

MICROPROCESSOR PROGRAMES

/* Adding 2 BCD nos without carry */

; For detail explanation of program refer page no. 135 of book

; Microprocessor And Microcomputer by A. P. Godse

```
MOV A, L    ; Get lower 2 digits of no. 1
ADD E      ; Add two lower digits
DAA       ; Adjust result to valid BCD
STA 2300H  ; Store partial result
MOV A, H    ; Get most significant 2 digits of no. 2
ADC D     ; Add two most significant digits
DAA       ; Adjust result to valid BCD
STA 2301H  ; Store partial result
HLT       ; Terminate program execution
```

/* Adding 2 Hexa-decimal nos without carry */

; For detail explanation of program refer page no. 145 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LDA 2200
MOV C, A    ; Initialize counter
SUB A      ; sum = 0
LXI H, 2201H; Initialize pointer
BACK: ADD M ; SUM = SUM + data
INX H     ; increment pointer
DCR C     ; Decrement counter
JNZ BACK  ; if counter 0 repeat
STA 2300H ; store sum
HLT
```

/* Adding 2 Hexa-Decimal nos with carry */

; For detail explanation of program refer page no. 145 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LDA 2200H
MOV C,A    ; Initialize counter
LXI H, 2201H; Initialize pointer
SUB A     ; Sum low = 0
MOV B,A   ; Sumhigh = 0
BACK: ADD M ; Sum = sum + data
JNC SKIP
INR B     ; Add carry to MSB of SUM
SKIP: INX H ; Increment pointer
DCR C     ; Decrement counter
JNZ BACK  ; Check if counter 0 repeat
STA 2300H ; Store lower byte
MOV A,B
STA 2301H ; Store higher byte
HLT       ; Terminate program execution
```

/*Block data transfer*/

; For detail explanation of program refer page no. 147 of book

; Microprocessor And Microcomputer by A. P. Godse

```
MVI C, 0AH ; Initialize counter
LXI H, 2200H ; Initialize source memory pointer
LXI D, 2300H ; Initialize destination memory pointer
BACK: MOV A, M ; Get byte from source memory block
STAX D ; Store byte in the destination memory block
INX H ; Increment source memory pointer
INX D ; Increment destination memory pointer
DCR C ; Decrement counter
JNZ BACK ; If counter 0 repeat
HLT ; Terminate program execution
```

/*Multiplication of two 8 bit nos*/

; For detail explanation of program refer page no. 149 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LDA 2200H
MOV E, A
MVI D, 00 ; Get the first number
LDA 2201H
MOV C, A ; Initialize counter
LXI H, 0000H; Result = 0
BACK: DAD D ; Result = result + first number
DCR C ; decrement count
JNZ BACK ; If count 0 repeat
SHLD 2300H ; Store result
HLT ; Terminate program execution
```

/*Division of two 8 bit nos*/

; For detail explanation of program refer page no. 150 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LHLD 2200H ; Get the dividend
LDA 2202H
MOV C, A ; Get the divisor
LXI D, 0000H; Quotient = 0
BACK: MOV A, L
SUB C
MOV L, A ; Save partial result
JNC SKIP ; if CY 1 jump
DCR H ; Subtract borrow of previous subtraction
SKIP: INX D ; Increment quotient
MOV A, H
CPI 00 ; Check if dividend
JNZ BACK ; < divisor
MOV A, L ; if no repeat
CMP C
JNC BACK
SHLD 2302H ; Store the remainder
XCHG
SHLD 2300H ; Store the quotient
HLT ; Terminate program execution
```

