Stack and Subroutine

Lecture objectives: at the end of this lecture the student will able to:

1- Define the stack memory.

2- Study the operation of PUSH and POP instructions.

3- Define the subroutine.

4- Write the subroutine.

1- Stack Memory

1-1 Definition of Stack Memory

Stack is part from R/W memory used with subroutine or with **PUSH** and **POP** instructions to temporary storage of data.

1-2Addressing of Stack Memory:

Stack memory is addressed by using stack pointer where the contents of stack pointer is represent the address of first memory location of stack. **LXI SP operand(*operand is 16-bit number*)** instruction will be used to determine the beginning of stack memory. Stack memory is addressing with standard of **FILO** (First In Last Out) or **LIFO** (Last IN First Out). The data will began saving in stack in memory location which has address little by one of 16-bit number that loaded in stack pointer.

1-3 PUSH and **POP** instructions: two instructions are used to data storage in stack memory which begin addressing by stack pointer.

1-3-1 PUSH operand

Transferring the contents of operand to the stack memory which beginning by contents of stack pointer. Operand is one of register pair (**BC, DE, HL, AF**).

1-3-1-a- **PUSH B** transferring the (BC) to stack memory. (**one byte instruction**)

1-3-1-b- **PUSH D** transferring the (DE) to stack memory. (**one byte instruction**)

1-3-1-c- **PUSH H** transferring the (HL) to stack memory. (**one byte instruction**)

1-3-1-d-**PUSH PSW** transferring the (AF) to stack memory. (**one byte instruction**)

Note: PSW is stand for(Program Status Word) and AF is the accumulator and flag register

1-3-2 POP operand

Transferring the contents of stack memory locations which beginning by contents of stack pointer to operand. Operand is one of register pair (**BC, DE, HL, AF**).

1-3-2-a- **POP B** transferring the contents of latest two stack memory locations to BC.

1-3-2-b- **POP D** transferring the contents of latest two stack memory locations to DE.

1-3-2-c- **POP H** transferring the contents of latest two stack memory locations to HL.

1-3-2-d-**POP PSW** transferring the contents of latest two stack memory locations to AF.

Example: write the contents of HL register and (SP) after execute the following instructions

```
LXI SP,209A
LXI H,423F
PUSH H
HLT
```

Solution:

First (**SP**)=209A (H)=42 (L)=3F

There for (SP)=2098 (H)=42 L=3F

As shown in figure (3-1)

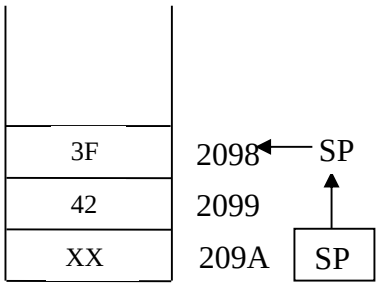

Figure 3-1

Example: write the contents of AF register and (SP) after execute the following instructions

```
LXI SP,3000
LXI D,200F
PUSH D
POP PSW
HLT
```

Solution:

First the (DE) transferred to stack memory locations (2FFF, 2FFE) by PUSH D instruction as shown in figure below (3-2):

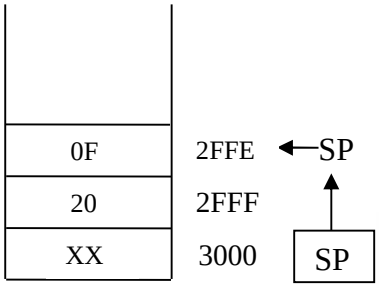

Figure 3-2

Then by POP PSW instruction, the content of memory location 2FFE transferring to flag register and contents of 2FFF to accumulator that mean

(SP)= 3000 (A)= 20 (F)=0F

See figure (3-3)

Figure 3-3

2- Subroutine

2-1- Subroutine definition: A subroutine is a group of instructions written separately from the main program to perform a function that occurs repeatedly in the main program.

A large software project is usually divided into sub tasks called modules. These modules are developed independently as subroutines by different programmers. Each programmer can use all the microprocessor registers to write a subroutine without affect other parts of the program.

The subroutine can be calling from main program by using Calling instructions (condition or unconditional calling according to application). The RET instruction used to return to main program (there are condition and unconditional return) **see lecture two**.

Some operations occur when microprocessor execute the subroutine are shown in points below:

1- contents of program counter are save in stack memory, where (PC) represent the address of memory location that loaded with code of instruction which follow the CALL instruction.

2- The operand of calling instruction will loaded in program counter, this means the execution sequence will transferred to subroutine.

3- When return instruction will used at end of subroutine, microprocessor will loaded in program counter the latest 16-bit number loaded in memory locations of stack.

Example: explain the steps of execution the following program.

Address of memory locations	Instruction	operand
1000	**LXI SP**	2300
	Some of Ins.	
100E	**MVI B**	34
1010	**CALL**	2070
1013	**.MOV A,C**	-------
	Some of Ins.	
1030	**HLT**	------
2070	**First ins. in subroutine**	
	Some of Ins.	-------
2070	**RET**	-------

Solution:

as shown in table in example the program storage in memory with starting address (1000H).

therefore with start execution

PC=1000

SP=2300

When execution of program arrive to CALL instruction, microprocessor addition two to PC that means the (PC) equal to (1013H) where the address (1013H) will saved in stack memory that have addresses (22FF 22FE). Then loading the operand (2070) of CALL instruction in PC to start the execution from first instruction in subroutine. At end of the subroutine (when execute the RET instruction), the microprocessor will loading the data in two locations from top of stack memory that means return to main program (exactly to MOV A,C instruction).