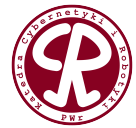
Sterowniki Robotów
Mikrokontrolery w robotyce

Wojciech Domski

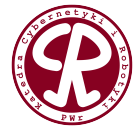
Katedra Cybernetyki i Robotyki,
Politechnika Wrocławska

Wrocław University
of Science and Technology



Plan prezentacji

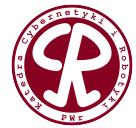
Wrocław University
of Science and Technology



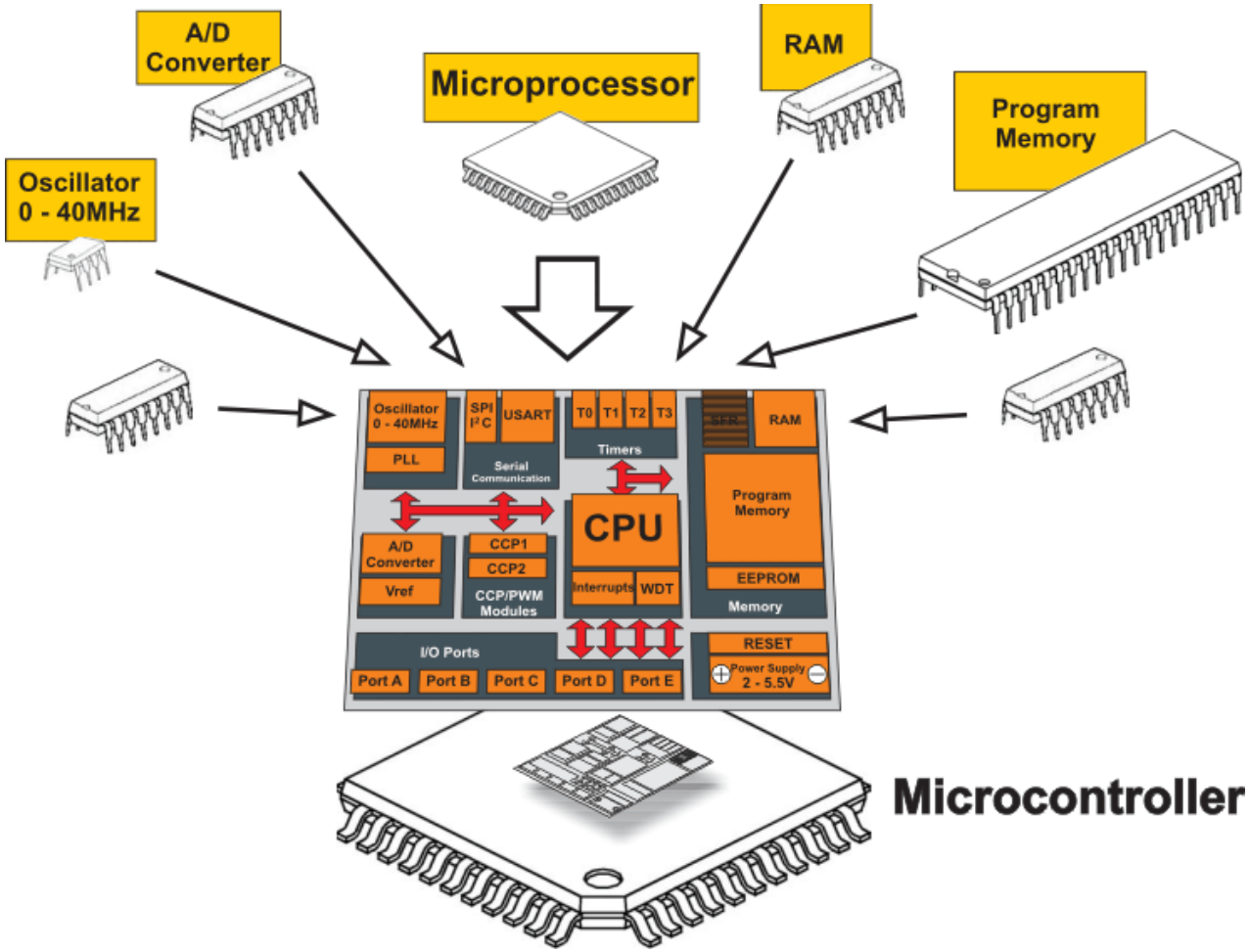
Mikrokontroler (1/2)

Czym właściwie jest mikrokontroler?

Wrocław University
of Science and Technology



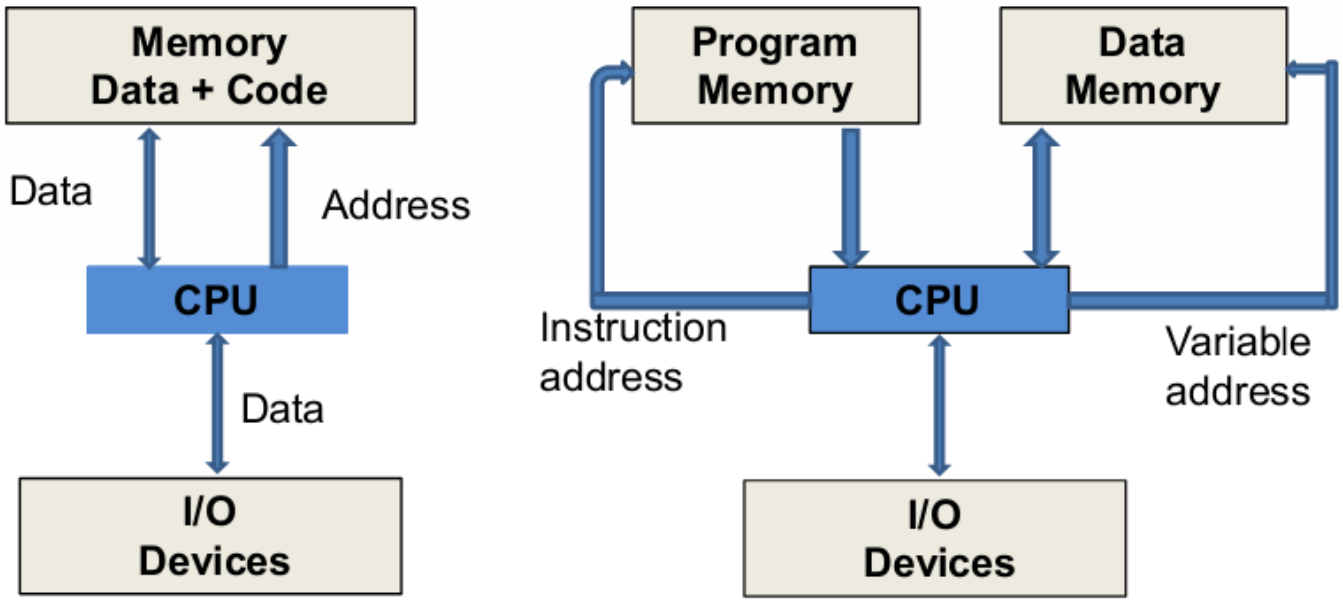
Mikrokontroler (2/2)



Terminologia

- ALU – Arithmetic Logic Unit
- CPU – Central Processing Unit
- FPU – Floating Point Unit
- DMA – Direct Memory Access
- MPU – Microprocessor Unit
- MCU – Microcontroller Unit
- DSP – Digital Signal Processor
- DSC – Digital Signal Controller
- CPLD – Complex Programmable Logic Device
- FPGA – Field-Programmable Gate Array
- ASIC – Application-Specific Integrated Circuit
- IP – Intellectual Property

