



Nazwa implementacji:

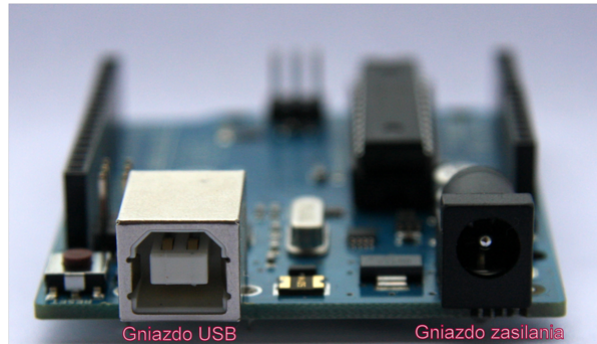
Budowa układów i programowanie modułu - a

**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.

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



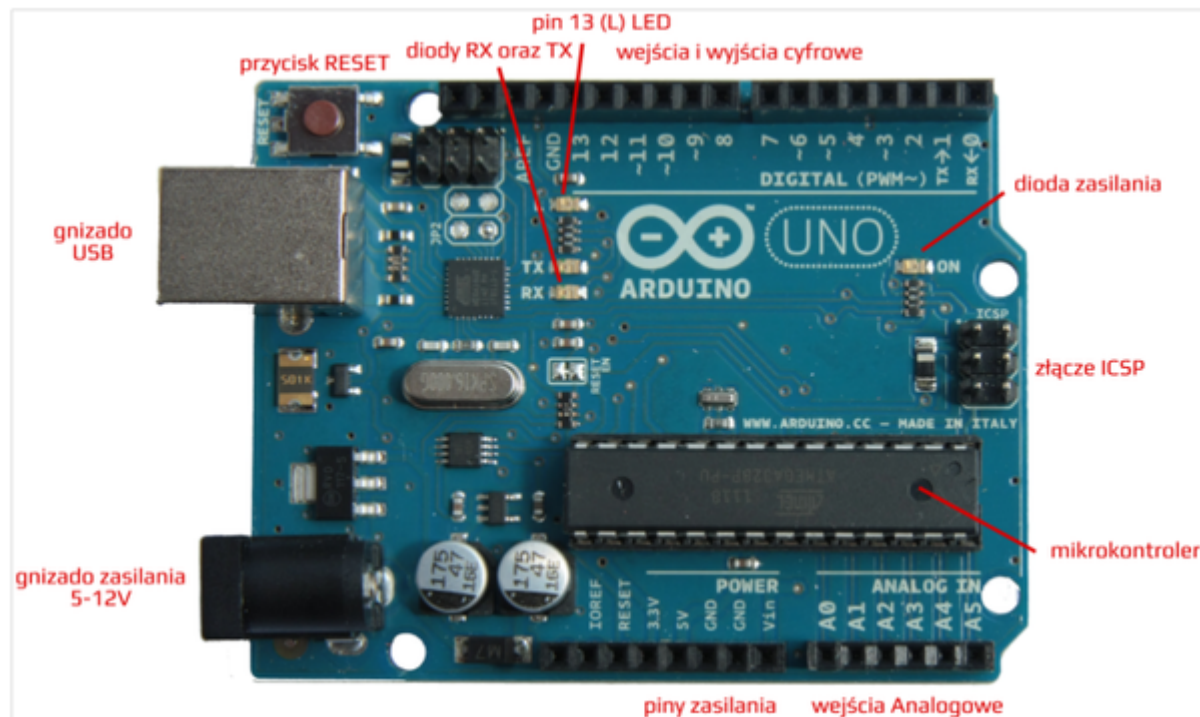


przewód usb wpinamy w gniazdo USB →

Układ należy podłączyć przewodem USB do komputera.

