|  |  |  |
| --- | --- | --- |
| **1. Stos** jest pamięcią typu LIFO, znajudje się on w pamięci procesora (RAM) oraz w procesorze 8086 jego maksymalny rozmiar to 64 kB. PUSH – wysyła 2 bajty danych na stos CALL – ustawia adres powrotu na stosie i wykonuje skok do podanego adresu 2. 2.**Kolejka adresowa** służy do przyspieszenia wykonywania instrukcji. Jest ona wykorzystywana do przechowywania następnych rozkazów, które mają zostać wykonane przez procesor.  **3.Procesor 8086** składa się z 9 flag: **C (Carry)** – oznacza przeniesienie w przypadku przeniesienia binarnego na ostatnim bicie**. P (Parity)** –wynik operacji ma parzystą liczbę ustawionych bitów w swoim binarnym zapisie **A (Auxiliary-Carry)** – w trackie operacji wystąpiło przeniesienie z 7 na 8 bit. Z (Zero) – oznacza, że wynik operacji jest równy zero **S (Sign)** – oznacza czy wynik operacji jest dodatni czy ujemny **O (Over-flow)** – sygnalizuje przepełnienie, wartość BL pomnożona razy AL jest większa niż 65535, jest ona ustawiana przykładowo przez MUL BL **T (Trap)** – powoduje, że procesor przechodzi w tryb jednokrokowy. Wykonuje jedną instrukcję po czym zatrzymuje się oczekując na kolejne instrukcje **I (Interrupt enable)** – oznacza, że procesor będzie ignorował żądania przerwań zewnętrznych **D (Direction)** – wpływa na kierunek przetwarzania danych, jeśli jest ustawiona to instrukcje przetwarzające dane wykonują się w trybie dekrementacji Przykładowe rozkazy korzystające z flag: SBB B – odejmowanie od A wartości B oraz flagi przeniesienia JZ xyz skok do etykiety „xyz” jeśli flaga Z jest ustawiona na 1. | **ALU – (Arithmetic Logical Unit)** – jednostka arytmetyczno-logiczna wykonująca operacje arytmetyczne (np. dodawanie, odejmowanie) lub operacje logiczne (np. negacja, porównanie) na dwóch liczbach. ALU jest podstawowym blokiem centralnej jednostki obliczeniowej komputera. Posiada 2 wejścia dla operandów (czasem wykorzystywane jest tylko jedno np. dla negacji arytmetycznej lub logicznej) oraz 1 dla wyboru rodzaju przeprowadzanej operacji. Posiada również wyjście z wynikiem oraz wyjście dla wartości wskaźników rejestru flagowego.  **TRYBY ADRESACJI ->**  5**. Adresacja natychmiastowa** (immediate addressing) – umieszczenie danych w pamięci następuje bezpośrednio po kodzie maszynowym operacji.  **Rozkazy 8086** MOV AX, 1234h ADD AX, 10 **Rozkazy 8080** MVI A, 12h ADI 10  **Adresacja bezpośrednia** (direct addressing) – polega na podaniu bezpośrednio w instrukcji adresu w pamięci, pod którym znajduje się wartość do odczytania lub zapisu.  **Rozkazy 8086** MOV AX, BX ADD BX, CX **Rozkazy 8080** MOV A, B ADD B  **Adresacja pośrednia** (indirect addressing) – polega na odwołaniu się do pamięci, w której znajduje się adres interesującego nas elementu. Adres ten jest zwykle przechowywany w rejestrze.  **Rozkazy 8086** MOV AX, [BX] ADD AX, [BP] **Rozkazy 8080** MOV M, B ADD M | **Adresacja indeksowa** (indexed addressing) – polega na dodaniu do adresu pamięci wartości przechowywanej w jednym z rejestrów indeksowych (np. SI, DI). **Rozkazy 8086** MOV AX, [BX+DI] ADD AX,[BP+SI]  **Adresacja względna** (relative addressing) – polega na określeniu adresu przez przesunięcie (offset) względem aktualnej wartości rejestru wskazującego na adres. **Rozkazy 8086** MOV AX, [BP] + 3, ADD AX, [BX] + 2  6. **Rejestry 8080:** Rejestry ogólnego przeznaczenia: A, B, C, D, E, H, L, Rejestry dodatkowe: RF (rejestr flagowy), RP (rejestr pomocniczy), RR (rejestr rozkazu), BMD (bufor magistrali danych), BMA (bufor magistrali adresowej), PC (Program Counter), SP (Stack Pointer), możemy je łączyć w następujące pary: RF oraz A, BC, DE, HL  **Rejestry 8086**: Ogólnego przeznaczenia: AX, BX, CX, DX Rejestry adresacji: SP, BP, SI, DI Rejestry segmentowe: CS, SS, DS, ES Rejestr flagowy: flagi (C, P, A, Z, S, O, T, I, D)  **FLAGI w 8080 ->** SF - flaga znaku, ZF – flaga zera, AC – flaga przeniesienia połówkowego, PF – flaga parzystości, CY – flaga przeniesienia binarnego  **FORMAT ROZKAZU 8086: Pierwszy bajt** Zawiera od 7-2 bitu kod rozkazu, następnie 1 bit D (Direction) oznacza kierunek przesyłanych danych (z 1 do 2 operandu lub z 2 do 1 operandu), później 0 bit W (Word) przyjmuje wartość 0 gdy używamy słowa 8-bit lub 1 gdy używamy słowa 16-bit**) Drugi bajt** Bity 7 i 6 to MODE, czyli tryb pracy Bity 5-3 zawierają REG (adres operandu operacji), pole to wskazuje rejestr zawierający pierwszy argument w operacjach dwuargumentowych Bity 2-0 zawierają pole R/M (wskazujące drugi operand operacji), wskazuje ono rejestr drugiego argumentu lub sposobu obliczania adresu logicznego |