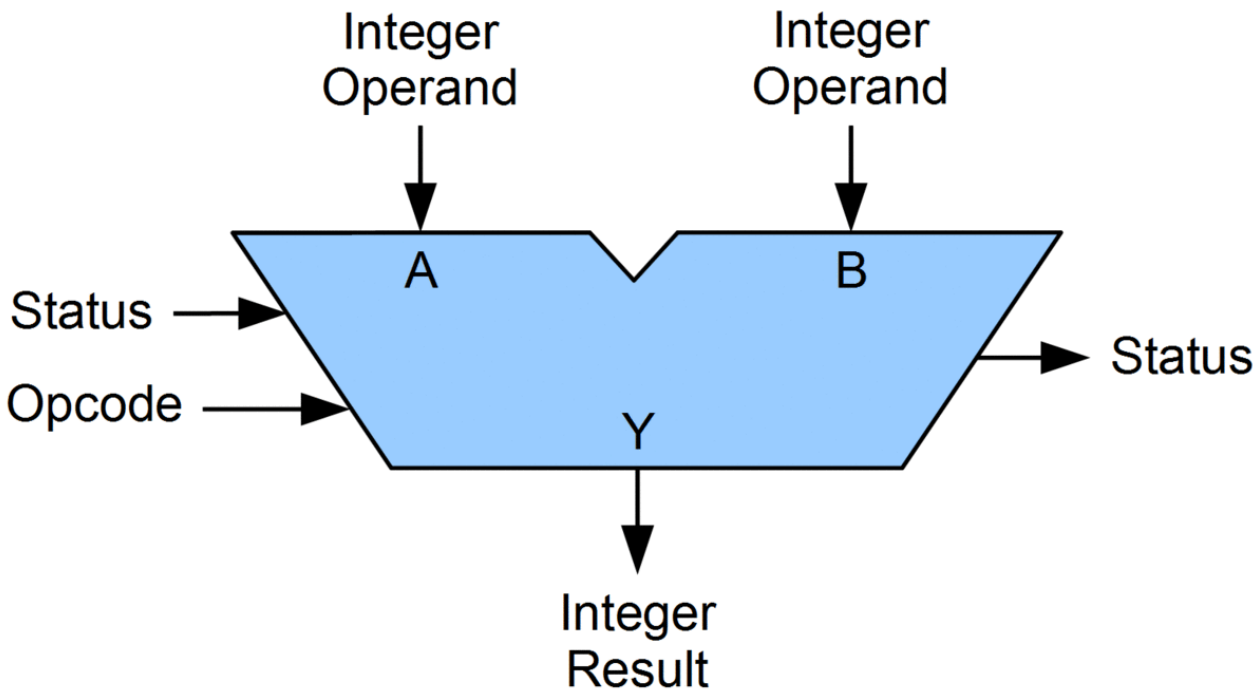
Architektura von Neumanna i architektura harwardzka



2

[2] Dilum Bandara, *CS2052 Computer Architecture*

ALU



3

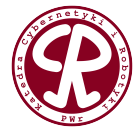
[3] Jim Lamberson, Wikimedia Commons

CISC vs. RISC (1/2)

Complex Instruction Set Computing

- Złożone instrukcje wykonywane w wielu cyklach
- Dowolna instrukcja możne wykonywać operacje na pamięci
- Przetwarzania potokowe – brak lub w niewielkim stopniu
- Instrukcje przetwarzane przez mikroprogram
- Różnorodny format instrukcji
- Wiele instrukcji i trybów adresowania
- Trudność w opracowywaniu mikrokodu
- Pojedynczy zestaw rejestrów

Wrocław University
of Science and Technology

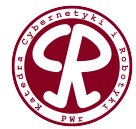


CISC vs. RISC (2/2)

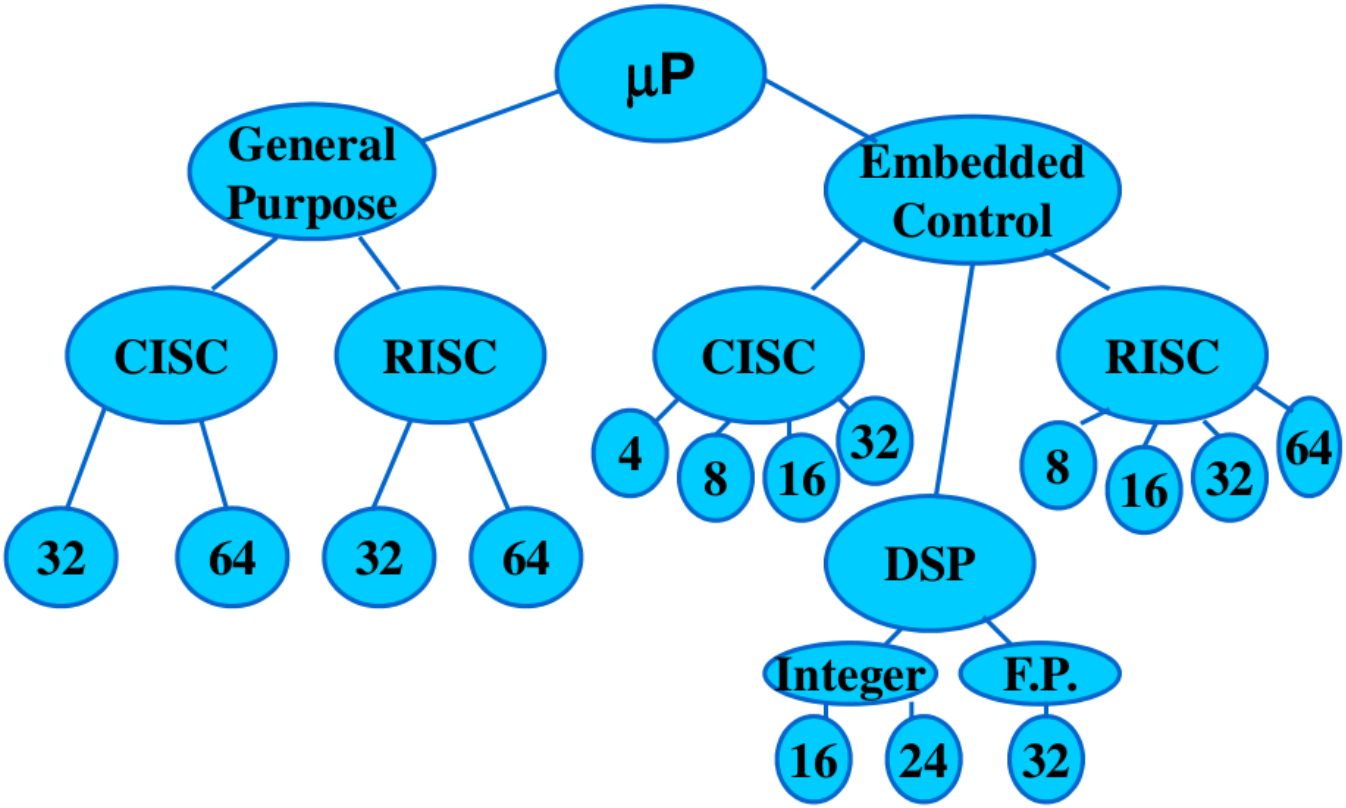
Reduced Instruction Set Computer

- Proste instrukcje wykonywane w jednym cyklu
- Operacje na pamięci tylko za pośrednictwem instrukcji LOAD/STORES
- Wielostopniowe przetwarzanie potokowe
- Instrukcje przetwarzane sprzętowo
- Ustalony format instrukcji
- Ograniczona liczba instrukcji i trybów adresowania
- Trudność w opracowywaniu kompilatorów
- Wiele zestawów rejestrów

Wrocław University of Science and Technology



Taksonomia



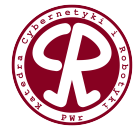
4

[4] D. G. Meyer, *Microcontroller-Based Digital System Design*

Producenci

- STMicroelectronics
- Raspberry Pi
- Renesas Electronics Corporation
- Infineon Technologies
- NXP
- Texas Instruments
- Microchip Technology
- *dawny* Atmel
- . . .