/*No of negative nos in the list of hexa decimal nos*/

; For detail explanation of program refer page no. 166 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LDA 2200H
MOV C,A      ; Initialize count
MVI B,00     ; Negative number = 0
LXI H,2201H  ; Initialize pointer
BACK:  MOV A,M      ; Get the number
ANI 80H      ; Check for MSB
JZ SKIP      ; If MSB = 1
INR B        ; Increment negative number count
SKIP: INX H    ; Increment pointer
DCR C        ; Decrement count
JNZ BACK     ; If count 0 repeat
MOV A,B
STA 2300H    ; Store the result
HLT         ; Terminate program execution
```

/*Maximum no in the list of hexa decimal nos*/

; For detail explanation of program refer page no. 167 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LDA 2200H
MOV C, A     ; Initialize counter
XRA A        ; Maximum = Minimum possible value = 0
LXI H, 2201H; Initialize pointer
BACK:  CMP M      ; Is number > maximum
JNC SKIP
MOV A, M     ; Yes, replace maximum
SKIP: INX H
DCR C
JNZ BACK
STA 2300H    ; Store maximum number
HLT         ; Terminate program execution
```

/*1's complement of a no*/

; For detail explanation of program refer page no. 168 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LDA 2200H    ; Get the number
CMA          ; Complement number
STA 2300H    ; Store the result
HLT          ; Terminate program execution
```

/*2's complement of a no*/

; For detail explanation of program refer page no.169 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LDA 2200 H   ; Get the number
CMA          ; Complement the number
ADI 01H      ; Add one in the number
STA 2300H    ; Store the result
HLT          ; Terminate program execution
```

/*Adding the 4 most significant bit to a hexa no*/

; For detail explanation of program refer page no. 169 of book
; Microprocessor And Microcomputer by A. P. Godse
LDA 2201H ; Get the Most significant BCD digit
RLC
RLC
RLC
RLC ; Adjust the position
MOV C, A ; store the partial result
LDA 2200H ; Get the lower BCD digit
ADD C ; Add lower BCD digit
STA 2300H ; Store the result
HLT ; Terminate program execution

/*Swapping the nos */

; For detail explanation of program refer page no. 117 of book
; Microprocessor And Microcomputer by A. P. Godse

LDA 2000H ; Get the contents of memory location 2000H into accumulator
MOV B,A ; save the contents in B register
LDA 2200H ; Get the contents of memory location 2200H into accumulator.
STA 2000H ; Store the contents of accumulator at address 2000H.
MOV A,B ; Get the saved contents back into A register
STA 2200H ; Store the contents of accumulator at address 2200H
HLT ; Terminate program execution

/*Swappin the hexa nos(much better program)*/

; For detail explanation of program refer page no. 117 of book
; Microprocessor And Microcomputer by A. P. Godse
LXI H,2000H ; Initialize HL register pair as a pointer
; to memory location 2000H
LXI D,2200 H ; Initialize DE register pair as a pointer
; to memory location 2200H
MOV B,M ; Get the contents of memory location
; 1000H into B register
LDAX D ; Get the contents of memory location
; 2000H into A register
MOV M,A ; Store the contents of A register into memory
; location 2000H
MOV A,B ; Copy the contents of B register into accumulator
STAX D ; Store the contents of A register into memory
; location 2200H.
HLT ; Terminate program execution

/*Separate the digits of a hexa decimal nos and store it in two diff. locations*/

; For detail explanation of program refer page no. 170 of book
; Microprocessor And Microcomputer by A. P. Godse
LDA 2200H ; Get the packed BCD number
ANI F0H ; Mask lower nibble
RRC
RRC
RRC
RRC ; Adjust higher BCD digit as a lower digit
STA 2300H ; Store the partial result
LDA 2200H ; Get the original BCD number
ANI 0FH ; Mask higher nibble
STA 2301H ; Store the result
HLT ; Terminate program execution

/*Sample program for subroutine*/

; For detail explanation of program refer page no. 122 of book
; Microprocessor by A. P. Godse
LXI SP, 2400H
LXI H, 2000H
LXI B, 1020H
CALL SUB
HLT

;Subroutine program :

