

Práticas em Linguagem Assembly

Exemplo de Instruções de movimentação de dados

.org 500 ; Todas as instruções são introduzidas após a diretiva .org (endereço de origem)

mvi H, 10h ;carrega o registrador H com o valor 10h

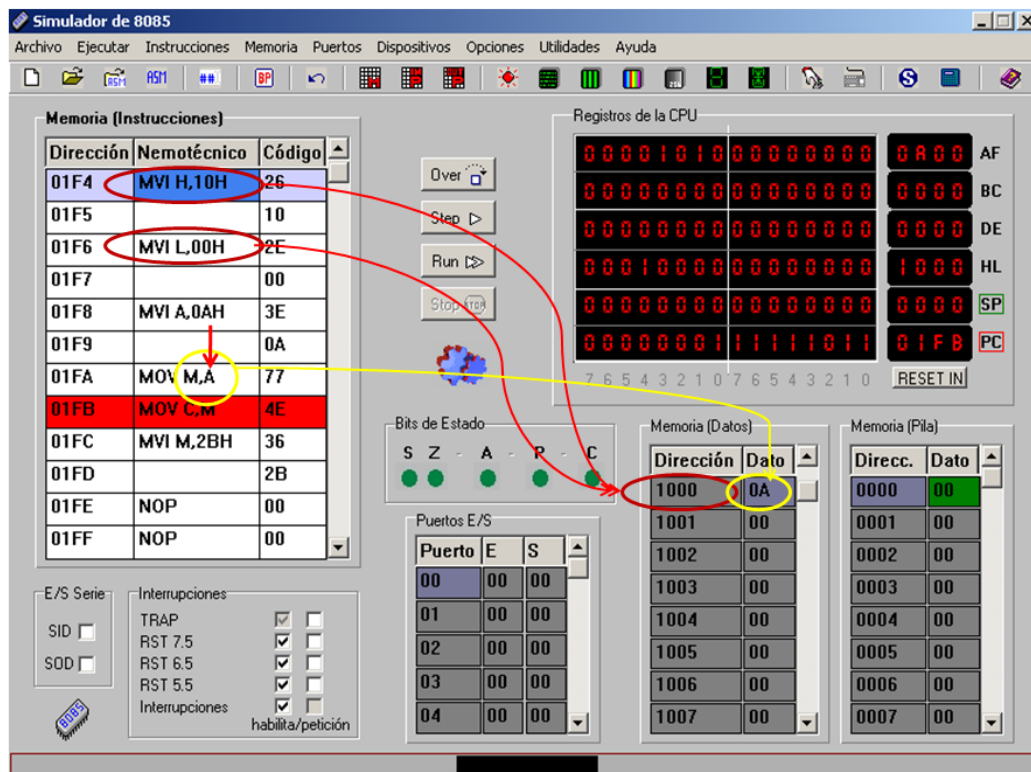
mvi L, 00h ;carrega registrador L com valor 00h

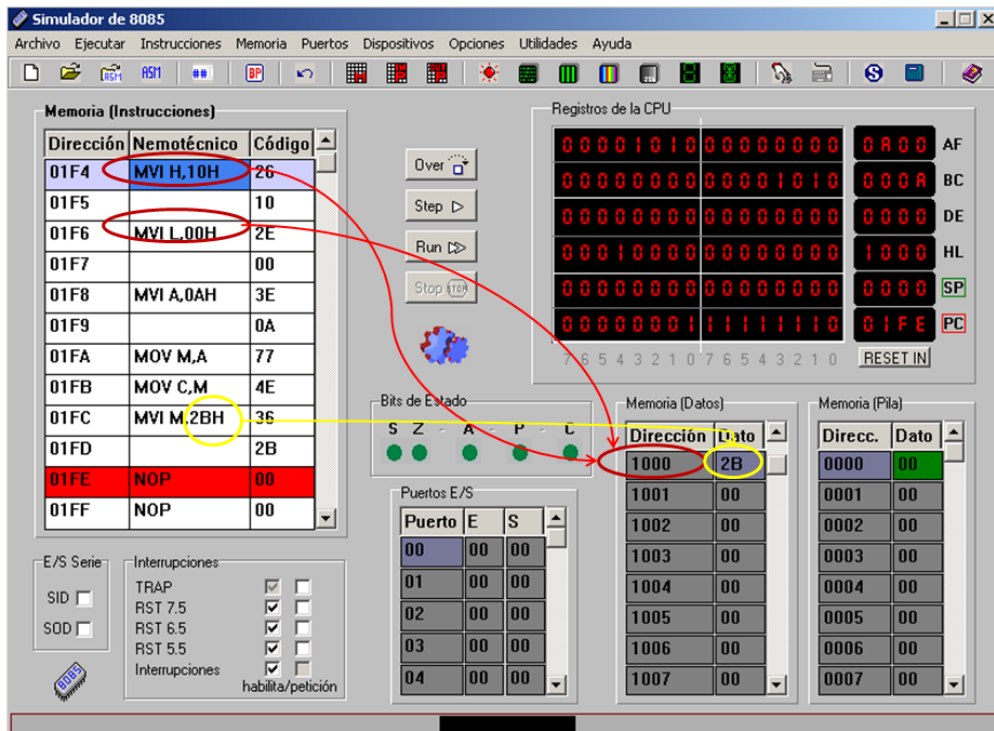
mvi A, 0Ah ;carrega acumulador com valor 0Ah

mov M,A ; move conteúdo de A pra posição de memória 1000h é uma instrução do tipo MOV M, r.
; Isso quer dizer que o conteúdo do registrador r é transferido para o endereço de
; memória indicado pelos registradores H-L. Na linha 2 temos mvi H, 10h, ou seja, foi
; carregado o valor 10h em H e na linha 3 mvi L,00h. Assim podemos dizer que o endereço
; de memória onde o acumulador irá armazenar 0Ah será 1000h, pois, M = HL.
; observe a figura abaixo.

mov C,M ;move conteúdo da posição de memória 1000h para o registrador c. c=0Ah

mvi M,2Bh ; coloca valor 2Bh na posição de memória 1000h.





