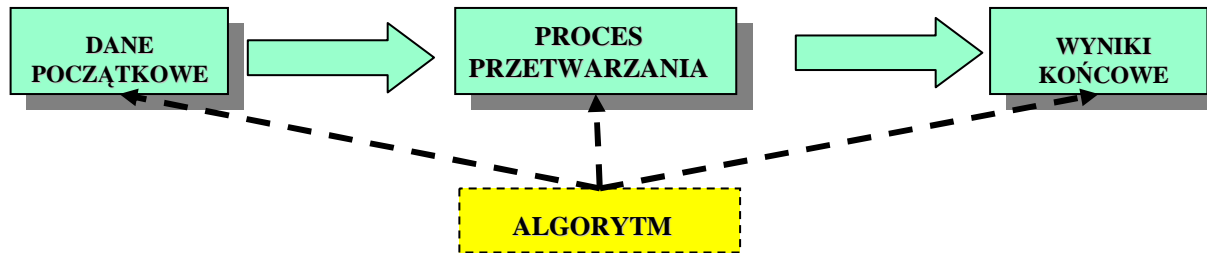


# **PODSTAWY ARCHITEKTURY KOMPUTERA**

- **Specjalizowany układ cyfrowy a maszyna cyfrowa**
- **Struktura systemu mikroprocesorowego**
- **Bloki funkcjonalne - organizacja i architektura**

# PRZETWARZANIE INFORMACJI

Wszelkie procesy zamierzonego przetwarzania informacji przebiegają według ustalonego algorytmu



Przetwarzanie informacji można zrealizować dwoma sposobami:

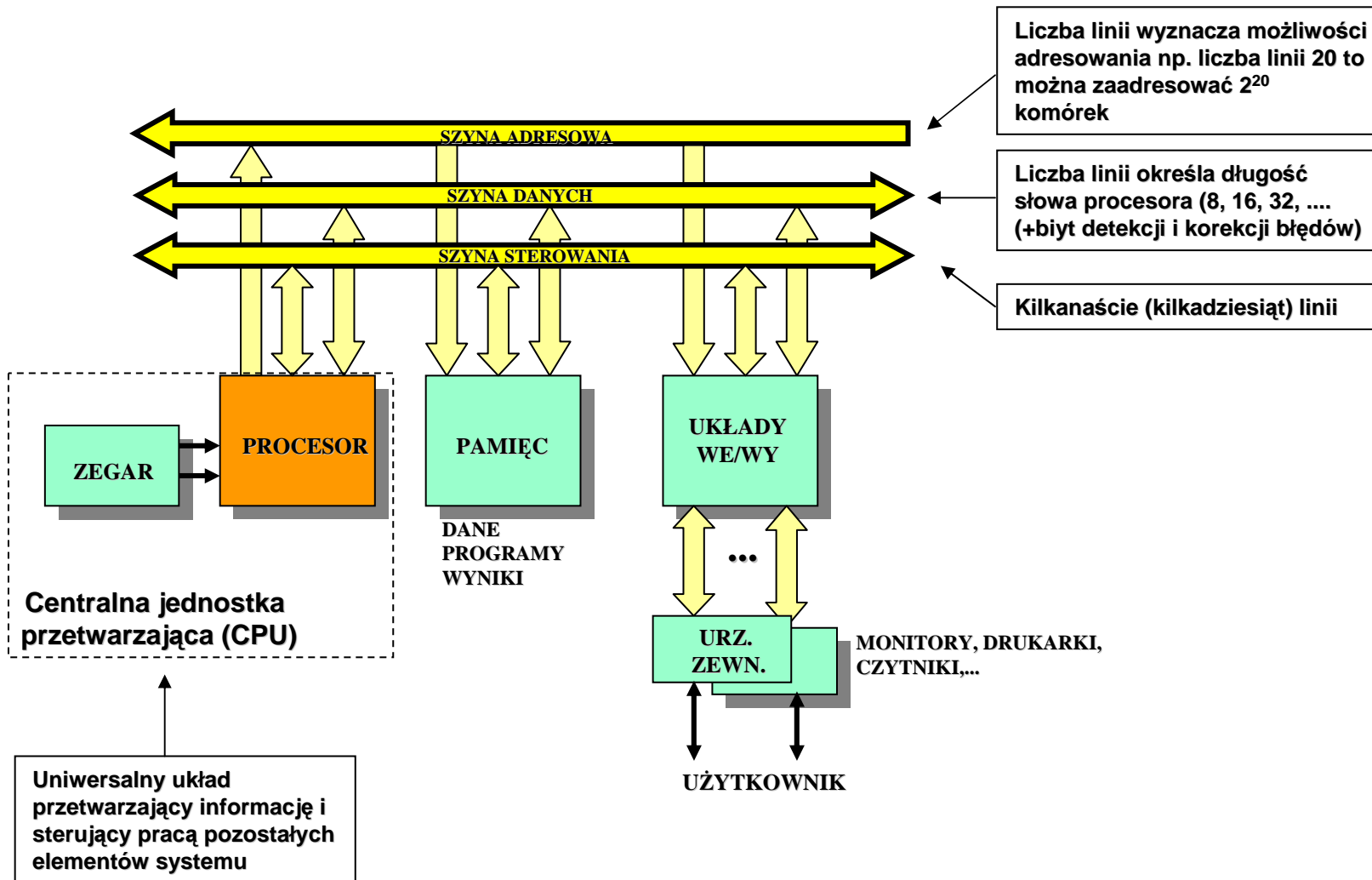
- Specjalizowany układ cyfrowy:



