

MIKROKONTROLERY

Mikrokontroler jest to układ scalony, w którego strukturze zintegrowane są wszystkie elementy kompletnego komputera: jednostka centralna, pamięć oraz urządzenia peryferyjne. Można stwierdzić że jest to komputer w jednym układzie scalonym.

Uzasadnione jest określanie tego układu także nazwą „mikrokomputer jednoukładowy” (*one chip microcomputer*). Ta nazwa była stosowana w początkowej fazie rozwoju tych układów.

Mikrokontroler komunikuje się z otoczeniem za pośrednictwem wewnętrznych układów peryferyjnych.



Mikrokontroler ATmega 328 firmy Atmel

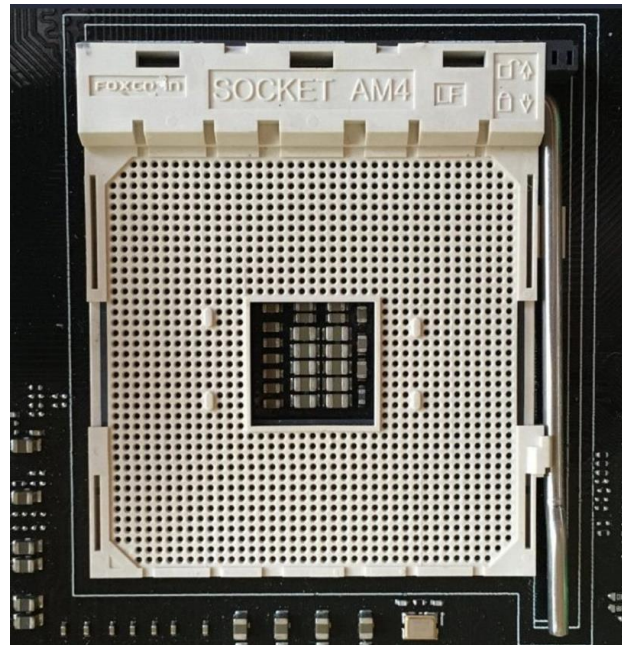


**Gniazdo do
mocowania
mikrokontrolera**

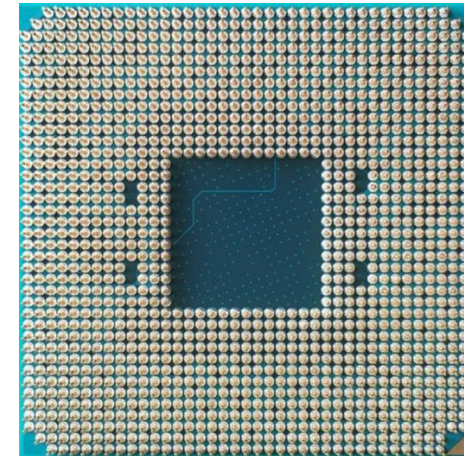
Mikroprocesor jest to układ scalony, w którego strukturze zintegrowana jest kompletna jednostka centralna komputera. **Jednostka centralna CPU (Central Processing Unit)** realizuje funkcje przetwarzania danych i steruje działaniem komputera



Przykładowe mikroprocesory



Gniazdo AM4 mikroprocesora (1331 pinów)



Procesor firmy AMD mocowany w gnieździe jak obok (widok od dołu od strony pinów)

mikrokontroler a mikroprocesor

Najważniejszą różnicą pomiędzy mikrokontrolerem a mikroprocesorem jest sposób rozmieszczenia różnych rodzajów pamięci.

W mikrokontrolerze pamięci współtworzą jego strukturę wewnętrzną.

W mikroprocesorze są one oddzielnymi podzespołami znajdującymi się na zewnątrz.

Mikrokontrolery odznaczają się znacznie mniejszym poborem energii oraz potrzebują do pracy mniejszej ilości elementów zewnętrznych, co znacznie obniża koszty ich zastosowania.

mikrokontroler a mikroprocesor (cd.1)

Zestawy instrukcji i rozkazów mikrokontrolerów są bardziej ograniczone i mniej rozbudowane, dzięki czemu z powodzeniem znajdują zastosowanie w wielu aplikacjach.

Mikroprocesory wykonują zwykle bardzo złożone zestawy instrukcji i rozkazów co wymaga udziału większej ilości pamięci.

Moduły pamięci mikroprocesora z powodu swego fizycznego rozmiaru stanowią osobne elementy i nie stanowią części składowej struktury wewnętrznej mikroprocesora.

mikrokontroler a mikroprocesor (cd.2)

Mikroprocesory ukierunkowane są na zastosowanie w technice obliczeniowej. Najbardziej masowo stosowane są w komputerach osobistych.

Mikroprocesory mają zapewnić możliwie dużą szybkość przetwarzania danych, a w ogólności posiadać dużą zdolność do przetwarzania danych, czyli dużą moc obliczeniową.

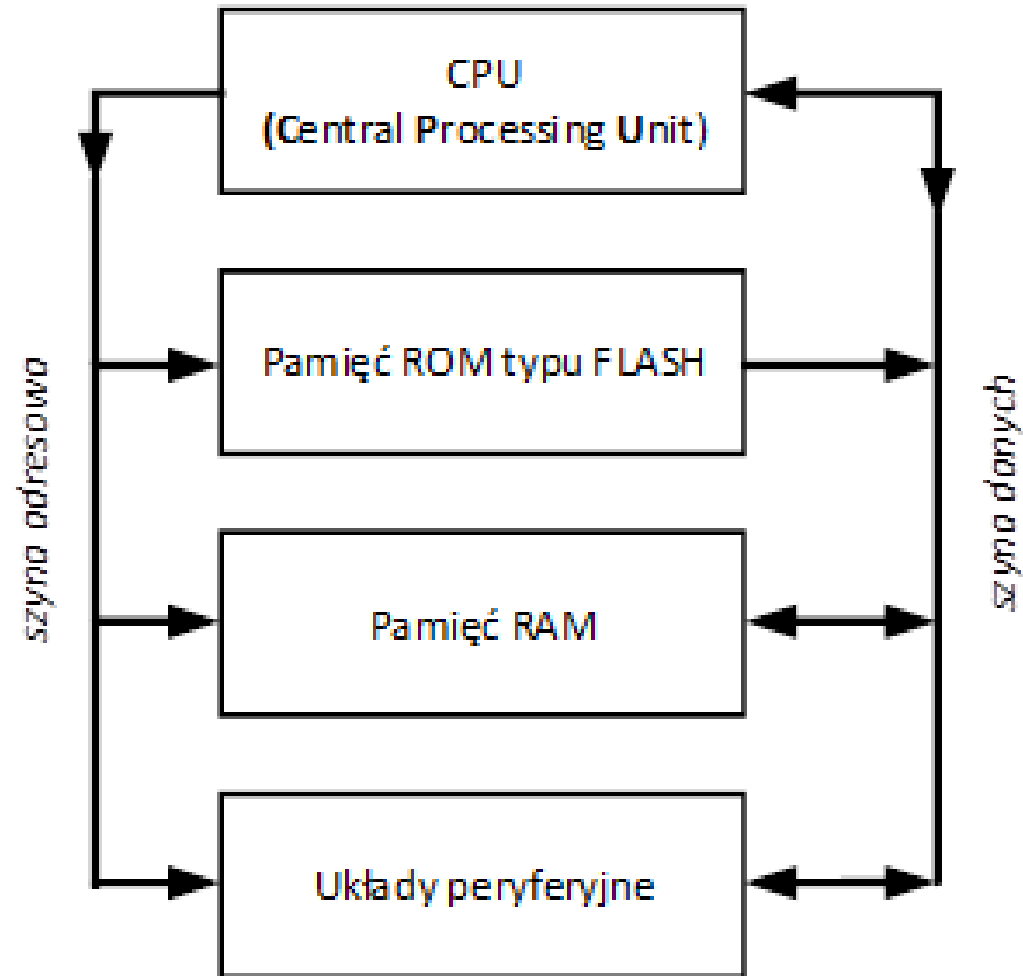
Głównym ich atutem mikrokontrolerów jest uniwersalność i łatwość implementacji, a sama wydajność pozostaje na drugim miejscu.

