

SMALLEST AND LARGEST

AIM

To develop an assembly language program to find the largest and smallest of 8-bit numbers.

ALGORITHM

Algorithm 1 Largest and smallest of 8-bit numbers

```
1: Start
2: Read the size of array (n).
3: Read the array (arr).
4: max = arr [0]
5: min = arr [0]
6: for i = 1 to (n- 1) do
7:     if arr[i]>max then
8:         max = arr[i]
9:     else if arr[i]< min then
10:        min = arr[i]
11:    end if
12: end if
13: Print largest (max) and smallest (min) of the numbers.
14: Stop
```

SOURCE CODE

```
ORG 0000H
MOV R0, #030H ;Start address of array
MOV R7, #04H ;number of elements
MOV A, @R0 ;smallest stored in R1
MOV R1, A
MOV B, @R0
MOV R2, B ;largest stored in R2
INC R0
Find: DJNZ R7, next
JMP END
next: CLR C
MOV A, R1
SUBB A, @R0
JNC small ;a is small
JMP check
small: MOV A, @R0
MOV R1, A
check: CLR C
MOV A, R2
SUBB A, @R0
JC large
JMP finish
large: MOV A, @R0
MOV R2, A
finish: INC R0
```

JMP find ;smallest in R1 ,largest in R2

END: END

INPUT :

	x0	x1	x2	x3	x4	x5	x6	x7	
00	00	00	00	00	00	00	00	00	00
08	00	00	00	00	00	00	00	00	00
10	00	00	00	00	00	00	00	00	00
18	00	00	00	00	00	00	00	00	00
20	00	00	00	00	00	00	00	00	00
28	00	00	00	00	00	00	00	00	00
30	10	15	09	45	00	00	00	00	00

HEX DEC BIN OCT CHAR

A: 00 0 00000000 0

B: 00 0 00000000 0

PSW: C AC F0 RS1 RS0 OV - P

R7 R6 R5 R4 R3 R2 R1 R0

00 00 00 00 00 00 00 00

Input at addresses starting from 0x30

OUTPUT :

3 : 8.asm - MCU 8051 IDE

File Edit View Project Simulator Virtual MCU Virtual HW Tools Utilities Configure Help

1.asm 2 1.asm(0) 3

9.asm 8.asm

Opened Files:

- 9.asm
- 8.asm

```
3 MOV R7, #04H ;number of elements
4 MOV A, @R0 ;smallest stored in R1
5 MOV R1, A
6 MOV B, @R0
7 MOV R2, B ;largest stored in R2
8 INC R0
9 Find: DJNZ R7, next
10 JMP END
11 next: CLR C
12 MOV A, R1
13 SUBB A, @R0
14 JNC small ;a is small
15 JMP check
```

Simulator mode INS NORM 8.asm ASM

Simulator C variables IO Ports Messages Notes Calculator Terminal Find in files Hide

	x0	x1	x2	x3	x4	x5	x6	x7	
00	34	09	45	00	00	00	00	00	00
08	00	00	00	00	00	00	00	00	00
10	00	00	00	00	00	00	00	00	00
18	00	00	00	00	00	00	00	00	00
20	00	00	00	00	00	00	00	00	00
28	00	00	00	00	00	00	00	00	00
30	10	15	09	45	00	00	00	00	00

HEX DEC BIN OCT CHAR

A: 45 69 01000101 105 E

B: 10 16 00010000 20

PSW: C AC F0 RS1 RS0 OV - P

R7 R6 R5 R4 R3 R2 R1 R0

00 00 00 00 00 45 09 34

TIMERS 0 & 1

TH1 TL1 TH0 TL0 T1 T0 TCON TMOD

00 00 00 00 0 0 00 00

TCON: TF1 TR1 TF0 TR0 IE1 IT1 IE0 IT0

TMOD: G1 CT1 M11 M01 G0 CT0 M10 M00

INTERRUPTS

IE: EA - ET2 ES ET1 EX1 ET0 EX0

IP: - - PT2 PS PT1 PX1 PT0 PX0

BIN HEX DPH DPL HEX

P0: 11111111 FF DPTR0: 00 00 SP: 07

P1: 11111111 FF DPTR1: 00 00 SBUF R: F7

P2: 11111111 FF Clock: 12000 SBUF T: 45

P3: 11111111 FF PC: 0028 DEC 40

Time: 70.000

PCON: SMOD - - GF1 GF0 PD IDL 00

Smallest in R1 and Largest in R2

RESULT:

Assembly language programs for finding the largest and smallest of 8-bit numbers have been developed and verified using MCU-8051-IDE.

SACHIN G
R0:54