SNS COLLEGE OF ENGINEERING

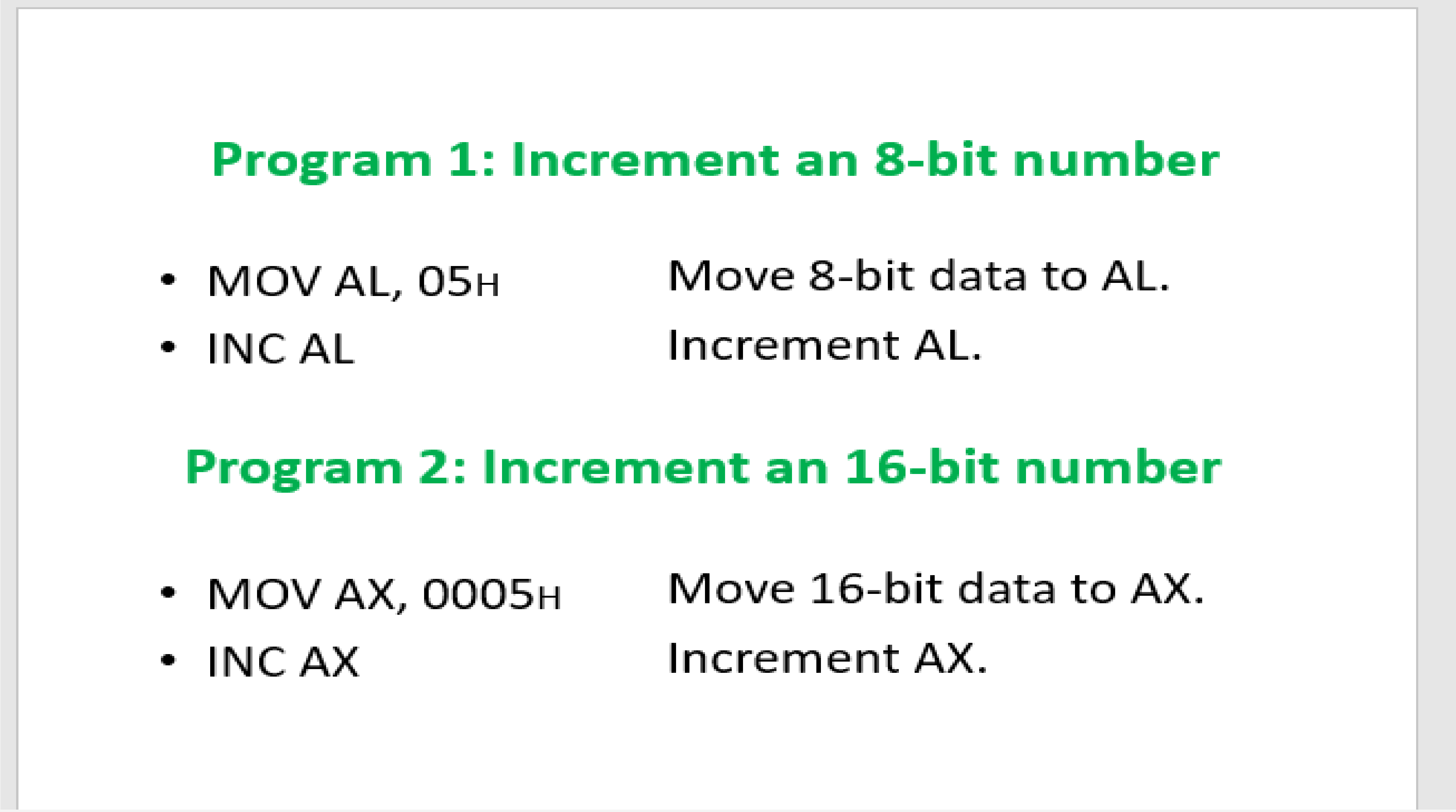
Kurumbapalayam(Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NAAC-UGC with ‘A’ Grade

Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

Assembly Language Programming(ALP) **8086**

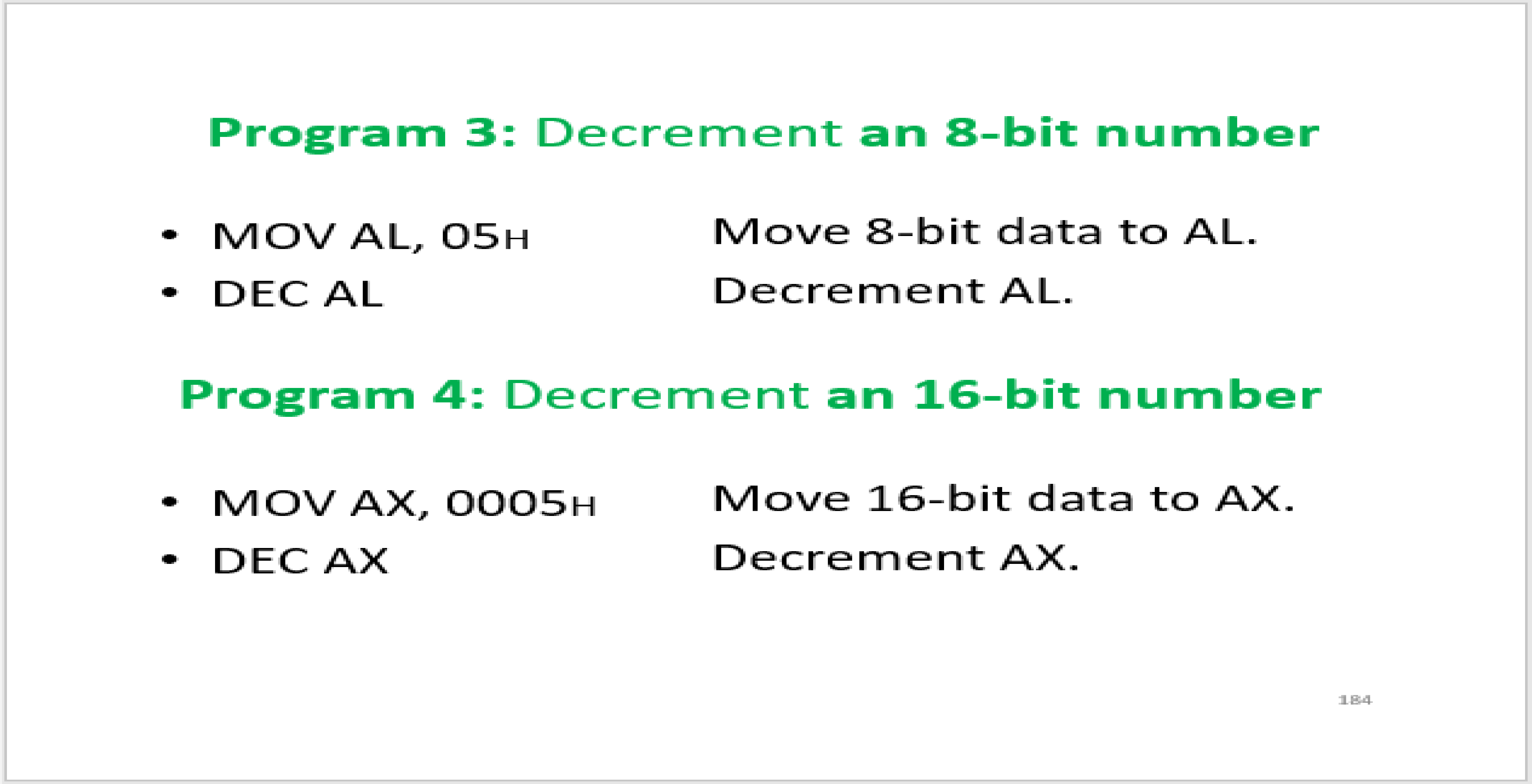


Program 1: Increment an 8-bit number

• MOV AL, 05H	Move 8-bit data to AL.
• INC AL	Increment AL.

Program 2: Increment an 16-bit number

• MOV AX, 0005H	Move 16-bit data to AX.
• INC AX	Increment AX.

