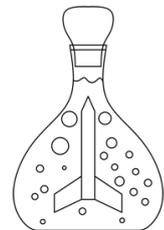
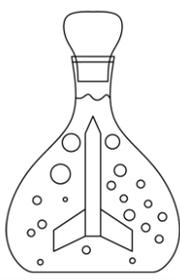


RESISTIVE TOUCH PANEL LIBRARY

Last update - 14.05.2015/



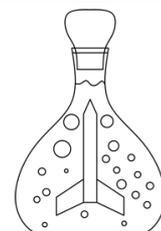


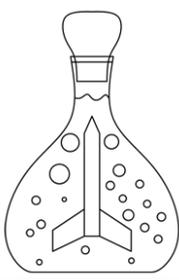
- The resistive touch panels work basically according to the resistance
- The sensibility of them is lower than other touch panels, but they are more enduring
- This library is consisted of 4 wire output resistive touch panel for make a control, which has written with the C and Micro C complier
- So as to control, family of PIC18 microprocessor is chosen
- As a driver, the circuit of the MikroElektronika brand is chosen. Details related to the circuits is mentioned in my page.
- For the touch panel 2 points of calibration method is chosen. This case can provide space and velocity saving. (in the methods of 3 points, 5 points and etc, is available)
- In this article, the functions will be explained. For more details and see the codes, you can visit my page.
- As a suggestion, in every start up if no calibration is desired to be done, the measured deviation ratio "Eproma" can be recorded. In this case in every startup nominal value can be read and there would not be any need for calibration.

PIN CONNECTIONS

```
sbit surucu_a at RA5_bit;           //dokunmatik panel surucu_a pini
sbit surucu_b at RB4_bit;           //dokunmatik panel surucu_b pini
sbit analog_x at RA0_bit;           //dokunmatik panel analog port ucu pini
sbit analog_y at RA1_bit;           //dokunmatik panel analog port ucu pini

sbit surucu_a_Direction at TRISA5_bit;           //dokunmatik panel surucu_a pini
sbit surucu_b_Direction at TRISB4_bit;           //dokunmatik panel surucu_b pini
sbit analog_x_Direction at TRISA0_bit;           //dokunmatik panel analog port ucu pini
sbit analog_y_Direction at TRISA1_bit;           //dokunmatik panel analog port ucu pini
```





- **D_PANEL_INIT(unsigned int Genislik , unsigned int Yukseklik ,unsigned char x analog kanali,unsigned char y analog kanali)**

The function of Init, provides making analysis through the touch panel software. Because of that, the wideness and height of the used panel is obtained.

By the way, for touch panel, analog channel in which the readout is done, they also will be obtained.

- **D_PANEL_X_OKUMA()**
- **D_PANEL_Y_OKUMA()**

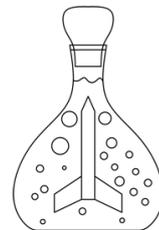
This make a readout in X and Y plane, but as a big difference, it would not read directly the coordinates, and at the touched point,the nominal values measured from the touch panel, *raw analog (uncured)* are read

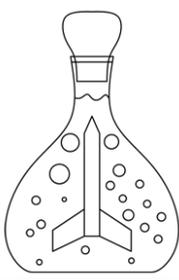
The nominal values of the touched area are returned in "float" value.

- **D_PANEL_DOKUNULDU_MU(unsigned int hassasiyet)**

It examines whether the panel is touched or not. It returns the value of 1 for the touched and 0 for not touched action.

"hassasiyet" parameter obtain the maximum value which is required to be read. For example if *"hassasiyet = 10"* , in case of rise above value of 10 the system counts as to be touched, so that the touch sensitivity can be adjusted.





- [D_PANEL_KALIBRASYON_NOKTA_1](#)
- [D_PANEL_KALIBRASYON_NOKTA_2](#)
- [D_PANEL_KALIBRASYON_NOKTA_3](#)
- [D_PANEL_KALIBRASYON_NOKTA_4](#)

It collects values from needed 4 points for calibration. When every function finished successfully, 1 value is returned.

Please check the video and my page. All details are mentioned there.

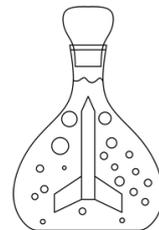
- [D_PANEL_KALIBRASYON_HESAPLAMA](#)

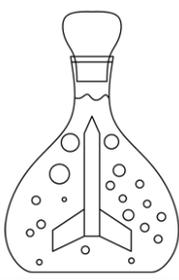
It is the function of calculation and summation of parameters. If the automatic calibration at every turn is not required to be done, it is called once and the global variables which are mentioned below is read and taken a note. Also in the beginning of the program, read values is assigned to the global variables.

In this case in ever startup there wouldn't be any need for calibration.

- [sapma_a](#)
- [sapma_b](#)
- [sapma_c](#)
- [sapma_d](#)

These variables are global and they are identified in the library. In Main, the type of the variables which can be assigned, is called "float".





- **D_PANEL_KONUM_X**
- **D_PANEL_KONUM_Y**

It shows the location of touched area on LCD..

As an example, when touching left top corner of a 240x128 screen (0,0), and touching right bottom corner (240,128) value is read.

Two function mentioned above, gives the processed raw analog values.

As to not being seen the fractions, the values are returned as "int" .

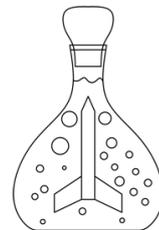
- **D_PANEL_KORDINAT_KONTROL**(unsigned char simge_x,unsigned char simge_y,unsigned char simge_genislik_px,unsigned char simge_yukseklk_px)

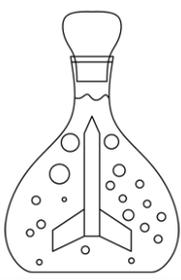
It has been designed as a function facilitator. It's main purpose is to determine easily the coordinates of a created button in LCD.

As an example, the values of a picture that the start position is (10,10) within the dimension of 30x30 px (wideness X height) is entered with parameters.

According to these information, it determines the function limits. Inside the limits, if one point is touched, it sends 1 value. If it is not touched, it sends 0 value.

When adding a button in this way, we can easily use it entering the button information and location.





**THIS LIBRARY IS NOT ALLOWED TO BE USED IN
ANY COMMERCIAL PROJECT**

All the rights are reserved©

www.ercankoclar.com

