

# Wprowadzenie do mikrokontrolerów

Od systemu wbudowanego do RPi Pico 2 W

Podstawy Systemów Mikroprocesorowych ■ Wykład 1

dr inż. Dariusz Tefelski



- ▶ Ocena z zajęć laboratoryjnych jest wystawiana w następujący sposób:
  - 3,0-3,5 - w przypadku zaliczenia wszystkich ćwiczeń laboratoryjnych oraz miniprojektu,
  - +1,0 stopnia - w przypadku wykonania projektu końcowego,
  - +0,5 stopnia - w przypadku wykonania dokumentacji technicznej do projektu końcowego
- ▶ Ocena z wykładu jest wystawiana na podstawie kolokwium zgodnie z poniższą tabelą:

| Ocena | Wynik procentowy z kolokwium |
|-------|------------------------------|
| 5,0   | [91;100]%                    |
| 4,5   | [81;91)%                     |
| 4,0   | [71;81)%                     |
| 3,5   | [61;71)%                     |
| 3,0   | [51;61)%                     |

| Ocena | Ocena końcowa |
|-------|---------------|
| 5,0   | [4,5; 5.0]%   |
| 4,5   | [4.25; 4.5)%  |
| 4,0   | [3.75; 4.25)% |
| 3,5   | [3.25; 3.75)% |
| 3,0   | [3.0; 3.25)%  |

- ▶ Ocena końcowa jest równa średniej ważonej ocen z laboratorium i wykładu z wagami odpowiednio: 2/3 i 1/3 odpowiednio (przy czym obydwie oceny muszą być pozytywne). Ocena końcowa jest wystawiana zgodnie z powyższą tabelą.
- ▶ Materiały do wykładu i laboratoriów: <https://labe.engined.eu>.



## Komputer ogólnego przeznaczenia

- ▶ wielordzeniowy procesor pracujący z częstotliwością rzędu GHz,
- ▶ gigabajty pamięci RAM i rozbudowany system operacyjny,
- ▶ uruchamia wiele różnych aplikacji,
- ▶ pobór mocy zwykle liczony w watach lub dziesiątkach watów.

wydajność

uniwersalność



## System wbudowany

- ▶ realizuje konkretną funkcję,
- ▶ pracuje blisko czujników i elementów wykonawczych,
- ▶ ma ograniczone zasoby,
- ▶ często musi działać długo i niezawodnie,
- ▶ zużycie energii może sięgać od  $\mu\text{W}$  /  $\text{mW}$  do kilku  $\text{W}$  — zależnie od urządzenia.

Źródła obrazków: <https://www.apple.com>, <https://centrumstrazaka.pl/>,  
<https://exsmart.pl/>, <https://kenys.pl>.

# Systemy wbudowane: gdzie ich szukać?

## Urządzenia osobiste

zegarek ■ słuchawki ■ kalkulator  
opaska sportowa ■ aparat

## Dom i budynek

pralka ■ termostat ■ alarm  
roleta ■ pompa ciepła

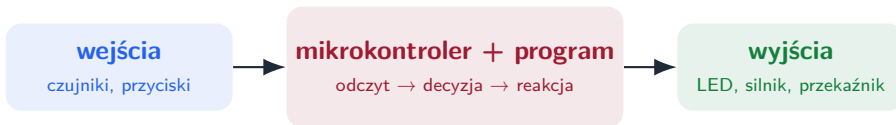
## Transport i robotyka

ABS ■ sterownik silnika ■ dron  
robot mobilny ■ e-bike

## Medycyna i przemysł

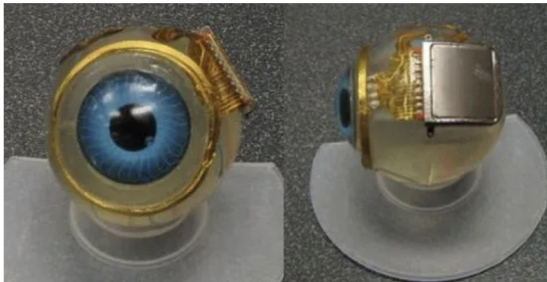
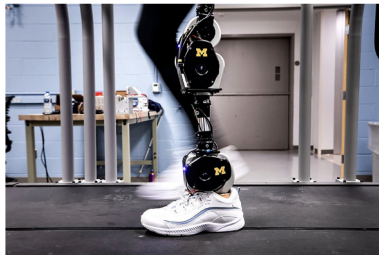
pulsoksymetr ■ pompa infuzyjna  
PLC ■ czujnik ■ rejestrator

