

# Rejestry

W procesorach x86 znajdziemy następujące 16 bitowe rejestry:

## Rejestry ogólnego przeznaczenia:

Służą do wykonywania różnych operacji.

- **AX** (Accumulator)- składający się z dwóch 8 bitowych rejestrów AH i AL. Jest podstawowym rejestrem , używanym głównie do wykonywania operacji arytmetycznych.
- **BX** (base register), rejestr odpowiadający za adresowanie przestrzeni danych.
- **CX** (count register), rejestr służący jako licznik przy operacjach wykonywanych w pętli.
- **DX** (data register), rejestr używany do operacji wejścia/wyjścia. Może być także użyty razem z rejestrem AX przy wykonywaniu operacji arytmetycznych, w których są duże liczby.

## Segmentowe:

Umożliwiają wskazanie fragmentów pamięci zawierających pewien kod maszynowy.

- **CS** (Code Segment) – Przechowuje wszystkie instrukcje, które mają zostać wykonane. Wskazuje na początek segmentu danych przydzielonego naszemu programowi.
- **DS** (Data Segment)– Zawiera dane, stałe oraz obszar roboczy. Przechowuje adres początkowy segmentu danych.
- **SS** (Stack Segment)– Zawiera dane oraz zwracanie adresy procedur. Zaimplementowany jako stos. Przechowuje adres początkowy stosu.

## Pozostałe rejestry ogólnego przeznaczenia:

- **Instruction Pointer (IP)** – Rejestr przechowujący przesunięte adresy instrukcji, które będą wykonywane. W połączeniu z rejestrem CS (CS:IP) otrzymujemy kompletny adres aktualnej instrukcji przechowywanej w CS.
- **Stack Pointer (SP)** – Rejestr zapewnia przesunięte wartości w stosie programu. W połączeniu z SS (SS:SP) odnosi się do aktualnej pozycji danych lub adresu w stosie danych.
- **Base Pointer (BP)** – Rejestr pomagający odwoływać się do zmiennych wysyłanych do podprogramu. Adres w rejestrze SS łączony z przesunięciem daje lokalizację parametru. Może zostać połączony z SI oraz DI jako podstawowy rejestr.
- **Source Index (SI)** – Jest używany jako indeks źródłowy dla operacji na stringach.
- **Destination Index (DI)** – Jest używany jako indeks docelowy dla operacji na stringach.

## Rejestry flag:

- **Overflow Flag (OF)** – Wskazuje przepełnienie najbardziej znaczącego bitu danych po operacji arytmetycznej.
- **Direction Flag (DF)** – Określa odpowiedni kierunek (lewo lub prawo) do przesunięcia lub porównania danych w stringu. Kiedy DF ma wartość 0, operacja wykonuje się od lewej do prawej, a gdy ma wartość 1, odwrotnie.
- **Interrupt Flag (IF)** – Określa czy działania zewnętrzne typu wciśnięcie klawisza na klawiaturze powinno być ignorowane czy przetwarzane. Gdy wartość wynosi 0, wtedy są wyłączone. Gdy 1 - włączone.
- **Trap Flag (TF)** – Pozwala ustawić operacje procesora na bycie wykonywanym krok po kroku. Dzięki czemu możemy przejść przez każdą instrukcję w czasie jej wykonywania.
- **Sign Flag (SF)** – Pokazuje znak wyniku operacji arytmetycznej. Flaga jest ustawiana w zależności od znaku obiektu kolejnej operacji arytmetycznej. Znak jest wskazywany przez najbardziej znaczący bit. Pozytywny wynik zmienia wartość na 0, a negatywny na 1.
- **Zero Flag (ZF)** – Wskazuje wynik operacji arytmetycznej lub operacji porównania. Wynik inny niż 0 zmienia wartość na 0, a wynik 0 zmienia wartość na 1.

- **Auxiliary Carry Flag (AF)** – Obejmuje przeniesienie z bitu 3 do bitu 4 kolejnej operacji arytmetycznej. Używany w specjalistycznej arytmetyce. Wartość wynosi 1 gdy 1 bajtowa operacja powoduje przeniesienie bitu z 3 na 4.
- **Parity Flag (PF)** – Wskazuje całkowitą liczbę pojedynczych bitów w wyniku otrzymanym z operacji arytmetycznej. Parzysta liczba ustawia wartość flagi na 0, a nieparzysta na 1.
- **Carry Flag (CF)** – Obejmuje przeniesienie 0 lub 1 z najbardziej znaczącego bitu po operacji arytmetycznej. Przechowuje również zawartość ostatniego bitu z operacji. Wartość wynosi 1 gdy w wyniku dodawania lub odejmowania przekroczono możliwy zakres wartości zmiennej

## Rejestry 32-bitowe

W procesorach z rodziny 80x86 jest 10 rejestrów 32 bitowych, które dzielą się na 3 kategorie:

- rejestry ogólnego przeznaczenia,
- -rejestry kontrolne,
- -rejestry segmentowe

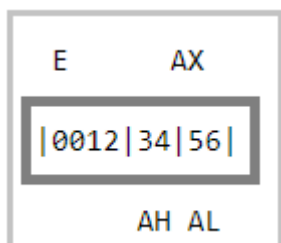
Rejestry ogólnego przeznaczenia dzielą się na kolejne podgrupy:

- rejestry danych,
- rejestry wskaźnikowe,
- rejestry indeksowe.

