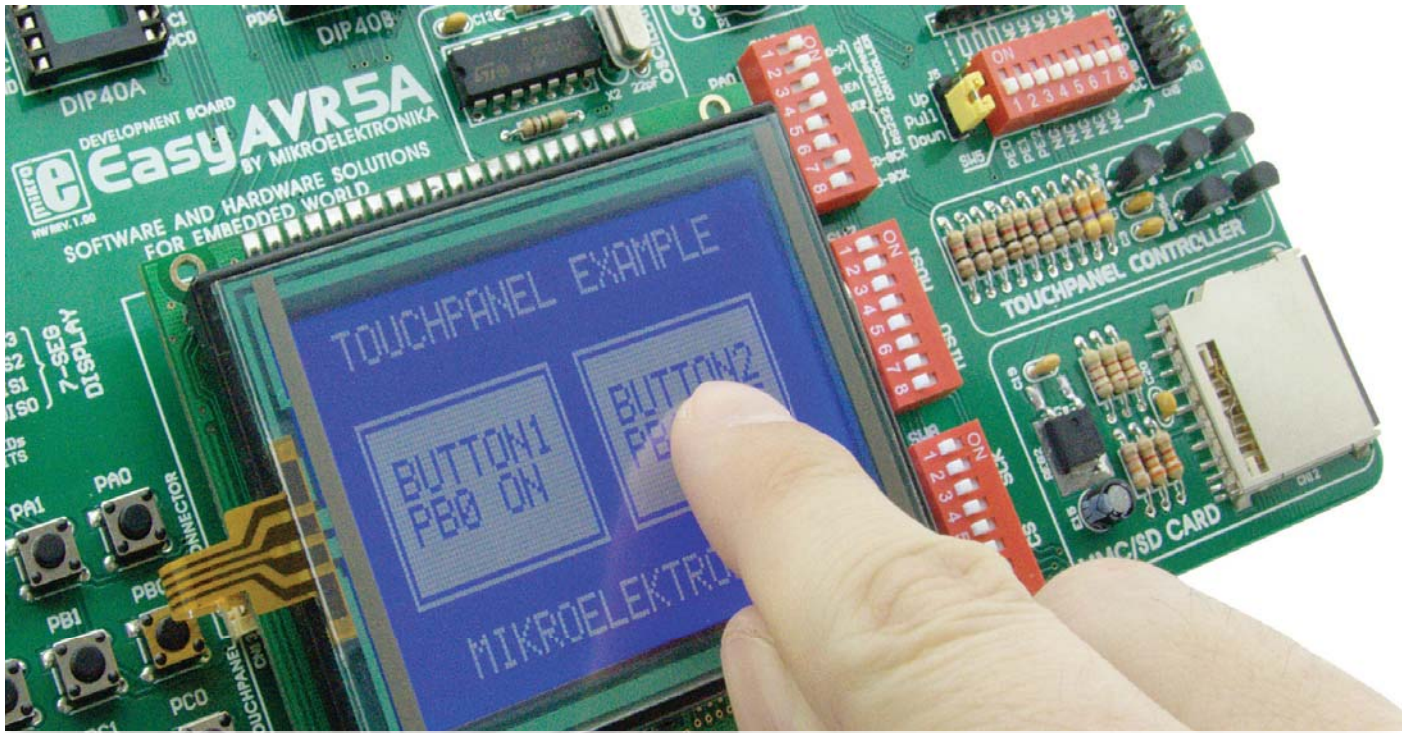


OK. Sie brauchen also einen ... TOUCHSCREEN



Von Dusan Mihajlovic

mikroElektronika Hardware-Abteilung

Soll Ihr aktuelles Projekt eine einfache und intuitive Schnittstelle haben? Wenn JA, dann ist ein grafisches LC-Display mit Touchpanel die beste Lösung, denn beides zusammen ergibt einen Touchscreen (GLCD + Touchpanel = Touchscreen). So können Sie aus wenigen elektronischen Bauteilen ein attraktives und benutzerfreundliches Gerät entwickeln.

Was ist ein Touchpanel? Ein Touchpanel ist dünnes, selbstklebendes, transparentes Tastenfeld, das auf den Bildschirm eines grafischen LC-Displays geklebt wird. Es ist sehr druckempfindlich, so dass bereits ein leichter Druck zu Änderungen des Ausgangssignals führt. Es gibt verschiedene Touchpanel-Typen. Am einfachsten ist das Widerstands-Touchpanel, das nachfolgend erläutert wird.

