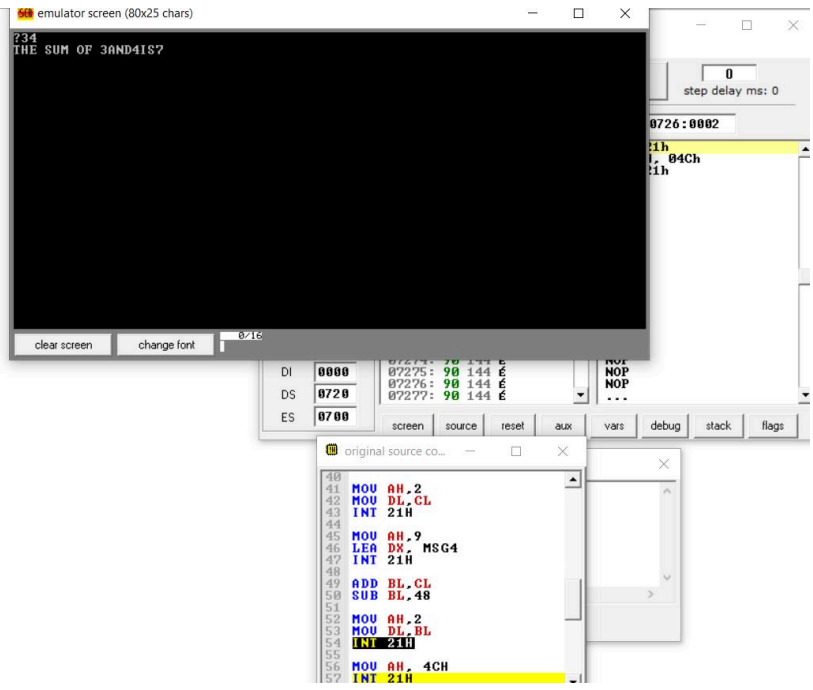
Program#1

```
.MODEL SMALL
.STACK 100H
.DATA

MSG1 DB "?$"
MSG2 DB 10,13, "THE SUM OF $"
MSG3 DB "AND$"
MSG4 DB "IS$"

.CODE
MAIN PROC
    MOV AX, @DATA
MOV DS, AX

MOV AH, 9
LEA DX, MSG1
INT 21H

MOV AH, 1
INT 21H

MOV BL,AL

MOV AH,1
INT 21H

MOV CL,AL

MOV AH,9
LEA DX,MSG2
INT 21H

MOV AH,2
MOV DL,BL
INT 21H

MOV AH,9
LEA DX, MSG3
INT 21H

MOV AH,2
MOV DL,CL
INT 21H

MOV AH,9
LEA DX, MSG4
INT 21H

ADD BL,CL
SUB BL,48

MOV AH,2
MOV DL,BL
INT 21H

MOV AH, 4CH
INT 21H
```