.org 1000

lxi B, 1000h ;carrega registrador duplo BC com valor 1000h

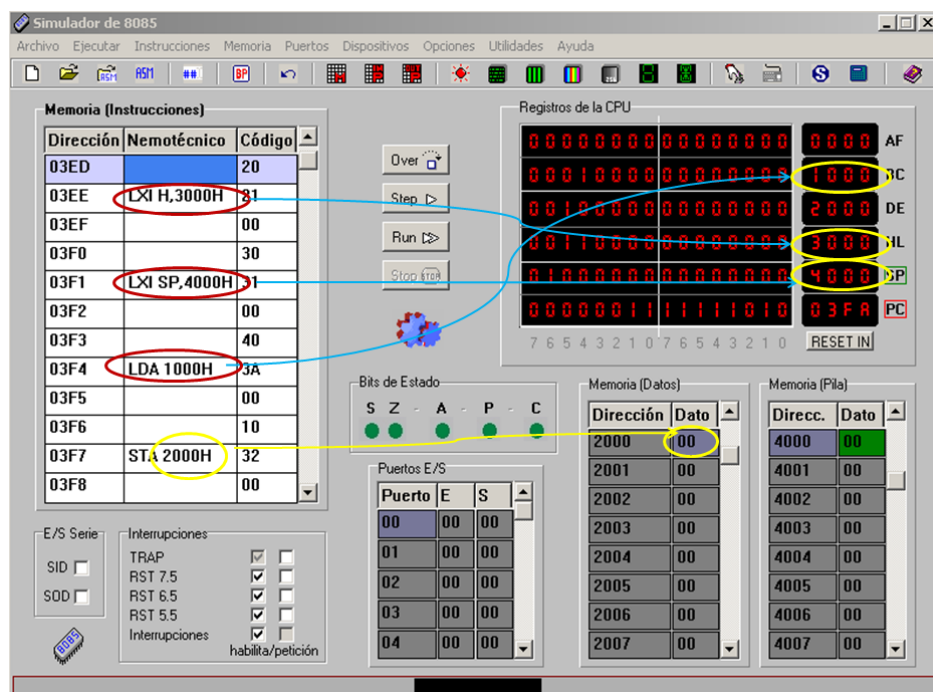
lxi D, 2000h ;carrega registrador duplo DE com valor 2000h

lxi H, 3000h ;carrega registrador duplo HL com valor 3000h

lxi SP, 4000h ;carrega registrador duplo SP (apontador da pilha) com valor 4000h

lda 1000h ; carrega acumulador com valor armazenado na posição de memória 1000h

sta 2000h ;move conteúdo do acumulador para posição de memória 2000h



Nota: O endereçamento por ponteiro de pilha (stack pointer) permite que o programador adicione ou retire um dado de 16 bits da lista de dados. A lista é designada por pilha (stack) e reside na RAM. São possíveis duas operações na pilha, a operação push e a operação pop. Numa operação push, o conteúdo de 16 bits de um par de registradores é transferido para a pilha, por transferência de 8 bits para a posição de memória indicada pelo registrador SP -1 e os restantes 8 bits para a posição de memória indicada pelo registrador SP. O conteúdo do registrador SP é então atualizado por o seu valor inicial menos 2, preparando-o assim para o armazenamento de novo dados de 16 bits.

Exemplo de instruções aritméticas

.org 100h

mvi A, 05h ;carrega o acumulador com o valor 05h

mvi C, 02h ;carrega o registrador C com o valor 02h

add C ;adiciona o conteúdo de C ao conteúdo do acumulador. $A = 05h + 02h = 07h$

adi 10h ;adiciona 10h ao conteúdo de A. $A = 07h + 10h = 17h$

adc C ; adiciona o conteúdo de C ao próprio conteúdo de A, incluindo o valor de carry.
; $A = 17h + 02h + 0 = 19h$. O valor de carry era zero.

aci 03h ; adiciona 03h e cy ao conteúdo do acumulador. $A = 19h + 03h = 1Ch$

Situação inicial após compilar e não executar nenhuma instrução

The screenshot shows the initial state of a Z80 emulator. On the left, the 'Memoria (Instrucciones)' table lists the compiled instructions:

Dirección	Nemotécnico	Código
0100	MVI A, 05H	3E
0101		05
0102	MVI C, 02H	0E
0103		02
0104	ADD C	81
0105	ADI 10H	C6
0106		10
0107	ADC A	8F
0108	ACI 03H	CE

In the center, there are control buttons: 'Over', 'Step', 'Run', and 'Stop'. Below them are 'Bits de Estado' (S, Z, A, P, C) and 'Memoria (Datos)' and 'Memoria (Pila)' sections.

On the right, the 'Registros de la CPU' section shows the initial values of the registers:

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Register
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SP
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	PC

mvi A, 05h ;carrega o acumulador com o valor 05h

mvi C, 02h ;carrega o registrador C com o valor 02h

This screenshot shows the state of the emulator after executing the first two instructions: 'mvi A, 05h' and 'mvi C, 02h'. The 'Memoria (Instrucciones)' table is the same as in the previous screenshot.

