



# Politechnika Łódzka

Instytut Elektroniki



Fundusze  
Europejskie  
Wiedza Edukacja Rozwój



Rzeczpospolita  
Polska

Unia Europejska  
Europejski Fundusz Społeczny



# Arduino 101

Ścieżki Młodych Odkrywców  
(edycja II, lato 2019)

Marcin Kociołek, Paweł Poryzała

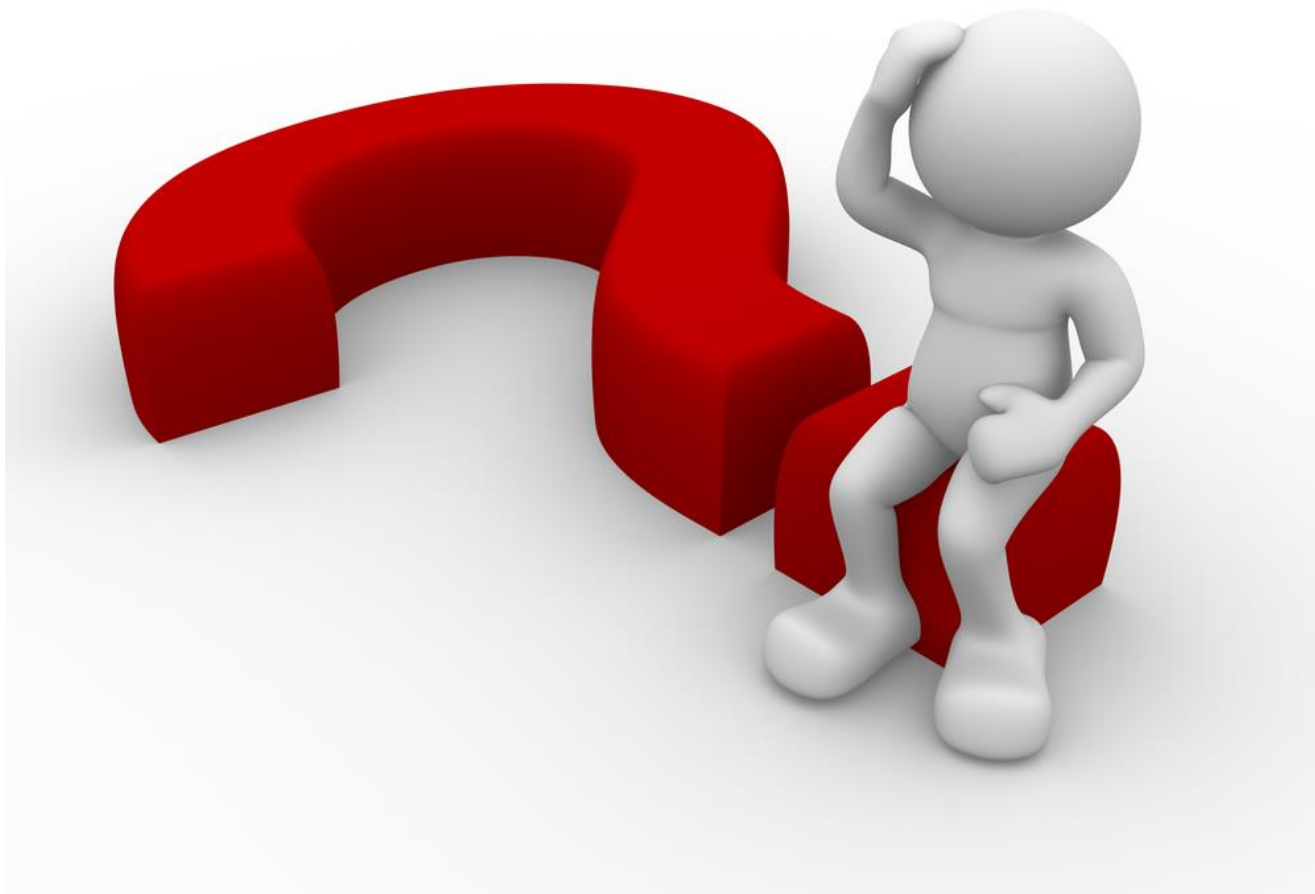


# Czym jest Arduino?

- Arduino, to zestaw uruchomieniowy z popularnym mikrokontrolerem AVR. Jego zalety to m.in.:
  - nie wymaga zewnętrznego programatora,
  - współpracuje z darmowym kompilatorem i środowiskiem IDE,
  - dedykowany język programowania bazujący na C/C++ (z dużą liczbą ogólnodostępnych bibliotek),
  - olbrzymia społeczność zgromadzona wokół projektu, ogromna ilość gotowych projektów, pomoc środowiska,
  - wiele odmian (tanich) płytek rozwojowych oraz modułów rozszerzeń (tzw. *shieldów*): od sterowników silników, wyświetlaczy, aż po moduły wykonawcze).

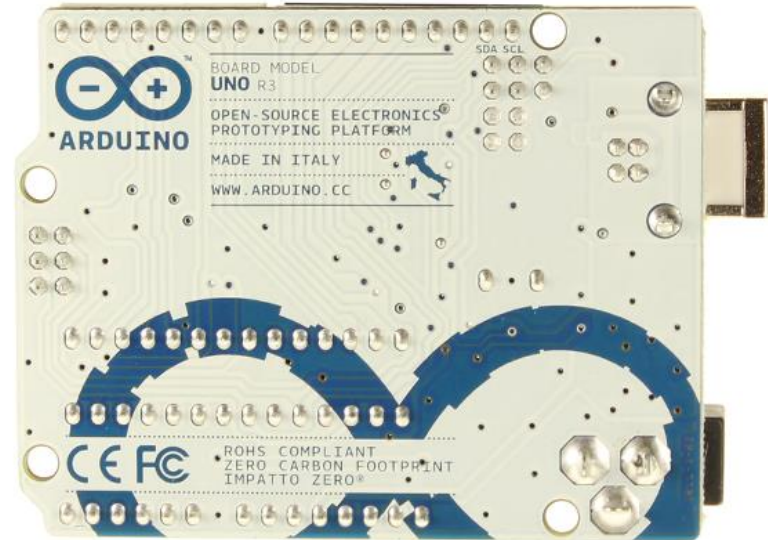
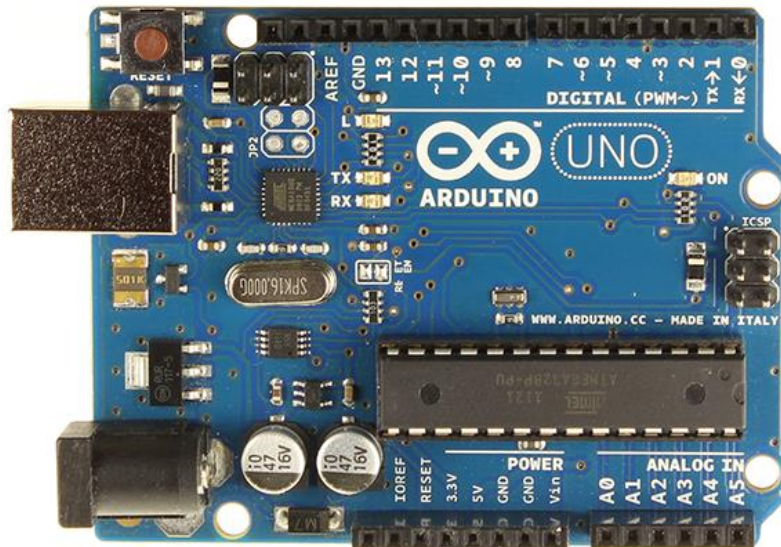
# Zastosowania

- projekty ogólnodostępne, elastyczne, tanie i niewymagające dużych nakładów finansowych (uczniowie i hobbisci),
- alternatywa dla osób, które nie mają dostępu do bardziej zaawansowanych kontrolerów, wymagających bardziej skomplikowanych narzędzi (studenci),
- projekty komercyjne (drukarki 3D).



