microprocessor 8085 programs

Swap two 8 bit numbers

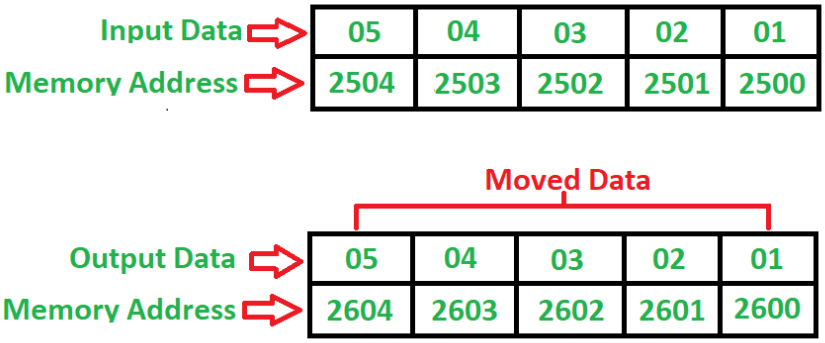
INPUT	
ADDRESS	DATA
2500	05
2501	06

OUTPUT	
ADDRESS	DATA
2500	06
2501	05

Swapped

Address	Mnemonics	Comments
2000	LDA 2500	A<-[2500]
2003	MOV B, A	B<-A
2004	LDA 2501	A<-[2501]
2007	STA 2500	2500<-[A]
200A	MOV A, B	A<-B
200B	STA 2501	2501<-[A]
200E	HLT	Terminates the program

To move a block of data bytes from one location to another location.



Memory	Mnemonics	Operands	Comment
2000	MVI	C, 05	[C] <- 05
2002	LXI	H, 2500	[H-L] <- 2500
2005	LXI	D, 2600	[D-E] <- 2600
2008	MOV	A, M	[A] <- [[H-L]]
2009	STAX	D	[A] -> [[D-E]]
200A	INX	H	[H-L] <- [H-L] + 1
200B	INX	D	[D-E] <- [D-E] + 1
200C	DCR	C	[C] <- [C] – 1
200D	JNZ	2008	Jump if not zero to 2008
2010	HLT		Stop

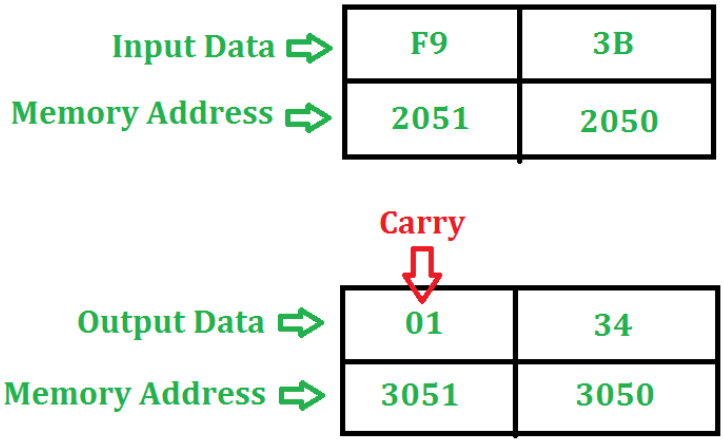
Sum of two 8 bit numbers without carry

```
Input: 2050: 03
     : 2051: 04
```

```
Output: 2052: 07
```

2000	LDA 2050	A<-[2050]
2003	MOV B, A	B<-A
2004	LDA 2051	A<-[2051]
2007	ADD B	A<-A+B
2008	STA 2052	[2052]<-A
200B	HLT	Terminate

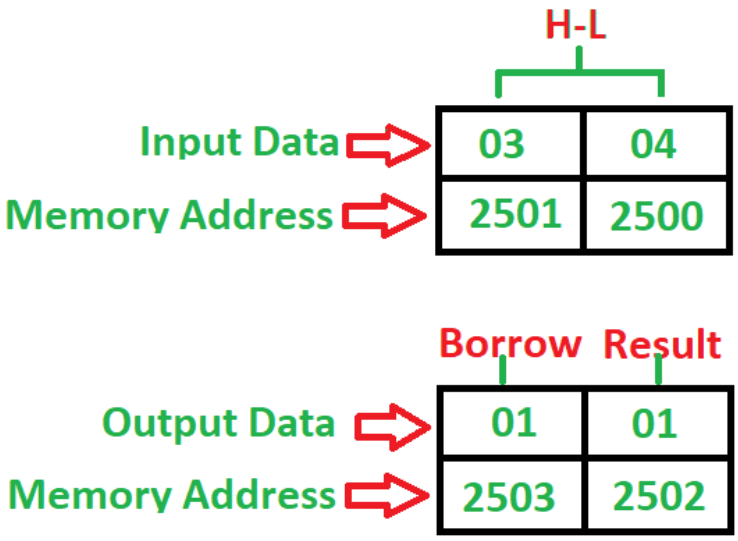
Sum of two 8 bit numbers with carry(addition)



