mTouch™ GUI

User's Guide

Contents

1. INTRODUCTION

When configured for the 2-way communication, the mTouch firmware can be fine-tuned at run time to achieve the highest performance and sensitivity.

The 2-Way PC GUI is a valuable tool, allowing not only to read and provide the user with the actual values for the critical parameters of mTouch products, but also to alter the values, for optimization purposes, depending on the hardware the mTouch firmware is running on.

1.1. Pre-requisites:

In order for the GUI to function, it is necessary to connect the computer running the application to an mTouch product, running a compatible firmware. The user needs:

- A computer running Windows
- Microsoft's .NET framework v2.0 and up, already installed.
- USB communication from the computer to the mTouch device.
- If using any Microchip Evaluation or Demonstration board with a 6-pin header connector, providing UART signals (Rx and Tx), it is necessary to bridge and convert UART signals to USB. PICKit Serial Analyzer will provide this functionality.
- If using PICKit Serial Analyzer, firmware version 0307 and up is mandatory
- A USB Cable
- A mTouch product running mTouch framework v2.0 and above

The following figure shows a typical mTouch communication platform

2. Application Initialization

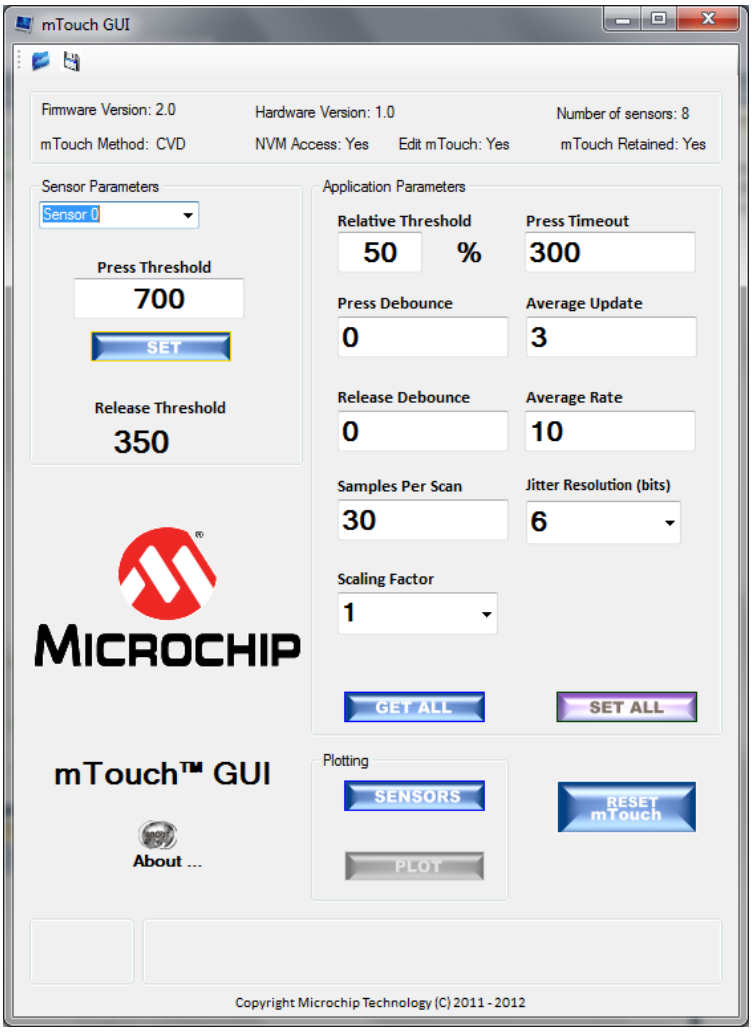
The mTouch GUI starts by establishing communication with the mTouch host application before it starts collecting all the necessary data for the session to operate. During this initialization phase, the application grants the user with a splash screen that displays the actual operation the GUI is performing.

The figure below shows the splash screen with an actual normal operation, at the bottom of the window:

If the application initialization fails (for multiple reasons), the GUI will notify the user, by displaying an error window, with additional indications when possible, to resolve the issue. An example of initialization failure (an actual board disconnection) is shown in the figure below:

3. The GUI Main Window:

Once the initialization phase has successfully completed, the mTouch GUI displays its main windows, providing the user with various pieces of information from the actual mTouch device, but allowing to update some of the parameters, as well.



3.1. The Toolbar:

The main toolbar is located at the top of the main window. It contains 2 icons as shown below:

Clicking this icon allows the user to select an mTouch configuration file with a set of parameters that can be loaded by the GUI into the current mTouch device, it is communicating with.

This icon allows the user to store the current mTouch parameter set (current mTouch configuration) , on the computer's hard drive, at the selected location. The data will be stored into the specified file name with an ".mtc" extension (**mT**ouch **C**onfiguration).

3.2. The mTouch Information block:

Right underneath the toolbar is the mTouch application information block. This data mainly describes how the application is configured and what hardware and software versions are being used:

Firmware Version: This field displays the version of the actual software running on the mTouch device. This parameter is read-only on the GUI, and can only be updated in the source code of the mTouch framework, before the new firmware is compiled.

Hardware Version: This field shows the revision number of the hardware, the mTouch application is currently running on. This information can only be read by the GUI, and can only be altered in the source code of the mTouch framework by the programmer, before the new firmware is compiled.

Number of Sensors: This parameter informs the user about the number of Touch sensors currently handled by the mTouch device. This number is read-only and can only be set by the programmer in the framework source code, before the new firmware is compiled.

mTouch method: This field the scanning method used by the mTouch device. The scan algorithm can only be selected by the mTouch programmer, depending on the hardware capabilities of the mTouch device. It is set at compile time.

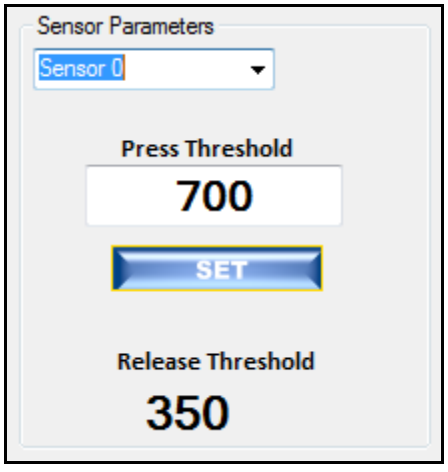
NVM Access: This field is binary and indicates whether the access to the non-volatile memory on the mTouch device is allowed or not. This information can only be set by the mTouch framework programmer at compile time.

