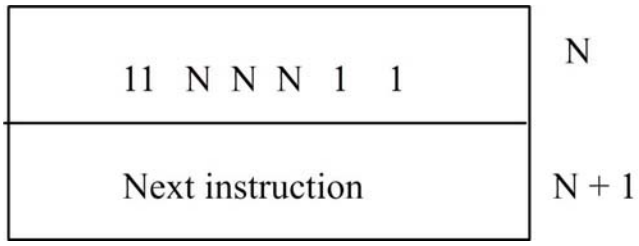
Lecture-26

6. RST n:

This is a single byte unconditional subroutine call instruction. The address of the subroutine is fixed depending upon the decimal no. n in the instruction the decimal no. n can change from 0 to 7. The comparing binary no$(NNN)_2$ is available as D_5 D_4D_3 bit of the operation code.

N N N	n	RST-n
0 0 0	0	RST-0
0 0 1	1	
1 1 1	7	RST-7

The single byte operation code is



The subroutine starting address is calibrated as follows;

Multiple the decimal no, by 8 convert it to the corresponding 2 digit hexadecimal no. Y_1,Y_2.

$(8\times n)_D \rightarrow (Y_1Y_0)_4$

Append $(00)_4$ for the two MSB Hex digit, $(Y_3Y_2)_4$ so that a 16-bit address

$Y_3Y_2Y_1Y_0$ 4 → 0 0 Y_1Y_0 4 is obtained e.g. the subroutine starting address for

RST 5 is $(8\times 5)_D = <_{10}$ D = 2 8 4 → 0 0 2 8 4

The macro RTL implemented is,

M [(sp)-1]← (PCH)
M [(sp)-2]← (PCL)
(sp) ← (sp)-2
(pc) ← $(8\times n)$

Where (PCH, PCL) is the return address.[this has 8 variations depending upon the NNN changing from 000 to 111. The addressing mode is register indirect, no flags affected.]

The high order 8 bits of the next instruction address are moved to the memory location whose address is one less than the content of register SP. The low order 8 bit of the next instruction address are moved to the memory location whose the address is two less than the content of register SP. The content of register sp is decremented by 2 control is transferred to the instruction whose address is eight times the content of NNN.

The micro RTL flow is,

MC - 1

OFMC	T_1 : AD_7- AD_0 ← (PCL)	, A_{15} – A_8← (PCH) ,ALE =
IO/$\overline{M}$ = 0	T_2 : $\overline{RD}$ = 0,	(PC) ← (PC)+1, AD_7- AD_0 ← M(AB)
S_1 = 1	T_3 : $\overline{RD}$ = 1, ↑.	(IR) ← (AD_7- AD_0)
S_0 = 1	T_4 : RST n = 1	
	$\begin{cases} T_5 \\ T_6 \end{cases}$:	(sp) ← (sp)-1 (address is calculated)

MC - 2

MRMC	T_1 : AD_7- AD_0 ← (SPL)	, A_{15} – A_8← (SPH) ,ALE =
IO/$\overline{M}$ = 0	T_2 : $\overline{WR}$ = 0,	(SP) ← (SP)+1, AD_7- AD_0 ← (PCH)
S_1 = 1	T_3 : $\overline{WR}$ = 1, ↑.	M(A_6) ← (AD_7- AD_0),(PCH) ←(00)
S_0 = 1		

