PSM

Podstawy systemów mikroprocesorowych

dr inż. Dariusz Tefelski

tefelski@if.pw.edu.pl

Pokój 225GF

Pierwotna wersja wykładu jest dziełem:

dr hab. Piotra Fronczaka

Na jednym końcu skali…

Procesor: Intel Core i7
Częstotliwość: 3.8GHz
Moc: 150 W max

Na drugim końcu skali

Systemy wbudowane
(Embedded Devices)
Moc: ~ mW

Czym są systemy wbudowane?

Przykłady

- **Urządzenia osobistego użytku:** telefon komórkowy, pager, zegarek, dyktafon, kalkulator
- **Komponenty komputerowe:** myszka, klawiatura, modem, fax, karta dźwiękowa, ładowarka baterii
- **Urządzenia domowe:** zamek do drzwi, budzik, termostat, klimatyzator, pilot, sprzęt fitness, pralka, zmywarka, kuchenka mikrofalowa
- **Zabawki:** gry wideo, samochody, lalki, itp.

Technologie komputerowe → dramatyczne zmiany

- **Procesor**
  - **2X szybość co każde 1.5 roku;**
    100X w ostatniej dekadzie
- **Pamięć**
  - **2X pojemność co każde 2 lata;**
  - **64X w ostatniej dekadze**
  - **Koszt jednego bitu: maleje o 25% na rok**
- **Dysk**
  - **pojemność > 2X każdego roku**
  - **250X w ostatniej dekadzie**
  - **Koszt jednego bitu: maleje o 50% na rok**

Technologie komputerowe → dramatyczne zmiany

° Komputer, który kupicie z okazji obrony pracy magisterskiej:

- Szybkość procesora: 10000 MegaHertzów (10.0 GigaHertzów)
- Pojemność pamięci: 64000 MegaBajtów (64.0 GigaBajtów)
- Pojemność dysku: 8000 GigaBajtów (8.0 TeraBajtów)
- Nowe jednostki! Mega => Giga, Giga => Tera

(Kilo, Mega, Giga, Tera, Peta, Exa, Zetta, Yotta = 10^{24})

“ Gdyby rozwój samochodów odzwierciedlał rozwój komputerów, Rolls Royce kosztowałby dziś 100 dolarów, pokonywałby milion kilometrów na jednym litrze paliwa i eksplodował raz do roku zabijając wszystkich wewnątrz. ”

Triumph of the Nerds – ***Robert X. Cringely***

Zmierzch ery PC?

Po co mam się uczyć PSM?

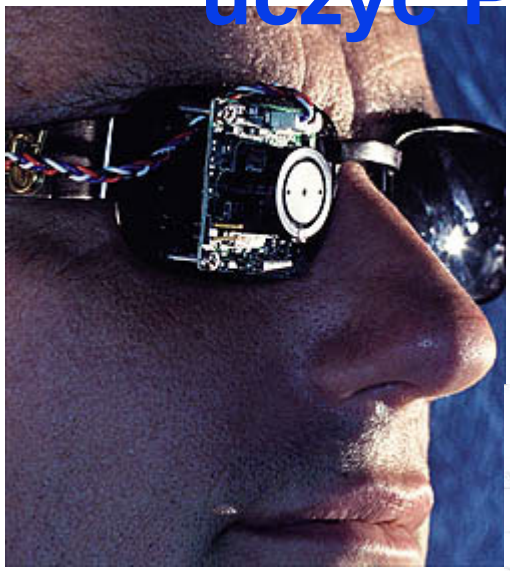
- **Bo to jest fajne! I będzie coraz fajniejsze!**
- **Komputery PC odpowiadają za sprzedaż mniej niż 1% wszystkich procesorów. Systemy wbudowane za ponad 99%.**

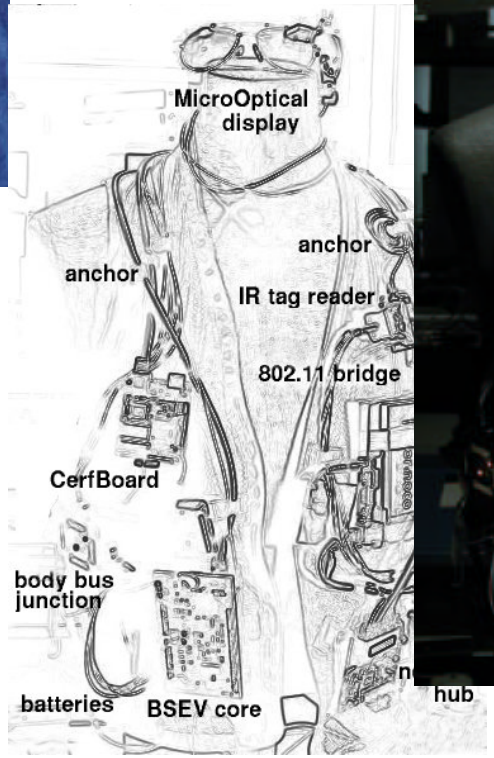
Bionika:
Czujniki w lateksowych palcach nieprzerwanie rejestrują temperaturę, electroniczny interfejs w sztucznej kończynie stymuluje zakończenia nerwów w ramieniu, które przekazują tę informację do mózgu.
Wart 3,000$ system pozwala czuć ciśnienie i ciężar. Po raz pierwszy od wypadku w 1986 roku ten strażak może podnieść szklankę bez zgniecenia jej i nie pozwalając jej się wyślizgnąć.

One Digital Day

Nie musisz mieć dyplomu lekarza, by pomagać ludziom.