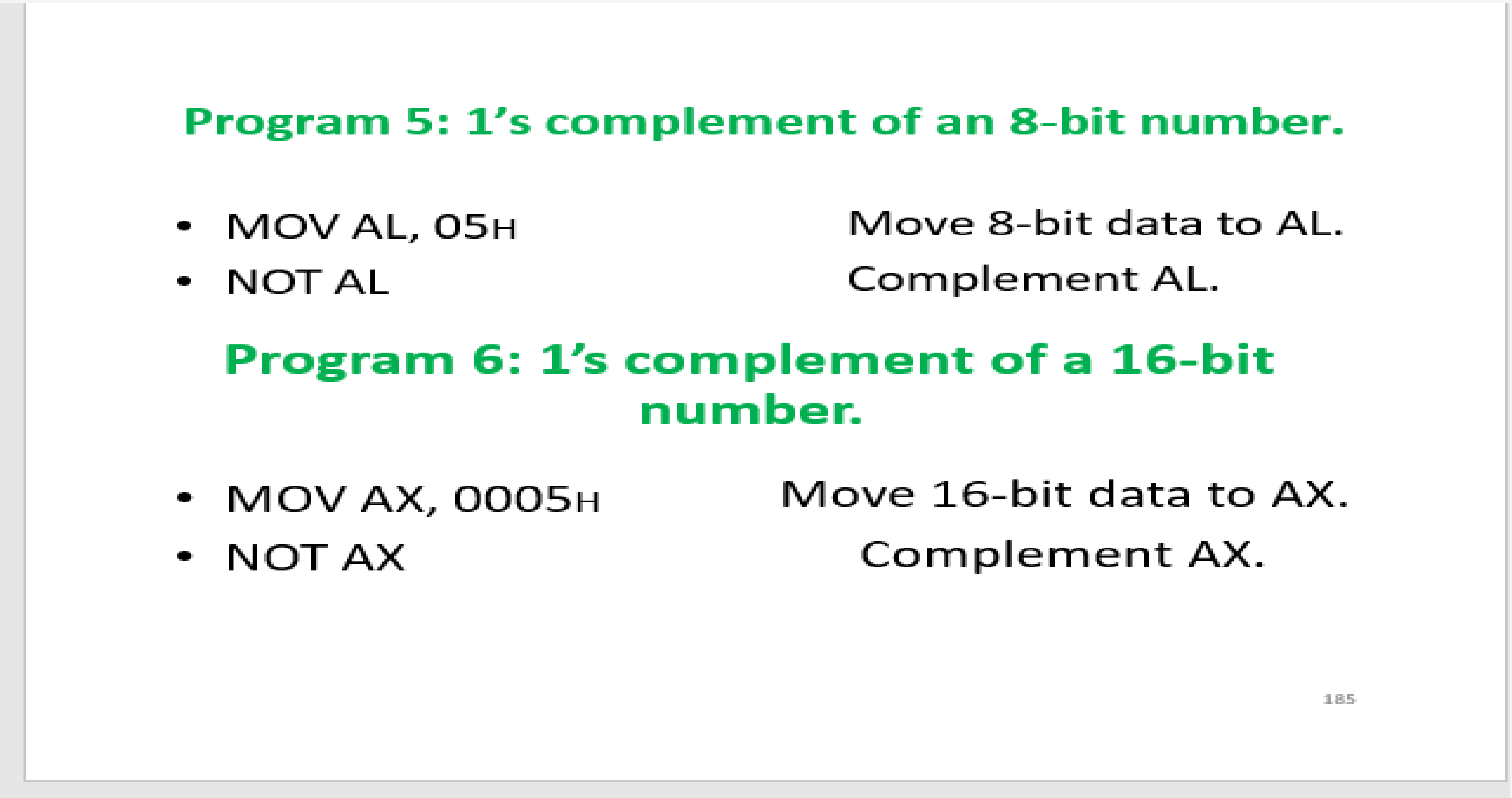


Program 3: Decrement an 8-bit number

• MOV AL, 05H	Move 8-bit data to AL.
• DEC AL	Decrement AL.

Program 4: Decrement an 16-bit number

• MOV AX, 0005H	Move 16-bit data to AX.
• DEC AX	Decrement AX.

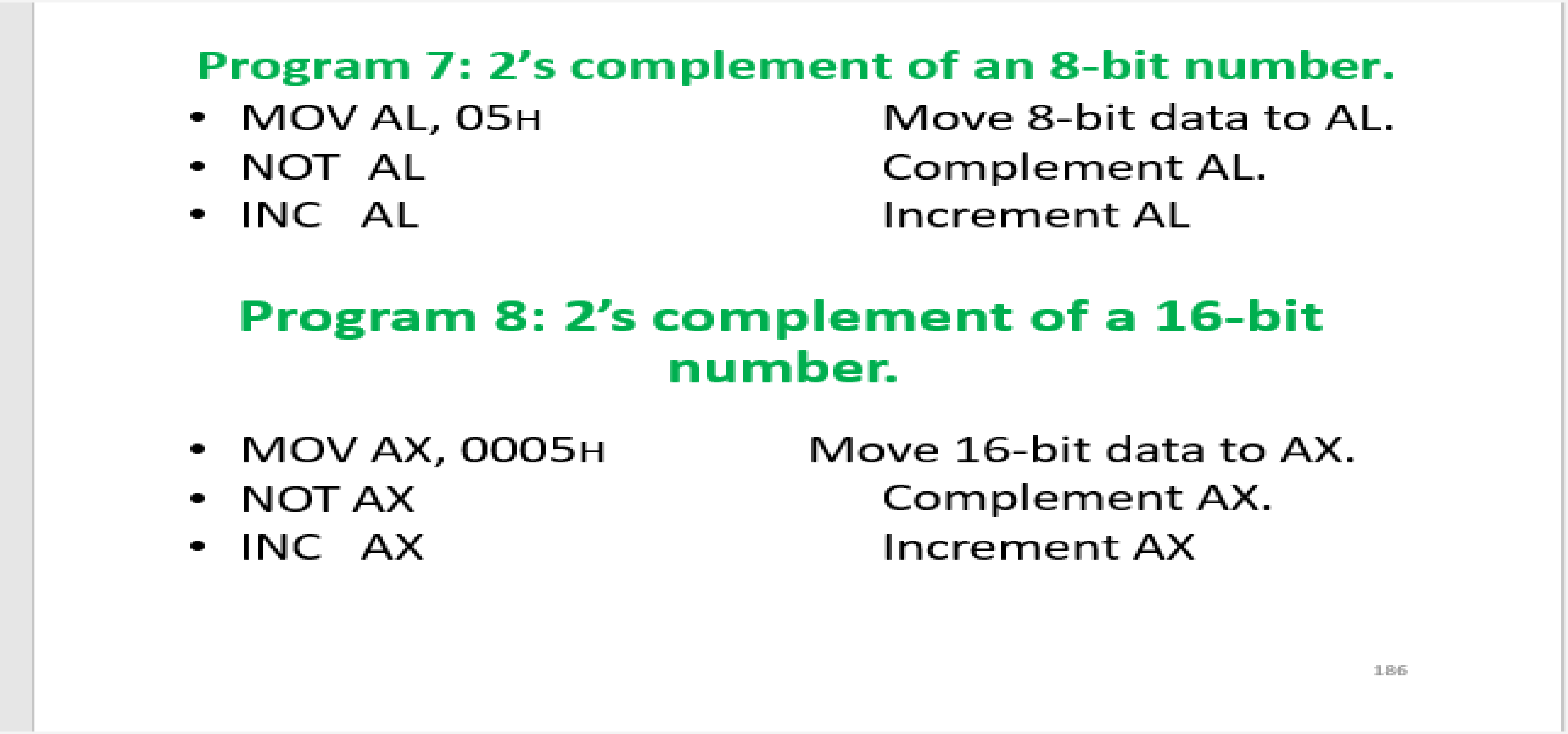


Program 5: 1’s complement of an 8-bit number.

