

Extrait du Association Québécoise de Robotique Amateur

<http://www.aqra.ca>

Librairie en C18 pour LCD

- Programmation -

Date de mise en ligne : jeudi 2 juin 2005

Association Québécoise de Robotique Amateur

Cette librairie a été écrite pour faciliter l'incorporation d'un afficheur LCD à un projet. Elle est écrite en C18, pour les PICs de la famille 18F. Plutôt que de faire un grand discours, regardons le code :

```
//Librairie pour afficheur LCD 16*1 caractères
//Révision du 01/06/2005

//N.B. Le LCD 16*1 est en fait deux LCDs 8*1 sur la même ligne
//c'est pourquoi on doit choisir entre ligne 1 et 2 (chaque ½ écran)
```

```
/*
Définitions des pins pour le LCD
À ajuster selon l'application
```

```
LCD => PIC
```

```
-----
```

```
14 => RD7
13 => RD6
12 => RD5
11 => RD4
10 => RD3
9  => RD2
8  => RD1
7  => RD0
6  => RC4 (En)
5  => RC5 (RW)
4  => RC3 (RS)
3  => x  (contraste)
2  => x  (GND)
1  => x  (5V)
```

```
*/
```

```
#define LCDdata PORTD
#define TRIS_LCDdata TRISD
#define LCDenable PORTCbits.RC4
#define TRIS_LCDenable TRISCbits.TRISC4
#define LCDrs PORTCbits.RC3
#define TRIS_LCDrs TRISCbits.TRISC3
#define LCDrw PORTCbits.RC5
#define TRIS_LCDrw TRISCbits.TRISC5
```

```
//Défines pour config du lcd
```

```
#define clear 1
#define home 2
#define cursorON 3
#define cursorOFF 4
#define shift 5
```

```
//Délais
```

```
#define wait Delay10KTCYx(1)
#define longwait Delay10KTCYx(15)
```

```
#define shortwait Delay1KTCYx(1)
#define Delay_05ms Delay10TCYx(250)

//variables utilisées ds les routines
int t;
int o;

//fonction d'initialisation de l'écran
void initlcd (void)
{
LCDdata = 0b00000000; //toutes les pins de données à 0
LCDrw = 0;
LCDenable = 0; //enable à 0
LCDrs = 0; //Register Select à 0 (configuration)

longwait;

//display ON
LCDdata = 0b00001111;
wait;
LCDenable = 1;
wait;
LCDenable = 0;
wait;

// mode 8 bit
LCDdata = 0b00000001;
wait;
LCDenable = 1;
wait;
LCDenable = 0;
wait;

// effacer l'écran
LCDdata = 0b00111000;
wait;
LCDenable = 1;
wait;
LCDenable = 0;
wait;
}

//fonction pour choisir entre quelle ligne d'écran
void line (int x)
{
if (x == 1)
{
//set display adress au début de l'écran (ligne 1)
LCDrs = 0;
LCDrw = 0;
LCDdata = 0x80;
wait;
}
```

```
LCDenable = 1;
wait;
LCDenable = 0;
wait;
}

if (x == 2)
{
//set display adress au 9e caractère (ligne 2)
LCDrs = 0;
LCDrw = 0;
LCDdata = 0xC0;
wait;
LCDenable = 1;
wait;
LCDenable = 0;
wait;
}
}

//configuration de l'écran (efface, home, etc)
void configlcd (int config)
{
if (config == clear)
{
//efface l'écran
LCDrs = 0;
LCDdata = 0b00000001;
shortwait;
LCDenable = 1;
shortwait;
LCDenable = 0;
wait;
}

if (config == home)
{
//curseur et écran home
LCDrs = 0;
LCDdata = 0b00000011;
wait;
LCDenable = 1;
wait;
LCDenable = 0;
wait;
}

if (config == cursorON)
{
//curseur ON
LCDrs = 0;
LCDdata = 0b00001111;
}
```

```
wait;
LCDenable = 1;
wait;
LCDenable = 0;
wait;
}

if (config == cursorOFF)
{
//curseur OFF
LCDrs = 0;
LCDdata = 0b00001100;
wait;
LCDenable = 1;
wait;
LCDenable = 0;
wait;
}

if (config == shift)
{
//shift display
LCDrs = 0;
LCDdata = 0b00011000;
wait;
LCDenable = 1;
wait;
LCDenable = 0;
wait;
}
}

//affichage d'un caractère sur l'écran
void printlcd (char text)
{
LCDrs = 1;
shortwait;
LCDdata = text; // print text
shortwait;
LCDenable = 1;
shortwait;
LCDenable = 0;
LCDdata = 0b00000000; //remet toutes les sorties à 0
LCDrs = 0;
}

//affichage d'une chaîne de caractères sur l'écran
void printstringlcd (char *str)
{
line(1);

for(t=0; t<=7; ++t) //imprime les premiers caractères
```

```
{
printlcd(str[t]);
}

line(2);

for(t=8; t<=15; ++t) //imprime les derniers caractères
{
printlcd(str[t]);
}
}
```

Le code source de la librairie vous étant maintenant connu, voici un petit exemple de comment l'utiliser pour afficher "Bonjour" sur votre écran LCD.

```
//includes
#include <p18f452.h> // for ports declarations
#include <delays.h> // for delay routines
#include <timers.h> //for interrupts
#include "liblcd.c"

//variables
char bonjour[] = "  Bonjour !  ";

//fonction principale du programme
void main ()
{
//On désactive l'acquisition analogique
ADCON1=6;

//définitions des I/O
TRIS_LCDdata = 0;
TRIS_LCDenable = 0;
TRIS_LCDrw = 0;
TRIS_LCDrs = 0;

initlcd();
printstringlcd(bonjour);

while(1)
{
}

//config du PIC
#pragma config OSC = HS
#pragma config PWRT = ON
#pragma config WDT = OFF
#pragma config BOR = ON
#pragma config BORV = 42
#pragma config LVP = OFF
```

Tout ce qui est écrit ci-dessus est pour un afficheur 16*1 et pour un PIC18F452. Vous pouvez bien sur changer le modèle de PIC sans problème, en ajustant les pins et le *#include* en haut du texte. Pour changer le modèle de LCD (par exemple pour utiliser un 16*2), il suffira de changer les valeurs des limites de ligne dans la fonction *printstringlcd*.