
Адресиране, свързване на логически адрес с физически, разпределение на адресното пространство

1

Адресното пространство

Всички устройства, които са включени в микропроцесорната система, се представят като набор от клетки, разположени на отделни поредни адреси. Максималният възможен брой адреси в една микропроцесорна система се нарича **адресно пространство**.

2

Адресното пространство

Адресите физически са набор от нисък или висок потенциал (логически състояния 0 и 1) на проводниците от адресната магистрала и могат да се представят като число в двоична форма. Всеки разряд (бит) от числото отразява логическото състояние на един проводник.

3

Адресното пространство

При 20-проводна адресна магистрала най-малкият адрес е 00000000000000000000В, а най-големият – 1111111111111111111В (1 048 576 адреса), където "В" означава, че числото е представено в двоична бройна система. За удобство по-често се ползва шестнадесетичната бройна система.

4

Адресното пространство

Премаването от двоично представено число в шестнадесетично е лесно и става, като се заменят четири разряда (бита) с една цифра от 0 до F от шестнадесетичната бройна система. При това адресите в двадесетпроводна адресна магистрала се представят като числа в диапазона от 00000H до FFFFFH, където "H" означава, че числото е представено в шестнадесетична бройна система.

5

Адресното пространство

В микропроцесорните системи с микропроцесор 8086 общото адресно пространство е съставено от две подпространства – на адресите на клетките от паметта и на адресите от клетките на интерфейса. Сигнал M/IO определя дали адресът, който е изведен на адресната магистрала, се отнася за клетка от паметта или от интерфейса.

6

Адресното пространство

Паметта се адресира при $M/IO=1$ чрез 20-битов адрес и физическия брой на клетките е $2^{20} = 1\,048\,576$. Интерфейсът се адресира при $M/IO=0$. При това се използва 16-битовото съдържание на регистър DX или 16-битов код, включен в кода на инструкцията. Физическият брой на клетките на интерфейса е $2^{16} = 65\,536$ адреса.

7

Адресното пространство

С всяка клетка от адресното пространство може да се извърши операция за четене на нейното съдържание и операция за запис на нови данни. Физически в оперативната памет RAM това е една и съща клетка – в нея се записва някакво число, което след това, от същия адрес, се протича. От клетките на постоянната памет ROM може само да се чете.

8

Адресното пространство

Извършването на операция за запис на адрес, на който се намира клетка от паметта ROM не води до никакви промени (в някои системи опитът за запис в ROM генерира сигнал за грешка). При интерфейса в много случаи на един адрес има физически две клетки.

9

Адресното пространство

Пример: в интерфейса схемата има два регистра- в единия микропроцесорът записва байт, който определя режима на работа на интерфейсната схема (регистър за режим), а от втория микропроцесорът чете данни за състоянието на интерфейсната схема (регистър за състоянието). Двата регистра обикновено се намират на един и същи адрес.

10

Адресното пространство

При запис сигнал $WR=0$ определя, че извършваната операция ще е с регистъра за режим, а при четене $RD=0$ определя, че извършваната операция ще е с регистъра за състояние. Примерът показва, че физическият брой на клетките може да бъде по-голям от броя на адресите, защото при избора на всяка клетка, освен проводниците от адресната магистрала, участват и сигналите M/IO , RD и WR .

11

Адресното пространство

В реалните микропроцесорни системи адресното пространство на паметта често се занимава напълно, докато клетките от интерфейса заемат малка част от определеното им адресно пространство.

12

Адресиране

Адресиране се наричат действията на микропроцесора за определяне на адресите на клетките от паметта, в които са записани инструкциите и данните. Различните микропроцесори използват различни начини и схеми решения за адресиране на клетките от паметта.

13

Адресиране

Сегментните регистри CS, SS, DS и ES определят сегмента (частта) от паметта, в която е разположена необходимата клетка. Началният адрес на сегмента е равен на съдържанието на сегментния регистър, изместено с 4 бита наляво.

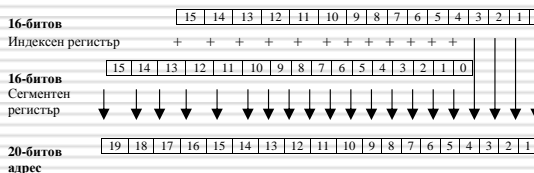
14

Адресиране

Вътрешната архитектура на микропроцесор 8086 е 16-битова и регистрите също са 16-битови. За да може микропроцесорът да работи с 20 – битов адрес, разработчикът е разделил блока с адресни регистри на две групи – сегментни и индексни.

15

Изчисляване на 20-битовия адрес в микропроцесор 8086



16

Адресиране

Индексните регистри IP, SP, BP, SI и DI определят конкретния адрес на клетката в този сегмент. Физическият адрес на клетката се изчислява като се събере съдържанието на сегментния регистър (изместено с 4 бита) със съдържанието на индексния регистър.

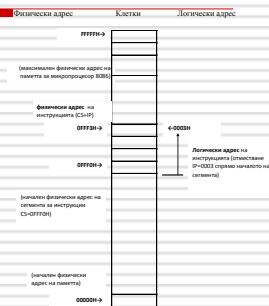
17

Адресиране

Ако CS=FFFFH (0000111111111111 в двоичен вид) и IP=003H (0000000000000011), физическият адрес на клетката, адресирана от микропроцесора ще бъде 0FFF3H (00001111111111110011). Това е онагледено на фиг. 3.2.

