

The background features three large, overlapping blue circles of varying sizes, each with a gradient from dark blue to light blue. Two thin, light blue lines intersect at the top left, forming a large 'V' shape that frames the circles.

LCD Keypad Shield (DFR0009)

D-Robotics UK (www.droboticsonline.com)

This is a very popular LCD Keypad shield for Arduino or Freeduino board

D-Robotics
6/25/2011

LCD Keypad Shield



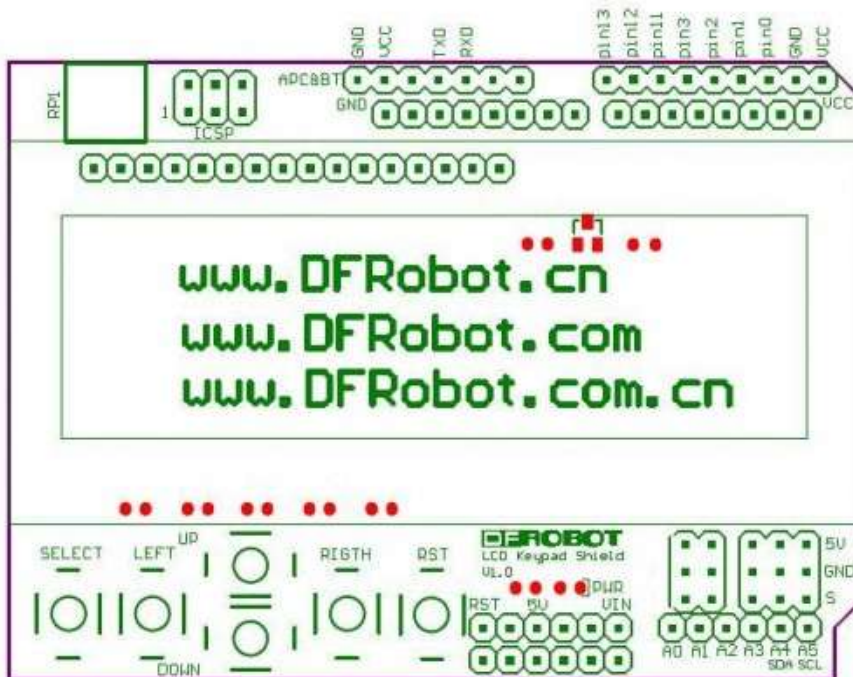
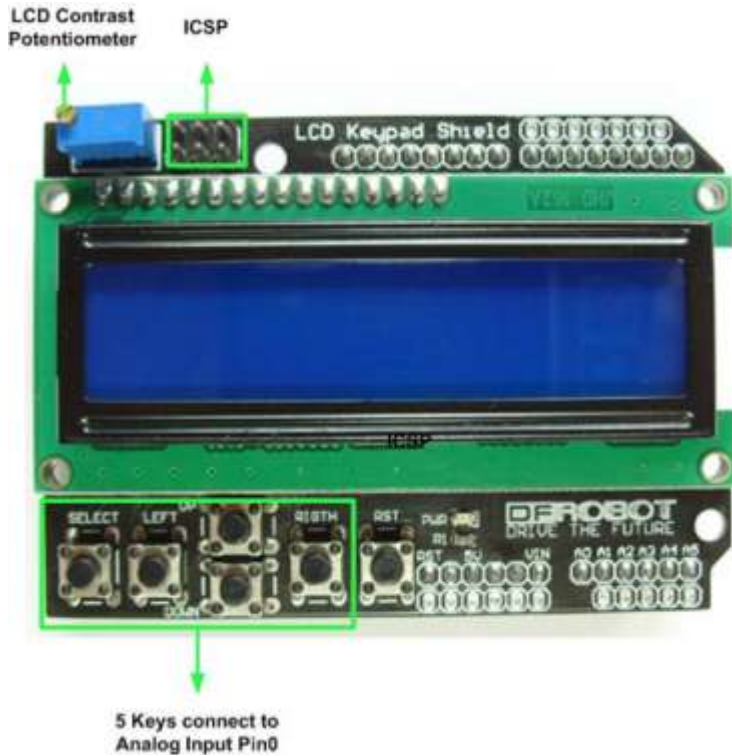
1. Introduction

This DFRobot LCD Keypad Shield is a very popular shield for Arduino or Freeduino board. The *LCD Keypad shield* is developed for Arduino compatible boards, to provide a user-friendly interface that allows users to go through the menu, make selections etc. It consists of a 1602 white character blue backlight LCD. The keypad consists of 5 keys — select, up, right, down and left. To save the digital IO pins, the keypad interface uses only one ADC channel. The key value is read through a 5 stage voltage divider.

2. Specifications

- Operating Voltage:5V
- 6 Push buttons
- Expanded Analog Pinout with standard DFRobot configuration.