- System mikroprocesorowy (maszyna cyfrowa):



# SYSTEM MIKROPROCESOROWY



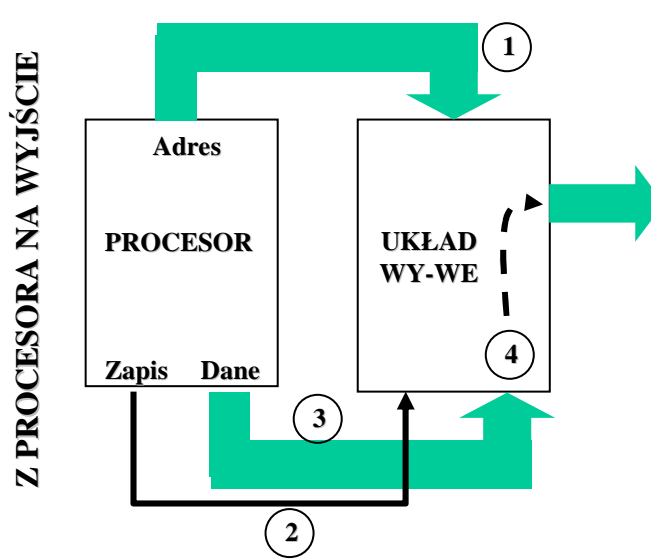
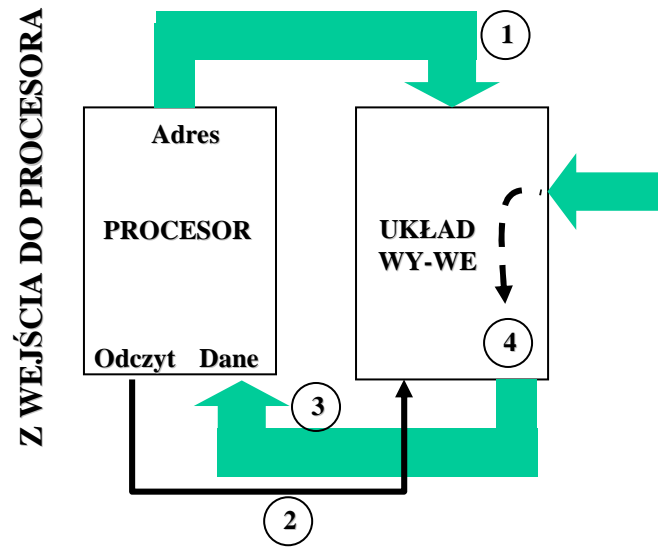
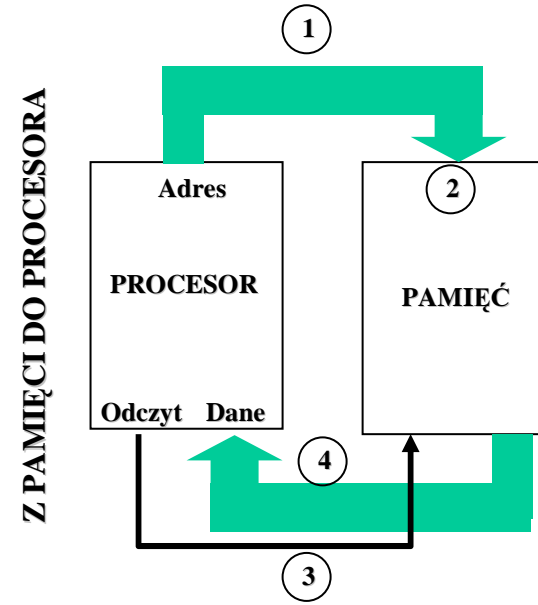
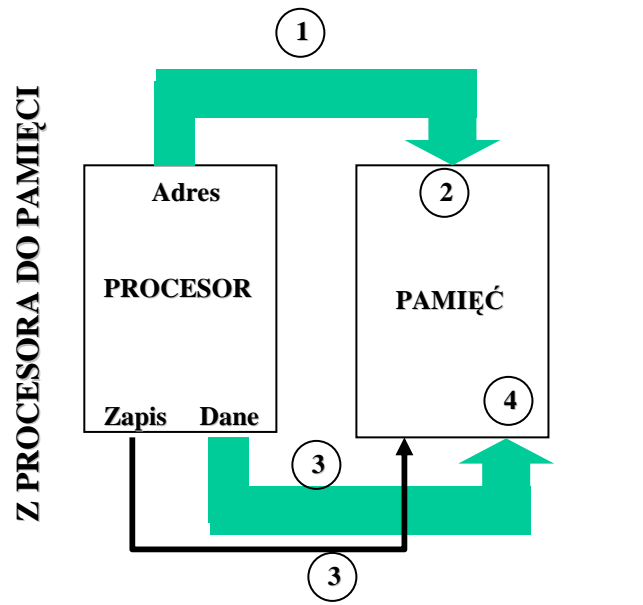
# ORGANIZACJA SYSTEMU

**Każde działanie wykonywane przez system jest wynikiem realizacji określonego programu bądź jego fragmentu**

**Architektura klasycznego komputera (wg von Neumana) opiera się na założeniach:**

- program wykonywany przez procesor wraz z danymi jest umieszczony w pamięci;**
- kolejność wykonywanych rozkazów zależy od ich umieszczenia w programie (w kolejnych komórkach pamięci), a zmiana tej zasady może być wykonana tylko:
  - przez program (rozkaz skoku);**
  - przez system operacyjny np. wykrycie błędu;**
  - przez operatora np. *reset*****
- procesor odczytuje kolejne rozkazy z pamięci wysyłając odpowiednie adresy**