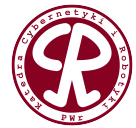
Wrocław University
of Science and Technology



Architektury MCU 32-bit

- Atmel AVR32
- Freescale ColdFire
- Renesans RX100, RX200, RX600, V850, RH850, SuperH
- Microchip PIC32
- Infineon AUDO, AURIX
- ARM Cortex-M
  - M0(+),
  - M1,
  - M3,
  - M4(F),
  - M7,
  - ...
- **RISC-V**,

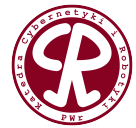
Wrocław University of Science and Technology



Architektury MCU 16-bit

- TI MSP430
- Microchip PIC24
- Freescale HCS12
- Infineon C166
- Renesas 78K0R
- . . .

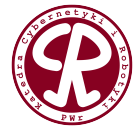
Wrocław University
of Science and Technology



Architektury MCU 8-bit

- Atmel AVR
- STMicroelectronics STM8
- Intel 8051
- Freescale HC08, HCS08
- Microchip PIC12, PIC16
- Renesas 78K0
- . . .

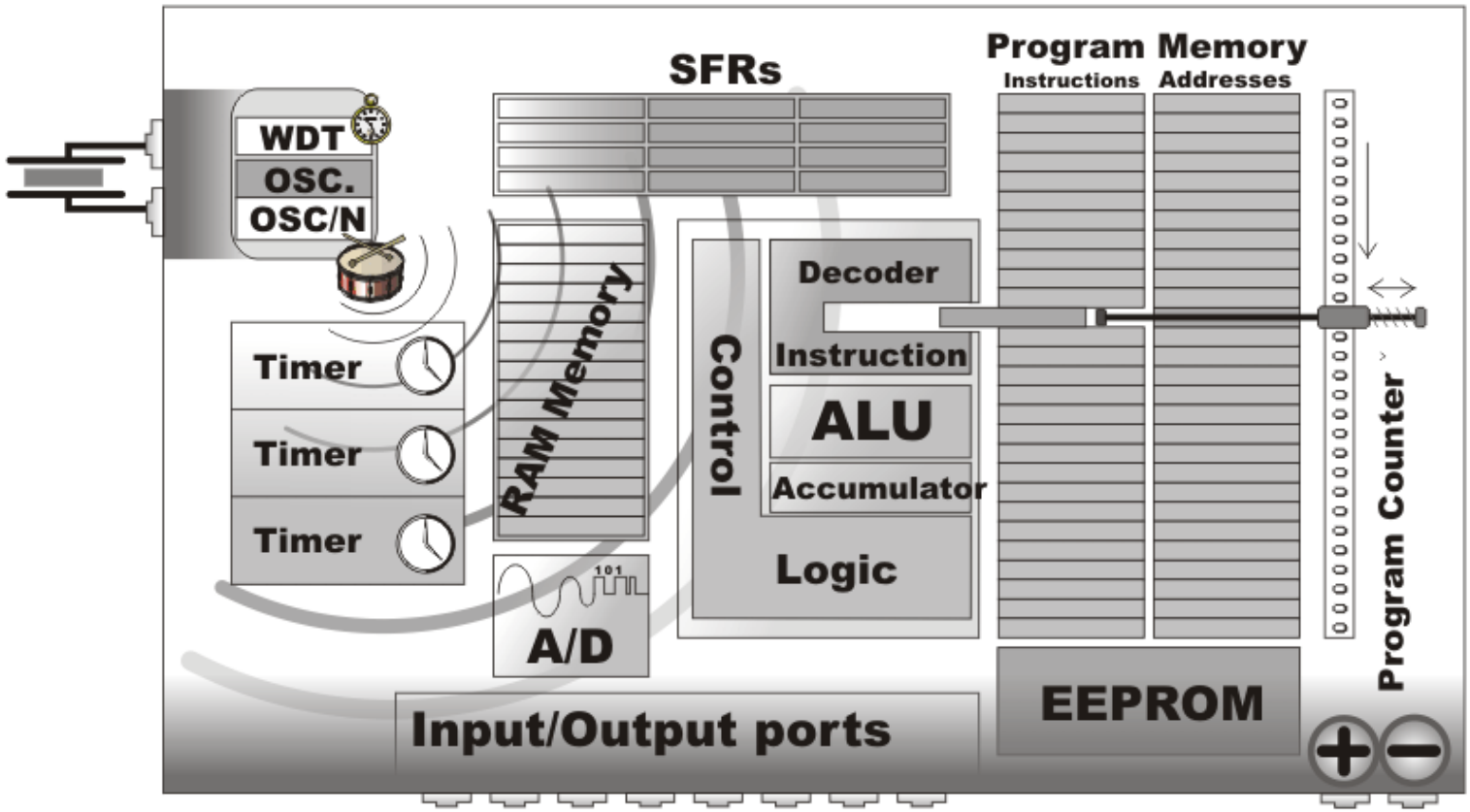
Wrocław University
of Science and Technology



Architektury DSP

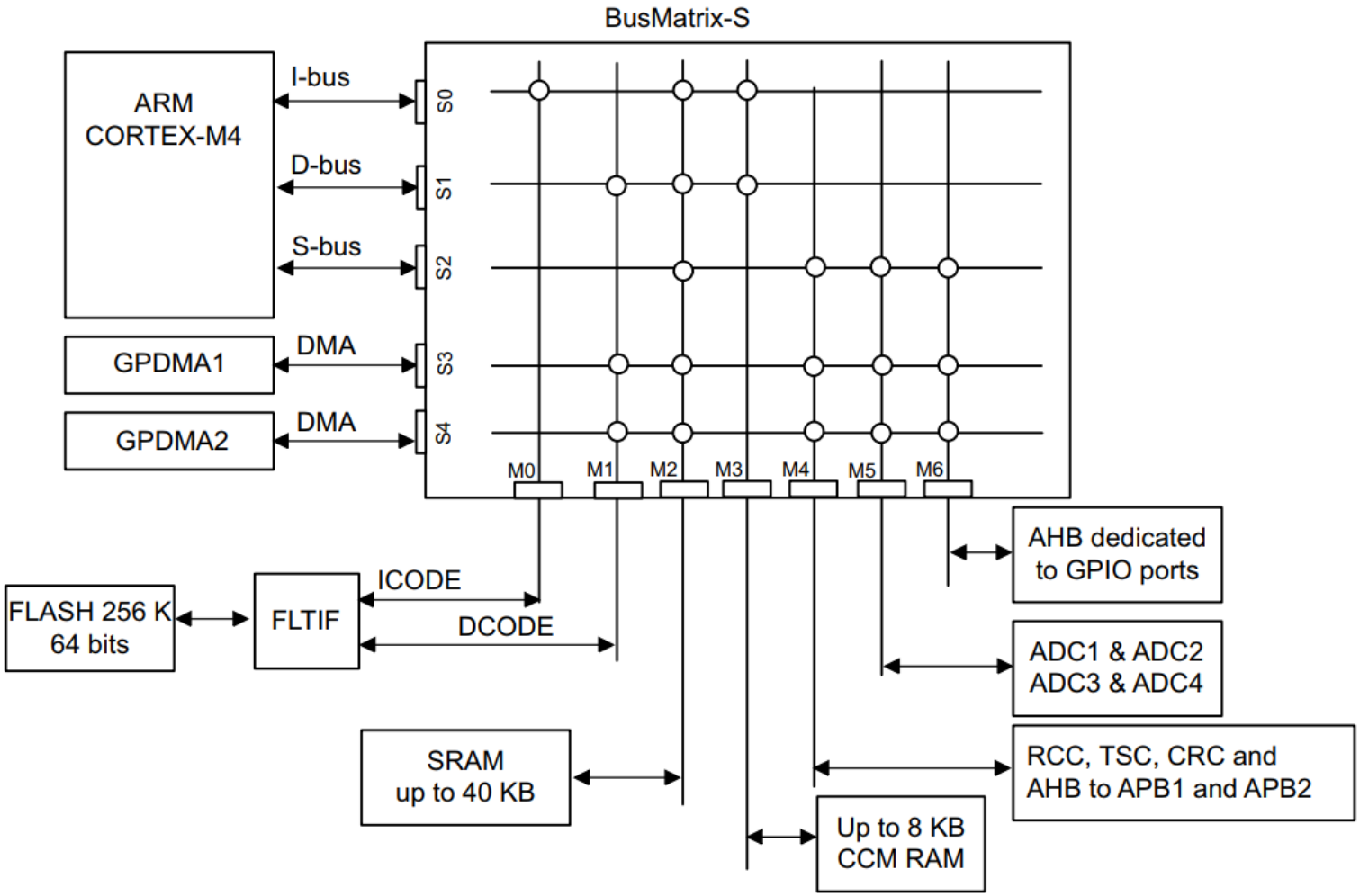
- Texas Instruments – Series DaVinci Digital Media Processors, C2000, C5000, C6000
- Analog Devices – Series ADSP-21xx, Blackfin, SHARC, TigerSHARC
- Freescale – Series 16-bit. StarCore, 24-bit. Symphony DSP56xxx, 16-bit. MC56Fxx