Muszą one z reguły pracować w czasie rzeczywistym, a główną dziedziną ich zastosowań są systemy kontrolno - pomiarowe, systemy sterowania oraz komunikacyjne, a zatem systemy w których występuje bezpośrednio sterowanie cyfrowe.

Mikrokontrolery pracują najczęściej w systemach czasu rzeczywistego

Systemy czasu rzeczywistego (real time systems) są to systemy komputerowe, w których obliczenia przeprowadzane są równoległe z przebiegiem zewnętrznego sterowanego procesu, a działanie systemu ma na celu terminowe reagowanie na zachodzące w procesie zdarzenia.

Uproszczona schematyczna budowa mikrokontrolera

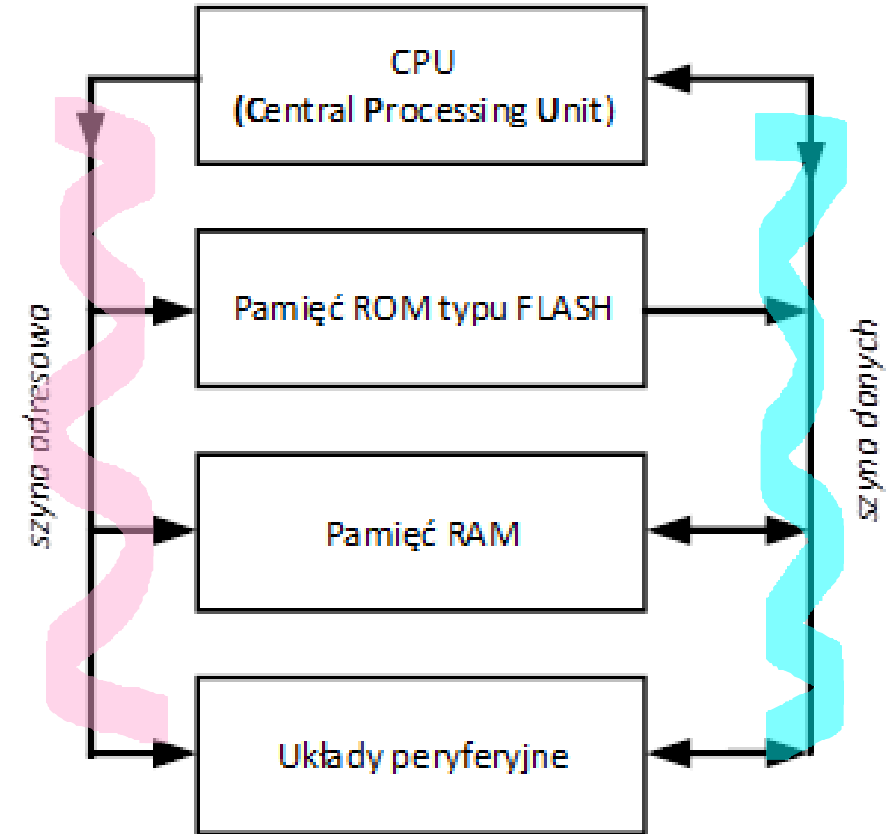


Szyna adresowa

Komunikacja na szynie adresowej zachodzi tylko w jedną stronę tj. od CPU do układu docelowego.

