PROJEKT

Emulator procesora Intel 8086

Paweł Kęska nr.albumu 13075

1.Wstęp

Program emulator procesora Intel 8086 napisany w języku C#. Emulator pozwala na symulacje rozkazów MOV oraz XCHG dla rejestrów oraz rejestrów i pamięci w trybach adresacji indeksowym, bazowym oraz indeksowo-bazowym. Pozwala również na operacje PUSH oraz POP na stosie.

2.Opis symulatora

Intel 8086 to 16 bitowy mikroprocesor wprowadzony na rynek 8 czerwca 1978r. Został zaprojektowany w technologii 3 μm jako rozszerzenie 8 bitowego procesora 8080 oraz 8085. Wykonywany głównie w obudowach 40pin. Do dzisiaj jest produkowany przez niektórych dostawców.

3.Architektura użyta w projekcie

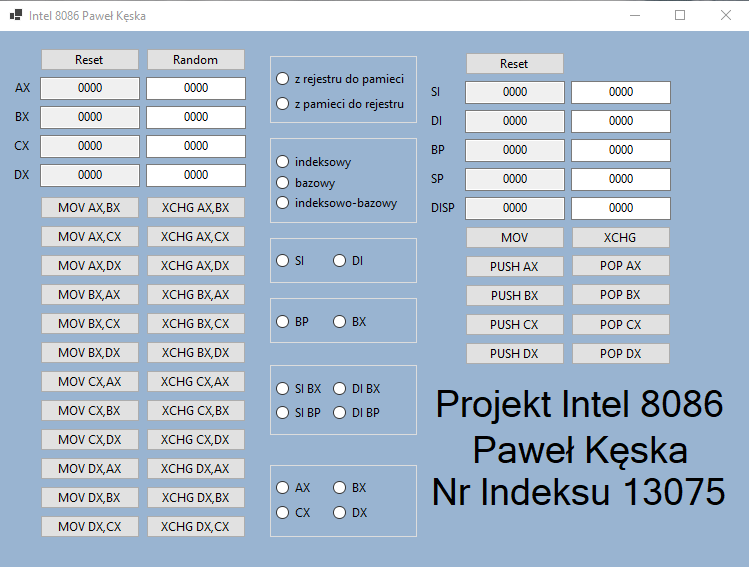
* Rejestry arytmetyczne – 16 bitowe rejestry ogólnego przeznaczenia AX,BX,CX,DC
  + AX – Akumulator
  + BX – Baza
  + CX – Licznik
  + DX – Dane
* Rejestry wskaźnikowe i indeksowe – 16 bitowe rejestry których głównym zadaniem jest wskazanie miejsca w pamięci
  + SI – Rejestr indeksowy źródła
  + BP – Wskaźnik bazy
  + DI – Rejestr indeksowy przeznaczenia
  + SP – Wskaźnik stosu
* Segment stosu
* Segment danych

4.Instrukcja obsługi

Aby uruchomić program należy dwukrotnie kliknąć na program “Intel 8086 Paweł Kęska.exe” ścieżka :

Intel 8086 Paweł Kęska\bin\Debug\net5.0-windows

Widok po uruchomieniu

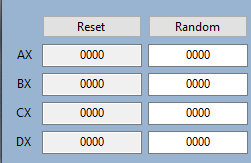


A)Rejestry

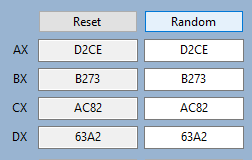
Rejestry AX,BX,CX,DX posiadają dwa pola. Pole lewe które jest tylko do odczytu i pokazuje wartość rejestru oraz prawe pole które służy **tylko** do wpisywania wartości.

Rejestry AX,BX,CX,DX można uzupełnić na dwa sposoby :

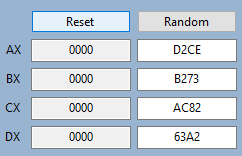
1. Wpisanie wartości w prawych okienkach oraz zatwierdzenie enterem. Okienko przyjmuje tylko liczby w systemie HEX czyli cyfry z zakresu 0-9 oraz litery od A do F.