The 'Registros de la CPU' section shows the updated values:

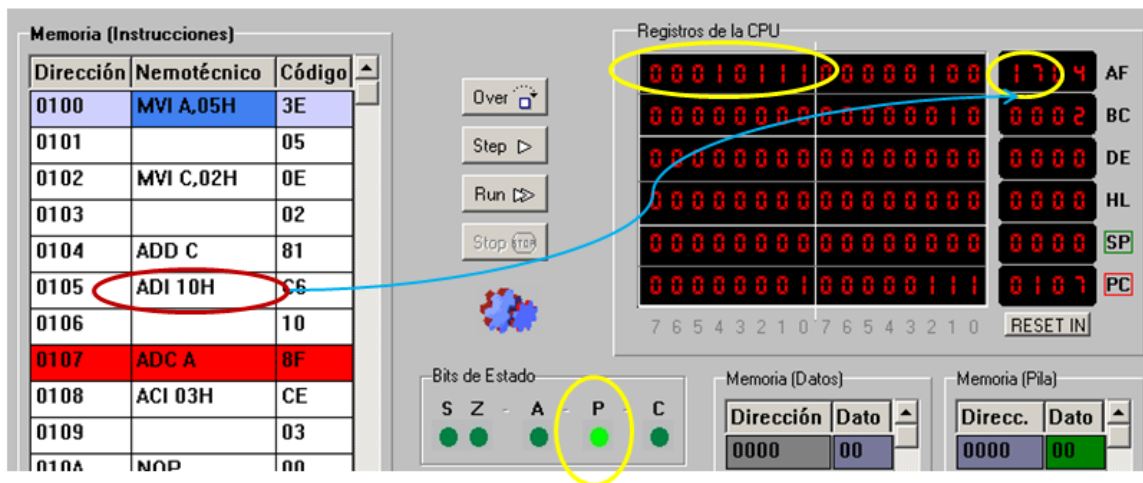
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Register
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SP
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	PC

Yellow circles highlight the changes: the 'MVI A, 05H' instruction is circled in blue, and the 'MVI C, 02H' instruction is circled in red. In the CPU registers, the AF register now contains 05 (00000101) and the BC register now contains 02 (00000010), both indicated by yellow circles and arrows from the instruction table.

add C ;adiciona o conteúdo de C ao conteúdo do acumulador. $A = 05h + 02h = 07h$



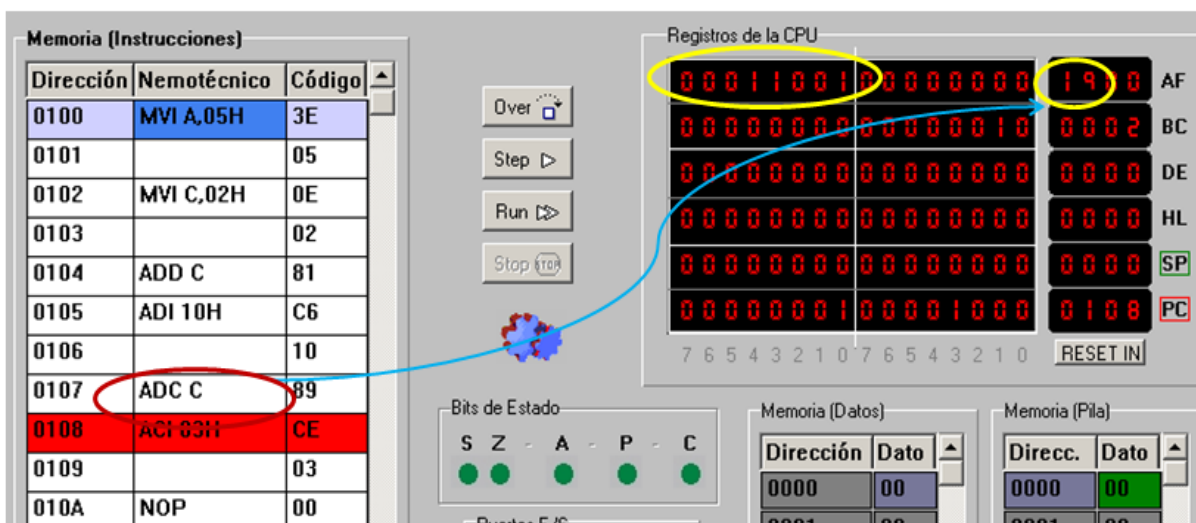
adi 10h ;adiciona 10h ao conteúdo de A. $A = 07h + 10h = 17h$



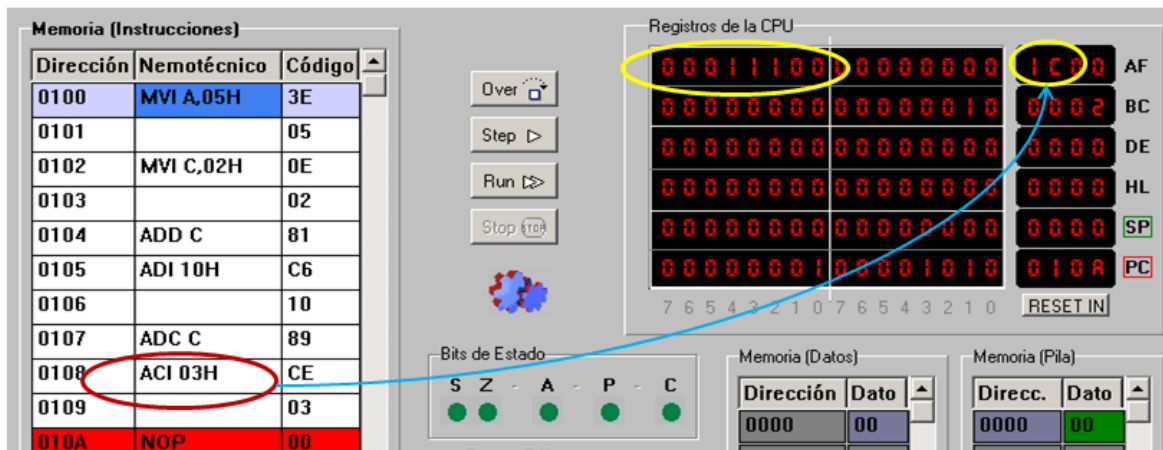
Existem cinco bits, sinal, zero, carry auxiliar, paridade e carry.

adc C ; adiciona o conteúdo de C ao próprio conteúdo de A, incluindo o valor de carry.

