23VLS1401: Microcontroller and Computer architecture

Lecture 3 (U5)

Arithmetic instructions and programming for Microcontroller 8051

A presentation by

Dr. Shubhangi Rathkanthiwar

Professor

Department of Electronics Engineering, YCCE, Nagpur, India

Session objectives

- To overview Arithmetic instructions for the Microcontroller 8051
- To develop the programming technique in assembly language for given problem statement, store the source data, execute the program and observe the result in destination register or memory location.

Arithmetic instructions

- INC destination
- DEC destination
- ADD/ADDC destination, source
- SUBB destination, source
- MUL AB
- DIV AB
- DA A

Arithmetic instructions

Opcode	Operand	Operation
INC	A	Increment contents of Accumulator by 1
INC	Rr	Increment contents of Register by 1
INC	addr	Increment contents of Specified register by 1
INC	@Rp	Increment contents of Memory location whose address is given Register-Pointer by 1
INC	DPTR	Increment contents of Accumulator by 1

Arithmetic instructions

Opcode	Operand	Operation
DEC	A	Decrement contents of Accumulator by 1
DEC	Rr	Decrement contents of Register by 1
DEC	addr	Decrement contents of Specified register by 1
DEC	@Rp	Decrement contents of Memory location whose address is given Register-Pointer by 1

Arithmetic instructions

Opcode	Operand	Operation
ADD	A,#n	Add immediate data ‘n’ with Accumulator. Place the sum in the accumulator.
ADD	A,Rr	Add contents of specified register with Accumulator. Place the sum in the accumulator.
ADD	A,addr	Add contents of memory location whose address is specified in the instruction, with Accumulator. Place the sum in the accumulator.
ADD	A,@Rp	Add contents of memory location whose address is specified in the register pointer, with Accumulator. Place the sum in the accumulator.

Arithmetic instructions

Opcode	Operand	Operation
ADDC	A,#n	Add immediate data ‘n’ and the Carry with Accumulator. Place the sum in the accumulator.
ADDC	A,Rr	Add contents of specified register and the Carry with Accumulator. Place the sum in the accumulator.
ADDC	A,addr	Add contents of memory location whose address is specified in the instruction, and the Carry with Accumulator. Place the sum in the accumulator.
ADDC	A,@Rp	Add contents of memory location whose address is specified in the register pointer, and the Carry with Accumulator. Place the sum in the accumulator.

Arithmetic instructions

Opcode	Operand	Operation
SUBB	A,#n	Subtract immediate data 'n' and the Borrow from the Accumulator. Place the Difference in the accumulator.
SUBB	A,Rr	Subtract contents of specified register and the Borrow from Accumulator. Place the Difference in the accumulator.
SUBB	A,addr	Subtract contents of memory location whose address is specified in the instruction, and the Borrow from Accumulator. Place the Difference in the accumulator.
SUBB	A,@Rp	Subtract contents of memory location whose address is specified in the register pointer, and the Borrow from Accumulator. Place the Difference in the accumulator.

Arithmetic instructions

Opcode	Operand	Operation
MUL	AB	Multiply A with B. Place lower order byte of the result in A and higher order byte in B
DIV	AB	Divide A by B. Place lower order byte of the result (quotient) in A and higher order byte (Remainder) in B
DA	A	Decimal adjust Accumulator after summing the two packed BCD numbers. (Only ADD and ADDC are adjusted to BCD after DA A)

Problem statement 1: WAP to add the data bytes stored in the external RAM locations 2501H and 2502H. Place the result in 2503 (Sum) and 2504H(Carry)

```
ORG 0000H
MOV DPTR,#2501H
MOVX A,@DPTR
MOV R0,A
INC DPTR
MOVX A,@DPTR
ADD A,R0
INC DPTR
MOVX @DPTR,A
MOV A,#00H
ADDC A,#00H
INC DPTR
MOVX @DPTR,A
END
```

```
ORG 0000H
MOV R1,#00H
MOV DPTR,#2501H
MOVX A,@DPTR
MOV R0,A
INC DPTR
MOVX A,@DPTR
ADD A,R0
INC DPTR
MOVX @DPTR,A
JNC AHEAD
INC R1
AHEAD: MOV A,R1
INC DPTR
MOVX @DPTR,A
END
```

Problem statement 2: WAP to add the data bytes stored in the internal RAM locations 25H and 26H. Place the result in 27H (Sum) and 28H(Carry)

```
ORG 0000H
MOV R0,#25H
MOV A,@R0
INC R0
ADD A,R0
INC R0
MOV @R0,A
MOV A,#00H
ADDC A,#00H
INC R0
MOV @R0,A
END
```

```
ORG 0000H
MOV R0,#25H
MOV R1,#00H
MOV A,@R0
INC R0
ADD A,R0
INC R0
MOV @R0,A
JNC AHEAD
INC R1
AHEAD: MOV A,R1
INC R0
MOV @R0,A
END
```

Problem statement 3: WAP to add the data bytes in a series stored in the external RAM locations starting at 2501H. Place the result in 2506H(Sum) and 2507H(Carry)

```
        ORG 0000H
        MOV DPTR,#2501H
        MOV R0,#00H
        MOV R1,#00H
        MOV R2,#05H
        L1: MOVX A,@DPTR
            ADD A,R0
            JNC AHEAD
        INC R1
AHEAD: MOV R0,A
        INC DPTR
        DJNZ R2,L1
        INC DPTR
        MOV A,R0
        MOVX @DPTR,A
        MOV A,R1
        INC DPTR
        MOVX @DPTR,A
        END
```

Problem statement 4: WAP to add the data bytes pairwise. A series of 5 pairs is stored in the internal RAM locations starting at 25H. Place the result in the same memory locations, i.e. the first data byte should be replaced by the Sum and second by the Carry

```
ORG 0000H
MOV R0,#25H
MOV R1,#05H
L1: MOV R2,#00H
    MOV A,@R0
    INC R0
    ADD A,@R0
   DEC R0
MOV @R0,A
INC R0
MOV A,#00H
ADDC A,#00H
MOV @R0,A
INC R0
DJNZ R1,L1
END
```

Problem statement 5: WAP to Multiply the data bytes pairwise. A series of 5 pairs is stored in the internal RAM locations starting at 25H. Place the result in the same memory locations, i.e. the first data byte should be replaced by the lower byte and second by the higher byte of the result of multiplication

```
ORG 0000H
MOV R0,#25H
MOV R1,#05H
L1: MOV A,@R0
    MOV B,A
    INC R0
    MOV A,@R0
   MUL AB
   DEC R0
   MOV @R0,A
   INC R0
   MOV A,B
   MOV @R0,A
   INC R0
   DJNZ R1,L1
   END
```


Thank
you