Szyna danych

Informacje na szynie danych płyną w obu kierunkach

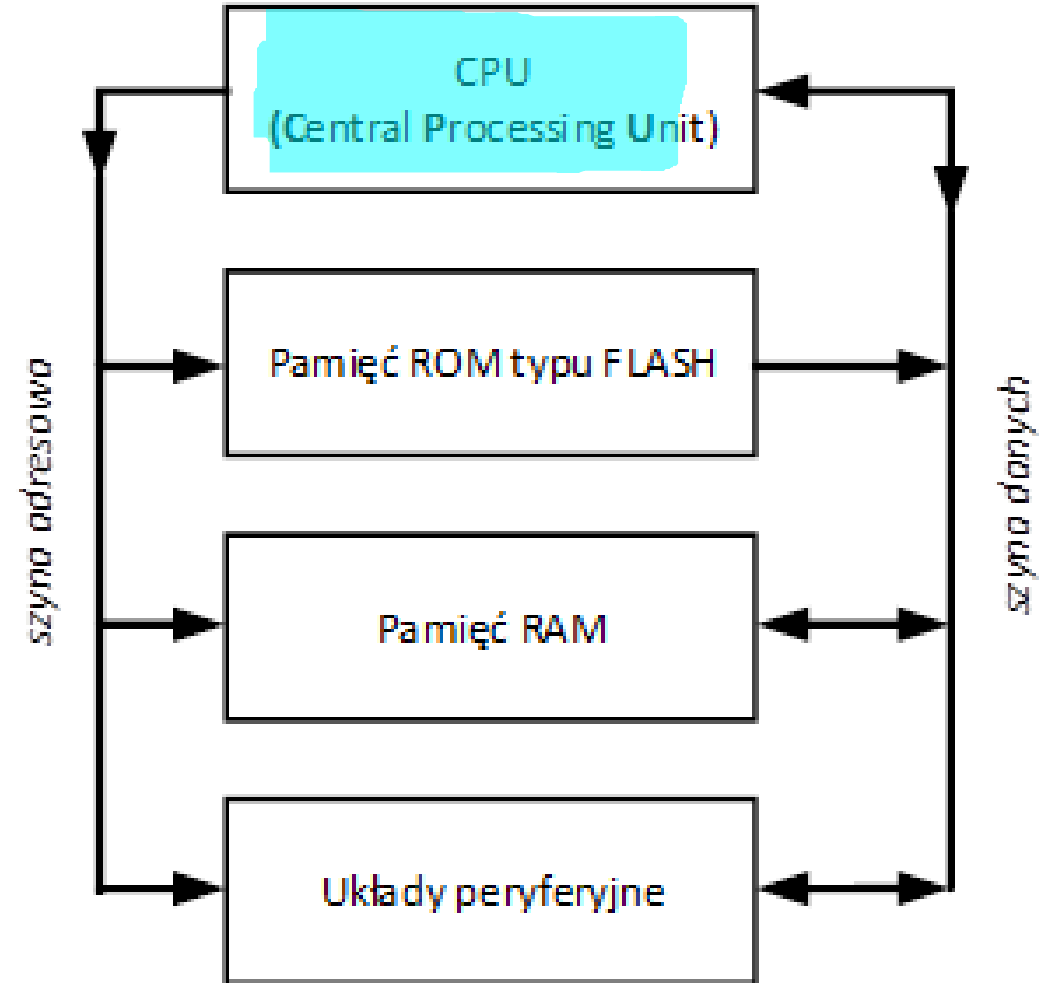


Procesor (CPU)

Zadaniem procesora CPU (Central Processing Unit) jest realizacja zapisanego w pamięci FLASH programu.

Program podzielony jest na rozkazy, które kolejno są przesyłane do CPU i wykonywane. W pojedynczym rozkazie zawarte są informacje o rodzaju operacji oraz o argumentach, na jakich ma ona zostać wykonana.

Zmienne w trakcie realizacji programu zapisywane są w pamięci RAM.



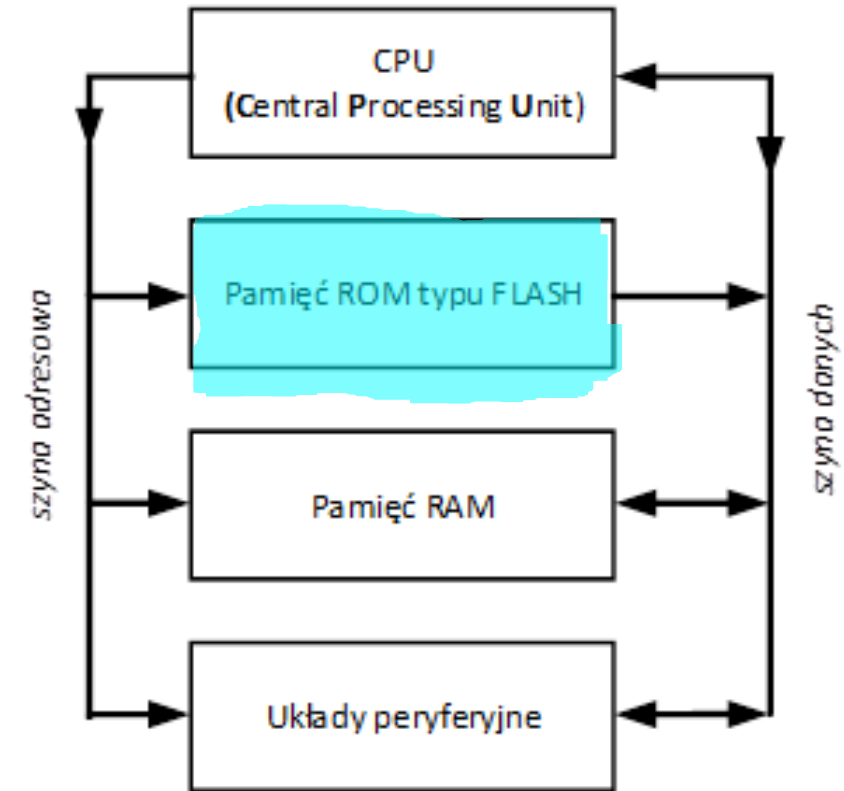
Pamięć ROM typu FLASH

Pamięć ta zwana jest pamięcią programu.

Jest to pamięć nieulotna (*tzn. że po odłączeniu zasilania zachowuje swoją zawartość*) o największej pojemności, służąca głównie do przechowywania rozkazów.

Informacje z tej pamięci dostarczane są na szynę danych jednokierunkowo. Spowodowane jest to tym, że CPU nie może na niej bezpośrednio zapisywać informacji a jedynie pobierać.

Oprócz programu w pamięci FLASH mogą się na niej znajdować tablice stałych oraz sekcja bootloader'a, umożliwiająca wgrywanie nowego programu bez użycia programatora.

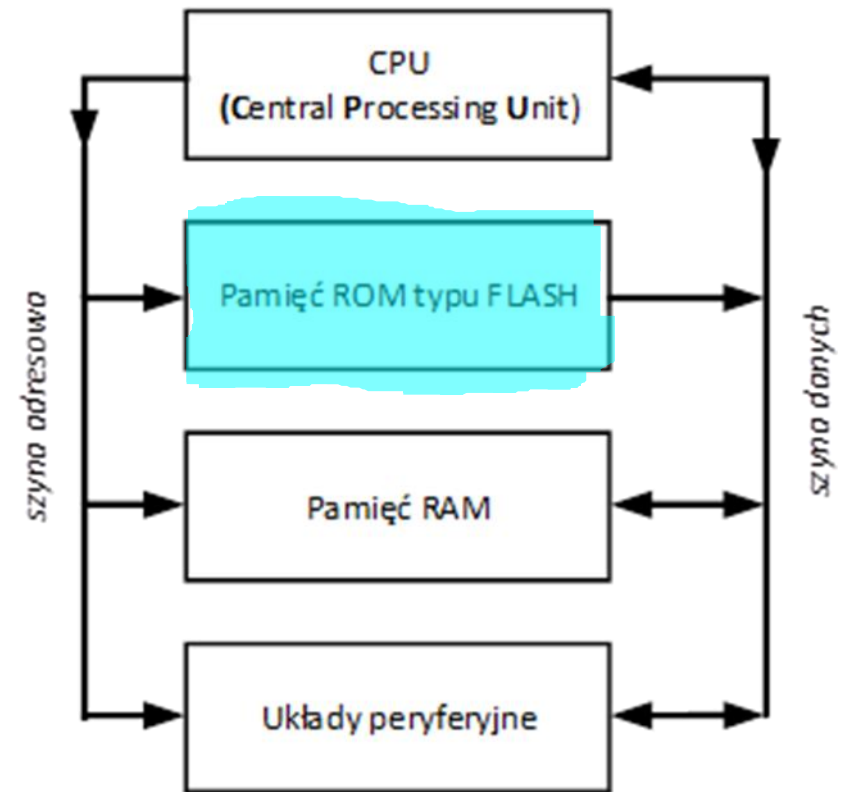


Pamięć ROM typu FLASH cd.

Pamięć FLASH jest podzielona na komórki o określonej pojemności, zależnej od używanego mikrokontrolera.

W mikrokontrolerach AVR to 8 bitów. Każda komórka ma przypisany adres w kodzie szesnastkowym.