# Co właściwie znaczy „system wbudowany”?



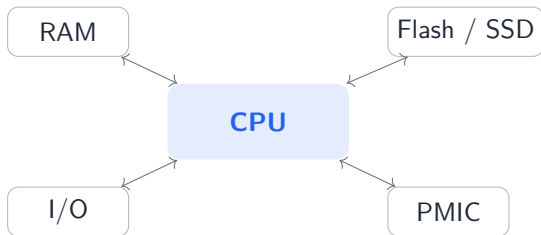
- 1 część większego urządzenia - komputer nie musi być „produktem” samym w sobie
- 2 określona funkcja - np. regulacja temperatury lub sterowanie napędem

# Po co uczyć się PSM?



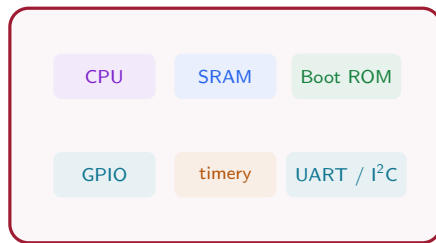
Źródła obrazków: <https://circuitdigest.com>, <https://thenewstack.io>, <https://www.pcworld.pl>

## Mikroprocesor (MPU)



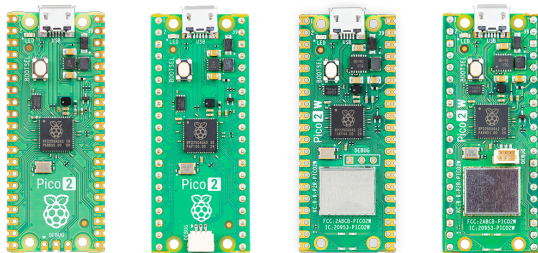
Układ skoncentrowany na wykonywaniu programu; kompletny system zwykle wymaga zewnętrznej pamięci i układów I/O.

## Mikrokontroler (MCU)



MCU integruje CPU, pamięć roboczą i peryferia. Pamięć programu Flash **nie musi** znajdować się w tym samym układzie scalonym.

# RP2350 na RPi Pico 2 W: pamięć Flash jest poza MCU

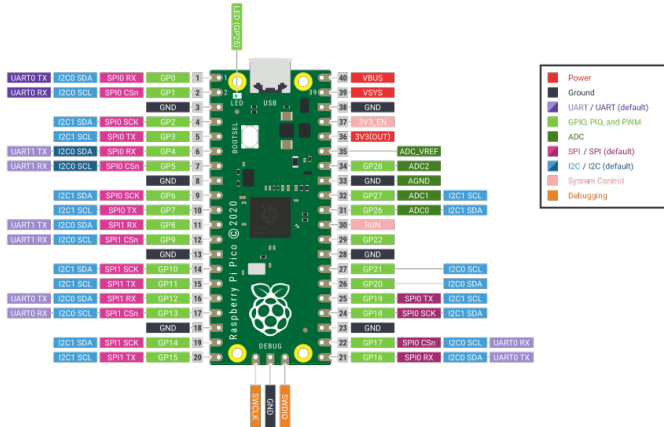
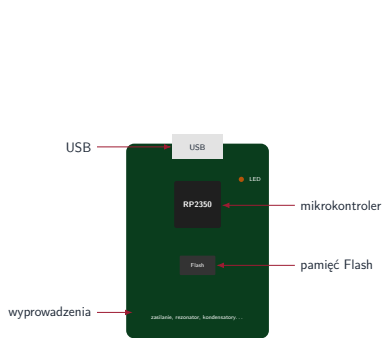