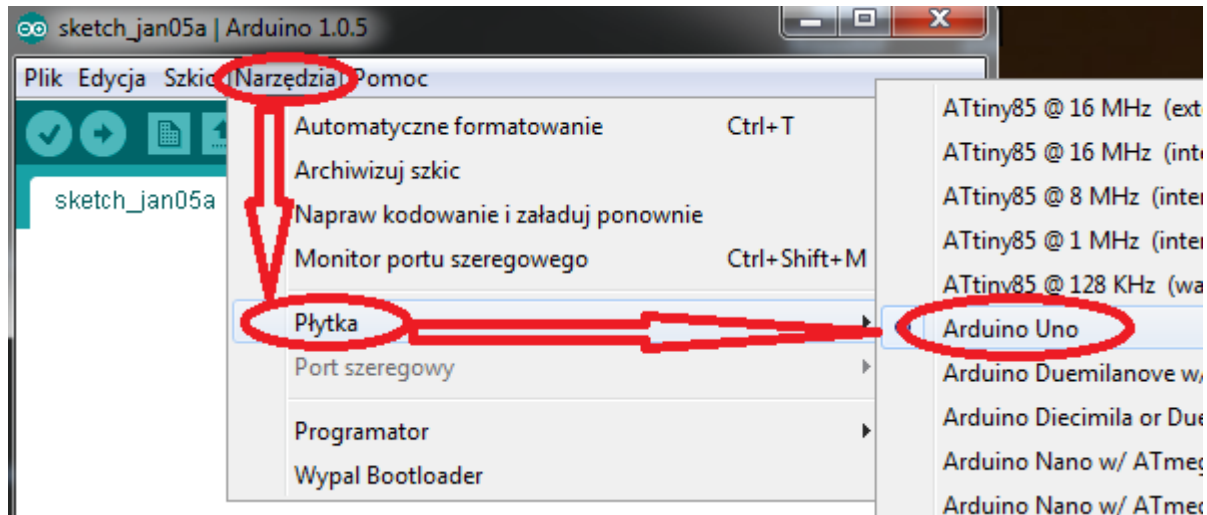


Opis wyprowadzeń Arduino UNO R3.

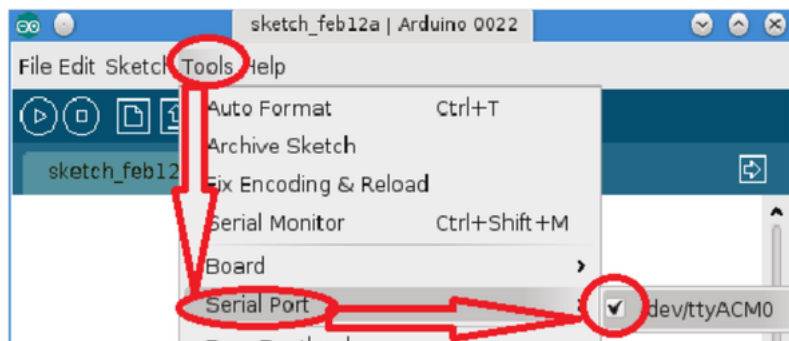




Uruchomienie i konfiguracja środowiska Arduino IDE.