Struktura wewnętrzna



[5] http://www.mikroe.com

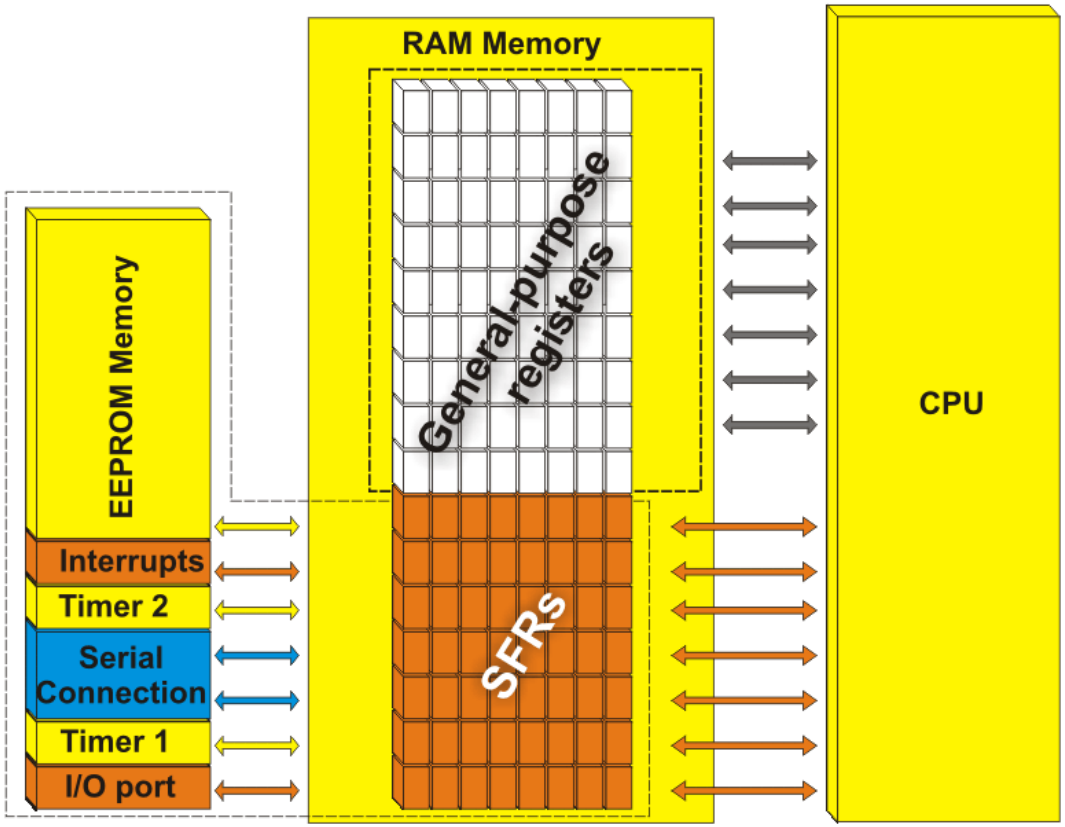
Szyna danych



Wrocław University of Science and Technology

[6] http://www.st.com

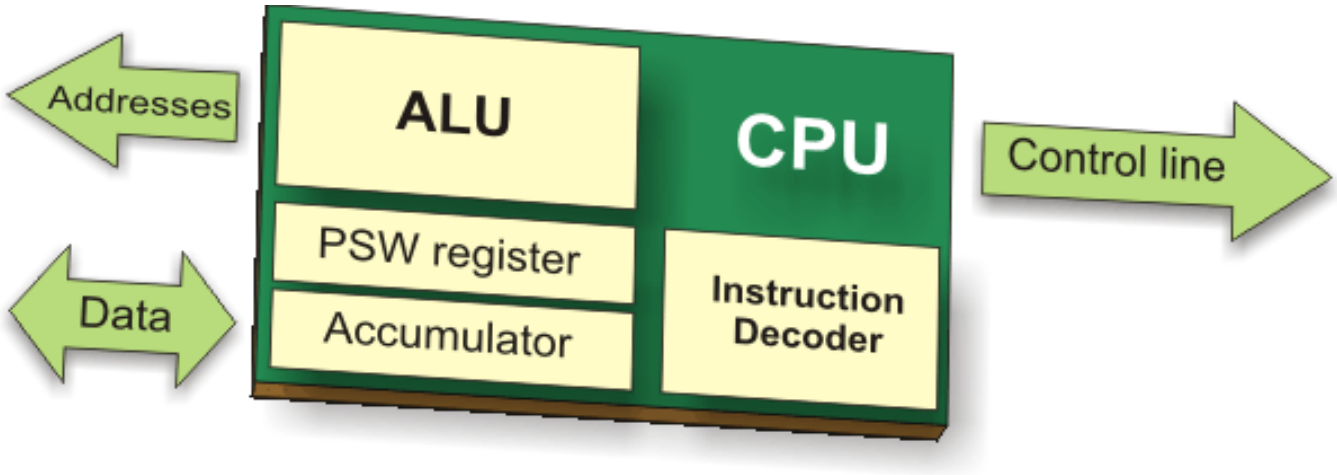
Mapa pamięci



[7] Wrocław University of Science and Technology

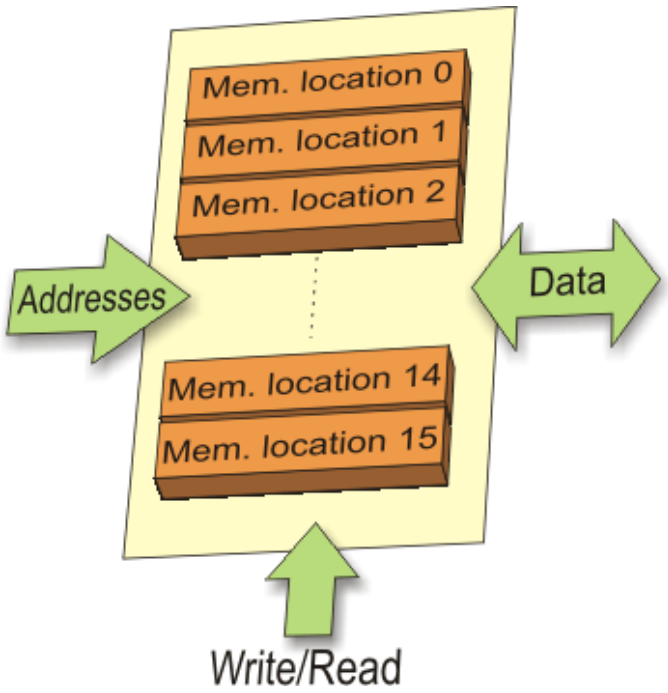
[7] http://www.mikroe.com