SUB: PUSH B
PUSH H
LXI B, 4080H
LXI H, 4090H
DAD B
SHLD 2200H
POP H
POP B
RET

//**

; For detail explanation of program refer page no. 181 of book
; Microprocessor and Microcomputer by A. P. Godse
LXI SP,2400 ; Initialize stack pointer
MVI C, 08H ; Initialize count with 8
BACK: MOV A,B ;
RRC ; Rotate B register contents right
MOV B,A ; Save contents of register B
JNC SKIP ; If no carry skip
MVI A,0C0H
SIM ; If carry, send high on SOD
JMP NEXT
SKIP: MVI A,40H
SIM ; If no carry, send low on SOD
NEXT:CALL DELAY ; Wait for specific time
DCR C ; Decrement count by 1
JNZ BACK ; if count = 0, if not repeat
HLT ; Stop program execution

```

DELAY: LXI D,0001
BACK1: DCX D
      MOV A,E
      ORA D
      JNZ BACK1
      RET

```

/**/

; For detail explanation of program refer page no. 181 of book
; Microprocessor & Microcomputer by A. P. Godse

```

      LXI SP, 27FFH
      LXI H, 2000H      ; Memory pointer
      RIM              ; Read SID
      ANI 80H          ; Check D7 bit of Accumulator
      CALL DELAY       ; 1/2 bit time delay for stop bit
      MVI B, 08H       ; Initialize bit counter
      MVI D, 00H       ; Clear data register
UP1:  CALL DELAY       ; 1bit time
      RIM              ; Read SID line
      ANI 80 H         ; Mask bits B6-B0
      ORA D            ; OR data bit with previous bits
      RRC
      MOV D, A         ; Store data bit at appropriate position
      DCR B
      JNZ UP1
      RLC              ; Shift left to correct result
      MOV M, A         ; Store result
      RIM              ; Read stop bit
      ANI 80H
      CZ ERROR         ; If not stop bit call error
      HLT              ; Terminate program.

```

```

DELAY: MVI E,01
BACK1: DCR E
      JNZ BACK1
      RET
ERROR: MVI A,0FF
      RET

```

/*Addition of two hexa decimal nos including carry*/

; For detail explanation of program refer page no. 184 of book
; Microprocessor and Microcomputer by A. P. Godse

```

LXI H,2000H ; HL Points 2000H
MOV A,M     ; Get first operand
INX H       ; HL Points 2001H
ADD M       ; Add second operand
INX H       ; HL Points 2002H
MOV M,A     ; Store the lower byte of result at 2002H
MVI A,00    ; Initialize higher byte result with 00H
ADC A       ; Add carry in the high byte result
INX H       ; HL Points 2003H
MOV M,A     ; Store the higher byte of result at 2003H
HLT

```

/*Rotate right a value 4 times*/

; For detail explanation of program refer page no. 185 of book
; Microprocessor And Microcomputer by A. P. Godse
MOV A,C

RAR
RAR
RAR
RAR
MOV C,A
HLT

/*Rotate right and store the result*/

; For detail explanation of program refer page no. 185 of book
; Microprocessor And Microcomputer by A. P. Godse
MOV A,B

RAR
MOV B,A
MOV A,C
RAR
MOV C,A
HLT

/*Addition of two hexa decimal nos*/

; For detail explanation of program refer page no. 130 of book
; Microprocessor And Microcomputer by A. P. Godse

LXI H, 2000H ; HL points 2000H
MOV A,M ; Get first operand
INX H ; HL points 2001H
ADD M ; Add second operand
INX H ; HL points 2002H
MOV M,A ; store result at 2002H
HLT ; Terminate program execution

/*To find the no of 1's in the byte*/

; For detail explanation of program refer page no. 187 of book
; Microprocessor & Microcomputer by A. P. Godse