Funktionsweise

Ein Widerstands-Touchpanel besteht aus zwei transparenten und steifen Folien, die übereinander liegen und deren Innenseiten mit einer Widerstandsschicht überzogen sind. Der Widerstand dieser Schichten liegt in der Regel nicht über 1 k Ω . Die einander gegenüberliegenden Seiten der Folien sind mit Kontakten versehen, an denen ein Flachkabel angeschlossen werden kann. Die Koordinaten der Stelle, an der das Touchpanel gedrückt wird, werden in zwei Schritten bestimmt. Im ersten Schritt wird die X-Koordinate und im zweiten Schritt die Y-Koordinate des Punkts festgelegt. Um die X-Koordinate zu bestimmen, müssen der linke Kontakt auf der

X-Fläche mit Masse und der rechte Kontakt mit der Versorgungsspannung verbunden werden. Auf diese Weise kann durch Drücken des Touchpanels ein Spannungsteiler erzeugt werden. Der Wert des Spannungsteilers wird am unteren Kontakt der Y-Fläche abgegriffen. Die Spannung kann im Bereich von 0 V bis zum Wert der Betriebsspannung liegen und hängt von der X-Koordinate ab. Je näher sich der Punkt am linken Kontakt der X-Fläche befindet, umso dichter liegt

die Spannung bei 0 V. Um die Y-Koordinate zu bestimmen, müssen der untere Kontakt auf der Y-Fläche mit der Masse und der obere Kontakt mit der Versorgungsspannung verbunden werden. In diesem Fall wird die Spannung am linken Kontakt auf der X-Fläche abgegriffen.

Verbindung mit einem Mikrocontroller

Um ein Touchpanel mit einem Mikrocontroller zu verbinden, muss eine Schaltung zur Steuerung des Touchpanels vorhanden sein. Mithilfe dieser

Flachkabel- Informationen

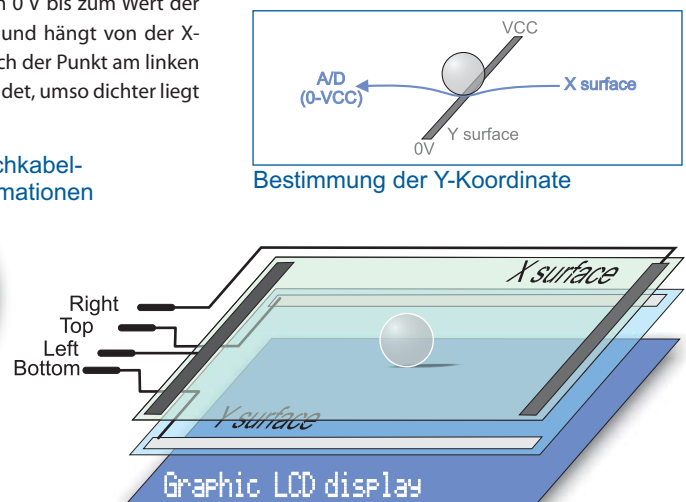
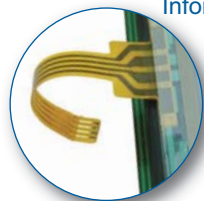
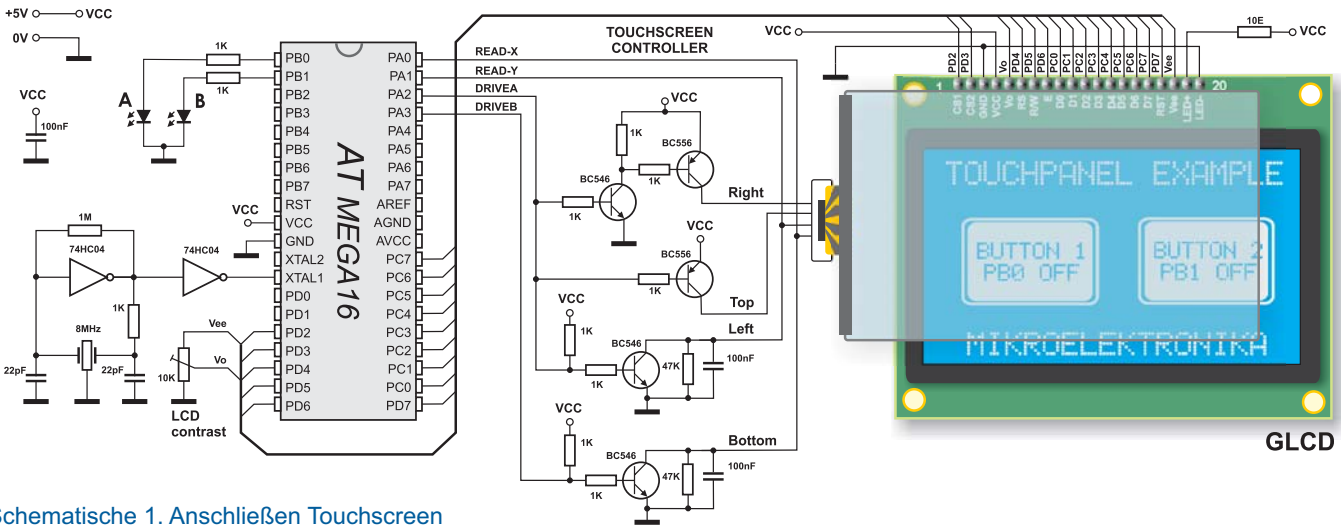