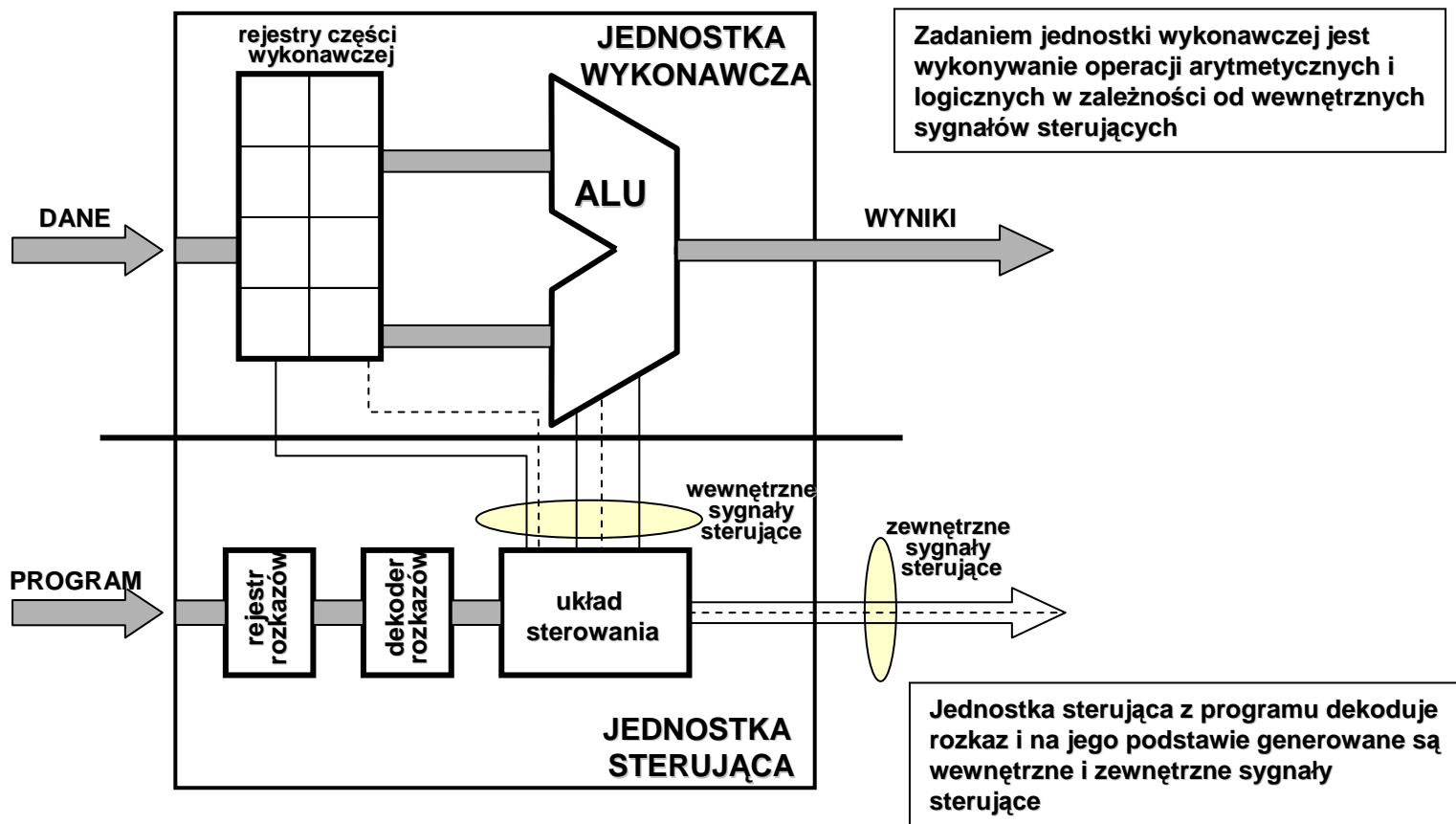
# PRZEPŁYW DANYCH W SYSTEMIE MIKROPROCESOROWYM

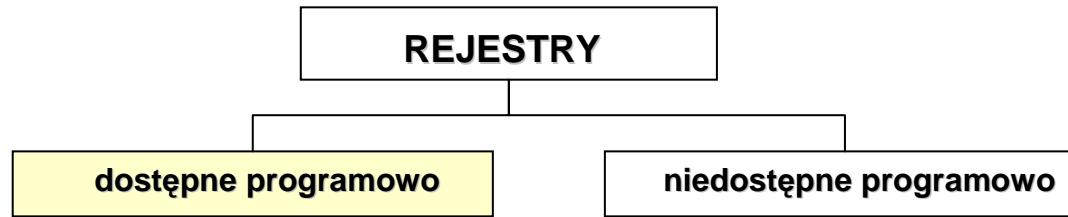


