第二章
指令定地址和執行

作業系統的功能
BIOS載入過程
程式載入
堆疊
指令執行和定地址
指令運算元

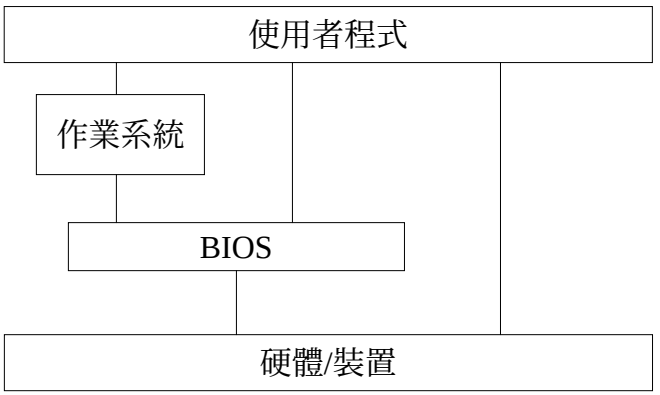
作業系統的功能

- 檔案管理
- 輸入/輸出
- 程式載入
- 記憶體管理
- 插斷處理

BIOS載入過程

- 8086重置(Reset)後, CS的值是FFFFH, IP是0。程式執行由FFFF0H處開始(BIOS的開始位置)
- BIOS(基本輸入/輸出系統)存放在ROM中
- 首先檢查裝置, 接著建立兩個資料區
  - 插斷向量表: 0到3FFH (256個插斷, 每個4位元組)
  - BIOS資料區: 由400H起
- 載入作業系統(Bootload)

輸入/輸出界面



可執行程式

- .exe檔: 分開的程式碼資料和堆疊段
- .com檔:程式碼資料和堆疊合併在一64K段中

.exe 程式的載入

- 由磁碟中取出檔案
- 在可用的記憶體建立一256位元組的PSP
- 程式碼放在PSP之後
- 將PSP的段號放到DS和ES
- 程式碼的段號放在CS, IP放0
- 堆疊段號放到SS, 堆疊大小(位元組)放到SP
- 開始執行

堆疊

- 作用
  - 呼叫副程式時貯存回轉的地址
  - 呼叫副程式時傳遞參數
  - 暫時存放暫存器值, 空出暫存器作其它計算
- 操作指令
  - PUSH: SP減2, 存入指定值(暫存器或記憶體)
  - POP: 取回值, SP加2
  - PUSHF, POPF
  - PUSHA, POPA
  - PUSHAD, POPAD

例子:

