Chapter 14

80x96 Family Microcontrollers

Lesson 7

80x96 Microcontroller Instruction Set

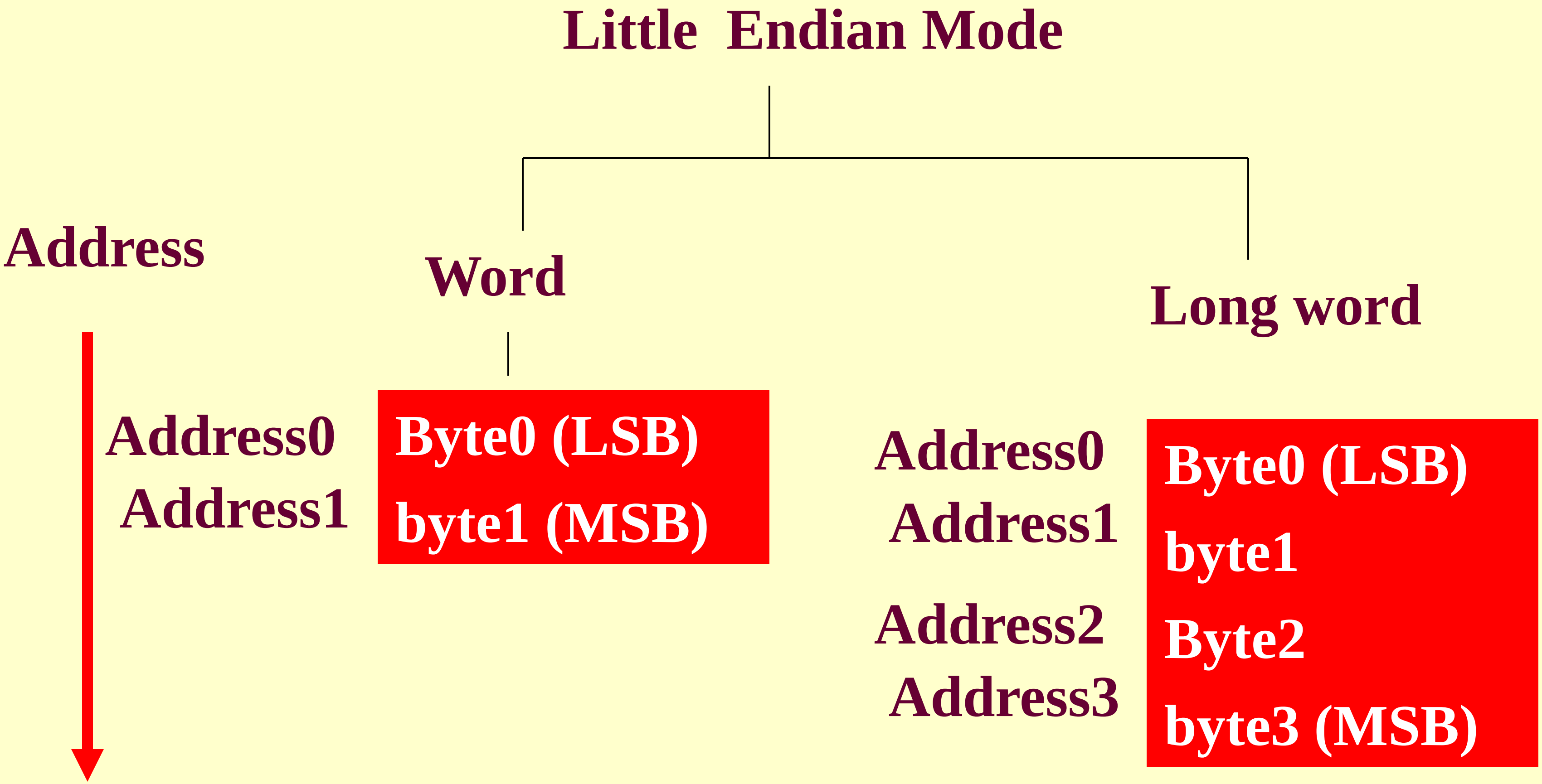
Basic Programming Feature and addressing Modes

