

Komputer pokładowy

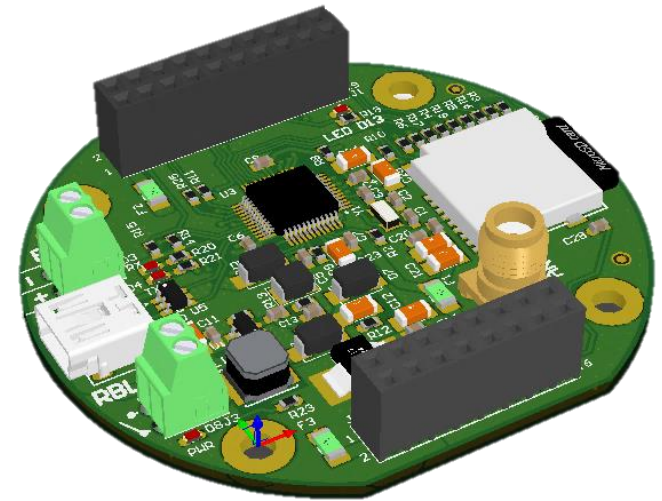


cansats in europe
2019 polish competition

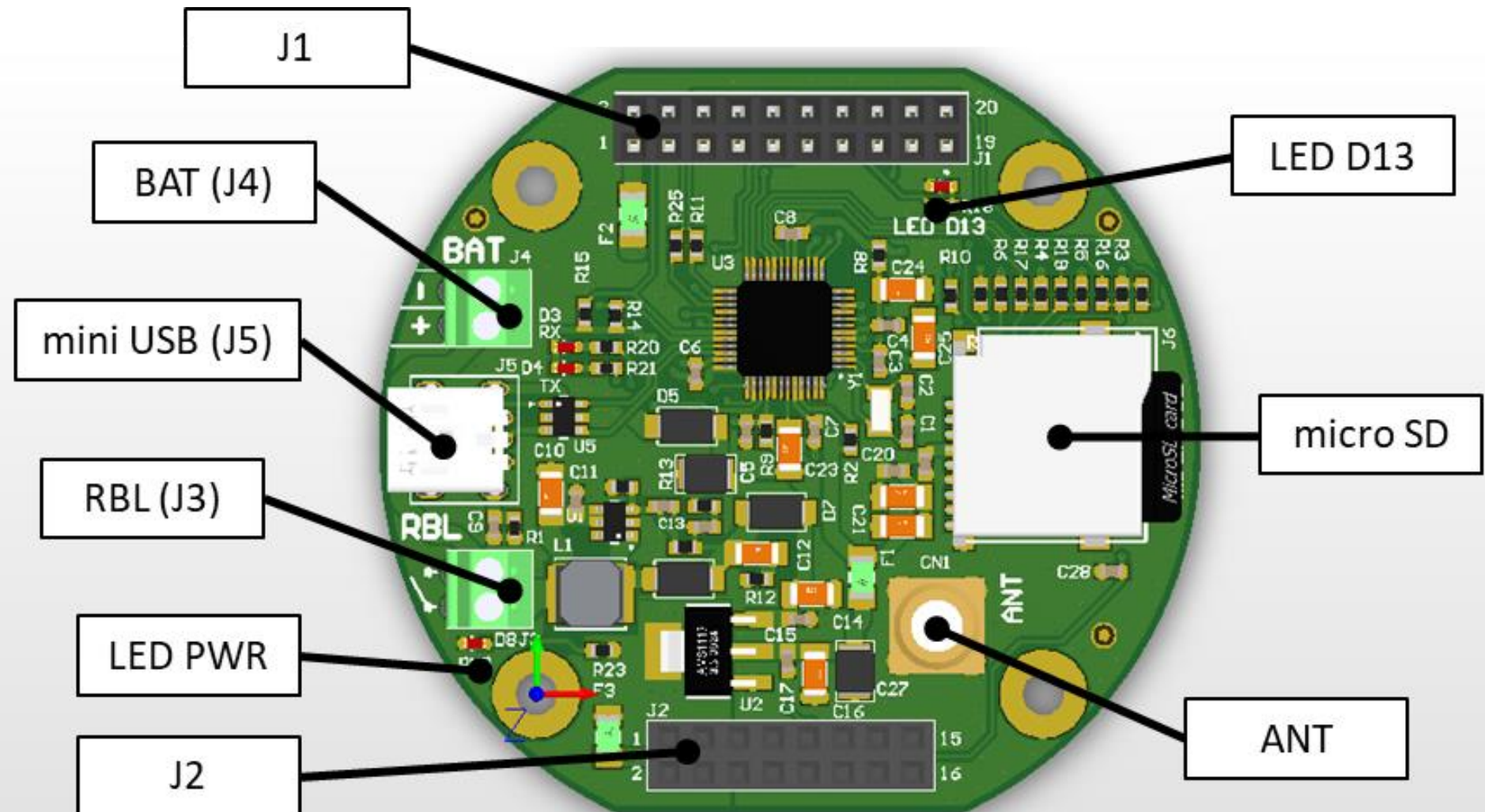


Komputer pokładowy

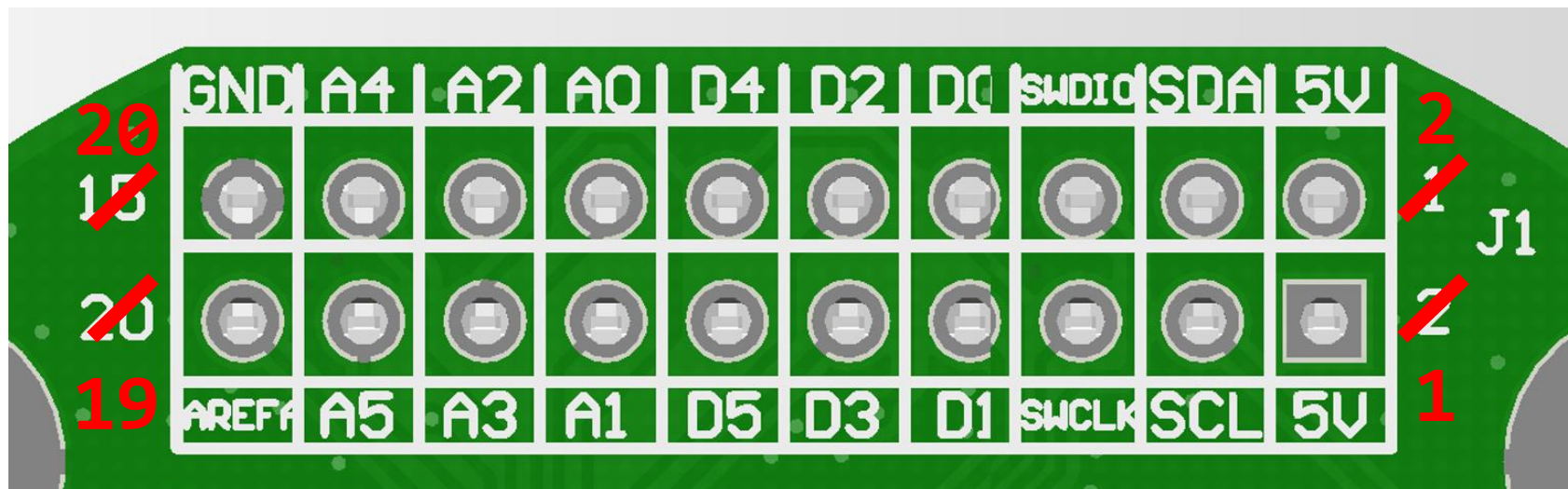
- **mikrokontroler Atmel ATSAM21G18**
- **kompatybilny z Arduino M0**
(lub podobnymi z SAM21D)
- **port mini USB**
programowanie + SerialUSB + zasilanie
- **wyprowadzone piny procesora**
(m.in. GPIO, ADC, DAC, SPI, I2C, UART)
- **wejścia/wyjścia mikrokontrolera w standardzie 3.3 V! Podłączenie wyższego napięcia może skutkować nieodwracalnym uszkodzeniem!**



