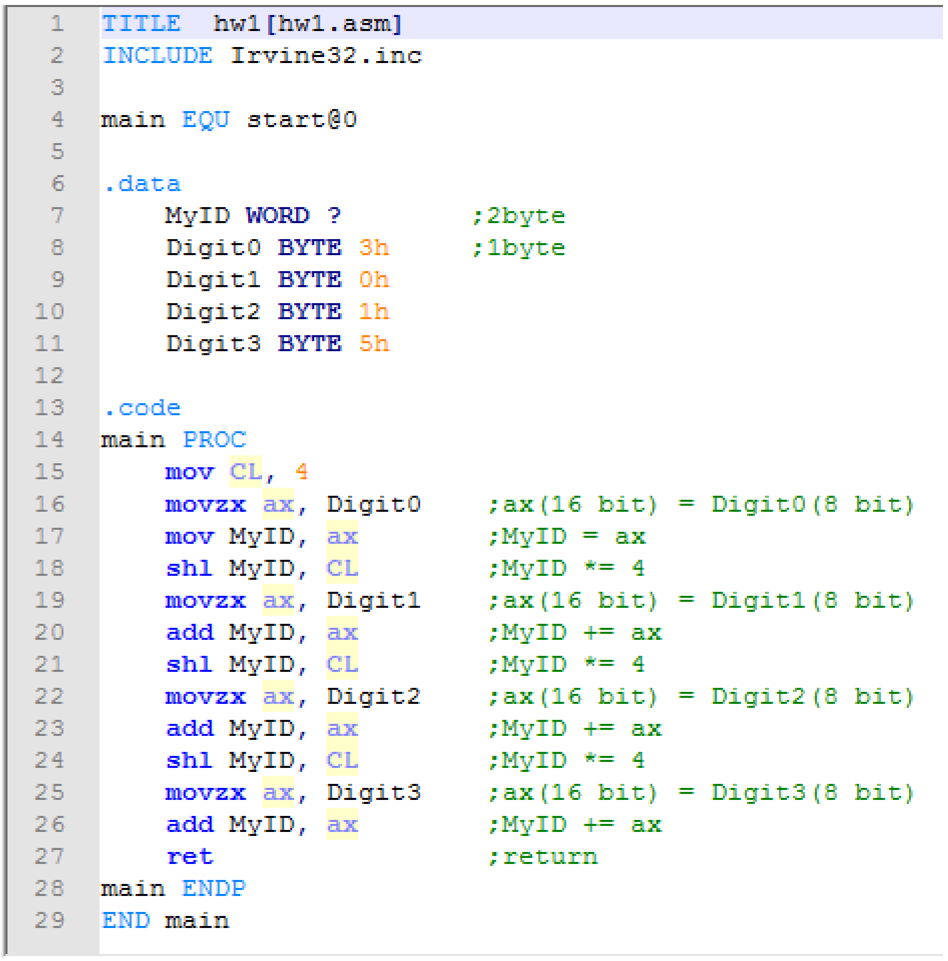
HW#1: ARITHMETIC

MIS 3A、102403015、程祥恩

CODE



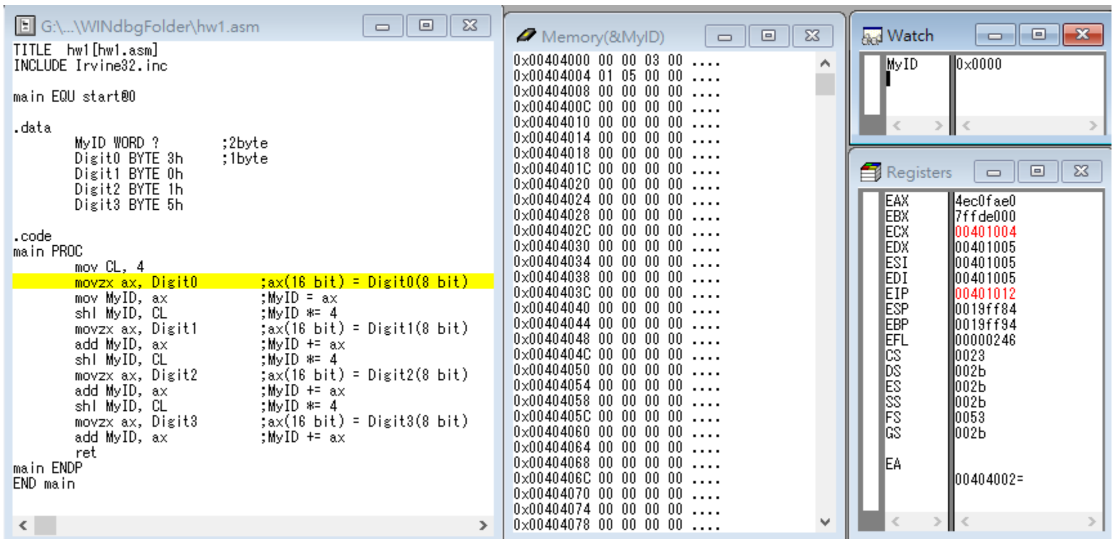
```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
    MyID WORD ?          ;2byte
    Digit0 BYTE 3h       ;1byte
    Digit1 BYTE 0h
    Digit2 BYTE 1h
    Digit3 BYTE 5h

.code
main PROC
    mov CL, 4
    movzx ax, Digit0     ;ax(16 bit) = Digit0(8 bit)
    mov MyID, ax         ;MyID = ax
    shl MyID, CL         ;MyID *= 4
    movzx ax, Digit1     ;ax(16 bit) = Digit1(8 bit)
    add MyID, ax         ;MyID += ax
    shl MyID, CL         ;MyID *= 4
    movzx ax, Digit2     ;ax(16 bit) = Digit2(8 bit)
    add MyID, ax         ;MyID += ax
    shl MyID, CL         ;MyID *= 4
    movzx ax, Digit3     ;ax(16 bit) = Digit3(8 bit)
    add MyID, ax         ;MyID += ax
    ret                  ;return
main ENDP
END main
```

Step 1: mov 4 to CL, which is used to shl MyID



```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
        MyID WORD ?          ;2byte
        Digit0 BYTE 3h       ;1byte
        Digit1 BYTE 0h
        Digit2 BYTE 1h
        Digit3 BYTE 5h

.code
main PROC
        mov CL, 4
        movzx ax, Digit0          ;ax(16 bit) = Digit0(8 bit)
        mov MyID, ax              ;MyID = ax
        shl MyID, CL              ;MyID *= 4
        movzx ax, Digit1          ;ax(16 bit) = Digit1(8 bit)
        add MyID, ax              ;MyID += ax
        shl MyID, CL              ;MyID *= 4
        movzx ax, Digit2          ;ax(16 bit) = Digit2(8 bit)
        add MyID, ax              ;MyID += ax
        shl MyID, CL              ;MyID *= 4
        movzx ax, Digit3          ;ax(16 bit) = Digit3(8 bit)
        add MyID, ax              ;MyID += ax
        ret
main ENDP
END main
```