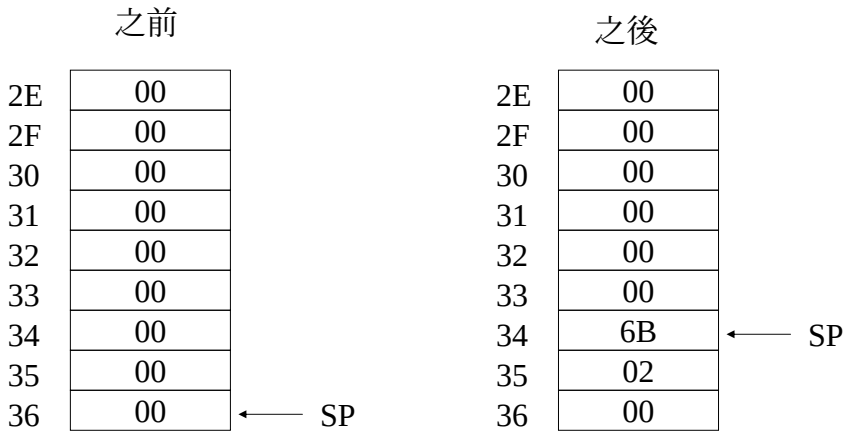
暫存器值

AX: 026BH, BX: 04E3H, SP:36H

執行指令

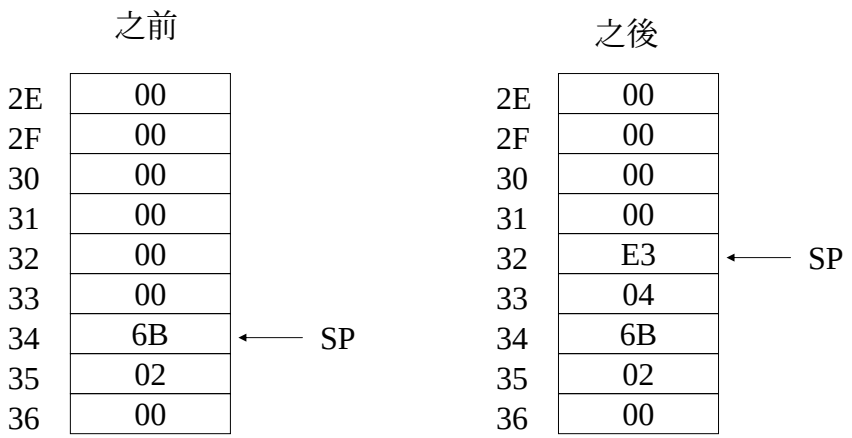
```
PUSH AX
PUSH BX
POP BX
POP AX
```

PUSH AX

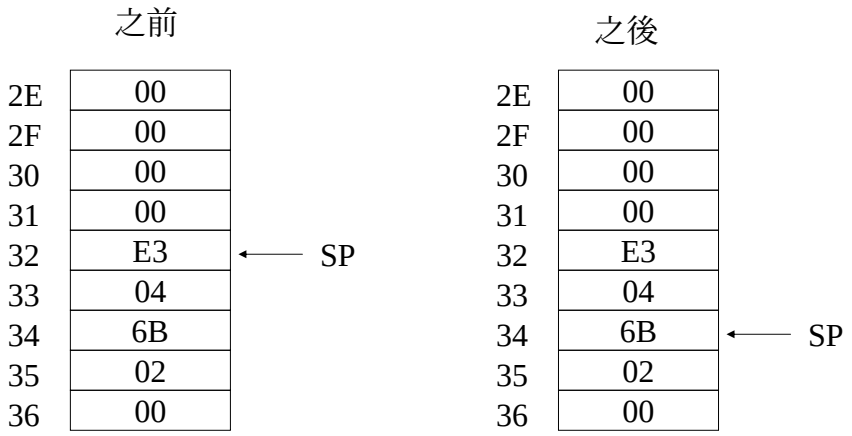


暫存器推入後, 暫存器原值仍存在

PUSH BX

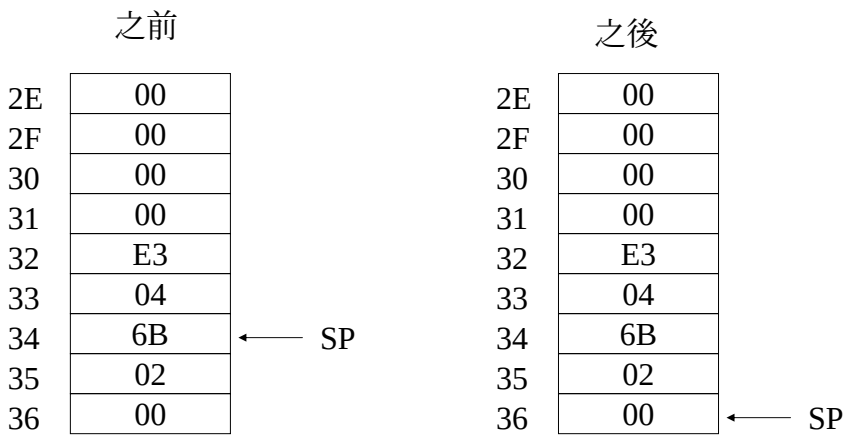


POP BX



記憶體資料讀出後, 原資料仍存在

POP AX



指令執行步驟

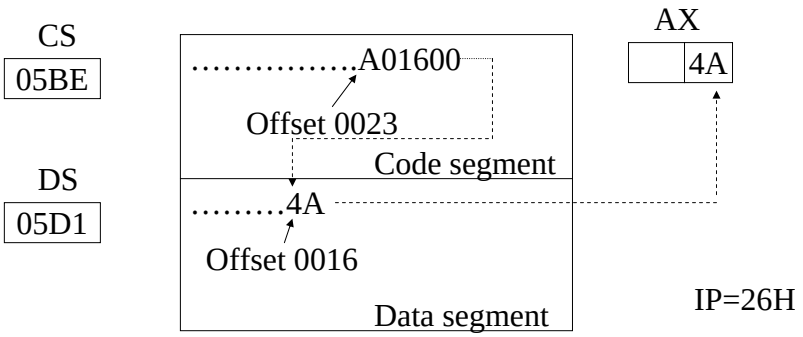
- 取指令
- 解碼
- 執行

管線執行

定地址

CS=05BEH, IP=23H

5C03H A01600 MOV AL,[0016]



MOV [0016],AX

AX=0248H

記憶體內容	48	02
Offset	0016	0017

指令運算元

```
WORDX   DW   0
        …
        MOV CX,WORDX        ;Direct (直接)
        MOV CX,25           ;immediate (立即)
        MOV CX,DX           ;register (暫存器)
        MOV CX,[DX]         ;indirect (間接)
```