Εργαστήριο Μικροϋπολογιστών

2η Εργαστηριακή Άσκηση

Γενικό θέμα 8085 - Άσκηση 5

```
       IN 10H
       MVI D,00H    ; Initialize the mod(256) accumulator.

       CALL PRINT

RD:
       CALL KIND    ; Reading x.
       CPI 83H      ; Check if STORE/INCR was pressed.
       JNZ CONT1    ; If so, do accumulator stuff.
       CALL ACCUMULATE
       JMP RD

CONT1:
       CPI 81H      ; Check if DECR was pressed.
       JNZ CONT2
       CALL CLEAR   ; If so, reset accumulator.
       JMP RD

CONT2:
       LXI H,0905H  ; We store the display data at address 0900H.
       MOV B,A      ; Store the first number in B.
       MOV M,A      ; Store for display.
       DCX H
       CALL KIND    ; Reading y.
       MOV C,A      ; Store the second number in C.
       MOV M,A      ; Store for display.

       CALL PRINT   ; Print the input in the leftmost 7-segment digits.

       CALL KIND
       CPI 0AH      ; Check if A was pressed.
       CZ ADD       ; If so, add the numbers.
       CPI 0FH      ; Check if F was pressed.
       CZ MUL       ; If so, multiply.
       JMP RD

PRINT:
       PUSH D       ; Push down D.
       LXI D,0900H  ; Store the data address before calling STDM.
       CALL STDM    ; Print.
       CALL DCD
       POP D        ; Restore D.
       RET

ADD:
       PUSH PSW     ; Push down A. Important for proper function in ADD,
       MOV A,B      ; MUL branching point.
       ADD C
       MOV B,A      ; Add the numbers and store the result in B.
       LXI H,0900H  ; Load the target memory before calling UPDATE.
       CALL UPDATE  ; Update the desired memory location pair.
       CALL PRINT
       POP PSW      ; Restore A.
       RET

MUL:
       PUSH PSW
       MVI E,00H    ; Initialize result accumulator.

LOPO:
       MOV A,B      ; (E)=(B)x(C), by adding (B) times the number (C) to 0.
       CPI 00H
       JZ DONE
       DCR B
       MOV A,E
       ADD C
       MOV E,A
       JMP LOPO

DONE:
       MOV A,E      ; Store result in (A) for updating memory data.
       MOV B,A      ; Update B, according to ACCUMULATE usage.
       LXI H,0900H
       CALL UPDATE
       CALL PRINT
       POP PSW
       RET

CLEAR:              ; A routine that clears the mod256 accumulator.
       PUSH PSW
       MVI D,00H    ; Clear the mod256 accumulator...
       MOV A,D      ; and store it in memory.
       LXI H,0902H
       CALL UPDATE
       CALL PRINT
```

```
      POP PSW
      RET

ACCUMULATE:        ; A routine for incrementing the mod256 accumulator.
      PUSH PSW     ; The mod256 value is stored in (D) and the previous
      MOV A,D      ; operation result is stored in (B).
      ADD B        ; Increment accumulator (D) by the calculated value (B)
      MOV D,A      ; And store its new value.
      MVI B,00H    ; Reset the operation result.
      LXI H,0902H
      CALL UPDATE  ; Then update the corresponding memory area and print.
      CALL PRINT
      POP PSW
      RET

UPDATE:            ; A helper routine for storing an 8-bit integer, stored
      PUSH B       ; in register (A), as two hex digits, in the memory
      MOV B,A      ; area specified in register pair (HL).
      ANI 0FH      ; First isolate 4 LSB bits...
      MOV M,A      ; and store them
      INX H
      MOV A,B
      ANI F0H      ; Then isolate 4 MSB bits...
      RRC          ; and store them.
      RRC
      RRC
      RRC
      MOV M,A
      POP B
      RET

END
```