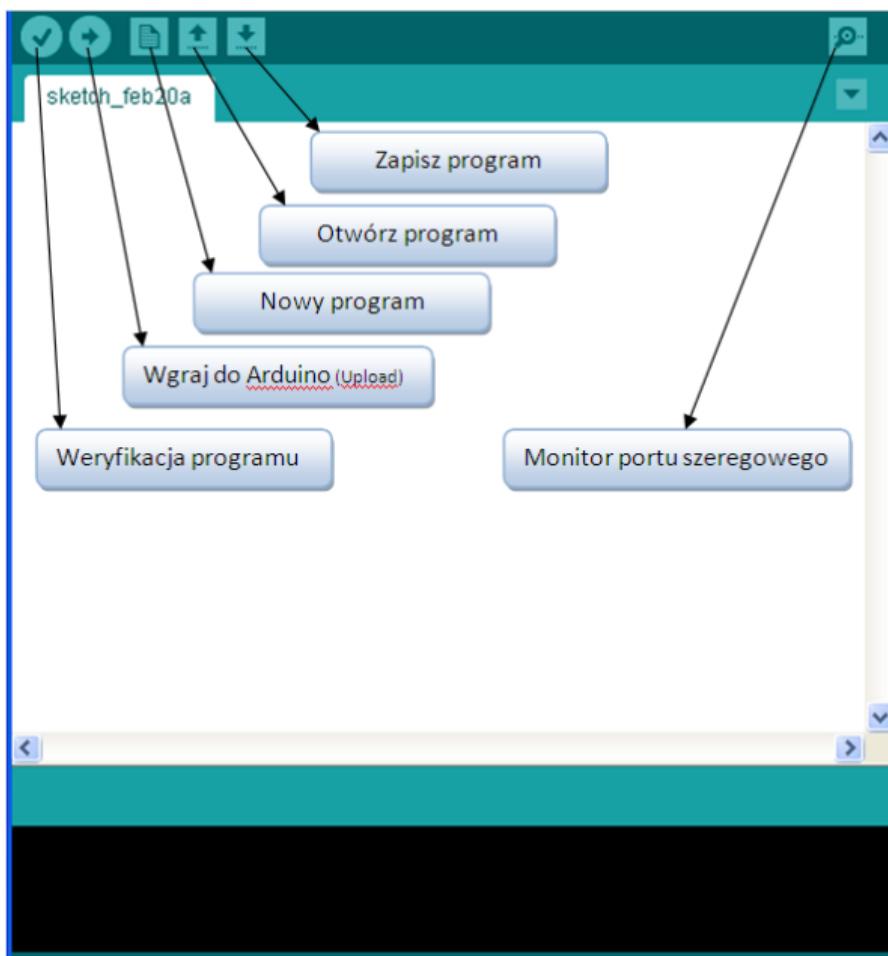
Z zakładki Tools wybieramy Board, a następnie wersję, którą posiadamy (Arduino UNO).

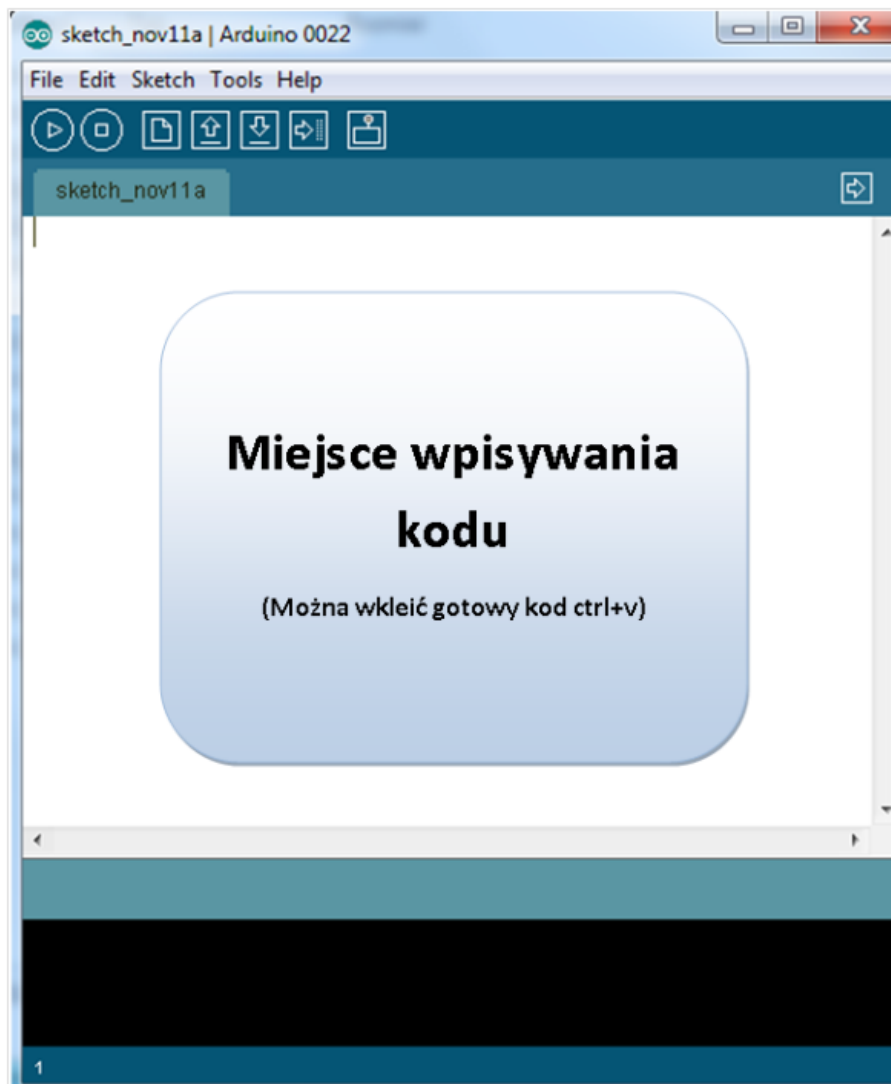


Następnie należy wybrać port, pod który został podłączony układ (Tools → Serial port → odpowiedni port).