Jakiego sprzętu będziemy używać?

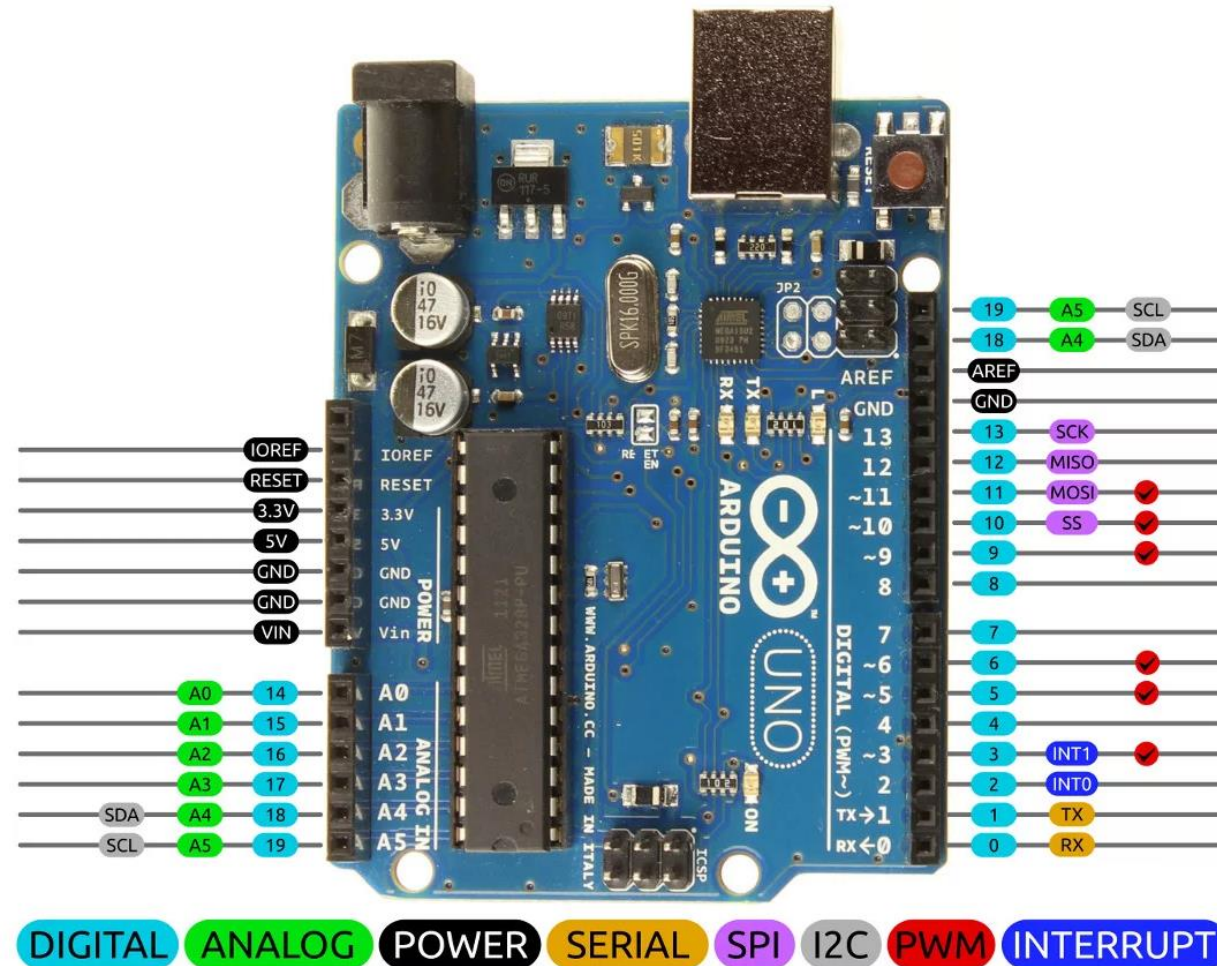
# Arduino UNO R3



# Co znajduje się na pokładzie?

- 8-bitowy mikrokontroler firmy **Microchip Corporation: AVR ATmega328** pracujący z częstotliwością 16 MHz (**szesnaście milionów operacji na sekundę !**)
  - **14 programowalnych cyfrowych wejść/wyjść**,
    - 6 z nich można używać jako wyjścia **PWM** (np. do sterowania silnikami),
  - **6 wejść analogowych** (pomiar czujników analogowych),
  - pamięci programu (32kB) oraz danych (2kB),
  - sygnał resetu (zerowanie systemu) oraz zasilanie.

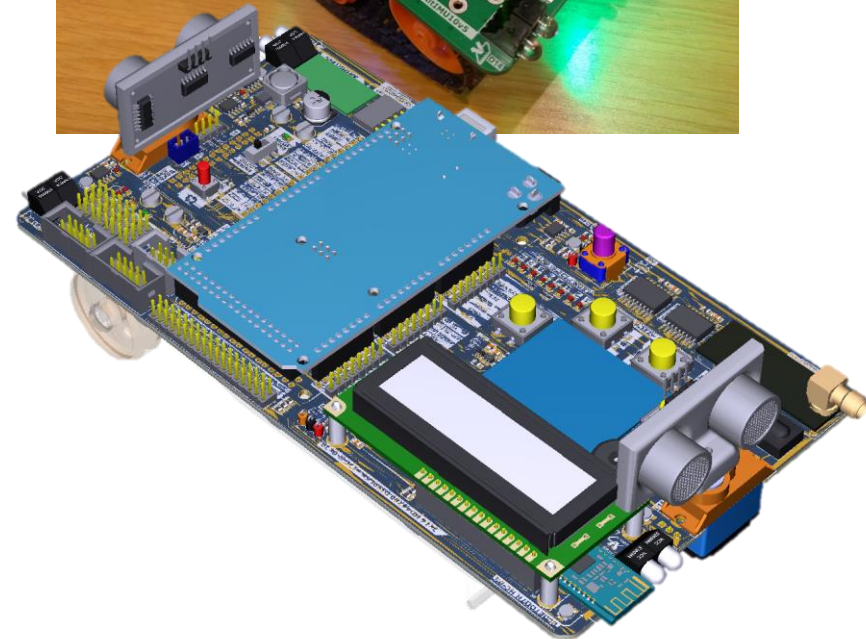
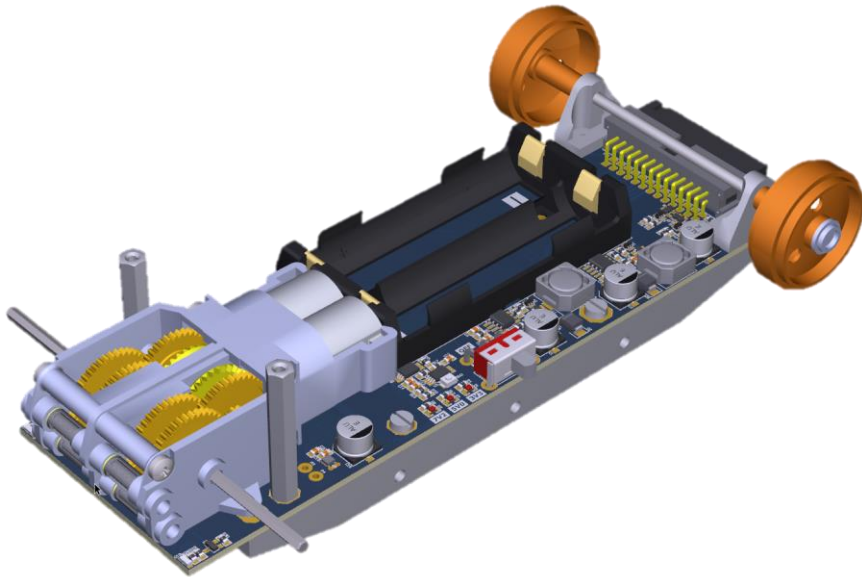
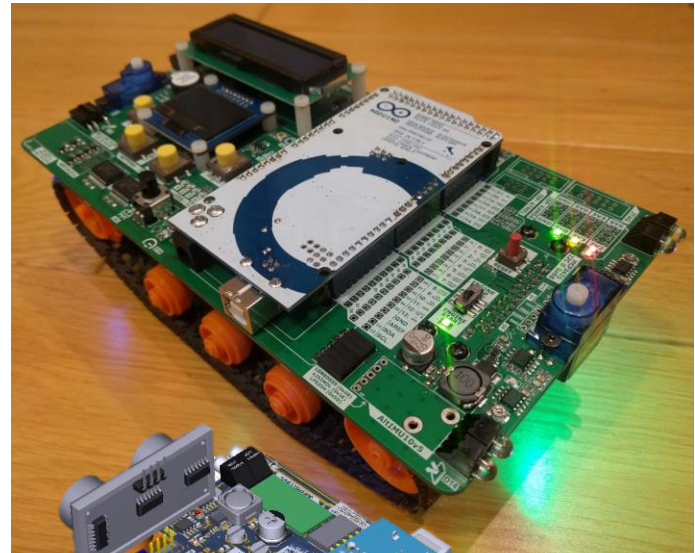
# Co znajduje się na pokładzie?



