Subtraction of two 16-bit numbers

By,
Subathra S

SUBTRACTION OF TWO 16-BIT NUMBERS

AIM

To write a program to find the subtraction of two 16-bit number (i) without borrow, (ii) with borrow

SUBTRACTION WITHOUT BORROW

ASSEMBLY LANGUAGE PROGRAM

```
C600 LHLD C500   2A ; Load the HL register pair directly with
C601             00 ; Minuend
C602             C5 ;
C603 XCHG        EB ; Exchange the content of HL register pair DE
                      register pair
C604 LHLD C502   2A ; Load the HL register pair directly with
C605             02 ; Subtrahend
C606             C5 ;
C607 MOV A E     7B ; Move the content of E register to
                      accumulator
C608 SUB L       95 ; Subtract the L register from accumulator
C609 STA  C504   32 ; Store the accumulator content(lower byte of
C60A             04 ; Difference) at address C504H
C60B             C5 ;
C60C MOV A D     7A ; Move the content of D register from
                      accumulator
C60D SUB H       94 ; Subtract the H register content from the
                      accumulator
C60E STA  C505   32 ; Store the accumulator content(Higher byte of
C60F             05 ; Difference) at address C505H
C610             C5 ;
C611 HLT         76 ; Halt the exection
```

EXECUTION

```
C500 EF ; Lower byte of Minuend(Input data)
C501 CD ; Higher byte of Minuend(Input data)
C502 CC ; Lower byte of Subtrahend(Input data)
C503 AB ; Higher byte of Subtrahend(Input data)
C504 23 ; Lower byte of Difference(Output data)
C505 22 ; Higher byte of Difference(Output data)
```

PROGRAM TRACE

Addr	MC	Mnemonic	A	B	C	D	E	H	L	SP	Flag Word
			00	00	00	00	00	00	00	0000	0000 0000
C600	2A	LHLD C500	00	00	00	00	00	CD	EF	0000	0000 0000
C603	EB	XCHG	00	00	00	CD	EF	00	00	0000	0000 0000
C604	2A	LHLD C502	00	00	00	CD	EF	AB	CC	0000	0000 0000
C607	7B	MOV A E	EF	00	00	CD	EF	AB	CC	0000	0000 0000
C608	95	SUB L	23	00	00	CD	EF	AB	CC	0000	0001 0000
C609	32	STA C504	23	00	00	CD	EF	AB	CC	0000	0001 0000
C60C	7A	MOV A D	CD	00	00	CD	EF	AB	CC	0000	0001 0000
C60D	94	SUB H	22	00	00	CD	EF	AB	CC	0000	0001 0100
C60E	32	STA C505	22	00	00	CD	EF	AB	CC	0000	0001 0100
C611	76	HLT	22	00	00	CD	EF	AB	CC	0000	0001 0100

FLAG WORD

S	Z	x	Ac	x	P	x	Cy
0	0	0	1	0	1	0	0

SUBTRACTION WITH BORROW

ASSEMBLY LANGUAGE PROGRAM

```
C800 MVI C 00    0E ; Initialise the C register with 00H
C801             00 ;
C802 LHLD C700   2A ; Load the HL register pair directly with
C803             00 ; Minuend
C804             C7 ;
C805 XCHG        EB ; Exchange the content of HL register pair DE
                    register pair
C806 LHLD C702   2A ; Load the HL register pair directly with
C807             02 ; Subtrahend
C808             C7 ;
C809 MOV A E     7B ; Move the content of E register to
                    accumulator
C80A SUB L       95 ; Subtract the L register from accumulator
C80B STA  C704   32 ; Store the accumulator content(lower byte of
C80C             04 ; Difference) at address C704H
C80D             C7 ;
C80E MOV A D     7A ; Move the content of D register from
                    accumulator
C80F SBB H       9C ; Subtract the H register content from
                    accumulator
C810 STA  C705   32 ; Store the content of accumulator(higher byte
C811             05 ; of Difference) at address C705H
C812             C7 ;
```

```
C813 JNC C819     D2 ; If carry = 0, then jump to C819H
C814               19 ;
C815               C8 ;
C816 INR C        0C ; Increment the C register content
C817 CMA          2F ; Complement the accumulator content
C818 INR A        3C ; Increment the accumulator content
C819 MOV A C      79 ; Move the C register content to accumulator
C81A STA  C708    32 ; Store the accumulator content(Borrow) to
C81B               08 ; memory address C708H
C81C               C7 ;
C81D HLT          76 ; Halt the execution
```