2000	LDA 2050	A<-[2050]
2003	MOV H, A	H<-A
2004	LDA 2051	A<-[2051]
2007	ADD H	A<-A+H
2008	MOV L, A	L←A
2009	MVI A 00	A←00
200B	ADC A	A←A+A+carry
200C	MOV H, A	H←A

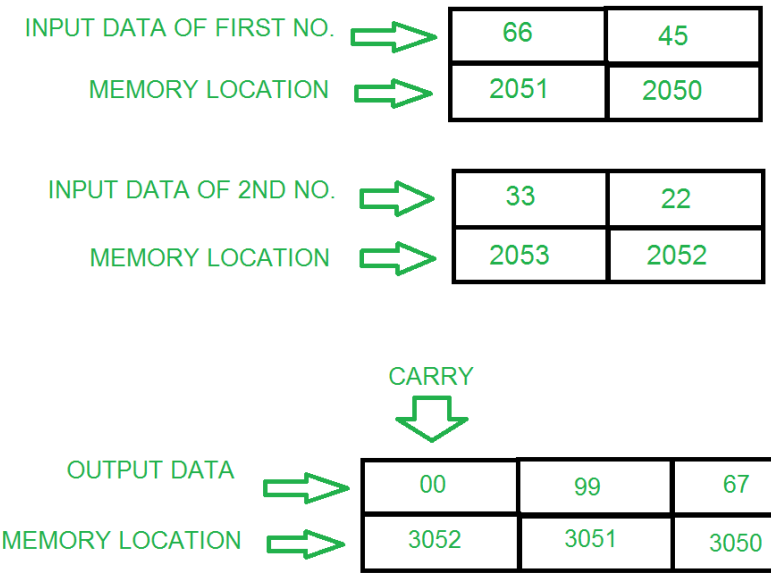
2000	LDA 2050	A<-[2050]
200D	SHLD 3050	H→3051, L→3050
2010	HLT	

difference of two 8 bit number with/without borrow(subtraction)



000	MVI	C, 00	[C] <- 00
2002	LHLD	2500	[H-L] <- [2500]
2005	MOV	A, H	[A] <- [H]
2006	SUB	L	[A] <- [A] – [L]
2007	JNC	200B	Jump If no borrow
200A	INR	C	[C] <- [C] + 1
200B	STA	2502	[A] -> [2502], Result
200E	MOV	A, C	[A] <- [C]
2010	STA	2503	[A] -> [2503], Borrow
2013	HLT		Stop

Addition of 16 bit numbers with carry.

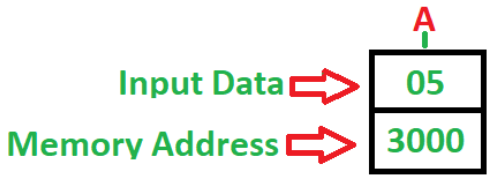


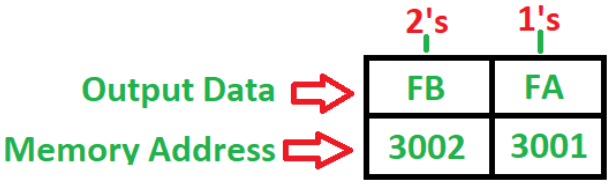
MEMORY ADDRESS	MNEMONICS	COMMENTS
2000	LDA 2050	A ← 2050
2003	MOV B, A	B ← A
2004	LDA 2052	A ← 2052
2007	ADD B	A ← A+B
2008	STA 3050	A → 3050
200B	LDA 2051	A ← 2051
200E	MOV B, A	B ← A
200F	LDA 2053	A ← 2053
2012	ADC B	A ← A+B+CY
2013	STA 3051	A → 3051
2016	HLT	Stops execution