Po co mam się uczyć PSM?





Plan wykładu

- Wprowadzenie.
- Mikrokontrolery
  - Architektura wewnętrzna
  - Zasada działania
- Programowanie mikrokontrolerów
- Komunikacja zewnętrzna i standardy komunikacyjne
  - Interfejsy szeregowe
  - Interfejsy równolegle
- Zastosowania z przykładami
- Mikrokontrolery w zaawansowanych systemach.

Literatura

- Ryszard Pełka, *Mikrokontrolery. Architektura, programowanie , zastosowania, WKŁ*
- Jarosław Doliński, *Mikrokontrolery AVR w praktyce, BTC*
- Piotr Gałka, Paweł Gałka, *Podstawy programowania mikrokontrolera 8051,* Mikom
- Tomasz Francuz, *Język C dla mikrokontrolerów AVR*, Helion (2011)
- Rafał Baranowski, *Mikrokontrolery AVR ATmega w praktyce, BTC*
- James M. Sibigtroth, *Zrozumieć małe mikrokontrolery,* BTC
- A. Pawluczuk, *Sztuka programowania mikrokontrolerów AVR, podstawy,* BTC
- Tomasz Jabłoński, *Mikrokontrolery PIC16F8x w praktyce,* BTC
- Jacek Bogusz, *Lokalne interfejsy szeregowe w systemach mikroprocesorowych,* BTC
- www.google.com
- Mirosław Kardaś, *Mikrokontrolery AVR, Język C*, Atnel (2011)

Regulamin

Średnia ważona ocen z lab. i wykładu z wagami odpowiednio: 2/3 i 1/3 (przy czym obydwie pozytywne)

Zaliczenie wykładu:

- **2 kolokwia po 6 pkt,**
- **Ocena: suma punktów uzyskanych z obydwu kolokwiów przeliczona następująco: 4-5pkt=3, 6-7pkt=3+, 8-9pkt=4, 10pkt=4+, 11-12pkt=5,**
- **1 poprawa kol. pod koniec zajęć w semestrze letnim.**

Zaliczenie lab:

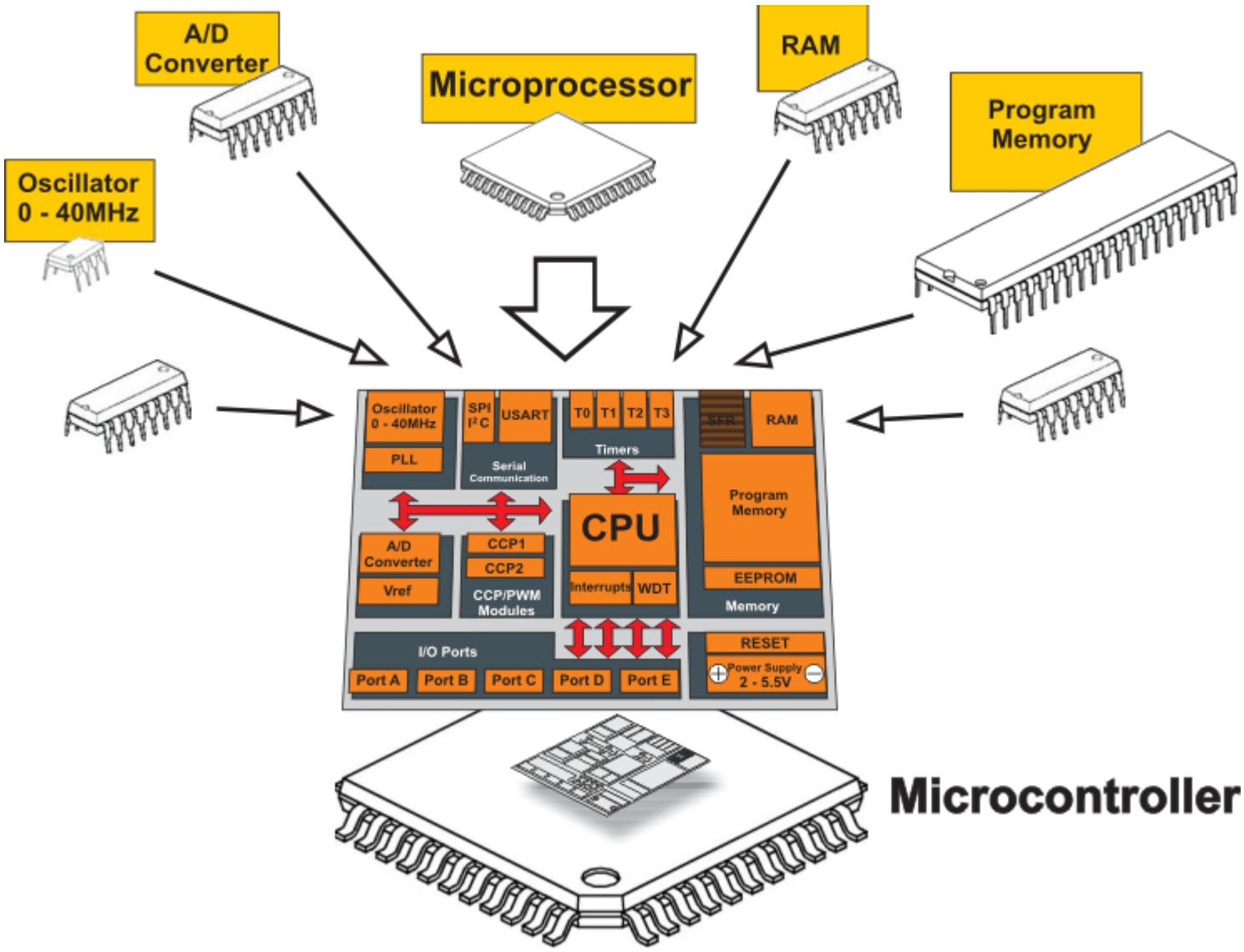
wykonanie wszystkich przewidzianych ćwiczeń, łącznie z miniprojektem ocena = 3.0
miniprojekt : ocena + 0.0, +0.5, +1.0
wykonanie projektu końcowego: ocena + 0.5
sprawozdanie w postaci dokumentacji technicznej wykonanego projektu: ocena + 0.5