- porty cyfrowe (wejście / wyjście): 0 lub 5V,
- wejścia analogowe: od 0 do 5V,
- linie kontrolne / zasilania,
- komunikacja szeregową: UART, SPI, I2C,
- linie o modulowanej szerokości impulsów (PWM),
- linie przerwań.



# IE-TRACK-KIT-PWR-1, ver. 1.2 + IE-TRACK-KIT-ARDUINO-1, ver. 1.1





# Co znajduje się na pokładzie?

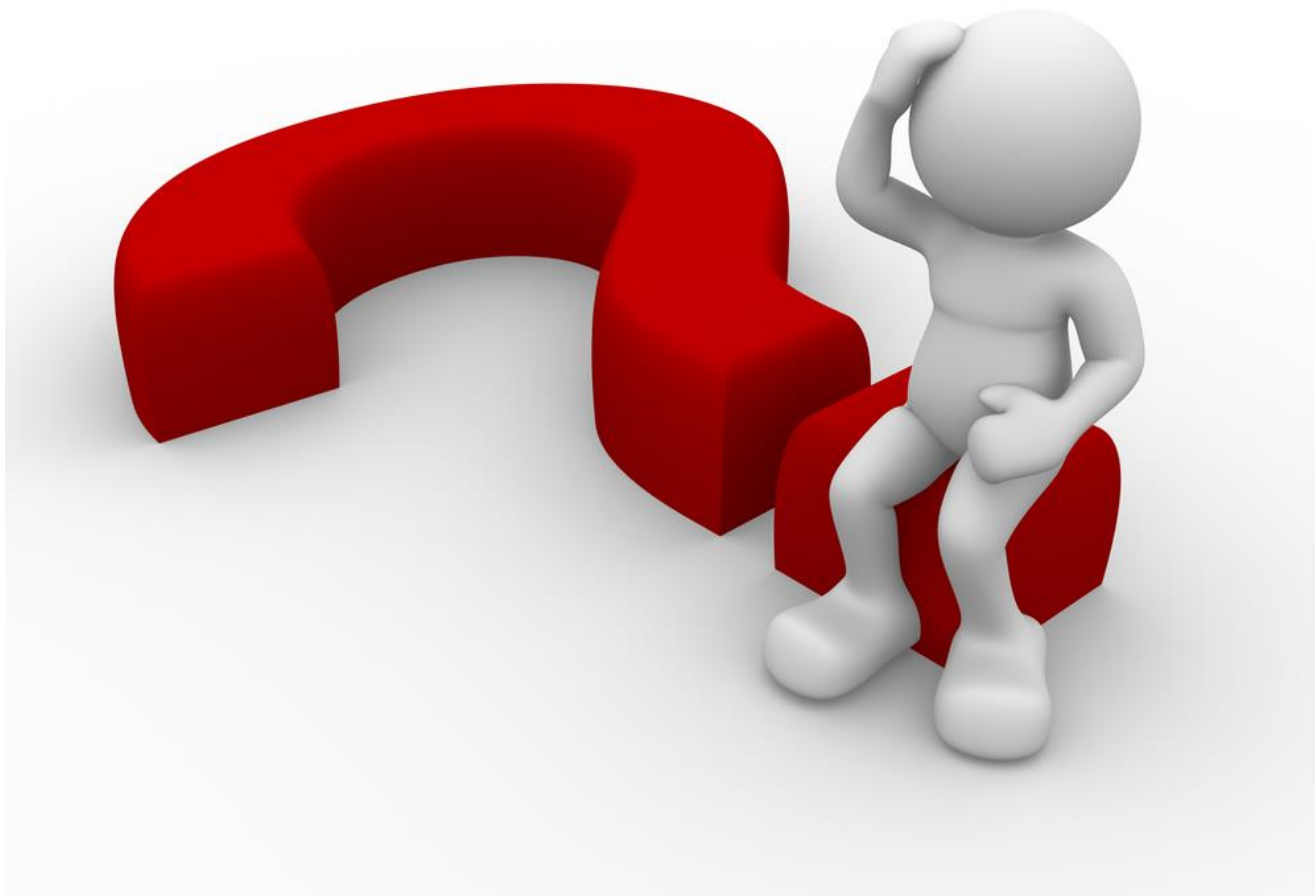


- Arduino UNO R3 lub Arduino MEGA 2560 R3,
- 4 przyciski,
- 24 diody LED (w tym 14 RGB),
- oświetlenie pojazdu (przód, tył, kierunkowskazy),
- dwa sterowniki silników DC (prawa i lewa gąsienica),
- wyświetlacz LCD 2x16 znaków,
- wyświetlacz OLED 128x64px,
- 3 cyfrowe detektory przeszkód (światło podczerwone),
- ultradźwiękowe czujniki odległości z możliwością skanowania przestrzeni przed/za pojazdem,