• MOV AL, 05H	Move 8-bit data to AL.
• NOT AL	Complement AL.

Program 6: 1’s complement of a 16-bit number.

• MOV AX, 0005H	Move 16-bit data to AX.
• NOT AX	Complement AX.

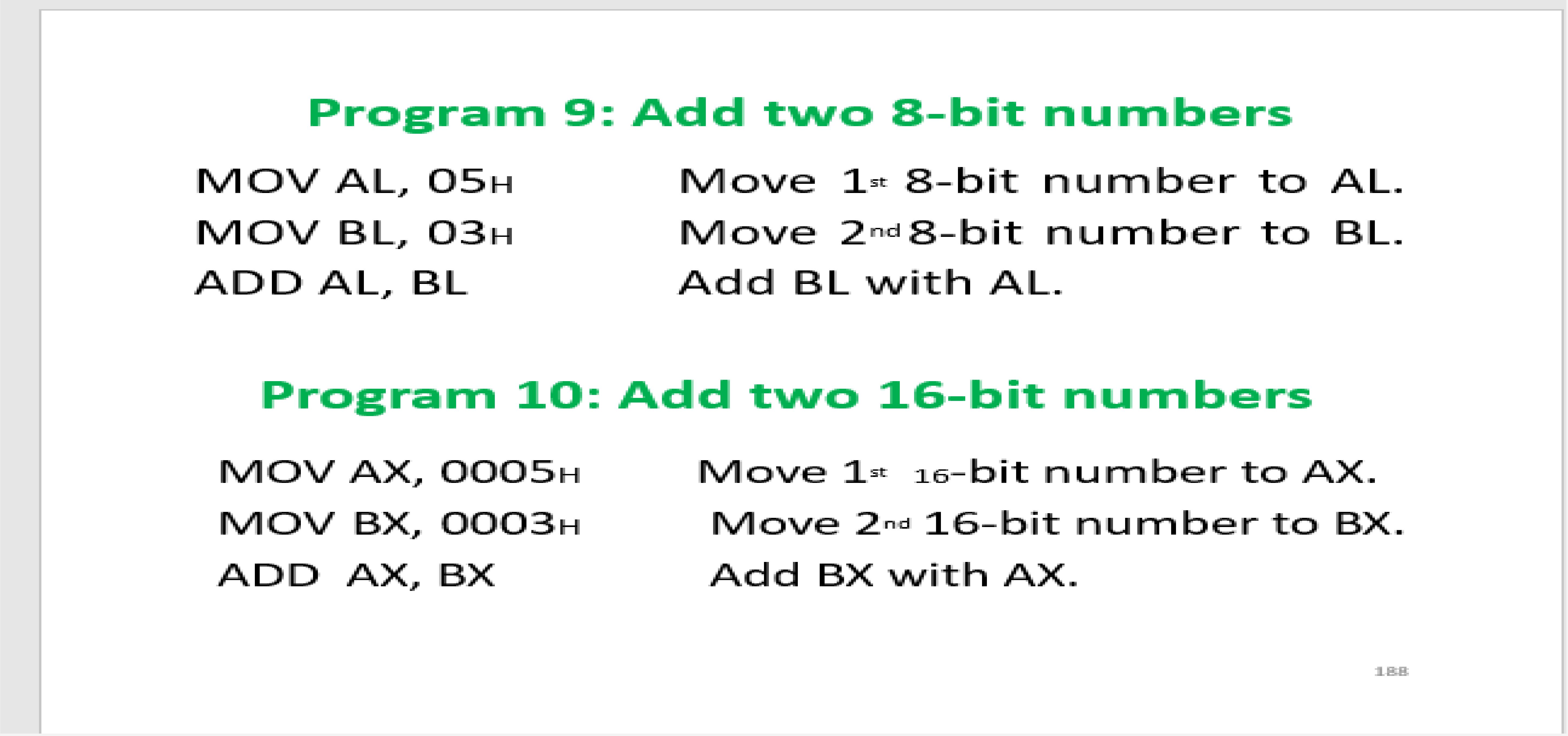


Program 7: 2’s complement of an 8-bit number.

```
• MOV AL, 05H                 Move 8-bit data to AL.
• NOT  AL                     Complement AL.
• INC   AL                    Increment AL
```

Program 8: 2’s complement of a 16-bit number.

```
• MOV AX, 0005H           Move 16-bit data to AX.
• NOT AX                      Complement AX.
• INC   AX                    Increment AX
```