Data Types support

- 8-bit byte
- 16-bit word
- 32-bit Long word

Word Alignment

- A word or long word alignment in memory is in little endian [least significant byte stored as lower bits (address 0) of a word]



Little Endian Mode

Address

Word

Long word

Address0	Byte0 (LSB)
Address1	byte1 (MSB)

Address0	Byte0 (LSB)
Address1	byte1
Address2	Byte2
Address3	byte3 (MSB)

Address0- even address; Address1- odd address

16-bit Word and 32-bit long word Alignment in Memory

Addressing Modes

Examples

Inherent → POPF, PUSHF (pop or push) flags on stack

Direct → 8-bit addresses in ADD 22H 20H 88H

Immediate → ADD WD, WS1, #200CH

Indexed → ADD WD, WS1, [WS2]

Add 16 bit words at WS1 with memory address pointed by WS2 and place result into WD

Add 16 bit words 20-21H with 88-89H and place result into 22-23H

Indexed (Indirect) Addressing mode-Examples of four ways

ADD WD, WS1, [WS2]

ADD WD, WS1, [WS2]+

ADD WD, WS1, WS2{offset}

ADD WD, WS1, dips [WS2]

Auto post increment to next word

Offset add inS2,WS2 no change later

WS2 changes. Add 8 bit displacement (-128 to +127) into WS2 then word at that address add with from WS1.

Basic Programming Features

- Memory addresses (for example, 1CH-24H) used as registers as there are no accumulator and index registers
- Only limited CPU registers (PSW, SP and PC)
- 1AH to FFH addresses used as registers/register file or RAM

Basic Programming Features-

- 16-bit un-segmented memory with device and system registers, RAM,ROM all 16-bit addresses
- 256B/512B address space has multiple V- windows can be addressed by 8-bit direct address in four options as per window selection

Basic Programming Features ..

- Devices/IO/System SFRs address space at page 0 (0000H-0019H) has multiple H- windows can be addressed by 8-bit direct address in four options H0-read,H0-write,H1 and H15 as per window selection

Data Transfer Instructions

Store/ MOV (Same Instructions)

- STYD, YS: Store a word from source operand to destined operand
- Six ways of specifying source operand- Four indirect addressing ways, immediate, direct

