STUDY OF MONITOR ROUTINE

BY

SUBATHRA S

STUDY OF MONITOR ROUTINE

OBJECTIVE

To write an assembly language program to study of the monitor routines.

APPARATUS REQUIRED

- 8085 Microprocessor trainer kit
- Power supply

OUTPUT ROUTINE

DESCRIPTION

Starting address=05FC$_H$
Inputs:
(A)=displays flag;
0=use address field, 1=use data field.
(B)=dot flag;
0=no dot, 1=dot.
Destroys:
A, H, L registers & flags.

The 0utput routine is used in the keyboard mode to output the character of the display. Either 4 char or 2 char are output using the HL register as a pointer to the character. The output flag determines whether a dot appear with last character or not.The display table at 05FC$_H$ is used to translate the code to the character to be displayed.

ALGORITHM

1.The inputs are given for both address field as well as data field.

2.For address field accumulator is given as 00$_H$ and for data field, accumulator is given as 01$_H$.

3.The output routine is called using CALL statement.

4.The output is displayed in data as well as address field.

5.Stop the program execution.

ADDRESS FIELD ONLY

ADDRESS	LABEL	MNEMONICS	OPCODE/OPERAND	COMMENT
C100		**LXI H,C500$_H$**	21 00 C5	Initialize HL register pair
C103		**MVI A,00$_H$**	3E 00	Move immediately 00$_H$ in to accumulator
C105		**MVI B,00$_H$**	06 00	Move immediately 00$_H$ to B register
C107		**CALL OUTPUT**	CD FC 05	Call OUTPUT monitor routine

C10A		**HLT**	76	Stop the execution

EXECUTION

ADDRESS	INPUT DATA
C500$_H$	0D$_H$
C501$_H$	0C$_H$
C502$_H$	0B$_H$
C503$_H$	0A$_H$

DISPLAY

A	B	C	D		

DATA FIELD ONLY

ADDRESS	LABEL	MNEMONICS	OPCODE/OPERAND	COMMENT
C10B		**LXI H,C504$_H$**	21 04 C5	Initialize HL register pair
C10E		**MVI A,01$_H$**	3E 01	Move immediately 01$_H$ in to accumulator
C110		**MVI B,00$_H$**	06 00	Move immediately 00$_H$ to B register
C112		**CALL OUTPUT**	CD FC 05	Call OUTPUT monitor routine
C115		**HLT**	76	Stop the execution

EXECUTION

ADDRESS	INPUT DATA
C504$_H$	05$_H$
C505$_H$	05$_H$

DISPLAY

				5	5

ADDRESS AND DATA FIELD

ADDRESS	LABEL	MNEMONICS	OPCODE/OPERAND	COMMENT
C116		**LXI H,C500$_H$**	21 00 C5	Initialize HL register pair
C119		**MVI A,00$_H$**	3E 00	Move immediately 00 in to accumulator
C11B		**MVI B,00$_H$**	06 00	Move immediately 00 to B register
C11D		**CALL OUTPUT**	CD FC 05	Call OUTPUT monitor routine
C120		**LXI H,C504$_H$**	21 04 C5	Initialize HL register pair
C123		**MVI A,01$_H$**	3E 01	Move immediately 00 in to accumulator
C125		**MVI B,00$_H$**	06 00	Move immediately 00 to B register
C127		**CALL OUTPUT**	CD FC 05	Call OUTPUT monitor routine
C12A		**HLT**	76	Stop the execution

EXECUTION

ADDRESS	INPUT DATA
C500$_H$	0D$_H$
C501$_H$	0C$_H$
C502$_H$	0B$_H$
C503$_H$	0A$_H$
C504$_H$	05$_H$
C505$_H$	05$_H$

DISPLAY

A	B	C	D	5	5

Addition Using UPDAD

DESCRIPTION

Starting address=06BC$_H$

Inputs:

FFF7$_H$, FFF8$_H$=address to be displayed

(B)=dot flag,0=no dot,1=dot.

Destroys:

A, B, C, D, E, H & L flags.

UPDAD is used in the keyboard and serial modes to update the address field display using the current address stored at location FFF7$_H$ and FFF8$_H$.