2000	LHLD 2050	H-L ← 2050
2003	XCHG	D H & E L
2004	LHLD 2052	H-L ← 2052
2007	DAD D	H ← H+D & L ← L+E
2008	SHLD 3050	A → 3050

2000	LHLD 2050	H-L ← 2050
200B	HLT	Stops execution

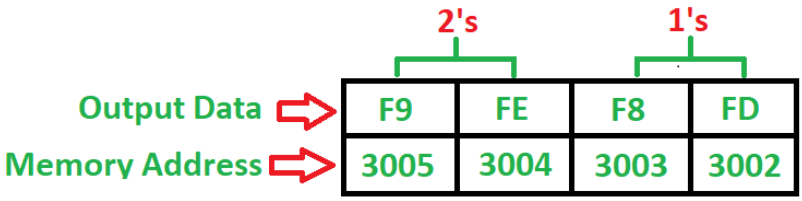
One's and two's complement of 8 bit data.





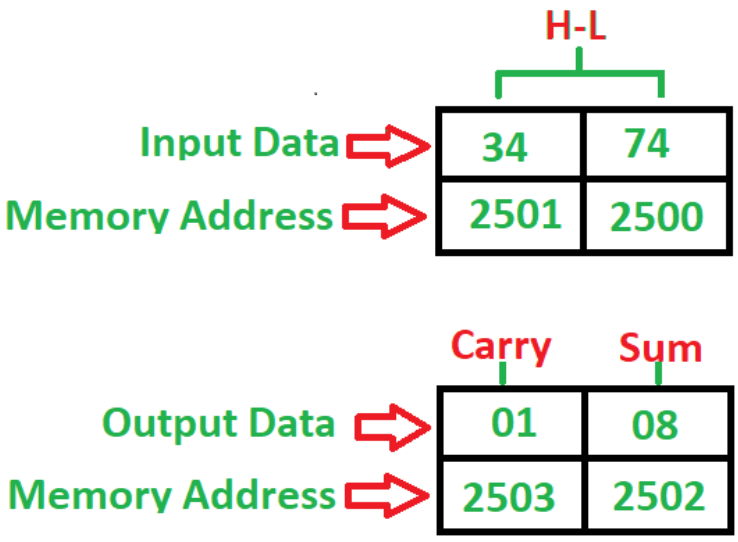
2000	LDA	[3000]	[A] <- [3000]
2003	CMA		[A] <- [A^]
2004	STA	[3001]	1’s complement
2007	ADI	01	[A] <- [A] + 01
2009	STA	[3002]	2’s complement
200C	HLT		Stop

One's and two's complement of 16 bit data.



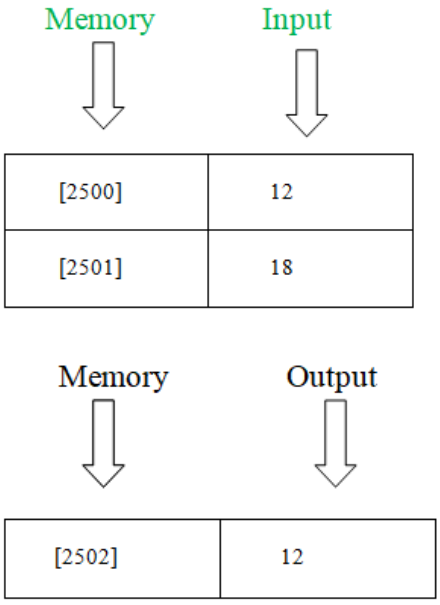
Memory	Mnemonics	Operands	Comment
2000	LHLD	[3000]	[H-L] <- [3000]
2003	MOV	A, L	[A] <- [L]
2004	CMA		[A] <- [A^]
2005	MOV	L, A	[L] <- [A]
2006	MOV	A, H	[A] <- [H]
2007	CMA		[A] <- [A^]
2008	MOV	H, A	[H] <- [A]
2009	SHLD	[3002]	1’s complement
200C	INX	H	[H-L] <- [H-L] + 1
200D	SHLD	[3004]	2’s complement
2010	HLT		Stop

Add two 8 bit BCD data.