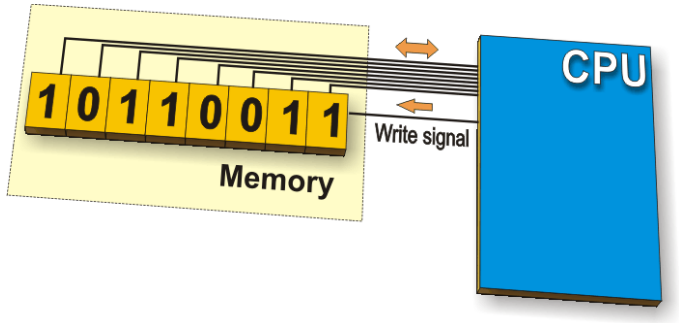
Central Processor Unit (CPU)

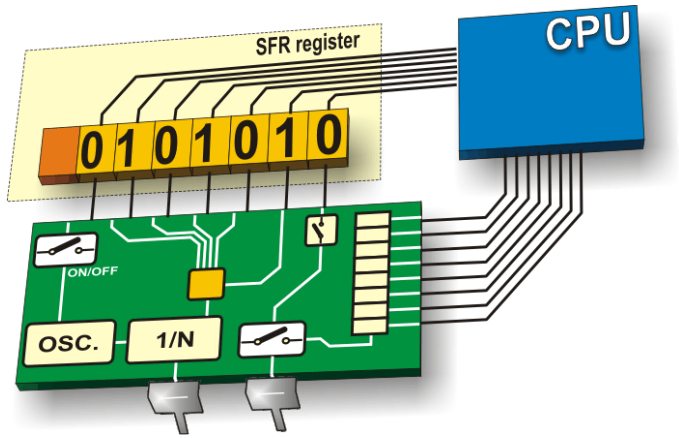


8

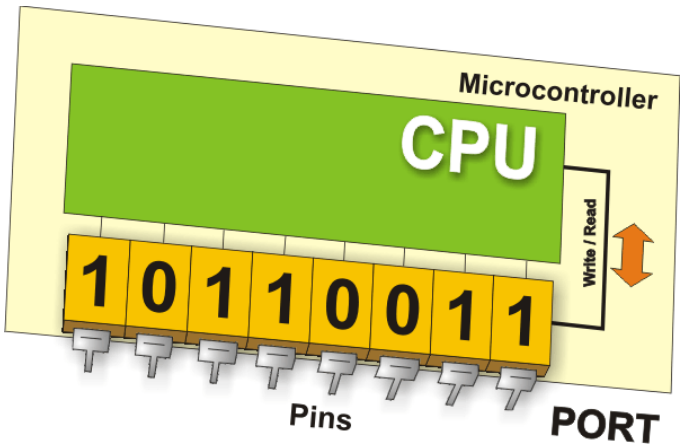
Pamięć i rejestry

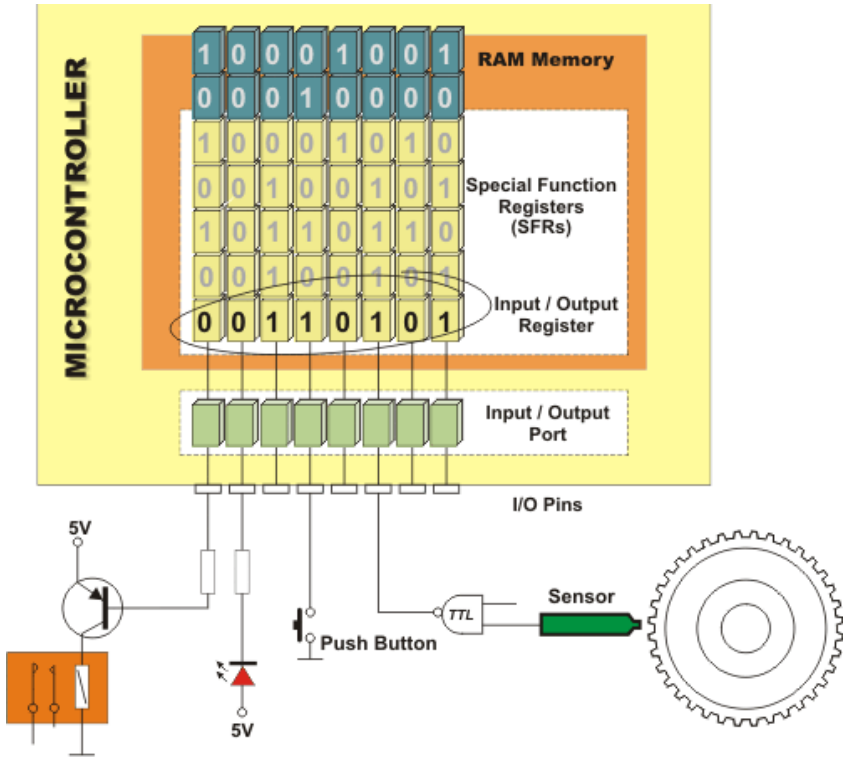






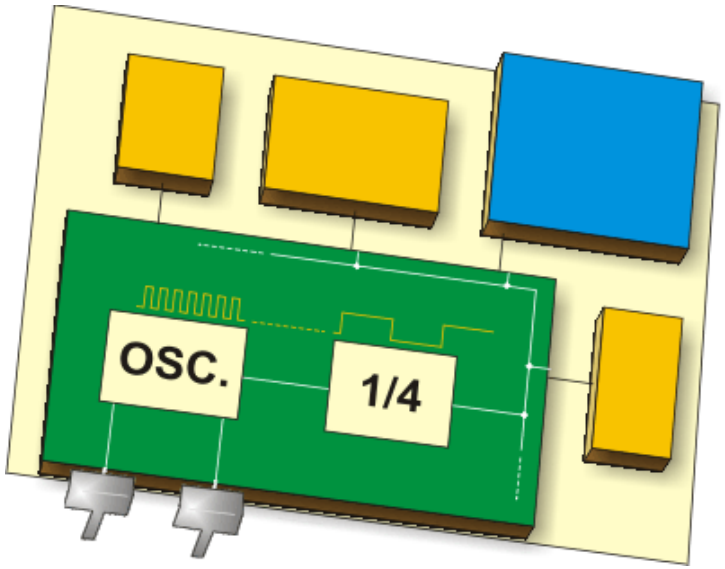
Porty wejścia/wyjścia





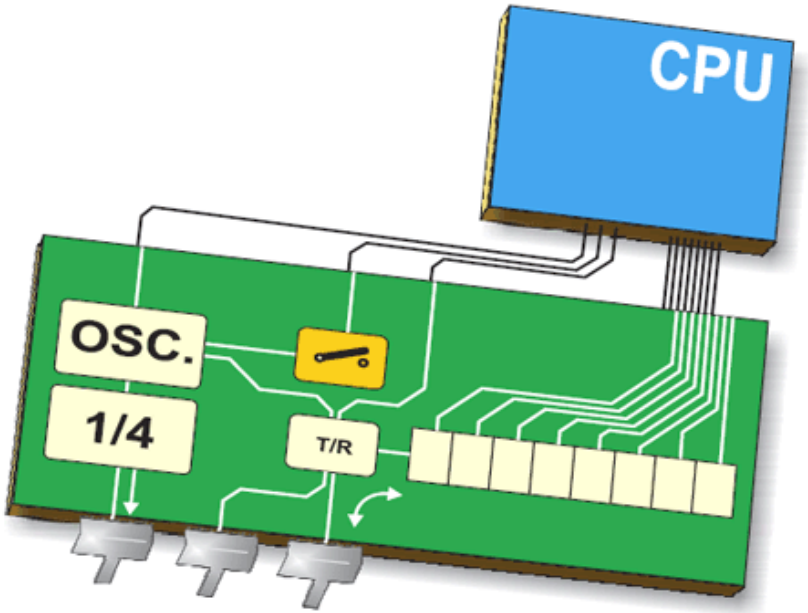
Wrocław University of Science and Technology