Pojemność tej pamięci liczona w kilobajtach (maksymalna przestrzeń programu od 1 kB do 32 kB). Ze względu na tak ograniczoną przestrzeń pamięciową programy działające na pojedynczym układzie nie mogą być zbyt rozbudowane.



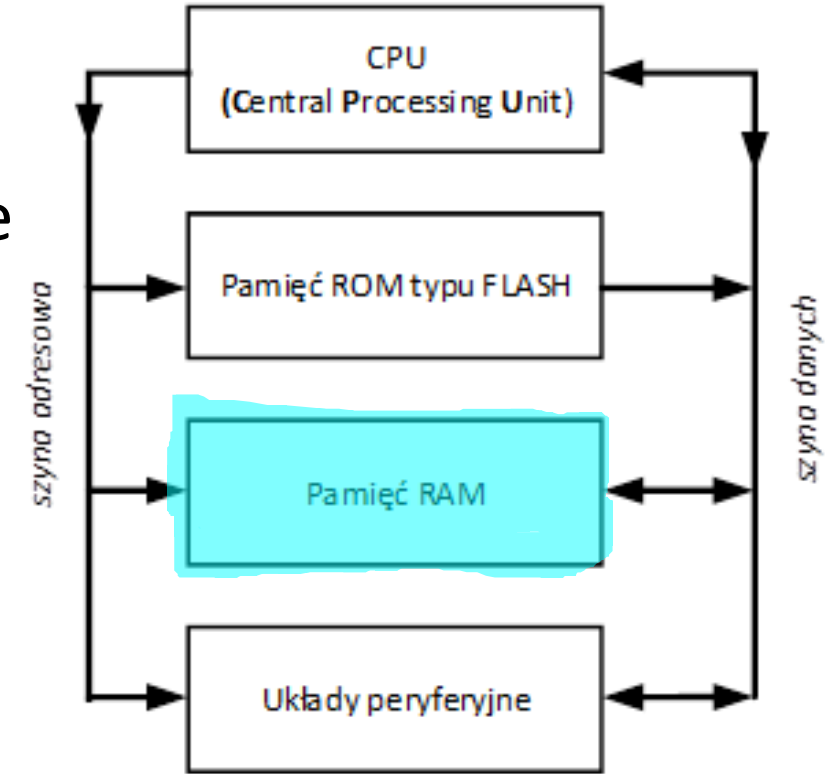
Pamięć RAM

Pamięć RAM jest pamięcią ulotną i przechowuje dane tylko wtedy kiedy układ pracuje (jest włączone zasilanie).

Wykorzystywana jest do przechowywania zmiennych w trakcie realizacji programu.

Dodatkowo jest używana jako stos sprzętowy.

Ostatni dostarczony bajt znajduje się na samym szczycie stosu, więc jest pierwszy do zdjęcia. Jeżeli dostarczony na stos będzie kolejny bajt, to przesunie poprzedni i to on będzie pierwszy. Stos zwykle jest umieszczony na końcu pamięci i rośnie w stronę początku.



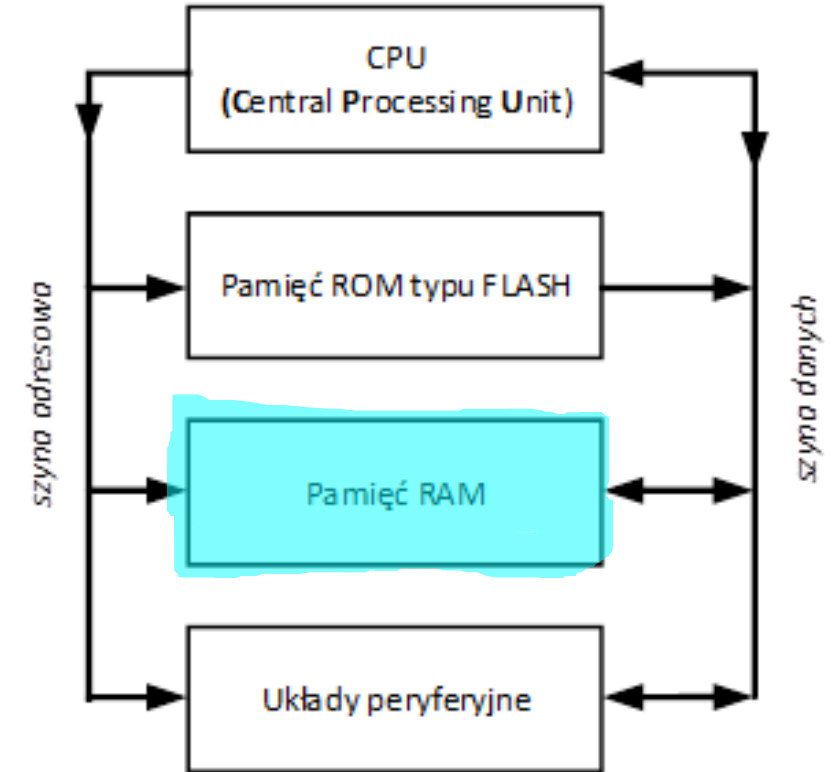
Pamięć RAM cd.

Mikrokontrolery mają też ograniczoną ilość pamięci RAM.

Np. układy z serii ATmega wyposażone są w zaledwie 1 kB pamięci.

Nie stanowi to przeszkody w zastosowaniu tych układów albowiem typowa aplikacja dla mikrokontrolera pobiera z wejścia strumień danych, względnie szybko te dane uwzględnia w realizacji programu i jak najszybciej wysyła dalej, bez stosowania wielkich buforów.

W przypadkach szczególnych można dołączyć do układu dodatkową pamięć RAM i w ten sposób obejść ograniczenie.

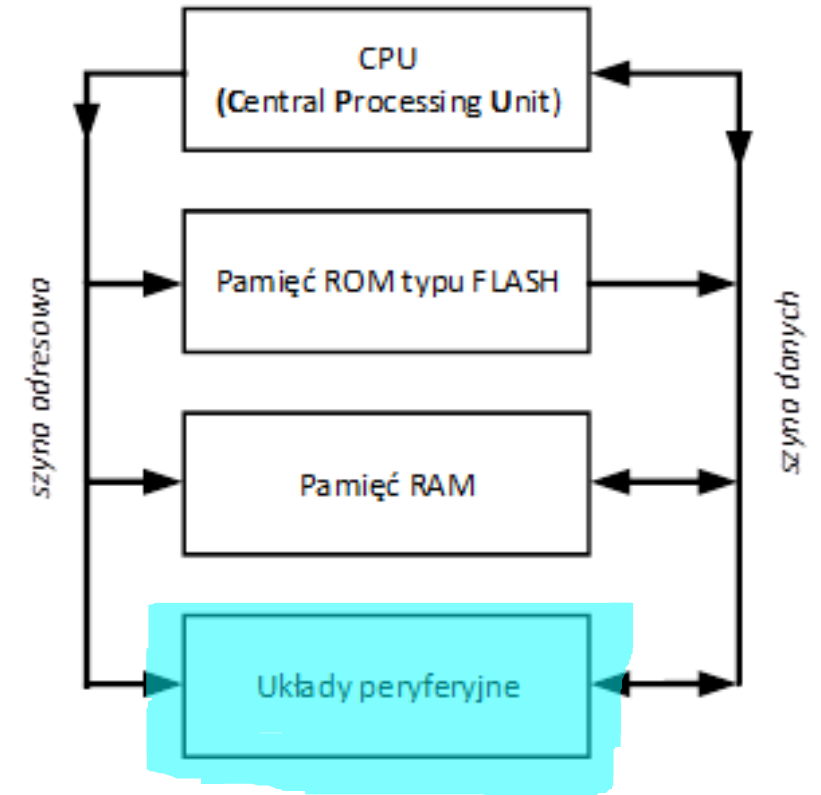


Układy peryferyjne

Układy peryferyjne służą do połączenia mikrokontrolera z otoczeniem.