Źródło obrazka: <https://www.raspberrypi.com>



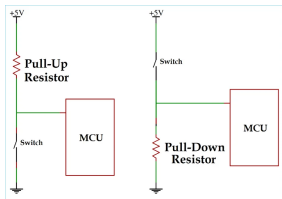
- ▶ Płytką **RPi Pico 2 W** trafiła do sprzedaży w 2024 r.
- ▶ Jej głównym mikrokontrolerem jest **RP2350**.
- ▶ RP2350 zawiera CPU, **520 kB SRAM**, Boot ROM oraz peryferia.
- ▶ Pamięć programu Flash znajduje się **poza RP2350**; na RPi Pico 2 W ma **4 MB**.
- ▶ RPi Pico 2 W zawiera także układ radiowy **CYW43439**: Wi-Fi 2,4 GHz i Bluetooth 5.2.

## Płytki rozwojowa $\neq$ mikrokontroler

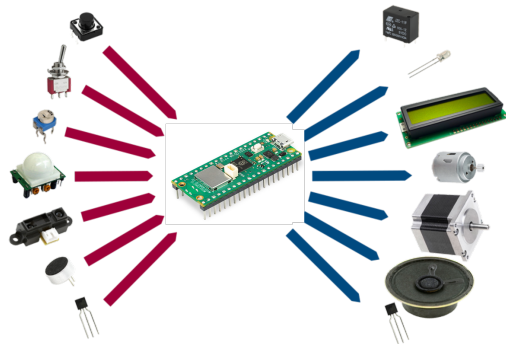


Źródło: <https://www.raspberrypi.com>.

- ▶ **GPIO** (*General Purpose Input/Output*) to wyprowadzenie mikrokontrolera, do którego można podłączyć zewnętrzne komponenty (czujniki, akuatory, diody itd.).
- ▶ Za pomocą odpowiedniej funkcji wybiera się, czy pin ma być wejściem, czy wyjściem.
- ▶ Piny mogą posiadać rezystory podciągające typu *pull-up* bądź *pull-down* po wybraniu odpowiedniej opcji.

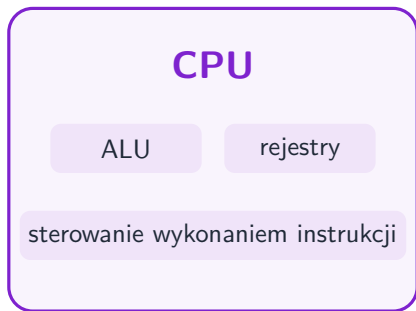


Źródło: <https://circuitdigest.com/tutorial/pull-up-and-pull-down-resistor>



Źródło oryginalnego obrazka: <https://blog.codebender.cc/2014/03/07/lesson-1-inputs-and-outputs/Projekt:IoT4scho>

# CPU: wykonuje instrukcje programu



CPU m.in. wykonuje operacje arytmetyczne i logiczne, przenosi dane, realizuje skoki i reaguje na przerwania.

- ▶ **CPU** (*Central Processing Unit*) wykonuje instrukcje programu i koordynuje pracę układu.
- ▶ **ALU** (*Arithmetic Logic Unit*) realizuje operacje arytmetyczne i logiczne.
- ▶ **Rejestry CPU** są bardzo szybką pamięcią wewnątrz rdzenia. W Cortex-M33 podstawowe rejestry rdzenia mają szerokość **32 bitów**.
- ▶ **Rejestry peryferiów i sterujące** służą do konfiguracji sprzętu; w mikrokontrolerach są zwykle dostępne pod określonymi adresami pamięci (*memory-mapped I/O*).
- ▶ Rejestr statusowy **xPSR** przechowuje m.in. flagi wyniku operacji.