Złącza komputera pokładowego



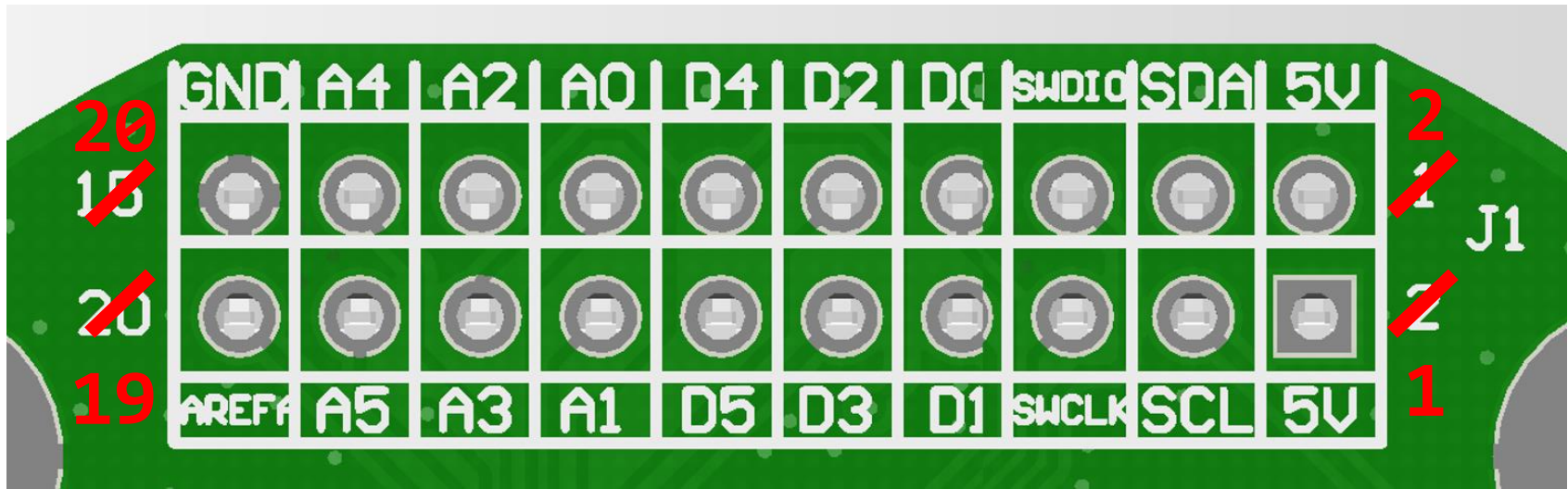
Złącza komputera pokładowego



UWAGA! – Wystąpił błąd w opisie złącza J1 od dolnej strony płytki! Szczegóły na następnym slajdzie lub w dokumentacji kitu!



Złącza komputera pokładowego - errata



Jak pokazano na powyższym obrazku piny złącza J1 zostały błędnie ponumerowane na opisie płytki od strony dolnej (bottom). Numeracja od strony górnej (top) jest prawidłowo oznaczona. Pole pinu nr 1 ma zawsze kształt kwadratowy.

Nazwy pinów np. GND, AREF, 5V itd. są prawidłowe i w prawidłowych miejscach!

np.:

pin nr 1 – 5 V

pin nr 3 – SCL

pin nr 19 – AREF

pin nr 20 – GND

Komputer pokładowy – porty

Wejścia/wyjścia cyfrowe

- włączanie/wyłączanie np. diod LED i innych urządzeń
- odczytywanie stanu wejścia – np. możliwe podłączenie przełączników

Wejścia analogowe

- pomiar napięcia (woltomierz): 0 – 3.3 V, rozdzielczość 0.81 mV
- czujniki analogowe (np. czujnik temperatury z zestawu)

Wyjścia analogowe (PWM lub DAC)

- generowanie napięcia, sygnału analogowego (np. sygnał dla brzęczyka)

Komputer pokładowy – porty

Serial port (UART)

- dwustronna komunikacja pomiędzy Arduino, a urządzeniami zewnętrznymi
- np. odbiorniki GPS, czujniki, komunikacja Arduino-komputer (SerialUSB)

