I2C_LCD_Library for I2C_PCF8574 \ A

This library uses the MikroC *I2C1*_Library function, all the commands are a carbon copy of the MikroC Lcd_Library,

The differences are as follows:

1. there is no need to define constant sbit's.
2. *I2C 1* module is used in this library.
3. *I2C1* module must be configured independently.
4. All function arguments are ***unsigned char*** type.

Library Routines

- I2C_Lcd_Init
- I2C_Lcd_Out
- I2C_Lcd_Chr
- I2C_Lcd_Cmd

I2C_Lcd_Init	
Prototype	**void** Lcd_Init(unsigned char I2C_address);
Returns	Nothing.
Description	Initializes I2C_Lcd module to multi line no cursor.
Requires	Global variables: • I2C address of the the module to be defined as unsigned char must be defined before using this function.

Example	`Unsigned char LCD_01_ADDRESS = 0x7E; //0x7E is the address of`
	`//PCF8574A with A0-2 at VSS`
	`I2C_LCD_init(LCD_01_ADDRESS)`

I2C_Lcd_Out	
Prototype	**void** Lcd_Out(**unsigned char** row, **unsigned char** column, **unsigned char** *text);
Returns	Nothing.
Description	Prints text on Lcd starting from specified position. Both string variables and literals can be passed as a text. Parameters : • `I2C_address`:**`the address of the I2C PCF8574 / A`** • `row`**:** starting position row number • `column`**:** starting position column number • `text`**:** text to be written
Requires	The Lcd module needs to be initialized. See I2C_Lcd_Init routine.

Example	`// Write text "Hello!" on Lcd starting from row 1, column 3:`
	`unsigned char LCD_01_ADDRESS = 0x7E;`
	
	`I2C_Lcd_Out(LCD_01_ADDRESS,1, 3, "Hello!");`

I2C_Lcd_Chr	
Prototype	**void** Lcd_Chr(**unsigned char** I2C_Address, **unsigned char** row, **unsigned char** column, **unsigned char** out_char);
Returns	Nothing.
Description	Prints character on Lcd at specified position. Both variables and literals can be passed as a character. Parameters : - `I2C_address`:**`the address of the I2C PCF8574 / A`** - `row`: writing position row number - `column`: writing position column number - `out_char`: character to be written
Requires	The Lcd module needs to be initialized. See I2C_Lcd_Init routine.

Example	`// Write character "i" at row 2, column 3:`
	`unsigned char LCD_01_ADDRESS = 0x7E;`
	
	`I2C_Lcd_Chr(LCD_01_ADDRESS ,2, 3, 'i');`

Prototype	**void** I2C_Lcd_Cmd(**unsigned char** I2C_Address ,Lcd Command,1);
Returns	Nothing.
Description	Sends command to Lcd. Parameters : - I2C_address:**the address of the I2C PCF8574 / A** - Lcd Command: writing Lcd Command - constant: 1 has no value in this context as reuse of function for other cmd **Note :** Predefined constants can be passed to the function, see Available Lcd Commands.
Requires	The Lcd module needs to be initialized. See I2C_Lcd_Init table.

Example	`// Clear Lcd display:`
	`unsigned char LCD_01_ADDRESS = 0x7E;`
	
	`I2C_Lcd_Cmd(LCD_01_ADDRESS,_LCD_CLEAR,1);`

Available Lcd Commands

typedef enum{ Lcd Command}Cmd_Type;
extern Cmd_Type Cmd; //this enum variable must be declared in the main source code

Lcd Command	**Purpose**
_LCD_FIRST_ROW	Move cursor to the 1st row
_LCD_SECOND_ROW	Move cursor to the 2nd row
_LCD_THIRD_ROW	Move cursor to the 3rd row
_LCD_FOURTH_ROW	Move cursor to the 4th row
_LCD_CLEAR	Clear display
_LCD_RETURN_HOME	Return cursor to home position, returns a shifted display to its original position. Display data RAM is unaffected.
_LCD_CURSOR_OFF	Turn off cursor
_LCD_UNDERLINE_ON	Underline cursor on
_LCD_BLINK_CURSOR_ON	Blink cursor on
_LCD_MOVE_CURSOR_LEFT	Move cursor left without changing display data RAM
_LCD_MOVE_CURSOR_RIGHT	Move cursor right without changing display data RAM
_LCD_TURN_ON	Turn Lcd display on
_LCD_TURN_OFF	Turn Lcd display off
_LCD_SHIFT_LEFT	Shift display left without changing display data RAM
_LCD_SHIFT_RIGHT	Shift display right without changing display data RAM

I2C module number

Prototype	void I2CNo_Init(I2C_Type I2C_No);
Description	Tells the file which I2C module to use
Parameters	typedef enum I2C_Type I2C_No
Returns	null
Requires	mcu with an I2C module
Example	***I2CNo_Init(I2Cn)*** ; //n is for 1-5 of a module
Notes	none

Available I2C number commands

I2C1	Uses the I2C1_xxx functions of the mcu
I2C2	Uses the I2C2_xxx functions of the mcu
I2C3	Uses the I2C3_xxx functions of the mcu
I2C4	Uses the I2C4_xxx functions of the mcu
I2C5	Uses the I2C5_xxx functions of the mcu

```
#include "I2C_LCD.h"
#include "built_in.h"

//UChar LCD_01_ADDRESS =  0x7E;     //PCF8574A
UChar LCD_01_ADDRESS = 0x4E;        //PCF8574T

char txt[] = "Hello World";

void main() {
  int i = 0;
  int j = 1;
  int k = 0;
//CHECON = 0x00;
  TRISE3_bit = 0;
  AD1PCFG = 0xFFFFFFFF;
  JTAGEN_bit = 0;
  I2C2_Init(100000);//INIT I2C AT 100KHZ
  I2CNo_Init(I2C2);
  I2C_Set_Active(&I2C2_Start, &I2C2_Restart, &I2C2_Read, &I2C2_Write,
               &I2C2_Stop,&I2C2_Is_Idle); // Sets the I2C1 module active
  Delay_ms(100);
  I2C_LCD_Init(LCD_01_ADDRESS);
  Delay_ms(100);
  I2C_Lcd_Cmd(LCD_01_ADDRESS,_LCD_FIRST_ROW,1);
  I2C_Lcd_Cmd(LCD_01_ADDRESS,_LCD_CURSOR_OFF,1);     // Cursor off
  I2C_Lcd_Cmd(LCD_01_ADDRESS,_LCD_CLEAR,1);          // Clear display
  Delay_ms(1000);
  i=1;j=0;
  while(1){
         j++;

         i=1;

         I2C_LCD_Out(LCD_01_ADDRESS,j,1,txt);

         if(j > 4){

            j=1;

         }

         while( i < 10){

               Delay_ms(500);

               I2C_Lcd_Cmd(LCD_01_ADDRESS,_LCD_SHIFT_RIGHT ,1);

               i++;

          }

          while( i > 0){

               Delay_ms(500);

               I2C_Lcd_Cmd(LCD_01_ADDRESS,_LCD_SHIFT_LEFT ,1);

               i--;

          }

          I2C_Lcd_Cmd(LCD_01_ADDRESS,_LCD_CLEAR,1);          // Clear display

  }

}
```