Edit mTouch: This field is binary. It informs the user about the ability to edit the mTouch parameters. If its value is set to "No", then all mTouch parameters (described in the next pages) can only be read by the GUI and cannot be altered. This switch is configured by the mTouch framework programmer, at compile time.

mTouch Retained: This field is binary. It lets the user know whether the mTouch device embeds non-volatile memory (EEPROM on PIC microcontrollers), allowing the mTouch application to retain all mTouch parameters even through a hardware power-cycle, or not. When this field reads "No", then the mTouch parameters are held in RAM, and a device power cycle would reset the factory configuration. Users still have the option to reload a previously stored configuration into the GUI, and send it to the mTouch device.

3.3. The Sensor-Related Parameter Block:

On the left side of the window you will notice a group of parameters all grouped in the "Sensor Parameters" section. The values shown in this section relate to a given sensor at a time:



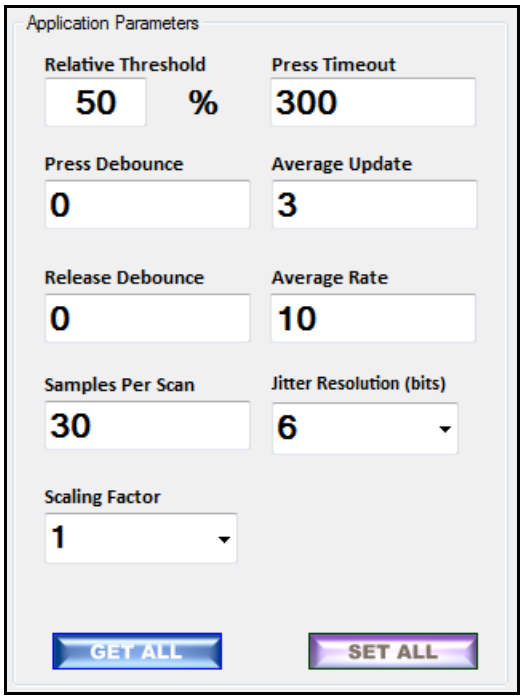
The Sensor Box: This Combo Box allows the user to select the sensor for which the values will be displayed.

Press Threshold: This parameter is the scan value from and above which the mTouch application considers the related sensor in a virtually pressed state. The "virtual" qualifier indicates that the mTouch system detects a press internally, but does not necessarily deliver this information to the outside world (refer to the "Press Debounce" parameter for further details). This Edit Box allows the user to update the Press Threshold for the active sensor at any time. Updating the Press Threshold requires to enter the new value in the corresponding Box, and to Press the "***SET***" button just below it. This action will immediately send the new value to the mTouch device.

Release Threshold: This parameter represents the scan value start from and below which the mTouch frameworks considers the sensor in a "Released" state. This field is read-only. It displays the Release Threshold value for the active sensor (currently selected in the box above). This value depends on both the Press Threshold and the Relative Threshold value, located slightly to the right, in the Application Parameter section. For example, if we have a Press Threshold of 700 and the Relative Threshold is 50%, then the Release Threshold will equal 350.

3.4. The Application Parameter Block:

The Application Parameters group holds all mTouch variables that are not sensor related, but which value, would rather act on the entire system's behavior:



Relative Threshold: This parameter represents the percentage change from the press threshold that will indicate a release. For instance, if the press threshold is 700, we will move the button to a pressed state when the raw value is 700 counts from the baseline. If the relative threshold is 50%, we will move the button to a released state when the raw value is 350 counts or below, from the baseline. This allows us to build hysteresis into the button on/off states. When the user updates this value, the GUI automatically computes the Release Threshold for each sensor, from its corresponding Press Threshold Value and sends the updated parameter to the mTouch device (Refer to Release Threshold described above).

Press Debounce: This parameter indicates how many scans the system (mTouch framework) must see the sensor pressed, before moving it to the pressed state. Practically, once a Press has been initially detected, the sensor has to remain pressed for the specified number of scans, before the application can be officially notified, otherwise the press is ignored.

Release Debounce: This parameter indicates how many scans the mTouch framework must see the sensor in a released state before it officially sets it makes this information available to the outside world (mTouch application). This parameter works the same way the press threshold does. The only difference is that it expects the sensor to remain in the Released state for the specified number of scans, before making the Release information, officially available to the application. The user can specify this last number of scans in the Edit Box.

Samples per Scan: This specifies the number of scans used and combined to build one mTouch Reading for each sensor, which is made available for the user's application. In effect, this represents oversampling. The higher this number is set, the better the sensitivity of all sensors, but the slower the reading's availability.

> **Scan**: The differential processing of 2 ADC samples, combined into a single piece of data.
>
> **Reading**: The result of the combination of a given number of scans, into a single piece of data that is made available to the application.

According to this definition, the previous parameter is rather a ***number of scans per reading***.

Scaling Factor: This parameter determines the amount of post-scaling that is implemented on the acquisition's accumulator register before being stored in the results register. The scaling factor should be set so that it is not possible for the result register to overflow (Max output value is 65535). In simple words, this parameter allows the user to get and set the precision of the raw data after it is combined by the mTouch framework.

Press Time Out: This parameter indicates how many scans that a sensor may be in the pressed state before the mTouch framework "times-out" (automatically release) the button. This mechanism helps avoid stuck buttons. When this value is zero, the button will never timeout.