; $A = 17h + 02h + 0 = 19h$. O valor de carry era zero.



aci 03h ; adiciona 03h e cy ao conteúdo do acumulador. $A = 19h + 03h = 1Ch$



.org 1000

mvi A, 05h ;carrega o acumulador com o valor 05h

mvi C,02h ;carrega o registrador C com o valor 02h

lxi H,2050h ;carrega o registrador duplo HL com o valor 2050h

mvi M, 08h ;move o valor 08h para a posição de memória 2050h (apontada por HL)

sub C ;subtrai o conteúdo de C do conteúdo de A. $A = 05h - 02h = 03h$

sui 02h ;subtrai 02h do conteúdo de A. $A = 03h - 02h = 01h$

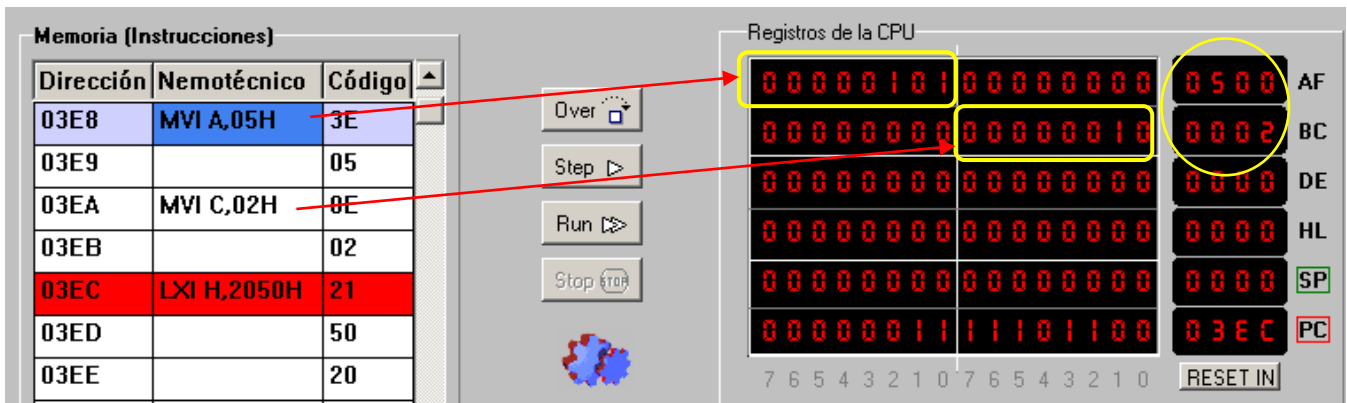
sbb C ;subtrai o conteúdo de C do conteúdo de A, incluindo o valor de carry.
;A = $01h - 02h - 0 = FFh$. O Carry antes foi Zero. Depois passou para 1.

sbi 03h ;subtrai 003h do conteúdo do acumulador, incluindo carry. $A = FFh - 02h - 1h = FBh$
;O Flag CY passa para Zero. CY = 0.

sbb M ;subtrai com carry o conteúdo da posição de memória 2050h do conteúdo do acumulador.
;A = $FBh - 08h - 0 = F3h$.

mvi A, 05h ;carrega o acumulador com o valor 05h

mvi C,02h ;carrega o registrador C com o valor 02h



lxi H,2050h ;carrega o registrador duplo HL com o valor 2050h

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03E8	MVI A,05H	3E
03E9		05
03EA	MVI C,02H	0E
03EB		02
03EC	LXI H,2050H	21
03ED		50
03EE		20

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0500	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0002	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	2050	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	1	1	1	1	1	1	0	1	1	1	1	03EF	PC

RESET IN

mvi M, 08h ;move o valor 08h para a posição de memória 2050h (apontada por HL)

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03E8	MVI A,05H	3E
03E9		05
03EA	MVI C,02H	0E
03EB		02
03EC	LXI H,2050H	21
03ED		50
03EE		20
03EF	MVI M,08H	36
03F0		08
03F1	SUB C	91
03F2	SUI 02H	D6
03F3		02

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0500	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0002	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	2050	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	1	1	1	1	1	1	0	0	0	1	1	03F1	PC

RESET IN

Bits de Estado

S Z A P C

Puertos E/S

Puerto	E	S
00	00	00
01	00	00
02	00	00
03	00	00
04	00	00

Memoria (Datos)

Dirección	Dato
204A	00
204B	00
204C	00
204D	00
204E	00
204F	00
2050	08
2051	00

Memoria (Pila)

Direcc.	Dato
0000	00
0001	00
0002	00
0003	00
0004	00
0005	00
0006	00
0007	00

E/S Serie

SID ☐ SOD ☐

Interrupciones

TRAP ☒ ☐

RST 7.5 ☒ ☐

RST 6.5 ☒ ☐

RST 5.5 ☒ ☐

Interrupciones ☒ ☐

habilita/petición

sub C ;subtrai o conteúdo de C do conteúdo de A. $A = 05h - 02h = 03h$

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03E8	MVI A,05H	3E
03E9		05
03EA	MVI C,02H	0E
03EB		02
03EC	LXI H,2050H	21
03ED		50
03EE		20
03EF	MVI M,08H	36
03F0		08
03F1	SUB C	91
03F2	SUI 02H	D6
03F3		02

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Registro	Valor
0	0	0	0	0	1	1	0	0	0	1	0	1	0	0	0	AF	03
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	BC	02
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	DE	00
0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	HL	2050
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SP	0000
0	0	0	0	0	1	1	1	1	1	0	0	1	0	0	0	PC	03F2

Bits de Estado: S Z A P C

Puertos E/S: Puerto E S

Memoria (Datos)

Dirección	Dato
204A	00
204B	00
204C	00

Memoria (Pila)