Mikrokontroler i mikroprocesor

- **Mikroprocesor** to cyfrowy układ scalony, którego działanie jest sterowane pobieranymi z zewnątrz rozkazami. Ciąg rozkazów sterujących pracą mikroprocesora stanowi jego program.
- **Mikrokontroler** to układ scalony, w którego strukturze zawarto mikroprocesor oraz pewien zestaw elementów zewnętrznych (peryferyjnych)

	µP	µ C
Peryferia zewn.	**RAM, ROM, porty I/O, liczniki, wiele innych**	**Pamięć (opcjonalnie)**
Applikacje	**PC, laptopy, tablety**	**Systemy wbudowane, sterowniki, motoryzacja, kalkulatory, …**
Szybkość zegara	**4M-4G**	**32K-20M**
Zużycie prądu	**5 - 150 W**	**100µW – 2 W**
Koszt	**20-2500 USD**	**1-10 USD**
Przykład	**Motorola 68000, Intel x86**	**Intel 8051, Atmel AVR**

Architektura mikrokontrolera



O liczbach - dla przypomnienia…

Od dziś umiemy się posługiwać systemami binarnymi, szesnastkowymi (i oczywiście dziesiątkowymi).

Bit – cyfra w układzie binarnym (podstawowa jednostka informacji)

Bajt – słowo, liczba złożona z 8 bitów

$$E4 = \underbrace{1110}_{E}\underbrace{0100}_{4}$$

DEC.	BINARY								HEX.
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	1
2	0	0	0	0	0	0	1	0	2
3	0	0	0	0	0	0	1	1	3
4	0	0	0	0	0	1	0	0	4
5	0	0	0	0	0	1	0	1	5
6	0	0	0	0	0	1	1	0	6
7	0	0	0	0	0	1	1	1	7
8	0	0	0	0	1	0	0	0	8
9	0	0	0	0	1	0	0	1	9
10	0	0	0	0	1	0	1	0	A
11	0	0	0	0	1	0	1	1	B
12	0	0	0	0	1	1	0	0	C
13	0	0	0	0	1	1	0	1	D
14	0	0	0	0	1	1	1	0	E
15	0	0	0	0	1	1	1	1	F
16	0	0	0	1	0	0	0	0	10
17	0	0	0	1	0	0	0	1	11
.....									
.....									
.....									
253	1	1	1	1	1	1	0	1	FD
254	1	1	1	1	1	1	1	0	FE
255	1	1	1	1	1	1	1	1	FF