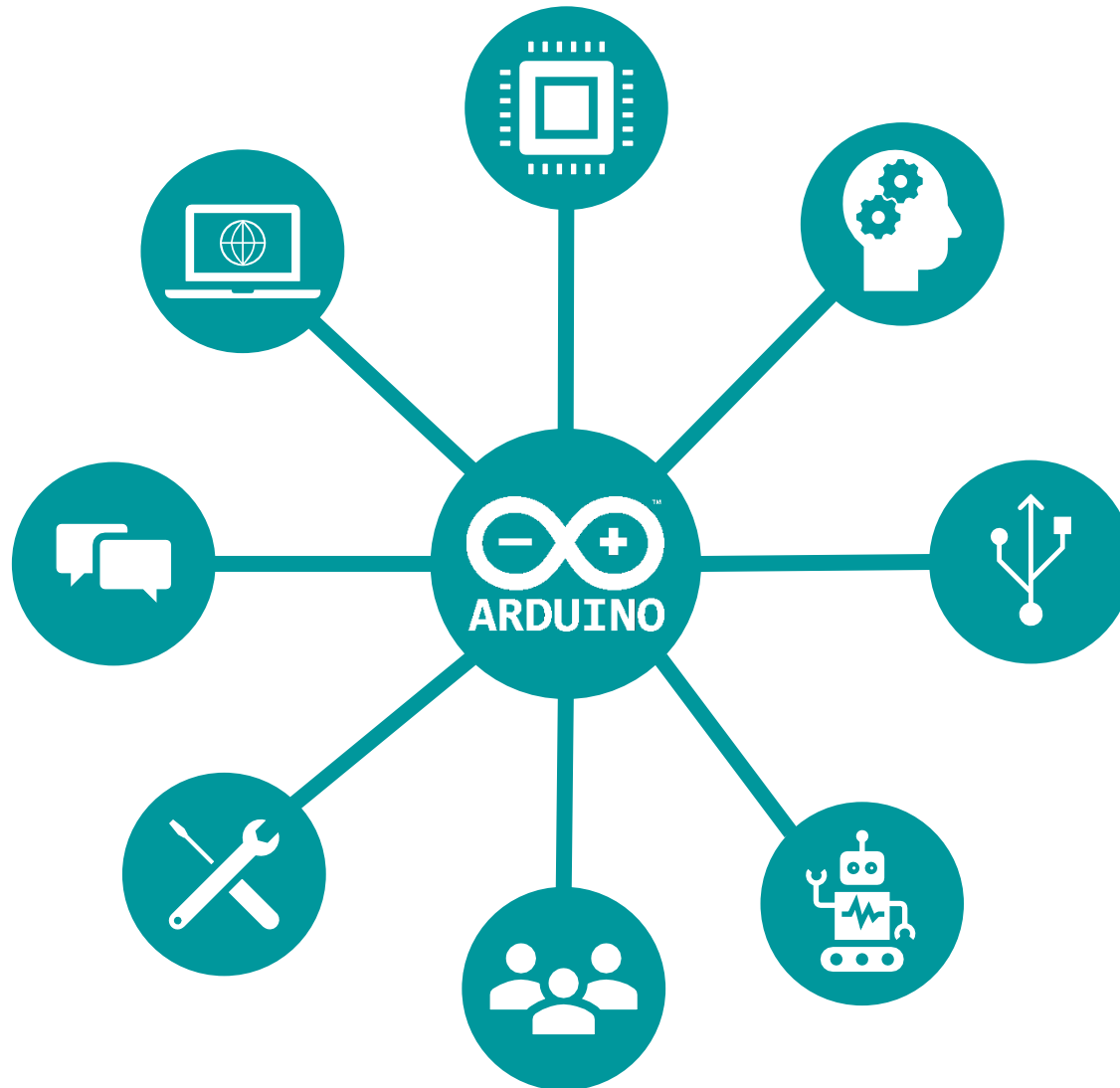
I2C

- cyfrowy interfejs komunikacyjny
- czujniki cyfrowe (np. czujnik ciśnienia), pamięci, itp.

