

Arithmetic Instructions

- **ADD Des, Src:**
 - It adds a byte to byte or a word to word.
 - It effects AF, CF, OF, PF, SF, ZF flags.
 - E.g.:
 - ADD AL, 74H
 - ADD DX, AX
 - ADD AX, [BX]

Arithmetic Instructions

- **ADC Des, Src:**
 - It adds the two operands with CF.
 - It effects AF, CF, OF, PF, SF, ZF flags.
 - E.g.:
 - ADC AL, 74H
 - ADC DX, AX
 - ADC AX, [BX]

Arithmetic Instructions

- **SUB Des, Src:**
 - It subtracts a byte from byte or a word from word.
 - It effects AF, CF, OF, PF, SF, ZF flags.
 - For subtraction, CF acts as borrow flag.
 - E.g.:
 - SUB AL, 74H
 - SUB DX, AX
 - SUB AX, [BX]

Arithmetic Instructions

- **SBB Des, Src:**
 - It subtracts the two operands and also the borrow from the result.
 - It effects AF, CF, OF, PF, SF, ZF flags.
 - E.g.:
 - SBB AL, 74H
 - SBB DX, AX
 - SBB AX, [BX]

Arithmetic Instructions

- **INC Src:**
 - It increments the byte or word by one.
 - The operand can be a register or memory location.
 - It effects AF, OF, PF, SF, ZF flags.
 - CF is not effected.
 - E.g.: INC AX

Arithmetic Instructions

- **DEC Src:**
 - It decrements the byte or word by one.
 - The operand can be a register or memory location.
 - It effects AF, OF, PF, SF, ZF flags.
 - CF is not effected.
 - E.g.: DEC AX

Arithmetic Instructions

- **AAA (ASCII Adjust after Addition):**
 - The data entered from the terminal is in ASCII format.
 - In ASCII, 0 – 9 are represented by 30H – 39H.
 - This instruction allows us to add the ASCII codes.
 - This instruction does not have any operand.
- **Other ASCII Instructions:**
 - **AAS** (ASCII Adjust after Subtraction)
 - **AAM** (ASCII Adjust after Multiplication)
 - **AAD** (ASCII Adjust Before Division)

Arithmetic Instructions

- **DAA (Decimal Adjust after Addition)**
 - It is used to make sure that the result of adding two BCD numbers is adjusted to be a correct BCD number.
 - It only works on AL register.
- **DAS (Decimal Adjust after Subtraction)**
 - It is used to make sure that the result of subtracting two BCD numbers is adjusted to be a correct BCD number.
 - It only works on AL register.

Arithmetic Instructions

- **NEG Src:**
 - It creates 2's complement of a given number.
 - That means, it changes the sign of a number.

Arithmetic Instructions

- **CMP Des, Src:**
 - It compares two specified bytes or words.
 - The Src and Des can be a constant, register or memory location.
 - Both operands cannot be a memory location at the same time.
 - The comparison is done simply by internally subtracting the source from destination.
 - The value of source and destination does not change, but the flags are modified to indicate the result.

Arithmetic Instructions

- **MUL Src:**

- It is an unsigned multiplication instruction.
- It multiplies two bytes to produce a word or two words to produce a double word.
- $AX = AL * Src$
- $DX : AX = AX * Src$
- This instruction assumes one of the operand in AL or AX.
- Src can be a register or memory location.

- **IMUL Src:**

- It is a signed multiplication instruction.

Arithmetic Instructions

- **DIV Src:**

- It is an unsigned division instruction.
- It divides word by byte or double word by word.
- The operand is stored in AX, divisor is Src and the result is stored as:
 - AH = remainder AL = quotient

- **IDIV Src:**

- It is a signed division instruction.

Arithmetic Instructions

- **CBW (Convert Byte to Word):**
 - This instruction converts byte in AL to word in AX.
 - The conversion is done by extending the sign bit of AL throughout AH.
- **CWD (Convert Word to Double Word):**
 - This instruction converts word in AX to double word in DX : AX.
 - The conversion is done by extending the sign bit of AX throughout DX.