najbardziej znaczący bit
MSB

najmniej znaczący bit
LSB

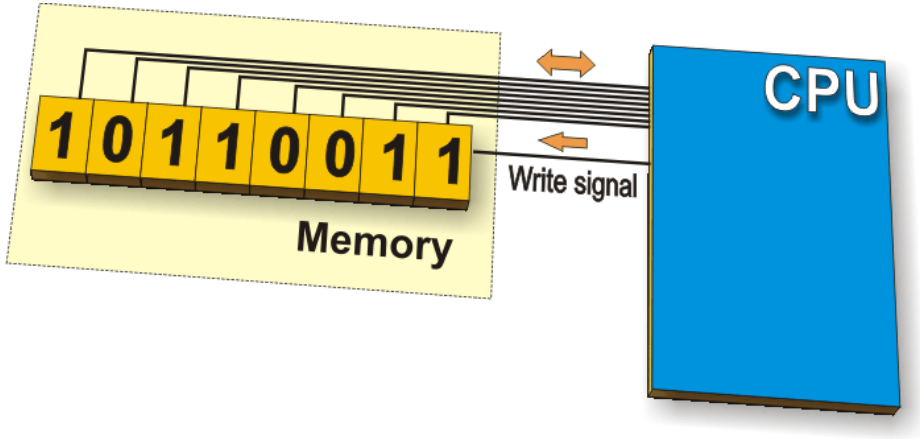
Zapis liczby w ukł. HEX

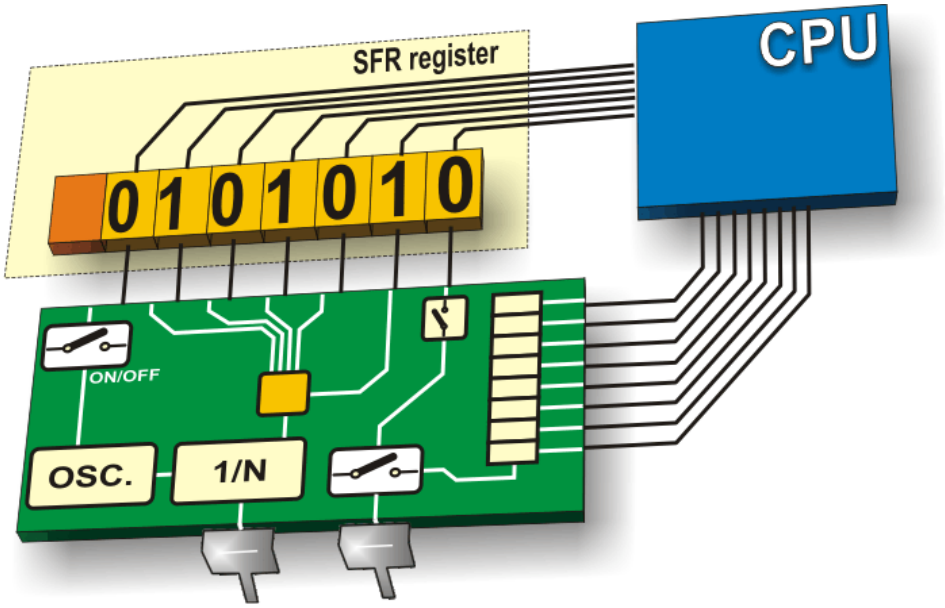
$10AF
0x10AF
0x10af
10AFh

Zapis liczby w ukł. BIN

%10010101
0b10010101

Rejestr – komórka pamięci, układ elektroniczny, który potrafi zapamiętać stan jednego bajta.





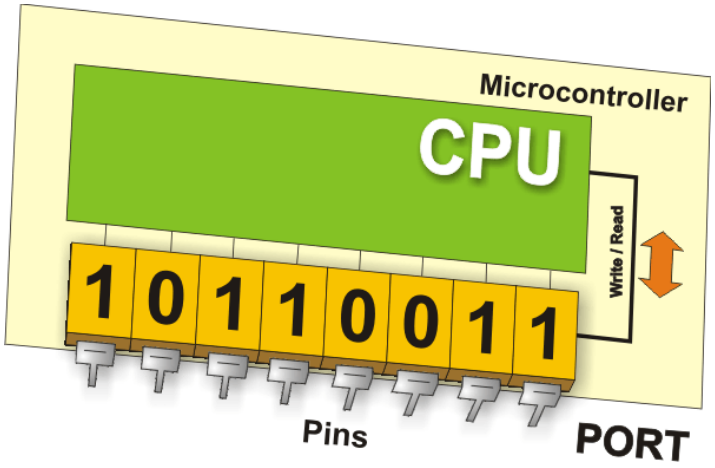
Rejestr specjalny (Special Function Register – SFR)

Oprócz rejestrów, które nie mają określonej z góry funkcji, każdy mikrokontroler posiada pewną liczbę rejestrów, których funkcja została zaprojektowana odgórnie przez producenta. Poszczególne bity tych rejestrów są dosłownie połączone z wewnętrznymi obwodami mikrokontrolera.

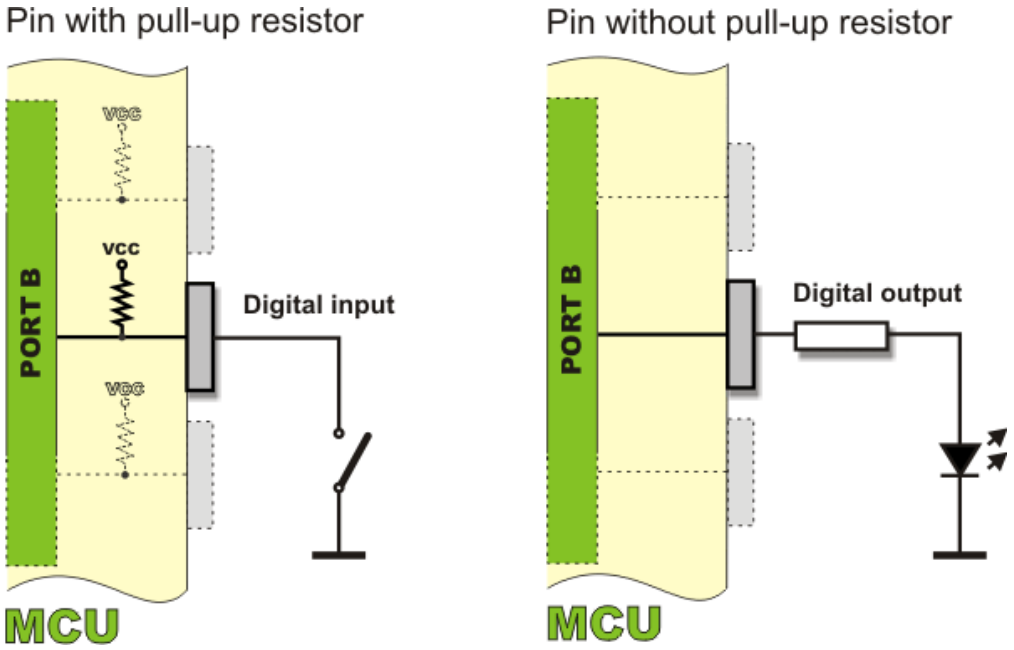
Porty wejścia/wyjścia

Rejestry podłączone do pinów (nóżek) mikrokontrolera.

