



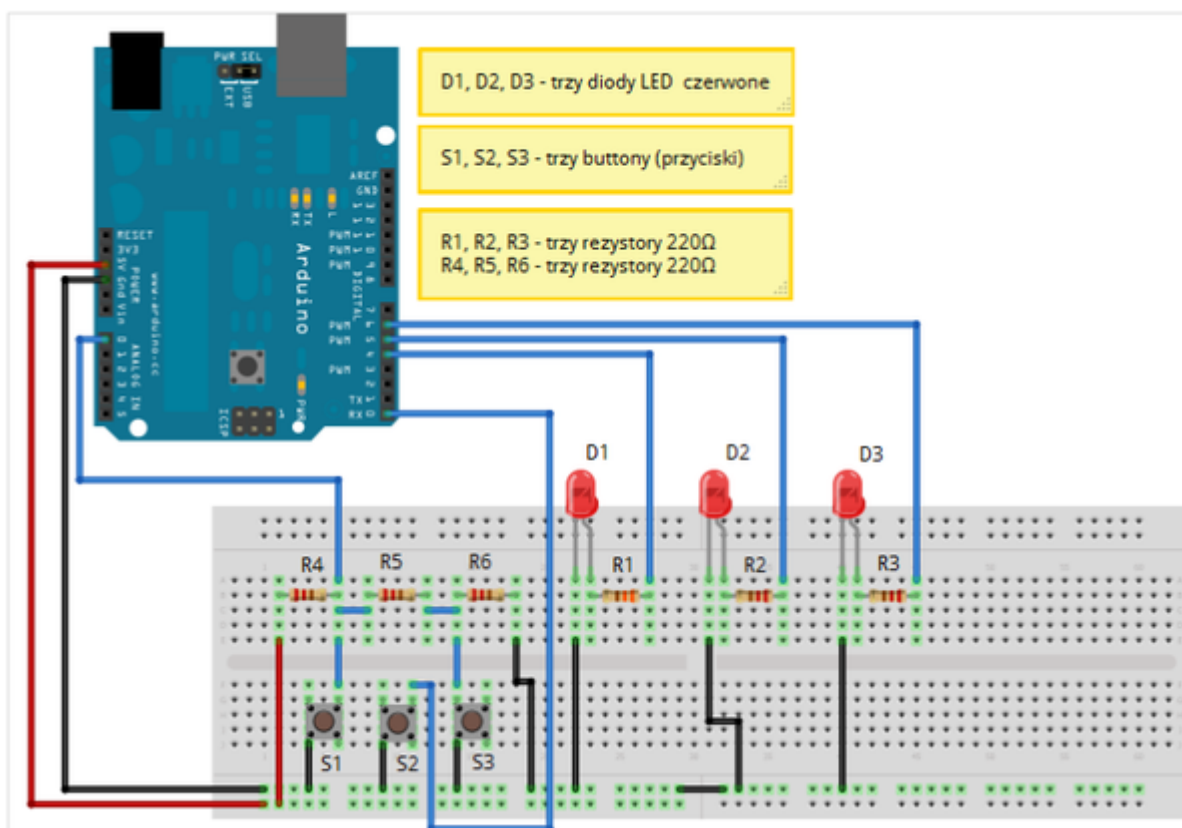
Nazwa implementacji:

Wykorzystanie wejścia analogowego - 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 potencjometru oraz z wykorzystaniem wyjścia PWM. Odczyt wartości z wejścia analogowego.

Schemat połączeń:



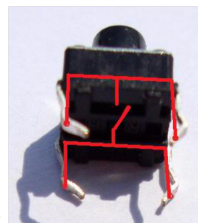
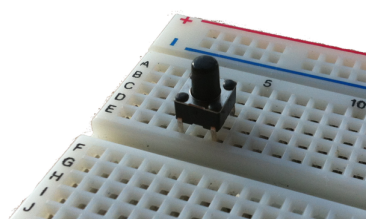
Zalecenia:

- w celu podłączenia zasilania dla +5 V używać przewodów o kolorze ciepłym (np.: czerwony);
- w celu podłączenia masy GND używać przewodów w kolorze zimnym (np.: niebieski, czarny);

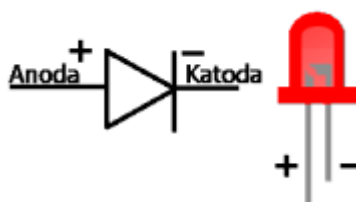
Uczeń/Uczennica po zestawieniu połączeń zgłasza nauczycielowi gotowość do sprawdzenia układu i wszystkich połączeń.