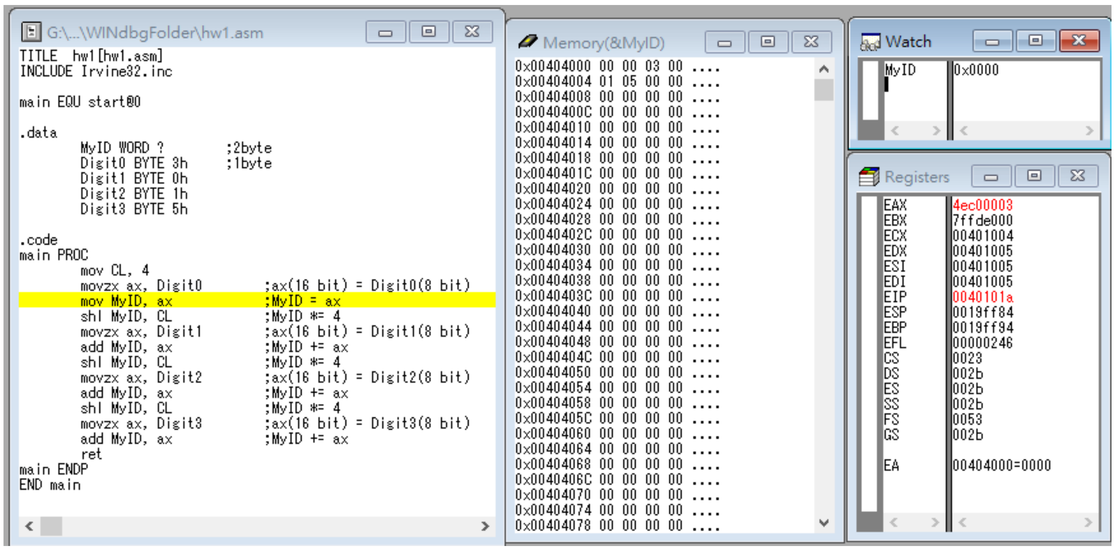
```
0x00404000 00 00 03 00 ....
0x00404004 01 05 00 00 ....
0x00404008 00 00 00 00 ....
0x0040400C 00 00 00 00 ....
0x00404010 00 00 00 00 ....
0x00404014 00 00 00 00 ....
0x00404018 00 00 00 00 ....
0x0040401C 00 00 00 00 ....
0x00404020 00 00 00 00 ....
0x00404024 00 00 00 00 ....
0x00404028 00 00 00 00 ....
0x0040402C 00 00 00 00 ....
0x00404030 00 00 00 00 ....
0x00404034 00 00 00 00 ....
0x00404038 00 00 00 00 ....
0x0040403C 00 00 00 00 ....
0x00404040 00 00 00 00 ....
0x00404044 00 00 00 00 ....
0x00404048 00 00 00 00 ....
0x0040404C 00 00 00 00 ....
0x00404050 00 00 00 00 ....
0x00404054 00 00 00 00 ....
0x00404058 00 00 00 00 ....
0x0040405C 00 00 00 00 ....
0x00404060 00 00 00 00 ....
0x00404064 00 00 00 00 ....
0x00404068 00 00 00 00 ....
0x0040406C 00 00 00 00 ....
0x00404070 00 00 00 00 ....
0x00404074 00 00 00 00 ....
0x00404078 00 00 00 00 ....
```

```
MyID    0x0000
```

```
EAX     4ec0fae0
EBX     7ffde000
ECX     00401004
EDX     00401005
ESI     00401005
EDI     00401005
EIP     00401012
ESP     0019ff84
EBP     0019ff94
EFL     00000246
CS      0023
DS      002b
ES      002b
SS      002b
FS      0053
GS      002b

EA
        00404002=
```

Step 2: movzx Digit0 to ax since ax is 16 bit, twice than Digit0(8bit). ax=0x0003



```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
        MyID WORD ?         ;2byte
        Digit0 BYTE 3h      ;1byte
        Digit1 BYTE 0h
        Digit2 BYTE 1h
        Digit3 BYTE 5h

.code
main PROC
        mov CL, 4
        movzx ax, Digit0         ;ax(16 bit) = Digit0(8 bit)
        mov MyID, ax             ;MyID = ax
        shl MyID, CL             ;MyID *= 4
        movzx ax, Digit1         ;ax(16 bit) = Digit1(8 bit)
        add MyID, ax             ;MyID += ax
        shl MyID, CL             ;MyID *= 4
        movzx ax, Digit2         ;ax(16 bit) = Digit2(8 bit)
        add MyID, ax             ;MyID += ax
        shl MyID, CL             ;MyID *= 4
        movzx ax, Digit3         ;ax(16 bit) = Digit3(8 bit)
        add MyID, ax             ;MyID += ax
        ret
main ENDP
END main
```

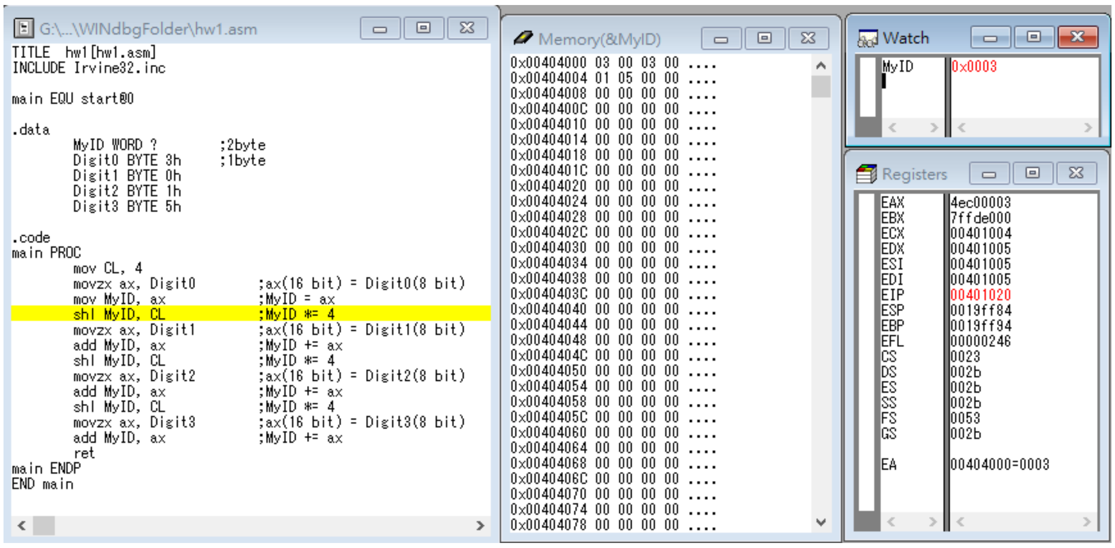
```
0x00404000 00 00 03 00 ....
0x00404004 01 05 00 00 ....
0x00404008 00 00 00 00 ....
0x0040400C 00 00 00 00 ....
0x00404010 00 00 00 00 ....
0x00404014 00 00 00 00 ....
0x00404018 00 00 00 00 ....
0x0040401C 00 00 00 00 ....
0x00404020 00 00 00 00 ....
0x00404024 00 00 00 00 ....
0x00404028 00 00 00 00 ....
0x0040402C 00 00 00 00 ....
0x00404030 00 00 00 00 ....
0x00404034 00 00 00 00 ....
0x00404038 00 00 00 00 ....
0x0040403C 00 00 00 00 ....
0x00404040 00 00 00 00 ....
0x00404044 00 00 00 00 ....
0x00404048 00 00 00 00 ....
0x0040404C 00 00 00 00 ....
0x00404050 00 00 00 00 ....
0x00404054 00 00 00 00 ....
0x00404058 00 00 00 00 ....
0x0040405C 00 00 00 00 ....
0x00404060 00 00 00 00 ....
0x00404064 00 00 00 00 ....
0x00404068 00 00 00 00 ....
0x0040406C 00 00 00 00 ....
0x00404070 00 00 00 00 ....
0x00404074 00 00 00 00 ....
0x00404078 00 00 00 00 ....
```

```
MyID    0x0000
```

```
EAX     4ec00003
EBX     7ffde000
ECX     00401004
EDX     00401005
ESI     00401005
EDI     00401005
EIP     0040101a
ESP     0019ff84
EBP     0019ff94
EFL     00000246
CS      0023
DS      002b
ES      002b
SS      002b
FS      0053
GS      002b

EA      00404000=0000
```