Direcc.	Dato
0000	00
0001	00
0002	00

sui 02h ;subtrai 02h do conteúdo de A. $A = 03h - 02h = 01h$

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03F1	SUB C	91
03F2	SUI 02H	D6
03F3		02
03F4	SBB C	99
03F5	SBI 03H	DE
03F6		03
03F7	SBB M	9E
03F8	NOP	00

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Registro	Valor
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	AF	01
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	BC	02
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	DE	00
0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	HL	2050
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SP	0000
0	0	0	0	0	1	1	1	1	1	0	1	0	0	0	0	PC	03F4

Bits de Estado: S Z A P C

Puertos E/S: Puerto E S

Memoria (Datos)

Dirección	Dato
204A	00

Memoria (Pila)

Direcc.	Dato
0000	00

sbb C ;subtrai o conteúdo de C do conteúdo de A, incluindo o valor de carry.

; $A = 01h - 02h - 0 = FFh$. O Carry antes foi Zero. Depois passou para 1.

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03F1	SUB C	91
03F2	SUI 02H	D6
03F3		02
03F4	SBB C	99
03F5	SBI 03H	DE
03F6		03
03F7	SBB M	9E
03F8	NOP	00
03F9	NOP	00
03FA	NOP	00
03FB	NOP	00

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Registro	Valor
1	1	1	1	1	1	1	1	0	0	1	0	1	0	1	0	AF	FF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	BC	02
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	DE	00
0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	HL	2050
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SP	0000
0	0	0	0	0	1	1	1	1	1	0	1	0	0	0	0	PC	03F5

Bits de Estado: S Z A P C

Puertos E/S: Puerto E S

Memoria (Datos)

Dirección	Dato
204A	00

Memoria (Pila)

Direcc.	Dato
0000	00

sbi 03h ;subtrai 003h do conteúdo do acumulador, incluindo carry. $A = FFh - 02h - 1h = FBh$
 ;O Flag CY passa para Zero. $CY = 0$.

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03F1	SUB C	91
03F2	SUI 02H	D6
03F3		02
03F4	SBB C	99
03F5	SBI 03H	DE
03F6		03
03F7	SBB M	9E
03F8	NOP	00
03F9	NOP	00
03FA	NOP	00

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
1	1	1	1	0	1	1	1	1	0	0	1	0	0	0	0	FBh	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0002	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	2050	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	1	1	1	1	1	0	1	1	1	1	03F7	PC

Bits de Estado

S Z A P C

Memoria (Datos)

Dirección	Dato
204A	00

Memoria (Pila)

Direcc.	Dato
0000	00

sbb M ;subtrai com carry o conteúdo da posição de memória 2050h do conteúdo do acumulador.
 ; $A = FBh - 08h - 0 = F3h$.

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03F1	SUB C	91
03F2	SUI 02H	D6
03F3		02
03F4	SBB C	99
03F5	SBI 03H	DE
03F6		03
03F7	SBB M	9E
03F8	NOP	00
03F9	NOP	00
03FA	NOP	00

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
1	1	1	0	0	1	1	1	1	0	0	0	1	0	0	0	F3h	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0002	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	2050	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	03F8	PC

Bits de Estado

S Z A P C

Memoria (Datos)

Dirección	Dato
2050	08

Memoria (Pila)

Direcc.	Dato
0000	00


```

.org 100h
mvi A, 0Fh      ;carrega o acumulador com o valor 0Fh
mvi C, 52h      ;carrega o valor 52h no registrador C
mvi B, 46h      ;carrega o valor 46h no registrador B
ana C           ;Faz operação A AND C, ou, 0F AND 52h ... A = 02h
ani 44h         ;Faz A AND 44h, ou 02h AND 44h... A = 00h
xri 23h         ;Faz A XOR 23h, ou 00h XOR 23hh...A = 23h
cpi 33h         ;Compara A com Dado imediato.
                ;Faz A - 33h, sem alterar A, ou, 23 - 33h ...Z=0 e CY = 1
rlc             ;Rotaciona A a esquerda. Resultado: A = 46h e CY = 0
cmp B           ;Compara registrador com Acumulador.
                ;Faz A - B, sem alterar A, ou, 46h - 46h ...Z = 1 e CY = 0
cmc             ;Complementa o flag de carry. CY = 1
rar             ;rotaciona A a direita com Carry.... A = A3h e CY = 0

```

mvi A, 0Fh ;carrega o acumulador com o valor 0Fh
mvi C, 52h ;carrega o valor 52h no registrador C
mvi B, 46h ;carrega o valor 46h no registrador B

Dirección	Nemotécnico	Código
0100	MVI A,0FH	3E
0101		0F
0102	MVI C,52H	0E
0103		52
0104	MVI B,46H	06
0105		46
0106	ANA C	A1

Registros de la CPU

00001111	00000000	0F00	AF
01001100	01010010	4652	BC
00000000	00000000	0000	DE
00000000	00000000	0000	HL
00000000	00000000	0000	SP
00000001	00000110	0106	PC

ana C ;Faz operação A AND C, ou, 0F AND 52h ... A = 02h

Dirección	Nemotécnico	Código
0100	MVI A,0FH	3E
0101		0F
0102	MVI C,52H	0E
0103		52
0104	MVI B,46H	06
0105		46
0106	ANA C	A1
0107	ANI 44H	E6
0108		44
0109	XRI 23H	EE

Registros de la CPU

00000010	00001000	0210	AF
01001100	01010010	4652	BC
00000000	00000000	0000	DE
00000000	00000000	0000	HL
00000000	00000000	0000	SP
00000001	00000111	0107	PC