Opis interfejsu środowiska Arduino IDE.



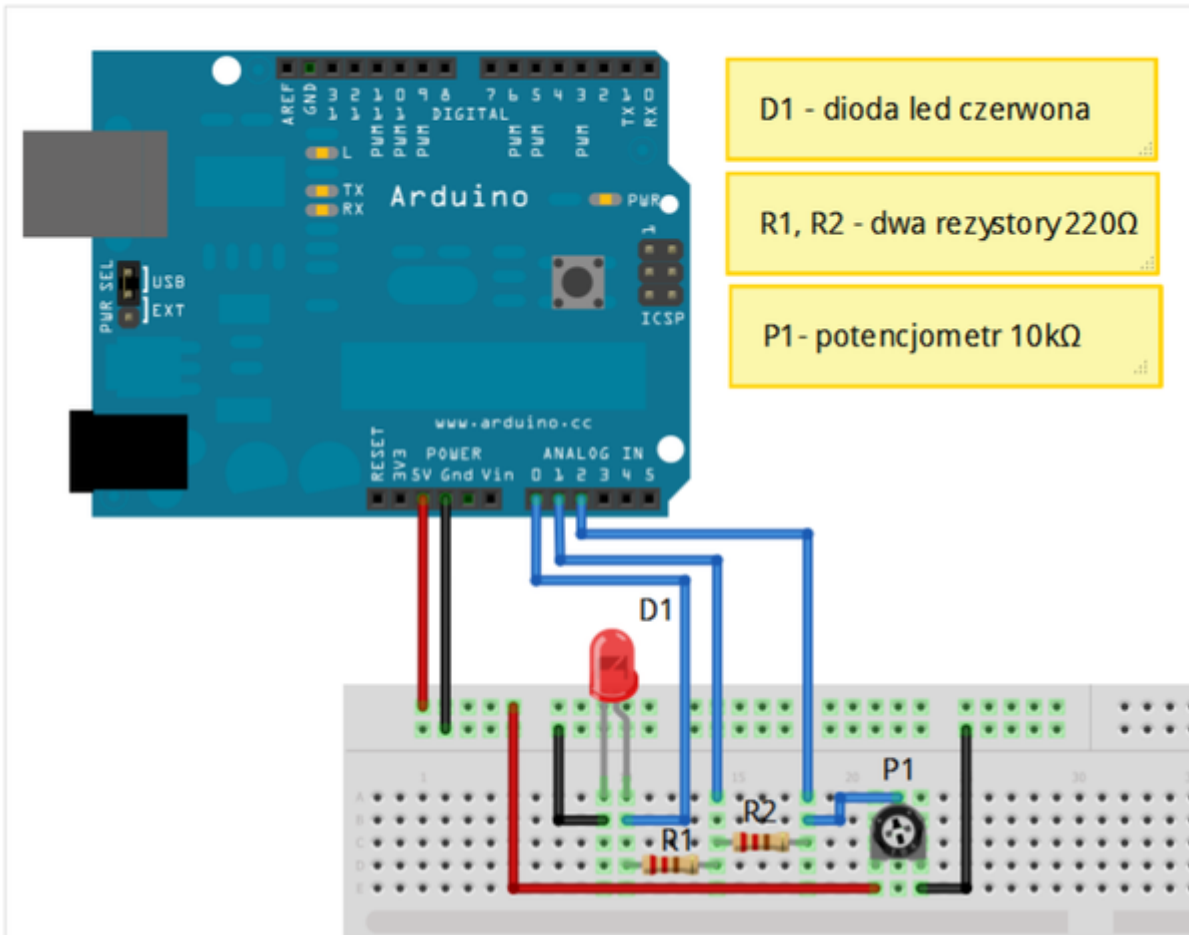




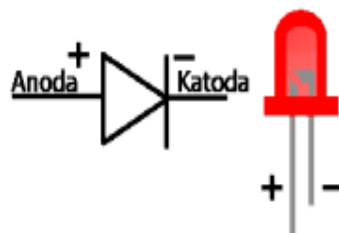
Schemat połączeń – sterowanie diodą z wykorzystaniem potencjometru.







