadzenie – opis wyprowadzeń, konfiguracja programu, podłączenie do komputera.



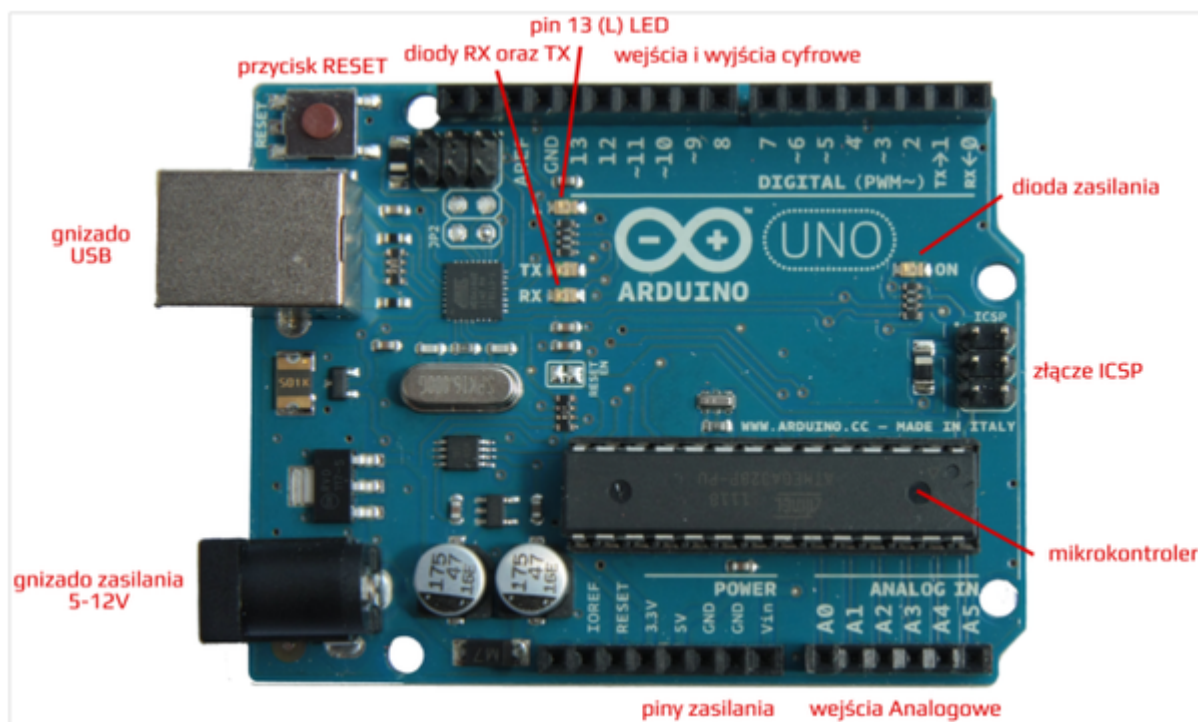
przewód usb wpinamy w gniazdo USB →

1

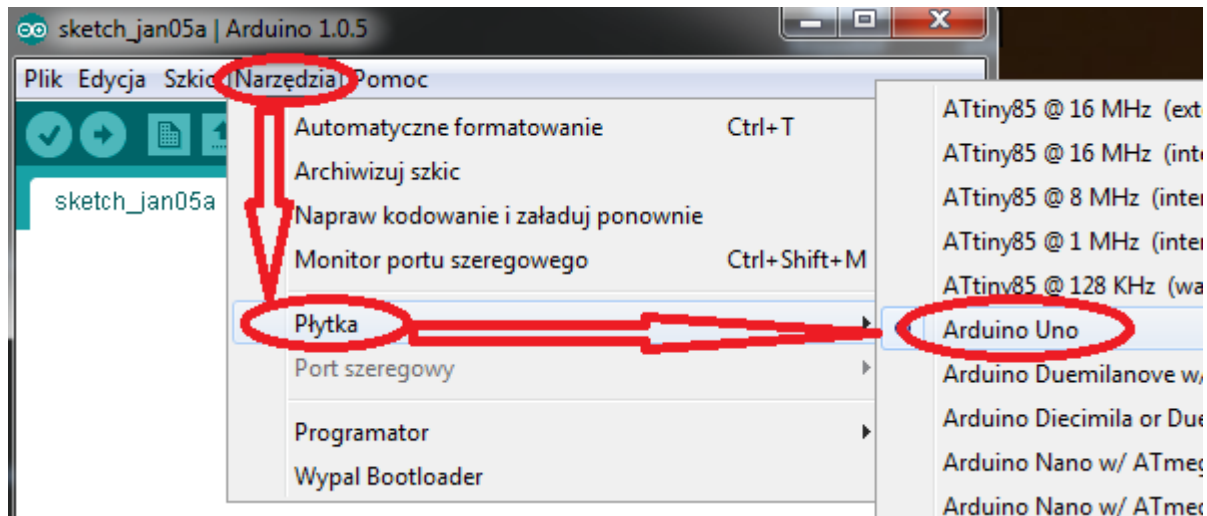
Układ należy podłączyć przewodem USB do komputera.