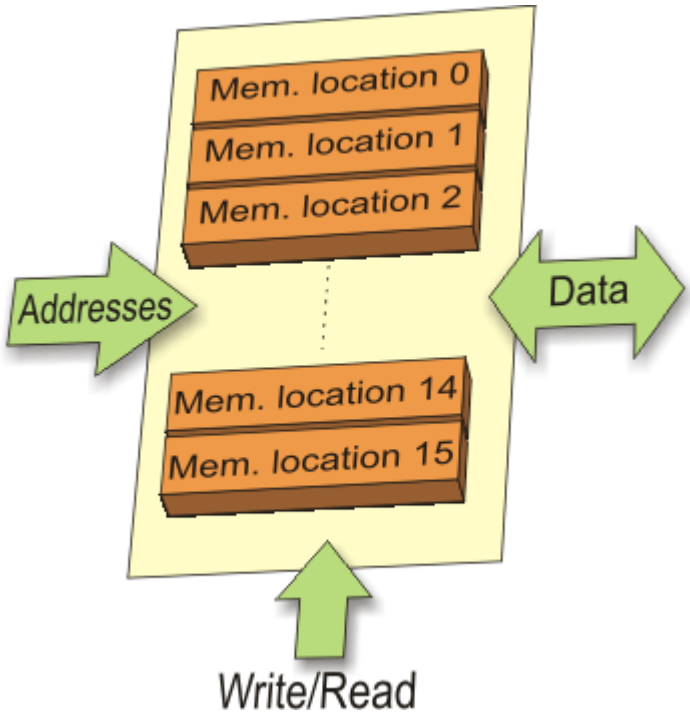
Funkcja pinów (wejście albo wyjście) jest również kontrolowana przez odpowiedni rejestr.



Mogą posiadać rezystory podciągające (czasem również kontrolowane przez odpowiedni rejestr).



Pamięć – część mikrokontrolera umożliwiająca przechowywanie danych i kodu programu.



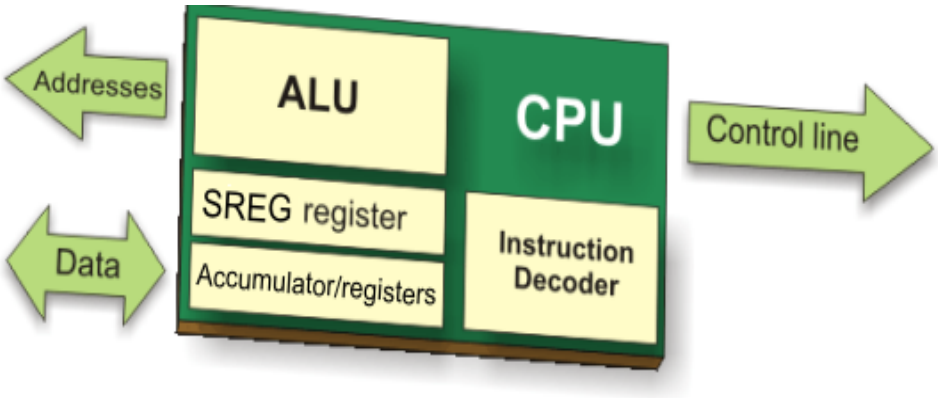
° **Pamięci nieulotne** (zawartość pamięci nie ulega skasowaniu w momencie wyłączenia zasilania)

- ROM – Read Only Memory - Pamięć tylko do odczytu
- EPROM – Erasable Programmable Read Only Memory
- EEPROM – Electrically Erasable Programmable Read Only Memory
- Flash

° **Pamięci ulotne** (zawartość pamięci ulega skasowaniu w momencie wyłączenia zasilania): SRAM, DRAM itp..

- SRAM – statyczne, droższe, mniej pakowne, szybsze;
- DRAM – dynamiczne, tańsze, bardziej pojemne, wolniejsze;

Centralna jednostka obliczeniowa (Central Processing Unit CPU)
Serce mikrokontrolera, monitoruje i kontroluje wszystkie procesy wewnątrz mikrokontrolera.



Dekoder instrukcji - na podstawie zawartości rejestru instrukcji, generuje odpowiednie sygnały sterujące dla automatu realizującego funkcje procesora. Zbiór instrukcji jest odmienny dla każdej rodziny mikrokontrolerów.

Jednostka arytmetyczno-logiczna (Arithmetical Logical Unit ALU) wykonuje wszystkie matematyczne i logiczne operacje na danych.

Wynik operacji jest umieszczany w dowolnym rejestrze (w starszych mikrokontrolerach tylko jeden taki rejestr - akumulator). Może być wiele ALU; ALU może zawierać FPU (Floating-Point Unit) do obliczeń zmiennoprzecinkowych

Rejestr SREG – jest rejestrem statusowym zawierającym informacje o rezultacie ostatnio wykonanej przez ALU operacji arytmetycznej (znak, przeniesienie, przepełnienie do dwóch, itd..)

Stos (stack) rodzaj pamięci danych (logiczny).