Są cztery 32-bitowe rejestry danych:

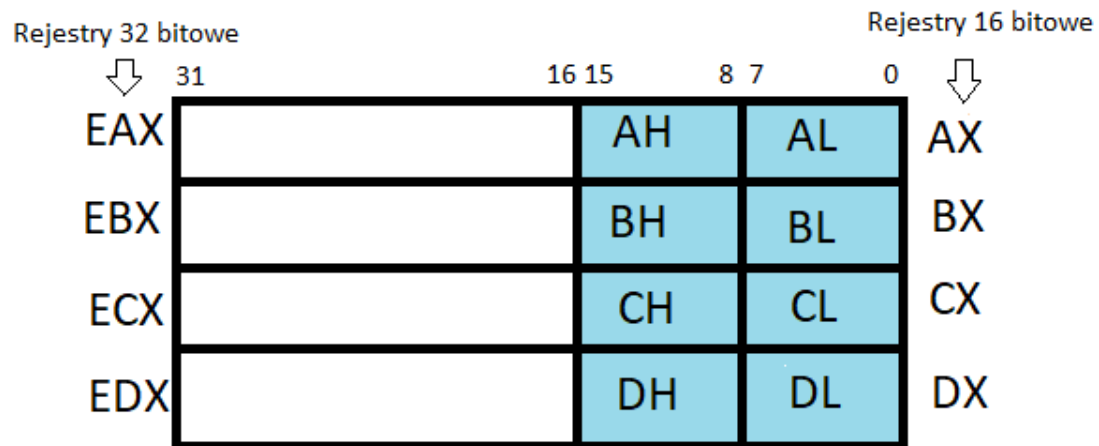
EAX, EBX, ECX, EDX - używane głównie do działań arytmetycznych.

Budowa rejestru:

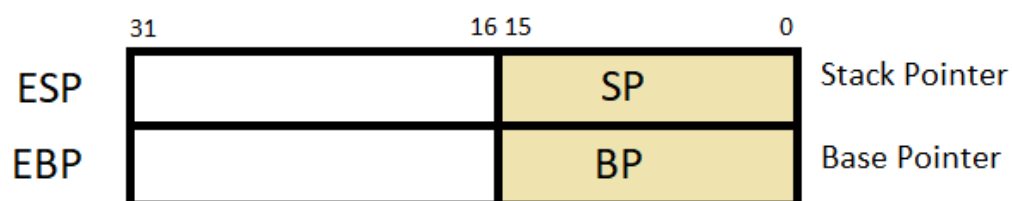


Mogą być użyte na 3 sposoby:

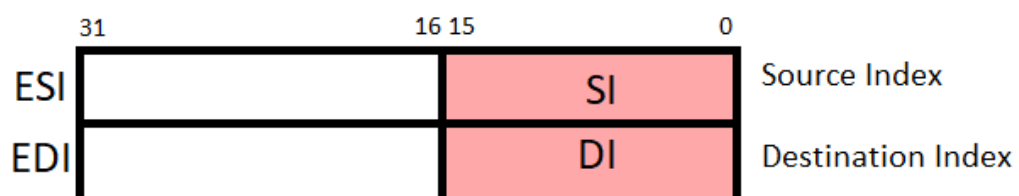
- jako kompletny 32 bitowy rejestr,
- dolne połówki 32 bitowych rejestrów mogą być użyte jako 16 bitowe rejestry,
- dolne oraz górne połówki 16 bitowych rejestrów mogą być użyte jako 8 bitowe rejestry.



Rejestry wskaźnikowe to:  
EIP, ESP, EBP



Rejestry indeksowe:  
ESI, EDI - używane do adresowania i czasem przy dodawaniu oraz odejmowaniu.



Ostatnim 32 bitowym rejestrem jest rejestr flag, który w połączeniu z EIP stanowi rejestr kontrolny.

# Rejestry 64-bitowe

W procesorach 64-bitowych do rejestrów o długości 64 bitów odwołuje się poprzez nazwę z przedrostkiem **R** zamiast **E** – np: **RAX**, **RBX** zamiast **EAX**, **EBX**