Average Update: The Box allows the user to get and alter the number of scan cycles before the Baseline of the system is updated and made available to the user's application. It determines the weight given to the raw value when updating the baseline. When the average updates itself using a new reading, this value determines what weight is given to the reading variable in the calculation of the new baseline. The reading variable will have a weight of 1/MTOUCH_BASELINE_WEIGHT in the baseline calculation. Example:
If MTOUCH_BASELINE_WEIGHT is 4, the baseline calculation would be:

$$Baseline = (1/4) * Reading + (3/4) * Baseline$$

Allowed Values: 1 – 4

Average Rate: Sets the update rate of the baseline when un-pressed. This value is used to seed a counter variable that will cause the baseline to be updated only once every n^{th} new reading value, where "n" is MTOUCH_BASELINE_RATE.

Example:
If MTOUCH_BASELINE_RATE is 2, the baseline will use every other new reading to update itself.

Allowed Values: 1 – 65535

Jitter Resolution: In order to avoid scanning signal frequencies that correspond to the system's sampling rate, the mTouch firmware uses a Jittering mechanism, allowing the framework to inject randomness in the system's sampling period. An internal seed I used to control this randomness. This parameter allows the user to specify the precision of the seed in number of bits (1 to 8).

3.4.1. Reading and writing the Application parameters:

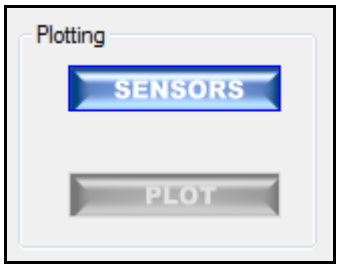
All Applications Parameters can be updated or altered at the same time, using 2 buttons:

GET ALL: This button allows the user to refresh all the Application Parameters at once. This is particularly useful when some or all parameters have been assigned new values but not yet updated. If the user decides to call back the mTouch parameters and roll the display back to the actual values, then all is needed is to press the "**GET ALL"** Button

SET ALL: After one or several parameters have been altered, and once the user is sure the values are correct, clicking this button will immediately send all Application Parameters to the mTouch application. The mTouch device, will instantly take them into account and the system's behavior will be impacted right away.

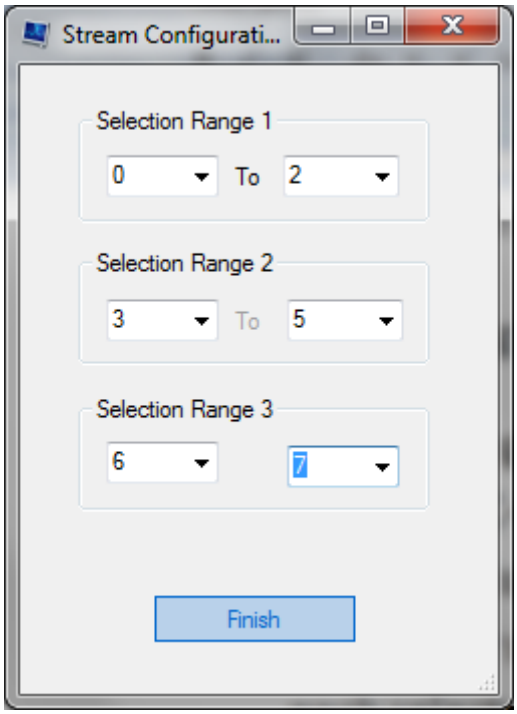
4. Monitoring the mTouch Sensors Behavior, using the Plot Capability of the PC-GUI

Fine-Tuning an mTouch application requires the user to monitor how the sensors behave at all times. The mTouch GUI provides a plot window used to graphically display in real-time, the states of the selected sensors, using multiple display layouts, the user can choose from. Beside the graphical representation of sensor states, the plotter shows the actual values for the raw data, the Baseline, the instantaneous sensitivity for each selected sensor, along with the maximum sensitivity the system has detected for each sensor.



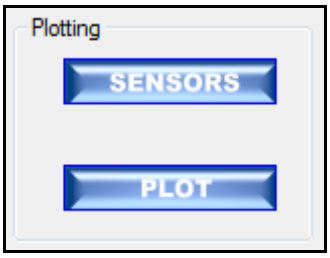
4.1. Selecting the sensors to monitor:

Before the plotter is enabled (grayed "**PLOT**" button above), the user needs to select at least one sensor to monitor. The "**SENSORS**" button allows to choose the set of sensors to debug through the following window:



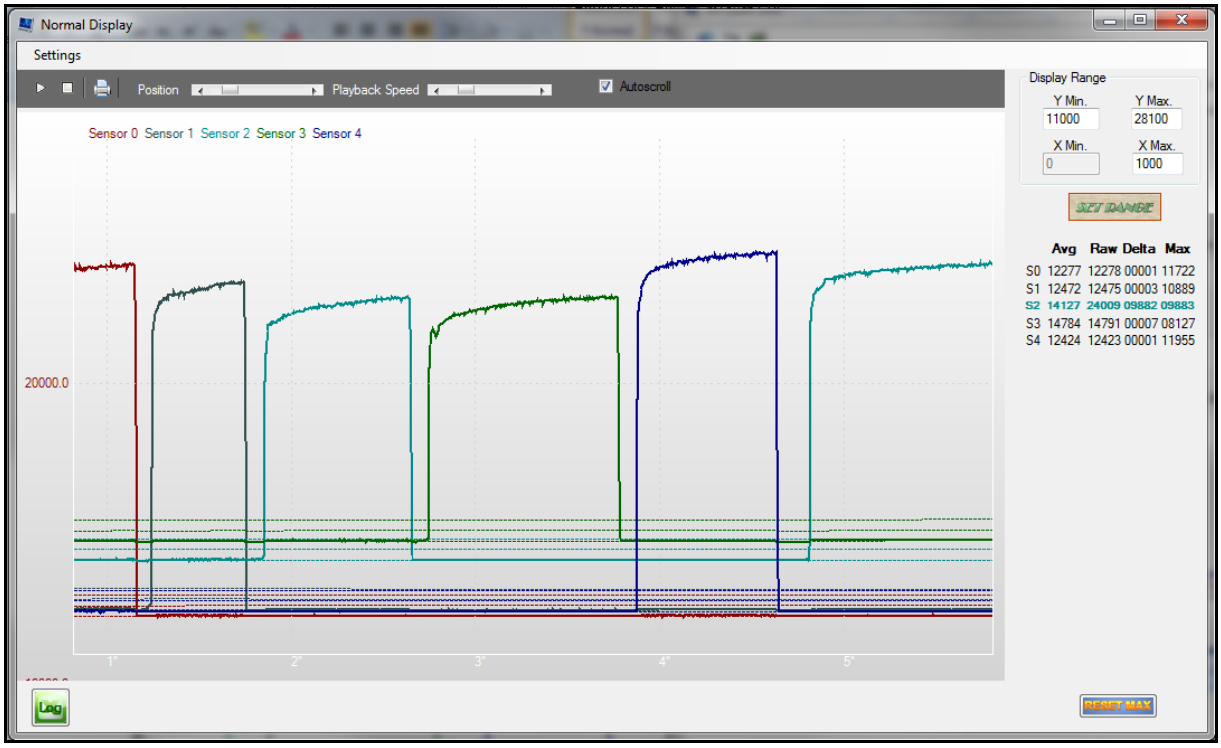
Up-to 3 ranges of sensors can be chosen from the number of available sensors handled by the mTouch application.