# Co znajduje się na pokładzie?

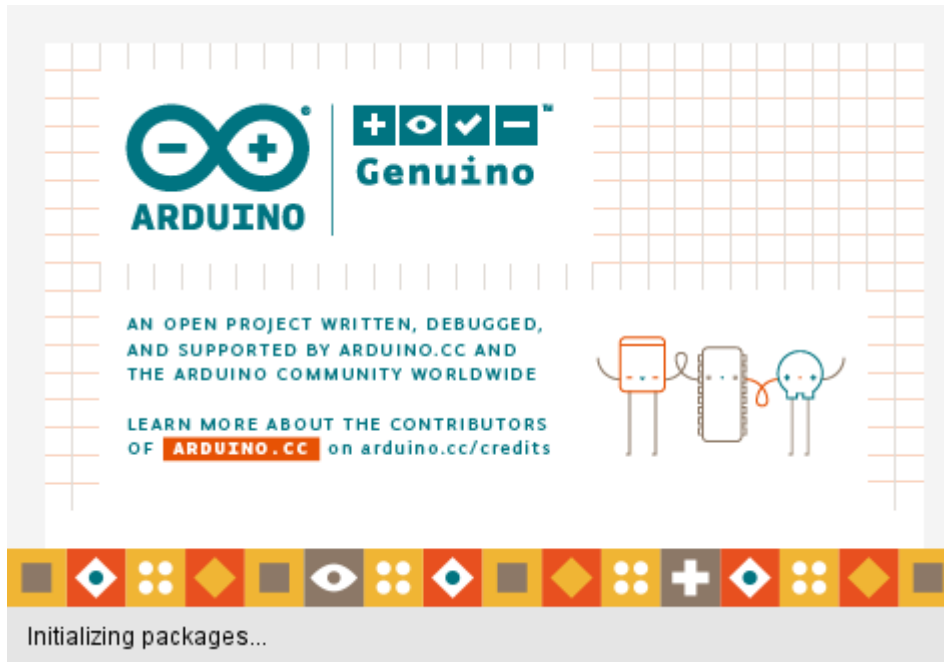


- potencjometr analogowy,
- czujnik temperatury,
- czujnik światła,
- detektor linii,
- brzęczyk piezoelektryczny (*buzzer*) z możliwością generowania melodii,
- odbiornik podczerwieni 36kHz (sterowanie z pilota IR),
- moduł komunikacji bezprzewodowej w standardzie Bluetooth 2.0,
- uniwersalny układ nadawczo-odbiorczy na pasmo 2,4GHz (ISM).

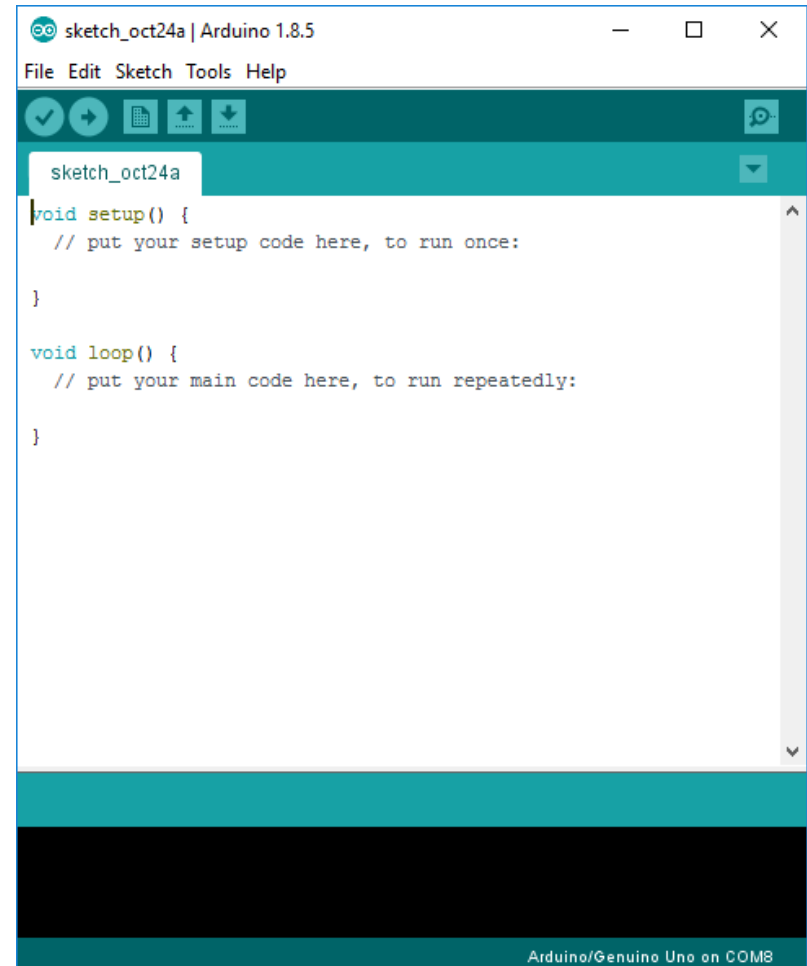


Jak my to wszystko wprawimy w ruch?

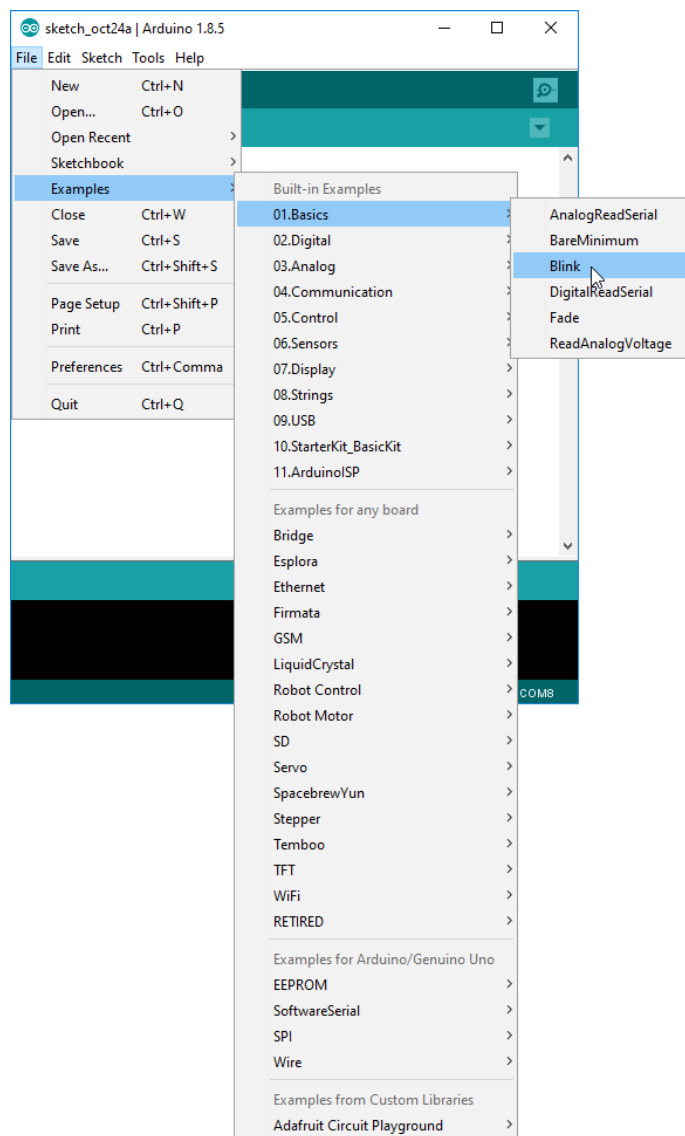
# Arduino IDE



<https://www.arduino.cc>



# Pierwszy program



- Jeden z wbudowanych w środowisko przykładów:
  - EXAMPLES
  - 01. BASICS
  - BLINK
- Omówmy kod przykładowego programu...