ALGORITHM

1.The two numbers are input for addition.
2.Add the numbers and display in LSB of address field.
3.The carry if present is displayed in MSB of address field.
4.This allotment of sum and carry is done using UPDAD.
5.stop the program execution.

ADDITION USING UPDAD

ADDRESS	LABEL	MNEMONICS	OPCODE/OPERAND	COMMENT
C150		**LXI H,C200$_H$**	21 00 C2	Initialize HL register pair
C153		**MVI A,M**	7E	Move memory content to accumulator
C154		**INX H**	23	Increment the register pair by 1
C155		**ADD M**	86	Add the memory content with accumulator content
C156		**STA FFF7$_H$**	32 F7 FF	Store the accumulator content at FFF7
C159		**MVI A,00$_H$**	3E 00	Move immediately 00 in to accumulator
C15B		**JNC L1**	D2 5F C1	Jump if no carry to L1
C15E		**INR A**	3C	Increment the register content by 1
C15F	**L1**	**STA FFF8$_H$**	32 F8 FF	Store the accumulator content at FFF8
C162		**CALL UPDAD**	CD BC 06	Call UPDAD monitor routine
C165		**HLT**	76	Stop the execution

EXECUTION

ADDRESS	INPUT DATA	OUTPUT DATA
C200$_H$	FF$_H$	
C201$_H$	FF$_H$	
FFF7$_H$		FE$_H$
FFF8$_H$		01$_H$

DISPLAY

0	1	F	E		

Counting number Of Zeros Using UPDDT

DESCRIPTION

Starting address=06D3$_H$

Inputs:

FFF9$_H$=data to be displayed

(B)=dot flag,0=no dot,1=dot.

Destroys:

A, B, C, D, E, H, L registers & flags

UPDDT is used in the keyboard and serial modes to update data field display using the current data at location FFF9$_H$.

ALGORITHM

1.Enter the total number of inputs.
2.Accumulator is made 0 and the number is added with it.
3.If the sum is zero, increment the counter register.
4.Else go to the next number and proceed from step 2.
5.Store the content of count register after scanning all inputs.
6.Output the number of zero in the data field.
7.Stop the program execution.

COUNTING NUMBER OF ZEROS USING UPDDT

ADDRESS	LABEL	MNEMONICS	OPCODE/OPERAND	COMMENT
C130		**LXI H,C300$_H$**	21 00 C3	Initialize HL register pair
C133		**MOV C,M**	4E	Move memory content to C register
C134		**MVI B,00$_H$**	06 00	Move immediately 00$_H$ to B register
C136	**REPEAT**	**XRA A**	AF	Exclusive OR the accumulator content
C137		**INX H**	23	Increment the register content by 1
C138		**ADD M**	86	Add the memory content with accumulator
C139		**JNZ COUNT**	C2 3D C1	Jump if not zero to COUNT

C13C		**INR B**	04	Increment the register pair by 1
C13D	**COUNT**	**DCR C**	0D	Decrement the register pair by 1
C13E		**JNZ REPEAT**	C2 36 C1	Jump if not zero to REPEAT
C141		**MOV A,B**	78	Move B register content to accumulator
C142		**STA FFF9$_H$**	32 F9 FF	Store accumulator content at FFF9$_H$
C145		**CALL UPDDT**	CD D3 06	Call UPDDT monitor routine
C148		**HLT**	76	Stop the execution

EXECUTION

ADDRESS	INPUT DATA	OUTPUT DATA
C300$_H$	05$_H$	
C301$_H$	02$_H$	
C302$_H$	00$_H$	
C303$_H$	00$_H$	
C304$_H$	01$_H$	
C305$_H$	00$_H$	
FFF9$_H$		03$_H$

DISPLAY

E				0	3

REFERENCE

1. Ramesh S.Gaonkar, Microprocessor Architecture, Programming, and Applications, Fourth Edition, Penram International Publishing (India), 2000.

2. S.Subathra, “Advanced Microprocessor Laboratory”, Record work, Adhiparashakthi Engineering College, Melmaruvathur, October 2002

3. S.Subathra, “Programming in 8085 Microprocessor and its applications – An Innovative Analysis”, Technical Report, Adhiparashakthi Engineering College, Melmaruvathur, March 2003

4. Micro-85 EB, User Manual, Version – 3.0, CAT #M85 EB-002, VI Microsystems Pvt. Ltd., Chennai.