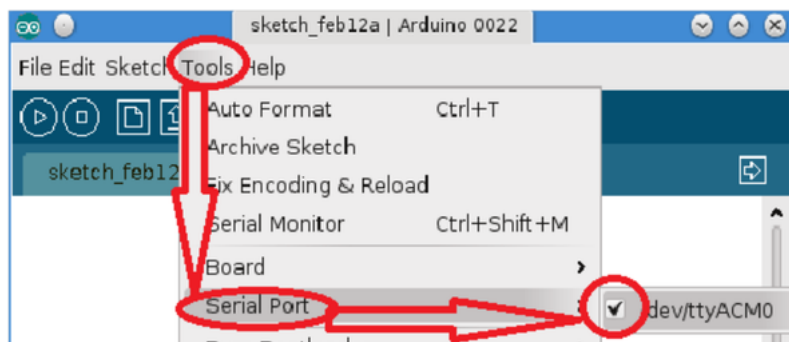
Opis wyprowadzeń Arduino UNO R3.



Uruchomienie i konfiguracja środowiska Arduino IDE.

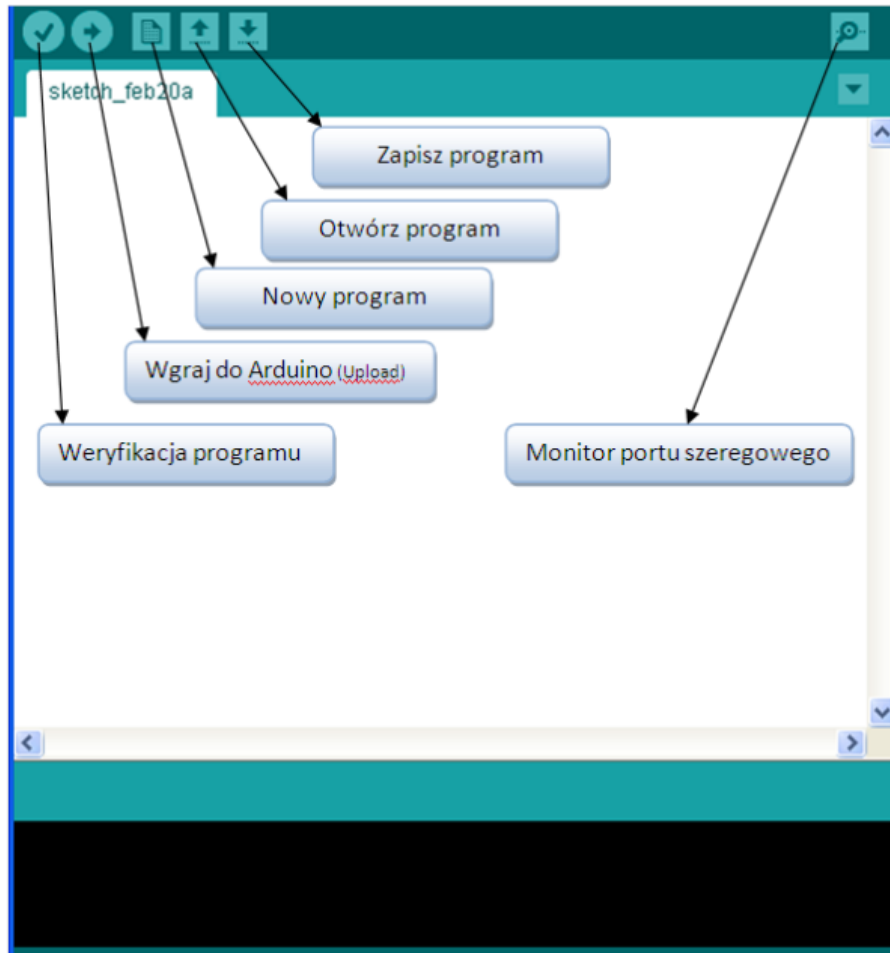


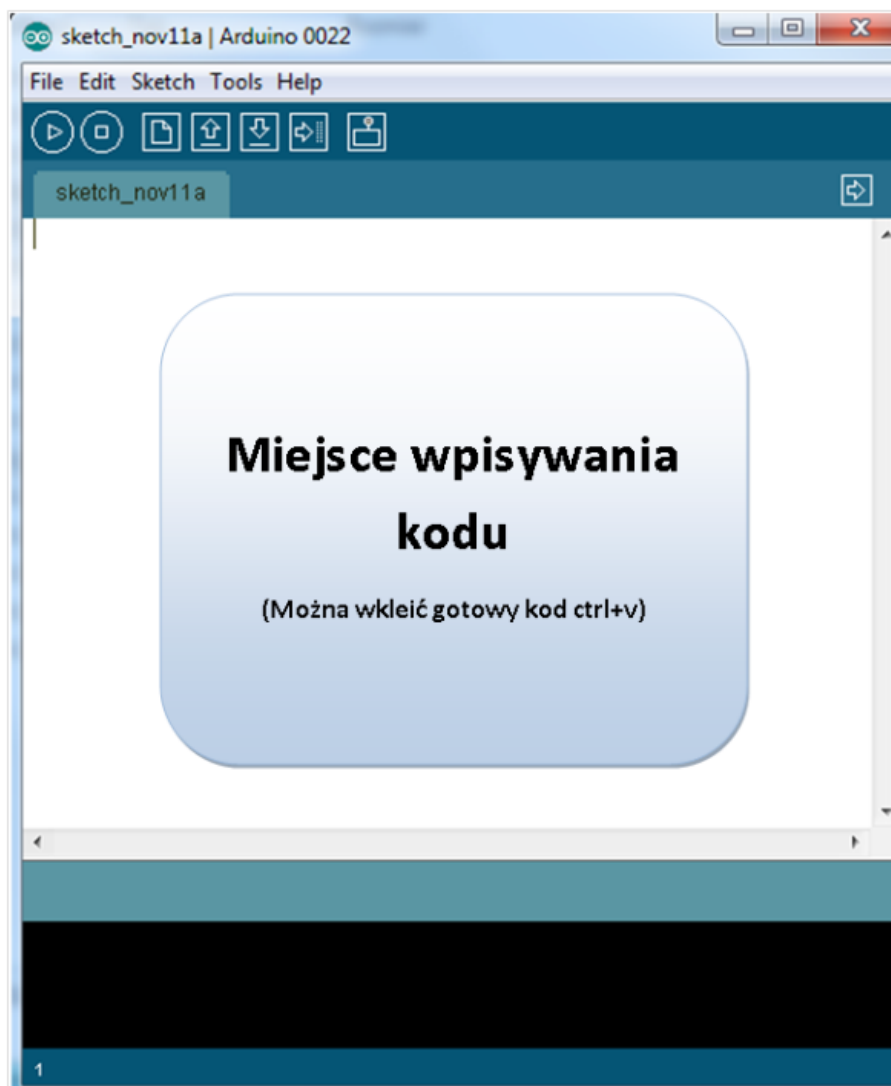
Z zakładki Tools wybieramy Board, a następnie wersję, którą posiadamy (Arduino UNO).