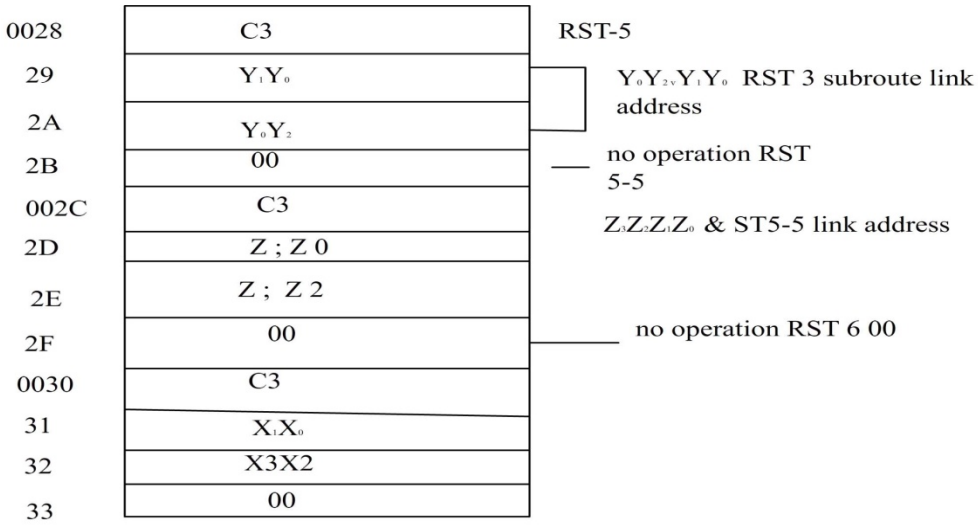
MC - 3

MRMC	T_1 : AD_7- AD_0 ← (SPL)	, A_{15} – A_8← (SPH) ,ALE =
IO/$\overline{M}$ = 0	T_2 : $\overline{WR}$ = 0,	AD_7- AD_0← (PCL)

$S_1 = 1$ $T_3 : \overline{WR} = 1, \uparrow$. $M(AB) \leftarrow (AD_7 - AD_0), (PCL) \leftarrow (8 \times N)_4$

Thus 12 states are required for the execution of this instruction. One important point while writing a subroutine program starting from the fixed location as calculated.

Consider for example RST 5 & D 576.the corresponding starting address of the subroutine are 0028 for RSTS, & 00304 for RST 6.recollect that there is RST 5-5 interrupt control signal input at pin no 9 of the μ p this also corresponding to an interrupt service routine starting from 00204. These addresses are shown in fig.



The fig shows that there are only four bytes, possible for each subroutine link and it is difficult to write subroutine here therefore, first byte will be C3 (jump Instruction) and 2nd & 3rd bytes will be the address of same memory locations which is the starting address of subroutine, this is known as subroutine link address 4th byte is no operation.

8) PCHL: this is a single byte instruction the operation code is,

11 101 001 $N = E\ 9_4$

There is no variation in this instruction. The macro RTL implemented is,

(PCL) ← (L)
(PCH) ← (H)
} (PC) ← (H,L)

The content of register (H) is merged to the (PCH) and content of register (L)is merged to (PCL).the meaning of the instructions is jump to (H) (L) location. Register indirect jump instruction. This is the only instruction

available in 8085 which allows the use to obtain a jump address from a register. This instruction uses register addressing mode other jump instruction use immediate reg indirect addressing mode.

This instruction is very useful in implanting select structure of the software program. The means RTL flow is,

OFMC T_1 : AD_7- $AD_0 \leftarrow$ (PCL) , $A_{15} - A_8 \leftarrow$ (PCH) ,ALE =

IO/$\overline{M}$ = 1 T_2 : $\overline{RD}$ = 0, (PC) $\leftarrow$ (PC)+1, AD_7- $AD_0 \leftarrow$ M(AB)

S_1 = 1 T_3 : $\overline{RD}$ = 1, ↑ (IR) $\leftarrow$ (AD_7- AD_0)

S_0 = 1 T_4 : PCHL = 1

$\begin{cases} T_5 \\ T_6 \end{cases}$: (pcl) $\leftarrow$ (l) (pc+1) $\leftarrow$(H)

Thus 6 states are required no flags affected.

This group of instructions manipulates the stack. Unless otherwise specified, condition flags are not affected by any instruction in there group.2f is the user responsibility to define stack area and to initialize the SP with the bottom address before these instructions are used.

1. The meaning of the instruction is push (save) the contents of register pair rp on top of the stack. The macro RTL is,

M [(sp)-1]$\leftarrow$ (rph)

M [(sp)-2]$\leftarrow$ (rpl)

(sp) $\leftarrow$ (sp)-2

The content of the higher order register of register pair rp is moved to the memory location whose address is one less than the content of register SP. The content of the low-order register of register pair is moved to the memory location whose address is two less than the content of register SP. the content of the register is decremented by 2 the operation code is,

11 RP 10 01	N

This is a single byte instruction has 3 variations.

RP = 00 ---- (B, C)

= 01 ---- (D, E)

10 ---- (H, L)

SP is not allowed, the addressing mode is register indirect. The macro RTL flow is,

OFMC T_1 : AD_7- $AD_0 \leftarrow$ (PCL) , $A_{15} - A_8 \leftarrow$ (PCH) , ALE =

IO/$\overline{M}$ = 1 T_2 : $\overline{RD}$ = 0, (PC) $\leftarrow$ (PC)+1, AD_7- $AD_0 \leftarrow$ M(AB)

S_1 = 1 T_3 : $\overline{RD}$ = 1, ↑. (IR) $\leftarrow$ (AD_7- AD_0)

