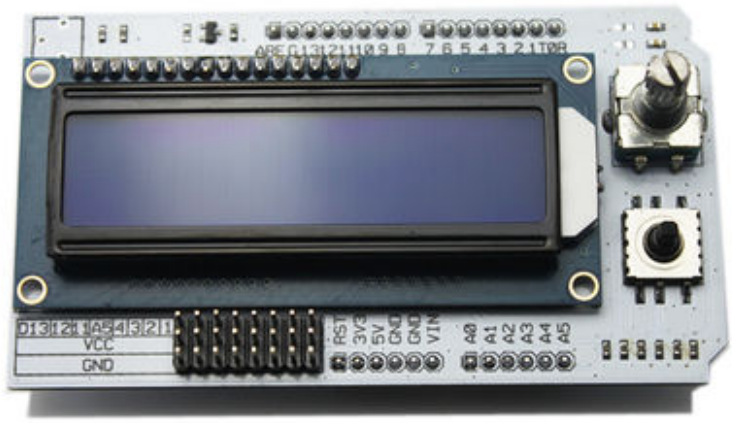
LCD Key Shield

From Wiki

Contents

Introduction

LCD Key Shield (http://www.elecfreaks.com/store/lcd-key-shield-p-688.html) is an LCD expansion board with a 1602 LCD display , a five-way key, a rotary encoder. It is fully compatible with the Arduino UNO board. You can achieve the desired display function only by connecting it with the Arduino UNO board, and then programming the corresponding code . In addition, we extended a five-way key and a rotary encoder, and (D13-D11) Digital IO, (A0-A5) Analog IO containing SPI, I2C interface, so you can achieve key function on LCD1602 screen and encoder function, and you can also assemble the electronic bricks and many other modules with our LCD Key Shield, therefore, you can display the results you need on the LCD.

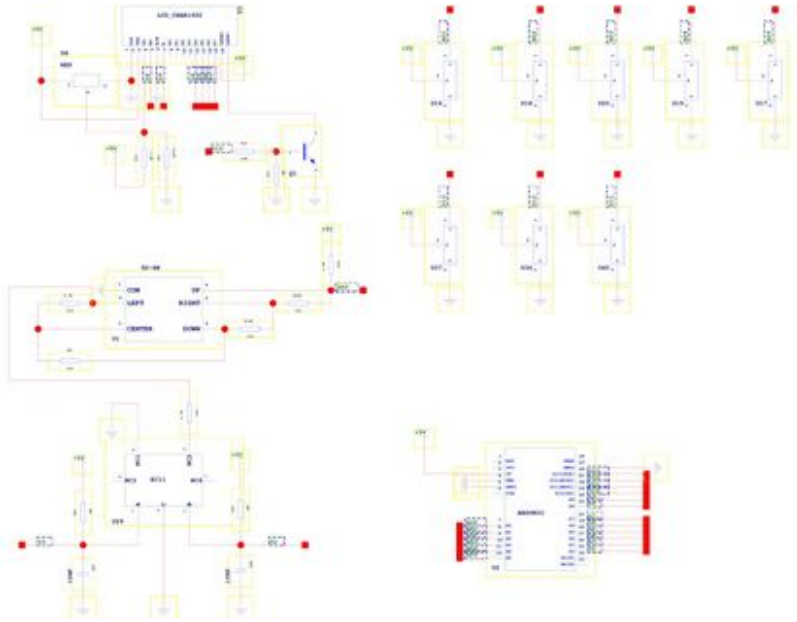


Model: EF02006

Feature

- Operating Voltage: 5V
- Can achieve six key functions, a rotary encoder function
- 3 Digital IO and 5 Analog IO interface

Schematic



Specification

Usage

Hardware Installation

Programming

Includes important code snippet. Demo code like :

```
Demo code:
//Sample using LiquidCrystal library
#include <LiquidCrystal.h>
/*********************************************************
This program will test the LCD panel and the buttons Mark Bramwell, August 2013
*********************************************************/
// select the pins used on the LCD panel

void  Encoder_san();
//================================================
//Set Encoder pin
//================================================
const int Encoder_A =  3;            // Incremental Encoder singal A is PD3
const int Encoder_B =  2;            // Incremental Encoder singal B is PD2
//const int ledPin    =  13;
unsigned int Encoder_number=0;
 int state=0;

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
#define btnEncodeOK  6
// 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(digitalRead(11)==0) return EncodeOK;
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 btnLEFT;
if (adc_key_in < 150)  return btnUP;
if (adc_key_in < 250)  return btnRIGHT;
if (adc_key_in < 450)  return btnSELECT;
if (adc_key_in < 700)  return btnDOWN;
if (adc_key_in < 850)  return btnEncodeOK;
// 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
pinMode(10,OUTPUT);
digitalWrite(10, 1);
lcd.setCursor(0,0);
lcd.print("Push the buttons"); // print a simple message
//======================================
 //pinMode(11,INPUT);
// digitalWrite(11, 1);
 //======================================
  pinMode(Encoder_A, INPUT);
  pinMode(Encoder_B, INPUT);
  digitalWrite(Encoder_A, 1);
  digitalWrite(Encoder_B, 1);
  //======================================
  attachInterrupt(1, Encoder_san, FALLING);        //interrupts: numbers 0 (on digital pin 2) and 1 (on digit
}
void loop()
{


// lcd.print(millis()/1000);      // display seconds elapsed since power-up
  if(state==1)
  {
    lcd.clear();
    lcd.setCursor(9,1);            // move cursor to second line "1" and 9 spaces over
    lcd.print(Encoder_number);
    lcd.setCursor(0,0);            // move cursor to second line "1" and 9 spaces over
    lcd.print("Push the buttons"); // print a simple message
    state=0;
  }
 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 btnEncodeOK:
 {
 lcd.print("EncdOK");
 break;     }

 case btnNONE:
 {
 lcd.print("NONE  ");
 break;     }
    }
}


 void Encoder_san()
{

        if(digitalRead(Encoder_B))
          {
             Encoder_number++;
          }
        else
          {
            Encoder_number--;
          }
          state=1;
}
```


VCC
GND


RST
3V3
5V
GND
GND
VIN
A0
A1
A2
A3
A4
A5