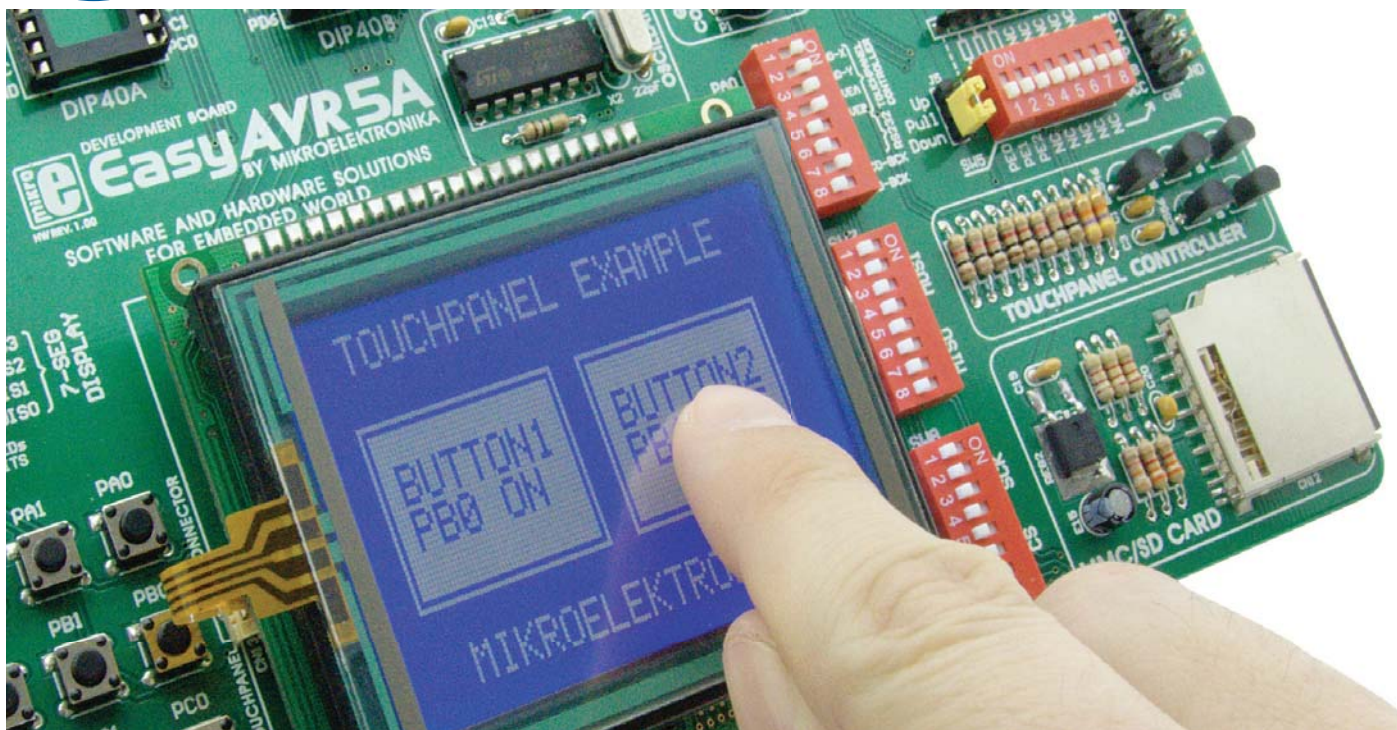


OK. Vama je potreban... TOUCHSCREEN



Dušan Mihajlović

Mikroelektronika - odsek za hardver

Želite li da vaš novi uređaj ima jednostavan i intuitivan interfejs? Touchscreen u kombinaciji sa grafičkim displejom je idealna stvar za to. Sa malim brojem elektronskih komponenti napravićete veoma atraktivan uređaj koji će uz to biti i izuzetno jednostavan za korišćenje.

Šta je touch panel? Touch panel je transparentna staklena pločica koja je osetljiva na dodir i koja se uz pomoć samolepljivog sloja pričvršćuje za površinu ekrana grafičkog displeja. Postoji nekoliko vrsta touch panela od kojih je najjednostavniji rezistivni tip koji ćemo ovom prilikom prezentovati.

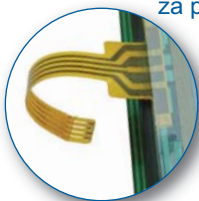
Princip rada

Rezistivni touch panel se sastoji od dve providne pločice koje formiraju "sendvič" i koje su sa unutrašnje strane prevučene providnim slojem čija otpornost obično ne prelazi 1Kohm. Po dve naspramne stranice ovih pločica su izvedene u vidu kontakata koji su dostupni za upotrebu preko flet kabla.