W dalszych slajdach omówiono układy peryferyjne dla sprzętu Arduino Uno.

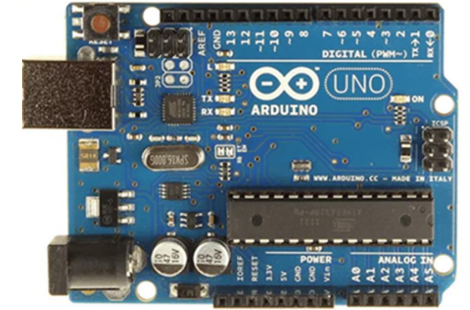
Arduino Uno to płytką z mikroprocesorem ATmega 328 firmy Atmel wraz z układami i wyprowadzeniami towarzyszącymi.





**

ARDUINO



Arduino to otwarta platforma elektronicznego projektowania oparta na elastycznym, łatwym w obsłudze sprzęcie i oprogramowaniu.

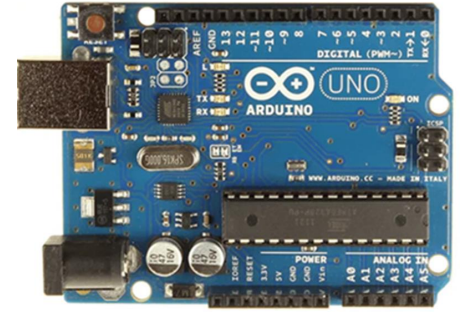
Sprzęt Arduino stanowi płytka z mikroprocesorem oraz pinami wejścia/wyjścia służącymi do komunikacji i sterowania obiektami fizycznymi.

Płytką jest zasilana przez złącze USB lub zewnętrzny zasilacz, co umożliwia z kolei zasilanie innego sprzętu do niej podłączanego.

Zintegrowane środowisko programistyczne Arduino (IDE) umożliwia pisanie kodu programu, kompilowanie go, a następnie przesyłanie go na płytkę Arduino w celu samodzielnego wykorzystania w realizowanych projektach.

Programy są pisane w języku podobnym do C/C++.

Koncepcja platformy Arduino została opracowana tak, aby niezbyt zaawansowanemu użytkownikowi pozwolić swobodnie rozwijać swoje pomysły, które w końcowej fazie otrzymują konkretne możliwości realizacyjne.



Trochę historii ARDUINO

Początki rozwoju Arduino sięgają roku 2003. Powstało jako prototyp platformy edukacyjnej do nauki programowania i elektroniki z użyciem mikrokontrolerów. Protoplastą pierwszych płytek Arduino, była platforma Wiring, zaprojektowana przez Hernando Barragána (proca magisterska).

Na Politechnice w Massachusetts, Benjamin Fry i Casey Rears tworzą środowisko Processing przeznaczone do programowania płytki Wiring. Stanowi to podstawę pod późniejsze Arduino IDE (zintegrowane środowisko programistyczne).

Następnie, zespół projektantów z Instytutu Projektowania Interakcji w Ivrei we Włoszech, pod kierownictwem Massimo Banziego (promotora Barragána) podjął prace nad przeprojektowaniem platformy Wiring tak, aby umożliwiała proste zaprogramowanie mikrokontrolerów i aby mogła funkcjonować bez pomocy zewnętrznego komputera oraz była dostępna w przystępnej cenie. Kolejne prototypy zostały wyposażone w port USB, zastępując w ten sposób port szeregowy RS-232, który był trudniejszy w obsłudze i mniej praktyczny.

Osiągnięto sukces i dużą popularność rozwiązania.



Płytką Arduino składa się z 8-bitowego mikrokontrolera Atmel AVR z uzupełniającymi elementami w celu ułatwienia programowania oraz włączenia innych układów. Większość płyt zawiera 5V stabilizator napięcia, 16 MHz rezonator. Mikrokontroler płytki Arduino jest wstępnie zaprogramowany z wykorzystaniem programu rozruchowego, co upraszcza przesyłanie programu do pamięci flash układu.

Uwaga

Oznaczenie AVR przy nazwie mikrokontrolera jest symbolem firmowym, którego znaczenie nie sposób odgadnąć.

Specyfikacja płytki Arduino UNO

Zasilanie:

- złącze koncentryczne 5,5/2,1 mm – napięcie od 7 – 12 V,
- złącze USB typu B – napięcie 5V.

Mikrokontroler: ATmega328P w obudowie DIP-28:

- pamięć SRAM: 2 kB,
- pamięć Flash: 32 kB (0,5 kB zarezerwowane dla bootloadera),
- pamięć EEPROM: 1 kB,
- częstotliwość zegara: 16 MHz.

20 uniwersalnych wyprowadzeń wejść/wyjść:

- 14 cyfrowych wejść/wyjść (w tym 6 z możliwością generowania 8-bitowego sygnału PWM),
- 6 wejść analogowych o rozdzielczości 10 bitów (mogących pełnić funkcję cyfrowych wejść/wyjść).

Interfejsy komunikacyjne: UART, I2C, SPI.

Zewnętrzne przerwania: 2.