# ARCHITEKTURA PROCESORA

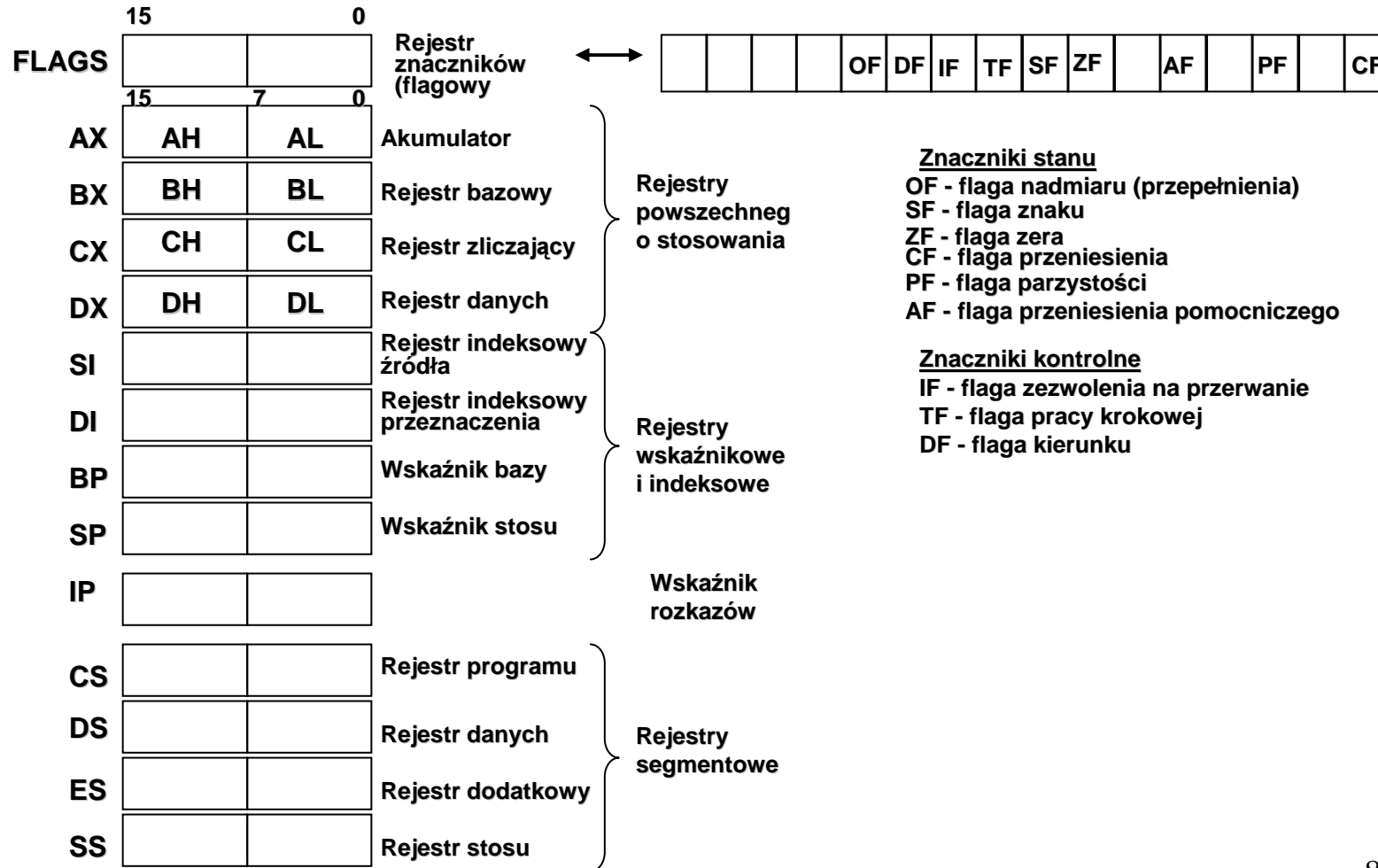
- **Schemat blokowy procesora**
- **Rejestry**
- **Blok arytmetyczno-logiczny (ALU)**
- **Układ sterowania**