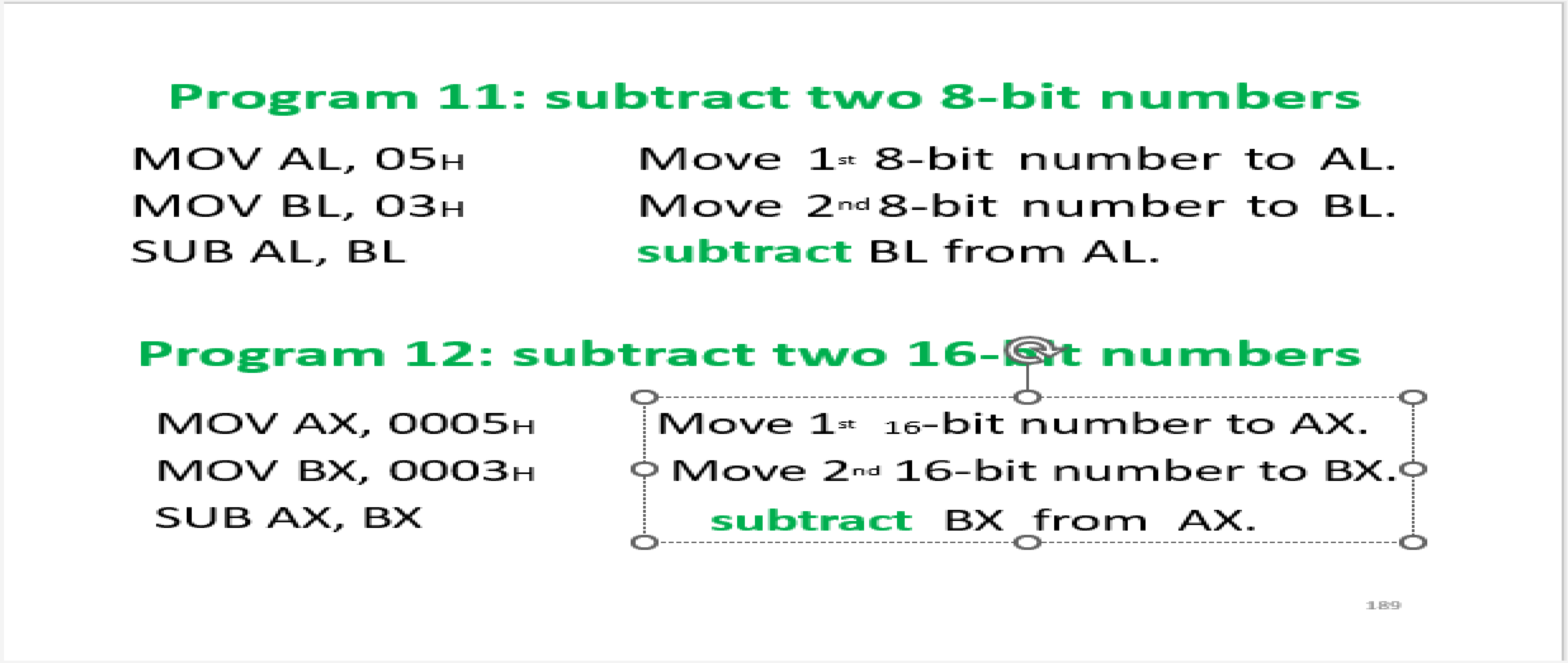


Program 9: Add two 8-bit numbers

```
MOV AL, 05H          Move 1st 8-bit number to AL.
MOV BL, 03H          Move 2nd 8-bit number to BL.
ADD AL, BL           Add BL with AL.
```

Program 10: Add two 16-bit numbers

```
MOV AX, 0005H        Move 1st 16-bit number to AX.
MOV BX, 0003H        Move 2nd 16-bit number to BX.
ADD  AX, BX          Add BX with AX.
```