Postupak određivanja koordinata tačke u kojoj je touch panel pritisnut odvija se u dva koraka. Prvi korak je određivanje X koordinata pritisnute tačke nakon čega se pristupa drugom koraku, odnosno određivanju Y koordinate pritisnute tačke. Da bismo odredili X koordinatu, neophodno je povezati kontakt na levoj strani X površine na masu i kontakt na desnoj strani X površine na napon napajanja. Ovim se omogućuje da se pritiskom na površinu touch panela dobije razdelnik napona čija se vrednost zatim čita sa donjeg kontakta Y površine. Vrednost napona će biti u

granicama od 0V do vrednosti napona napajanja i menjaće se u zavisnosti od vrednosti X koordinate. Ukoliko je touch panel pritisnut bliže levom kontaktu X površine vrednost očitano napona će biti bliže vrednosti od 0V. Da bismo odredili Y koordinatu pritisnute tačke neophodno je povezati kontakt na donjoj strani Y površine na masu, a kontakt na gornjoj strani Y površine na napon napajanja. U ovom slučaju, vrednost napona se čita sa levog kontakta X površine.

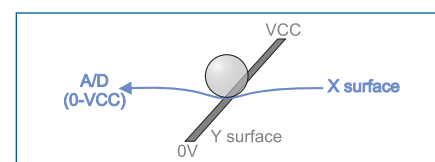
Trakasti kabl
za povezivanje



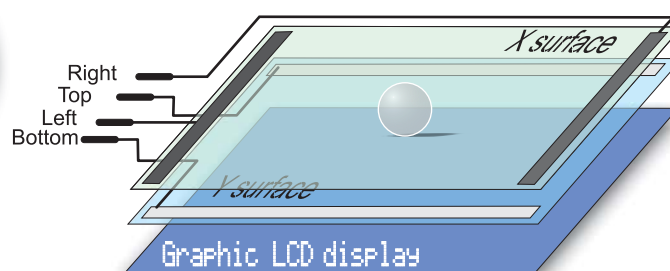
Povezivanje sa mikrokontrolerom

Da bismo povezali touch panel sa mikrokontrolerom, neophodno je napraviti električno kolo za kontrolu touch panela. Pomoću ovog kola mikrokontroler povezuje odgovarajuće kontakte touch panela na masu i napon napajanja (kao što je opisano) u cilju određivanja X i Y koordinata (Šema 1).

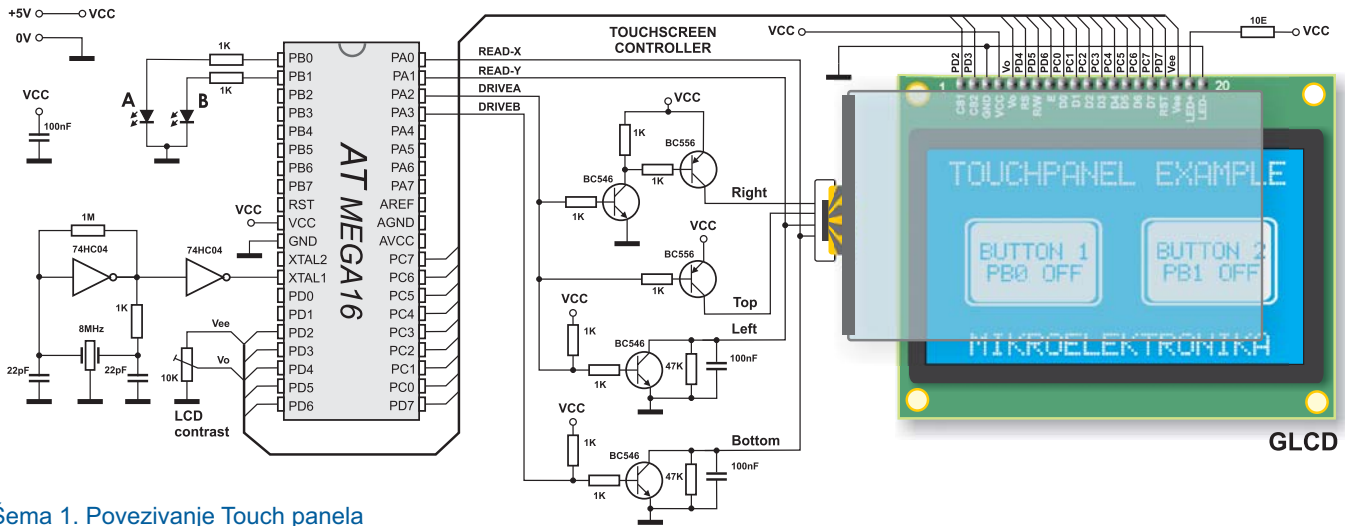
Kontakt na donjoj strani Y površine, odnosno kontakt na levoj strani X površine povezani su na A/D konvertor mikrokontrolera. Meranjem na-



Određivanje Y koordinate



Slika 1. Unutrašnja struktura touch panela



Šema 1. Povezivanje Touch panela

pona na ovim kontaktima određuje se X, odnosno Y koordinata pritisnute tačke. Softver koji je neophodan da bi upravljao ovim uređajem je veoma jednostavan. Suština njegovog rada je ispisivanje menija na grafičkom displeju, uključenje/isključenje električnog kola za kontrolu touch panela i čitanje vrednosti A/D konvertora koje zapravo predstavljaju X i Y koordinate pritisnute tačke.