SPI

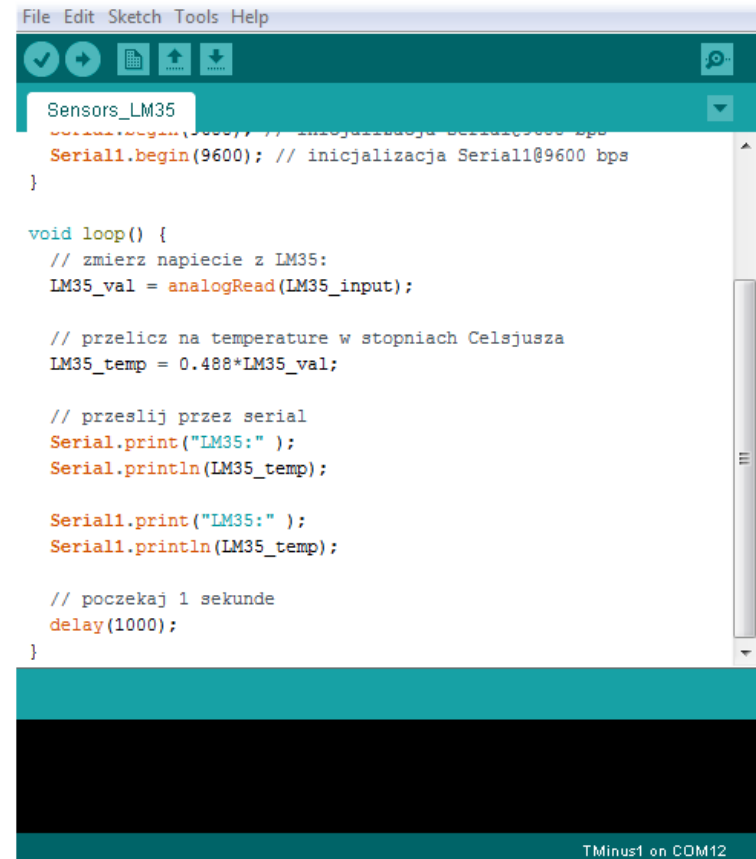
- cyfrowy interfejs komunikacyjny
- czujniki cyfrowe, pamięci (np. karta SD), moduły radiowe, itp.

Ekosystem Arduino



Programowanie – Arduino IDE

- **multiplatformowe środowisko**
(Windows, Linux, Mac OS)
- **programy w języku C++**
- **kompilator i programator**
- **biblioteki Arduino do obsługi płytek Arduino oraz urządzeń zewnętrznych**
- **wbudowana baza przykładów**
- **przydatne narzędzia:**
monitor portu szeregowego,
kreślarka (Serial Plotter)

The screenshot shows the Arduino IDE interface. At the top is a menu bar with 'File', 'Edit', 'Sketch', 'Tools', and 'Help'. Below the menu bar is a toolbar with icons for opening files, saving, and running. The main text area contains a C++ sketch named 'Sensors_LM35'. The sketch includes comments in Polish and code for initializing a serial port and reading an LM35 sensor. The status bar at the bottom indicates 'TMinus1 on COM12'.

```
File Edit Sketch Tools Help

Sensors_LM35

// Inicjalizacja Serialu 9600 bps
Serial1.begin(9600); // inicjalizacja Serial1@9600 bps
}

void loop() {
  // zmierz napięcie z LM35:
  LM35_val = analogRead(LM35_input);

  // przelicz na temperature w stopniach Celsjusza
  LM35_temp = 0.488*LM35_val;

  // przeslij przez serial
  Serial.print("LM35:");
  Serial.println(LM35_temp);

  Serial1.print("LM35:");
  Serial1.println(LM35_temp);

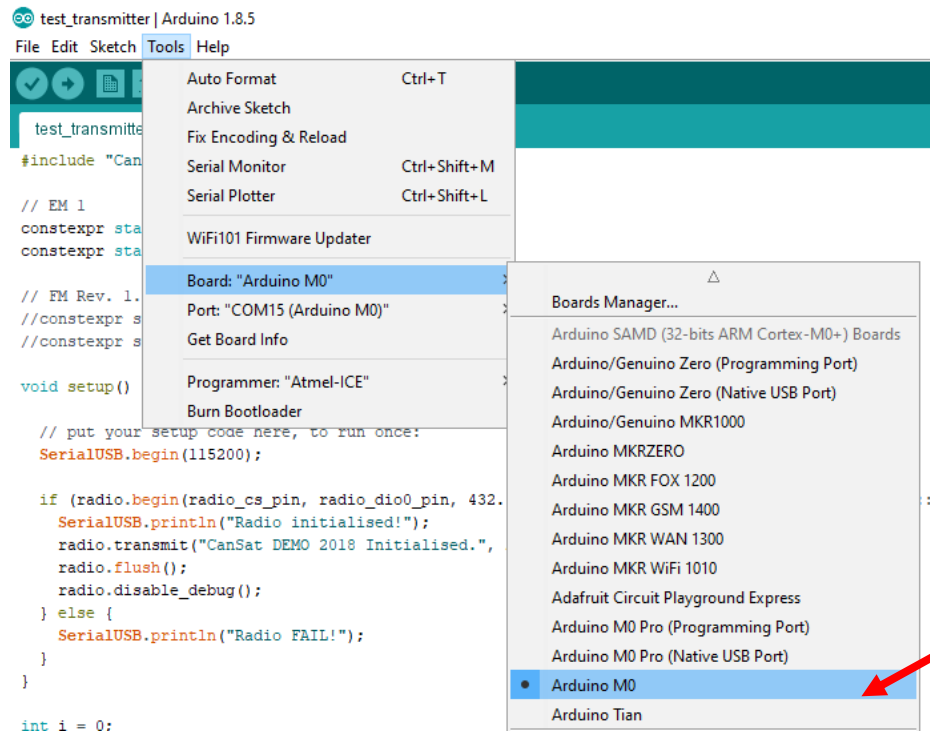
  // poczekaj 1 sekunde
  delay(1000);
}
```