MVI B,00H
MVI C,08H
MOV A,D
BACK: RAR
JNC SKIP
INR B
SKIP: DCR C
JNZ BACK
HLT

/*In a set of nos add only the even nos*/

; For detail explanation of program refer page no. 187 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LDA 2200H
MOV C, A ; Initialize counter
MVI B, 00H ; sum = 0
LXI H, 2201H; Initialize pointer
BACK:    MOV A, M ; Get the number
ANI, 01H ; Mask Bit1 to Bit7
JNZ SKIP ; Don't add if number is ODD
MOV A, B ; Get the sum
ADD M ; SUM = SUM + data
MOV B, A ; Store result in B register
SKIP:   INX H ; increment pointer
DCR C ; Decrement counter
JNZ BACK ; if counter 0 repeat
STA 2210H ; store sum
HLT ; Terminate program execution
```

/*In a set of nos add only the odd nos*/

; For detail explanation of program refer page 190 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LDA 2200H
MOV C, A ; Initialize counter
LXI H, 2201H ; Initialize pointer
MVI E, 00 ; Sumlow = 0
MOV D, E ; Sumhigh = 0
BACK:    MOV A, M ; Get the number
ANI 01H ; Mask Bit1 to Bit7
JZ SKIP ; Do not add if number is even
MOV A, E ; Get the lower byte of sum
ADD M ; Sum = sum + data
MOV E, A ; Store result in E register
JNC SKIP
INR D ; Add carry to MSB of SUM
SKIP:   INX H ; Increment pointer
DCR C ; Decrement counter
JNZ BACK ; Check if counter 0 repeat
MOV A, E
STA 2300H ; Store lower byte
MOV A, D
STA 2301H ; Store higher byte
HLT ; Terminate program execution
```

/*store the nos in a memory loc. In the reverse order in another memory loc.*/

; For detail explanation of program refer page no. 191 of book

; Microprocessor And Microcomputer by A. P. Godse

```
MVI C, 0AH ; Initialize counter
LXI H, 2200H ; Initialize source memory pointer
LXI D, 2309H ; Initialize destination memory pointer
BACK:    MOV A, M ; Get byte from source memory block
STAX D ; Store byte in the destination memory block
```

INX H ; Increment source memory pointer
 DCX D ; Decrement destination memory pointer
 DCR C ; Decrement counter
 JNZ BACK ; If counter 0 repeat
 HLT ; Terminate program execution

/*Transfer 5 nos in memory pointed by loc. value in another memory loc. */

; For detail explanation of program refer page no. 192 of book

; Microprocessor And Microcomputer by A. P. Godse

LXI H,2200H ; Initialize lookup table pointer
 LXI D,2100H ; Initialize source memory pointer
 LXI B,2000H ; Initialize destination memory pointer

BACK: LDAX D ; Get the number
 MOV L,A ; A point to the square
 MOV A,M ; Get the square
 STAX B ; Store the result at destination memory location
 INX D ; Increment source memory pointer
 INX B ; Increment destination memory pointer
 MOV A,C ;
 CPI 05H ; Check for last number
 JNZ BACK ; If not repeat
 HLT ; End of program

/*Search for a no in the list if found store address where it is found else store 0000*/

; For detail explanation of program refer page no. 194 of book

; Microprocessor And Microcomputer by A. P. Godse

LXI H,2000H ; Initialize memory pointer
 MVI B, 52H ; Initialize counter

BACK: MOV A,M ; Get the number
 CMP C ; Compare with the given byte
 JZ LAST ; Go last if match occurs
 INX H ; Increment memory pointer
 DCR B ; Decrement counter
 JNZ BACK ; If not zero, repeat
 LXI H, 0000H
 SHLD 2200H
 JMP END ; Store 00 at 2200H and 2201H
 LAST: SHLD 2200H ; Store memory address
 END: HLT ; Stop

/*Complement the flag content*/

; For detail explanation of program refer page no. 195 of book

; Microprocessor And Microcomputer by A. P. Godse

LXI SP, 2700 ; Initialize Stack
 PUSH PSW ; Save flags on stack
 POP H ; Retrieve flags in 'L'
 MOV A,L ; Flags in accumulator
 CMA ; Complement accumulator
 MOV L,A ; Accumulator in 'L'
 PUSH H ; Save on stack
 POP PSW ; Back to flag register
 HLT ; Terminate program execution

