Rotina:serial_int_stram_LCD.c

Prof.: Alberto Willian Mascarenhas

```
#include <16F877A.h>
#device adc=8

#FUSES NOWDT                    //No Watch Dog Timer
#FUSES HS                       //High speed Osc (> 4mhz for PCM/PCH) (>10mhz for PCD)
#FUSES NOPUT                    //No Power Up Timer
#FUSES NOPROTECT                //Code not protected from reading
#FUSES NODEBUG                  //No Debug mode for ICD
#FUSES BROWNOUT                 //Reset when brownout detected
#FUSES NOLVP                    //No low voltage prgming, B3(PIC16) or B5(PIC18) used
for I/O
#FUSES NOCPD                    //No EE protection
#FUSES NOWRT                    //Program memory not write protected

#use delay(clock=20000000)

#use rs232(baud=9600,parity=N,xmit=PIN_C6,rcv=PIN_C7,bits=8,STREAM=Wireless)

#ifndef lcd_enable
   #define lcd_enable     pin_E1      // pino enable do LCD
   #define lcd_rs         pin_E2      // pino rs do LCD
   //#define lcd_rw       pin_e2      // pino rw do LCD
   #define lcd_d4         pin_d4      // pino de dados d4 do LCD
   #define lcd_d5         pin_d5      // pino de dados d5 do LCD
   #define lcd_d6         pin_d6      // pino de dados d6 do LCD
   #define lcd_d7         pin_d7      // pino de dados d7 do LCD
#endif

#include "C:\Alberto\IFMT 2023-II\Microcontroladores\Driver\mod_lcd.c"

void main()
{
   unsigned int valor =8;

   setup_adc_ports(AN0_AN1_AN3);
   setup_adc(ADC_CLOCK_DIV_16);
   setup_psp(PSP_DISABLED);
   setup_spi(SPI_SS_DISABLED);
   setup_timer_0(RTCC_INTERNAL|RTCC_DIV_1);
   setup_timer_1(T1_DISABLED);
   setup_timer_2(T2_DISABLED,0,1);
   setup_comparator(NC_NC_NC_NC);
   setup_vref(FALSE);

   set_adc_channel(0);
   delay_us( 50 );

   lcd_ini();
   delay_ms(50);

   while(true){

       valor = read_adc();
       //valor = 65;
       lcd_escreve ('\f');  // Apaga (limpa) o display
       printf(lcd_escreve,"   IFMT 2023    \r\n");
       printf (lcd_escreve," Valor = %u",valor);
       fprintf(Wireless,"%u \n",valor); // Formato para enviar para o
                                        // BasicRealtimePlotter - Processing
       delay_ms(100);
   }
}
```