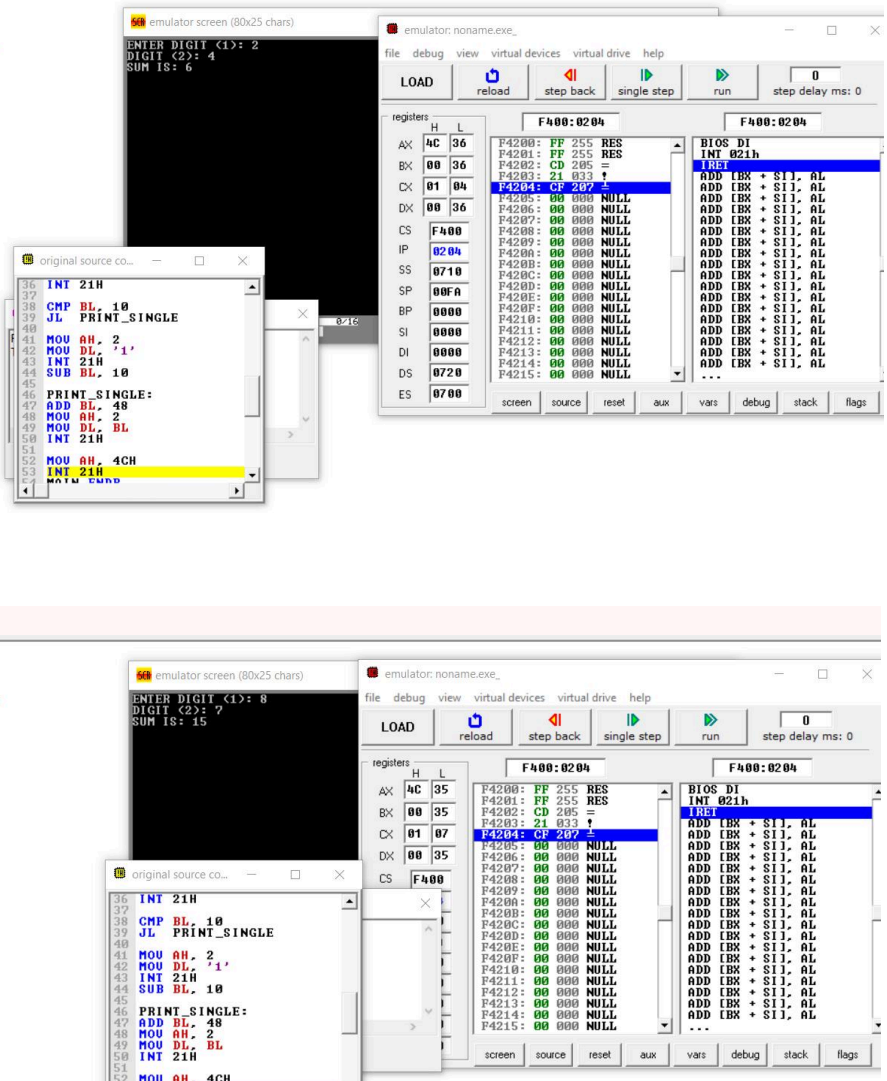


Program#2

Single-Digit Decimal Adder with ASCII Conversion

```
; ================================================================================
; PURPOSE:
; This program takes two single-digit inputs from the user, calculates their
; sum, and prints the result. It includes logic to correctly handle both
; single-digit results (0-9) and two-digit results (10-18).
;
; WORKING STEPS:
; 1. Initialization: Setup the Data Segment so the CPU can access messages.
; 2. Input Phase: Prompt user for two digits, read them (INT 21H, AH=1), and
;    subtract 48 (ASCII conversion) to treat them as actual integers.
; 3. Processing: Perform addition (ADD) of the two values.
; 4. Output Phase: Compare the sum. If >= 10, print '1' manually and adjust
;    the remaining value, then print the final digit as an ASCII character.
; ================================================================================
```

```
.MODEL SMALL
.STACK 100H

.DATA
    MSG1 DB 'ENTER DIGIT (1): $'
    MSG2 DB 13, 10, 'DIGIT (2): $'
    MSG3 DB 13, 10, 'SUM IS: $'

.CODE
MAIN PROC
    MOV AX, @DATA
    MOV DS, AX

    MOV AH, 9
    LEA DX, MSG1
    INT 21H

    MOV AH, 1
    INT 21H
    SUB AL, 48
    MOV BL, AL

    MOV AH, 9
    LEA DX, MSG2
    INT 21H

    MOV AH, 1
    INT 21H
    SUB AL, 48
    MOV CL, AL

    ADD BL, CL

    MOV AH, 9
    LEA DX, MSG3
    INT 21H

    CMP BL, 10
    JL  PRINT_SINGLE

    MOV AH, 2
    MOV DL, '1'
    INT 21H
    SUB BL, 10

PRINT_SINGLE:
    ADD BL, 48
    MOV AH, 2
    MOV DL, BL
    INT 21H

    MOV AH, 4CH
    INT 21H
MAIN ENDP
END MAIN
```




```
.MODEL SMALL
.STACK 100H

.DATA
    MSG1 DB 'ENTER DIGIT (1): $'
    MSG2 DB 13, 10, 'DIGIT (2): $'
    MSG3 DB 13, 10, 'SUM IS: $'

.CODE
MAIN PROC
    MOV AX, @DATA
    MOV DS, AX

    MOV AH, 9
    LEA DX, MSG1
    INT 21H

    MOV AH, 1
    INT 21H
    SUB AL, 48
    MOV BL, AL

    MOV AH, 9
    LEA DX, MSG2
    INT 21H

    MOV AH, 1
    INT 21H
    SUB AL, 48
    MOV CL, AL

    ADD BL, CL

    MOV AH, 9
    LEA DX, MSG3
    INT 21H

    CMP BL, 10
    JL  PRINT_SINGLE

    MOV AH, 2
    MOV DL, '1'
    INT 21H
    SUB BL, 10

PRINT_SINGLE:
    ADD BL, 48
    MOV AH, 2
    MOV DL, BL
    INT 21H

    MOV AH, 4CH
    INT 21H
MAIN ENDP
END MAIN
```