# Architektura Von-Neumann-a a Harvard-a

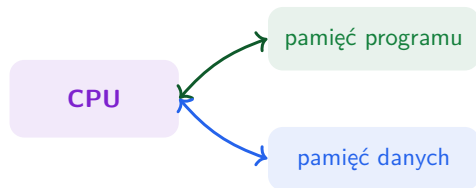


Źródło: [https://www.researchgate.net/publication/331160423\\_Von-Neumann\\_Architecture\\_Vs\\_Harvard\\_Architecture](https://www.researchgate.net/publication/331160423_Von-Neumann_Architecture_Vs_Harvard_Architecture).

- ▶ Instrukcje programu i dane znajdują się w **wspólnej przestrzeni pamięci**.
- ▶ CPU korzysta ze wspólnej drogi dostępu do instrukcji i danych.
- ▶ Model jest prosty i elastyczny, ale wspólny kanał może stać się ograniczeniem przepustowości (*von Neumann bottleneck*).



- ▶ Instrukcje i dane mają **oddzielne przestrzenie i/lub ścieżki dostępu**.
- ▶ CPU może równoległe pobierać instrukcję i wykonywać dostęp do danych.
- ▶ Współczesne procesory często stosują **zmodyfikowaną architekturę Harvard**: rozdzielone ścieżki blisko rdzenia, przy bardziej złożonej wspólnej hierarchii pamięci.
- ▶ Rdzeń **Arm Cortex-M33** używany w RP2350 ma 3-stopniowy potok i architekturę Harvard.



## RISC (*Reduced Instruction Set Computer*)

- ▶ regularny, relatywnie prosty zestaw instrukcji,
- ▶ architektura często typu *load/store*,
- ▶ dużo rejestrów i rozwiązania sprzyjające potokowaniu,
- ▶ przykłady: ARM, RISC-V.

RP2350: Cortex-M33 lub Hazard3

## CISC (*Complex Instruction Set Computer*)

- ▶ więcej złożonych instrukcji i trybów adresowania,
- ▶ pojedyncza instrukcja może wykonywać bardziej rozbudowaną operację,
- ▶ klasyczny przykład: rodzina x86.

**Uwaga:** RISC/CISC oraz Harvard/von Neumann opisują **inne cechy architektury**. Nie należy ich utożsamiać.

- ▶ **Pamięć** przechowuje kod programu i dane.
- ▶ **Pamięci nieulotne** zachowują zawartość po wyłączeniu zasilania:

| Typ    | Kasowanie/zapis               | Ziarnistość        | Typowe zastosowanie            |
|--------|-------------------------------|--------------------|--------------------------------|
| ROM    | brak zapisu w normalnej pracy | –                  | kod startowy, stałe tablice    |
| EPROM  | kasowanie UV                  | cała kość          | starsze urządzenia             |
| EEPROM | elektryczne                   | bajt / mała strona | ustawienia, konfiguracja       |
| Flash  | elektryczne                   | bloki / sektory    | firmware, pliki, pamięć masowa |

- ▶ **Pamięci ulotne:**
  - **SRAM** — szybka i typowa dla mikrokontrolerów;
  - **DRAM** — większa gęstość, wymaga odświeżania i jest typowa dla większych systemów.

# Pamięć: dlaczego mikrokontroler potrzebuje kilku jej rodzajów?

## RAM

szybka, ulotna

## Flash

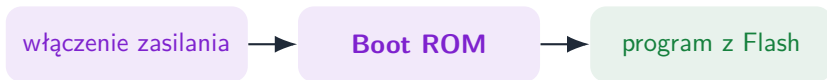
nieulotna, program

## ROM

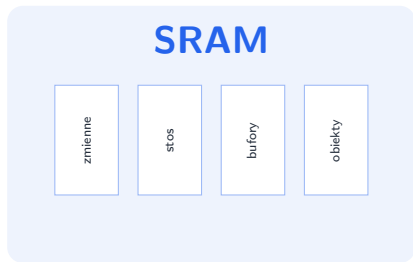
stała, kod startowy

Każda optymalizuje coś innego: **szybkość**, **pojemność**, **trwałość** lub **koszt**.