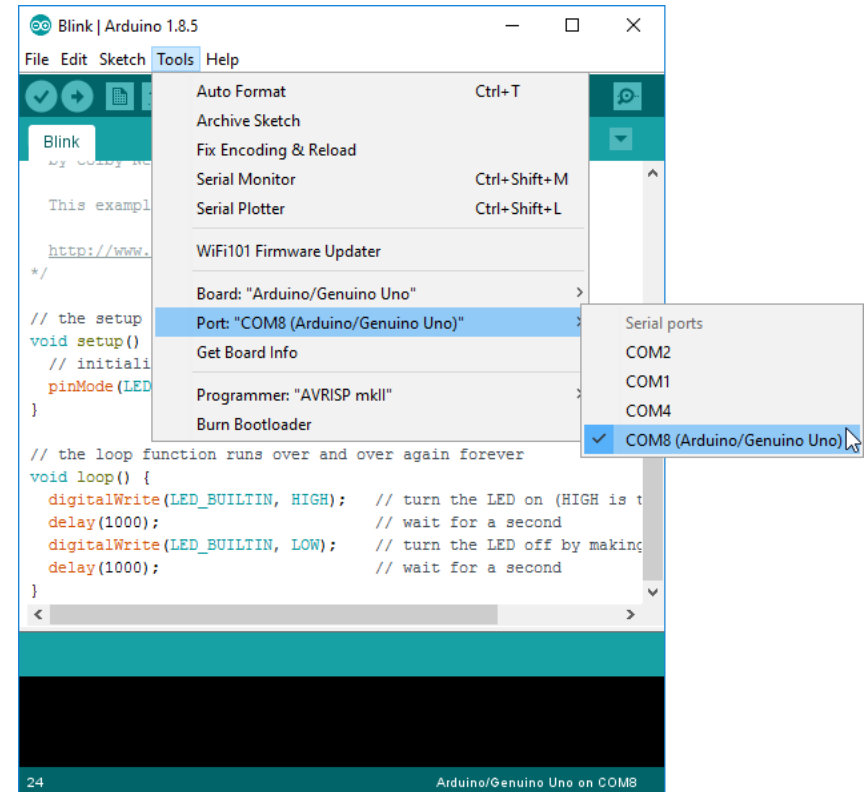
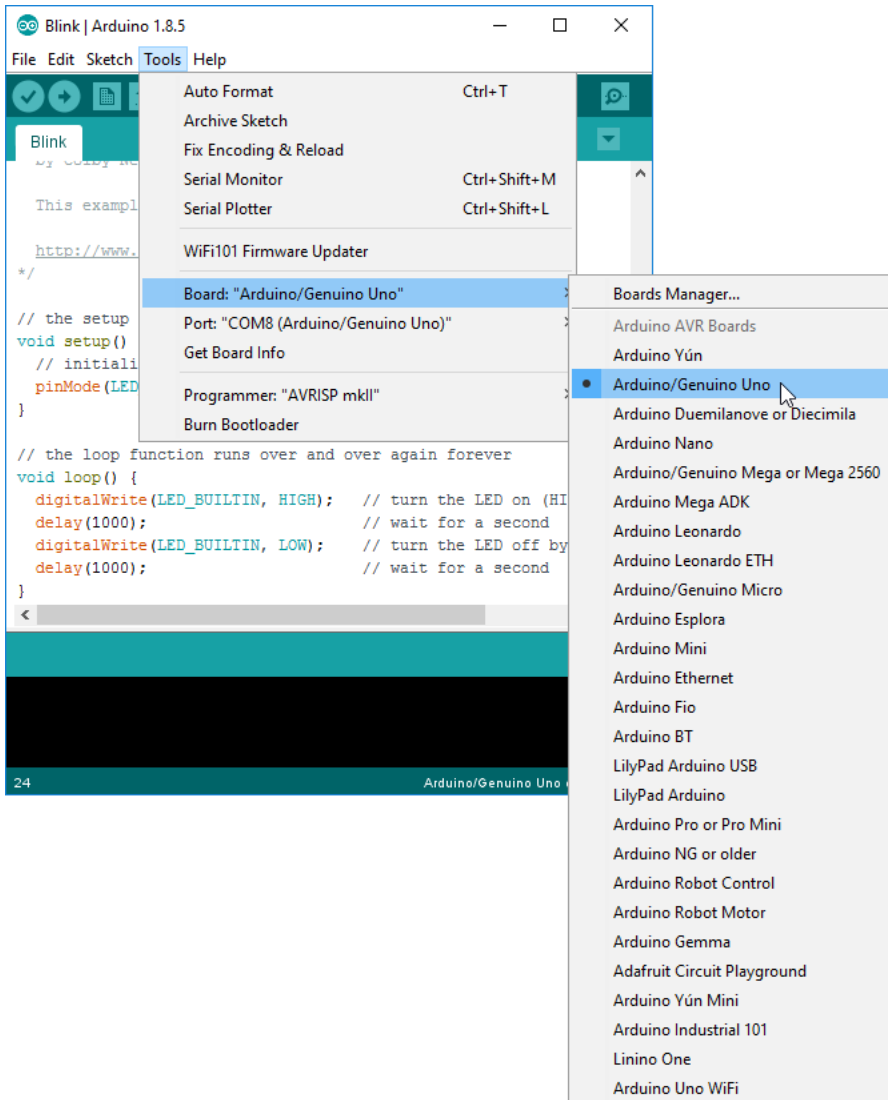
# Pierwszy program

```
/*  
  Blink ...  
  Turns an LED on for one second, then off for one second, repeatedly.  
  Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO  
  it is attached to digital pin 13, ...  
  
  This example code is in the public domain.  
  http://www.arduino.cc/en/Tutorial/Blink  
*/  
  
// the setup function runs once when you press reset or power the board  
void setup() {  
  // initialize digital pin LED_BUILTIN as an output.  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(1000); // wait for a second  
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW  
  delay(1000);  
}
```

