

## ConvertCtoF

Write a subroutine that convert a temperature reading in Celsius (Variable C) to Fahrenheit (variable F).

Using Cmax value (124)

The screenshot shows the AVR Processor window with the following assembly code:

```
.INCLUDE <m328pdef.inc>

.DSEG
    C: .BYTE 1
    F: .BYTE 1

.CSEG

.DEF Denominator=R19
.DEF Quotient=R22

.ORG 0x0000

//Input a celsius value into r17
TestConvertCtoF:
    ldi r17,124
    sts C, R17
    //call subroutine ConvertCtoF
    rcall ConvertCtoF
    rjmp TestConvertCtoF

/*****
 * subroutine converts a temperature reading in Celsius (variable C) to Fahrenheit (variable F).
 * F=(C × 9/5) + 32 ==(C × 18/10) + 32
 * Range for C input is from (0 to 124), since F max is 255
 *****/

ConvertCtoF:
    //Load the C value into r18
    lds r18, C
    //Input the constant value 18 in reg. 16
    ldi r16,18
    //This part calculates (C*18)
    mul r18, r16
    //Move the products into r25H and r24L
    movw r25:r24, r1:r0
    //Input the denominator into r19
    ldi Denominator, 10 //r19,10
    //Call the 16 bit by 8 bit division
    rcall Div16_8
    // add 32 to the quotient (18*C)/10 + 32
    ldi r26,32
    add Quotient,r26 //r22,r26
    adc r23,r26
    //Store the answer into F
    sts F, r22
    ret
```

The Watch window shows the following data:

Name	Value	Type	Location
C	124 'i'	SRAM Locat:	0x0100 [SRAM]
F	255 'y'	SRAM Locat:	0x0101 [SRAM]
R24	2 'i'	Register	R24

From the calculation 124 degree Celsius = 255.2 degree Fahrenheit.

Using Cmin Value (0)

The screenshot shows an AVR assembly editor with the following components:

- Registers Window:** Lists registers R00 through R29. R00-R03 are 0x00. R04-R07 are 0x00. R08-R12 are 0x00. R13-R15 are 0x00. R16 is 0x12. R17 is 0x00. R18 is 0x00. R19 is 0x0A. R20 is 0x00. R21 is 0x00. R22 is 0x20. R23 is 0xFF. R24 is 0x00. R25 is 0x00. R26 is 0x20. R27 is 0x00. R28 is 0x00. R29 is 0x00.
- Code Editor:**

```

.INCLUDE <m328pdef.inc>
.DSEG
    C: .BYTE 1
    F: .BYTE 1
.CSEG
.DEF Denominator=R19
.DEF Quotient=R22
.ORG 0x0000

//Input a celsius value into r17
TestConvertCtoF:
    ldi r17,0
    sts C, R17
    //call subroutine ConvertCtoF
    rcall ConvertCtoF
    rjmp TestConvertCtoF

/*****
 * subroutine converts a temperature reading in Celsius (variable C) to Fahrenheit (variable F).
 * F=(C x 9/5) + 32 == (C x 18/10) + 32
 * Range for C input is from (0 to 124), since F max is 255
 *****/

ConvertCtoF:
    //Load the C value into r18
    lds r18, C
    //Input the constant value 18 in reg. 16
    ldi r16,18
    //This part calculates (C*18)
    mul r18, r16
    //Move the products into r25H and r24L
    movw r25:r24, r1:r0
    //Input the denominator into r19
    ldi Denominator, 10 //r19,10
    //Call the 16 bit by 8 bit division
    rcall Div16_8
    // add 32 to the quotient (18*C)/10 + 32
    ldi r26,32
    add Quotient,r26 //r22,r26
    adc r23,r2

    //Store the answer into F
    sts F, r22
    ret
        
```
- Watch Window:**

Name	Value	Type	Location
C	0	SRAM Locat: 0x0100 [SRAM]	
F	32	SRAM Locat: 0x0101 [SRAM]	
R24	0	Register R24	

From the calculation 0 degree Celsius = 32 degree Fahrenheit.

C random value (74)

The screenshot shows an AVR simulator interface. On the left, the 'Registers' window displays the state of various registers, with R17 containing 0x4A and R24 containing 0x02. The main window shows assembly code for a Celsius-to-Fahrenheit conversion. The code includes comments and assembly instructions. A 'Watch' window is open, showing the values of variables C (74), F (165.2), and R24 (2).

```

.INCLUDE <m328pdef.inc>

.DSEG
    C: .BYTE 1
    F: .BYTE 1

.CSEG

.DEF Denominator=R19
.DEF Quotient=R22

.ORG 0x0000

//Input a celsius value into r17
TestConvertCtoF:
    ldi r17,74
    sts C, R17
    //call subroutine ConvertCtoF
    rcall ConvertCtoF
    rjmp TestConvertCtoF

/*****
 * subroutine converts a temperature reading in Celsius (variable C) to Fahrenheit (variable F).
 * F=(C x 9/5) + 32 == (C x 18/10) + 32
 * Range for C input is from (0 to 124), since F max is 255
 *****/

ConvertCtoF:
    //Load the C value into r18
    lds r18, C
    //Input the constant value 18 in reg. 16
    ldi r16,18
    //This part calculates (C*18)
    mul r18, r16
    //Move the products into r25H and r24L
    movw r25:r24, r1:r0
    //Input the denominator into r19
    ldi Denominator, 10 //r19,10
    //Call the 16 bit by 8 bit division
    rcall Div16_8
    // add 32 to the quotient (18*C)/10 + 32
    ldi r26,32
    add Quotient,r26 //r22,r26
    adc r23,r2

    //Store the answer into F
    sts F, r22
    ret
    
```

The 'Watch' window shows the following data:

Name	Value	Type	Location
C	74 'J'	SRAM Locat: 0x0100 [SRAM]	
F	165.2 'Y'	SRAM Locat: 0x0101 [SRAM]	
R24	2 'j'	Register R24	

From the calculation 74 degree Celsius = 165.2 degree Fahrenheit.