

ITDB02_Touch - Arduino library support for ITDB02 Touch function

Copyright (C)2010 Henning Karlsen. All right reserved

You can find the latest version of the library at <http://www.henningkarlsen.com/electronics>

This library has been made for the 2.4" TFT LCD Screen Module: "ITDB02-2.4", the 3.2" TFT LCD Screen Module: "ITDB02-3.2" and the 3.2" Wide-screen module "ITDB02-3.2WC" by ITEad studio.

If you make any modifications or improvements to the code, I would appreciate that you share the code with me so that I might include it in the next release. I can be contacted through <http://www.henningkarlsen.com/electronics/contact.php>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

| | | | |
|----------|------|-------------|--|
| Version: | 1.0 | Sep 13 2010 | initial release |
| | 1.01 | Sep 18 2010 | Added example: ITDB02_Touch_ButtonTest |
| | 1.1 | Oct 7 2010 | Fixed incompatibility with ITDB02-3.2 |
| | 1.2 | Oct 12 2010 | Added support for ITDB02-3.2WC |
| | 1.21 | Mar 27 2011 | Updated some of the examples to be compatible with ITDB02_Graph(16) v4.0 |

Defined Literals:

| Orientation | |
|--------------------------|---|
| For use with InitTouch() | |
| PORTRAIT: | 0 |
| LANDSCAPE: | 1 |

| Precision | |
|-----------------------------|---|
| For use with setPrecision() | |
| PREC_LOW: | 1 |
| PREC_MEDIUM: | 2 |
| PREC_HI: | 3 |
| PREC_EXTREME: | 4 |

Functions:

ITDB02_Touch(TCLK, TCS, TDIN, TDOUT, IRQ);

The main class of the interface.

Parameters: TCLK: Arduino pin for Touch Clock
 TCS: Arduino pin for Touch Chip Select
 TDIN: Arduino pin for Touch Data input
 TDOUT: Arduino pin for Touch Data output
 IRQ: Arduino pin for Touch IRQ
Usage: ITDB02_Touch myTouch(15,10,14,9,8); // Start an instance of the ITDB02_Touch class

InitTouch(orientation);

Initialize the touch screen and set display orientation. If the library is used together with the ITDB02_Graph(16) the orientation should be set to the same orientation for both libraries.

Parameters: orientation: PORTRAIT (default)
 LANDSCAPE
Returns: Nothing
Usage: myTouch.InitTouch();// Initialize the touch screen

dataAvailable();

Check to see if new data from the touch screen is waiting.

Parameters: None
Returns: Boolean: true means data is waiting, otherwise false
Usage: check = myTouch.dataAvailable() // See if data is waiting

read();

Read waiting data from the touch screen. This function should be called if dataAvailable() is true. Use getX() and getY() to get the coordinates.

Parameters: None
Returns: Nothing
Usage: myTouch.read(); // Read data from touch screen
Notes: After calling read(), raw data from the touch screen is available in the variables TP_X and TP_Y. Do not use these if you do not know how to handle the raw data. Use getX() and getY() instead.

getX();

Get the x-coordinate of the last position read from the touch screen.

Parameters: None
Returns: Integer
Usage: x = myTouch.getX(); // Get the x-coordinate

getY();

Get the y-coordinate of the last position read from the touch screen.

Parameters: None
Returns: Integer
Usage: y = myTouch.getY(); // Get the y-coordinate

setPrecision(precision);

Set the precision of the touch screen.

Parameters: precision: **PREC_LOW, PREC_MEDIUM, PREC_HI, PREC_EXTREME**
Returns: Nothing
Usage: myTouch.setPrecision(PREC_MEDIUM); // Set precision to medium
Notes: Higher precision data will take longer to read, so take care when using PREC_HI or PREC_EXTREME with fast-moving input.