2.Przycisk Random – który losuje cztery liczby w systemie HEX



Rejestry można również wyzerować za pomocą przycisku Reset.



Rejestry SI,DI,BP.SP,DISP posiadają dwa pola. Pole lewe które jest tylko do odczytu i pokazuje wartość rejestru oraz prawe pole które służy **tylko** do wpisywania wartości.

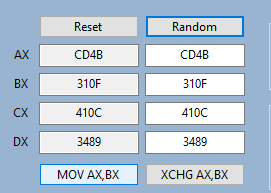
Rejestry SI,DI,BP.SP,DISP można uzupełnić tylko jednym sposobem:

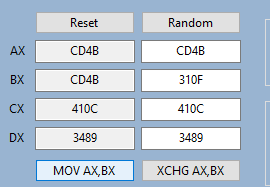
Wpisanie wartości w prawych okienkach oraz zatwierdzenie enterem. Okienko przyjmuje tylko liczby w systemie HEX czyli cyfry z zakresu 0-9 oraz litery od A do F.

B)Rozkazy MOV oraz XCHG dla rejestrów

Emulator pozwala na symulacje rozkazów MOV oraz XCHG między rejestrami. Rozkaz MOV pozwala na przeniesienie wartości z jednego rejestru do drugiego, zostawiając pierwszy rejestr z tą samą wartością. Rozkaz XCHG pozwala na zamianę wartości w dwóch rejestrach. Wartość pierwszego rejestru przechodzi do drugiego, a wartość drugiego rejestru przechodzi do pierwszego.

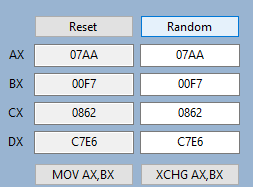
Aby zasymulować rozkazy MOV należy uzupełnić rejestry które chcemy wykorzystać ( np. przycisk Random). Następnie należy wybrać z opcji poniżej odpowiedni rozkaz.





**Przed :** **Po :**

Analogicznie działa opcja XCHG

****

**Przed :** **Po :**

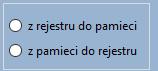
C)Rozkazy MOV oraz XCHG między rejestrami a pamięcią

Emulator pozwala również na symulacje rozkazów MOV oraz XCHG między rejestrami a pamięcią.

Wszystkie wartości z czterech rejestrów ogólnego przeznaczenia czyli AX,BX,CX,DX możemy zapisać w pamięci adresując za pomocą trzech trybów : Tryb indeksowy, Tryb Bazowy oraz Tryb Indeksowo-Bazowy.

Pamięć ma maksymalną wartość 64x1024 co daje 65536 komórek. W przypadku próby rozkazu MOV na adresie większym niż maksymalna wartość pamięci, wartość zapisze się w dwóch ostatnich komórkach pamięci.

Aby zasymulować rozkaz MOV należy najpierw wybrać czy chcemy przenieść wartość z rejestru do pamięci czy z pamięci do rejestru.

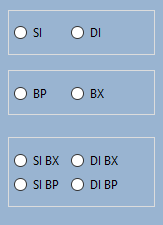


Następnie wybieramy tryb adresacji. W zależności od tego który tryb wybierzemy, program będzie liczył miejsce pamięci z rejestrów. Po wybraniu trybu adresacji ( Adressing Modes) musimy wybrać rejestry które chcemy użyć do wyznaczenia miejsca w pamięci.

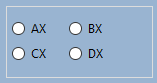
indeksowy– tryb indeksowy (Możliwe wybory SI oraz DI)

bazowy – tryb bazowy (Możliwe wybory BP oraz BX)

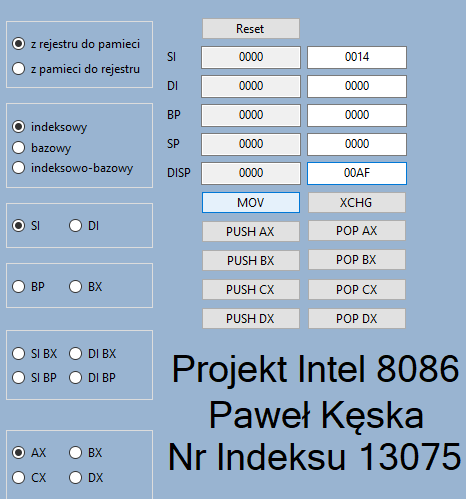
indeksowo – bazowy – Tryb bazowo indeksowy (Możliwe wybory DI BP,DI BX, SI BP oraz SI BX)