Bits de Estado: S Z A P C

Memoria (Datos): Dirección 0000, Dato 00

Memoria (Pila): Dirección 0000, Dato 00

ani 44h

;Faz A AND 44h, ou 02h AND 44h... A = 00h

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
0100	MVI A,0FH	3E
0101		0F
0102	MVI C,52H	0E
0103		52
0104	MVI B,46H	06
0105		46
0106	ANA C	A1
0107	ANI 44H	E6
0108		44
0109	XRI 23H	EE

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor Hex	Registro
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	00	AF
0	1	0	0	0	1	1	0	0	1	0	1	0	0	1	0	4652	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0109	PC

Bits de Estado: S Z A P C (Todos están activados)

Memoria (Datos): Dirección 0000, Dato 00

Memoria (Pila): Direcc. 0000, Dato 00

xri 23h

;Faz A XOR 23h, ou 00h XOR 23hh...A = 23h

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
0100	MVI A,0FH	3E
0101		0F
0102	MVI C,52H	0E
0103		52
0104	MVI B,46H	06
0105		46
0106	ANA C	A1
0107	ANI 44H	E6
0108		44
0109	XRI 23H	EE

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor Hex	Registro
0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	2310	AF
0	1	0	0	0	1	1	0	0	1	0	1	0	0	1	0	4652	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	0109	PC

Bits de Estado: S Z A P C (Todos están activados)

Memoria (Datos): Dirección 0000, Dato 00

Memoria (Pila): Direcc. 0000, Dato 00

cpi 33h ;Compara A com Dado imediato.

;Faz A - 33h, sem alterar A, ou, 23 - 33h ...Z = 0 e CY = 1

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
0109	XRI 23H	EE
010A		23
010B	CPI 33H	FE
010C		33
010D	RLC	07
010E	CMP B	B8
010F	CMC	3F
0110	RAR	1F
0111	NOP	00
0112	NOP	00

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor Hex	Registro
0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	2310	AF
0	1	0	0	0	1	1	0	0	1	0	1	0	0	1	0	4652	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	010D	PC

Bits de Estado: S Z A P C (Todos están activados)

Memoria (Datos): Dirección 0000, Dato 00

Memoria (Pila): Direcc. 0000, Dato 00

rlc ;Rotaciona A a esquerda. Resultado: A = 46h e CY = 0

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
0109	XRI 23H	EE
010A		23
010B	CPI 33H	FE
010C		33
010D	RLC	07
010E	CMP B	B8
010F	CMC	3F
0110	RAR	1F
0111	NOP	00
0112	NOP	00
0113	NOP	00

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	1	0	0	0	1	1	0	0	1	0	0	0	1	0	0	46h	AF
0	1	0	0	0	1	1	0	0	1	0	1	0	0	1	0	46h	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	0	1	0	0	0	0	1	1	1	0	010E	PC

Bits de Estado: S Z A P C (Todos los bits están encendidos)

Memoria (Datos): Dirección 0000, Dato 00

Memoria (Pila): Direcc. 0000, Dato 00

cmp B ;Compara registrador com Acumulador.

;Faz A – B, sem alterar A, ou, 46h – 46h ...Z = 1 e CY = 0

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
0109	XRI 23H	EE
010A		23
010B	CPI 33H	FE
010C		33
010D	RLC	07
010E	CMP B	B8
010F	CMC	3F
0110	RAR	1F
0111	NOP	00
0112	NOP	00
0113	NOP	00

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	1	0	0	0	1	1	0	0	1	0	0	0	1	0	0	46h	AF
0	1	0	0	0	1	1	0	0	1	0	1	0	0	1	0	46h	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	0	1	0	0	0	1	1	1	1	0	010F	PC

Bits de Estado: S Z A P C (S, Z, A, P están encendidos; C está apagado)

Memoria (Datos): Dirección 0000, Dato 00

Memoria (Pila): Direcc. 0000, Dato 00

cmc ;Complementa o flag de carry. CY = 1

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
0109	XRI 23H	EE
010A		23
010B	CPI 33H	FE
010C		33
010D	RLC	07
010E	CMP B	B8
010F	CMC	3F
0110	RAR	1F
0111	NOP	00
0112	NOP	00
0113	NOP	00

Registros de la CPU

6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	1	0	0	0	1	1	0	1	0	0	0	1	0	0	46h	AF
0	1	0	0	0	1	1	0	1	0	1	0	0	1	0	46h	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0110	PC

Bits de Estado: S Z A P C (S, Z, A, P están encendidos; C está encendido)

Memoria (Datos): Dirección 0000, Dato 00

Memoria (Pila): Direcc. 0000, Dato 00

rar ;rotaciona A a direita com Carry.... A = A3h e CY = 0

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
0109	XRI 23H	EE
010A		23
010B	CPI 33H	FE
010C		33
010D	RLC	07
010E	CMP B	B8
010F	CMC	3F
0110	RAR	1F
0111	N JP	00
0112	NOP	00

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
1	0	1	0	0	0	1	1	0	1	0	0	0	1	0	0	A3h	AF
0	1	0	0	0	1	1	0	0	1	0	0	1	0	1	0	4652	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0111	PC

Bits de Estado: S Z A P C

Memoria (Datos)

Dirección	Dato
0000	00

Memoria (Pila)

Direcc.	Dato
0000	00

.org 1000

mvi C, 10h ;carrega o registrador C com o valor 10h.

volta: dcr c ;decrementa conteúdo do registrador C

jz fim ;se o resultado de dcr C for Zero, desvia para "fim".

mov a,c ;copia conteúdo de C em A. Não afeta nenhum flag.

jmp volta ;desvio incondicional para "volta"

fim: hlt

mvi C, 10h ;carrega o registrador C com o valor 10h.