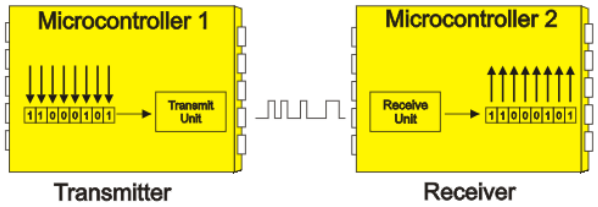
Oscylator

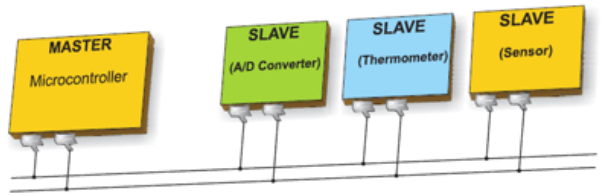


- wewnętrzne
  - szybkie
  - wolne
- zewnętrzne
  - szybkie
  - wolne

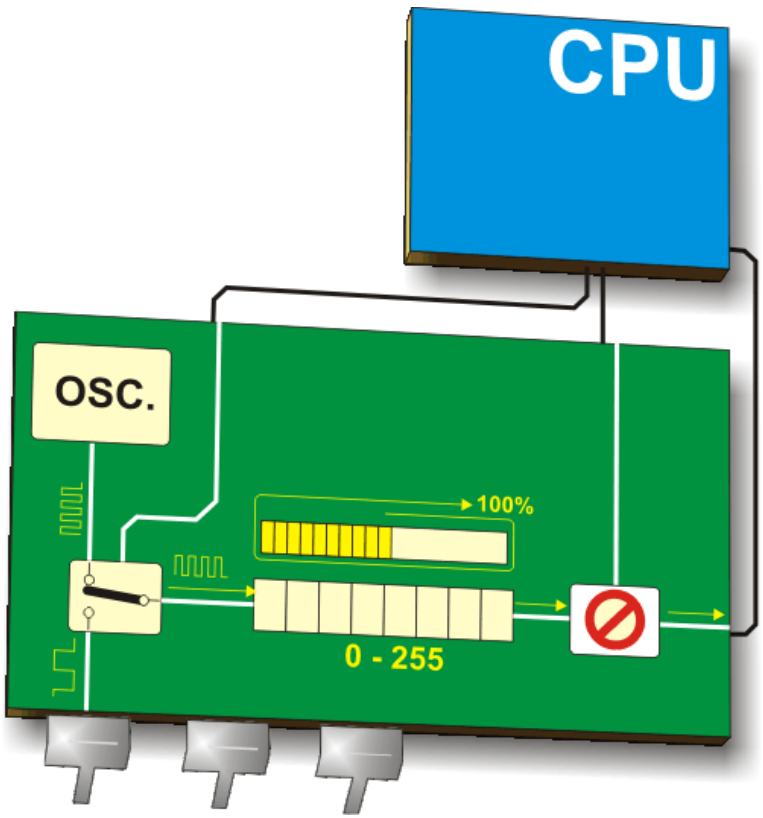
Komunikacja szeregowa

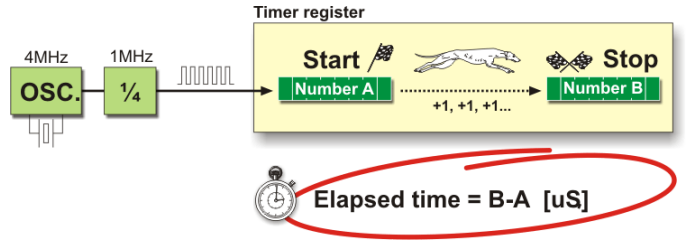


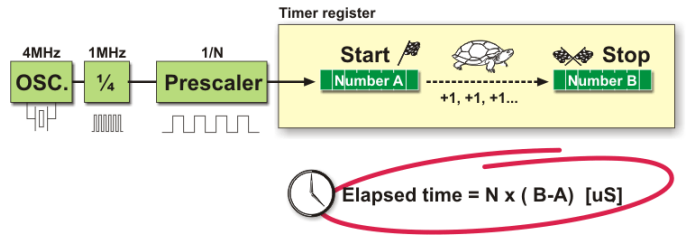


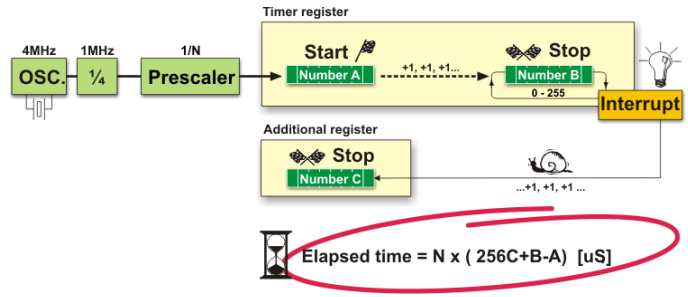


Układy czasowo-licznikowe



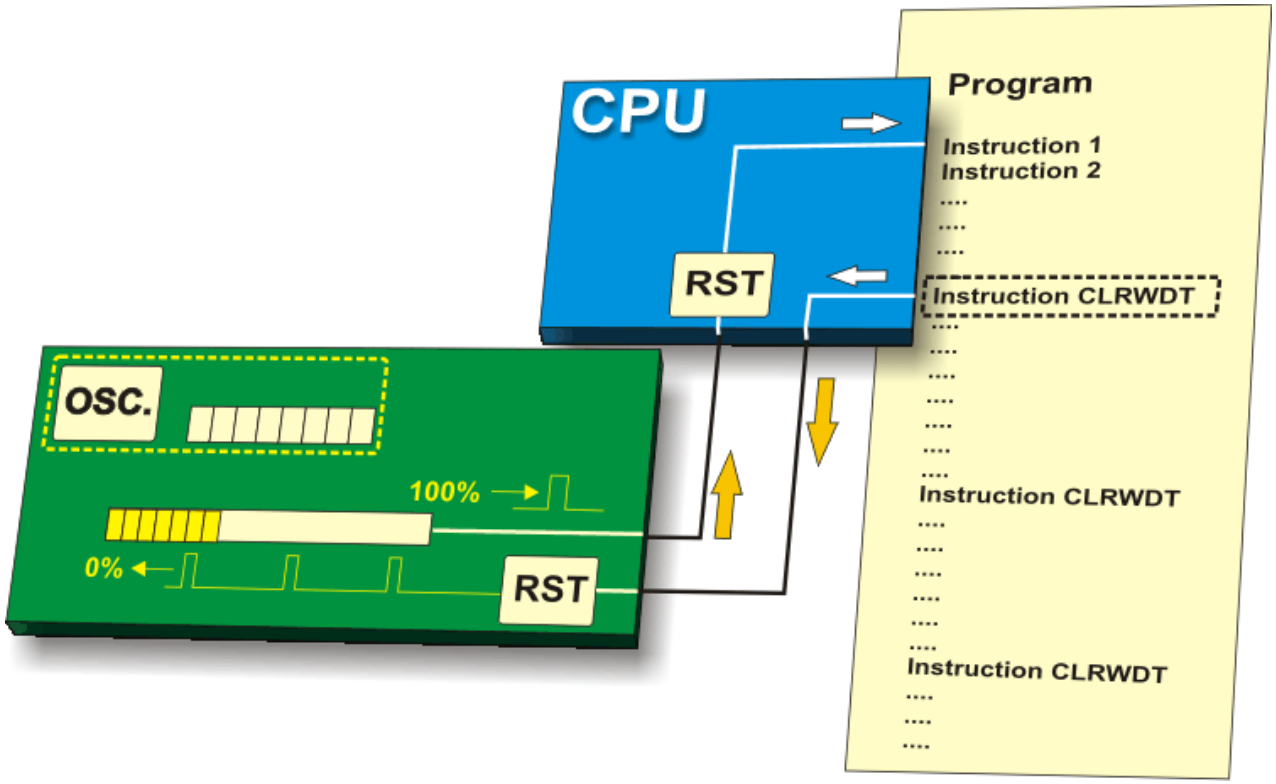






Wrocław University of Science and Technology

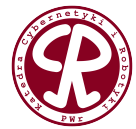
Watchdog Timer



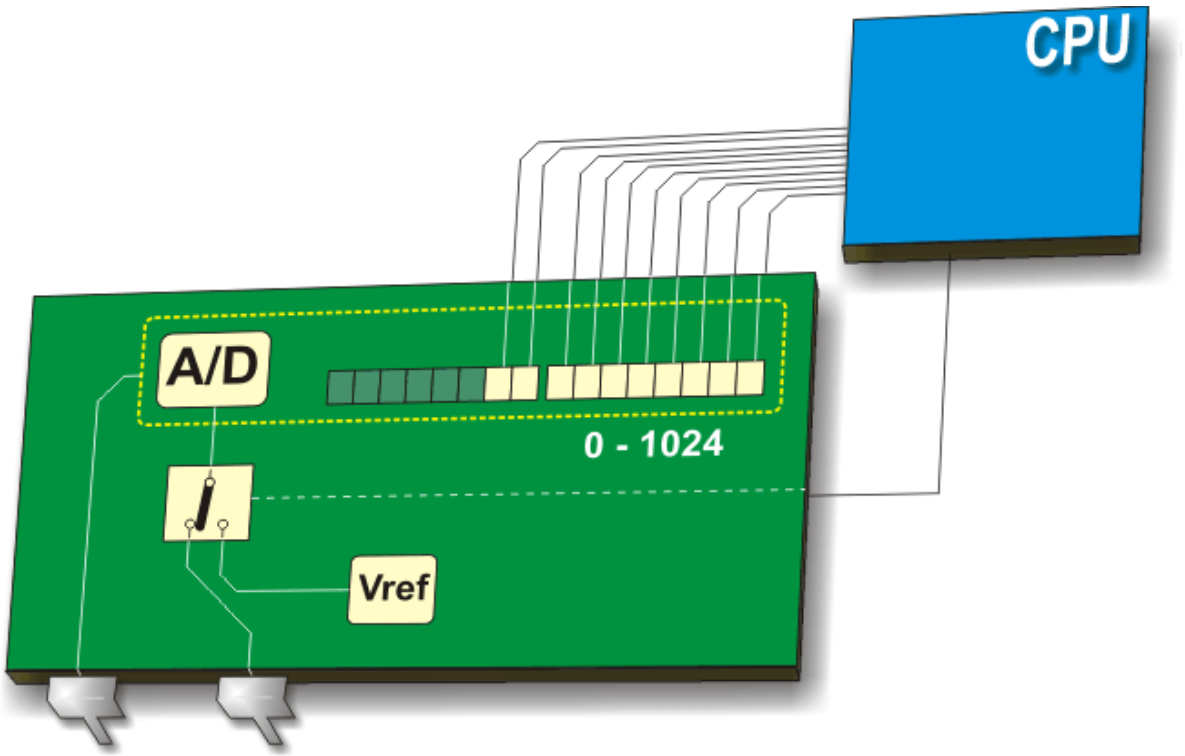
9

Wrocław University
of Science and Technology

[9] http://www.mikroe.com



Przetwornik analogowo-cyfrowy

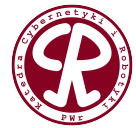


10

Podstawowe narzędzia

- Asembler
- Kompilator
- Edytor
- Debugger + interfejs sprzętowy (np. SWD/JTAG + OpenOCD, Segger SDT)
- IDE (zazwyczaj na bazie Eclipse + gcc/g++ + gdb + git)
- Biblioteki (SPL, BSP, USB, GPL)
- SDK
- System budowania (np. CMake),
- Zestaw startowy/rozwojowy
- ...

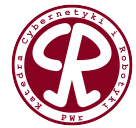
Wrocław University of Science and Technology



Podstawowe narzędzia

- Asembler
- Kompilator
- Edytor
- Debugger + interfejs sprzętowy (np. SWD/JTAG + OpenOCD, Segger SDT)
- IDE (zazwyczaj na bazie Eclipse + gcc/g++ + gdb + git)
- Biblioteki (SPL, BSP, USB, GPL)
- SDK
- System budowania (np. CMake),
- Zestaw startowy/rozwojowy
- ...

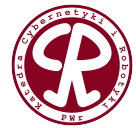