Zamieniamy `'LED_BUILTIN'` na `'6'`

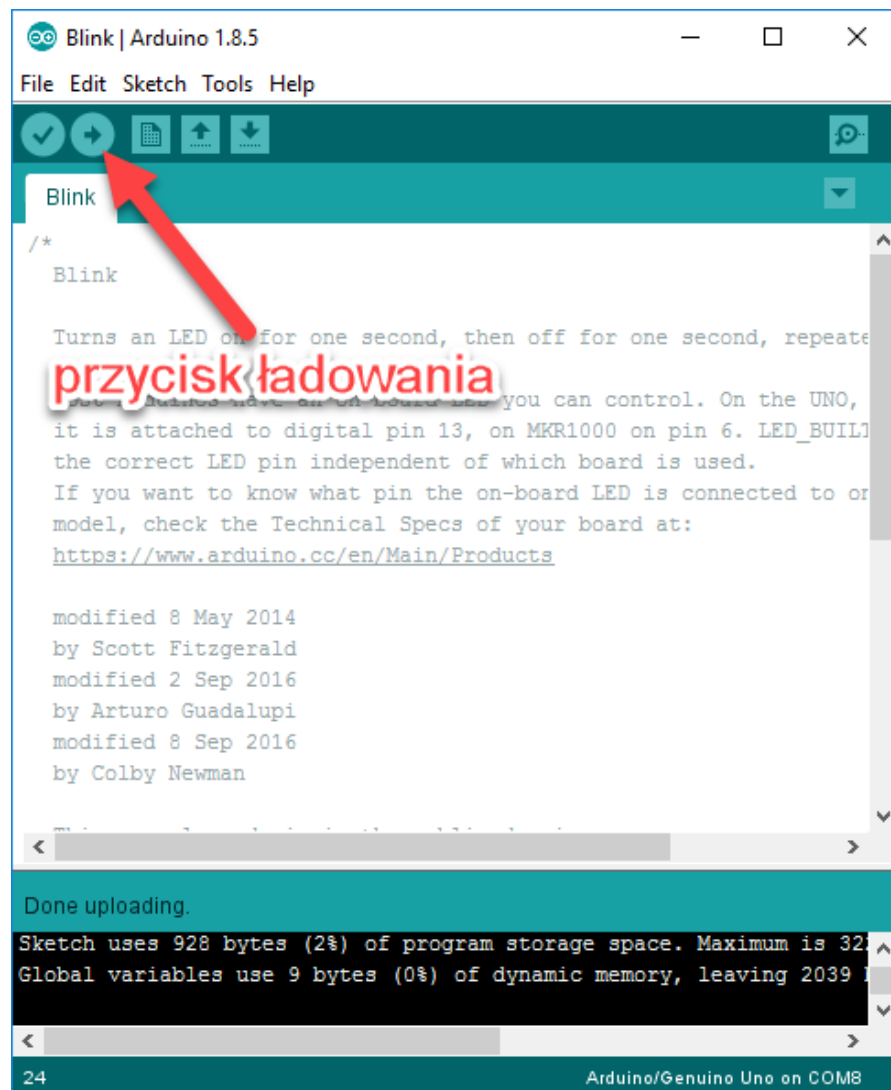
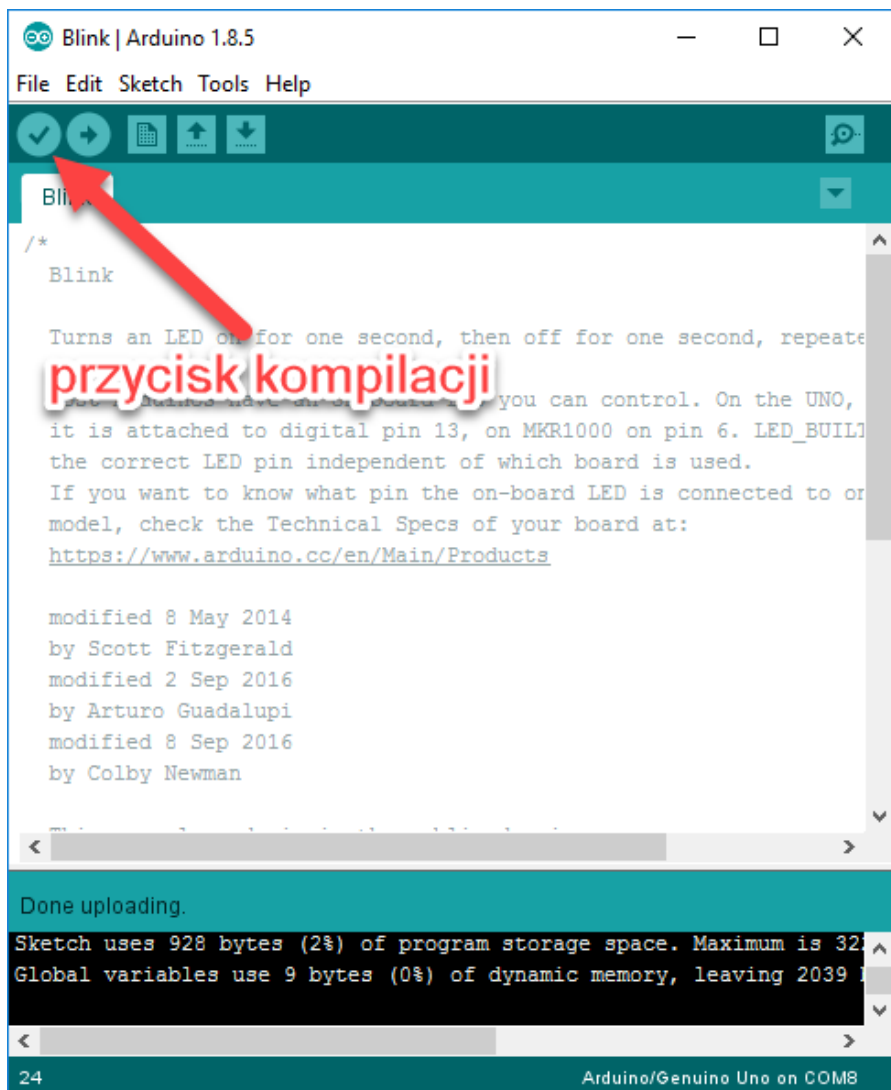