EXECUTION – 1

```
C700 88 ; Lower byte of Minuend(Input data)
C701 89 ; Higher byte of Minuend(Input data)
C702 C5 ; Lower byte of Subtrahend(Input data)
C703 C4 ; Higher byte of Subtrahend(Input data)
C704 C3 ; Lower byte of Difference(Output data)
C705 C4 ; Higher byte of Difference(Output data)
C708 01 ; Borrow(Output data)
```

PROGRAM TRACE

Addr	MC	Mnemonic	A	B	C	D	E	H	L	SP	Flag Word
			00	00	00	00	00	00	00	0000	0000 0000
C800	0E	MVI C 00	00	00	00	00	00	00	00	0000	0000 0000
C802	2A	LHLD C700	00	00	00	00	00	89	88	0000	0000 0000
C805	EB	XCHG	00	00	00	89	88	00	00	0000	0000 0000
C806	2A	LHLD C702	00	00	00	89	88	C4	C5	0000	0000 0000
C809	7B	MOV A E	88	00	00	89	88	C4	C5	0000	0000 0000
C80A	95	SUB L	C3	00	00	89	88	C4	C5	0000	1001 0101
C80B	32	STA C704	C3	00	00	89	88	C4	C5	0000	1001 0101
C80E	7A	MOV A D	89	00	00	89	88	C4	C5	0000	1001 0101
C80F	9C	SBB H	C4	00	00	89	88	C4	C5	0000	1001 0001
C810	32	STA C705	C4	00	00	89	88	C4	C5	0000	1001 0001
C813	D2	JNC C819	C4	00	00	89	88	C4	C5	0000	1001 0001
C816	0C	INR C	C4	00	01	89	88	C4	C5	0000	0000 0001
C817	2F	CMA	3B	00	01	89	88	C4	C5	0000	0000 0001
C818	3C	INR A	3C	00	01	89	88	C4	C5	0000	0000 0101
C819	79	MOV A C	01	00	01	89	88	C4	C5	0000	0000 0101
C81A	32	STA C708	01	00	01	89	88	C4	C5	0000	0000 0101
C81D	76	HLT	01	00	01	89	88	C4	C5	0000	0000 0101

FLAG WORD

S	Z	x	Ac	x	P	x	Cy
0	0	0	0	0	1	0	1

EXECUTION –2

```
C700 C5 ; Lower byte of Subtrahend(Input data)
C701 C4 ; Higher byte of Subtrahend(Input data)
C702 88 ; Lower byte of Minuend(Input data)
C703 89 ; Higher byte of Minuend(Input data)
C704 3D ; Lower byte of Difference(Output data)
C705 3B ; Higher byte of Difference(Output data)
C708 00 ; Borrow(Output data)
```

PROGRAM TRACE

Addr	MC	Mnemonic	A	B	C	D	E	H	L	SP	Flag Word
			00	00	00	00	00	00	00	0000	0000 0000
C800	0E	MVI C 00	00	00	00	00	00	00	00	0000	0000 0000
C802	2A	LHLD C700	00	00	00	00	00	C4	C5	0000	0000 0000
C805	EB	XCHG	00	00	00	C4	C5	00	00	0000	0000 0000
C806	2A	LHLD C702	00	00	00	C4	C5	89	88	0000	0000 0000
C809	7B	MOV A E	C5	00	00	C4	C5	89	88	0000	0000 0000
C80A	95	SUB L	3D	00	00	C4	C5	89	88	0000	0000 0000
C80B	32	STA C704	3D	00	00	C4	C5	89	88	0000	0000 0000
C80E	7A	MOV A D	C4	00	00	C4	C5	89	88	0000	0000 0000
C80F	9C	SBB H	3B	00	00	C4	C5	89	88	0000	0000 0000
C810	32	STA C705	3B	00	00	C4	C5	89	88	0000	0000 0000
C813	D2	JNC C819	3B	00	00	C4	C5	89	88	0000	0000 0000
C819	79	MOV A C	00	00	00	C4	C5	89	88	0000	0000 0000
C81A	32	STA C708	00	00	00	C4	C5	89	88	0000	0000 0000
C81D	76	HLT	00	00	00	C4	C5	89	88	0000	0000 0000

FLAG WORD

S	Z	x	Ac	x	P	x	Cy
0	0	0	0	0	0	0	0

REFERENCE

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

2. S.Subathra, “Microprocessor Laboratory”, Record work, Adhiparashakthi Engineering College, Melmaruvathur, March 2001

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.

5. Micro85 simulation software, Infotech Solutions, Calcutta.