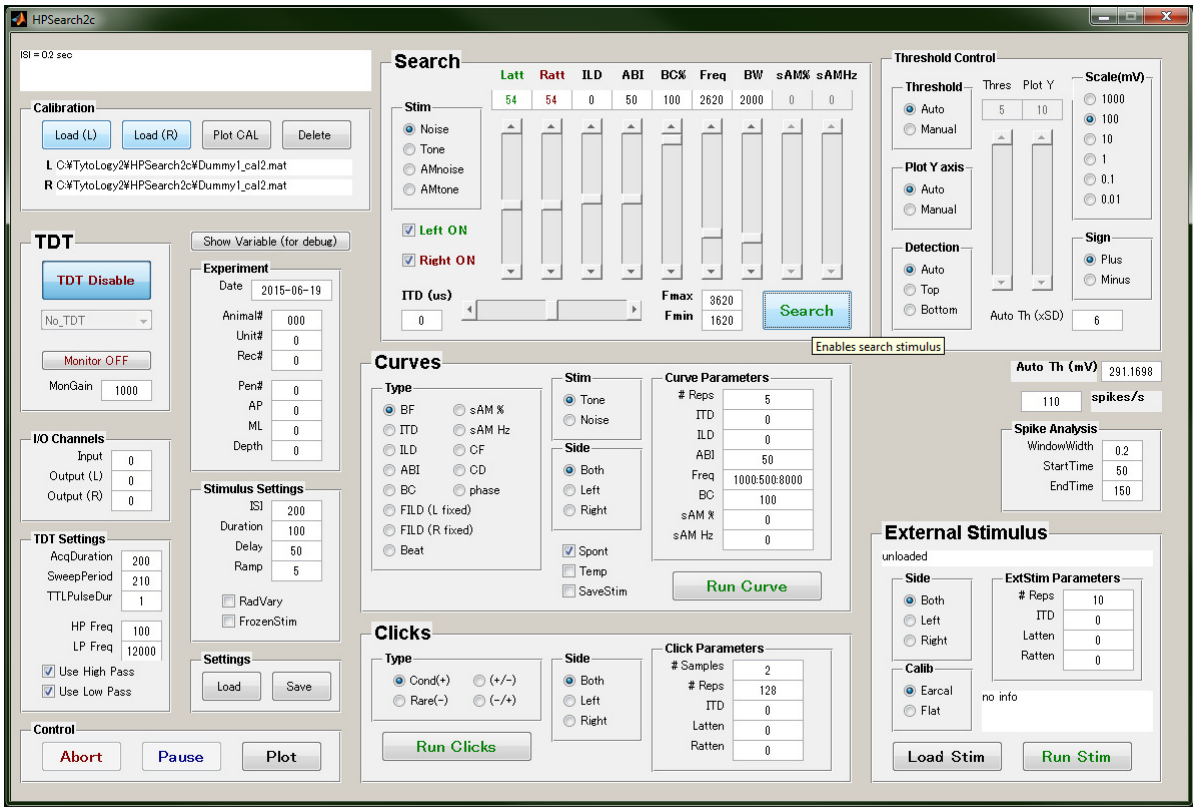
TytoLogy2 functions

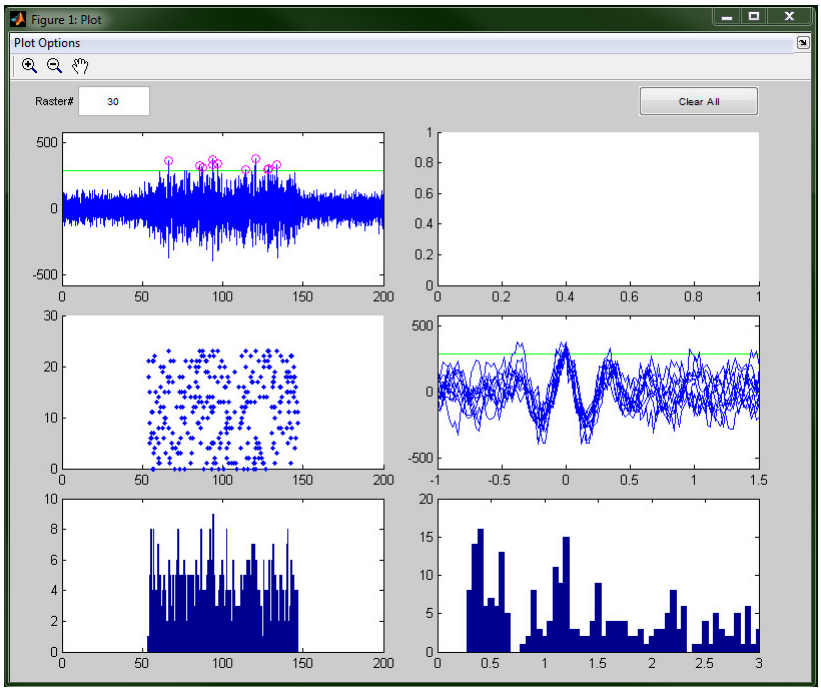
December 2015

Go Ashida (go.ashida@uni-oldenburg.de)

HPSearch2 - Main Window



HPSearch2 - Plot Window



Standard Flow of Experiments

*** Before Starting Experiments**

+ TDT hardware is already set-up and properly wired to the computer and other hardware.
+ Miniature microphones have been calibrated with **`MicrophoneCal2`**.
+ Microphone calibration data are saved as a Matlab file: **`filename_fr2.mat`**.

*** Calibrating Earphones**

+ Set the earphones (headphones) to the animal, open **`HeadphoneCal2`**, set the microphone calibration data file, set frequency ranges and other parameters, and calibrate the earphones.
+ Headphone calibration data are saved as a Matlab file: **`filename_cal2.mat`**.

*** Starting HPSearch2c and TDT**

+ Move to a directory where you want to save your data.
+ Open **`HPSearch2c`**.
+ Select your TDT hardware from the dropdown list on the **TDT** module of the GUI.
+ Check if your TDT hardware is turned on.
+ Start TDT by pressing the **`[TDT Enable]`** button.
+ Check if your TDT is running (see the processor usage on the your TDT hardware).

*** Setting-up Calibration Files and Parameters**

+ Load your earphone calibration file(s) using the **Calibration** module of the GUI.
+ Set I/O channels, TDT settings, Experiment Settings, Stimulus Settings in the GUI.
+ You may load your settings file to reuse your previous setting information.

*** Running Search Stimuli**

+ Turn on your earphones by clicking the **`[Left ON]`** and **`[Right ON]`** checkboxes in the **Search** module of the GUI.
+ Choose stimulus type and set frequency, intensity, ITD, etc.
+ Click the **`[Search]`** button to play sounds.
+ Sound stimuli are repeated until you stop them by clicking the **`[Search]`** button again.