# Skompilujmy i załadujmy go do procesora...

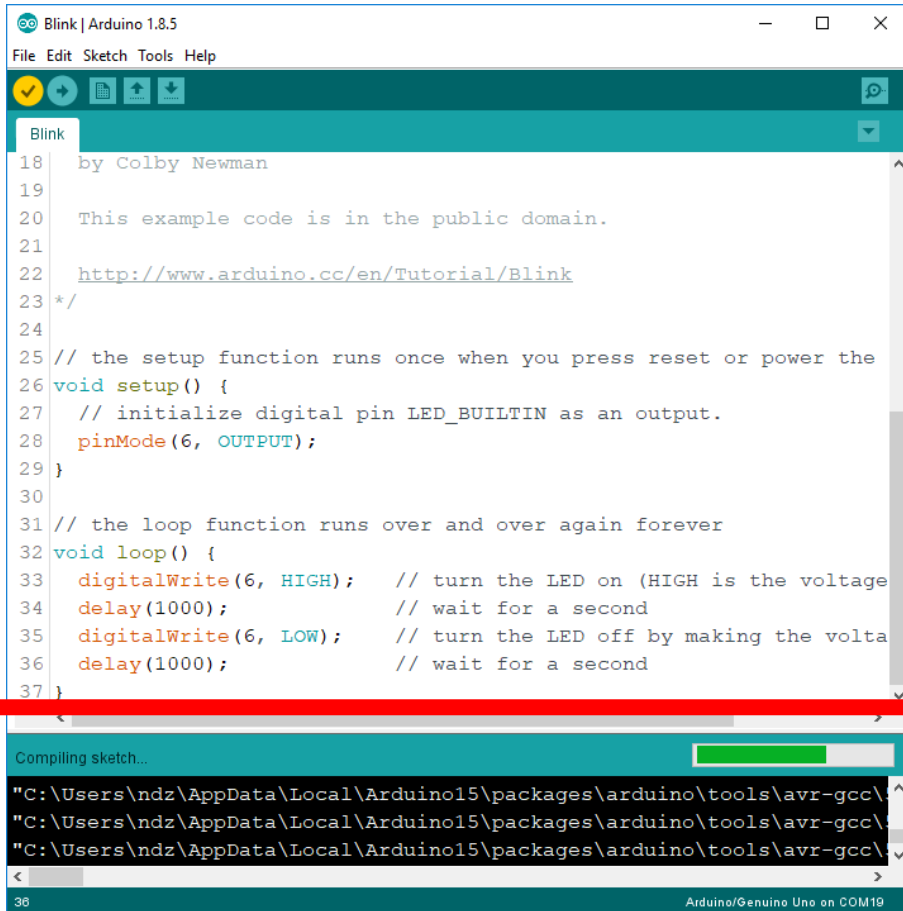


wybór płytki prototypowej i  
wybór portu komunikacji

# Skompilujmy i załadujmy go do procesora...



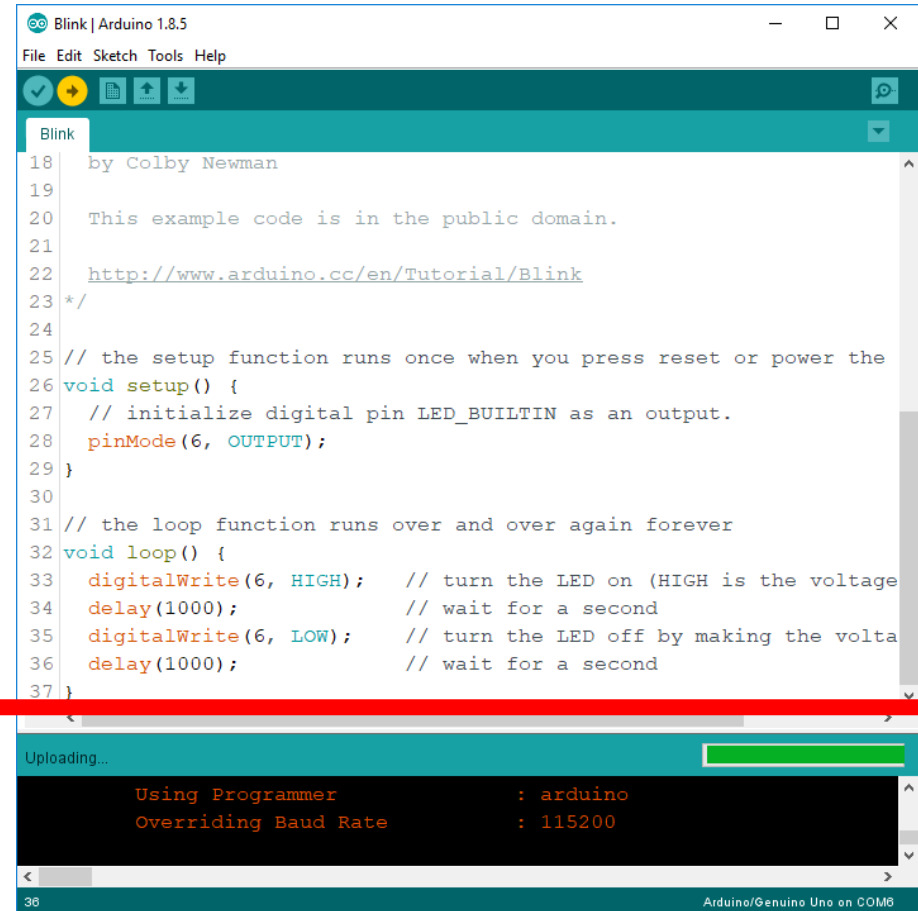
# Skompilujmy i załadujmy go do procesora...



```
18 by Colby Newman
19
20 This example code is in the public domain.
21
22 http://www.arduino.cc/en/Tutorial/Blink
23 */
24
25 // the setup function runs once when you press reset or power the
26 void setup() {
27   // initialize digital pin LED_BUILTIN as an output.
28   pinMode(6, OUTPUT);
29 }
30
31 // the loop function runs over and over again forever
32 void loop() {
33   digitalWrite(6, HIGH); // turn the LED on (HIGH is the voltage
34   delay(1000);           // wait for a second
35   digitalWrite(6, LOW);  // turn the LED off by making the volta
36   delay(1000);           // wait for a second
37 }
```

Compiling sketch...

"C:\Users\ndz\AppData\Local\Arduino15\packages\arduino\tools\avr-gcc\  
"C:\Users\ndz\AppData\Local\Arduino15\packages\arduino\tools\avr-gcc\  
"C:\Users\ndz\AppData\Local\Arduino15\packages\arduino\tools\avr-gcc\  
36 Arduino/Genuino Uno on COM19



```
18 by Colby Newman
19
20 This example code is in the public domain.
21
22 http://www.arduino.cc/en/Tutorial/Blink
23 */
24
25 // the setup function runs once when you press reset or power the
26 void setup() {
27   // initialize digital pin LED_BUILTIN as an output.
28   pinMode(6, OUTPUT);
29 }
30
31 // the loop function runs over and over again forever
32 void loop() {
33   digitalWrite(6, HIGH); // turn the LED on (HIGH is the voltage
34   delay(1000);           // wait for a second
35   digitalWrite(6, LOW);  // turn the LED off by making the volta
36   delay(1000);           // wait for a second
37 }
```

Uploading...

Using Programmer : arduino  
Overriding Baud Rate : 115200

36 Arduino/Genuino Uno on COM8

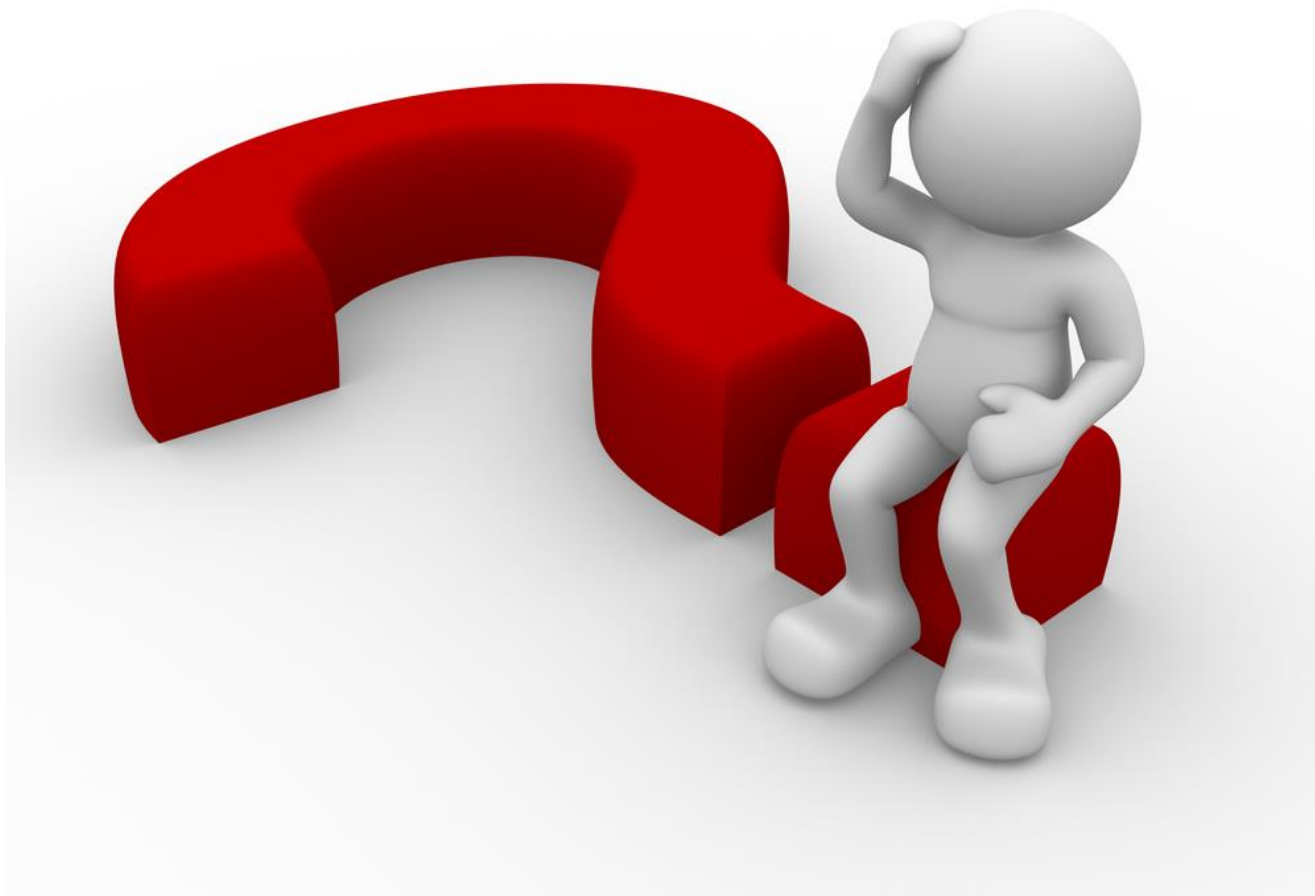
Sketch uses 930 bytes (2%) of program storage space. Maximum is 32256 bytes.