Wrocław University of Science and Technology



Podstawowe narzędzia

- Asembler
- Kompilator
- Edytor
- Debugger + interfejs sprzętowy (np. SWD/JTAG + OpenOCD, Segger SDT)
- IDE (zazwyczaj na bazie Eclipse + gcc/g++ + gdb + git)
- Biblioteki (SPL, BSP, USB, GPL)
- SDK
- System budowania (np. CMake),
- Zestaw startowy/rozwojowy
- ...

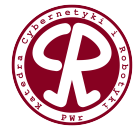
Wrocław University of Science and Technology



Podstawowe narzędzia

- Asembler
- Kompilator
- Edytor
- Debugger + interfejs sprzętowy (np. SWD/JTAG + OpenOCD, Segger SDT)
- IDE (zazwyczaj na bazie Eclipse + gcc/g++ + gdb + git)
- Biblioteki (SPL, BSP, USB, GPL)
- SDK
- System budowania (np. CMake),
- Zestaw startowy/rozwojowy
- ...

Wrocław University of Science and Technology



Podstawowe narzędzia

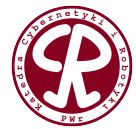
- Asembler
- Kompilator
- Edytor
- Debugger + interfejs sprzętowy (np. SWD/JTAG + OpenOCD, Segger SDT)
- IDE (zazwyczaj na bazie Eclipse + gcc/g++ + gdb + git)
- Biblioteki (SPL, BSP, USB, GPL)
- SDK
- System budowania (np. CMake),
- Zestaw startowy/rozwojowy
- ...

Wrocław University of Science and Technology

Podstawowe narzędzia

- Asembler
- Kompilator
- Edytor
- Debugger + interfejs sprzętowy (np. SWD/JTAG + OpenOCD, Segger SDT)
- IDE (zazwyczaj na bazie Eclipse + gcc/g++ + gdb + git)
- Biblioteki (SPL, BSP, USB, GPL)
- SDK
- System budowania (np. CMake),
- Zestaw startowy/rozwojowy
- ...