## ROM: kod, który ma być zawsze dostępny



RP2350 zawiera w ROM m.in. kod startowy i mechanizm bootowania USB. To dzięki niemu możemy łatwo wgrywać firmware metodą **BOOTSEL/UF2**.



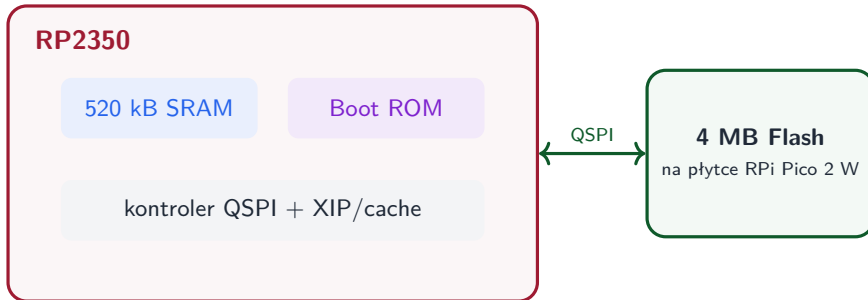
- ▶ dane używane **teraz**,
- ▶ bardzo szybki dostęp,
- ▶ zawartość znika po odłączeniu zasilania,
- ▶ RP2350 ma **520 kB SRAM**.

# Flash: „szafa” z programem i danymi trwałymi

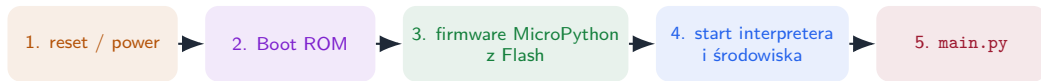


- ▶ pamięć **nieulotna**,
- ▶ przechowuje program po wyłączeniu zasilania,
- ▶ zapis jest wolniejszy od SRAM i pamięć ma ograniczoną liczbę cykli zapisu/kasowania,
- ▶ **RPi Pico 2 W** ma 4 MB zewnętrznej QSPI Flash.

## Podsumowanie: pamięć w RP2350 i na RPi Pico 2 W

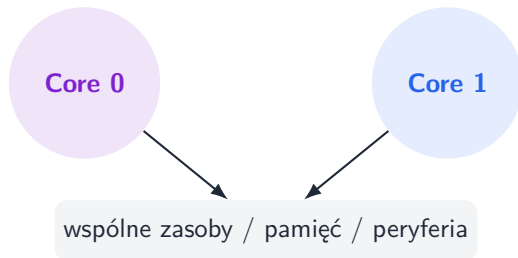


# Co dzieje się po włączeniu zasilania?



MicroPython nie jest osobną „warstwą po firmware” — **interpreter jest częścią firmware** wgranego do Flash.

## Dwa rdzenie: dwóch wykonawców, ale jedna praca nie dzieli się sama



RP2350 może pracować jako układ z **dwoma rdzeniami ARM Cortex-M33** albo **dwoma rdzeniami RISC-V Hazard3**, z częstotliwością do **150 MHz**. Drugi rdzeń nie przyspiesza programu automatycznie — zadanie trzeba podzielić i zsynchronizować.

# Zegar: rytm pracy układu



$$f = \frac{1}{T}$$

Zegar synchronizuje pracę CPU i wielu peryferiów. Większa częstotliwość może zwiększać wydajność, ale zwykle oznacza też większe zużycie energii.

## Czy 150 MHz oznacza 150 milionów linii Pythona na sekundę?

**Nie. Częstotliwość zegara  $\neq$  szybkość wykonywania programu**



Na czas wykonania wpływają m.in. liczba instrukcji, architektura CPU, dostęp do pamięci, interpreter oraz oczekiwanie na urządzenia peryferyjne.