# SCHEMAT BLOKOWY PROCESORA





(na przykładzie 8 bitowego procesora 8086/8088)





# REJESTRY

**Rejestr znaczników (flagowy)** - rejestr przeznaczony do przechowywania dodatkowych cech wyniku operacji (np. znak, przekroczenie zakresu, znak parzystości, ...) w postaci tzw. flagi (znacznika). Każdy znacznik jest bitem w rejestrze, który wskazuje czy określony stan wystąpił. Znaczniki mogą być wykorzystywane przez procesor lub programistę na dwa sposoby:

- ustawienie znacznika dla zapamiętania określonego stanu po wykonaniu rozkazu
- testowanie znacznika celem umożliwienia decyzji o sposobie dalszego postępowania (przetwarzania danych)

**Znaczniki stanu:**

- CF - flaga przeniesienia lub pożyczki, ustawiana przy przekroczeniu zakresu długości słowa, w którym zapisywany jest wynik (1 wystąpiło przeniesienie)
- PF - flaga parzystości (1) wskazuje na parzystą lub nieparzystą liczbę bitów o wartości jedynek
- ZF - flaga zera - sygnalizująca, że wynikiem ostatnio wykonywanej operacji jest zero
- DF - flaga przeniesienia pomocniczego jest wykorzystywana przy działaniach na liczbach w kodzie BCD. Ustawiana jest gdy następuje przeniesienie lub pożyczka z najstarszego bitu pierwszej tetrady wyniku
- SF - flaga znaku, ustawiana gdy najstarszy bit wyniku jest równy 1 (operacje liczb ze znakiem - kod U2)
- OF - flaga przepełnienia, sygnalizująca przekroczenie zakresu dla operacji arytmetycznych (operacje liczb ze znakiem - kod U2)

**Znaczniki kontrolne:**

- TF- znacznik pracy krokowej - ustawienie 1 pozwala na wykonanie przez procesor po wykonaniu każdego rozkazu przerwania i przejście do specjalnych procedur obsługi
- IF- znacznik zezwolenia na przerwanie - ustawienie 1 powoduje odblokowanie systemu przerw w procesorze (0 - procesor ignoruje przerwania)
- DF - znacznik kierunku - umożliwia realizację przetwarzania łańcuchów (ciągu słów) przy rosnących adresach (1) lub malejących adresach (0)

# REJESTRY

**Rejestry ogólnego przeznaczenia** - rejestry przeznaczone do przechowywania dowolnych danych i wykonywania operacji (arytmetycznych i logicznych), ale jednocześnie spełniające pewne funkcje specjalne. Szczególną funkcję pełni akumulator

**Akumulator** - rejestr przeznaczony do przechowywania jednego z operandów (argumentów) wykonywanej operacji oraz wyniku wykonywanej operacji (czasami wynik może być umieszczany w innym rejestrze)

**Rejestry segmentowe** - są rejestrami wykorzystywanymi do adresowania pamięci operacyjnej. Rejestry te zawierają adresy początkowe segmentów

- CS rejestr segmentowy programu wskazuje segment programu, z którego aktualnie są pobierane kolejne rozkazy do wykonania,
- DS rejestr segmentowy stosu wskazuje segment, w którym zapamiętane są zmienne używane w programie,
- ES rejestr segmentowy dodatkowy wskazuje dodatkowy segment danych,
- SS rejestr segmentowy stosu wskazuje segment pamięci, w którym zdefiniowany jest stos.