Step 3: mov ax to MyID. MyID equals to 0x0003



```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
        MyID WORD ?         ;2byte
        Digit0 BYTE 3h      ;1byte
        Digit1 BYTE 0h
        Digit2 BYTE 1h
        Digit3 BYTE 5h

.code
main PROC
        mov CL, 4
        movzx ax, Digit0         ;ax(16 bit) = Digit0(8 bit)
        mov MyID, ax             ;MyID = ax
        shl MyID, CL             ;MyID *= 4
        movzx ax, Digit1         ;ax(16 bit) = Digit1(8 bit)
        add MyID, ax             ;MyID += ax
        shl MyID, CL             ;MyID *= 4
        movzx ax, Digit2         ;ax(16 bit) = Digit2(8 bit)
        add MyID, ax             ;MyID += ax
        shl MyID, CL             ;MyID *= 4
        movzx ax, Digit3         ;ax(16 bit) = Digit3(8 bit)
        add MyID, ax             ;MyID += ax
        ret
main ENDP
END main
```

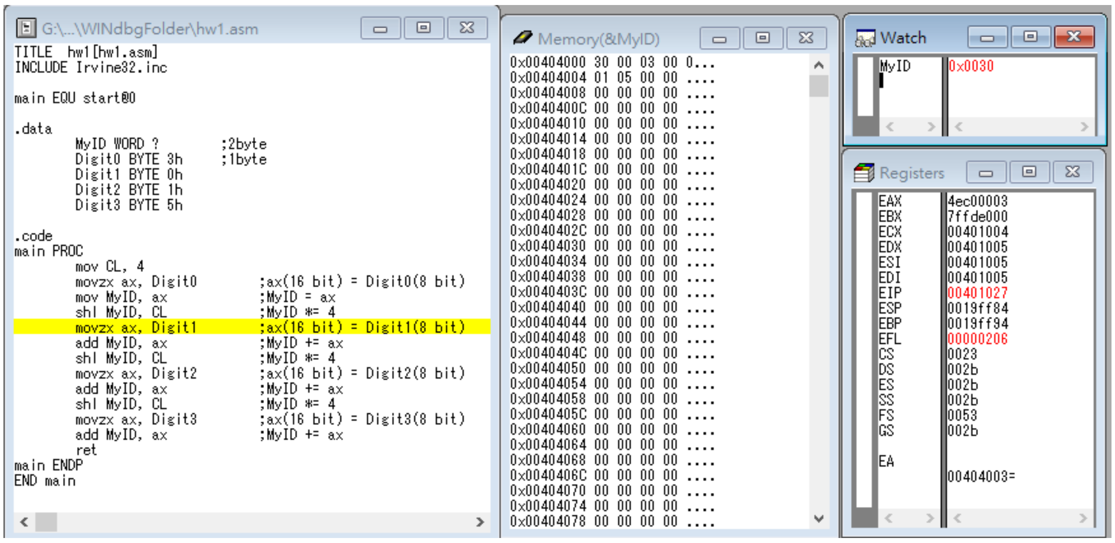
```
0x00404000 03 00 03 00 ....
0x00404004 01 05 00 00 ....
0x00404008 00 00 00 00 ....
0x0040400C 00 00 00 00 ....
0x00404010 00 00 00 00 ....
0x00404014 00 00 00 00 ....
0x00404018 00 00 00 00 ....
0x0040401C 00 00 00 00 ....
0x00404020 00 00 00 00 ....
0x00404024 00 00 00 00 ....
0x00404028 00 00 00 00 ....
0x0040402C 00 00 00 00 ....
0x00404030 00 00 00 00 ....
0x00404034 00 00 00 00 ....
0x00404038 00 00 00 00 ....
0x0040403C 00 00 00 00 ....
0x00404040 00 00 00 00 ....
0x00404044 00 00 00 00 ....
0x00404048 00 00 00 00 ....
0x0040404C 00 00 00 00 ....
0x00404050 00 00 00 00 ....
0x00404054 00 00 00 00 ....
0x00404058 00 00 00 00 ....
0x0040405C 00 00 00 00 ....
0x00404060 00 00 00 00 ....
0x00404064 00 00 00 00 ....
0x00404068 00 00 00 00 ....
0x0040406C 00 00 00 00 ....
0x00404070 00 00 00 00 ....
0x00404074 00 00 00 00 ....
0x00404078 00 00 00 00 ....
```

```
MyID    0x0003
```

```
EAX     4ec00003
EBX     7ffde000
ECX     00401004
EDX     00401005
ESI     00401005
EDI     00401005
EIP     00401020
ESP     0019ff84
EBP     0019ff94
EFL     00000246
CS      0023
DS      002b
ES      002b
SS      002b
FS      0053
GS      002b

EA      00404000=0003
```

Step 4: shl MyID for 4 bits, which makes MyID from 0x0003 to 0x0030



```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
        MyID WORD ?         ;2byte
        Digit0 BYTE 3h      ;1byte
        Digit1 BYTE 0h
        Digit2 BYTE 1h
        Digit3 BYTE 5h

.code
main PROC
        mov CL, 4
        movzx ax, Digit0         ;ax(16 bit) = Digit0(8 bit)
        mov MyID, ax             ;MyID = ax
        shl MyID, CL             ;MyID *= 4
        movzx ax, Digit1         ;ax(16 bit) = Digit1(8 bit)
        add MyID, ax             ;MyID += ax
        shl MyID, CL             ;MyID *= 4
        movzx ax, Digit2         ;ax(16 bit) = Digit2(8 bit)
        add MyID, ax             ;MyID += ax
        shl MyID, CL             ;MyID *= 4
        movzx ax, Digit3         ;ax(16 bit) = Digit3(8 bit)
        add MyID, ax             ;MyID += ax
        ret
main ENDP
END main
```

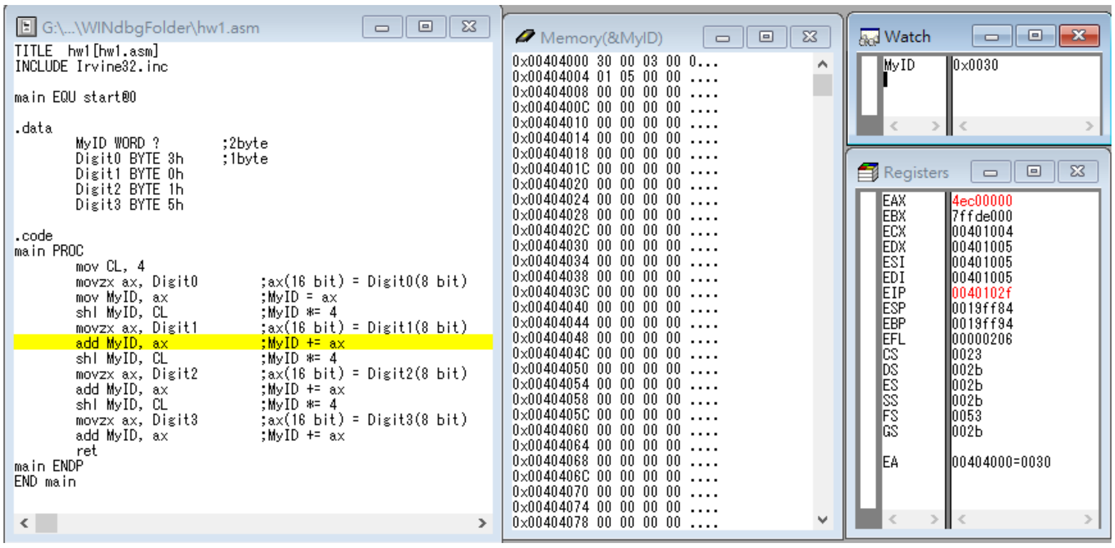
```
0x00404000 30 00 03 00 0...
0x00404004 01 05 00 00 ....
0x00404008 00 00 00 00 ....
0x0040400C 00 00 00 00 ....
0x00404010 00 00 00 00 ....
0x00404014 00 00 00 00 ....
0x00404018 00 00 00 00 ....
0x0040401C 00 00 00 00 ....
0x00404020 00 00 00 00 ....
0x00404024 00 00 00 00 ....
0x00404028 00 00 00 00 ....
0x0040402C 00 00 00 00 ....
0x00404030 00 00 00 00 ....
0x00404034 00 00 00 00 ....
0x00404038 00 00 00 00 ....
0x0040403C 00 00 00 00 ....
0x00404040 00 00 00 00 ....
0x00404044 00 00 00 00 ....
0x00404048 00 00 00 00 ....
0x0040404C 00 00 00 00 ....
0x00404050 00 00 00 00 ....
0x00404054 00 00 00 00 ....
0x00404058 00 00 00 00 ....
0x0040405C 00 00 00 00 ....
0x00404060 00 00 00 00 ....
0x00404064 00 00 00 00 ....
0x00404068 00 00 00 00 ....
0x0040406C 00 00 00 00 ....
0x00404070 00 00 00 00 ....
0x00404074 00 00 00 00 ....
0x00404078 00 00 00 00 ....
```

```
MyID    0x0030
```

```
EAX     4ec00003
EBX     7ffde000
ECX     00401004
EDX     00401005
ESI     00401005
EDI     00401005
EIP     00401027
ESP     0019ff84
EBP     0019ff94
EFL     00000206
CS      0023
DS      002b
ES      002b
SS      002b
FS      0053
GS      002b

EA      00404003=
```