# Firmware, interpreter, aplikacja — jak to się łączy?

**aplikacja:** Wasz kod `main.py`

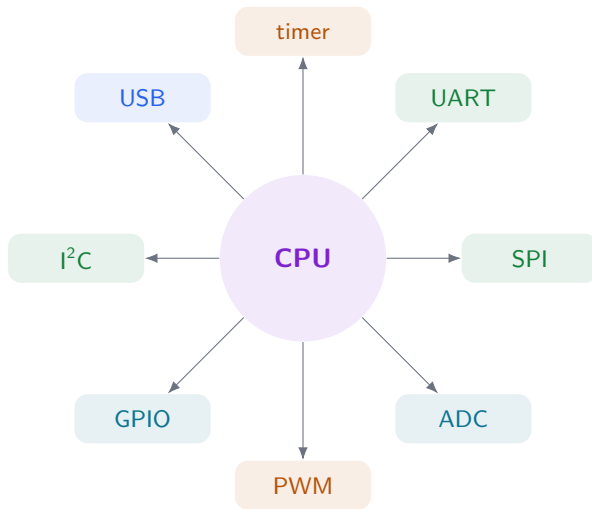
**firmware** MicroPython

interpreter + biblioteki + kod niskiego poziomu dla platformy

**hardware:** RP2350 + Flash + układ radiowy + płytki + urządzenia

Wgrywając nową wersję MicroPythona, zmieniamy **firmware**. Edytując `main.py`, zmieniamy **aplikację**.

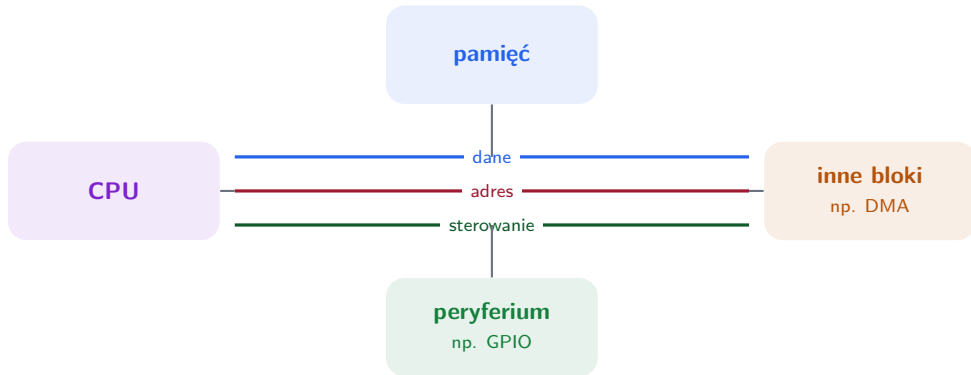
## Peryferia: sprzętowi pomocnicy CPU



Peryferia wykonują wyspecjalizowane zadania sprzętowo, dzięki czemu CPU nie musi „ręcznie” generować każdego impulsu.

# Magistrala: dane, adres i sterowanie

- Klasyczny model magistrali pomaga zrozumieć, **co** jest przesyłane między CPU, pamięcią i peryferiami:



**dane**  
co przesyłamy?

**adres**  
skąd / dokąd?

**sterowanie**  
jaki typ operacji?

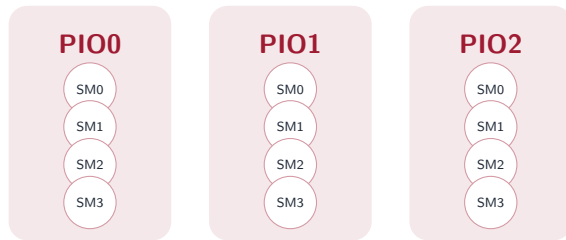
Uwaga: to model dydaktyczny. RP2350 ma bardziej złożoną wewnętrzną strukturę magistralową (m.in. fabric/crossbar), a nie jedną fizyczną „szynę” biegnącą przez cały układ.

## USB: nie tylko „gniazdo do wgrywania programu”



RP2350 ma kontroler USB i PHY. Na RPi Pico 2 W port USB służy m.in. do zasilania, programowania i komunikacji z komputerem.