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03E8	MVI C, 10H	0E
03E9		10
03EA	DCR C	0D
03EB	JZ 03F2H	CA
03EC		F2
03ED		03
03EE	MOV A, C	79

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0010	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	1	1	1	1	0	1	0	1	0	1	03EA	PC

Bits de Estado: S Z A P C

Memoria (Datos)

Dirección	Dato
0000	00

Memoria (Pila)

Direcc.	Dato
0000	00

dcr c ;decrementa conteúdo do registrador C

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03E8	MVI C, 10H	0E
03E9		10
03EA	DCR C	0D
03EB	JZ 03F2H	CA
03EC		F2
03ED		03
03EE	MOV A, C	79
03EF	JMP 03EAH	C3
03F0		EA
03F1		03

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0014	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0007	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	1	1	1	1	0	1	0	1	1	1	03EB	PC

Bits de Estado: S Z A P C

Memoria (Datos)

Dirección	Dato
0000	00

Memoria (Pila)

Direcc.	Dato
0000	00

jz fim ;se o resultado de dcr C for Zero, desvia para “fim”.

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03E8	MVI C,10H	0E
03E9		10
03EA	DCR C	0D
03EB	JZ 03F2H	CA
03EC		F2
03ED		03
03EE	MOV A,C	79
03EF	JMP 03EAH	C3
03F0		EA
03F1		03
03F2	HLT	76

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	AF
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SP
0	0	0	0	0	0	1	1	1	1	0	1	1	1	0	0	PC

Bits de Estado: S Z A P C (all green)

Memoria (Datos): Dirección 0000, Dato 00

Memoria (Pila): Dirección 0000, Dato 00

mov a,c ;copia conteúdo de C em A. Não afeta nenhum flag.

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03E8	MVI C,10H	0E
03E9		10
03EA	DCR C	0D
03EB	JZ 03F2H	CA
03EC		F2
03ED		03
03EE	MOV A,C	79
03EF	JMP 03EAH	C3
03F0		EA
03F1		03
03F2	HLT	76

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	AF
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SP
0	0	0	0	0	0	1	1	1	1	0	1	1	1	0	0	PC

Bits de Estado: S Z A P C (all green)

Memoria (Datos): Dirección 0000, Dato 00

Memoria (Pila): Dirección 0000, Dato 00

jmp volta ;desvio incondicional para “volta”

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03E8	MVI C,10H	0E
03E9		10
03EA	DCR C	0D
03EB	JZ 03F2H	CA
03EC		F2
03ED		03
03EE	MOV A,C	79
03EF	JMP 03EAH	C3
03F0		EA
03F1		03
03F2	HLT	76

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	AF
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SP
0	0	0	0	0	0	1	1	1	1	0	1	0	1	0	0	PC

Bits de Estado: S Z A P C (all green)

Memoria (Datos): Dirección 0000, Dato 00

Memoria (Pila): Dirección 0000, Dato 00

Nota: A instrução JMP é um Salto incondicional. O programa acima ficará em “loop” até que a condição para a instrução JZ seja satisfeita, isto é, jz fim ;se o resultado de dcr C for Zero, o programa desviará para “fim”..

.org 1000

mvi A, 07h ;carrega o registrador A com o valor 07h

mvi B, 00h ;carrega o registrador B com o valor 00h

Volta:

inr B ;incrementa em "1" o conteúdo do registrador B

cmp B ;compara o conteúdo do registrador B com o conteúdo de A, sem alterar A.

jnc volta ;Desvia para "volta" se o flag CY=0. CY=0 se A > B ou A = B

hlt ;para o processamento quando CY = 1, ou seja, quando B=8 (A < B).

mvi A, 07h ;carrega o registrador A com o valor 07h
mvi B, 00h ;carrega o registrador B com o valor 00h

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03E8	MVI A,07H	3E
03E9		07
03EA	MVI B,00H	06
03EB		00
03EC	INR B	04
03ED	CMP B	B8
03EE	JNC 03ECH	D2

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	07	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	00	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	00	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	00	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	00	SP
0	0	0	0	0	0	1	1	1	1	0	1	1	0	0	0	03EC	PC

inr B ;incrementa em "1" o conteúdo do registrador B

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03E8	MVI A,07H	3E
03E9		07
03EA	MVI B,00H	06
03EB		00
03EC	INR B	04
03ED	CMP B	B8
03EE	JNC 03ECH	D2

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	07	AF
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	01	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	00	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	00	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	00	SP
0	0	0	0	0	0	1	1	1	1	0	1	1	0	0	0	03ED	PC

cmp B ;compara o conteúdo do registrador B com o conteúdo de A, sem alterar A.

Memoria (Instrucciones)

Dirección	Nemotécnico	Código
03E8	MVI A,07H	3E
03E9		07
03EA	MVI B,00H	06
03EB		00
03EC	INR B	04
03ED	CMP B	B8
03EE	JNC 03ECH	D2
03EF		EC
03F0		03
03F1	HLT	76
03F2	MOD	nn

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	0	0	0	0	1	1	1	0	0	0	1	0	1	0	0	0714	AF
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0100	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	1	1	1	1	0	1	1	0	0	0	03EE	PC

Bits de Estado: S Z A P C

Memoria (Datos)

Dirección	Dato
0000	00

Memoria (Pila)

Dirección	Dato
0000	00

Nota: Condições: Se $(A) = (r)$ então $Z = 1$ e $CY = 0$

Se $(A) > (r)$ então $Z = 0$ e $CY = 0$

Se $(A) < (r)$ então $Z = 0$ e $CY = 1$

jnc volta ;Desvia para “volta” se o flag CY=0. CY =0 se $A > B$ ou $A = B$

The screenshot shows a 8085 assembly simulator interface. On the left, the 'Memoria (Instrucciones)' table lists instructions: 03E8 MVI A,07H (3E), 03E9 (07), 03EA MVI B,00H (06), 03EB (00), 03EC INR B (04), 03ED CMP B (B8), 03EE JNC 03ECH (D2), 03EF (EC), 03F0 (03), 03F1 HLT (76), and 03F2 NOP (00). The instruction at 03EE is highlighted with a red box. In the center, there are control buttons: Over, Step, Run, and Stop. Below them is a gear icon and 'Bits de Estado' (S, Z, A, P, C) with green indicator lights. On the right, 'Registros de la CPU' shows AF (0714), BC (0100), DE (0000), HL (0000), SP (0000), and PC (03EC). At the bottom right, 'Memoria (Datos)' and 'Memoria (Pila)' both show address 0000 with data 00.

