Simple Instructions 1

Operation MOV (move)
- Word (16 bit) operation
  - MOV.W #0x45ac, R10
  - Copy 0x45ac into core register R10
  - Operands
- Byte (8 bit) operation
  - MOV.B #0x45, R10

Immediate mode, operand is 0x45ac
- #0x45ac
- Operand
- R10

Register mode, operand is the content of R10

ADD.W #0x2ABF R10
- Word operation
  - Add 0x2ABF to content of R10 and store result in R10

ADD.B #0x45, R10
- Byte operation

ADD.B R9, R10
- Add content of R9 to content of R10 and store the result in R10
Simple Instructions 2

- **INC.W R10**
  - Word (16 bit) operation
- **INC.B R10**
  - Byte (8 bit) operation

- **R10**
  -Operand is what is inside R10
  -Register mode, operand is the content of R10

- **DEC.W R10**
  - Word operation

- **DEC.B R10**
  - Byte operation
The Preprocessor

Preprocessor is a program that runs before the compiler.

It runs automatically when you build a project in CCS

Build -> Preprocessor + Compiler + Linker

Assembly Language -> Preprocessor -> Assembly Language

Beside instructions your program also contains:
- directives for the preprocessor
- .text
- .data
- .word
- .space
- .set
- etc

Modified
Program contains only executable instructions
- directives to the preprocessor are missing
Put whatever follows in RAM can be changed at runtime.

Set aside space for 4 words array of four words

Global variable array

Var1
var1+2
var1+4
var1+6

Global variable array initialized to zero

Set aside space for 6 bytes or 3 words

Symbolic constant

scon1 is a symbolic constant the preprocessor changes every occurrence of "scon1" in your program by 0x500.

No memory is allocated for scon1 in RAM or ROM.
place whatever follows in FLASH (ROM) can NOT be changed at run time

Set aside space for one word in ROM

con3 is an address

GLOBAL CONSTANT

Set aside space for two words in ROM

con4 is an address

GLOBAL CONSTANT ARRAY

Program instructions go into ROM

ECE2560 Lesson06 Page 5

mov.w #STACK_END, SP
StopWDT mov.w #WDTPW|WDTHOLD, &WDTCTL

mov.w &var1, R10
; copy content of address var1 to core register R10

mov.w &var1+2, R10
; copy content of address (var1+2) to core register R10

var1 is an address
&var1 means contents of the address var1

&var1 + 2 means contents of the address (var1 + 2)

Absolute Mode

Comments are ignored by the compiler
.cdecls C,LIST,"msp430.h"

;-------------------------------------------------------------------------------
.data
    var1: .word  0x1, 0x2, 0x3, 0x4
    var2: .space  6
    scon1: .set  0x500

;-------------------------------------------------------------------------------
.text
    con3: .word  0x400
    con4: .word  0x2, 0x4

;-------------------------------------------------------------------------------
RESET
    mov.w #__STACK_END,SP
    StopWDT  mov.w #WDTPW|WDTHOLD,&WDTCTL

mov.w   &var1,    R10
mov.w   &var1+2, R10

dec.w   &var1
inc.w   &var2+4

mov.w   &var1, &var2+4
mov.w   #scon1, &var1+2

dec.w   &con3
add.w   &con4, &var2

Loop:    jmp    Loop

;-------------------------------------------------------------------------------
.global __STACK_END
.sect .stack

;-------------------------------------------------------------------------------
.sect ".reset"
.short  RESET