
1 / 8

2 / 8

```

pocatecnich nul
lcd_instr(0x01);wait_ms(3)
#define cursor_off()    lcd_instr(0x0c)
#define cursor_on()    lcd_instr(0x0e)
#define cursor_blink() lcd_instr(0x0f)
#define shift_lcd_left() lcd_instr(0x18)
#define shift_lcd_right() lcd_instr(0x1c)
#define gotoxy(lin,col) lcd_instr(64*((lin)+1)+((col)-1))
//jdi na radek (lin),
// sloupec (col) //napis bajt (tri cifry, znak = znak
misto uvodnich nul)
void
lcd_num3(unsigned int num,unsigned char znak) {
    unsigned char i;
    num%=1000;
    //oriznuti desetitisicu
    i=num/100;
    //i=stovky
    if(i==0)
        //pokud je cislo<100
        lcd_char(znak);
        //zobraz uvodni znak
    else
        //jinak
        {
            lcd_char(0x30);
            +i;
            //zobraz stovky (0x30 je v ascii
            tabulce nula)
            znak=('0');
            //cislo je triciferni dalsi nula jiz neni uvodni
        }
    lcd_num2(num,znak);
    //vypis zbyvajici 2 cislice
    //napis cislo (dve cifry, znak = znak misto uvodnich nul)

```

```

</span><span class="cpp1-reservedword">void</span><span class="cpp1-space">
</span><span class="cpp1-identifier">lcd_num2(</span><span
class="cpp1-reservedword">unsigned</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">char</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">num,</span><span class="cpp1-reservedword">unsigned</span><span
class="cpp1-space"> </span><span class="cpp1-reservedword">char</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">znak) { </span><span
class="cpp1-reservedword">unsigned</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">char</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">i; num%=</span><span class="cpp1-number">100</span><span
class="cpp1-symbol">;</span><span class="cpp1-space"> </span><span
class="cpp1-comment">//oriznuti tisicu </span><span
class="cpp1-identifier">i=num/</span><span class="cpp1-number">10</span><span
class="cpp1-symbol">;</span><span class="cpp1-space"> </span><span
class="cpp1-comment">//i=desitky </span><span class="cpp1-reservedword">if</span><span
class="cpp1-brackets">>(i==</span><span class="cpp1-number">0</span><span
class="cpp1-brackets">></span><span class="cpp1-space"> </span><span
class="cpp1-comment">//pokud je cislo<10 </span><span class="cpp1-space">
</span><span class="cpp1-identifier">lcd_char(znak);</span><span class="cpp1-space">
</span><span class="cpp1-comment">//zobraz uvodni znak </span><span
class="cpp1-reservedword">else</span><span class="cpp1-space"> </span><span
class="cpp1-comment">//jinak </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">lcd_char(</span><span class="cpp1-hexadecimal">0x30</span><span
class="cpp1-symbol">+i);</span><span class="cpp1-comment">//zobraz desitky
</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">lcd_char(</span><span class="cpp1-hexadecimal">0x30</span><span
class="cpp1-symbol">+num%</span><span class="cpp1-number">10</span><span
class="cpp1-brackets">);</span><span class="cpp1-comment">//zobraz jednotky
</span><span class="cpp1-brackets">} </span><span class="cpp1-comment">//napis string,
umisteny v RAM </span><span class="cpp1-reservedword">void</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">lcd_text(</span><span
class="cpp1-reservedword">char</span><span class="cpp1-symbol">*</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">text) { </span><span
class="cpp1-reservedword">unsigned</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">char</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">i=</span><span class="cpp1-number">0</span><span
class="cpp1-symbol">;</span><span class="cpp1-reservedword">do </span><span
class="cpp1-space"> </span><span class="cpp1-brackets">{ </span><span
class="cpp1-space"> </span><span class="cpp1-identifier">temp</span><span
class="cpp1-space"> </span><span class="cpp1-symbol">=</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">text[i]; </span><span
class="cpp1-space"> </span><span class="cpp1-reservedword">if</span><span
class="cpp1-brackets">>(temp==</span><span class="cpp1-number">0</span><span
class="cpp1-brackets">></span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">break</span><span class="cpp1-symbol">;</span><span
class="cpp1-space"> </span><span class="cpp1-comment">//kazdy string je ukonceny