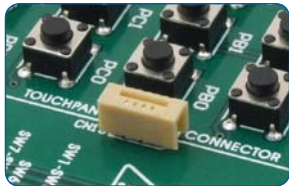
Abb. 1. Interne Touchpanel-Struktur



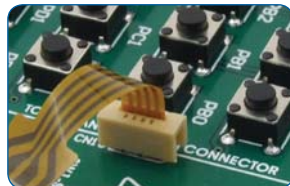
Schematische 1. Anschließen Touchscreen

Schaltung verbindet der Mikrocontroller die entsprechenden Kontakte des Touchpanels mit Masse und mit der Spannungsversorgung (wie oben beschrieben), um die X- und Y-Koordinaten zu bestimmen (siehe Schema 1). Der untere Kontakt auf der Y-Fläche und der linke Kontakt auf der X-Fläche werden mit dem A/D-Wandler des Mikrocontrollers verbunden. Die X- und Y-Koordinaten werden durch Messen der jeweiligen Spannung an diesen Kontakten ermittelt. Die Software generiert ein Menü auf einem grafischen LC-Display, aktiviert und deaktiviert die Schaltung für die Touchpanel-Steuerung und liest die Werte des A/D-Wandlers, die die X- und Y-Koordinaten des Punktes darstellen.

Sobald die Koordinaten festgelegt sind, können wir entscheiden, was der Mikrocontroller tun soll. Beispiel 1 veranschaulicht, wie zwei digitale Mikrocontroller-Ausgänge, die mit LED A und B verbunden sind, über ein Display und ein Touchpanel ein- und ausgeschaltet werden.



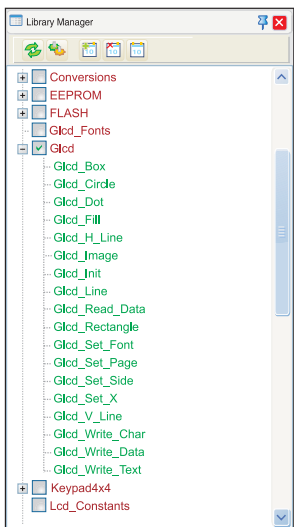
Flachkabel-Anschluss auf Platine vor ...



... und nach Anschließen von Touchpanel

Da die Fläche des Touchpanels etwas größer als die Fläche des grafischen LC-Displays ist, muss zur präziseren Bestimmung der Koordinaten eine Softwarekalibrierung des Touchpanels ausgeführt werden.

In dem Programm verwendete Funktionen



- ADC_Read() Analogwert lesen
 - Delay_ms() Verzögerung
 - Glcd_box() Gefülltes Rechteck zeichnen*
 - Glcd_circle() Kreis zeichnen
 - Glcd_Dot() Punkt zeichnen
 - Glcd_Fill() Anzeige löschen/füllen*
 - Glcd_H_Line() Horizontale Linie zeichnen
 - Glcd_Image() Bild importieren
 - Glcd_Init() Initialisierung von LCD-Display*
 - Glcd_Line() Linie zeichnen
 - Glcd_Read_Data() Daten von LCD lesen
 - Glcd_Rectangle() Rechteck zeichnen*
 - Glcd_Set_Font() Vorderseite wählen*
 - Glcd_Set_Page() Seite wählen
 - Glcd_Set_Side() Displayseite wählen
 - Glcd_Set_X() X-Koordinate bestimmen
 - Glcd_V_line() Vertikale Linie zeichnen
 - Glcd_Write_Char() Zeichen schreiben
 - Glcd_Write_Data() Daten schreiben
 - Glcd_Write_Text() Text schreiben*
- * Im Programm verwendete GLCD-Bibliotheksfunktionen

mikroC PRO for AVR® Bibliothekseditor mit einsatzbereiten Bibliotheken wie z. B.: Ethernet, CAN, SD/MMC usw.