oznaczenie kodem barwnym rezystora 220 Ω →



← button →



dioda LED czerwona →

Kod implementacji:

```

int led1= 4;           // przyznanie etykiety dla numeru pinu
int led2= 5;           // przyznanie etykiety dla numeru pinu
int led3= 6;           // przyznanie etykiety dla numeru pinu
int x= 0;              // tworzymy zmienną typu intiger
int key= 0;            // tworzymy zmienną typu intiger
int button= 0;         // tworzymy zmienną typu intiger

void setup()           // początkowa konfiguracja - część przygotowująca
{                       // układ do działania
  Serial.begin(9600);   // ustawienie prędkości transmisji
  pinMode(led1,OUTPUT); // ustawienie pinu jako wyjście
  pinMode(led2,OUTPUT); // ustawienie pinu jako wyjście
  pinMode(led3,OUTPUT); // ustawienie pinu jako wyjście
  digitalWrite(led1, LOW); // ustawienie na pinie stanu niskiego - dioda nie świeci
  digitalWrite(led2, LOW); // ustawienie na pinie stanu niskiego - dioda nie świeci
  digitalWrite(led3, LOW); // ustawienie na pinie stanu niskiego - dioda nie świeci
  pinMode(button, INPUT); // ustawienie pinu przycisków jako wejścia
  digitalWrite(button, HIGH); // włączenie wewnętrznych rezystorów podciągających
}

void loop()            // główna pętla
{
  buton();             // wywołanie funkcji buton
  if (key == 1)        // pętla if - wejście gdy spełniony warunek
    2

```





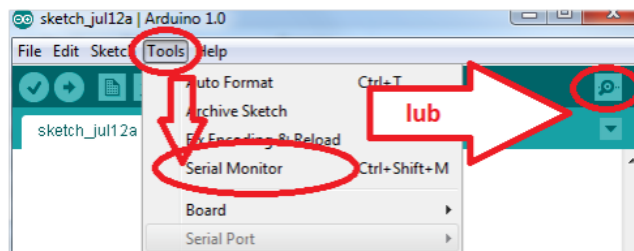
```
{
  set(5,true);           //wywołanie funkcji set
  set(6,false);         //wywołanie funkcji set
}
if (key == -1)          // wywołanie instrukcji warunkowej – wejście gdy spełniony warunek
{
  set(5,false);         //wywołanie funkcji set
  set(6,true);         //wywołanie funkcji set
}
if (digitalRead(button) == LOW)    // wywołanie instrukcji warunkowej
{
  // spełniony
  digitalWrite(led1, HIGH);        //zapalenie diody
  digitalWrite(led2, HIGH);        //zapalenie diody
  digitalWrite(led3, HIGH);        //zapalenie diody
  delay(2500);                      //odczekanie 2500ms = 2,5s
  digitalWrite(led1, LOW);         //dioda nie świeci
  digitalWrite(led2, LOW);         //dioda nie świeci
  digitalWrite(led3, LOW);         //dioda nie świeci
}
}

void set(int i, boolean s)         // funkcja set
{
  if(i == 5)                       // wywołanie instrukcji warunkowej
    digitalWrite(led1, s ? HIGH : LOW); //zapalenie lub nie diody - operator wyrażenia warunkowego
  if(i == 6)                       // wywołanie instrukcji warunkowej
    digitalWrite(led3, s ? HIGH : LOW); //zapalenie lub nie diody - operator wyrażenia warunkowego
}

void buton()
{
  int b=analogRead(0);              // przypisanie b – wartości odczytanej z wej. analogowego
  delay(150);                       // odczekanie 150ms = 0,15s
  if (b<20)                          // wywołanie instrukcji warunkowej
  {
    Serial.println("lewy");          // wyświetlenie tekstu na komputerze
    key = 1;
  }
  else                                // gdy powyższy warunek nie spełniony wykonaj
  if (b > 20 && b < 550)             // wywołanie instrukcji warunkowej
  {
    Serial.println("prawy");        // wyświetlenie tekstu na komputerze
    key = -1;
  }
}
```

Po wgraniu kodu należy w programie Arduino IDE uruchomić Serial Monitor (lub terminal), aby obserwować wyświetlany tekst.





Filmy instruktażowe:

<http://youtu.be/FK9fpn7J1pM>