Maksymalny prąd:

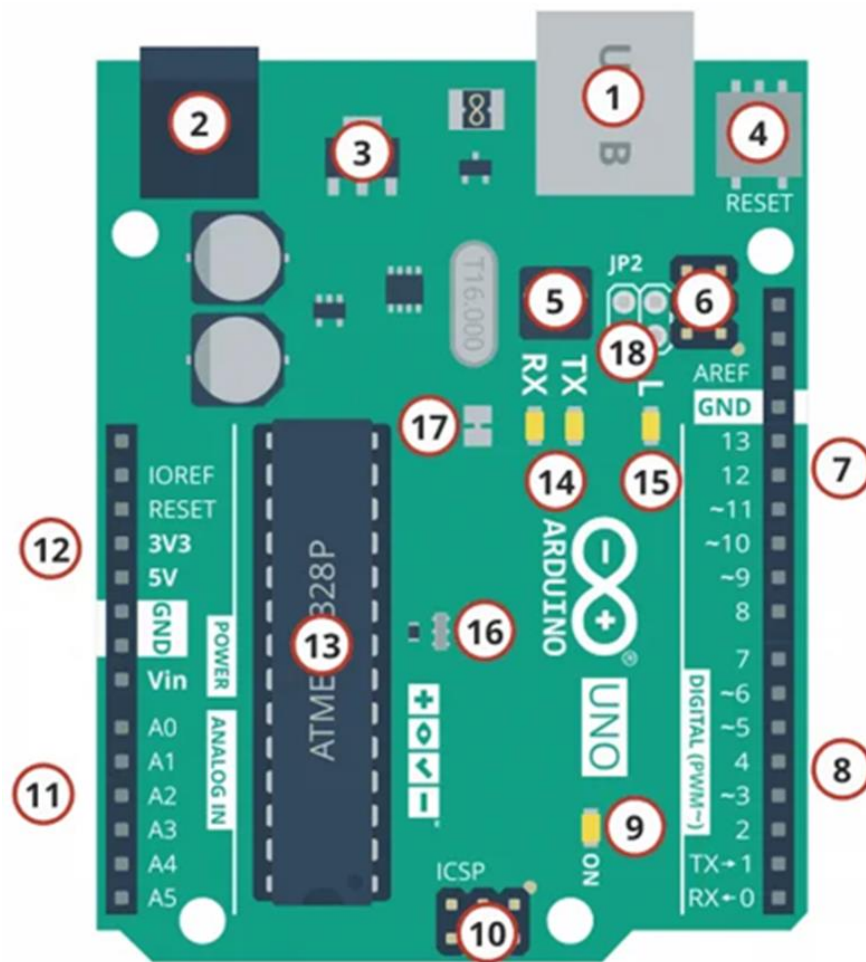
- dla wyjścia 5 V: 500 mA,
- dla wyjścia 3,3 V: 50 mA,

dla poszczególnych GPIO: 20 mA *GPIO (ang. general-purpose input/output), czyli uniwersalne wejścia/wyjścia*

Rozmiary płytki: 68,6 × 53,4 mm.



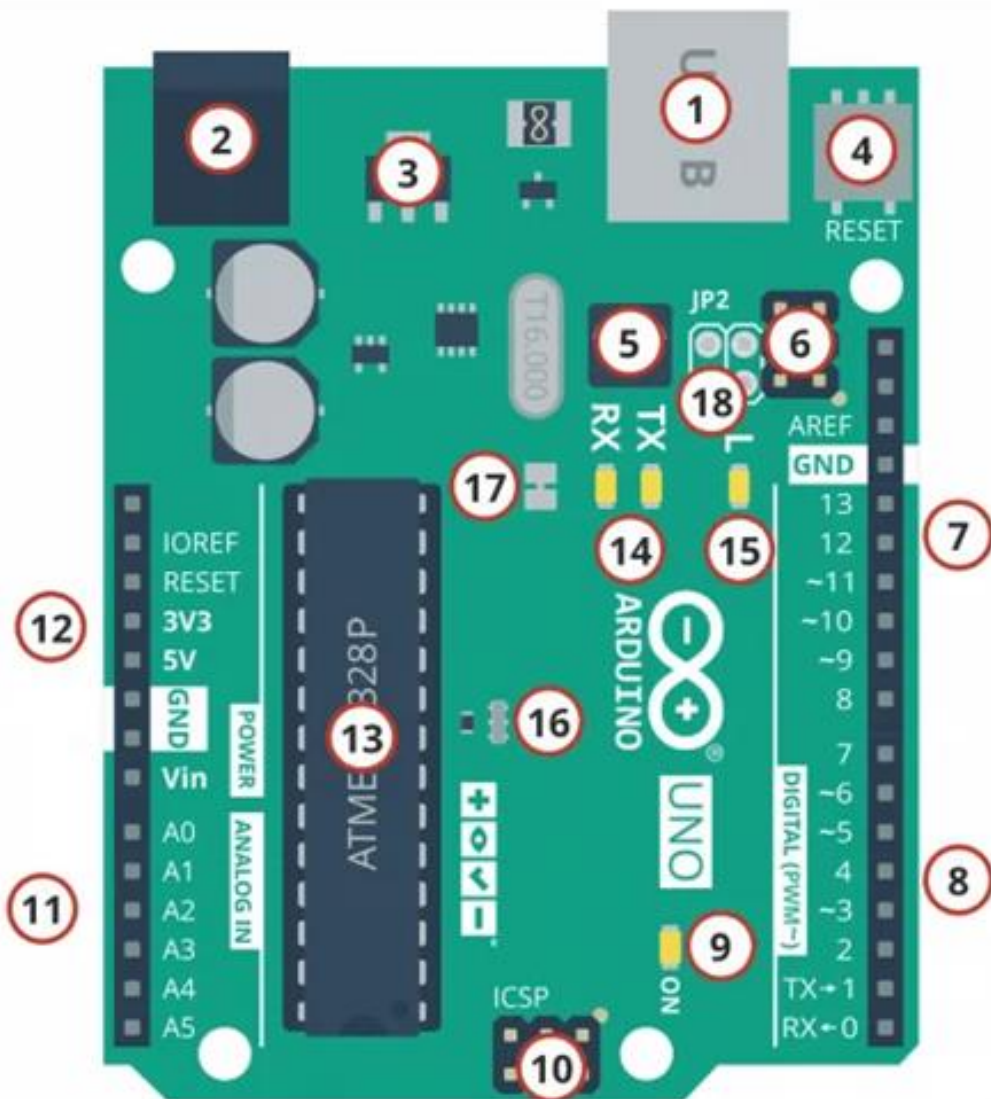
płytko Arduino



[*]

