Guide de mise en marche du shield LCD 2x16 GT1113

Matériel nécessaire :

- 1 x carte compatible Uno®
- 1 x shield LCD 2x16 GT1113

Présentation du module :

Shield LCD 2 x 16 caractères blancs à rétroéclairage bleu pour carte compatible Uno® et équipé de 6 boutons-poussoirs pour la sélection (select, left, down, up, right et reset).

Alimentation : 5 Vcc via la carte Arduino®
Dimensions : 80 x 58 x 22 mm

Table de correspondance :

Carte microcontrôleur	Shield LCD
A0	Boutons-poussoirs
D4	DB4 (LCD)
D5	DB5 (LCD)
D6	DB6 (LCD)
D7	DB7 (LCD)
D8	RS
D9	RW
D10	Rétroéclairage

Exemple de programme :

L'exemple de code suivant (à copier dans l'IDE Arduino®) permet de tester les différents boutons du module.

```
#include <LiquidCrystal.h>

//Sample using LiquidCrystal library


/*********************************************************

This program will test the LCD panel and the buttons
Mark Bramwell, July 2010

*********************************************************/

// select the pins used on the LCD panel
LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

// define some values used by the panel and buttons
int lcd_key     = 0;
int adc_key_in  = 0;
#define btnRIGHT  0
#define btnUP     1
#define btnDOWN   2
#define btnLEFT   3
#define btnSELECT 4
#define btnNONE   5

// read the buttons
int read_LCD_buttons()
{
 adc_key_in = analogRead(0);      // read the value from the sensor
 // my buttons when read are centered at these valies: 0, 144, 329, 504, 741
 // we add approx 50 to those values and check to see if we are close
 if (adc_key_in > 1000) return btnNONE; // We make this the 1st option for speed reasons since
it will be the most likely result
 // For V1.1 us this threshold
 if (adc_key_in < 50)   return btnRIGHT;
 if (adc_key_in < 250)  return btnUP;
 if (adc_key_in < 450)  return btnDOWN;
 if (adc_key_in < 650)  return btnLEFT;
 if (adc_key_in < 850)  return btnSELECT;

 // For V1.0 comment the other threshold and use the one below:
/*
 if (adc_key_in < 50)   return btnRIGHT;
 if (adc_key_in < 195)  return btnUP;
 if (adc_key_in < 380)  return btnDOWN;
 if (adc_key_in < 555)  return btnLEFT;
 if (adc_key_in < 790)  return btnSELECT;
*/


 return btnNONE;  // when all others fail, return this...
}

void setup()
{
 lcd.begin(16, 2);              // start the library
 lcd.setCursor(0,0);
 lcd.print("Push the buttons"); // print a simple message
}

void loop()
{
 lcd.setCursor(9,1);            // move cursor to second line "1" and 9 spaces over
 lcd.print(millis()/1000);      // display seconds elapsed since power-up


 lcd.setCursor(0,1);            // move to the begining of the second line
```

```
  lcd_key = read_LCD_buttons();  // read the buttons

  switch (lcd_key)               // depending on which button was pushed, we perform an action
  {
    case btnRIGHT:
      {
      lcd.print("RIGHT ");
      break;
      }
    case btnLEFT:
      {
      lcd.print("LEFT   ");
      break;
      }
    case btnUP:
      {
      lcd.print("UP    ");
      break;
      }
    case btnDOWN:
      {
      lcd.print("DOWN  ");
      break;
      }
    case btnSELECT:
      {
      lcd.print("SELECT");
      break;
      }
      case btnNONE:
      {
      lcd.print("NONE  ");
      break;
      }
  }

}
```

GO TRONIC

ROBOTIQUE ET COMPOSANTS ÉLECTRONIQUES

Si vous rencontrez des problèmes, merci de nous contacter par courriel à :

sav@gotronic.fr