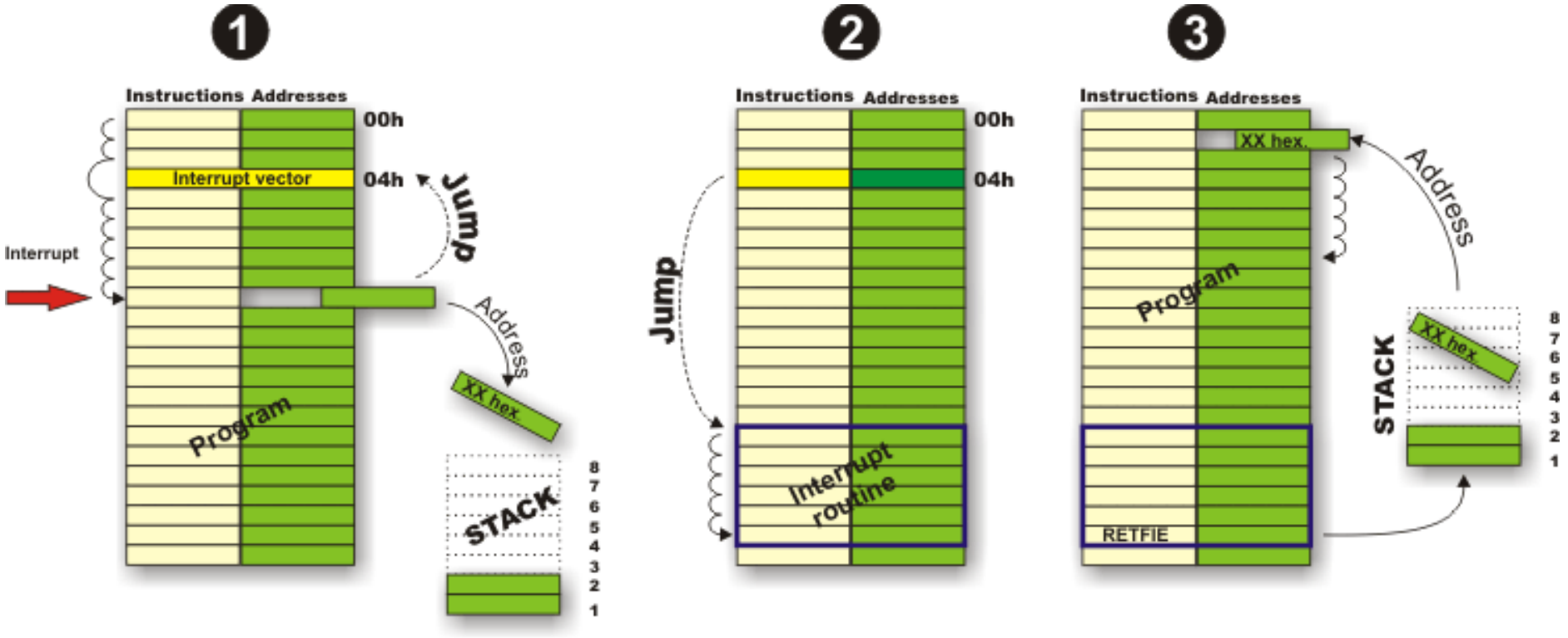
Pobieranie danych ze stosu musi być wykonywane w kolejności odwrotnej niż ich umieszczanie na stosie - element przesłany na stos jako ostatni musi być zeń zdjęty jako pierwszy. Tak więc w danej chwili możliwy jest dostęp do jednego tylko – ostatniego elementu stosu.

Podstawowym zastosowaniem stosu jest zapamiętywanie adresów powrotu podczas wywoływania procedur. Stos wykorzystywany jest też jako rodzaj podręcznej pamięci do chwilowego przechowywania danych.

Przy małych pojemnościach pamięci SRAM należy zwracać baczną uwagę, by odkładanie danych na stos (np. przy wykonywaniu wielokrotnie zagnieżdzonych skoków) nie powodowało jego rozszerzenia na obszar zmiennych programowych.

Przerwanie (Interrupt)

Idea przerwań w mikrokontrolerach polega na tym, że w odpowiedzi na określony sygnał (sygnał przerwania) mikrokontroler zawiesza chwilowo wykonywanie programu głównego i wykonuje specjalną procedurę określaną mianem **procedury obsługi przerwania**. Po zakończeniu tej procedury mikrokontroler wraca do wykonywania programu głównego, począwszy od miejsca, w którym zostało ono zawieszone.



- Podstawową funkcją przerwań jest umożliwienie szybkiego reagowania na zdarzenie zewnętrzne.
- Przerwania nie zawsze mają postać sygnałów zewnętrznych - mogą być one generowane także przez układy wewnętrzne, np. liczniki, łącza szeregowe.
- Stosowanie przerwań umożliwia maksymalne wykorzystanie możliwości przetwarzania danych, jakie daje mikrokontroler.

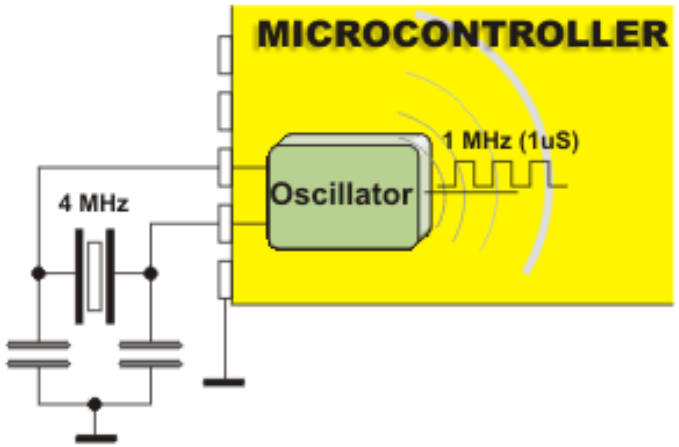
Zegar

Serce procesora wyznaczające rytm jego działania.

Każdy takt zegara – cykl zegarowy.

Do niedawna procesory wymagały kilku cykli zegarowych do wykonania jednej instrukcji.

Obecnie procesor może wykonać więcej niż jedną instrukcję na cykl (pipelining).