Step 5: movzx Digit1 to ax. ax = 0x0000



G:\...\WINdbgFolder\hw1.asm

```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
        MyID WORD ?          ;2byte
        Digit0 BYTE 3h       ;1byte
        Digit1 BYTE 0h
        Digit2 BYTE 1h
        Digit3 BYTE 5h

.code
main PROC
        mov CL, 4
        movzx ax, Digit0          ;ax(16 bit) = Digit0(8 bit)
        mov MyID, ax              ;MyID = ax
        shl MyID, CL              ;MyID *= 4
        movzx ax, Digit1          ;ax(16 bit) = Digit1(8 bit)
        add MyID, ax              ;MyID += ax
        shl MyID, CL              ;MyID *= 4
        movzx ax, Digit2          ;ax(16 bit) = Digit2(8 bit)
        add MyID, ax              ;MyID += ax
        shl MyID, CL              ;MyID *= 4
        movzx ax, Digit3          ;ax(16 bit) = Digit3(8 bit)
        add MyID, ax              ;MyID += ax
        ret
main ENDP
END main
```

Memory(&MyID)

```
0x00404000 30 00 03 00 0...
0x00404004 01 05 00 00 ....
0x00404008 00 00 00 00 ....
0x0040400C 00 00 00 00 ....
0x00404010 00 00 00 00 ....
0x00404014 00 00 00 00 ....
0x00404018 00 00 00 00 ....
0x0040401C 00 00 00 00 ....
0x00404020 00 00 00 00 ....
0x00404024 00 00 00 00 ....
0x00404028 00 00 00 00 ....
0x0040402C 00 00 00 00 ....
0x00404030 00 00 00 00 ....
0x00404034 00 00 00 00 ....
0x00404038 00 00 00 00 ....
0x0040403C 00 00 00 00 ....
0x00404040 00 00 00 00 ....
0x00404044 00 00 00 00 ....
0x00404048 00 00 00 00 ....
0x0040404C 00 00 00 00 ....
0x00404050 00 00 00 00 ....
0x00404054 00 00 00 00 ....
0x00404058 00 00 00 00 ....
0x0040405C 00 00 00 00 ....
0x00404060 00 00 00 00 ....
0x00404064 00 00 00 00 ....
0x00404068 00 00 00 00 ....
0x0040406C 00 00 00 00 ....
0x00404070 00 00 00 00 ....
0x00404074 00 00 00 00 ....
0x00404078 00 00 00 00 ....
```

Watch

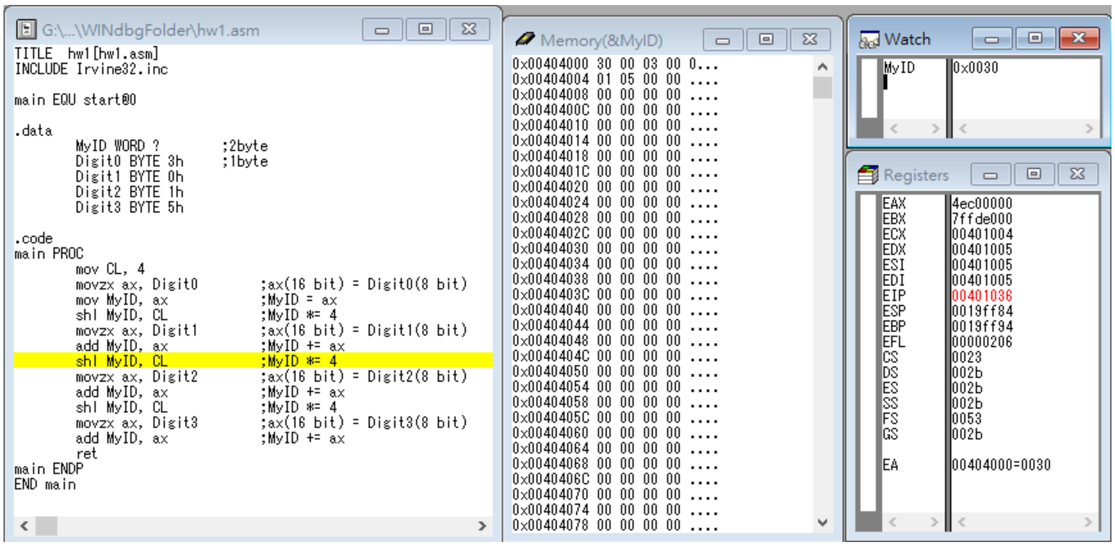
```
MyID     0x0030
```

Registers

```
EAX      4ec00000
EBX      7ffde000
ECX      00401004
EDX      00401005
ESI      00401005
EDI      00401005
EIP      0040102f
ESP      0019ff84
EBP      0019ff94
EFL      00000206
CS       0023
DS       002b
ES       002b
SS       002b
FS       0053
GS       002b

EA       00404000=0030
```

Step 6:add ax to MyID, MyID = 0x0030



G:\...\WINdbgFolder\hw1.asm

```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
        MyID WORD ?          ;2byte
        Digit0 BYTE 3h       ;1byte
        Digit1 BYTE 0h
        Digit2 BYTE 1h
        Digit3 BYTE 5h

.code
main PROC
        mov CL, 4
        movzx ax, Digit0          ;ax(16 bit) = Digit0(8 bit)
        mov MyID, ax              ;MyID = ax
        shl MyID, CL              ;MyID *= 4
        movzx ax, Digit1          ;ax(16 bit) = Digit1(8 bit)
        add MyID, ax              ;MyID += ax
        shl MyID, CL              ;MyID *= 4
        movzx ax, Digit2          ;ax(16 bit) = Digit2(8 bit)
        add MyID, ax              ;MyID += ax
        shl MyID, CL              ;MyID *= 4
        movzx ax, Digit3          ;ax(16 bit) = Digit3(8 bit)
        add MyID, ax              ;MyID += ax
        ret
main ENDP
END main
```