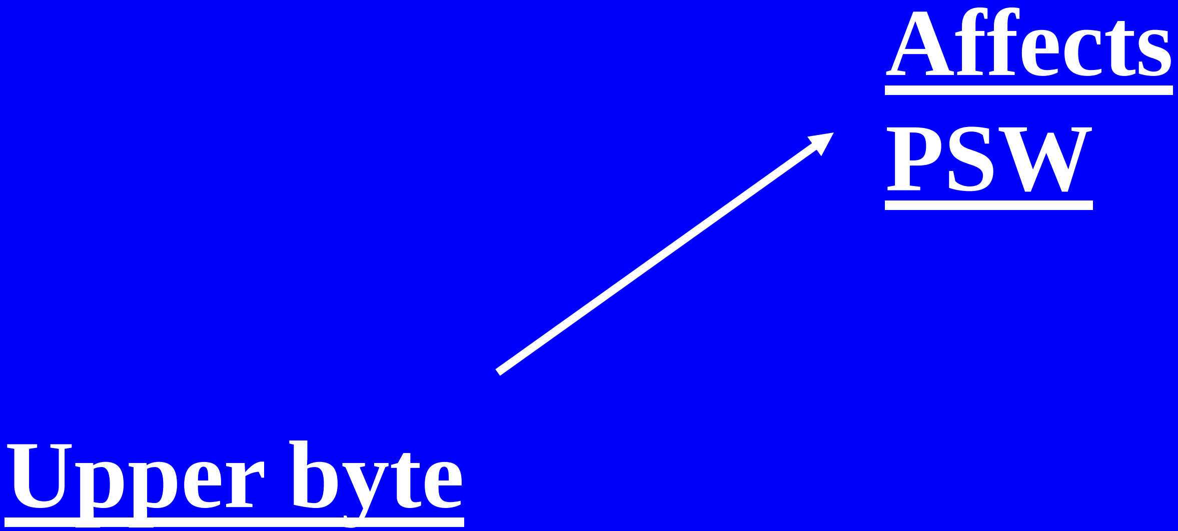
Push/POP Instructions

- PUSH YS: Push a word from source operand to stack
- POP YD: Push a word from stack to destination operand

Six ways of specifying source or destination operand - Four indirect addressing ways, immediate, direct