/*Decimal addition of 2 nos in 6 different mem. Loc. And store in 6 other mem. loc.*/

; For detail explanation of program refer page no. 196 of book

; Microprocessor And Microcomputer by A. P. Godse

LXI H,2000H ; Initialize pointer1 to first number

LXI D,2100H ; Initialize pointer2 to second number

LXI B,2200H ; Initialize pointer3 to result

STC

CMC ; Carry = 0

BACK: LDAX D ; Get the digit

ADD M ; Add two digits

DAA ; Adjust for decimal

STAX B ; Store the result

INX H ; Increment pointer1

INX D ; Increment pointer2

INX B ; Increment result pointer

MOV A, L

CPI , 06 ; Check for last digit

JNZ BACK ; If not last digit repeat

HLT ; Terminate program execution

/*Subtraction of 2 nos in memory */

; For detail explanation of program refer page no. 130 of book

; Microprocessor And Microcomputer by A. P. Godse

LXI H, 2000H ; HL points 2000H

MOV A,M ; Get first operand

INX H ; HL points 2001H

SUB M ; Subtract second operand

INX H ; HL points 2002H

MOV M,A ; Store result at 2002H

HLT ; Terminate program execution

/*Bubble sort of 10 nos in memory*/

; For detail explanation of program refer page no. 196 of book

; Microprocessor & Microcomputer by A. P. Godse

MVI B, 09 ; Initialize counter 1

START: LXI H, 2200H ; Initialize memory pointer

MVI C, 09H ; Initialize counter 2

BACK: MOV A, M ; Get the number

INX H ; Increment memory pointer

CMP M ; Compare number with next number

JC SKIP ; If less, don't interchange

JZ SKIP ; If equal, don't interchange

MOV D, M

MOV M, A

DCX H

MOV M, D

INX H ; Interchange two numbers

SKIP: DCR C ; Decrement counter 2

JNZ BACK ; If not zero, repeat

DCR B ; Decrement counter 1

JNZ START ; If not zero, repeat

HLT ; Terminate program execution

/*Addition of 2 nos in 6 different mem. Loc. And store in 10 other mem. loc.*/

; For detail explanation of program refer page no. 198 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LXI H, 2200H      ; Initialize memory pointer 1
LXI B, 2300H      ; Initialize memory pointer 2
LXI D, 2400H      ; Initialize result pointer
BACK: LDAX B       ; Get the number from array 2
ADD M             ; Add it with number in array 1
STAX D           ; Store the addition in array 3
INX H            ; Increment pointer1
INX B            ; Increment pointer2
INX D            ; Increment result pointer
MOV A, L
CPI, 0A          ; Check pointer1 for last number
JNZ BACK        ; If not, repeat
HLT             ; Stop
```

/*In a list of 50 nos separate the even nos and store the in different memory loc.*/

; For detail explanation of program refer page no. 199 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LXI H, 2200H      ; Initialize memory pointer1
LXI D, 2300H      ; Initialize memory pointer2
MVI C, 32H       ; Initialize counter
BACK: MOV A, M     ; Get the number
ANI 01H          ; Check for even number
JNZ SKIP         ; If ODD, don't store
MOV A, M         ; Get the number
STAX D           ; Store the number in result list
INX D            ; Increment pointer 2
SKIP: INX H       ; Increment pointer1
DCR C            ; Decrement counter
JNZ BACK        ; If not zero, repeat
HLT             ; Stop
```

/*convert the set of nos to odd parity if the no is even parity*/

; For detail explanation of program refer page no. 199 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LXI H, 2040H
MOV C,M          ; Counter for character
REPEAT: INX H     ; Memory pointer to character
MOV A,M         ; Char. in accumulator
ORA A           ; ORing with itself to check parity.
JPO PAREVEN
ORI 80H         ; If odd parity place even parity in D7(80).
PAREVEN: MOV M,A  ; Store converted even parity character.
DCR C           ; Decrement counter.
JNZ REPEAT     ; If not zero go for next character.
HLT            ; Terminate program execution
```

/*Block data transfer of 15 nos*/

; For detail explanation of program refer page no. 202 of book

; Microprocessor And Microcomputer by A. P. Godse