Kada odredimo vrednosti koordinata, na osnovu njih možemo odrediti šta želimo da mikrokontroler uradi. Kao prost primer (Primer 1) prikazano je kako se uz pomoć displeja i touch panela može izvršiti uključenje/isključenje dva digitalna pina na mikrokontroleru (LED diode A i B).



Konektor za povezivanje trakastim kablom pre...



...i posle priključenja touch panela.

S obzirom da je površina touch panela malo veća od površine grafičkog displeja, u slučaju da želite veću preciznost određivanja koordinata pritisnute tačke, neophodno je uraditi softversku kalibraciju touch panela.

Spisak funkcija korišćenih u programu

ADC_Read()	Čitanje analogne vrednosti
Delay_ms()	Kašnjenje
Glcd_box()	Crtanje ispunjenog pravougaonika*
Glcd_circle()	Crtanje kružnice
Glcd_Dot()	Crtranje tačke
Glcd_Fill()	Brisanje/popuna ekrana*
Glcd_H_Line()	Crtranje horizontalne linije
Glcd_Image()	Unošenje slike
Glcd_Init()	Inicijalizacija LCD displeja*
Glcd_Line()	Crtranje linije
Glcd_Read_Data()	Čitanje podataka iz LCD-a
Glcd_Rectangle()	Crtranje pravougaonika*
Glcd_Set_Font()	Izbor fonta*
Glcd_Set_Page()	Izbor strane
Glcd_Set_Side()	Izbor polovine ekrana
Glcd_Set_X()	Određivanje X koordinate
Glcd_V_line()	Crtranje vertikalne linije
Glcd_Write_Char()	Ispisivanje karaktera
Glcd_Write_Data()	Ispisivanje bilo kog podatka
Glcd_Write_Text()	Ispisivanje teksta*

* Funkcije koje su iskoristene u programu.

Primer 1: Program za demonstraciju rada touch panela

```

program TouchPanel
dim GLCD_DataPort as byte at PORTC
GLCD_DataPort_Direction as byte at DDRC

dim GLCD_CS1 as sbit at PORTD.B2
GLCD_CS2 as sbit at PORTD.B3
GLCD_RS as sbit at PORTD.B4
GLCD_RW as sbit at PORTD.B5
GLCD_EN as sbit at PORTD.B6
GLCD_RST as sbit at PORTD.B7
GLCD_CS1_Direction as sbit at DDRD.B2
GLCD_CS2_Direction as sbit at DDRD.B3
GLCD_RS_Direction as sbit at DDRD.B4
GLCD_RW_Direction as sbit at DDRD.B5
GLCD_EN_Direction as sbit at DDRD.B6
GLCD_RST_Direction as sbit at DDRD.B7

dim DRIVE_A as sbit at PORTA.B2
DRIVE_B as sbit at PORTA.B3
DRIVE_A_Direction as sbit at DDRA.B2
DRIVE_B_Direction as sbit at DDRA.B3

dim x_coord, y_coord, x_coord128, y_coord64 as longint

sub function GetX() as word
DRIVE_A = 1
DRIVE_B = 0
Delay_ms(5)
result = ADC_Read(0)
end sub

sub function GetY() as word
DRIVE_A = 0
DRIVE_B = 1
Delay_ms(5)
result = ADC_Read(1)
end sub

main:
DRIVE_A_Direction = 1
DRIVE_B_Direction = 1
PORTB.B0 = 0
DDR.B0 = 1
PORTB.B1 = 0
DDR.B1 = 1

Glcd_Init()
Glcd_Set_Font(font5x7, 5, 7, 32)
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)
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 TRUE
x_coord = GetX()
y_coord = GetY()
x_coord128 = (x_coord * 128) / 1024
y_coord64 = 64 - (y_coord * 64) / 1024
if (x_coord128 >= 10) and (x_coord128 <= 58) and (y_coord64 >= 18) and (y_coord64 <= 46) then
if (PORTB.B0 = 0) then
PORTB.B0 = 1
Glcd_Write_Text("PB0 ON";14,4,0)
else
PORTB.B0 = 0
Glcd_Write_Text("PB0 OFF";14,4,0)
end if
if (x_coord128 >= 70) and (x_coord128 <= 118) and (y_coord64 >= 18) and (y_coord64 <= 46) then
if (PORTB.B1 = 0) then
PORTB.B1 = 1
Glcd_Write_Text("PB1 ON";74,4,0)
else
PORTB.B1 = 0
Glcd_Write_Text("PB1 OFF";74,4,0)
end if
end if
Delay_ms(100)
wend while true
end
    
```

NOTE: Ovaj program, pisan za AVR mikrokontrolere u programima C, Basic i Pascal kao i programe napisane za mikrokontrolere PIC® i dsPIC® možete pronaći na našem web sajtu: www.mikroe.com/en/article/