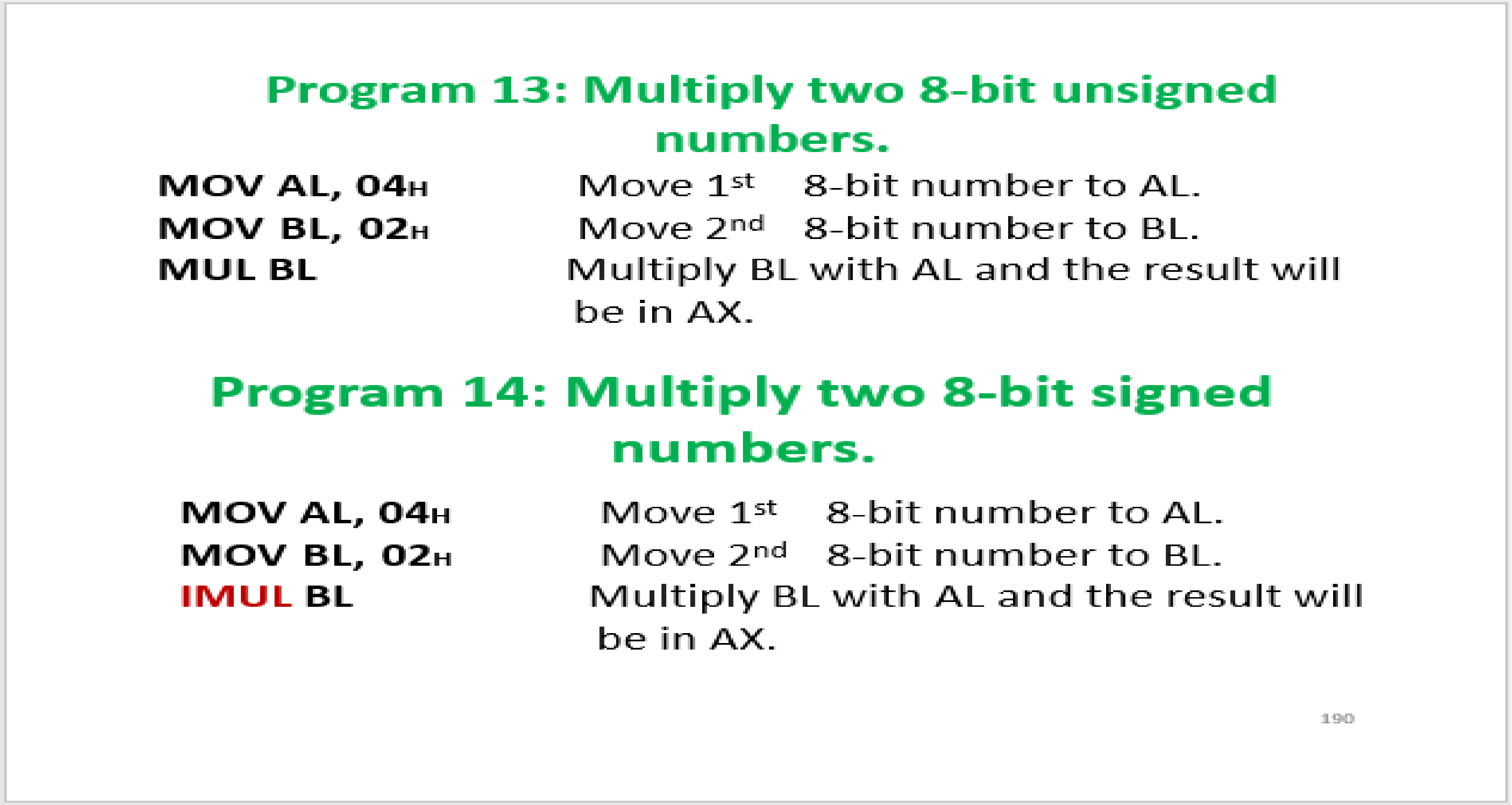


Program 11: subtract two 8-bit numbers

```
MOV AL, 05H          Move 1st 8-bit number to AL.
MOV BL, 03H          Move 2nd 8-bit number to BL.
SUB AL, BL           subtract BL from AL.
```

Program 12: subtract two 16-bit numbers

```
MOV AX, 0005H        Move 1st 16-bit number to AX.
MOV BX, 0003H        Move 2nd 16-bit number to BX.
SUB AX, BX           subtract BX from AX.
```