```
MVI C, 0FFH      ; Initialize counter
LXI H, 20FFH     ; Initialize source memory pointer
LXI D, 214FH     ; Initialize destination memory pointer
BACK:  MOV A, M   ; Get byte from source memory block
      STAX D      ; Store byte in the destination memory block
      DCX H       ; Decrement source memory pointer
      DCX D       ; Decrement destination memory pointer
      DCR C       ; Decrement counter
      JNZ BACK    ; If counter # 0 repeat
      HLT        ; Stop execution
```

; For detail explanation of program refer page no. 202 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LXI SP, 2400H    ; Initialize stack pointer
MVI C, 90H       ; Initialize counter
BACK:  MOV A, C   ;
      ANI 0F      ; Mask higher nibble
      CPI 0F
      JNZ SKIP
      MOV A, C
      SUI 06      ; Subtract 6 to adjust decimal count
      MOV C, A
SKIP:  MOV A, C
      OUT 05      ; send count on output port
      CALL DELAY ; Wait for 0.5 seconds
      DCR C       ; decrement count
      JNZ BACK    ; If not zero, repeat
      HLT        ; Stop execution
DELAY: LXI D, 0001
BACK1: DCX D
      MOV A, E
      ORA D
      JNZ BACK1
      RET
```

/*Add '2' 16 bit no including a carry from D7 bit to D8 bit*/

; For detail explanation of program refer page no. 131 of book

; Microprocessor And Microcomputer by A. P. Godse

```
LHLD 2000H ; Get first 16-bit number in HL
XCHG      ; Save first 16-bit number in DE
LHLD 2002H ; Get second 16-bit number in HL
MOV A, E   ; Get lower byte of the first number
ADD L      ; Add lower byte of the second number
MOV L, A   ; Store result in L register
MOV A, D   ; Get higher byte of the first number
ADC H      ; Add higher byte of the second number with carry
MOV H, A   ; Store result in H register
SHLD 2004H ; Store 16-bit result in memory locations 2004H and 2005H.
HLT        ; Terminate program execution
```

/*Add '2' 16 bit no*/

; For detail explanation of program refer page no. 131 of book
; Microprocessor And Microcomputer by A. P. Godse
LHLD 2000H ; Get first 16-bit number
XCHG ; Save first 16-bit number in DE
LHLD 2002H ; Get second 16-bit number in HL
DAD D ; Add DE and HE
SHLD 2004H ; Store 16-bit result in memory locations 2004H and 2005H.
HLT ; Terminate program execution

/*Add '2' 16 bit no including a carry from D7 bit to D8 bit*/

; For detail explanation of program refer page no. 132 of book
; Microprocessor And Microcomputer by A. P. Godse
LHLD 2000H ; Get first 16-bit number in HL
XCHG ; Save first 16-bit number in DE
LHLD 2002H ; Get second 16-bit number in HL
MOV A,E ; Get lower byte of the first number
SUB L ; Subtract lower byte of the second number
MOV L,A ; Store the result in L register
MOV A,D ; Get higher byte of the first number
SBB H ; Subtract higher byte of second number with borrow
MOV H,A
SHLD 2004H ; Store 16-bit result in memory locations 2004H and 2005H.
HLT ; Terminate program execution

/*Decimal addition*/

; For detail explanation of program refer page no. 135 of book
; Microprocessor And Microcomputer by A. P. Godse
LXI H, 2200H ; Initialize pointer
MOV A,M ; Get the first number
INX H ; Increment the pointer
ADD M ; Add two numbers
DAA ; Convert HEX to valid BCD
STA 2300H ; Store the result
HLT ; Terminate program execution