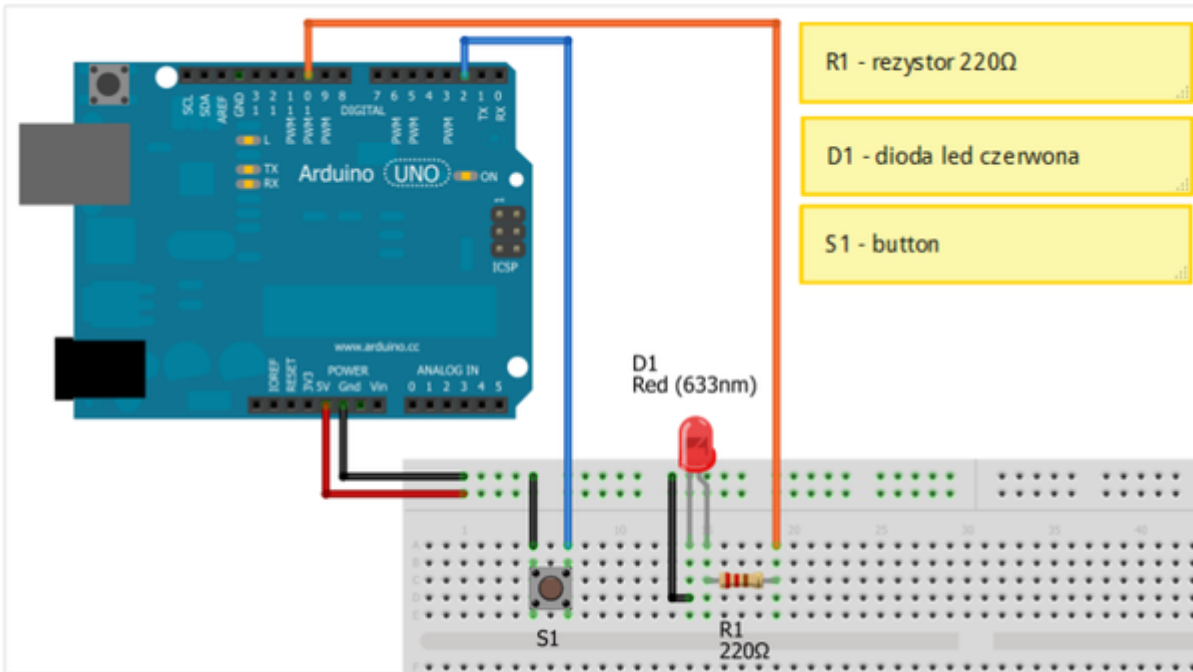
Następnie należy wybrać port, pod który został podłączony układ (Tools → Serial port → odpowiedni port).

Opis interfejsu środowiska Arduino IDE.

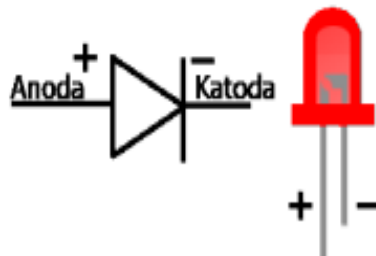




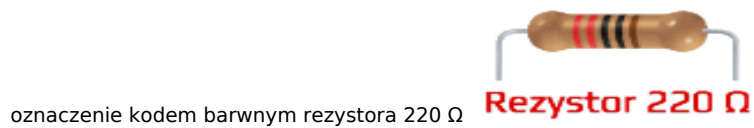
Schemat połączeń - sterowanie z wykorzystaniem buttona



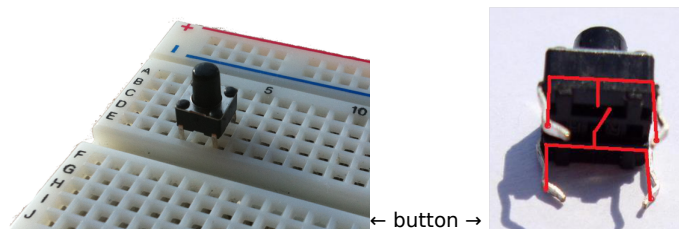
Uczeń/Uczennica po zestawieniu połączeń zgłasza nauczycielowi gotowość do sprawdzenia układu i wszystkich połączeń.