Memory(&MyID)

```
0x00404000 30 00 03 00 0...
0x00404004 01 05 00 00 ....
0x00404008 00 00 00 00 ....
0x0040400C 00 00 00 00 ....
0x00404010 00 00 00 00 ....
0x00404014 00 00 00 00 ....
0x00404018 00 00 00 00 ....
0x0040401C 00 00 00 00 ....
0x00404020 00 00 00 00 ....
0x00404024 00 00 00 00 ....
0x00404028 00 00 00 00 ....
0x0040402C 00 00 00 00 ....
0x00404030 00 00 00 00 ....
0x00404034 00 00 00 00 ....
0x00404038 00 00 00 00 ....
0x0040403C 00 00 00 00 ....
0x00404040 00 00 00 00 ....
0x00404044 00 00 00 00 ....
0x00404048 00 00 00 00 ....
0x0040404C 00 00 00 00 ....
0x00404050 00 00 00 00 ....
0x00404054 00 00 00 00 ....
0x00404058 00 00 00 00 ....
0x0040405C 00 00 00 00 ....
0x00404060 00 00 00 00 ....
0x00404064 00 00 00 00 ....
0x00404068 00 00 00 00 ....
0x0040406C 00 00 00 00 ....
0x00404070 00 00 00 00 ....
0x00404074 00 00 00 00 ....
0x00404078 00 00 00 00 ....
```

Watch

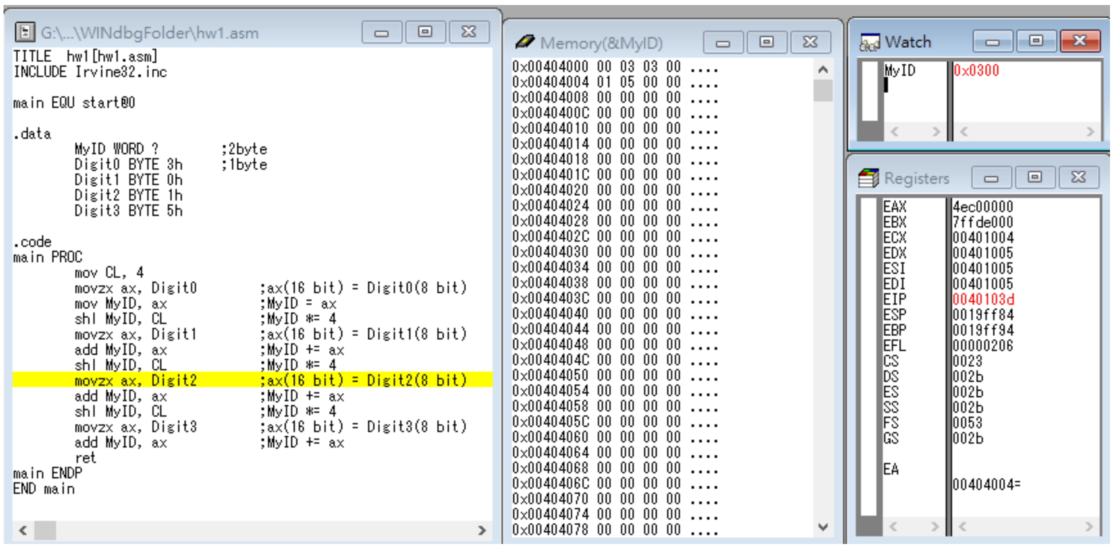
```
MyID     0x0030
```

Registers

```
EAX      4ec00000
EBX      7ffde000
ECX      00401004
EDX      00401005
ESI      00401005
EDI      00401005
EIP      00401036
ESP      0019ff84
EBP      0019ff94
EFL      00000206
CS       0023
DS       002b
ES       002b
SS       002b
FS       0053
GS       002b

EA       00404000=0030
```

Step 7: shl MyID for 4 bits, which makes MyID from 0x0030 to 0x0300



G:\...\WINdbgFolder\hw1.asm

```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
        MyID WORD ?          ;2byte
        Digit0 BYTE 3h       ;1byte
        Digit1 BYTE 0h
        Digit2 BYTE 1h
        Digit3 BYTE 5h

.code
main PROC
        mov CL, 4
        movzx ax, Digit0          ;ax(16 bit) = Digit0(8 bit)
        mov MyID, ax              ;MyID = ax
        shl MyID, CL              ;MyID *= 4
        movzx ax, Digit1          ;ax(16 bit) = Digit1(8 bit)
        add MyID, ax              ;MyID += ax
        shl MyID, CL              ;MyID *= 4
        movzx ax, Digit2          ;ax(16 bit) = Digit2(8 bit)
        add MyID, ax              ;MyID += ax
        shl MyID, CL              ;MyID *= 4
        movzx ax, Digit3          ;ax(16 bit) = Digit3(8 bit)
        add MyID, ax              ;MyID += ax
        ret
main ENDP
END main
```

Memory(&MyID)

```
0x00404000 00 03 03 00 ....
0x00404004 01 05 00 00 ....
0x00404008 00 00 00 00 ....
0x0040400C 00 00 00 00 ....
0x00404010 00 00 00 00 ....
0x00404014 00 00 00 00 ....
0x00404018 00 00 00 00 ....
0x0040401C 00 00 00 00 ....
0x00404020 00 00 00 00 ....
0x00404024 00 00 00 00 ....
0x00404028 00 00 00 00 ....
0x0040402C 00 00 00 00 ....
0x00404030 00 00 00 00 ....
0x00404034 00 00 00 00 ....
0x00404038 00 00 00 00 ....
0x0040403C 00 00 00 00 ....
0x00404040 00 00 00 00 ....
0x00404044 00 00 00 00 ....
0x00404048 00 00 00 00 ....
0x0040404C 00 00 00 00 ....
0x00404050 00 00 00 00 ....
0x00404054 00 00 00 00 ....
0x00404058 00 00 00 00 ....
0x0040405C 00 00 00 00 ....
0x00404060 00 00 00 00 ....
0x00404064 00 00 00 00 ....
0x00404068 00 00 00 00 ....
0x0040406C 00 00 00 00 ....
0x00404070 00 00 00 00 ....
0x00404074 00 00 00 00 ....
0x00404078 00 00 00 00 ....
```

Watch

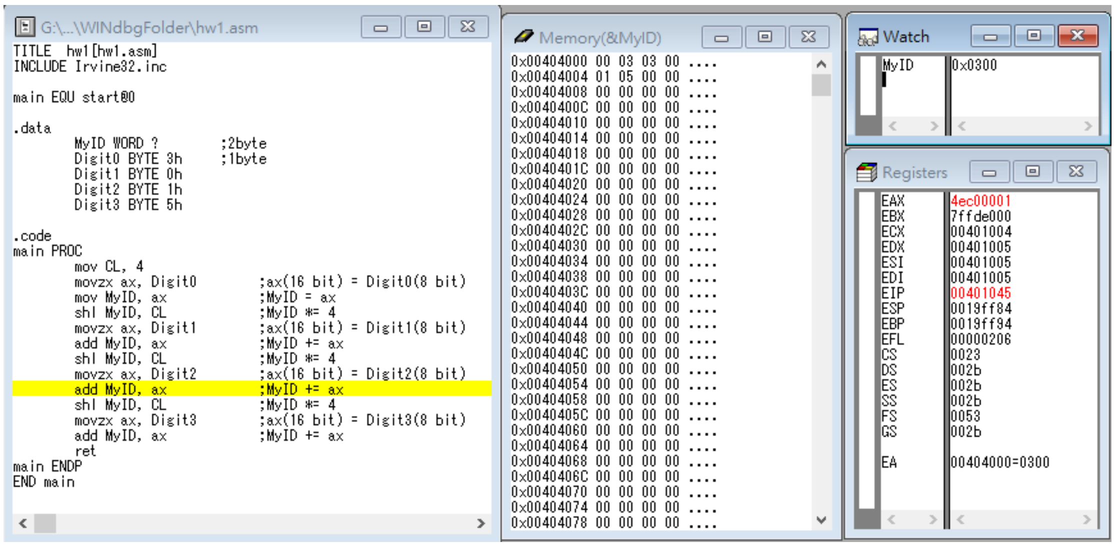
```
MyID     0x0300
```

Registers

```
EAX      4ec00000
EBX      7ffde000
ECX      00401004
EDX      00401005
ESI      00401005
EDI      00401005
EIP      0040103d
ESP      0019ff84
EBP      0019ff94
EFL      00000206
CS       0023
DS       002b
ES       002b
SS       002b
FS       0053
GS       002b

EA       00404004=
```