```

```

nulou </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">lcd_char(temp); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">i++; </span><span class="cpp1-space"> </span><span
class="cpp1-brackets">} </span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">while</span><span class="cpp1-brackets">(temp</span><span
class="cpp1-number">0</span><span class="cpp1-brackets">); } </span><span
class="cpp1-comment">//napis string, umisteny ve FLASH </span><span
class="cpp1-reservedword">void</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">lcd_ftext(</span><span
class="cpp1-reservedword">const</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">char</span><span class="cpp1-symbol">*</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">text) { </span><span
class="cpp1-reservedword">unsigned</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">char</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">i=</span><span class="cpp1-number">0</span><span
class="cpp1-symbol">,</span>temp; </span><span class="cpp1-reservedword">do </span><span
class="cpp1-space"> </span><span class="cpp1-brackets">{ </span><span
class="cpp1-space"> </span><span class="cpp1-identifier">temp</span><span
class="cpp1-space"> </span><span
class="cpp1-symbol">=pgm_read_byte(text+i);</span><span class="cpp1-space">
</span><span class="cpp1-comment">//cti bajt z pameti FLASH </span><span
class="cpp1-space"> </span><span class="cpp1-reservedword">if</span><span
class="cpp1-brackets">(temp==</span><span class="cpp1-number">0</span><span
class="cpp1-brackets">)</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">break</span><span class="cpp1-symbol">; </span><span
class="cpp1-space"> </span><span class="cpp1-identifier">lcd_char(temp); </span><span
class="cpp1-space"> </span><span class="cpp1-identifier">i++; </span><span
class="cpp1-space"> </span><span class="cpp1-brackets">} </span><span
class="cpp1-space"> </span><span class="cpp1-reservedword">while</span><span
class="cpp1-brackets">(temp</span><span class="cpp1-number">0</span><span
class="cpp1-brackets">); } </span><span class="cpp1-reservedword">void</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">len(</span><span
class="cpp1-reservedword">void</span><span class="cpp1-brackets">)</span><span
class="cpp1-space"> </span><span class="cpp1-comment">//impuls Enable do
LCD </span><span class="cpp1-brackets">{ </span><span class="cpp1-space">
</span><span class="cpp1-identifier">setb(PORT_EN,EN); </span><span
class="cpp1-space"> </span><span class="cpp1-identifier">wait_us(</span><span
class="cpp1-number">1</span><span class="cpp1-brackets">); </span><span
class="cpp1-space"> </span><span class="cpp1-identifier">clrb(PORT_EN,EN);
</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">wait_us(</span><span class="cpp1-number">50</span><span
class="cpp1-brackets">); } </span><span class="cpp1-reservedword">void</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">write(</span><span
class="cpp1-reservedword">char</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">c) { </span><span class="cpp1-reservedword">char</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">temp; </span><span

```

```

class="cpp1-space"> </span><span class="cpp1-identifier">temp</span><span
class="cpp1-space"> </span><span class="cpp1-symbol">=</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">c</span><span
class="cpp1-space"> </span><span class="cpp1-symbol">&</span><span
class="cpp1-space"> </span><span class="cpp1-hexadecimal">0xf0</span><span
class="cpp1-symbol">;</span><span class="cpp1-space"> </span><span
class="cpp1-comment">//vymaskovani hornich cttyrech bitu </span><span
class="cpp1-space"> </span><span class="cpp1-comment">//vystaveni hornich cttyrech bitu na
spravne piny: </span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">if</span><span class="cpp1-brackets">(temp&</span><span
class="cpp1-number">1</span><span class="cpp1-symbol"><<<</span><span
class="cpp1-number">4</span><span class="cpp1-brackets">))</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">setb(PORT_D4,D4);</span><span
class="cpp1-space"> </span><span class="cpp1-reservedword">else</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">clrb(PORT_D4,D4);</span><span
class="cpp1-space"> </span><span class="cpp1-reservedword">if</span><span
class="cpp1-brackets">(temp&</span><span class="cpp1-number">1</span><span
class="cpp1-symbol"><<<</span><span class="cpp1-number">5</span><span
class="cpp1-brackets">))</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">setb(PORT_D5,D5);</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">else</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">clrb(PORT_D5,D5); </span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">if</span><span class="cpp1-brackets">(temp&</span><span
class="cpp1-number">1</span><span class="cpp1-symbol"><<<</span><span
class="cpp1-number">6</span><span class="cpp1-brackets">))</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">setb(PORT_D6,D6);</span><span
class="cpp1-space"> </span><span class="cpp1-reservedword">else</span><span
class="cpp1-space"> </span><span class="cpp1-identifier">clrb(PORT_D6,D6); </span><span
class="cpp1-space"> </span><span class="cpp1-reservedword">if</span><span
class="cpp1-brackets">(temp&</span><span class="cpp1-number">1</span><span
class="cpp1-symbol"><<<</span><span class="cpp1-number">7</span><span
class="cpp1-brackets">))</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">setb(PORT_D7,D7);</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">else</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">clrb(PORT_D7,D7); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">len();</span><span class="cpp1-space"> </span><span
class="cpp1-comment">//potvrdit (pulz EN) </span><span class="cpp1-space">
</span><span class="cpp1-identifier">temp</span><span class="cpp1-space"> </span><span
class="cpp1-symbol">=</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">c</span><span class="cpp1-space"> </span><span
class="cpp1-symbol">&</span><span class="cpp1-space"> </span><span
class="cpp1-hexadecimal">0x0f</span><span class="cpp1-symbol">;</span><span
class="cpp1-comment">//vymaskovani dolnich cttyrech bitu </span><span class="cpp1-space">
</span><span class="cpp1-identifier">temp</span><span class="cpp1-space"> </span><span
class="cpp1-symbol"><<<<=</span><span class="cpp1-space"> </span><span
class="cpp1-number">4</span><span class="cpp1-symbol">;</span><span>