S_0 = 1 T_4 : PUSH r/d = 1

$\{T_5, T_6$: (sp) $\leftarrow$ (sp) -1

MC - 2

MRMC T_1 : AD_7- $AD_0 \leftarrow$ (SPL) , $A_{15} - A_8 \leftarrow$ (SPH), ALE =

IO/$\overline{M}$ = 0 T_2 : $\overline{WR}$ = 0, (SP) $\leftarrow$ (SP)-1, AD_7- $AD_0 \leftarrow$ (rph)

S_1 = 1 T_3 : $\overline{WR}$ = 1, ↑. M(AB) $\leftarrow$ (AD_7- AD_0)

S_0 = 1

MC - 3

MRMC T_1 : AD_7- $AD_0 \leftarrow$ (SPL) , $A_{15} - A_8 \leftarrow$ (SPH) , ALE =

IO/$\overline{M}$ = 0 T_2 : $\overline{WR}$ = 0, AD_7- $AD_0 \leftarrow$ (rpl)

S_1 = 1 T_3 : $\overline{WR}$ = 1, ↑. M(AB) $\leftarrow$ (AD_7- AD_0)

Thus it required 12 states.

2. <u>PUSH PSW</u> (push processor status word):

The meaning of the instruction is save (push) the processor status word on the top of the stack. The accumulator & flag register together form a 16- bit word known as the processor status word (psw), ACC will occupy the higher order 8 bits and flag register will occupy the lower order 8-bits in PSW.

The operation code is

11 11 01 01	N = $(F5)_H$

The macro RTL implemented is

M (sp)-1 $\leftarrow$ (A)

M (sp)-2 $\leftarrow$ (PR)

(sp) ← (sp)-2

The content of the register A is moved to the memory location whose address is one less than the register SP. The contents of flag register are moved to the memory location whose address is two less than the content of register SP. The content of the register SP is decremented by 2.2f requires 3-M/C cycles and 12 states, the macro RTL flow is same as push rp the addressing mode is register indirect.

3. The meaning of the instruction is pop (load) the content form the top of the stack into register pair rp. The operation code is,

The macro RTL implemented is

(11 RP 0 0 01

(rph) ← M(SP+1)

(sp) ← (sp)+2

The content of the memory location points by the content of register SP is moved to the low order register of register pair rp. He content of the memory location whose address is one more than the content of register SP is moved to high order register of register pair. The content of register SP is not included in this. It has 3 variations none of the flags affected the micro RTL flow is,

MC-1

OFMC T_1 : AD_7- AD_0 ← (PCL) , A_{15} – A_8← (PCH) ,ALE = ⎍

IO/$\overline{M}$ = 1 T_2 : $\overline{RD}$ = 0, (PC) ← (PC)+1, AD_7- AD_0 ← M(AB)

S_1 = 1 T_3 : $\overline{RD}$ = 1, ↑ (IR) ← (AD_7- AD_0)

S_0 = 1 T_4 : POP rp = 1

MC - 2

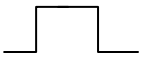
MRMC T_1 : AD_7- AD_0 ← (SPL) , A_{15} – A_8← (SPH) ,ALE = ⎍

IO/$\overline{M}$ = 0 T_2 : $\overline{RD}$ = 0, (SP) ← (SP)+1, AD_7- AD_0 ← M(AB)

$S_1 = 1$ $T_3 : \overline{RD} = 1, \uparrow$ (rpl) ← M(AB)

$S_0 = 1$

MC - 3

MRMC $T_1 : AD_7\text{-}AD_0 \leftarrow (SPL)$, $A_{15} - A_8 \leftarrow (SPH)$,ALE = 

IO/$\overline{M}$ = 0 $T_2 : \overline{RD} = 0$, (SP) ← (SP)+1, $AD_7\text{-}AD_0 \leftarrow$ M(AB)

$S_1 = 1$ $T_3 : \overline{RD} = 1, \uparrow$ (rpH) ← M(AB)

$S_0 = 1$

It requires 3 m/c cycles and 10 states. The addressing mode is register indirect addressing mode.

4) POP PSW: The meaning of the instruction is load the processor vector word register from the top of the stack. The operation code format is,

1111
0001

$= F1_H$

It has no variation. The macro RTL implemented is,

(FR) ← M (SP)

(A) ← M (SP+1)

(SP) ← (SP) +2

The content of the memory location pointed by SP is moved to the flag register (i.e. to restore various condition flags). The content of the memory location whose address is one move than the content of SP is moved to register A. the content of SP is incremented by 2.

The micro RTL flow is same as Popup. It also requires 3 m/c cycles and 10 states the addressing mode is register indirect. Flags affected are Z, S, AC, P, and CY.

