Department of Computer Engineering Techniques (Stage: 4)

Advance Computer Technologies

Dr.: Mayas Aljibawi

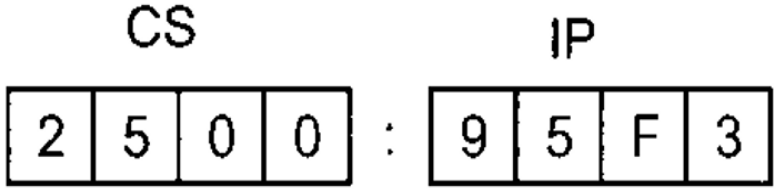
INTRODUCTION TO PROGRAM SEGMENTS

A typical Assembly language program consists of at least three segments: a code segment, a data segment, and a stack segment.

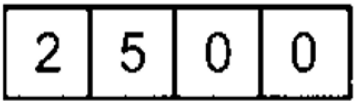
Logical address and physical address

In Intel literature concerning the 8086, there are three types of addresses mentioned frequently: the physical address, the offset address, and the logical address. The *physical address* is the 20-bit address that is actually put on the address pins of the 8086 microprocessor and decoded by the memory interfacing circuitry. This address can have a range of 00000H to FFFFFH for the 8086 and real-mode 286, 386, and 486 CPUs. This is an actual physical location in RAM or ROM within the 1 megabyte memory range. The *offset address* is a location within a 64K-byte segment range. Therefore, an offset address can range from 0000H to FFFFH. The *logical address* consists of a segment value and an offset address. The differences among these addresses and the process of converting from one to another is best understood in the context of some examples, as shown next.

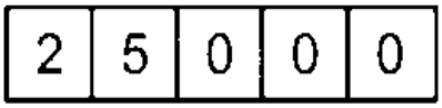
Code segment



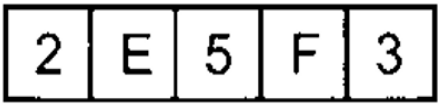
1. Start with CS.



2. Shift left CS.



3. Add IP.



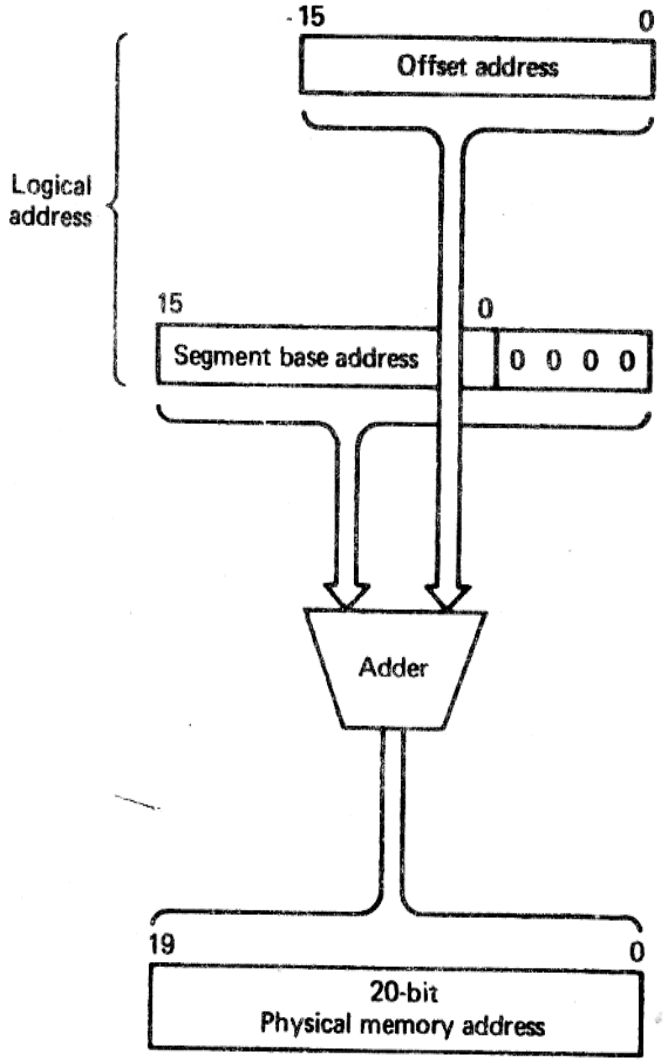


Figure 2.17 Real-mode physical address generation. (Reprinted by permission of Intel Corp. Copyright/Intel 1981)

Example 1-1

If CS = 24F6H and IP = 634AH, show:

(a) The logical address

(b) The offset address

and calculate:

(c) The physical address

(d) The lower range

(e) The upper range of the code segment

Solution:

(a) 24F6:634A

(b) 634A

(c) 2B2AA (24F60 + 634A)

(d) 24F60 (24F60 + 0000)

(e) 34F5F (24F60 + FFFF)

Logical address vs. physical address in the code segment

Logical address CS:IP	Machine language opcode and operand	Assembly language mnemonics and operand
1132:0100	B057	MOV AL,57
1132:0102	B686	MOV DH,86
1132:0104	B272	MOV DL,72
1132:0106	89D1	MOV CX,DX
1132:0108	88C7	MOV BH,AL
1132:010A	B39F	MOV BL,9F
1132:010C	B420	MOV AH,20
1132:010E	01D0	ADD AX,DX
1132:0110	01D9	ADD CX,BX
1132:0112	05351F	ADD AX,1F35