**Rejestry wskaźnikowe i indeksowe** - posiadają dwa rejestry wskaźnikowe i dwa rejestry indeksowe.

Rejestry wskaźnikowe (SP, BP) są stosowane do adresowania danych w obrębie wydzielonego obszaru pamięci (stosu).

Rejestry indeksowe (SI, DI) służą do adresowania danych w obszarze pamięci zwanym segmentem danych.

Wszystkie rejestry mogą być wykorzystywane jako argumenty większości rozkazów arytmetycznych i logicznych.

**Wskaźnik rozkazów (wskaźnik instrukcji)** - łącznie z rejestrem segmentowym CS adresuje kolejne rozkazy przeznaczone do wykonania. Wskazuje adres względem początku segmentu programu

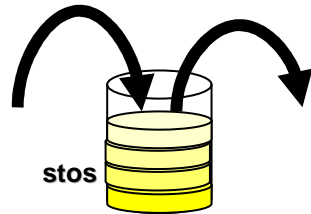
# REJESTRY

**Wskaźnik stosu(SP)** - rejestr zawierający adres ostatniej wypełnionej komórki stosu (wierzchołka stosu)

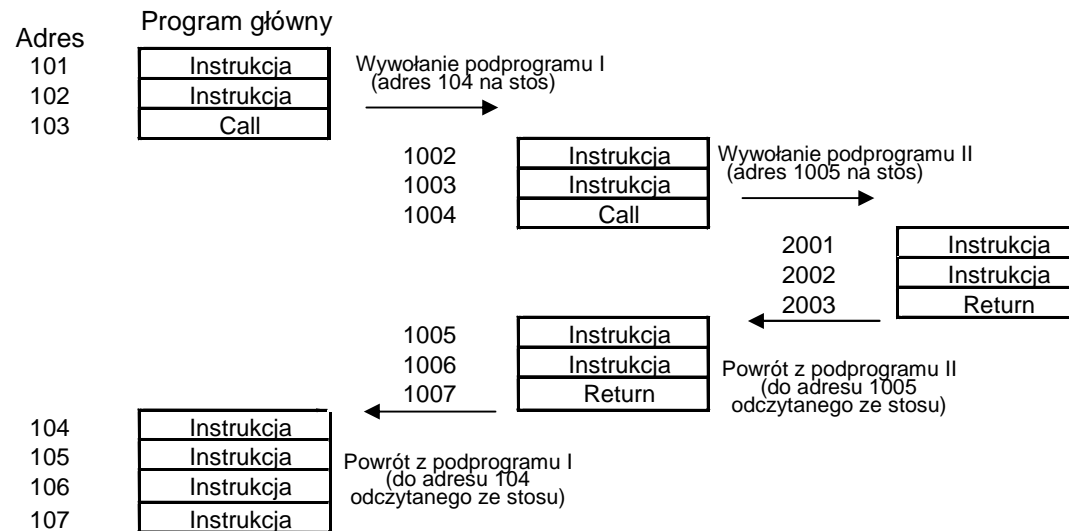
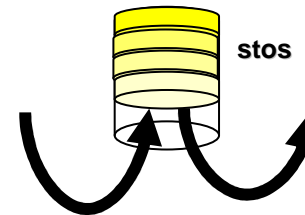
Stosem nazywamy wyróżniony obszar w pamięci używany wg regu:

- informacje zapisane są na stos do kolejnych komórek (pod kolejnymi adresami), przy czym żadnego adresu nie wolno pominąć
- odcytujemy informacje w kolejności odwrotnej do ich zapisu
- informacje odcytujemy z ostatnio wypełnionej komórki, natomiast zapisujemy do pierwszej wolnej