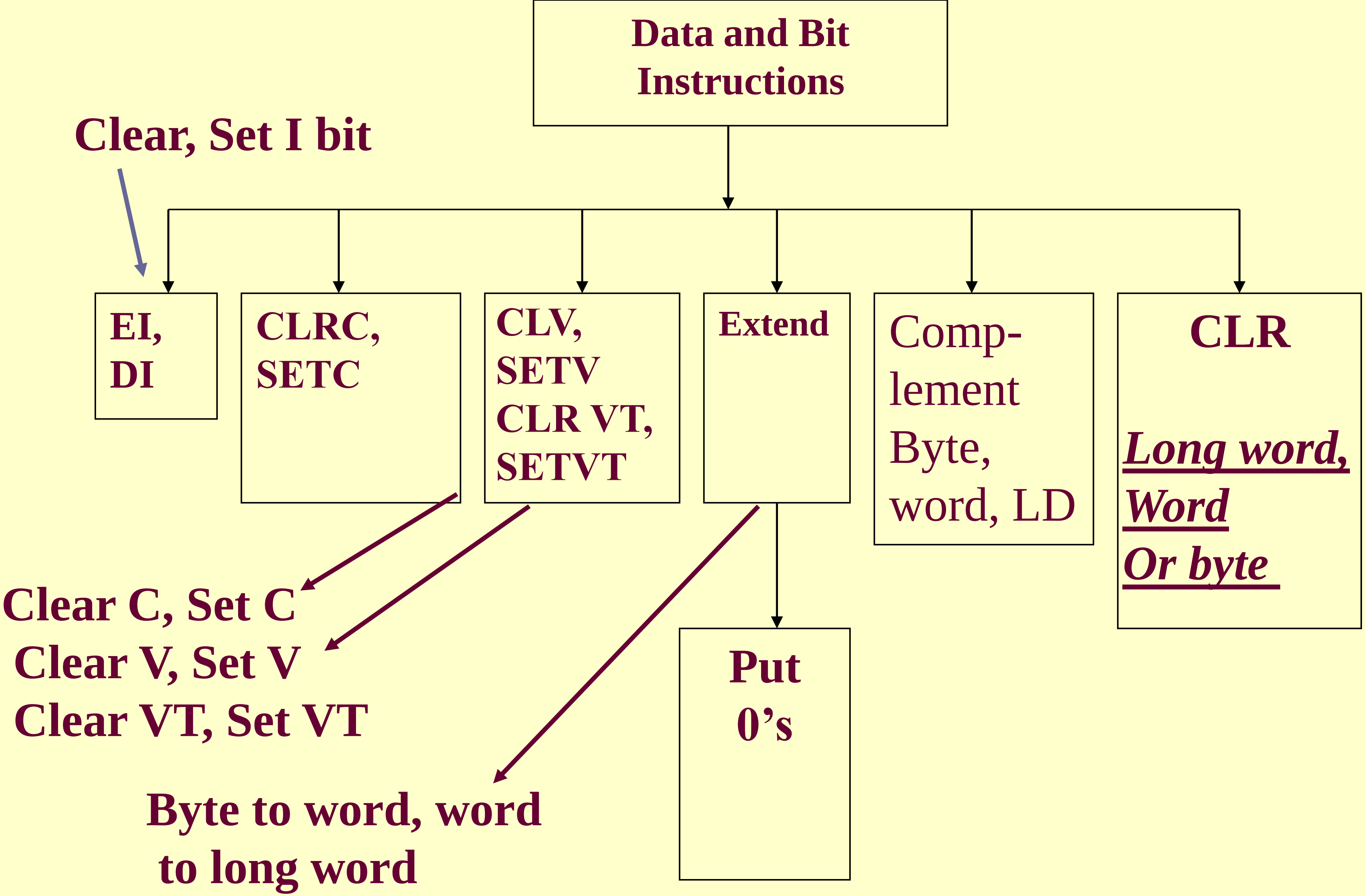
Data and Bit Manipulation Instructions

Data and Bit Manipulation of flags



Affects PSW

Upper byte



Data and Bit Instructions

Clear, Set I bit

EI, DI

CLRC, SETC

CLV, SETV CLR VT, SETVT

Extend

Comp-lement Byte, word, LD

CLR

<u>Long word, Word Or byte</u>

Clear C, Set C

Clear V, Set V

Clear VT, Set VT

Byte to word, word to long word

Put 0’s

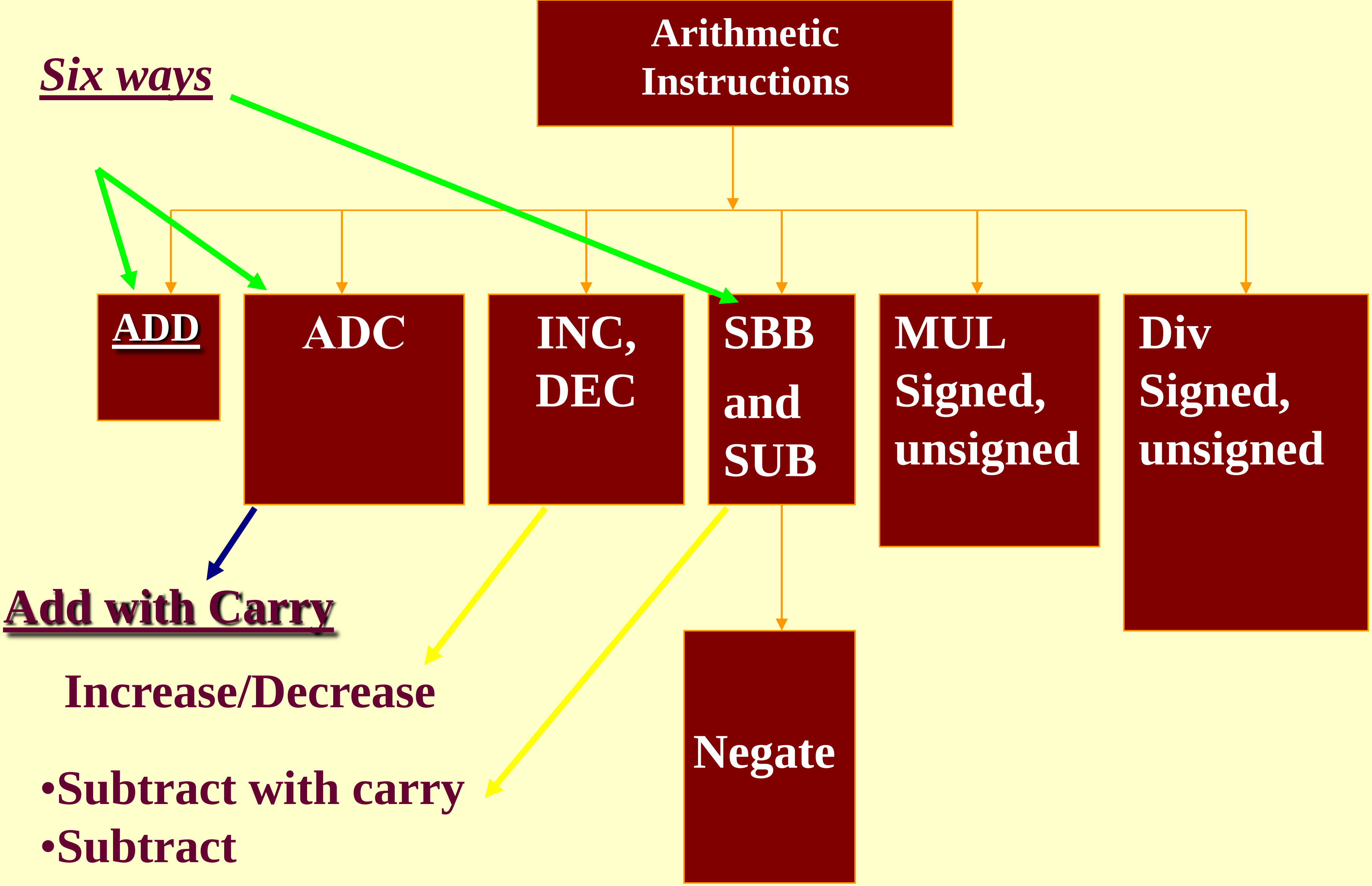