Once the sensors have been selected, the user clicks "Finish" which will close the selection window and return to the Main Window, enabling this time the "PLOT" button as shown below:



4.2. Starting the Plot Window:

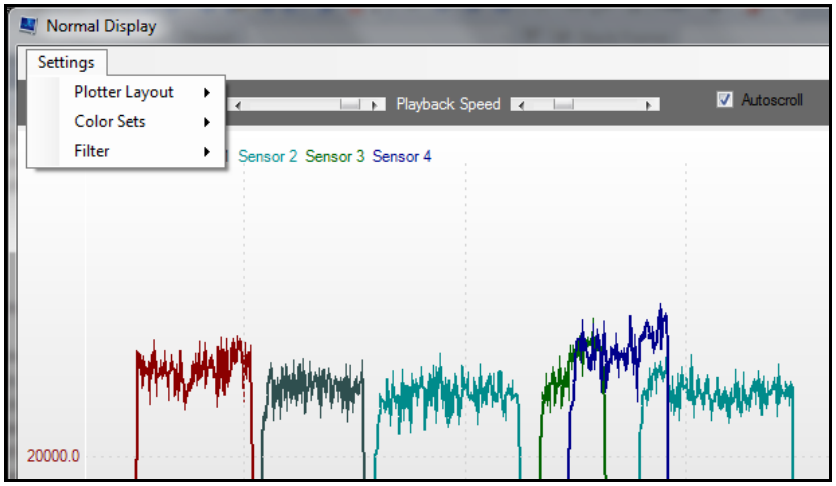
When the user clicks the "**PLOT**" button, the mTouch board is instructed to start streaming data for the selected sensors and the Plot Window opens up as shown below:



4.3. mTouch GUI Plot Window Description:

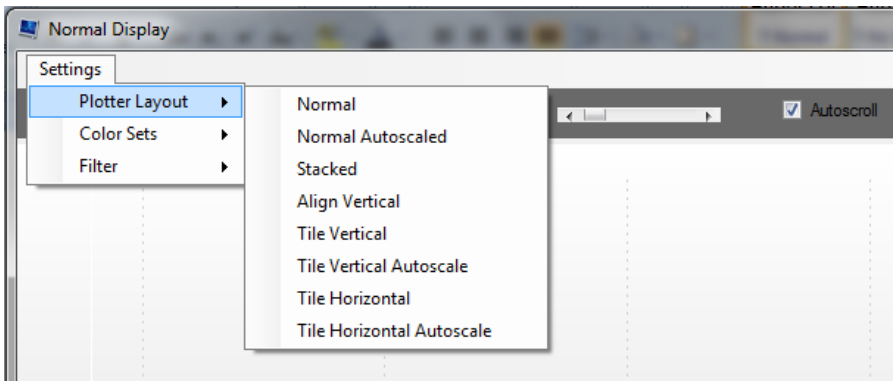
4.3.1. Main Menu:

Settings: This option allows the user to configure the plot window's appearance, as shown by the following figure:

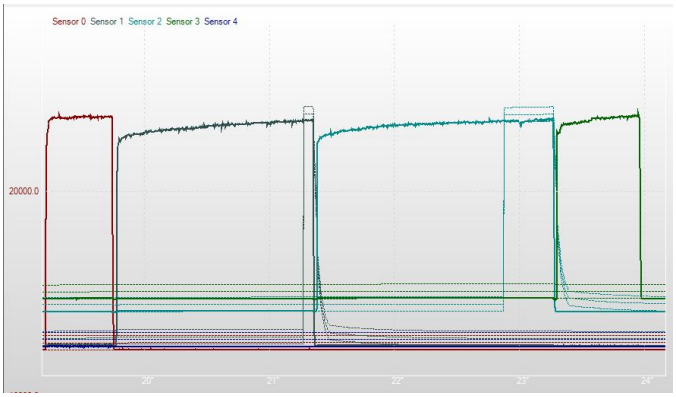


4.3.1.1. Selecting a new plot window layout:

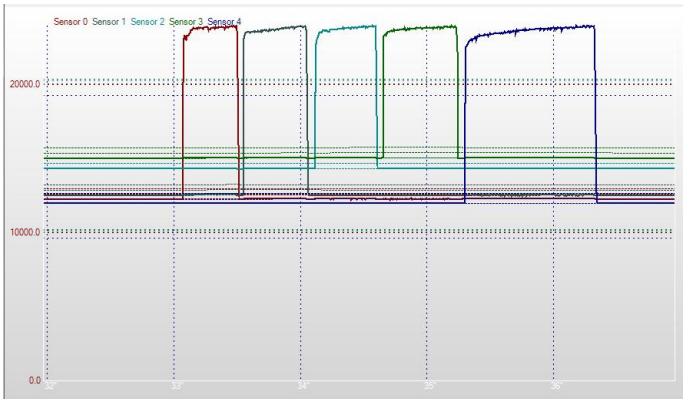
When the plot window opens up, the "Normal" layout is automatically selected by the GUI. The user can still choose from 6 other layouts:



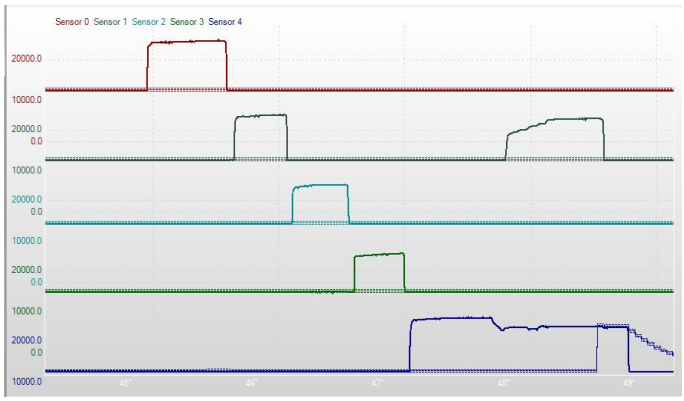
Normal: All graphs are drawn in the same plot window; the user sets the horizontal and vertical scales of the display.



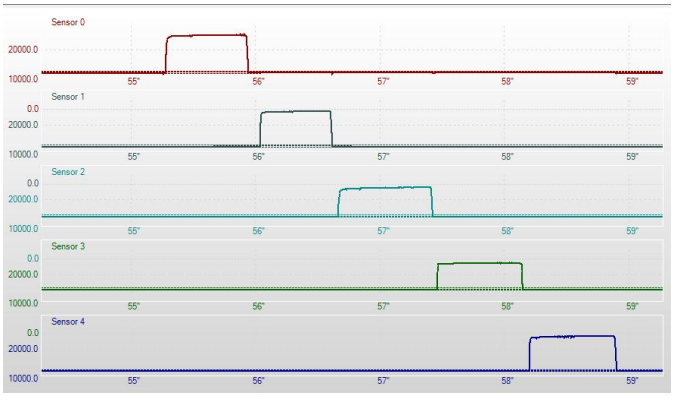
Normal Autoscaled: The graphs are drawn on a single window; the GUI automatically adjusts the plot window scale.



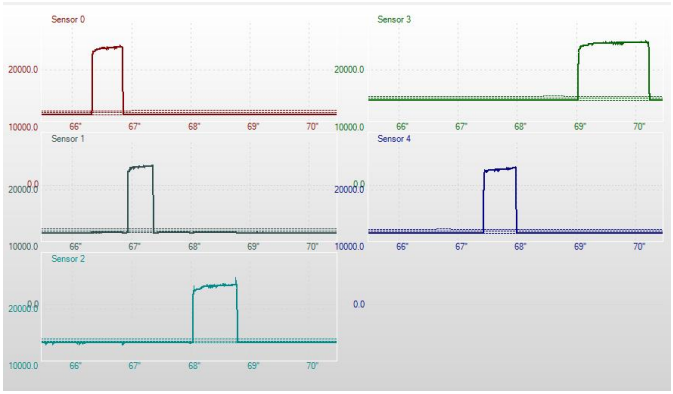
Stacked: The graphs are vertically stacked on the same main plot window. The user still sets the horizontal and vertical scales.



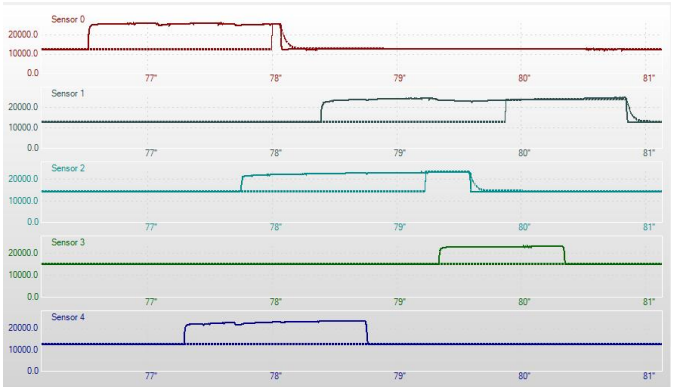
Align Vertical: This layout is similar to the "Stacked" layout with exception that in this mode, each graph has its own X and Y axes.



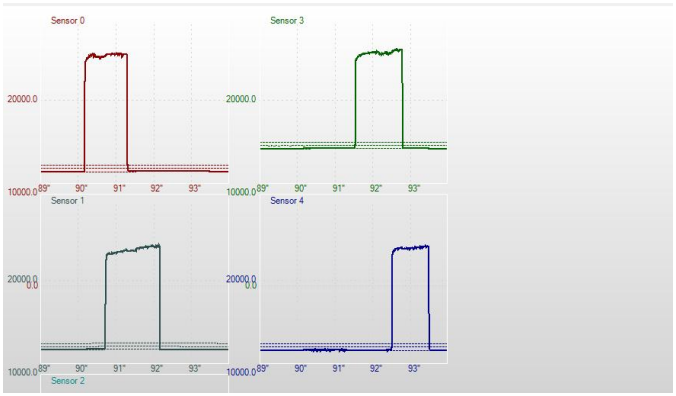
Tile Vertical: In this mode, the graphs are drawn in separate mini-windows (or tiles) each with its own X and Y axes. The tiles are vertically sorted from top to bottom of the plot window. The GUI automatically creates a new column of tiles, once it reaches the bottom of the plot window.



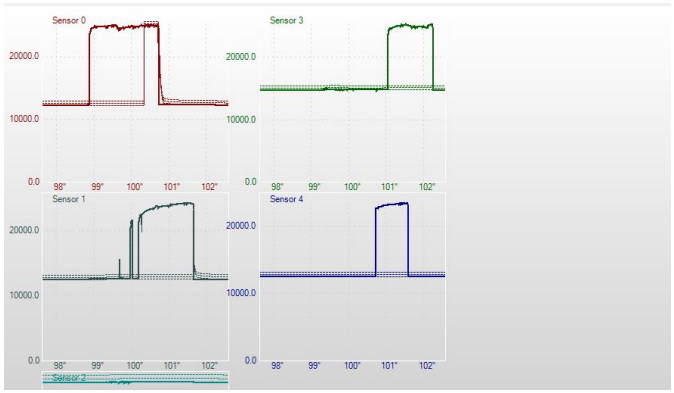
Tile Vertical Autoscale: This mode is similar to the previous "Tile Vertical" layout. The GUI adds here the automatic scaling of the graphs in real-time.