dioda LED czerwona



oznaczenie kodem barwnym rezystora 220 Ω



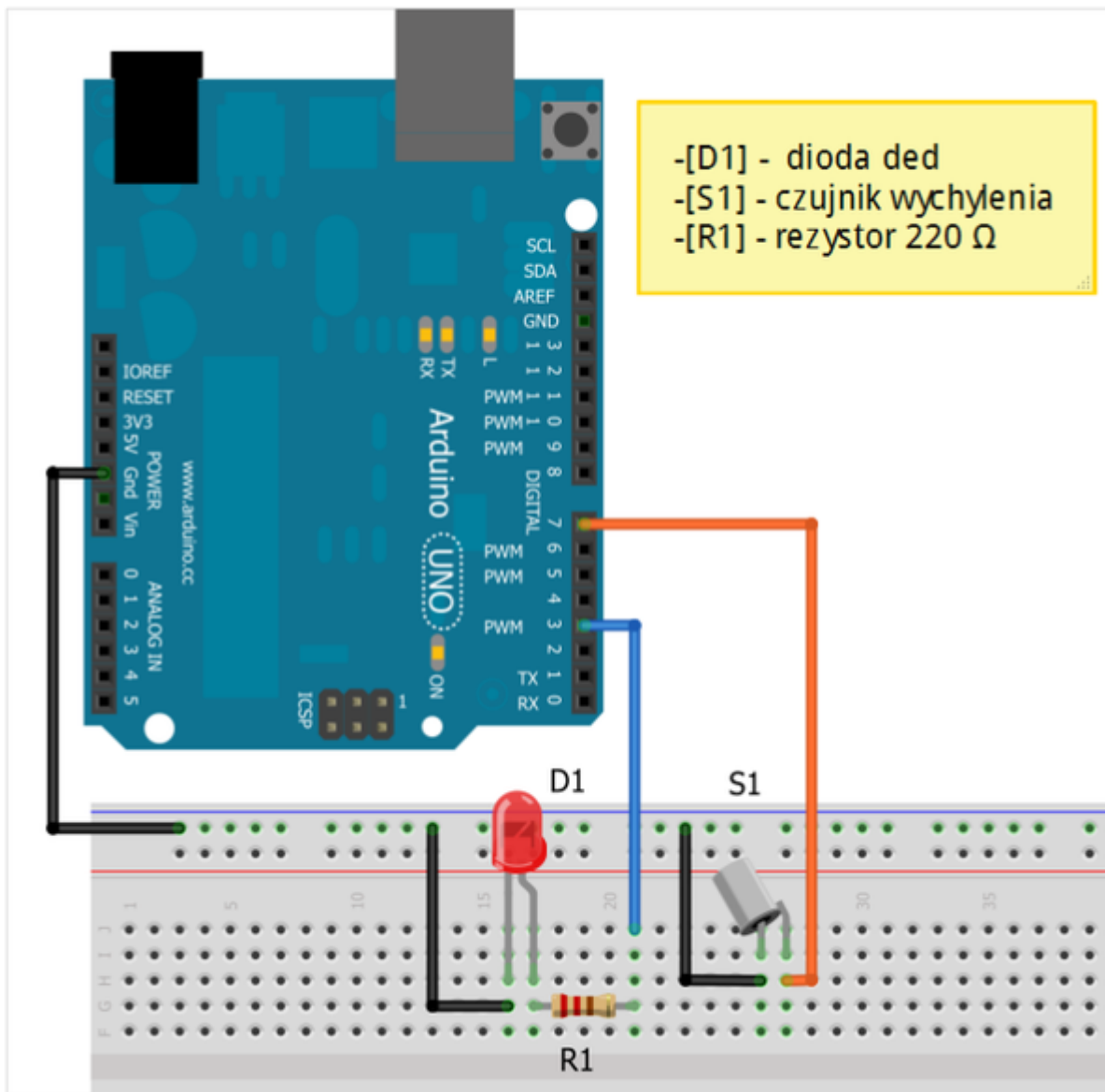


Kod implementacji:

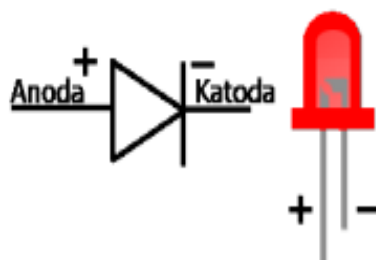
```
int dioda = 10;           // przyznanie etykiety dla numeru pinu
int buton = 2;           // przyznanie etykiety dla numeru pinu
void setup()             // początkowa konfiguracja - część przygotowująca układ do
{                         // działania
  pinMode(dioda, OUTPUT); // ustawienie pinu jako wyjście
  digitalWrite(dioda, LOW); // ustawienie stanu niskiego
  pinMode(buton, INPUT); // ustawienie pinu jako wejście
  digitalWrite(buton, HIGH); // ustawienie stanu wysokiego (pullup)
};
void loop()              // główna pętla
{
  if(digitalRead(buton) == LOW && digitalRead(dioda)==LOW)
    // warunek sprawdza czy wdużono button i czy dioda nie świeci
  {
    // wdużenie buttona powoduje zmianę stanu z niskiego na wysoki
    digitalWrite(dioda, HIGH); // ustawienie stanu wysoki dioda świeci
    delay(200); // czekaj 200ms
  }
  if(digitalRead(buton) == LOW && digitalRead(dioda)==HIGH)
    // warunek sprawdza czy wdużono button i czy dioda świeci
  {
    digitalWrite(dioda, LOW); // ustawienie stanu niskiego
    delay(200); // czekaj 200ms
  }
}
```

Schemat połączeń - sterowanie z wykorzystaniem czujnika wychylenia





Uczeń/Uczennica po zestawieniu połączeń zgłasza nauczycielowi gotowość do sprawdzenia układu i wszystkich połączeń.

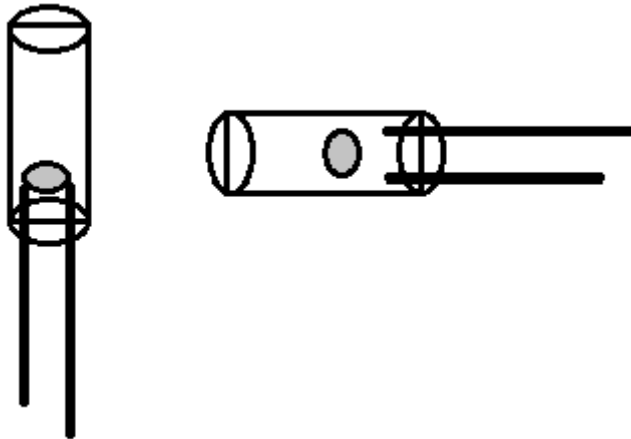


dioda LED czerwona



oznaczenie kodem barwnym rezystora 220 Ω

Rezystor 220 Ω



Czujnik nachylenia

Montując czujnik należy tak ustawić na płytce, aby był w stanie nie przewodzenia (rozłączony).

Kod implementacji:

```
int dioda = 3;           // przyznanie etykiety dla numeru pinu
int buton = 7;          // przyznanie etykiety dla numeru pinu

void setup()            // początkowa konfiguracja - część przygotowująca układ do
{                        // działania
  pinMode(dioda, OUTPUT); // ustawienie pinu jako wyjście
  digitalWrite(dioda, LOW); // ustawienie stanu niskiego
  pinMode(buton, INPUT); // ustawienie pinu jako wejście
  digitalWrite(buton, HIGH); // ustawienie stanu wysokiego (pullup)
}

void loop()              // główna pętla
{
  if(digitalRead(buton) == LOW && digitalRead(dioda) == LOW)
    // warunek sprawdza czy wdużono button i czy dioda nie świeci
    {
      // wdużenie buttona powoduje zmianę stanu z niskiego na wysoki
      digitalWrite(dioda, HIGH); // ustawienie stanu wysokiego dioda świeci
      delay(500); // czekaj 200ms
    }
  if(digitalRead(buton) == HIGH && digitalRead(dioda) == HIGH)
    // warunek sprawdza czy wdużono button i czy dioda świeci

```

9



```
{  
digitalWrite(dioda, LOW);           // ustawienie stanu niskiego  
delay(500);                          // czekaj 200ms  
}  
}
```