płytko Arduino z zaznaczeniem elementów budowy

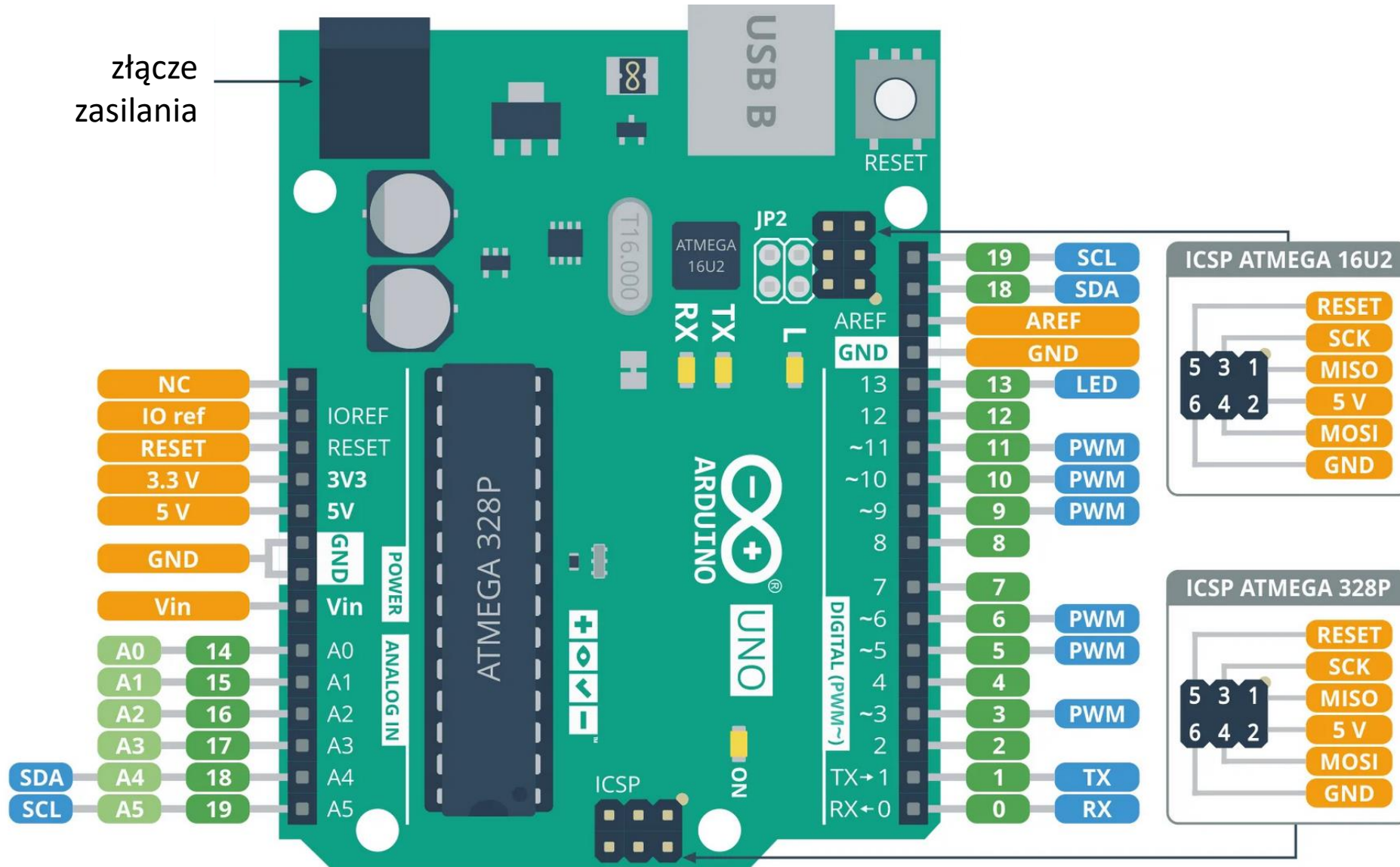
elementy umieszczone na płytce Arduino



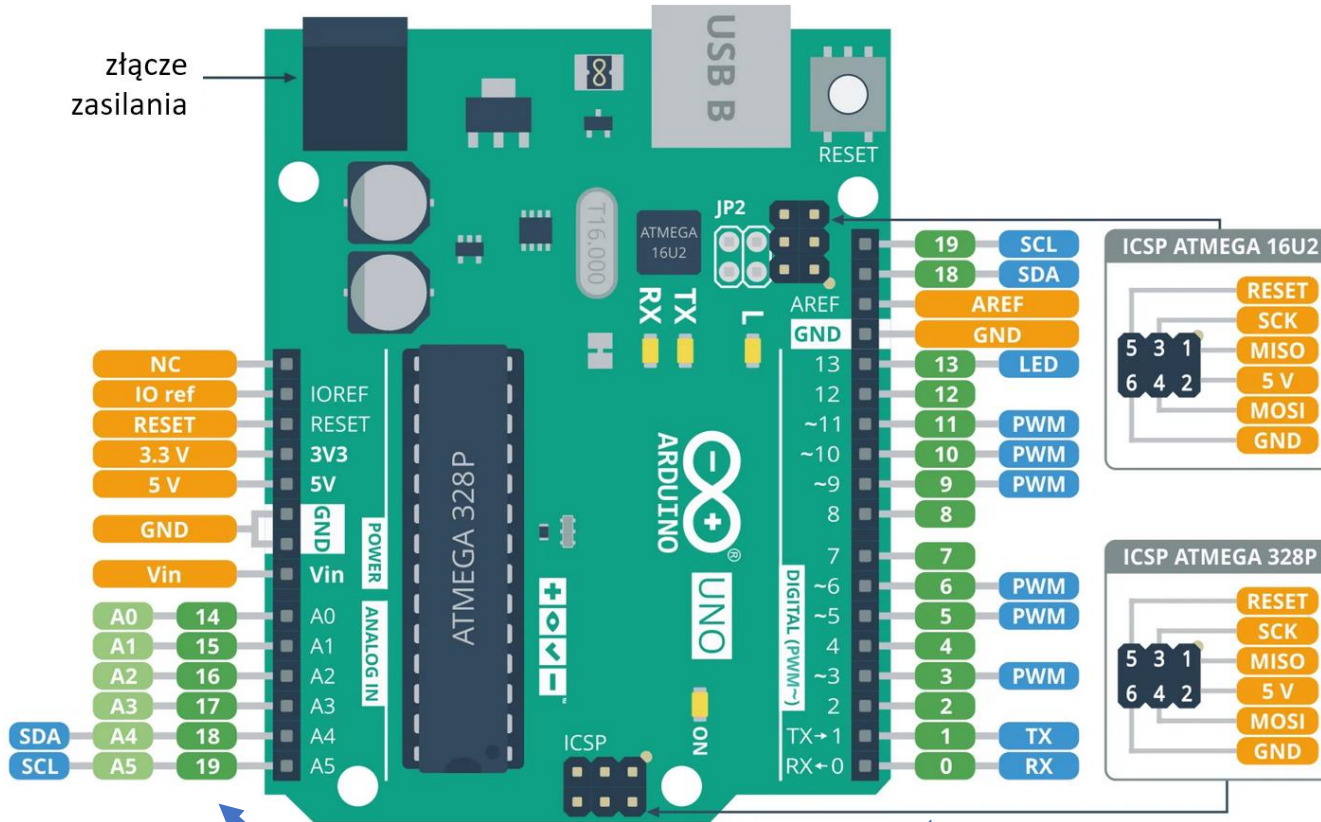
1. Złącze USB - wykorzystywane do zasilania, programowania oraz komunikacji z komputerem
2. Złącze zasilania zewnętrznego (7V - 12V)
3. Stabilizator napięcia - napięcie wejściowe ze złącza nr 2 obniżane jest do 5V
4. Przycisk resetu - resetuje płytke Arduino
5. Mikrokontroler odpowiedzialny za komunikację z komputerem przez USB
6. Złącze programowania do mikrokontrolera z punktu 5.
7. Złącze sygnałowe
8. Złącze sygnałowe
9. Dioda LED sygnalizująca podłączenie napięcia do Arduino
10. Wyjście programatora dla mikrokontrolera z punktu 13.
11. Złącze sygnałowe
12. Złącze zasilania
13. Główny mikrokontroler AVR ATmega328
14. Diody LED sygnalizujące transmisję do/z komputera
15. Dioda LED do dyspozycji użytkownika połączona z pinem 13
16. Rezonator ceramiczny taktujący mikrokontroler z punktu 13 z częstotliwością 16MHz
17. Zworka, której przecięcie wyłącza automatyczne resetowanie Arduino.
18. Pola lutownicze z wyprowadzonymi sygnałami mikrokontrolera z punktu 5, (używane bardzo rzadko)