|  |  |  |
| --- | --- | --- |
| 8**.Sygnały sterujące: HLT** – sygnał służący do zatrzymania pracy procesora**.**  **INT** – służy do inicjowania przerwań programowych. Po otrzymaniu sygnału INT przechodzi do trybu obsługi przerwań.  **9. Opisać magistrale 8080: Magistrala danych** – służy do przesyłania danych między procesorem a pamięcią operacyjną lub innymi urządzeniami. **Magistrala sterująca** – Magistrala sterująca służy do kontrolowania operacji przesyłania danych i adresów między procesorem a pamięcią lub innymi urządzeniami. **Magistrala adresowa** – służy do przesyłania adresów pamięciowych, które wskazują na miejsce w pamięci, gdzie znajdują się dane lub instrukcje do pobrania. W procesorze 8086 magistrala adresowa jest 20-bitowa. Dzięki temu procesor może obsłużyć do 1 MB (2 ^20) pamięci operacyjnej. **Budowa magistral**: Magistrala zbudowana jest z dużej ilości linii (w 8086 z 20 linii), na końcu każdej magistrali znajdują się zwrotnice.  **WYLICZANIE ADRESÓW ROZKAZÓW:** W adresacji indeksowanej możemy użyć rejestrów (BX, BP, SI, DI) w połączeniu z rejestrami segmentowymi lub możemy połączyć je w pary BX SI/DI LUB BP SI/DI. Adres fizyczny jest oblicznay przez dodanie zawartości adresowej segment  **12. RODZAJE ADRESÓW: Adres fizyczny** – adres fizyczny składa się z 20 bitów i wskazuje on konkretną komórkę w pamięci. Aby obliczyć adres fizyczny tzreba dodać adres logiczny i efektywny**. Adres logiczny** – adres początku segmentu, znajdują się one w CS, DS, SS i ES **Adres efektywny** – adres komórki pamięci wewnątrz segmentu (16 bitowy) | 13. **WYMIENIĆ CYKLE ROZKAZOWE:**  **Cykl pobierania rozkazu** (fetch cycle) – procesor pobiera rozkaz z pamięci operacayjnej do wewnętrznego bufora instrukcji.  **Cykl dekodowania rozkazu** (decode cycle) – procesor interpretuje rozkaz, analizując jego kod operacji, argumenty i tryb adresowania.  **Cykl wykonania rozkazu** (execute cycle) – procesor wykonuje operację określoną przez rozkaz.  **Cykl zapisu wyniku** (write back cycle) – procesor zapisuje wynik operacji w pamięci operacyjnej lub rejestrach procesora.  **Lista rozkazów procesora 16-to bitowego.** • Rozkazy transmisji danych  MOV - prześlij bajt lub słowo,  PUSH - prześlij słowo na szczyt stosu,  • Rozkazy arytmetyczne i logiczne  ADD - dodawanie bajtów lub słów 16-bitowych,  SUB - odejmowanie bajtów lub słów 16-bitowych,  • Rozkazy przetwarzania tablic lub bloków danych  STOS - zapamiętaj element tablicy lub bloku znajdujący się w akumulatorze,  CMPS - porównaj elementy dwóch łańcuchów.  • Rozkazy skoków  JMP - zwykły,  CALL - do podprogramu,  • Rozkazy przerwań  INT - skok do podprogramu obsługi przerwania,  INTO - jak wyżej, gdy wystąpi przepełnienie,  • Rozkazy sterujące procesorem  STC - ustaw znacznik przeniesienia,  CLC - zeruj znacznik przeniesienia  • Pętle programowe  LOOP - powtórz sekwencję rozkazów jeśli zawartość CX ≠ 0  JCXZ - skok względny, gdy CX = 0 | **Lista rozkazów procesora 8-mio bitowego.**  r – rejestry 8-bitowe ; rr – 16 bitowe pary rejestrów ;  M – komórka pamięci wskazywana przez  HL ; nn – liczba 16 bitowa ; n – liczba 8 bitowa  **• Przesłania 8-bitowe**  MOV r,r1 z r1 do r r:=r1  MOV M,r z r pod adres wskazywany przez HL  **• Przesłania 16-bitowe**  PUSH B odkłada na stos BC  POP B zdejmuje ze stosu BC  **• 8-bitowe rozkazy arytmetyczne i logiczne (modyfikują rejestr flagowy)**  ADD r A zwiększone o r  CMP r porównanie A z r  **• 16-bitowe rozkazy arytmetyczne (nie modyfikują rejestru flagowego)**  INX rr rr++  DCX rr rr--  **• Obroty (modyfikują rejestr flagowy)**  RLC obrót bitów akumulatora w lewo (z modyfikacją CY)  RRC obrót bitów akumulatora w prawo (z modyfikacją CY)  **• Rozgałęzienia**  JMP nn skocz pod adres nn  PCHL skocz pod adres zawarty w HL  **• Rozkazy I/O**  IN n czytaj bajt z portu n  OUT n zapisz bajt na port n  **• Rozkazy sterujące**  DI zakaz przyjmowania przerwań  EI zezwolenie na przerwanie |