Logical address	Physical address	Machine code contents
1132:0100	11420	B0
1132:0101	11421	57
1132:0102	11422	B6
1132:0103	11423	86
1132:0104	11424	B2
1132:0105	11425	72
1132:0106	11426	89
1132:0107	11427	D1
1132:0108	11428	88
1132:0109	11429	C7
1132:010A	1142A	B3
1132:010B	1142B	9F
1132:010C	1142C	B4
1132:010D	1142D	20
1132:010E	1142E	01
1132:010F	1142F	D0
1132:0110	11430	01
1132:0111	11431	D9
1132:0112	11432	05
1132:0113	11433	35
1132:0114	11434	1F

Data segment

Assume that a program is being written to add 5 bytes of data, such as 25H, 12H, 15H, 1FH, and 2BH, where each byte represents a person's daily overtime pay. One way to add them is as follows:

```
MOV    AL,00H        ;initialize AL
ADD    AL,25H        ;add 25H to AL
ADD    AL,12H        ;add 12H to AL
ADD    AL,15H        ;add 15H to AL
ADD    AL,1FH        ;add 1FH to AL
ADD    AL,2BH        ;add 2BH to AL
```

DS:0200 = 25
DS:0201 = 12
DS:0202 = 15
DS:0203 = 1F
DS:0204 = 2B

and the program can be rewritten as follows:

```
MOV    AL,0            ;clear AL
ADD    AL,[0200]       ;add the contents of DS:200 to AL
ADD    AL,[0201]       ;add the contents of DS:201 to AL
ADD    AL,[0202]       ;add the contents of DS:202 to AL
ADD    AL,[0203]       ;add the contents of DS:203 to AL
ADD    AL,[0204]       ;add the contents of DS:204 to AL
```

This program will run with any set of data. Changing the data has no effect on the code.

The 8086/88 allows only the use of registers BX, SI, and DI as offset registers for the data segment. In other words, while CS uses only the IP register as an offset, DS uses only BX, DI, and SI to hold the offset address of the data. The term *pointer* is often used for a register holding an offset address. In the following example, BX is used as a pointer:

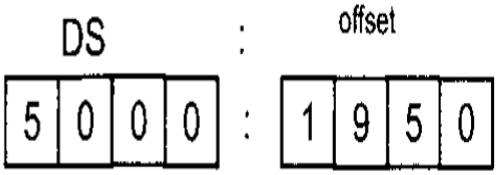
```
MOV   AL,0          ;initialize AL
MOV   BX,0200H      ;BX points to the offset addr of first byte
ADD   AL,[BX]       ;add the first byte to AL
INC   BX            ;increment BX to point to the next byte
ADD   AL,[BX]       ;add the next byte to AL
INC   BX            ;increment the pointer
ADD   AL,[BX]       ;add the next byte to AL
INC   BX            ;increment the pointer
ADD   AL,[BX]       ;add the last byte to AL
```

Logical address and physical address in the data segment

Example 1-2

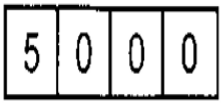
Assume that DS is 5000 and the offset is 1950. Calculate the physical address of the byte.

Solution:

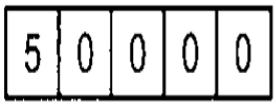


The physical address will be 50000 + 1950 = 51950.

1. Start with DS.



2. Shift DS left.



3. Add the offset.

Example 1-3

If DS = 7FA2H and the offset is 438EH,

(a) Calculate the physical address.
(b) Calculate the lower range.
(c) Calculate the upper range of the data segment.
(d) Show the logical address.

Solution:

(a) 83DAE (7FA20 + 438E)
(b) 7FA20 (7FA20 + 0000)
(c) 8FA1F (7FA20 + FFFF)
(d) 7FA2:438E

Example 1-4

Assume that the DS register is 578C. To access a given byte of data at physical memory location 67F66, does the data segment cover the range where the data is located? If not, what changes need to be made?

Solution:

No, since the range is 578C0 to 678BF, location 67F66 is not included in this range. To access that byte, DS must be changed so that its range will include that byte.

Little endian convention

Previous examples used 8-bit or 1-byte data. In this case the bytes are stored one after another in memory. What happens when 16-bit data is used? For example:

```
MOV   AX,35F3H      ;load 35F3H into AX
MOV   [1500],AX     ;copy the contents of AX to offset 1500H
```

In cases like this, the low byte goes to the low memory location and the high byte goes to the high memory address. In the example above, memory location DS:1500 contains F3H and memory location DS:1501 contains 35H.

DS:1500 = F3 DS:1501 = 35

Example 1-5

Assume memory locations with the following contents: DS:6826 = 48 and DS:6827 = 22. Show the contents of register BX in the instruction "MOV BX,[6826]".

Solution:

According to the little endian convention used in all 80x86 microprocessors, register BL should contain the value from the low offset address 6826 and register BH the value from offset address 6827, giving BL = 48H and BH = 22H.

DS:6826 = 48
DS:6827 = 22

BH	BL
22	48

Review Questions

1. A segment is an area of memory that includes up to ___ bytes.
2. How large is a segment in the 8086? Can the physical address 346E0 be the starting address for a segment? Why or why not?
3. State the difference between the physical and logical addresses.
4. A physical address is a ___-bit address; an offset address is a ___-bit address.
5. Which register is used as the offset register with segment register CS?
6. If BX = 1234H and the instruction "MOV [2400],BX" were executed, what would be the contents of memory locations at offsets 2400 and 2401?