Czyli obowiązuje reguła LIFO - ostatni wchodzi pierwszy wychodzi



lub



# BLOK ARYTMETYCZNO-LOGICZNY

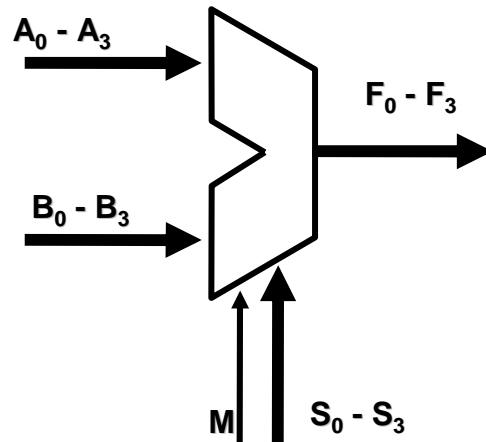
**Blok arytmetyczno-logiczny (ALU)** - jest uniwersalnym układem kombinacyjnym, który realizuje operacje matematyczne i logiczne w zależności od zaprogramowanej operacji tj. rozkazu umieszczonego w programie.

Rozkazy mogą dotyczyć

- operacji dwuargumentowych:
  - operacji arytmetycznych (dodawanie i odejmowanie)
  - operacji logicznych (sumowanie mnożenie, sumowanie mod 2, itp.)
- operacji jednoargumentowych (np. negowanie bitów lub przesuwanie zawartości rejestrów)

Argumentami rozkazów są najczęściej dwa słowa binarne, od długości których mówi się o liczbie bitów ALU.

## Przykład 4 bitowego ALU SN74181



Sygnaly sterujace do wyboru mikrooperacji logicznej (M=0) lub arytmetycznej (M=1)

Lp.	S <sub>0</sub> - S <sub>3</sub>	M=0	M=1
1	0000	$F = \bar{A}$	$F = A + C_0$
2	1000	$F = A \vee B$	$F = (A \vee B) + C_0$
3	0100	$F = \bar{A}B$	$F = (A \vee \bar{B}) + C_0$
4	1100	$F = 0$	$F = C_0 - 1$
5	0010	$F = \bar{A}B$	$F = A + \bar{A}B + C_0$
6	1010	$F = B$	$F = (A \vee B) + \bar{A}B + C_0$
7	0110	$F = A \oplus B$	$F = A - B - (1 - C_0)$
8	1110	$F = \bar{A}B$	$F = A - (1 - C_0)$
9	0001	$F = A \vee \bar{B}$	$F = A + AB + C_0$
10	1001	$F = \overline{A \oplus B}$	$F = A + B + C_0$
11	0101	$F = B$	$F = (A \vee \bar{B}) + AB + C_0$
12	1101	$F = AB$	$F = AB - (1 - C_0)$
13	0011	$F = 1$	$F = 2A + C_0$
14	1011	$F = A \vee \bar{B}$	$F = (AB) + A + C_0$
15	0111	$F = A \vee B$	$F = (A \vee \bar{B}) + A + C_0$
16	1111	$F = A$	$F = A - (1 - C_0)$

# UKŁAD STEROWANIA

Często (a może najczęściej) procesor wykonuje rozkazy nie w jednym kroku (jak np. dodawanie) ale w wielu krokach (np. mnożenie lub dzielenie jako ciąg dodawań i przesunięć).

W tym celu potrzebny jest złożony automat sekwencyjny, generujący odpowiednie ciągi słów podawanych na wejścia sterujące układu ALU - układ sterowania.

Cechy takiego automatu to:

- konieczność posiadania bardzo dużej liczby stanów dostosowanych do wymaganej liczby wykonywanych rozkazów;
- konieczność zapewnienia synchronizacji pracy układu sterowania i wykonawczego (uwzględnienie czasów wykonywania poszczególnych operacji).

W praktyce realizowane są jako:

- generatory sekwencyjne
- układy mikroprogramowalne