TMinus1 on COM12

Uruchomienie płytki

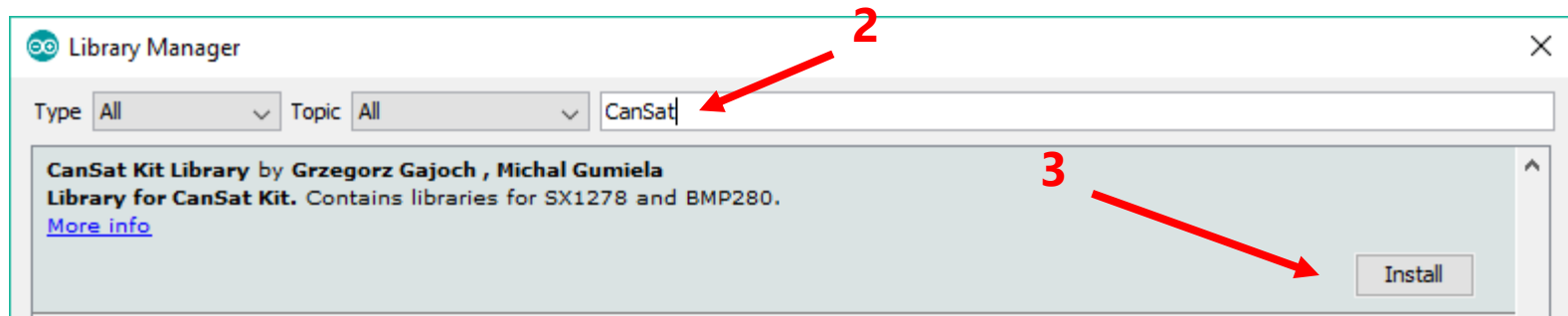
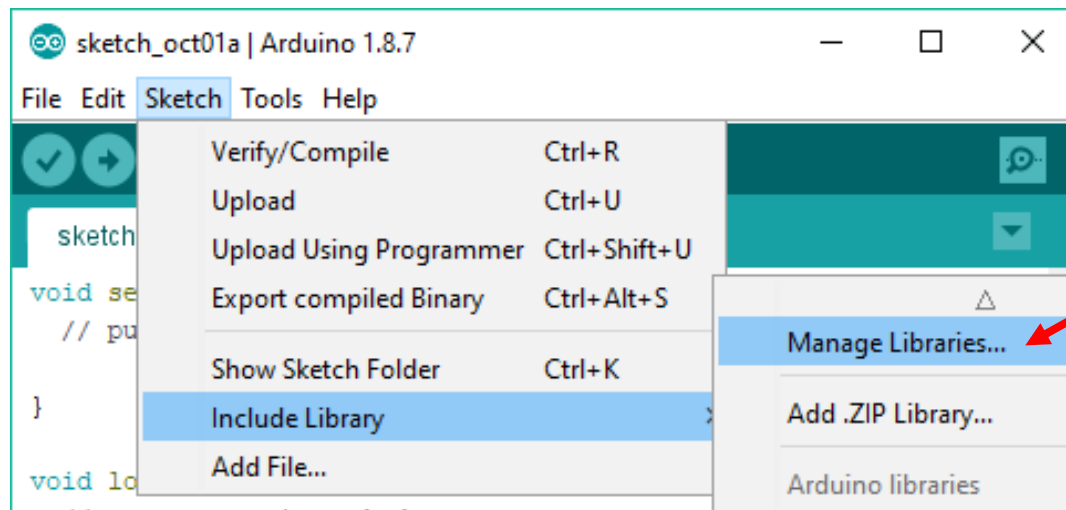
Cała procedura: rozdział 5. dokumentacji: Uruchomienie zestawu