```

```

class="cpp1-space"> </span><span class="cpp1-comment">//posunuti dolnich 4 bitu na misto
hornich 4 bitu </span><span class="cpp1-space"> </span><span
class="cpp1-comment">//vystaveni dolni poloviny bytu na spravne piny </span><span
class="cpp1-space"> </span><span class="cpp1-reservedword">if</span><span
class="cpp1-brackets">(temp&</span><span class="cpp1-number">1</span><span><span
class="cpp1-symbol"><<</span><span class="cpp1-number">4</span><span><span
class="cpp1-brackets">))</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">setb(PORT_D4,D4);</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">else</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">clrb(PORT_D4,D4);</span><span class="cpp1-space">
</span><span class="cpp1-reservedword">if</span><span><span
class="cpp1-brackets">(temp&</span><span class="cpp1-number">1</span><span><span
class="cpp1-symbol"><<</span><span class="cpp1-number">5</span><span><span
class="cpp1-brackets">))</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">setb(PORT_D5,D5);</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">else</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">clrb(PORT_D5,D5); </span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">if</span><span><span
class="cpp1-brackets">(temp&</span><span class="cpp1-number">1</span><span><span
class="cpp1-symbol"><<</span><span class="cpp1-number">6</span><span><span
class="cpp1-brackets">))</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">setb(PORT_D6,D6);</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">else</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">clrb(PORT_D6,D6); </span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">if</span><span><span
class="cpp1-brackets">(temp&</span><span class="cpp1-number">1</span><span><span
class="cpp1-symbol"><<</span><span class="cpp1-number">7</span><span><span
class="cpp1-brackets">))</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">setb(PORT_D7,D7);</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">else</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">clrb(PORT_D7,D7); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">len();</span><span class="cpp1-space"> </span><span
class="cpp1-comment">//potvrdit (pulz EN) </span><span class="cpp1-space">
</span><span class="cpp1-brackets">} </span><span class="cpp1-comment">// Napis znak
na LCD: </span><span class="cpp1-reservedword">void</span><span class="cpp1-space">
</span><span class="cpp1-identifier">lcd_char(</span><span><span
class="cpp1-reservedword">unsigned</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">char</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">dat) { setb(PORT_RS,RS); write(dat); } </span><span
class="cpp1-comment">// Posli instrukci do LCD: </span><span><span
class="cpp1-reservedword">void</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">lcd_instr(</span><span><span
class="cpp1-reservedword">unsigned</span><span class="cpp1-space"> </span><span
class="cpp1-reservedword">char</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">ins) { clrb(PORT_RS,RS); write(ins); } </span><span
class="cpp1-comment">// Inicializace LCD ve 4-bitovem modu </span><span><span
class="cpp1-reservedword">void</span><span class="cpp1-space"> </span><span><span

```

```

class="cpp1-identifier">lcd_ini(</span><span class="cpp1-reservedword">void</span><span
class="cpp1-brackets">) { </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">setb(DDR_D4,D4); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">setb(DDR_D5,D5); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">setb(DDR_D6,D6); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">setb(DDR_D7,D7); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">setb(DDR_RS,RS); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">setb(DDR_EN,EN); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">clrb(PORT_RS,RS); </span><span class="cpp1-space">
</span><span class="cpp1-identifier">clrb(PORT_EN,EN); </span><span
class="cpp1-space"> </span><span class="cpp1-identifier">wait_ms(</span><span
class="cpp1-number">16</span><span class="cpp1-brackets">); </span><span
class="cpp1-space"> </span><span class="cpp1-identifier">setb(PORT_D4,D4);
</span><span class="cpp1-space"> </span><span class="cpp1-identifier">setb(PORT_D5,D5);
</span><span class="cpp1-space"> </span><span class="cpp1-identifier">len();
</span><span class="cpp1-space"> </span><span
class="cpp1-identifier">wait_ms(</span><span class="cpp1-number">5</span><span
class="cpp1-brackets">); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">len(); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">wait_us(</span><span class="cpp1-number">6</span><span
class="cpp1-brackets">); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">len(); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">clrb(PORT_D4,D4); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">len(); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">lcd_instr(</span><span class="cpp1-hexadecimal">0x28</span><span
class="cpp1-brackets">); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">lcd_instr(</span><span class="cpp1-hexadecimal">0x06</span><span
class="cpp1-brackets">); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">lcd_instr(</span><span class="cpp1-hexadecimal">0x0c</span><span
class="cpp1-brackets">); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">lcd_instr(</span><span class="cpp1-hexadecimal">0x01</span><span
class="cpp1-brackets">); </span><span class="cpp1-space"> </span><span
class="cpp1-identifier">wait_ms(</span><span class="cpp1-number">3</span><span
class="cpp1-brackets">); } </span><span class="cpp1-comment"> //eof //(c) OK1ZKV 2012
</span></span>

```