Arithmetic and Logic Instructions

Data Types support for ALU operations

- 8-bit byte
- 16-bit word

Addressing for ALU operations

- Direct address 8-bit



Arithmetic Instructions

ADD

ADC

INC, DEC

SBB and SUB

MUL Signed, unsigned

Div Signed, unsigned

Negate

Six ways

Add with Carry

Increase/Decrease

- **Subtract with carry**
- **Subtract**

Affects PSW-Hi

- Affects C in ADD, No borrow in SUB, NEG,INC, DEC, MUL

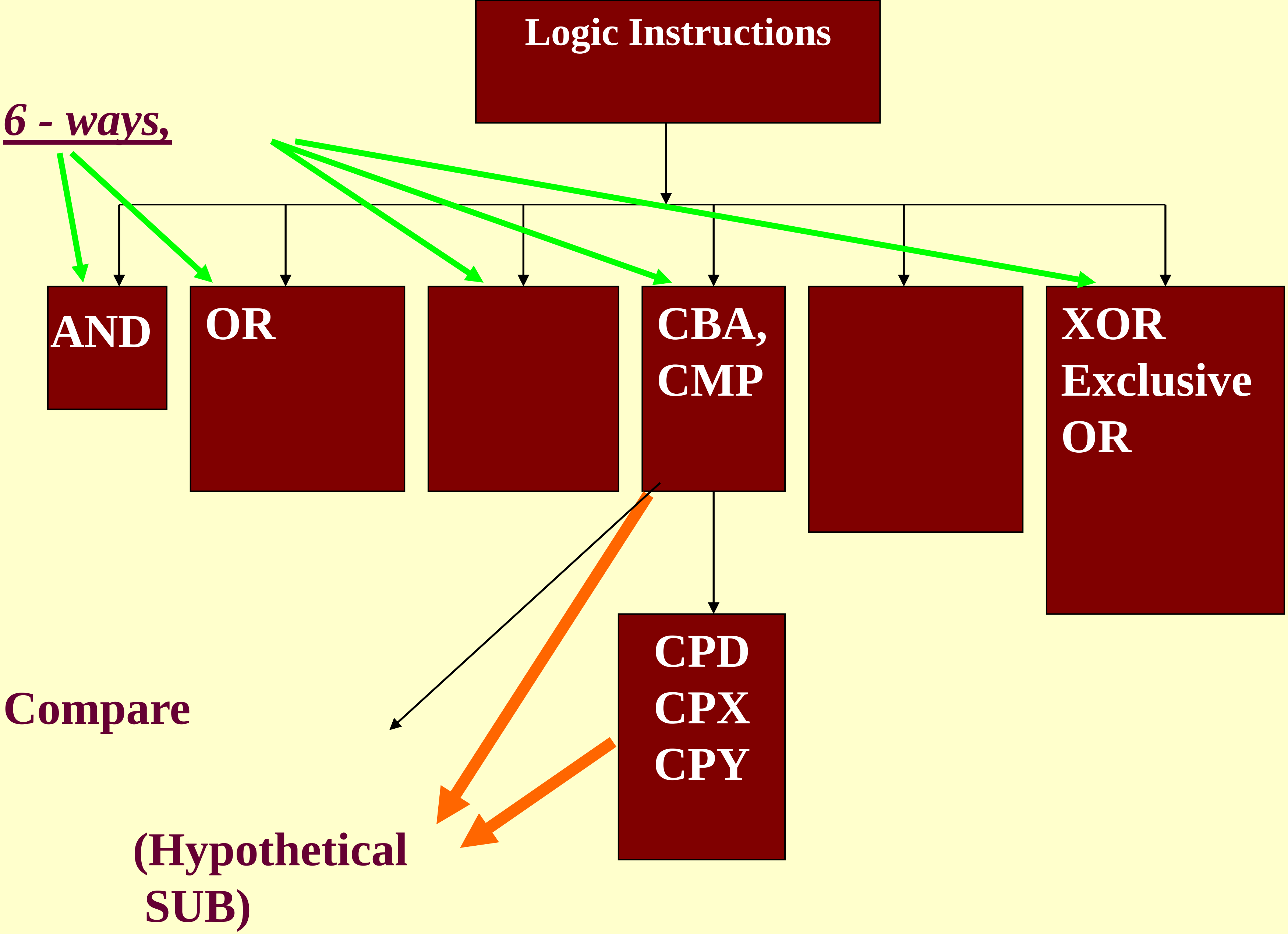
- Affects Z in INC, DEC, NEG,
- V sets if overflow in two’s complement operation

Arithmetic Instructions Using 6 ways

- ADC: Add with Carry, ADD:Add
- SBB: Subtract with carry , SUB: Subtract
- NEG, INC, DEC

MUL and DIV Instructions..

- MUL: Unsigned 16 × 16
- MUL: signed 16×16
- DIV: Unsigned 16 ×16
- DIV: signed 16 × 16



Logic Instructions

6 - ways,

AND

OR

CBA, CMP

XOR Exclusive OR

CPD CPX CPY

Compare

(Hypothetical SUB)

