

## Malá záhada Bascomu -

---

Napsal/a: Žirafka

Datum zveřejnění: : 27. 06. 2015 v 7:14

Před časem jsem získala, za velmi dobrou cenu, LCD modul v konfiguraci 4 řádky po 40 znacích. Vlastně největší běžně dostupný displej, co se dá koupit.

Jeho zprovoznění nebylo tak jednoduché, jak jsem si původně myslela, ale nakonec se to povedlo a displej začal hezky komunikovat a zobrazovat vše tak, jak měl.

Během zkoušení jsem stvořila tento jednoduchý program a právě v tom je ta malá záhada. Ačkoli asi vím, jak záhada vzniklá, na první pohled to vypadá divně a trvalo mi docela dlouho přijít na to, co je „špatně“ ačkoli toto slovo vlastně není správné.

```
-----  
----- Nastaveni prekladace -----  
-----  
  
'$sim                ' Odremovat pro zkouseni na simulatoru, jinak nechat takto  
$regfile = "ATtiny2313.dat" ' pouzity procesor  
$crystal = 8000000        ' frekvence oscilatoru 8MHz (nutno nastavit v programatoru)  
$baud = 1200              ' baud rate we want  
$hwstack = 32  
$swstack = 8  
$framesize = 40  
  
$lib "lcd4e2.lbx"        ' use the alternative library  
  
Dim __lcd_e As Byte  
Dim X As Byte  
Dim Y As Byte  
Dim A As Byte  
  
'in order for simulation to work correct, you need to specify the used pins  
'for lcd4.lbx, the pins are fixed  
'Rs = PortB.0  
'RW = PortB.1          we dont use the R/W option of the LCD in this version so connect to ground  
' E = PortB.2  
'E2 = PortB.3          optional for lcd with 2 chips  
'Db4 = PortB.4         the data bits must be in a nibble to save code  
'Db5 = PortB.5  
'Db6 = PortB.6  
'Db7 = PortB.7  
  
Config Lcdpin = Pin , Rs = Portb.0 , E = Portb.2 , E2 = Portb.3 , Db4 = Portb.4 , Db5 =  
Portb.5 , Db6 = Portb.6 , Db7 = Postb.7  
  
Config Lcd = 40 * 4      ' Konfigurace LCD modulu  
  
__lcd_e = 0              ' prepnuti na horni dva radky
```

```
Cls                ' smazani displeje
Cursor Off        ' a vypnuti kurzoru
__lcde = 1        ' to same pro spodni radky
Cls
Cursor Off

Do                ' hlavni smycka

  X = Rnd(40) + 1  ' sloupec
  Y = Rnd(4) + 1   ' radek
  A = Rnd(95) + 32 ' nahodne pismenko

  If Y = 1 Or Y = 2 Then ' pokud jsem na radku 1 nebo 2
    __lcde = 0          ' obslouzim horni radic
  Else                 ' kdyz ne, tak spodni radic
    __lcde = 1
    Y = Y - 2
  End If

  Locate Y , X
  Lcd Chr(a)

  Waitms 200

Loop                ' a stale dokolecka dokola

End
```

Program je v pořádku, dá se přeložit a v procesoru správně funguje. Tady ta záhada není. Otázka tedy zní: Co program má dělat a co program dělá ve skutečnosti? Respektive, jak vypadá výstup na displeji?

Kdo správně odpoví, zaslouží si dobrý pocit z toho, že na to přišel 😊 Já odpověď pochopitelně vím, takže se nezúčastním. Přesně vysvětlení ale nemám, ačkoli to, na které jsem přišla, je asi nejpravděpodobnější.