18

Физически и ефективен адрес на сегментираната памет



19

Адресиране

Използването на сегментни регистри по същество разделя адресното пространство на паметта на части. В сегментния регистър е записан началния адрес на сегмента, а индексният регистър съдържа отместването от началото на сегмента, наречено **логически адрес**. Използва се и термина **ефективен адрес**.

20

Адресиране

Наличието на сегментни регистри дава следните **преимущества**:

- Размерът на паметта може да достигне 1М байт (1 048 575 байта), въпреки че в изчисленията участва съдържанието на 16-битови регистри;

21

Адресиране

- Инструкциите на програмата и данните могат да се разполагат в различни области от паметта;
- Една програмата може да се зарежда в различни области на паметта, което е важно при многопроцесорни системи.

22

Адресиране

Всеки от сегментните и индексните регистри има специфично предназначение.

23

Адресиране

Когато се извлича код на инструкция, тогава за адресиране се използва съдържанието на двойката регистри CS и IP. Регистри SS и SP се използват при работа с клетки от стека. Останалите регистри в различни комбинации се използват при адресиране на клетки, използвани за съхранение на операнди и резултати.

24

Адресиране

Съответно адресното пространство се разделя на следните сегменти: за инструкции, за стека и за данни. Няма ограничение за числото, което ще бъде записано във всеки от адресните регистри. Следователно разпределението на адресното пространство може да бъде направено по най-удобния за програмиста начин. Сегментите могат да се припокриват частично или изцяло.

25

Адресиране

Адресът на операндите може да бъде изчислен по няколко начина, които се наричат **режими на адресация**.

За микропроцесор 8086 режимите за адресация се разделят на два класа – режими за адресация на данните и режими за адресация на преходите.

26

Адресиране

За изясняване на режимите на адресация на данни, ще разгледаме по колко начина може да бъде адресиран 16-битовия операнд A (опА) в инструкцията "запиши опА в акумулатора AX". След като микропроцесорът извлече кода на тази инструкция, управляващият блок определя в зависимост от режима на адресация от къде (от какъв адрес) трябва да извлече опА и да го запише в регистър AL.

27

Адресиране

При определяне на адресите на клетки с данни по подразбиране се използва сегментния регистър DS.

28

Адресиране

Режимите на адресация са:

- **Непосредствен.** Кодът на инструкцията съдържа самия операнд.
Команда, опА.
- **Пряк.** Кодът на инструкцията съдържа адреса на клетката, в която се съдържа опА. (в сегмента за данни).
Команда, 16-битов адрес на опА.

29

Адресиране

- **Регистърен.** В кода на инструкцията е посочен регистъра, в който е записан опА. Може да се използва всеки от регистрите AX, BX, CX, DX, SI, DI, SP и BP.

Команда, регистър.

30

Адресиране

- **Регистърен косвен.** В кода на инструкцията е посочен регистър, в който е записан адреса на клетката, съдържаща опА.

Команда, (регистър).

Скобите означават, че в регистъра е записан ефективния адрес на операнда, а не самият операнд. При този режим могат да се ползват регистри BX, SI и DI.

31

Адресиране

- **Регистърен относителен.** В кода на инструкцията е посочен регистър, в който е записан адреса и 8 или 16-битово отместване, което трябва да се добави към съдържанието на регистъра за да се получи адреса на клетката, съдържаща опА.

Команда, (регистър), отместване.

При този режим могат да се ползват регистри BX, BP, SI и DI.

32

Адресиране

- **Базов индексен.** В кода на инструкцията е посочен базов регистър и индексен регистър. Съдържанието на двата регистъра трябва да се сумира за да се получи адреса на клетката, съдържаща опА.

Команда, (базов регистър), (индексен регистър).

При този режим могат да се ползват – за базов регистър BX и BP, а за индексен - SI и DI.

33

Адресиране

- **Относителен базов индексен.** Режимът е като предходния. Съдържанието на двата регистъра се сумира и се прибавя 8 или 16-битово отместване за да се получи адреса на клетката, съдържаща опА.

Команда, (базов регистър), (индексен регистър), отместване.

При този режим могат да се използват – за базов регистър BX, BP, а за индексен - SI и DI.

34

Адресиране

Инструкциите в програмата са записани последователно в клетките на паметта и адреса на следващата инструкция се определя като адреса на предходната инструкция се увеличи на дадена инструкция да се извърши преход към инструкция, която се намира на някакъв несъседен адрес.

35

Адресиране

Режимите за адресиране на прехода са следните:

- **Вътрешносегментен пряк.** Ефективния адрес се получава, като към подразбиращият се регистър IP се добави 8 или 16-битово отместване.

Команда, отместване.

36

Адресиране

- **Вътрешносегментен косвен.** Ефективният адрес се получава, като към подразбиращия се регистър IP се добави съдържанието на регистър или на клетка от паметта.
Команда, (регистър или клетка).

37

Адресиране

- **Междусегментен пряк.** Адресът на прехода се получава, като съдържанието на сегментния регистър CS и на програмния брояч IP се заменят съответно с втората и с третата част от кода на инструкцията.

Команда, адрес на сегмент, ефективен адрес.

38

Адресиране

- **Междусегментен косвен.** Адресът на прехода се получава, като съдържанието на сегментния регистър CS и на програмния брояч IP се заменят съответно със съдържанието на регистъра или клетката, адресите на които са посочени във втората и третата част от кода на инструкцията.
Команда, (регистър или клетка), (регистър или клетка).

39

Адресиране

Голямото разнообразие от режими на адресация е важно предимство на микропроцесора 8086, защото адресирането и извличането на операндите може да става много гъвкаво, без загуба на време от излишни прехвърляния и подготовка на операнди.

40

Край на част 2

41