Tile Horizontal: In this layout, the graphs are drawn in independent tiles. The GUI sorts the tiles (graphs) horizontally and creates a new row when the previous graph reaches the right edge of the plot window.

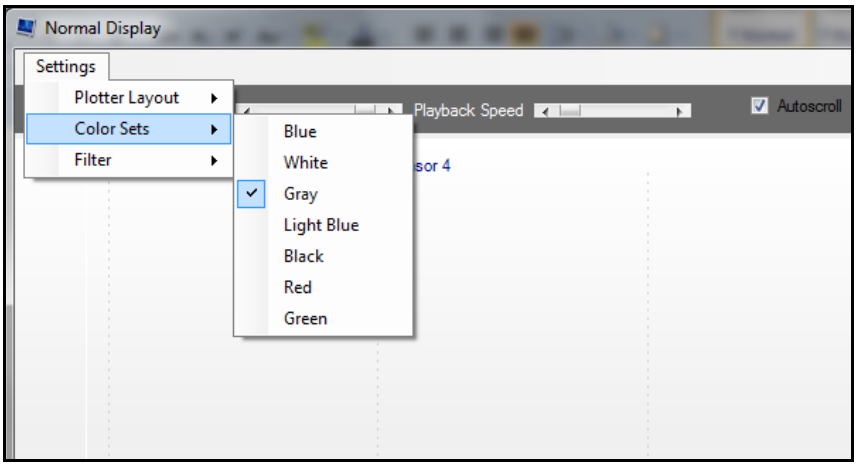


Tile Horizontal Autoscale: This mode is identical to the previous "Tile Horizontal" layout. The GUI automatically adjusts the scale for each tile, in real-time.



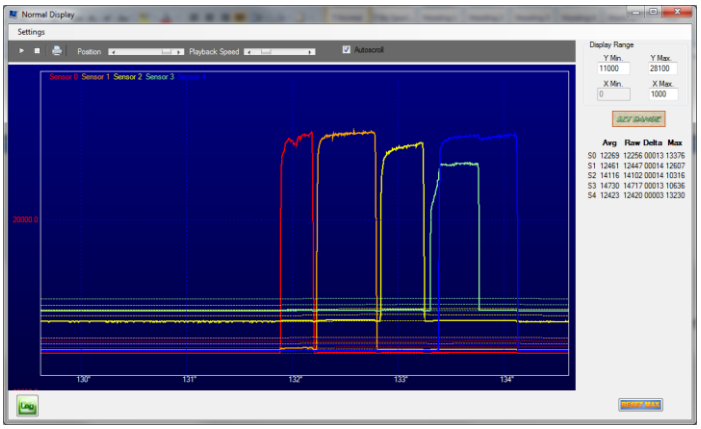
4.3.1.2. Selecting a different Color Set for the Plot Window:

The mTouch GUI allows the user to choose different color sets changing globally the way the plot window looks. All available sets are accessible through the Settings menu as shown below:

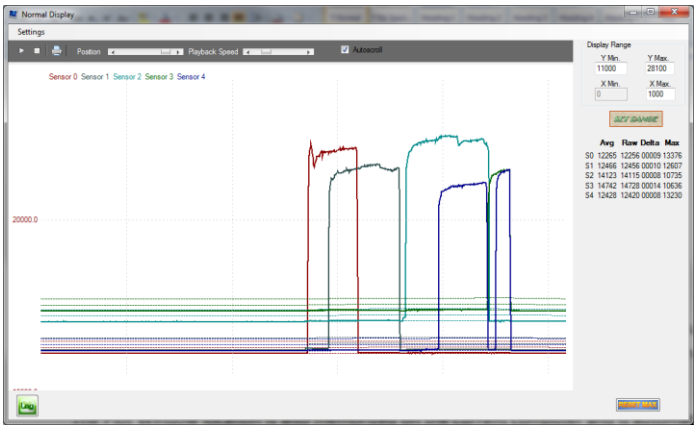


When the plot window is opened, the GUI automatically selects the "Gray" color set. Here is what look, the user can expect when selecting through the remaining color sets:

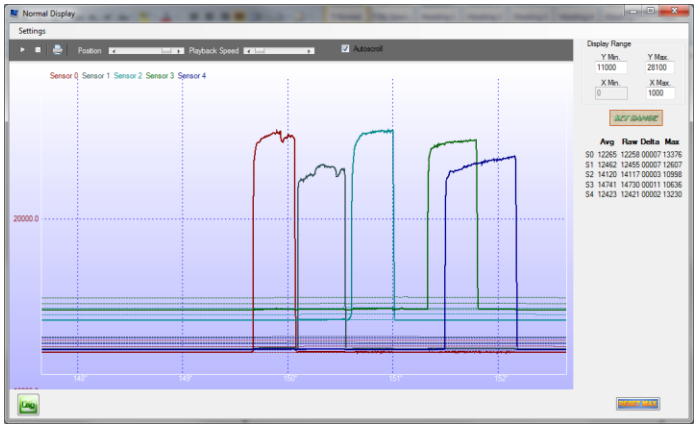
Blue



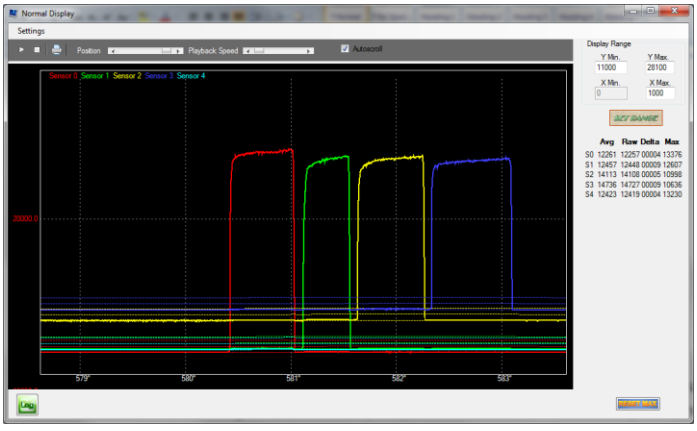
White



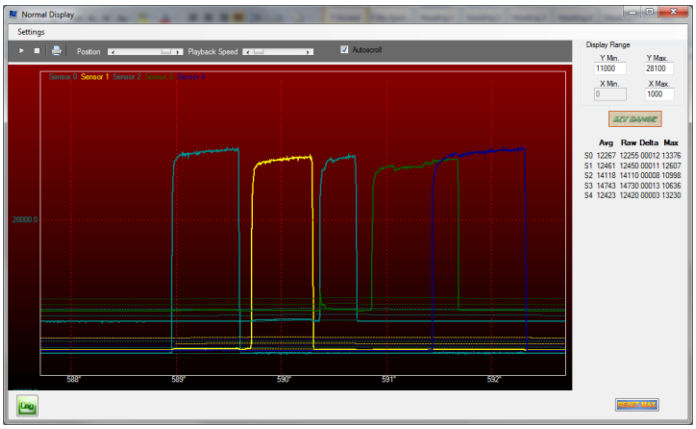
Light Blue

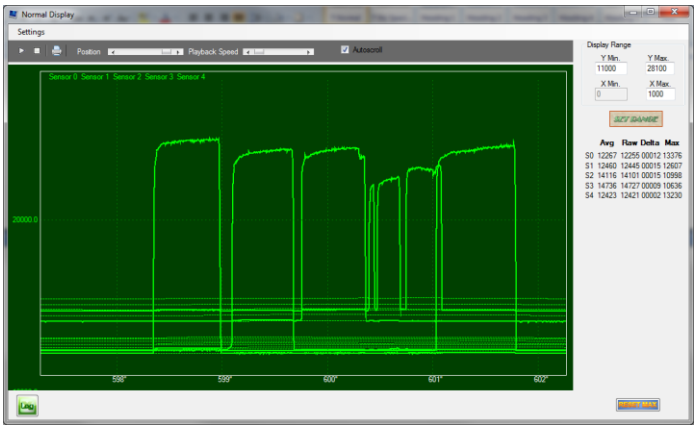


Black



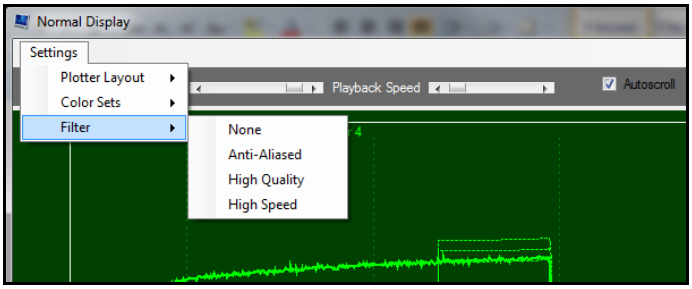
Red **Green**





4.3.1.3. Using the Plotter Filtering Option:

This menu option allows the user to specify a graph smoothing mode, the GUI will apply. This will basically affect both the quality of the graph and the speed of the system.



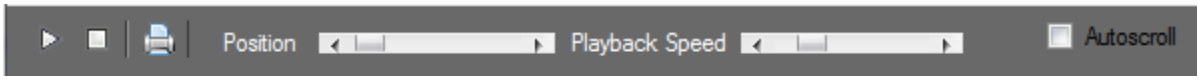
Anti-Aliased: Selects Windows anti-aliased rendering mode.

High Quality: Instructs the plotter engine to use Windows high quality, low speed rendering.

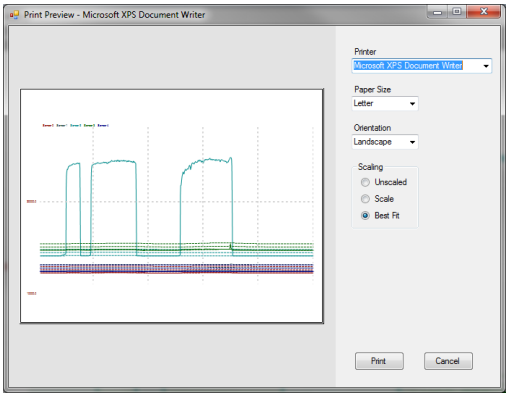
High Speed: Instructs the plotter engine to use Windows high speed, low quality rendering.

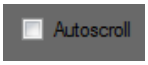
4.3.2. The Plotter Toolbar:

The dark gray toolbar located at the top of the plot window, provide tools to control the plotter behavior. It also provides a print capability of the current plots.



Clicking this icon will call up the print setting window, which will send the current visual to the selected printer as shown:





Un-checking this box will stop the automatic scrolling of the graphs. The plotter will still process the readings as they come in, but without scrolling the display from right to left. Checking back the box will start scrolling from the actual location.

Clicking this icon will start the data playback from the actual location. This button is available only when the data playback is paused or stopped.

Clicking this icon will freeze playback at the current location. This button is available only when data playback is on.

Clicking this icon will stop the on-going playback but will also reset the playback pointer to the beginning of the stream, when the plot window was opened. Next time the playback button is clicked, playback will start from the beginning.

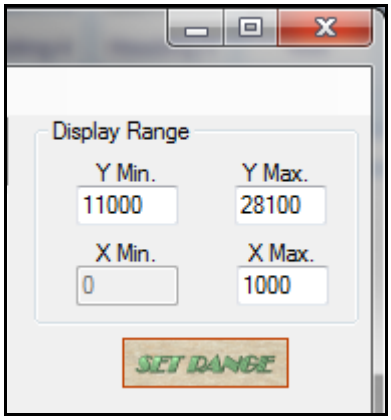
Once data playback has been stopped, the user can drag the display area using the mouse (from left to right) to call up whatever data has been recorded since the plot window was started.

This slider allows the user to rapidly reposition the display range to any previously recorded location, up to when the plot window was started.