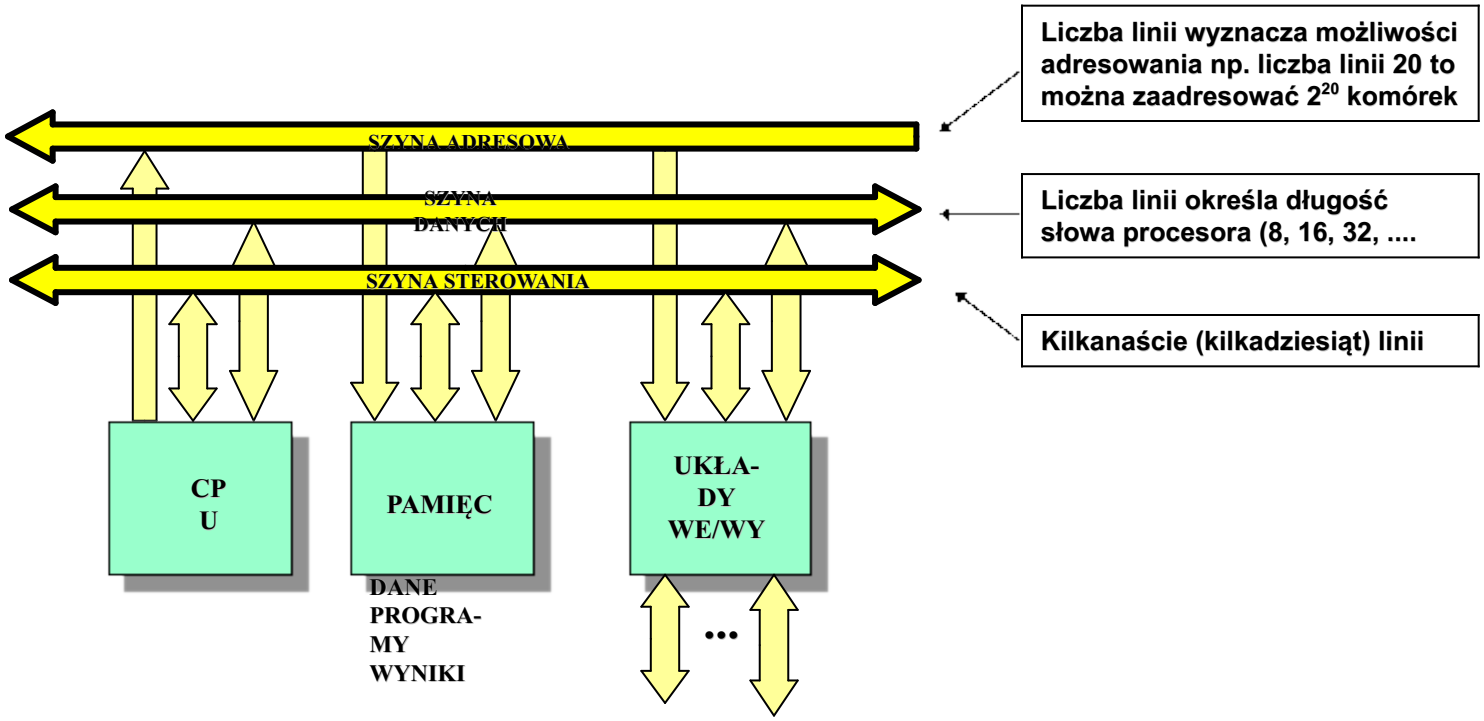
Magistrala (bus) – szyna komunikacyjna (zbiór linii, drucików)

Zawarte w magistralach linie można podzielić na trzy grupy funkcjonalne:
linie danych, adresów i sterowania.

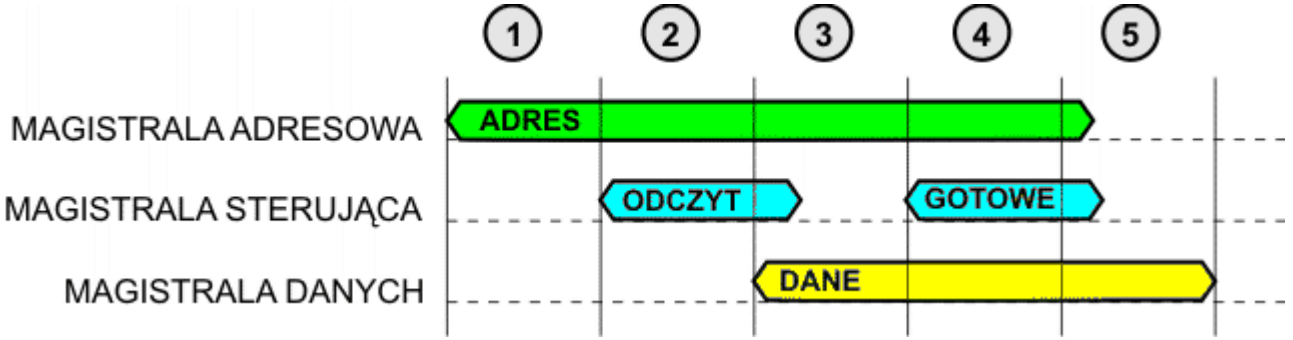
Linie danych (data bus) są ścieżkami służącymi do przenoszenia danych między modułami systemu. Szyna danych składa się typowo z 8, 16 lub 32 oddzielnych linii, przy czym liczba linii określa szerokość tej szyny. Ponieważ w danym momencie każda linia może przenosić tylko 1 bit, z liczby linii wnika, ile bitów można jednocześnie przenosić. Szerokość szyny danych jest kluczowym czynnikiem określającym wydajność całego systemu. Jeśli na przykład szyna danych ma szerokość 8 bitów, a każdy rozkaz ma długość 16 bitów, to procesor musi łączyć się z modułem pamięci dwukrotnie w czasie każdego cyklu rozkazu.