1. Podłącz płytkę przez USB do komputera
2. Uruchom Arduino IDE
3. Wybierz model płytki: **Arduino M0**



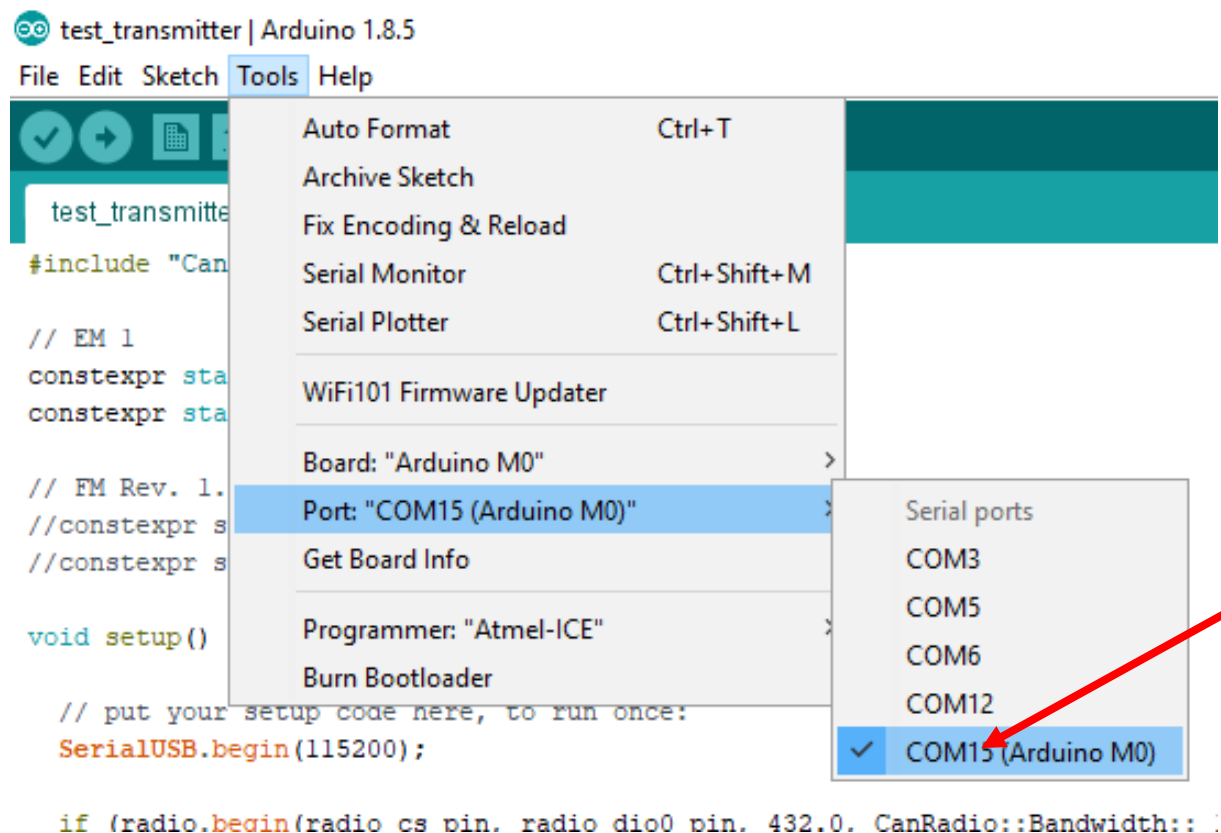
Uruchomienie – instalacja bibliotek

4. Instalacja bibliotek programistycznych CanSat Kit:



Uruchomienie płytki

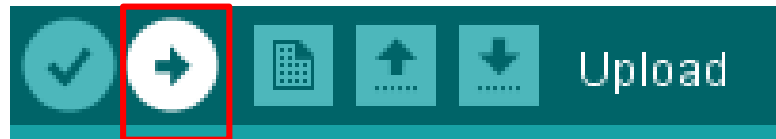
5. Wybierz port COM pod którym wykryta jest płytka:



UWAGA! Numer portu COM może się zmieniać po wgrywaniu programu, resetowaniu komputera!

Wgrywanie pierwszego programu

- Otwórz przykład „**HelloCanSat**” z menu „Plik -> Przykłady -> CanSat Kit Library”
- **skompiluj i wgraj** program do mikrokontrolera:



Compiling sketch...



Program HelloCanSat!