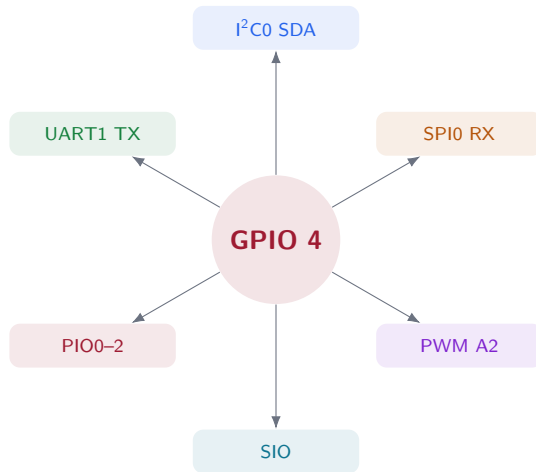
# PIO: nietypowa moc RP2350



**3 bloki PIO × 4 maszyny stanów = 12 SM**

- ▶ programowalne wejścia/wyjścia,
- ▶ precyzyjne generowanie i odczyt przebiegów,
- ▶ implementacja nietypowych protokołów,
- ▶ odciążenie głównych rdzeni CPU.

## Jeden pin może mieć wiele funkcji



Funkcja pinu jest wybierana przez konfigurację. Dlatego pinout wygląda „gęsto” — ten sam fizyczny pin może obsługiwać różne peryferia.

# RPi Pico 2 W / RP2350 — liczby, które warto kojarzyć

**2 rdzenie @ 150 MHz**

Cortex-M33 lub Hazard3

**520 kB SRAM**

w RP2350

**4 MB Flash**

na RPi Pico 2 W

**26 GPIO**

w tym 3 wejścia ADC na pinach

**Wi-Fi + Bluetooth**

2,4 GHz 802.11n + BT 5.2

**2× UART, I<sup>2</sup>C, SPI**

USB + 3×PIO (12 SM)

# O LICZBACH - DLA PRZYPOMNIENIA...

- ▶ **Bit** - cyfra w układzie binarnym (podstawowa jednostka informacji).
- ▶ **Bajt** - słowo, liczba złożona z 8 bitów.

Przeliczanie liczb:

- ▶ z systemu dwójkowego na dziesiętny:

$$\text{np. } 1100 = 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 = 8 + 4 + 0 + 0 = 12$$

- ▶ z systemu dwójkowego na szesnastkowy:

np. 1101 0010

$$0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 2$$

$$1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 13$$

W systemie szesnastkowym: A=10, B=11, C=12, D=13, E=14, F=15, zatem:

1101 0010 = 0xD2, alternatywne zapisy: 0xd2, D2h.

# O LICZBACH - DLA PRZYPOMNIENIA...

► z systemu dziesiętnego na dowolny :

(A) liczba 21 z dziesiętnego na czwórkowy:

| Iteracja | Operacja      | Reszta | Aktualna liczba |
|----------|---------------|--------|-----------------|
| 1        | $21 // 4 = 5$ | 1      | 1               |
| 2        | $5 // 4 = 1$  | 1      | 11              |
| 3        | $1 // 4 = 0$  | 1      | 111             |

czyli 21 jest równe 111 w systemie czwórkowym. Inny zapis:  $111_4$ .  
Indeks informuje o podstawie.

(B) liczba 21 z dziesiętnego na piątkowy:

| Iteracja | Operacja      | Reszta | Aktualna liczba |
|----------|---------------|--------|-----------------|
| 1        | $21 // 5 = 4$ | 1      | 1               |
| 2        | $4 // 5 = 0$  | 4      | 41              |

czyli 21 to  $41_5$ .

► z dowolnego systemu liczbowego na dziesiętny:

$$(A) \quad 120_3 = 1 \cdot 3^2 + 2 \cdot 3^1 + 0 \cdot 3^0 = 9 + 6 + 0 = 15$$

$$(B) \quad 33_9 = 3 \cdot 9^1 + 3 \cdot 9^0 = 27 + 3 = 30$$

$$(C) \quad 1h_{20} = 1 \cdot 20^1 + 17 \cdot 20^0 = 37, \text{ bo } a=10, \dots, h=17.$$