5) The meaning of the instruction is exchange the top of the stack with the content of (H, L) register pair. The macro RTL implemented is,

M (SP) ⟷ (L)

M (SP+1) ⟷ (H)

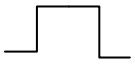
The content of register L is exchanged with the content of the memory location whose address is specified by the content of SP. The content of register H is exchanged with content the memory location whose address is one more than the content of SP.

The operation code format is

11 100 011	$=E3_H$

It has no variations. It is a single byte instruction. The addressing mode is register indirect addressing mode none flags affected. The micro RTL flow is,

MC-1

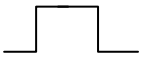
OFMC T_1 : AD_7- $AD_0 \leftarrow$ (PCL) , $A_{15} - A_8 \leftarrow$ (PCH) ,ALE = 

IO/$\overline{M}$ = 1 T_2 : $\overline{RD}$ = 0, (PC) $\leftarrow$ (PC)+1, AD_7- $AD_0 \leftarrow$ M(AB)

S_1 = 1 T_3 : $\overline{RD}$ = 1, ↑. (IR) $\leftarrow$ (AD_7- AD_0)

S_0 = 1 T_4 : XTHL = 1

MC - 2

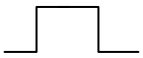
MRMC T_1 : AD_7- $AD_0 \leftarrow$ (SPL) , $A_{15} - A_8 \leftarrow$ (SPH), ALE = 

IO/$\overline{M}$ = 0 T_2 : $\overline{RD}$ = 0, (SP) $\leftarrow$ (SP)+1, AD_7- $AD_0 \leftarrow$ M(AB)

S_1 = 1 T_3 : $\overline{RD}$ = 1, ↑. (Z) $\leftarrow$ (AD_7-AD_0)

S_0 = 1

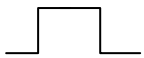
MC - 3

MWRMC T_1 : AD_7- $AD_0 \leftarrow$ (SPL) , $A_{15} - A_8 \leftarrow$ (SPH), ALE = 

IO/$\overline{M}$ = 0 T_2 : $\overline{WR}$ = 0, AD_7- $AD_0 \leftarrow$ (L), (SP) $\leftarrow$ (SP)+1

S_1 = 1 T_3 : $\overline{WR}$ = 1, ↑. M(AB) $\leftarrow$ (AD_7-AD_0),(L) $\leftarrow$ (Z)

MC - 4

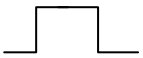
MRMC T_1 : AD_7- $AD_0 \leftarrow$ (SPL) , $A_{15} - A_8 \leftarrow$ (SPH), ALE = 

IO/$\overline{M}$ = 0 T_2 : $\overline{RD}$ = 0, AD_7- $AD_0 \leftarrow$ M(AB)

S_1 = 1 T_3 : $\overline{RD}$ = 1, ↑. (Z) $\leftarrow$ (AD_7-AD_0)

S_0 = 0 or W

MC - 5

MRMC T_1 : AD_7- AD_0 ← (SPL) , A_{15} – A_8← (SPH), ALE = 

IO/$\overline{M}$ = 0 T_2 : $\overline{WR}$ = 0, (SP) ← (SP)-1, AD_7- AD_0 ← (H)

S_1 = 0 T_3 : $\overline{WR}$ = 1, ↑ . M(AB) ← (AD_7-AD_0),(H) ← (Z),(SP) ← (SP)-1

S_0 = 1

Thus it requires 16 states and 5 m/c cycles flags affection.

6. SPHL: The meaning of the instruction is the contents of register H & L are moved to register SP the macro RTL implements is,

(SPH) ← (L)

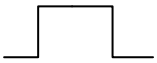
(SPH) ← (H)

The operation code format is

11 111 001	=$E9_H$

It has no variations the addressing mode is register addressing mode flags affected nnone.tha micro RTL flow is,

MC-1

OFMC T_1 : AD_7- AD_0 ← (PCL) , A_{15} – A_8← (PCH), ALE = 

IO/$\overline{M}$ = 1 T_2 : $\overline{RD}$ = 0, (PC) ← (PC)+1, AD_7- AD_0 ← M(AB)

S_1 = 1 T_3 : $\overline{RD}$ = 1, ↑ . (IR) ← (AD_7- AD_0)

S_0 = 1 T_4 : SPHL = 1

T_5 : (SPL) ← (L) (SPH) ← (H)

T_5 : x

2f requires one m/c cycle and 6 states.