[*]

rozwinięty opis złączy (wyprowadzeń, pinów) na płytce Arduino



[*]



piny o numerach od 0 do 19 są to **uniwersalne cyfrowe piny wejścia/wyjścia (I/O)**

dotychczas piny o numerach od 14 do 19 są mogą pracować jako **wejścia analogowe A0 - A5**

alternatywne funkcje pinów (zaznaczone na niebiesko):

SDA, SCL - wyprowadzenia magistrali I²C wykorzystywanej np.: do komunikacji z bardziej zaawansowanymi czujnikami, wyprowadzenia tych pinów są zdublowane - lewym dolnym i prawym górnym róg płytki, I²C (*Inter-Integrated Circuit*) - szeregowa, dwukierunkowa magistrala służąca do przesyłania danych: linie: SDA – linia danych (*Serial Data Line*) i SCL – linia zegara (*Serial Clock Line*)

TX, RX - interfejs UART, wykorzystywany głównie do komunikacji z komputerem (TX wysyłanie danych, RX odbieranie danych),

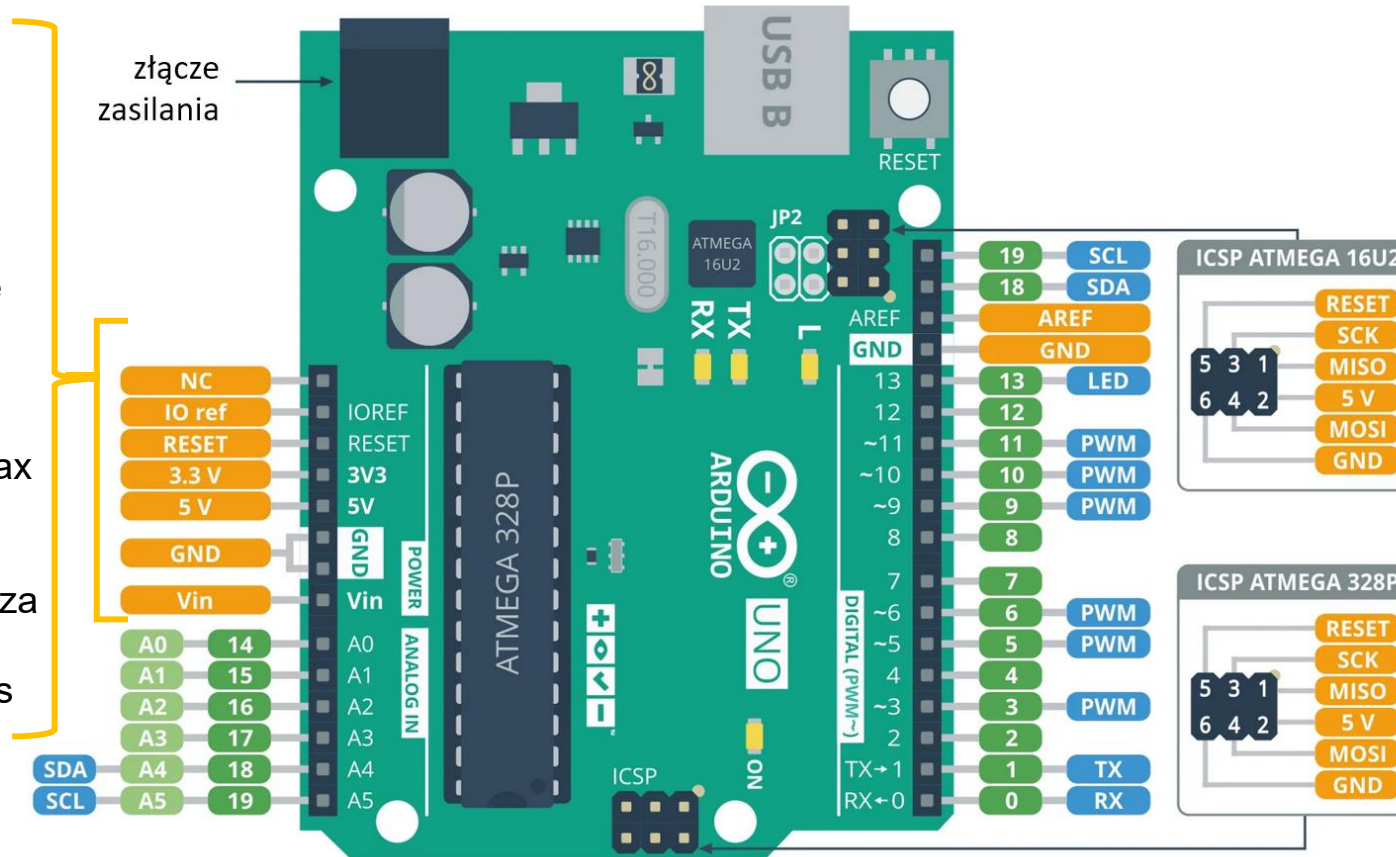
PWM – modulacja szerokości impulsu, wyprowadzenia, na których możliwe jest generowanie sygnału prostokątnego o zmiennym wypełnieniu (np. do sterowania serwomechanizmami),

LED - dioda świecąca, wbudowana na stałe w Arduino (oznaczona na płytce L), która połączona jest z pinem nr 13.

[*]

wyprowadzenia
które odpowiadają
za zasilanie układu

- NC** - not connected
- IO ref** - sygnalizuje modułom rozszerzającym Arduino jakim napięciem operują piny cyfrowe
- RESET** pełni funkcję tę samą co przycisk RESET (gdy zewrze się ten pin z pinem masy GND)
- 3V3** - wyjście stabilizatora 3,3V (max prąd 50 mA)
- 5V** - wyjście stabilizatora 5V (max prąd 500 mA)
- GND** – masa płytki
- Vin** - napięcie wejściowe zasilacza zewnętrznego podłączonego do złącza zasilania; zalecany zakres napięć zasilania 7V-12V



złącza do bezpośredniego programowania mikrokontrolera ATmega 16U2 obsługi komunikacji z komputerem przez USB

złącza do bezpośredniego programowania mikrokontrolera ATmega 328P

[*]

MIKROKONTROLERY

KONIEC

[*] <https://forbot.pl/blog/>