Example 10

4

PIC16F887 Microcontroller ◆ Example 10

```
 'Header*******************************************************
program example_10              ' Program name

dim LCD_RS as sbit  at RB4_bit              ' Lcd module connections
    LCD_EN as sbit  at RB5_bit
    LCD_D4 as sbit  at RB0_bit
    LCD_D5 as sbit  at RB1_bit
    LCD_D6 as sbit  at RB2_bit
    LCD_D7 as sbit  at RB3_bit

    LCD_RS_Direction as sbit at TRISB4_bit
    LCD_EN_Direction as sbit at TRISB5_bit
    LCD_D4_Direction as sbit at TRISB0_bit
    LCD_D5_Direction as sbit at TRISB1_bit
    LCD_D6_Direction as sbit at TRISB2_bit
    LCD_D7_Direction as sbit at TRISB3_bit  ' End Lcd module connections

dim text as string [16]      ' Variable text is of string type
dim ch, adc_rd as word       ' Variables ch and adc_rd are of word type
dim tlong as longword        ' Variable tlong is of longword type

main:                        ' Start of program
  TRISB = 0                  ' All port PORTB pins are configured as outputs
  PORTB = 0xFF
  INTCON = 0                 ' All interrupts disabled
  ANSEL  = 0x04              ' Pin RA2 is configured as an analog input
  TRISA  = 0x04
  ANSELH = 0                 ' Rest of pins is configured as digital
  Lcd_Init()                 ' LCD display initialization
  Lcd_Cmd(_LCD_CURSOR_OFF)   ' LCD command (cursor off)
  Lcd_Cmd(_LCD_CLEAR)        ' LCD command (clear LCD)

  text = "mikroElektronika"  ' Define the first message
  Lcd_Out(1,1,text)          ' Write the first message in the first line
  text = "LCD example"       ' Define the second message
  Lcd_Out(2,1,text)          ' Write the second message in the second line

  ADCON1      = 0x80         ' A/D voltage reference is VCC
  TRISA       = 0xFF         ' All PORTA pins are configured as inputs
  Delay_ms(2000)
  text  = "Voltage="         ' Define the third message

  while 1                    ' Endless loop
    adc_rd  = ADC_Read(2)    ' A/D conversion. Pin RA2 is an input.
    Lcd_Out(2,1,text)        ' Write result in the second line

    tlong = adc_rd * 5000    ' Convert the result in millivolts
    tlong = tlong / 1023     ' 0..1023 -> 0-5000mV

    ch = (tlong / 1000) mod 10       ' Extract volts (thousands of millivolts)
                                     ' from result
    Lcd_Chr(2,9,48+ch)               ' Write result in ASCII format

    Lcd_Chr_CP(".")                  ' Write the decimal pint

    ch = (tlong / 100) mod 10        ' Extract hundreds of millivolts
    Lcd_Chr_CP(48+ch)                ' Write result in ASCII format

    ch = (tlong / 10) mod 10         ' Extract tens of millivolts
    Lcd_Chr_CP(48+ch)                ' Write result in ASCII format

    ch = tlong mod 10                ' Extract digits for millivolts
    Lcd_Chr_CP(48+ch)                ' Write result in ASCII format

    Lcd_Chr_CP("V")                  ' Write a mark for voltage "V"

    Delay_ms(1)                      ' 1mS delay
  wend

      end.                   ' End of program
```