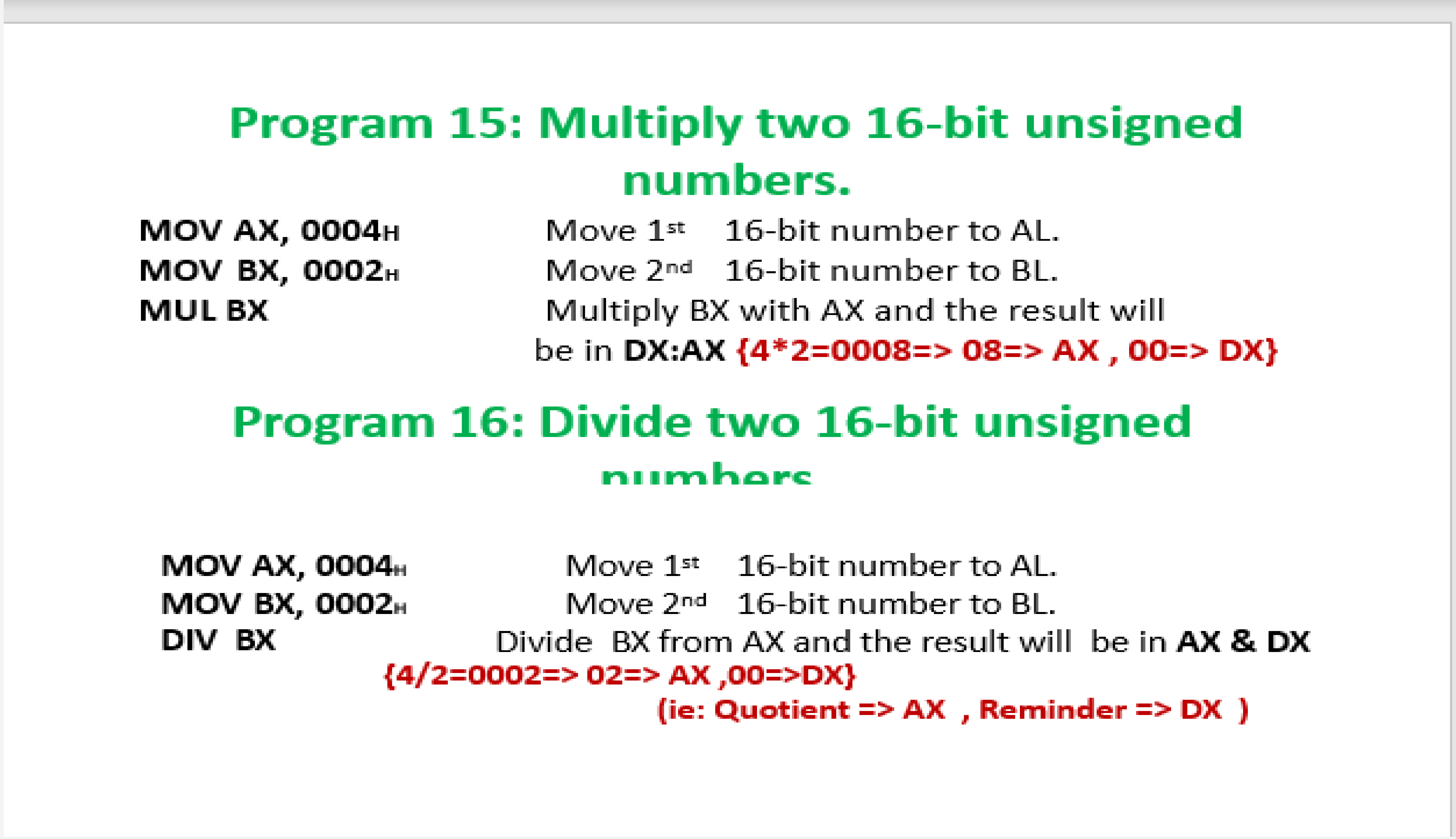


Program 13: Multiply two 8-bit unsigned numbers.

```
MOV AL, 04H          Move 1st  8-bit number to AL.
MOV BL, 02H          Move 2nd  8-bit number to BL.
MUL BL              Multiply BL with AL and the result will
                     be in AX.
```

Program 14: Multiply two 8-bit signed numbers.

```
MOV AL, 04H          Move 1st  8-bit number to AL.
MOV BL, 02H          Move 2nd  8-bit number to BL.
IMUL BL             Multiply BL with AL and the result will
                     be in AX.
```



Program 15: Multiply two 16-bit unsigned numbers.

MOV AX, 0004$_H$ Move 1st 16-bit number to AL.

MOV BX, 0002$_H$ Move 2nd 16-bit number to BL.

MUL BX Multiply BX with AX and the result will be in **DX:AX** {4*2=0008=> 08=> AX , 00=> DX}

Program 16: Divide two 16-bit unsigned numbers

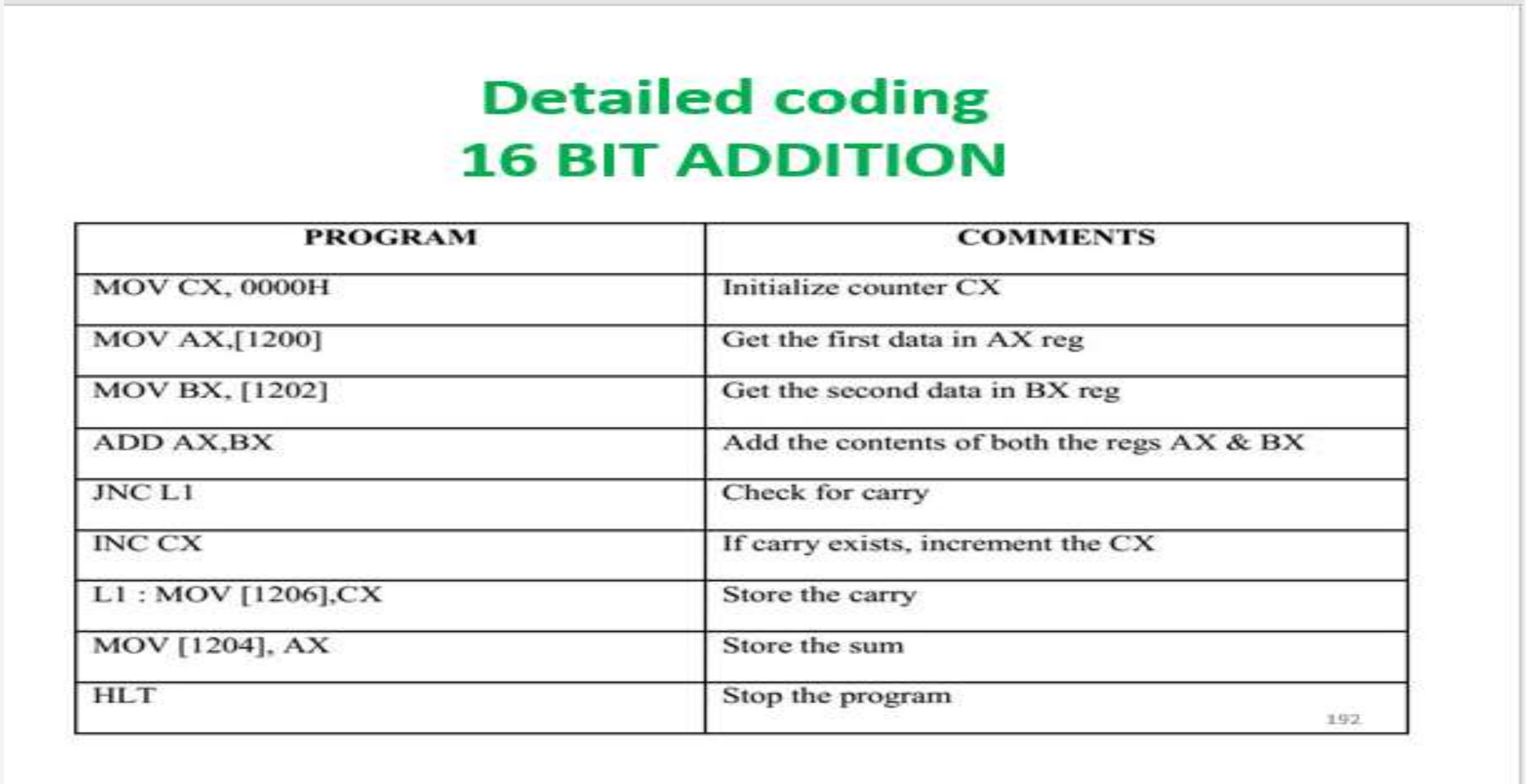
MOV AX, 0004$_H$ Move 1st 16-bit number to AL.