This slider allows the user to the scrolling speed of the plots, when data playback has been initiated.

4.3.3. Manually setting the plotter's display range:

When monitoring with any of the "Manual" (non-autoscale) layouts, the user can adjust the display range which will allow to basically zoom in and out f the displayed graphs. On the top right side of the plot windows, the GUI user can set the horizontal and vertical ranges, depending on the hardware used for the Debug session. Once the ranges have been set, they are saved on the current computer and are automatically reloaded and set, each time the GUI is re-started



The Y-Range should be set after a few presses on the sensors, once the user has determined how big the shift is on the mTouch device they are optimizing.

The X-Range represents the number of samples that the GUI rendering engine will fit into the visual range of the plot window. The example above shows that the plotter will fit the next 1000 samples it is

going to receive from the mTouch device into the visual range, then it is going to start shifting the graphs to the left.

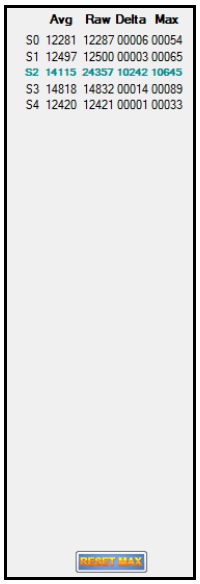
The user sets the desired values into the corresponding edit boxes, and then clicks the "SET RANGE" button to get the new values immediately taken into consideration and also memorized for the next time the plotter is started.

4.3.4. Real-Time data display area:

On the right side of the plot window and in the middle, the GUI displays for each selected sensor its current:

- Raw reading
- The actual Baseline's value
- The current sensitivity (Delta between the raw and average values)
- The maximum sensitivity detected from either the time the plotter was started, or the last Maximum reset action ("RESET MAX" button)

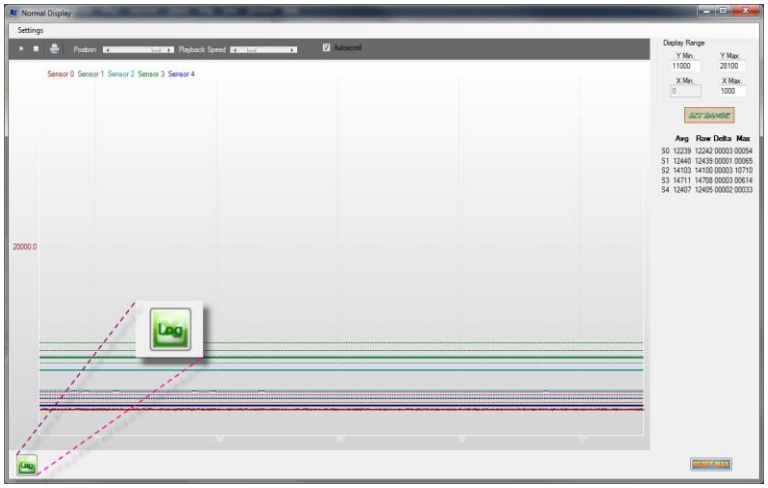
When the user presses a given sensor, the plotter highlights the corresponding numbers in the sensor's color, for easy follow up, as shown below:



4.3.5. Saving the streams to the hard drive:

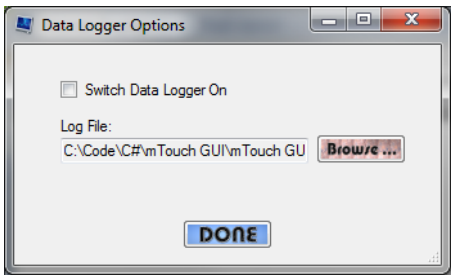
The GUI also provides the nice capability of saving the incoming data contained in the stream (for the selected sensors) into a Microsoft Excel file (CSV format). This allows to track specific events among large amounts of data (long debug sessions), as it allows to get a quick graphical overview of the saved streams using Microsoft Excel.

To start the data logger, the user needs to specify a file name as well as a destination on the Hard drive. If nothing is provided, the data logger will use a default "mTouch Log.csv" file that will be created in the GUI application's directory. In order to specify a file name and to start logging data, the user needs to click the "Log" Button, located in the bottom-left corner of the plot window:

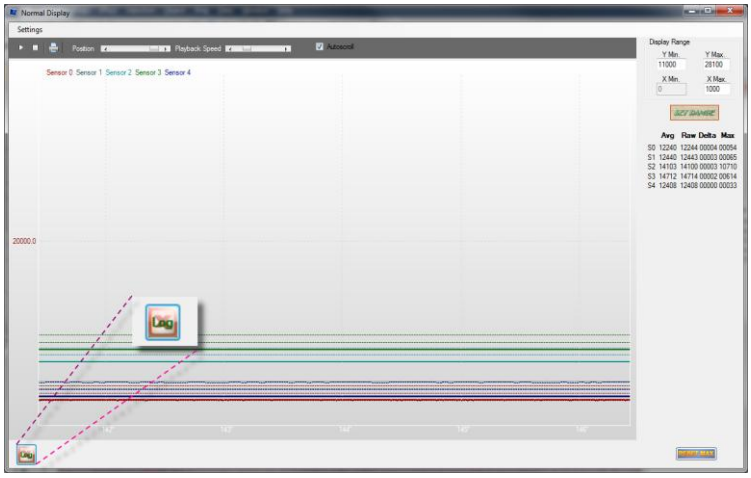


This opens up a small window allowing to:

- Specify a file name and location, or let the data logger automatically overwrite an existing one.
- Switch the actual logging function on and off.



Clicking the "Switch Data Logger On" box will immediately start recording the data to the hard drive and turn the previous "Log" button's color from green to brown, informing the user that the stream is being logged onto the hard drive, once the "Done" button above is pressed and the window closed.



The Plot Window location is also memorized on the current computer and is automatically used the next time the GUI is started and the Plotter opened.

The GUI allows simultaneous access to the Main window while the Plot window is opened. This very nice feature allows the user to adjust the parameters on the main window, and instantly watch the impact and changes on the graphs depending on what parameter was altered.