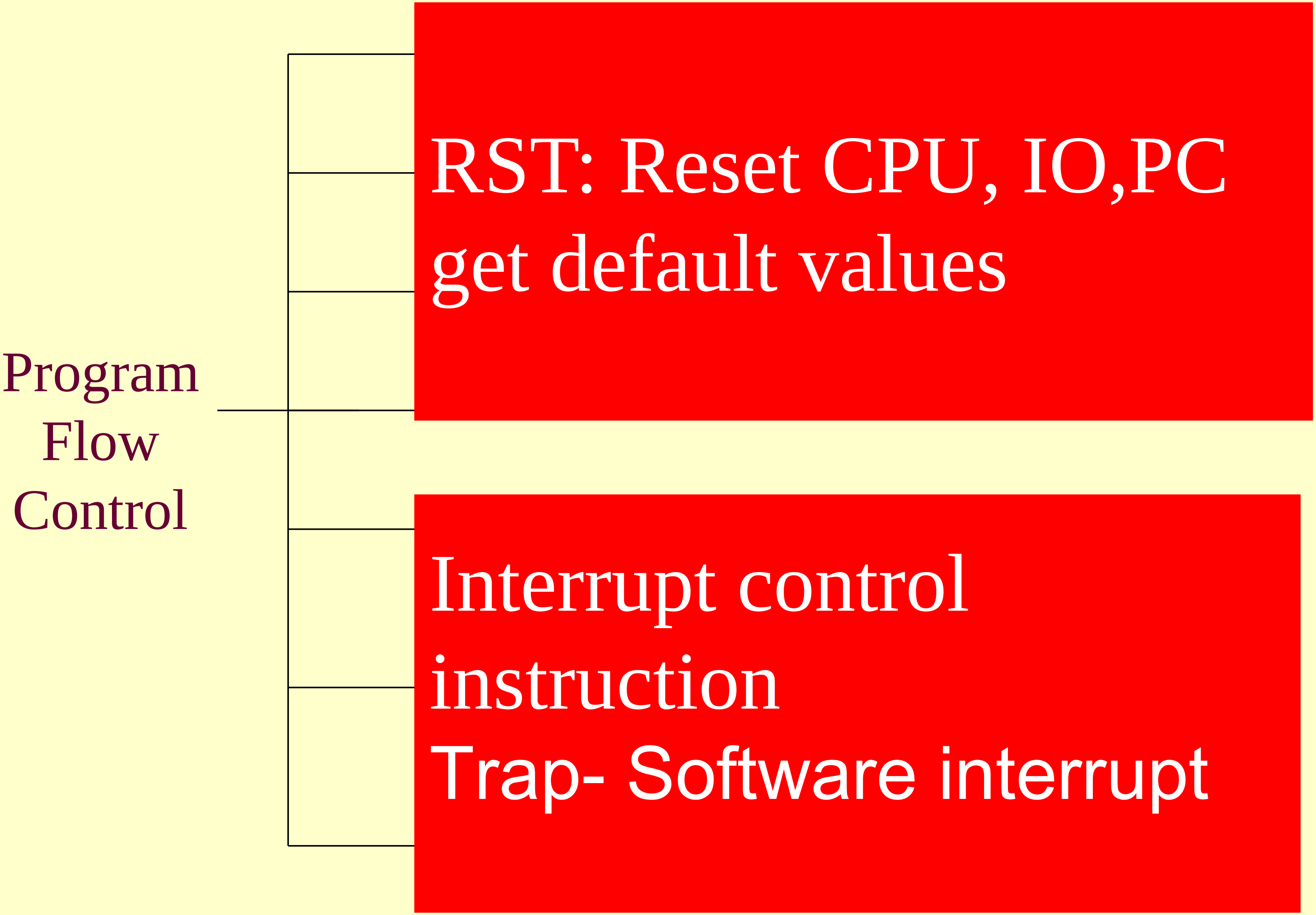
Program Flow Control and Interrupt Instructions

Program Flow Control

- Conditional Branch Instructions as per flag(s) conditions
- LJMP Label (signed 17bit), SJMP Rel: Unconditional Branch Branch to PC+Rel (-1024to +023)
- NOP: Branch PC+1, ←
- SKIP:Branch Next Branch to PC+2

Program Flow Control

- Unconditional call to subroutine SCALL Rel and LCALL label
- RET:Return from routine,Pop PC back from stack



Program Flow Control

RST: Reset CPU, IO,PC get default values

Interrupt control instruction
Trap- Software interrupt

Summary

We learnt

- Little endian 16-bit data
- 8-bit byte, 16-bit word and 16-bit long word data types

We learnt

Addressing Modes

- Inherent/Register
- Direct
- Immediate
- Index- Four ways of indirect addressing

We learnt

- Store, push and pop data transfer instructions
- Data bit manipulation,

We learnt

Instructions

- ADD,ADC
- SUB,SBB, NEG
- MUL and DIV
- INC, DEC
- EOR,OR, AND
- Compare

We learnt

Interrupt control instruction

- Trap

Program flow Instructions

- SJMP
- LJMP
- NOP
- SKIP

We learnt

Program flow Instructions

- SCALL
- LCALL
- RET

End of Lesson 4 on 80x96 Microcontroller Instruction Set