MOV BX, 0002$_H$ Move 2nd 16-bit number to BL.

DIV BX Divide BX from AX and the result will be in **AX & DX**

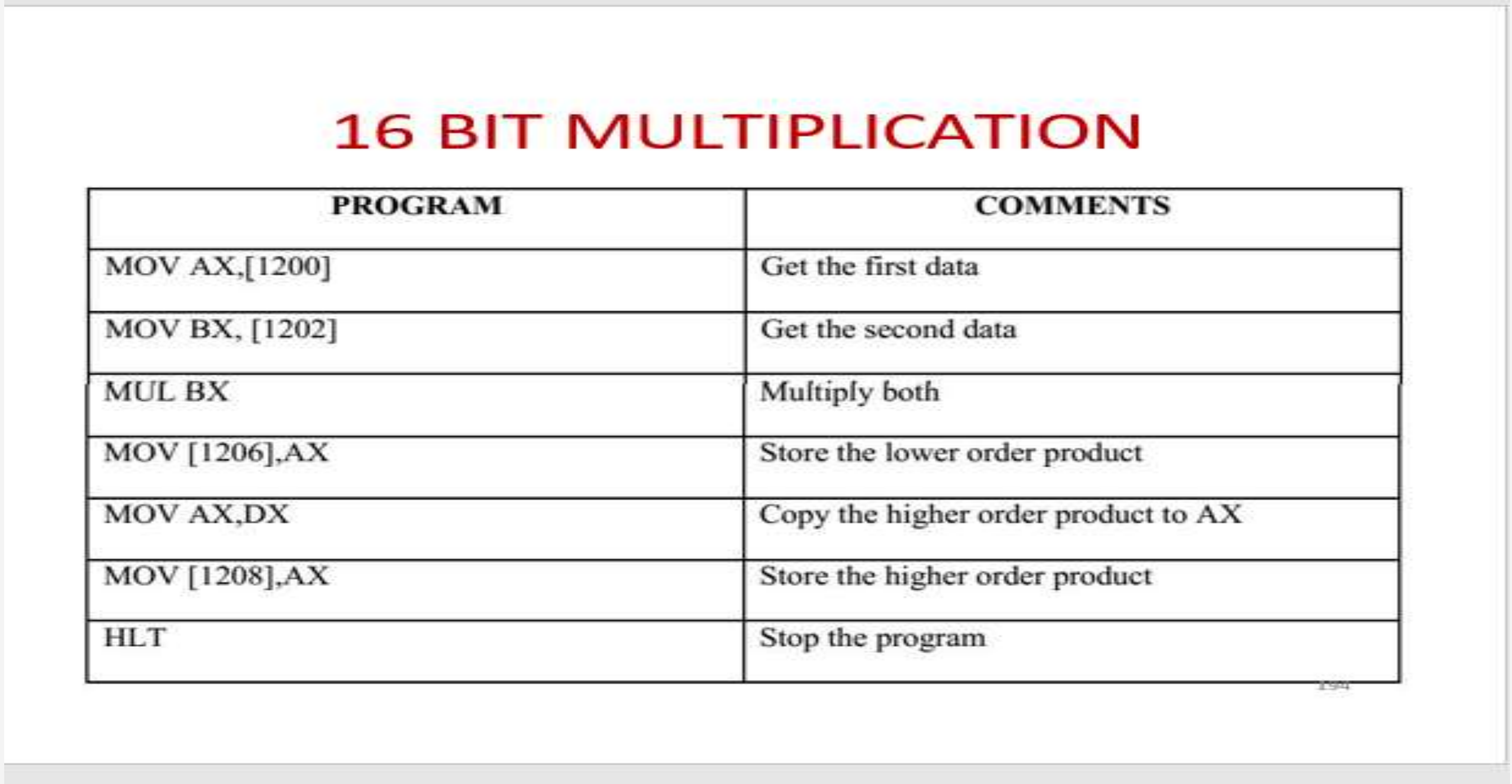
{4/2=0002=> 02=> AX ,00=>DX}

(ie: Quotient => AX , Reminder => DX)



Detailed coding
16 BIT ADDITION

PROGRAM	COMMENTS
MOV CX, 0000H	Initialize counter CX
MOV AX,[1200]	Get the first data in AX reg
MOV BX, [1202]	Get the second data in BX reg
ADD AX,BX	Add the contents of both the regs AX & BX
JNC L1	Check for carry
INC CX	If carry exists, increment the CX
L1 : MOV [1206],CX	Store the carry
MOV [1204], AX	Store the sum
HLT	Stop the program



16 BIT MULTIPLICATION

PROGRAM	COMMENTS
MOV AX,[1200]	Get the first data
MOV BX, [1202]	Get the second data
MUL BX	Multiply both
MOV [1206],AX	Store the lower order product
MOV AX,DX	Copy the higher order product to AX
MOV [1208],AX	Store the higher order product
HLT	Stop the program

THANK YOU