Nota: A instrução JNC representa “Saltar se não ocorrer carry”.

Programa para efetuar multiplicação de 4 por 3

.org 1000

mvi A, 00h ;Zera acumulador. A = 00h

mvi C, 03h ;Carrega registrador C com 03h. “C” será usado como contador.

Volta:

adi 04h ;adiciona imediato 04h ao acumulador. A = A + 04h

dcr C ;decrementa o contador C = C - 1

jnz volta ;”volta” se o resultado de DCR C não for zero (Se Z = 0)

hlt ;encerra processamento quando C = 0, ou seja, flag Z = 1

mvi A, 00h ;Zera acumulador. A = 00h

mvi C, 03h ;Carrega registrador C com 03h. "C" será usado como contador.

Memoria (Instruções)

Dirección	Nemotécnico	Código
03E8	MVI A,00H	3E
03E9		00
03EA	MVI C,03H	0E
03EB		03
03EC	ADI 04H	C6
03ED		04
03EE	DCR C	0D

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0003	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	0	1	1	1	1	0	1	1	0	0	03EC	PC

adi 04h ;adiciona imediato 04h ao acumulador. A = A + 04h

Memoria (Instruções)

Dirección	Nemotécnico	Código
03E8	MVI A,00H	3E
03E9		00
03EA	MVI C,03H	0E
03EB		03
03EC	ADI 04H	C6
03ED		04
03EE	DCR C	0D

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0400	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0003	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	0	1	1	1	1	0	1	1	0	0	03EE	PC

dcr C ;decrementa o contador C = C - 1

Memoria (Instruções)

Dirección	Nemotécnico	Código
03E8	MVI A,00H	3E
03E9		00
03EA	MVI C,03H	0E
03EB		03
03EC	ADI 04H	C6
03ED		04
03EE	DCR C	0D

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0400	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0102	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	0	1	1	1	1	0	1	1	0	0	03EF	PC

jnz volta ;"volta" se o resultado de DCR C não for zero (Se Z = 0)

Memoria (Instruções)

Dirección	Nemotécnico	Código
03E8	MVI A,00H	3E
03E9		00
03EA	MVI C,03H	0E
03EB		03
03EC	ADI 04H	C6
03ED		04
03EE	DCR C	0D
03EF	JNZ 03ECH	C2
03F0		EC
03F1		03

Registros de la CPU

7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	Valor	Registro
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0400	AF
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0102	BC
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	DE
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	HL
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	SP
0	0	0	0	0	0	0	1	1	1	1	0	1	1	0	0	03EC	PC

Bits de Estado

S	Z	A	P	C
●	●	●	●	●

Memoria (Datos)

Dirección	Dato
0000	00

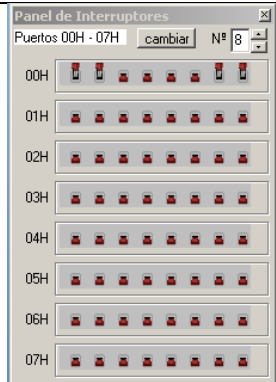
Memoria (Pila)

Dirección	Dato
0000	00

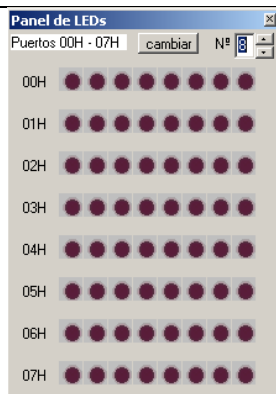
Informações úteis

Instrução in (entrada de dados)

A instrução IN PORT lê os 8 bits de dados que estão na “PORTA” especificada e os carrega no acumulador. O operando deve ser um número ou uma expressão que produza um valor compreendido entre 00H e FFH.

	<ul style="list-style-type: none"> – Pode ser utilizado o Painel de Interruptores para carregar um valor no acumulador. – Observe que a Porta de endereço 00H está carregada com o valor binário 11000011; – Para acessar a porta 00h através do programa basta digitar a instrução: <ul style="list-style-type: none"> – in 00h
--	---

Instrução out (saída de dados)

	<ul style="list-style-type: none"> – Existem diversos dispositivos de saída: <ul style="list-style-type: none"> – LEDs, display de 7 segmentos, display de 15 segmentos, display monocromático e display colorido. – Pode ser utilizado o Painel de LEDs para carregar um valor existente no acumulador. – Para acessar a porta 00h através do programa basta digitar a instrução: <ul style="list-style-type: none"> – out 00h
--	--

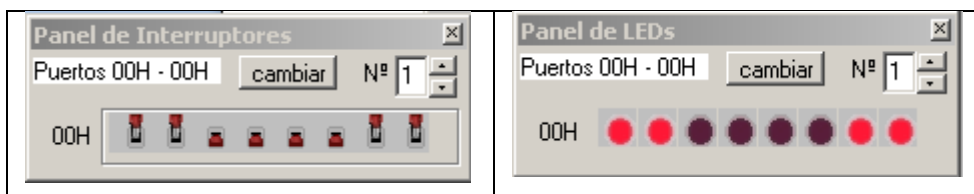
Exemplo de programa:

```
.org 1000
```

```
in 00h
```

```
out 00h
```

Resultado:



Outros Programas

Programa para contar de 0 a 60

;aulax_15.asm

```
.org 1000
    mvi A,00h
volta:
    inr A
    out 00h
    cpi 60h
    jz fim
    jmp volta
fim:
    hlt
```