Global variables use 9 bytes (0%) of dynamic memory, leaving 2039 bytes for local variables. Maximum is 2048 bytes.

Skompilujmy i załadujmy go do procesora...



sprawdźmy pozostałe diody... eksperymentujmy...



O co chodzi w tym programowaniu ARDUINO?

# Podstawowa struktura programu

```
void setup() {  
    // Instrukcje, które zostaną wykonane  
    // tylko jeden raz – podczas  
    // inicjalizacji systemu  
}  
  
void loop() {  
    // Instrukcje, które będą wykonywane  
    // do czasu zaniku zasilania lub  
    // resetu (zerowania) systemu  
}
```



# Podstawowa struktura programu

```
// the setup function runs once when you press reset or power the board
void setup() {
    // initialize digital pin LED_BUILTIN as an output.
    pinMode(6, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
    // turn the LED on (HIGH is the voltage level)
    digitalWrite(6, HIGH);
    // wait for a second
    delay(1000);
    // turn the LED off by making the voltage LOW
    digitalWrite(6, LOW);
    // wait for a second
    delay(1000);
}
```

# Dyrektywa #define

```
#define LIGHTS_FRONT 6

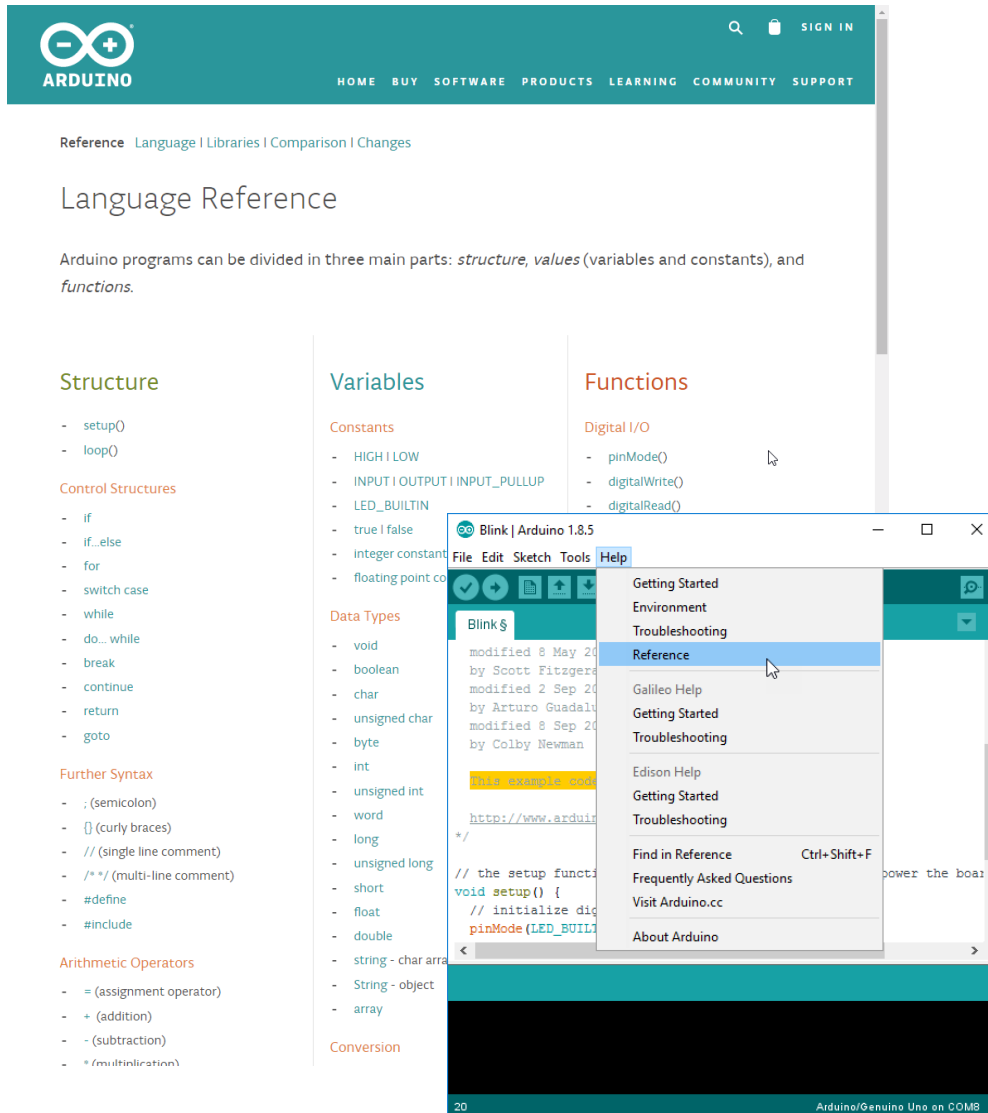
void setup() {
    //Konfiguracja linii 6 jako wyjście
    pinMode(LIGHTS_FRONT, OUTPUT);
}

void loop() {
    //Włączenie diody
    digitalWrite(LIGHTS_FRONT, HIGH);
    //Odczekanie 1 sekundy (1000 ms)
    delay(1000);

    //Wyłączenie diody
    digitalWrite(LIGHTS_FRONT, LOW);
    //Odczekanie 1 sekundy (1000 ms)
    delay(1000);
}
```

- Wykorzystajmy instrukcję #define w praktyce...
  - Zmiana numeru diody LED lub przycisku, którym chcemy sterować...

# Gdzie szukać pomocy i dodatkowych informacji?



Po angielsku:

- HELP -> REFERENCE
- [www.arduino.cc](http://www.arduino.cc) -> LEARNING ->
  - GETTING STARTED,
  - TUTORIALS,
  - REFERENCE,
  - EDUCATION.

Po polsku:

- forbot.pl
- książki (wyd. HELION)

# Gdzie / jak nas znaleźć...

Politechnika Łódzka

Instytut Elektroniki

Ul. Wólczańska 211/215

90-924 Łódź

<http://eletel.p.lodz.pl>

- Marcin Kociołek: [marcin.kociolek@p.lodz.pl](mailto:marcin.kociolek@p.lodz.pl)
- Paweł Poryzała: [pawel.poryzala@p.lodz.pl](mailto:pawel.poryzala@p.lodz.pl)