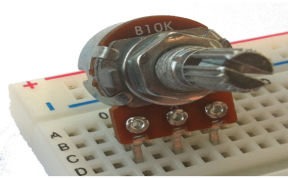
Uczeń/Uczennica po zestawieniu połączeń zgłasza nauczycielowi gotowość do sprawdzenia układu i wszystkich połączeń.



dioda LED czerwona



oznaczenie kodem barwnym rezystora 220 Ω



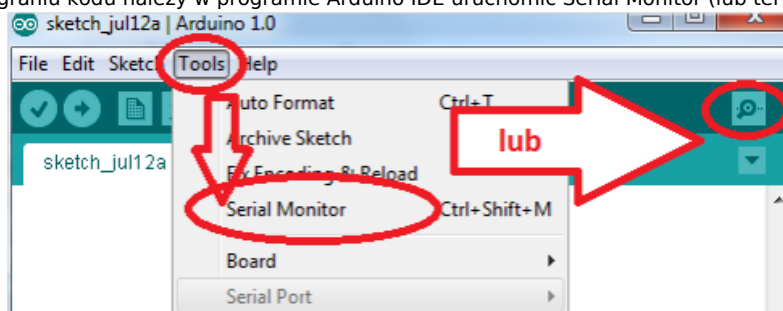
potencjometr 10kΩ →

Kod implementacji:

W zbudowanym układzie sterujemy jasnością diody, regulując wartość rezystancji na potencjometrze. Dodatkowo podłączyliśmy się w trzech miejscach do wejść analogowych w celu pomiaru wartości, a otrzymane wyniki zaprezentowane są na wyświetlaczu monitora. Uwaga wynik pomiaru może być obciążony błędem wynikającym z zasilania USB (nie zawsze na wyjściu jest 5 V).

```
float x,y,z;           // tworzymy zmienną typu float do przechowywania i pracy z
                      // liczbami zmiennoprzecinkowymi
void setup()          // początkowa konfiguracja - część przygotowująca układ do
{                     // działania
  Serial.begin(9600); // ustawienie prędkości komunikacji
}
void loop()           // główna pętla
{
  x=analogRead(0)*5/1024.0; // przypisanie wartości odczytanej z wej. analogowego
  y=analogRead(1)*5/1024.0; // i przeliczenie na napięcie (*); podzielone przez dokładność
  z=analogRead(2)*5/1024.0; // przetwornika analogowo cyfrowego
  Serial.println(x);      // wysłanie do komputera wartości x
  Serial.println(y);      // wysłanie do komputera wartości y
  Serial.println(z);      // wysłanie do komputera wartości z
  delay(2500);           // oczekiwanie 2,5 sekundy
  Serial.println("Następny pomiar"); // wysłanie komunikatu
  Serial.println("-----"); // wysłanie komunikatu
}
```

Po wgraniu kodu należy w programie Arduino IDE uruchomić Serial Monitor (lub terminal).



Komentarz (nie wpisywać do kodu implementacji):

void setup() - początkowa konfiguracja - część przygotowująca;





void loop() - główna pętla - wpisany program będzie wykonywany cyklicznie, wciśnięcie przycisku reset powoduje uruchomienie kodu od początku;

funkcja "pinMode" ustawia kierunek sygnału na pinie modułu (INPUT, OUTPUT);

funkcja "digitalWrite" ustawia cyfrowy stan sygnału na pinie modułu (HIGH, LOW);

funkcja "delay" czeka ustaloną w milisekundach długość czasu (1000 ms = 1 s);

(\*) maksymalne napięcie mierzone przez układ to 5V, (\*\*) dokładność przetwornika A/C to 10 bitów, czyli 1024 wartości;