Memory	Mnemonics	Operands	Comment
2000	MVI	C, 00H	[C] <- 00H, carry
2002	LHLD	[2500]	[H-L] <- [2500]
2005	MOV	A, L	[A] <- [L]
2006	ADD	H	[A] <- [A] + [H]
2007	DAA		Add 06 if sum > 9 or AC = 1
2008	JNC	200C	Jump if no carry
200B	INR	C	[C] <- [C] + 1
200C	STA	[2502]	[A] -> [2502], sum
200F	MOV	A, C	[A] <- [C]
2010	STA	[2503]	[A] -> [2503], carry
2013	HLT		Stop

Find larger/smaller number between two numbers.



Memory	Mnemonics	Use Operand	Comments
2000	LDA	[2500]	[A]<-[2500]
2003	MOV B, A		[B]<-[A]
2004	LDA	2501	[A]<-[2501]
2007	CMP B		[A]<-[A]-[B]
2008	JC *	[200C]	jump carry
200B	MOV A, B		[A]<-[B]
200C	STA	[2502]	[A]->[2502]
200F	HLT		STOP

find largest /smallest in a series of n number.

Address	Label	Instruction	Comment
2000H		LXI H, 2050H	Load starting address of list
2003H		MOV B, M	Store maximum
2004H		MOV C, M	Store minimum

Address	Label	Instruction	Comment
2005H		MVI D, 0AH	Counter for 10 elements
2007H	LOOP	MOV A, M	Retrieve list element in Accumulator
2008H		CMP B	Compare element with maximum number
2009H		JC MIN	Jump to MIN if not maximum
200CH		MOV B, A	Transfer contents of A to B as A > B
200DH	MIN	CMP C	Compare element with minimum number
200EH		JNC SKIP	Jump to SKIP if not minimum
2011H		MOV C, A	Transfer contents of A to C if A < minimum
2012H	SKIP	INX H	Increment memory
2013H		DCR D	Decrement counter
2014H		JNZ LOOP	Jump to LOOP if D > 0
2017H		LXI H, 2060H	Load address to store maximum
201AH		MOV M, B	Move maximum to 2060H
201BH		INX H	Increment memory
201CH		MOV M, C	Move minimum to 2061H
201DH		HLT	Halt

Multiplication of 8 bit number.

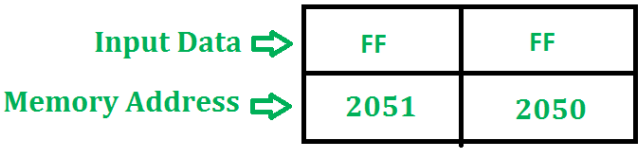
Input Data ⇨	07	43
Memory Address ⇨	2051	2050

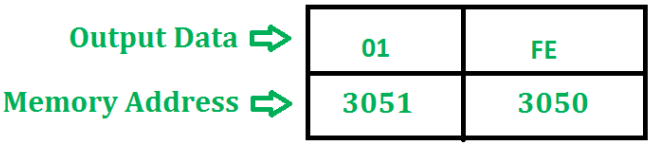
Output Data ⇨	01	D5
Memory Address ⇨	3051	3050

Memory Address	Mnemonics	Comment
2000	LHLD 2050	H←2051, L←2050
2003	XCHG	H↔D, L↔E
2004	MOV C, D	C←D
2005	MVI D 00	D←00
2007	LXI H 0000	H←00, L←00
200A	DAD D	HL←HL+DE
200B	DCR C	C←C-1
200C	JNZ 200A	If Zero Flag=0, goto 200A
200F	SHLD 3050	H→3051, L→3050
2012	HLT	

ADDRESS	MNEMONICS	COMMENT
2000	LXI H, 2050H	
2003	MOV B, M	B←M
2004	INX H	
2005	MOV C, M	C←M
2006	MVI A, 00H	A←00
2008	TOP:ADD B	A<-A+B
2009	DCR C	C←C-1
200A	JNZ TOP	
200D	INX H	
200E	MOV M, A	M←A
200F	HLT	terminate the program

Division of 8 bit number.





ADDRESS	MNEMONICS	COMMENT
2000	LXI H, 2050	
2003	MOV B, M	B<-M
2004	MVI C, 00	C<-00H
2006	INX H	
2007	MOV A, M	A<-M
2008	CMP B	
2009	JC 2011	check for carry
200C	SUB B	A<-A-B
200D	INR C	C<-C+1
200E	JMP 2008	
2011	STA 3050	3050<-A
2014	MOV A, C	A<-C
2015	STA 3051	3051<-A
2018	HLT	terminate the program