Beispiel 1: Programm zur Demonstration des Touchscreen-Betriebs

```
// Glcd module connections
char GLCD_DataPort at PORTC;

sbit GLCD_CS1 at PORTD.B2;
sbit GLCD_CS2 at PORTD.B3;
sbit GLCD_RS at PORTD.B4;
sbit GLCD_RW at PORTD.B5;
sbit GLCD_EN at PORTD.B6;
sbit GLCD_RST at PORTD.B7;

char GLCD_DataPort_Direction at DDRD;
sbit GLCD_CS1_Direction at DDRD.B2;
sbit GLCD_CS2_Direction at DDRD.B3;
sbit GLCD_RS_Direction at DDRD.B4;
sbit GLCD_RW_Direction at DDRD.B5;
sbit GLCD_EN_Direction at DDRD.B6;
sbit GLCD_RST_Direction at DDRD.B7;
// End Glcd module connections

sbit DRIVE_A at PORTA.B2;
sbit DRIVE_B at PORTA.B3;
sbit DRIVE_A_Direction at DDRA.B2; // Touch Panel module connections
sbit DRIVE_B_Direction at DDRA.B3; // End Touch Panel module connections

long x_coord, y_coord, x_coord128, y_coord64; // scaled x-y position

unsigned int GetX() { //reading X
    DRIVE_A = 1; // DRIVEA = 1 (LEFT drive on, RIGHT drive on, TOP drive off)
    DRIVE_B = 0; // DRIVEB = 0 (BOTTOM drive off)
    Delay_ms(5);
    return ADC_Read(0); // READ-X (BOTTOM)
}

unsigned int GetY() { //reading Y
    DRIVE_A = 0; // DRIVEA = 0 (LEFT drive off, RIGHT drive off, TOP drive on)
    DRIVE_B = 1; // DRIVEB = 1 (BOTTOM drive on)
    Delay_ms(5);
    return ADC_Read(1); // READ-X (LEFT)
}

void main() {
    DRIVE_A_Direction = 1; // Set DRIVE_A pin as output
    DRIVE_B_Direction = 1; // Set DRIVE_B pin as output
    PORTB.B0 = 0; // Set PB0 pin as output (Default value 0)
    DDRB.B0 = 1; // Set PB0 pin as output (Default value 0)
    PORTB.B1 = 0; // Set PB1 pin as output (Default value 0)
    DDRB.B1 = 1; // Set PB1 pin as output (Default value 0)

    Glcd_Init(); // Initialize GLCD
    Glcd_Fill(0); // Clear GLCD
    Glcd_Set_Font(font5x7, 5, 7, 32); // Choose font,
    Glcd_Fill(0);

    Glcd_Write_Text("TOUCHPANEL EXAMPLE",10,0,1);
    Glcd_Write_Text("MIKROELEKTRONIKA",17,7,1);

    Glcd_Rectangle(8,16,60,48,1); //Display Buttons on GLCD:
    Glcd_Rectangle(68,16,120,48,1);
    Glcd_Box(10,18,58,46,1);
    Glcd_Box(70,18,118,46,1);
    Glcd_Write_Text("BUTTON1",14,3,0);
    Glcd_Write_Text("PB0 OFF",14,4,0);
    Glcd_Write_Text("BUTTON2",74,3,0);
    Glcd_Write_Text("PB1 OFF",74,4,0);

    while(1) { // read X-Y and convert it to 128x64 space
        x_coord = GetX();
        y_coord = GetY();
        x_coord128 = (x_coord * 128) / 1024;
        y_coord64 = 64 - (y_coord * 64) / 1024;

        //if BUTTON1 is selected
        if ((x_coord128 >= 10) && (x_coord128 <= 58) && (y_coord64 >= 18) && (y_coord64 <= 46)) {
            if(PORTB.B0 == 0) {
                PORTB.B0 = 1;
                Glcd_Write_Text("PB0 ON",14,4,0);
            }
            else {
                PORTB.B0 = 0;
                Glcd_Write_Text("PB0 OFF",14,4,0);
            }
        }

        //if BUTTON2 is selected
        if ((x_coord128 >= 70) && (x_coord128 <= 118) && (y_coord64 >= 18) && (y_coord64 <= 46)) {
            if(PORTB.B1 == 0) {
                PORTB.B1 = 1;
                Glcd_Write_Text("PB1 ON",74,4,0);
            }
            else {
                PORTB.B1 = 0;
                Glcd_Write_Text("PB1 OFF",74,4,0);
            }
        }

        Delay_ms(100);
    }
}
```



HINWEIS:

Zu diesem Beispiel finden Sie für AVR® Microcontroller in C, Basic und Pascal geschriebenen Code und die für PIC® und dsPIC® Microcontroller geschriebenen Programme auf unserer Website unter www.mikroe.com/en/article/.