Step 8: movzx Digit2 to ax. ax = 0x0001



G:\...\WINdbgFolder\hw1.asm

```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
        MyID WORD ?           ;2byte
        Digit0 BYTE 3h        ;1byte
        Digit1 BYTE 0h
        Digit2 BYTE 1h
        Digit3 BYTE 5h

.code
main PROC
        mov CL, 4
        movzx ax, Digit0           ;ax(16 bit) = Digit0(8 bit)
        mov MyID, ax               ;MyID = ax
        shl MyID, CL               ;MyID *= 4
        movzx ax, Digit1           ;ax(16 bit) = Digit1(8 bit)
        add MyID, ax               ;MyID += ax
        shl MyID, CL               ;MyID *= 4
        movzx ax, Digit2           ;ax(16 bit) = Digit2(8 bit)
        add MyID, ax               ;MyID += ax
        shl MyID, CL               ;MyID *= 4
        movzx ax, Digit3           ;ax(16 bit) = Digit3(8 bit)
        add MyID, ax               ;MyID += ax
        ret
main ENDP
END main
```

Memory(&MyID)

```
0x00404000 00 03 03 00 ....
0x00404004 01 05 00 00 ....
0x00404008 00 00 00 00 ....
0x0040400C 00 00 00 00 ....
0x00404010 00 00 00 00 ....
0x00404014 00 00 00 00 ....
0x00404018 00 00 00 00 ....
0x0040401C 00 00 00 00 ....
0x00404020 00 00 00 00 ....
0x00404024 00 00 00 00 ....
0x00404028 00 00 00 00 ....
0x0040402C 00 00 00 00 ....
0x00404030 00 00 00 00 ....
0x00404034 00 00 00 00 ....
0x00404038 00 00 00 00 ....
0x0040403C 00 00 00 00 ....
0x00404040 00 00 00 00 ....
0x00404044 00 00 00 00 ....
0x00404048 00 00 00 00 ....
0x0040404C 00 00 00 00 ....
0x00404050 00 00 00 00 ....
0x00404054 00 00 00 00 ....
0x00404058 00 00 00 00 ....
0x0040405C 00 00 00 00 ....
0x00404060 00 00 00 00 ....
0x00404064 00 00 00 00 ....
0x00404068 00 00 00 00 ....
0x0040406C 00 00 00 00 ....
0x00404070 00 00 00 00 ....
0x00404074 00 00 00 00 ....
0x00404078 00 00 00 00 ....
```

Watch

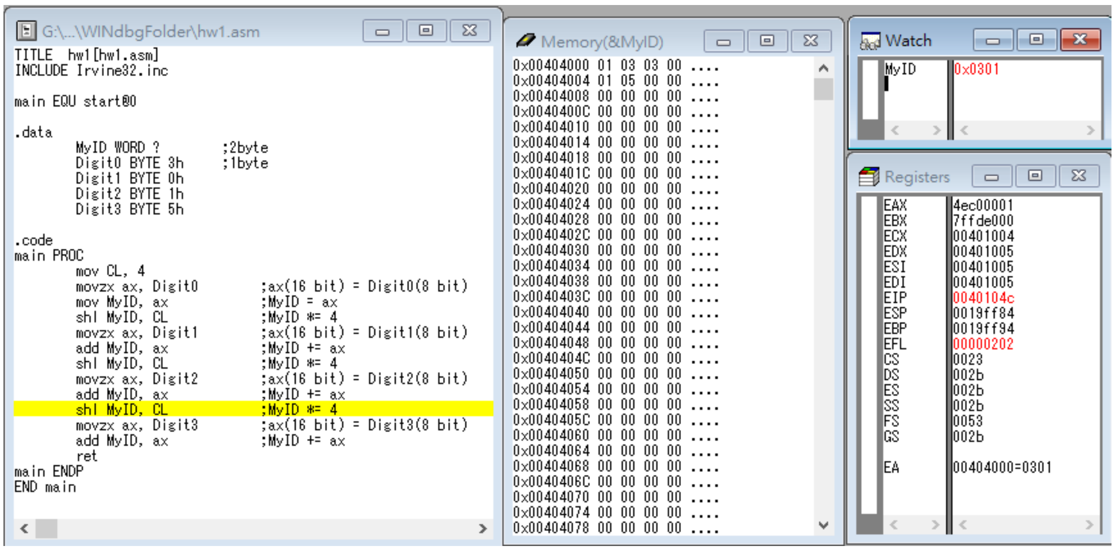
```
MyID    0x0300
```

Registers

```
EAX  4ec00001
EBX  7ffde000
ECX  00401004
EDX  00401005
ESI  00401005
EDI  00401005
EIP  00401045
ESP  0019ff84
EBP  0019ff94
EFL  00000206
CS   0023
DS   002b
ES   002b
SS   002b
FS   0053
GS   002b

EA   00404000=0300
```

Step 9: add ax to MyID, MyID = 0x0301



G:\...\WINdbgFolder\hw1.asm

```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
        MyID WORD ?           ;2byte
        Digit0 BYTE 3h        ;1byte
        Digit1 BYTE 0h
        Digit2 BYTE 1h
        Digit3 BYTE 5h

.code
main PROC
        mov CL, 4
        movzx ax, Digit0           ;ax(16 bit) = Digit0(8 bit)
        mov MyID, ax               ;MyID = ax
        shl MyID, CL               ;MyID *= 4
        movzx ax, Digit1           ;ax(16 bit) = Digit1(8 bit)
        add MyID, ax               ;MyID += ax
        shl MyID, CL               ;MyID *= 4
        movzx ax, Digit2           ;ax(16 bit) = Digit2(8 bit)
        add MyID, ax               ;MyID += ax
        shl MyID, CL               ;MyID *= 4
        movzx ax, Digit3           ;ax(16 bit) = Digit3(8 bit)
        add MyID, ax               ;MyID += ax
        ret
main ENDP
END main
```

Memory(&MyID)

```
0x00404000 01 03 03 00 ....
0x00404004 01 05 00 00 ....
0x00404008 00 00 00 00 ....
0x0040400C 00 00 00 00 ....
0x00404010 00 00 00 00 ....
0x00404014 00 00 00 00 ....
0x00404018 00 00 00 00 ....
0x0040401C 00 00 00 00 ....
0x00404020 00 00 00 00 ....
0x00404024 00 00 00 00 ....
0x00404028 00 00 00 00 ....
0x0040402C 00 00 00 00 ....
0x00404030 00 00 00 00 ....
0x00404034 00 00 00 00 ....
0x00404038 00 00 00 00 ....
0x0040403C 00 00 00 00 ....
0x00404040 00 00 00 00 ....
0x00404044 00 00 00 00 ....
0x00404048 00 00 00 00 ....
0x0040404C 00 00 00 00 ....
0x00404050 00 00 00 00 ....
0x00404054 00 00 00 00 ....
0x00404058 00 00 00 00 ....
0x0040405C 00 00 00 00 ....
0x00404060 00 00 00 00 ....
0x00404064 00 00 00 00 ....
0x00404068 00 00 00 00 ....
0x0040406C 00 00 00 00 ....
0x00404070 00 00 00 00 ....
0x00404074 00 00 00 00 ....
0x00404078 00 00 00 00 ....
```