Nastęnie należy wybrać rejestr którego chcemy użyć:



Przykład rozkazu MOV rejestru AX do pamięci o adresie indeksowym SI 0014 oraz DISP 00AF. Po wciśnięciu przycisku MOV (Buttons for register-memory operations) wartość rejestru AX zostanie przeniesiona do komórki pamięci.



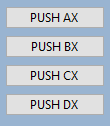
Sprawdzenie – aby sprawdzić czy program dobrze działa można zrobić operacje odwrotną tj. rozkaz MOV z pamięci do rejestru. W tym celu zerujemy rejestr AX oraz zaznaczamy przycisk From memory to register.

Rozkazy XCHG dla rejestrów oraz pamięci działają analogicznie. Jedyna różnica jest taka że nie potrzebny jest wybór przycisku Operation Mode, zaznaczenie go nie wpływa na wynik rozkazu. W przypadku rozkazów na komórkach pamięci które nie zostały jeszcze zapisane wartości pamięci będą miały wartość „0000”, a w przypadku próby rozkazu na adresie większym niż liczba miejsc w pamięci rozkaz wykona się na ostatnich dwóch komórkach pamięci.

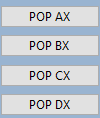
D)Stos oraz rozkazy PUSH i POP

Symulator umożliwia operacje na stosie.

Rozkaz PUSH umożliwia przeniesienie wartości wybranego rejestru do stosu. Po wykonaniu takiego rozkazu wskaźnik stosu ( SP ) rośnie o 2.



Rozkaz Pop umożliwia przeniesienie wartości z stosu do danego rejestru, w związku z tym iż jest to stos wartość którą można przenieść z stosu jest ostatnią wartością zapisaną w nim. Po wykonaniu takiego rozkazu wskaźnik stosu (SP) maleje o 2.

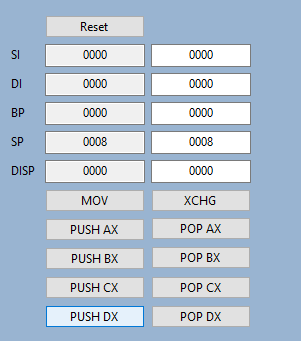


Wskaźnik stosu pokazuje obecną wartość w której jest najwyższa zapisana komórka w stosie. Wartośc tą można zmienić poprzez wpisanie jakiejkolwiek w systemie HEX.

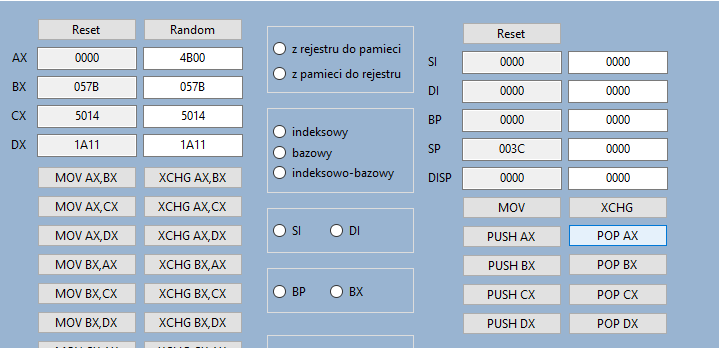


Przykład – Użycie czterech rozkazów Push AX, PUSH BX, PUSH CX oraz PUSH DX, wyzerowanie rejestrów a nastepnie użycie rozkazów POP w odwrotnej kolejności.

Rozkazy PUSH



Rozkazy POP w odwrotnej kolejności aby rejestry były ułożone tak jak początkowo.



W przypadku próby użycia rozkazu POP na pustym stosie pokaże się komunikat.