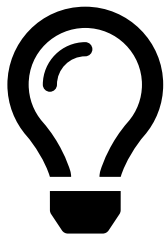


SerialUSB



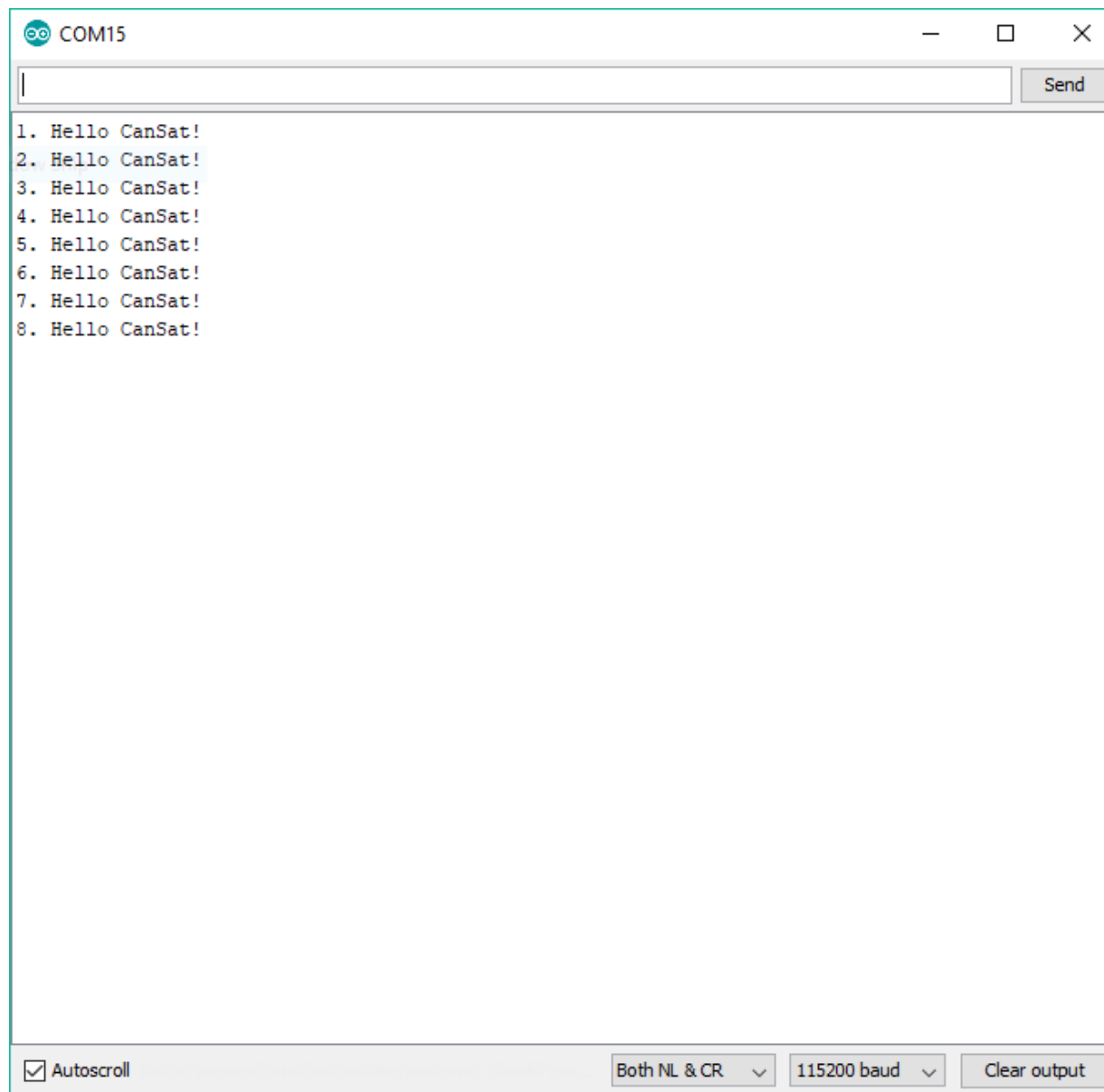
Serial Monitor

„x. Hello CanSat!”



**LED 13
(pin D13)**

Program HelloCanSat!



Podsumowanie

Gdzie jesteśmy?

- ✓ zainstalowane sterowniki do płytki
- ✓ skonfigurowane środowisko programistyczne (Arduino IDE)
- ✓ gotowa do pracy płytka komputera pokładowego