3. Diagram



4. Pin Allocations

Pin	Function
Analog 0	Button (Select Up, Right, Down and Left)
Digital 4	DB4
Digital 5	DB5
Digital 6	DB6
Digital 7	DB7
Digital 8	RS (Data or Signal Display Selection)
Digital 9	Enable
Digital 10	Backlit Control

5. Sample Code

5.1 Example use of LCD4Bit_mod library

Download Link: http://droboticsonline.com/ebaydownloads/LCD4Bit_mod.zip

```
1 //
2 #include <LCD4Bit_mod.h>
3 //create object to control an LCD.
4 //number of lines in display=1
5 LCD4Bit_mod lcd = LCD4Bit_mod(2);
6 //Key message
7 char msgs[5][15] = {"Right Key OK ",
8                   "Up Key OK  ",
9                   "Down Key OK ",
10                  "Left Key OK ",
11                  "Select Key OK" };
12 int  adc_key_val[5] = {30, 150, 360, 535, 760 };
13 int  NUM_KEYS = 5;
14 int  adc_key_in;
15 int  key=-1;
16 int  oldkey=-1;
17 void setup() {
18     pinMode(13, OUTPUT); //we'll use the debug LED to output a heartbeat
19
20     lcd.init();
21     //optionally, now set up our application-specific display settings, overriding whatever the lcd did in lcd.init()
22     //lcd.commandWrite(0x0F);//cursor on, display on, blink on. (nasty!)
23     lcd.clear();
24     lcd.println("KEYPAD testing... pressing");
25 }
26 void loop()
27 {
28     adc_key_in = analogRead(0); // read the value from the sensor
```

```

29 digitalWrite(13, HIGH);
30 key = get_key(adc_key_in); // convert into key press
31 if (key != oldkey) // if keypress is detected
32 {
33     delay(50); // wait for debounce time
34     adc_key_in = analogRead(0); // read the value from the sensor
35     key = get_key(adc_key_in); // convert into key press
36     if (key != oldkey)
37     {
38         oldkey = key;
39         if (key >=0){
40             lcd.cursorTo(2, 0); //line=2, x=0
41             lcd.println(msgs[key]);
42         }
43     }
44 }
45 digitalWrite(13, LOW);
46 }
47 // Convert ADC value to key number
48 int get_key(unsigned int input)
49 { int k;
50   for (k = 0; k < NUM_KEYS; k++)
51   {
52     if (input < adc_key_val[k])
53     { return k; }
54   }
55   if (k >= NUM_KEYS)
56     k = -1; // No valid key pressed
57   return k;
58 }

```

5.2 Example use of LiquidCrystal library

```

1 //Sample using LiquidCrystal library
2 #include <LiquidCrystal.h>
3
4 /*****
5
6 This program will test the LCD panel and the buttons
7 Mark Bramwell, July 2010
8
9 *****/
10
11 // select the pins used on the LCD panel
12 LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

```

```

13
14 // define some values used by the panel and buttons
15 int lcd_key = 0;
16 int adc_key_in = 0;
17 #define btnRIGHT 0
18 #define btnUP 1
19 #define btnDOWN 2
20 #define btnLEFT 3
21 #define btnSELECT 4
22 #define btnNONE 5
23
24 // read the buttons
25 int read_LCD_buttons()
26 {
27   adc_key_in = analogRead(0); // read the value from the sensor
28   // my buttons when read are centered at these values: 0, 144, 329, 504, 741
29   // we add approx 50 to those values and check to see if we are close
30   if (adc_key_in > 1000) return btnNONE; // We make this the 1st option for speed reasons since it will be the
    most likely result
31   if (adc_key_in < 50) return btnRIGHT;
32   if (adc_key_in < 195) return btnUP;
33   if (adc_key_in < 380) return btnDOWN;
34   if (adc_key_in < 555) return btnLEFT;
35   if (adc_key_in < 790) return btnSELECT;
36   return btnNONE; // when all others fail, return this...
37 }
38
39 void setup()
40 {
41   lcd.begin(16, 2); // start the library
42   lcd.setCursor(0,0);
43   lcd.print("Push the buttons"); // print a simple message
44 }
45
46 void loop()
47 {
48   lcd.setCursor(9,1); // move cursor to second line "1" and 9 spaces over
49   lcd.print(millis()/1000); // display seconds elapsed since power-up
50
51
52   lcd.setCursor(0,1); // move to the beginning of the second line
53   lcd_key = read_LCD_buttons(); // read the buttons
54
55   switch (lcd_key) // depending on which button was pushed, we perform an action
56   {
57     case btnRIGHT:
58     {
59       lcd.print("RIGHT ");

```

```

60     break;
61     }
62     case btnLEFT:
63     {
64         lcd.print("LEFT ");
65         break;
66     }
67     case btnUP:
68     {
69         lcd.print("UP ");
70         break;
71     }
72     case btnDOWN:
73     {
74         lcd.print("DOWN ");
75         break;
76     }
77     case btnSELECT:
78     {
79         lcd.print("SELECT");
80         break;
81     }
82     case btnNONE:
83     {
84         lcd.print("NONE ");
85         break;
86     }
87 }
88
89 }

```

5.3 Example use of Enhanced LiquidCrystal_I2C library

This library inherits LiquidCrystal and adds another method: button - to read button pushed on a keypad.

Library download Link: http://droboticsonline.com/ebaydownloads/LCDKeypad_by_Fj604.zip

Discussion about this library: <http://www.dfrobot.com/forum/index.php?topic=31.0>

Declaim:

This manual is provided by the manufacturerl. Although the due care has been taken during the translation, D-Robotics is not responsible for the accuracy of the information contained in this document. D-Robotics: www.droboticsonline.com; Email contact: d_robotics@hotmail.co.uk