Watch

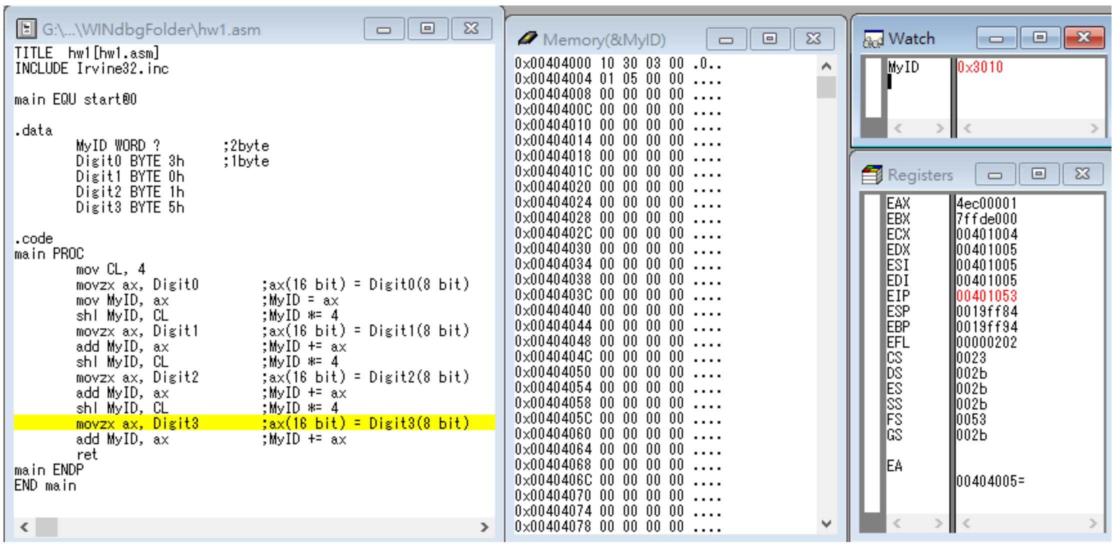
```
MyID    0x0301
```

Registers

```
EAX  4ec00001
EBX  7ffde000
ECX  00401004
EDX  00401005
ESI  00401005
EDI  00401005
EIP  0040104c
ESP  0019ff84
EBP  0019ff94
EFL  00000202
CS   0023
DS   002b
ES   002b
SS   002b
FS   0053
GS   002b

EA   00404000=0301
```

Step 10: shl MyID for 4 bits, which makes MyID from 0x0301 to 0x3010



G:\...\WINdbgFolder\hw1.asm

```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
        MyID WORD ?           ;2byte
        Digit0 BYTE 3h        ;1byte
        Digit1 BYTE 0h
        Digit2 BYTE 1h
        Digit3 BYTE 5h

.code
main PROC
        mov CL, 4
        movzx ax, Digit0           ;ax(16 bit) = Digit0(8 bit)
        mov MyID, ax               ;MyID = ax
        shl MyID, CL               ;MyID *= 4
        movzx ax, Digit1           ;ax(16 bit) = Digit1(8 bit)
        add MyID, ax               ;MyID += ax
        shl MyID, CL               ;MyID *= 4
        movzx ax, Digit2           ;ax(16 bit) = Digit2(8 bit)
        add MyID, ax               ;MyID += ax
        shl MyID, CL               ;MyID *= 4
        movzx ax, Digit3           ;ax(16 bit) = Digit3(8 bit)
        add MyID, ax               ;MyID += ax
        ret
main ENDP
END main
```

Memory(&MyID)

```
0x00404000 10 30 03 00 .0..
0x00404004 01 05 00 00 ....
0x00404008 00 00 00 00 ....
0x0040400C 00 00 00 00 ....
0x00404010 00 00 00 00 ....
0x00404014 00 00 00 00 ....
0x00404018 00 00 00 00 ....
0x0040401C 00 00 00 00 ....
0x00404020 00 00 00 00 ....
0x00404024 00 00 00 00 ....
0x00404028 00 00 00 00 ....
0x0040402C 00 00 00 00 ....
0x00404030 00 00 00 00 ....
0x00404034 00 00 00 00 ....
0x00404038 00 00 00 00 ....
0x0040403C 00 00 00 00 ....
0x00404040 00 00 00 00 ....
0x00404044 00 00 00 00 ....
0x00404048 00 00 00 00 ....
0x0040404C 00 00 00 00 ....
0x00404050 00 00 00 00 ....
0x00404054 00 00 00 00 ....
0x00404058 00 00 00 00 ....
0x0040405C 00 00 00 00 ....
0x00404060 00 00 00 00 ....
0x00404064 00 00 00 00 ....
0x00404068 00 00 00 00 ....
0x0040406C 00 00 00 00 ....
0x00404070 00 00 00 00 ....
0x00404074 00 00 00 00 ....
0x00404078 00 00 00 00 ....
```

Watch

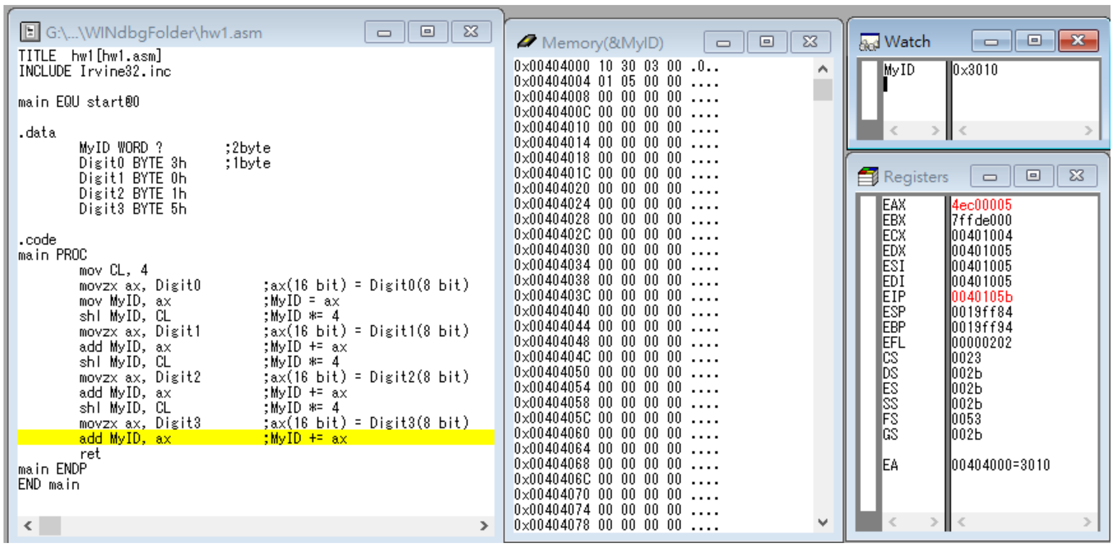
```
MyID    0x3010
```

Registers

```
EAX  4ec00001
EBX  7ffde000
ECX  00401004
EDX  00401005
ESI  00401005
EDI  00401005
EIP  00401053
ESP  0019ff84
EBP  0019ff94
EFL  00000202
CS   0023
DS   002b
ES   002b
SS   002b
FS   0053
GS   002b

EA   00404005=
```