Linie adresowe są wykorzystywane do określania źródła lub miejsca przeznaczenia danych przesyłanych magistralą. Jeśli na przykład procesor ma zamiar odczytać słowo (8, 16 lub 32 bity) danych z pamięci, umieszcza adres potrzebnego słowa na linii adresowej. Szerokość szyny adresowej determinuje maksymalną możliwą pojemność pamięci systemu. Ponadto linie adresowe są również używane do adresowania portów wejścia-wyjścia.

Linii sterowania używa się do sterowania dostępem do linii danych i linii adresowych, a także do sterowania ich wykorzystaniem. Ponieważ linie danych i adresowe służą wszystkim zespołom, musi istnieć sposób sterowania ich używaniem. Sygnały sterujące przekazywane między modułami systemu zawierają zarówno rozkazy, jak i informacje regulujące czas (taktujące). Sygnały czasowe określają ważność danych i adresów. Sygnały rozkazów precyzują operacje, które mają być przeprowadzone.

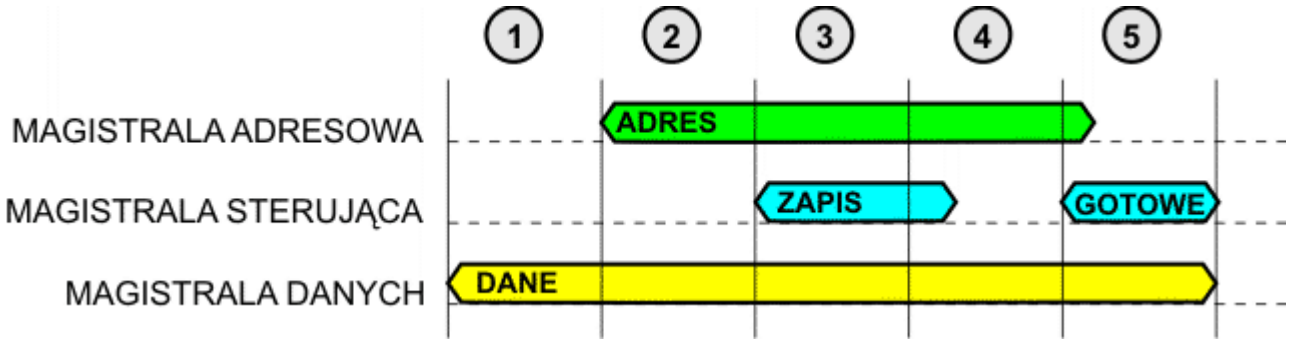


Gdy procesor chce odczytać dane z pamięci lub z urządzenia wejścia, to postępuje następująco:



1) Umieszcza na magistrali adresowej adres komórki pamięci lub numer urządzenia we/wy.
2) Na magistrali sterującej umieszcza informację, że chce odczytać dane z pamięci lub z urządzenia we/wy.
3) Pamięć lub urządzenie wejścia umieszcza dane na magistrali danych.
4) Pamięć lub urządzenie wejścia umieszcza na magistrali sterującej informację, że dane są gotowe do odczytu z magistrali danych.
5) Procesor odczytuje dane z magistrali danych.

Podobnie wygląda zapis do pamięci lub do urządzenia wyjścia:



1) Procesor umieszcza dane na magistrali danych.
2) Procesor umieszcza adres komórki pamięci lub numer urządzenia wyjścia na magistrali adresowej.
3) Procesor umieszcza na magistrali sterującej informację, że chce zapisać dane do pamięci lub urządzenia we/wy.
4) Pamięć lub urządzenie wyjścia odczytują dane z magistrali danych. Pamięć umieszcza dane pod adresem odczytanym z magistrali adresowej. Dla urządzeń wyjścia adres określa numer urządzenia, dla którego są przeznaczone dane.
5) Pamięć lub urządzenie wyjścia umieszczają na magistrali sterującej potwierdzenie odebrania danych.

Architektura instrukcji

CISC (Complex Instruction Set Computer):
- duża liczba rozkazów (instrukcji) (ponad 200)
- niektóre rozkazy potrzebują dużej liczby cykli procesora do wykonania
- występowanie złożonych, specjalistycznych rozkazów
- duża liczba trybów adresowania
-powolne działanie dekodera rozkazów
-Krótkie skompilowane programy

```
83 A6 0C 01 00 00 FE  and  dword ptr [esi+0x10C],0xFE
8D 4E EC              lea  ecx,[esi-0x14]
6A FF           push 0xFF
6A 01           push 0x1
FF 50 24              call [eax+0x24]
8B 86 80 00 00 00     mov  eax,[esi+0x80]
3B C7                 cmp  eax,edi
```

RISC (Reduced Instruction Set Computer):
- zredukowana liczba rozkazów (instrukcji) – nawet poniżej 30!
- większość rozkazów wykonywana w jednym cyklu procesora
- rozkazy proste lub bardzo proste (+, -, kopiowanie)
- bardzo mała liczba trybów adresowania
- ograniczony dostęp do pamięci
-szybkie działanie dekodera rozkazów
-skompilowane progamy dłuższe

```
80 E6     ldi  R24, 0x60
90 E0     ldi  R25, 0x00
9C 8B     std  Y+20, R25
8B 8B     std  Y+19, R24
98 E0     ldi  R25, 0x08
9D 8B     std  Y+21, R25
EB 89     ldd  R30, Y+19
FC 89     ldd  R31, Y+20
00 80     ld   R0, Z
4B 89     ldd  R20, Y+19
5C 89     ldd  R21, Y+20
4F 5F     subi R20, 0xFF
```