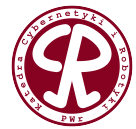
Wrocław University of Science and Technology



Podstawowe narzędzia

- Asembler
- Kompilator
- Edytor
- Debugger + interfejs sprzętowy (np. SWD/JTAG + OpenOCD, Segger SDT)
- IDE (zazwyczaj na bazie Eclipse + gcc/g++ + gdb + git)
- Biblioteki (SPL, BSP, USB, GPL)
- SDK
- System budowania (np. CMake),
- Zestaw startowy/rozwojowy
- ...

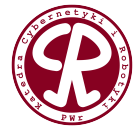
Wrocław University of Science and Technology



Popularne środowiska

- STM32CubeIDE
- Visual Studio Code *z zainstalowanymi rozszerzeniami*
- Keil MDK Microcontroller Development Kit
- IAR Embedded Workbench for ARM
- Atollic TrueSTUDIO
- CooCox (Free/open ARM Cortex-M Development Tool-chain)
- AC6 System Workbench for STM32
- NXP CodeWarrior (*dawny Freescale*)
- NXP LPCXpresso
- TI Code Composer Studio (CCS)
- Infineon DAVE
- ...

Wrocław University of Science and Technology



Etapy kompilacji (1/4)

Front-end:

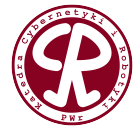
1. *Rekonstrukcja linii* (z ang. *line reconstruction*) przekształca sekwencję znaków wejściowych w postać kanoniczną gotową do analizy przez parser.
2. Przetwarzanie wstępne (z ang. *preprocessing*) odpowiada za podstawianie makr i kompilację warunkową.
3. Analiza leksykalna, tokenizacja (z ang. *lexical analysis*) dzieli tekst kodu źródłowego na sekwencję małych fragmentów nazywanych leksykalnymi tokenami. Składa się z dwóch etapów: skanowania i ewaluacji.
4. Analiza składni (z ang. *Syntax analysis*) polega na analizie sekwencji tokenów w celu zidentyfikowania składniowej struktury programu poprzez budowę drzewa analizy składniowej.

Wrocław University of Science and Technology

Etapy kompilacji (2/4)

5. Analiza semantyczna (z ang. *Semantic analysis*) dodaje informacje semantyczne do drzewa analizy składniowej i buduje tablicę symboli. Sprawdzane są np. typy, czy asocjacja obiektów.

Wrocław University
of Science and Technology

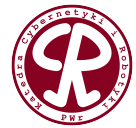


Etapy kompilacji (3/4)

Middle-end:

1. Analiza, gromadzenie informacji programu z reprezentacji pośredniej pochodzącej z wejścia; analiza przepływu danych służy do budowy łańcuchów użyć-definicji, wraz z ich analizą.
2. Optymalizacja, reprezentacja języka pośredniego jest przekształcana w formy równoważne funkcjonalnie, ale szybsze (lub mniejsze).

Wrocław University of Science and Technology

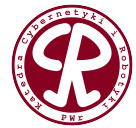


Etapy kompilacji (4/4)

Back-end:

1. Optymalizacje zależne od architektury, optymalizacje, które zależą od szczegółów architektury CPU, którą kompilator jest ukierunkowany.
   Generacja kodu: przekształcona reprezentacja języka pośredniego jest tłumaczona na język wyjściowy, zazwyczaj na natywny język maszynowy systemu.

Wrocław University
of Science and Technology



Konsolidacja

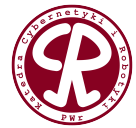
Konsolidacja kodu polega na łączeniu wielu plików obiektowych w:

- jeden skonsolidowany plik „obiektowy”,
- program wykonywalny,
- bibliotekę.

Typy konsolidacji:

- statyczna,
- dynamiczna.

Wrocław University of Science and Technology



Segmenty pamięci

- `.text`: Kod maszynowy do wykonania, z odwołaniami do funkcji w pliku obiektowym.
- `.rodata`: Dane tylko do odczytu, często mieszające się z sekcją `.text`.
- `.data`: Zainicjowane zmienne globalne i statyczne.
- `.bss`: Zmienne globalne i statyczne niezainicjowane, często wyzerowane przez kod startowy.
- `.isr_vector`: Adresy procedur obsługi przerwań, wymagane dla mikrokontrolerów Cortex-M.
- `.stack`: Obszar pamięci stosu, używany do przechowywania lokalnych zmiennych i adresów powrotu podczas wywoływania funkcji.
- `.heap`: Dynamicznie alokowana pamięć, wykorzystywana przez program do przechowywania danych w trakcie działania.

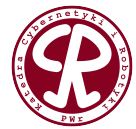
Wrocław University of Science and Technology

Executable and Linkable Format (1/2)

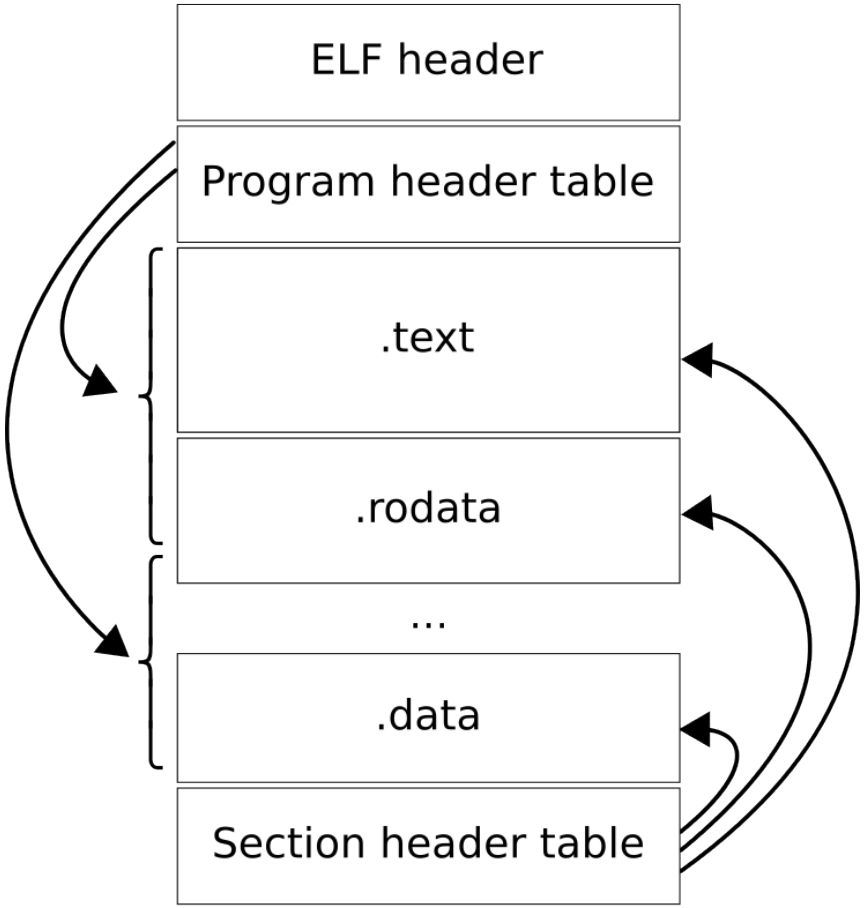
Każdy plik ELF składa się z jednego nagłówka ELF, po którym następują dane pliku.

- Nagłówek – szczegóły dotyczące identyfikacji i wykonywania
  - Nagłówek ELF – identyfikacja pliku ELF oraz architektury
  - Tabela nagłówka programu – informacje dotyczące wykonywania
- Sekcje – zawartość pliku wykonywalnego
  - `.text` – kod maszynowy,
  - `.data` – dane,
  - Nazwy sekcji
- Nagłówek – informacje wykorzystywane podczas konsolidacji,
  - Tabela nagłówków sekcji

Wrocław University of Science and Technology



Executable and Linkable Format (2/2)



[11] Wrocław University of Science and Technology

[11] Surueña, wikimedia.org