Step 11: movzx Digit2 to ax. ax = 0x0005



G:\...\WINdbgFolder\hw1.asm

```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
        MyID WORD ?             ;2byte
        Digit0 BYTE 3h          ;1byte
        Digit1 BYTE 0h
        Digit2 BYTE 1h
        Digit3 BYTE 5h

.code
main PROC
        mov CL, 4
        movzx ax, Digit0             ;ax(16 bit) = Digit0(8 bit)
        mov MyID, ax                 ;MyID = ax
        shl MyID, CL                 ;MyID *= 4
        movzx ax, Digit1             ;ax(16 bit) = Digit1(8 bit)
        add MyID, ax                 ;MyID += ax
        shl MyID, CL                 ;MyID *= 4
        movzx ax, Digit2             ;ax(16 bit) = Digit2(8 bit)
        add MyID, ax                 ;MyID += ax
        shl MyID, CL                 ;MyID *= 4
        movzx ax, Digit3             ;ax(16 bit) = Digit3(8 bit)
        add MyID, ax                 ;MyID += ax
        ret
main ENDP
END main
```

Memory(&MyID)

```
0x00404000 10 30 03 00 .0..
0x00404004 01 05 00 00 ....
0x00404008 00 00 00 00 ....
0x0040400C 00 00 00 00 ....
0x00404010 00 00 00 00 ....
0x00404014 00 00 00 00 ....
0x00404018 00 00 00 00 ....
0x0040401C 00 00 00 00 ....
0x00404020 00 00 00 00 ....
0x00404024 00 00 00 00 ....
0x00404028 00 00 00 00 ....
0x0040402C 00 00 00 00 ....
0x00404030 00 00 00 00 ....
0x00404034 00 00 00 00 ....
0x00404038 00 00 00 00 ....
0x0040403C 00 00 00 00 ....
0x00404040 00 00 00 00 ....
0x00404044 00 00 00 00 ....
0x00404048 00 00 00 00 ....
0x0040404C 00 00 00 00 ....
0x00404050 00 00 00 00 ....
0x00404054 00 00 00 00 ....
0x00404058 00 00 00 00 ....
0x0040405C 00 00 00 00 ....
0x00404060 00 00 00 00 ....
0x00404064 00 00 00 00 ....
0x00404068 00 00 00 00 ....
0x0040406C 00 00 00 00 ....
0x00404070 00 00 00 00 ....
0x00404074 00 00 00 00 ....
0x00404078 00 00 00 00 ....
```

Watch

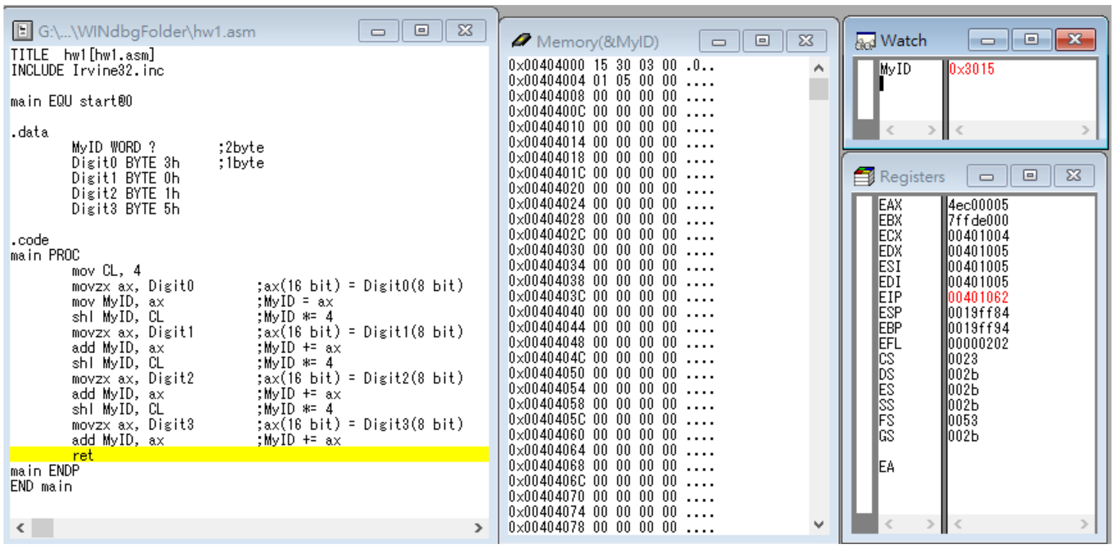
```
MyID   0x3010
```

Registers

```
EAX  4ec00005
EBX  7ffde000
ECX  00401004
EDX  00401005
ESI  00401005
EDI  00401005
EIP  0040105b
ESP  0019ff84
EBP  0019ff94
EFL  00000202
CS   0023
DS   002b
ES   002b
SS   002b
FS   0053
GS   002b

EA   00404000=3010
```

Step 12: add ax to MyID, MyID = 0x3015



G:\...\WINdbgFolder\hw1.asm

```
TITLE  hw1[hw1.asm]
INCLUDE Irvine32.inc

main EQU start@0

.data
        MyID WORD ?             ;2byte
        Digit0 BYTE 3h          ;1byte
        Digit1 BYTE 0h
        Digit2 BYTE 1h
        Digit3 BYTE 5h

.code
main PROC
        mov CL, 4
        movzx ax, Digit0             ;ax(16 bit) = Digit0(8 bit)
        mov MyID, ax                 ;MyID = ax
        shl MyID, CL                 ;MyID *= 4
        movzx ax, Digit1             ;ax(16 bit) = Digit1(8 bit)
        add MyID, ax                 ;MyID += ax
        shl MyID, CL                 ;MyID *= 4
        movzx ax, Digit2             ;ax(16 bit) = Digit2(8 bit)
        add MyID, ax                 ;MyID += ax
        shl MyID, CL                 ;MyID *= 4
        movzx ax, Digit3             ;ax(16 bit) = Digit3(8 bit)
        add MyID, ax                 ;MyID += ax
        ret
main ENDP
END main
```

Memory(&MyID)

```
0x00404000 15 30 03 00 .0..
0x00404004 01 05 00 00 ....
0x00404008 00 00 00 00 ....
0x0040400C 00 00 00 00 ....
0x00404010 00 00 00 00 ....
0x00404014 00 00 00 00 ....
0x00404018 00 00 00 00 ....
0x0040401C 00 00 00 00 ....
0x00404020 00 00 00 00 ....
0x00404024 00 00 00 00 ....
0x00404028 00 00 00 00 ....
0x0040402C 00 00 00 00 ....
0x00404030 00 00 00 00 ....
0x00404034 00 00 00 00 ....
0x00404038 00 00 00 00 ....
0x0040403C 00 00 00 00 ....
0x00404040 00 00 00 00 ....
0x00404044 00 00 00 00 ....
0x00404048 00 00 00 00 ....
0x0040404C 00 00 00 00 ....
0x00404050 00 00 00 00 ....
0x00404054 00 00 00 00 ....
0x00404058 00 00 00 00 ....
0x0040405C 00 00 00 00 ....
0x00404060 00 00 00 00 ....
0x00404064 00 00 00 00 ....
0x00404068 00 00 00 00 ....
0x0040406C 00 00 00 00 ....
0x00404070 00 00 00 00 ....
0x00404074 00 00 00 00 ....
0x00404078 00 00 00 00 ....
```

Watch

```
MyID   0x3015
```

Registers

```
EAX  4ec00005
EBX  7ffde000
ECX  00401004
EDX  00401005
ESI  00401005
EDI  00401005
EIP  00401062
ESP  0019ff84
EBP  0019ff94
EFL  00000202
CS   0023
DS   002b
ES   002b
SS   002b
FS   0053
GS   002b

EA
```

Review:

At first time, I used "add" instruction to count Digit0 to Digit4 and "mov" to MyID, but the answer was wrong and I realized that it couldn't be done by simple addition. I search for some idea from Google and found that I should use "shl" to swift the last bit to left. This way, it would not be a problem to merge other value to "MyID". Kept doing in this way for 4 times but without using "shl" in the last time, the output would be correct. By the way, may I ask how detailed should I screenshot the steps of the code? If homework become more complicated, it would be tough to put all screenshot in the report.