+ Recorded neural data are shown in the plotting window, in which you can specify which panel to show.

*** Setting Spike Discrimination Thresholds**

+ Using the **Threshold Control** module, you can set the threshold of the built-in spike detector.
+ In the **Spike Analysis** module you can specify the analysis time range of your interest. Saved spike statistics is based on these parameters.

*** Recording Neural Responses**

+ The **Curve** module is for playing sounds (tone, noise, etc.) with varied parameters (frequency, ITD, intensity, etc.) and recording neural responses.
+ The **Click** module is for playing click stimuli and recording neural responses.
+ The **External Stimulus** module (beta version) is for playing external sound file and recording neural responses. You have to load your stimulus file before playing it.
+ Specify the desired parameters in these modules and **Stimulus Setting** module.
+ Click **`[Run Curve]`**, **`[Run Clicks]`**, or **`[Run Stim]`** button to start playing/recording.
+ Recorded data are saved in a Matlab file (only if the recording is not aborted) and in a binary file (**`.dat2`**).

*** Closing HPSearch2c and TDT**

+ By closing the main GUI window, the program automatically calls corresponding functions to stop TDT circuit and close the connection to the TDT hardware.

HPSearch2c Functions

*** Some Important Notes**

+ All parameters are stored in the structure named **`handles`**. To save parameters, you have to call
```
guidata(hObject, handles);
```
before returning from a sub-function.

+ Most TDT-related parameters are defined in **`HPSearch2c_config`** and stored under **`handles.h2.config`**.
+ Other parameters are defined in **`HPSearch2c_init`** and stored under corresponding entries of **`handles.h2`**.

+ In the following, RX6 is used as TDT hardware.
+ If other hardware is used, see corresponding functions defined in **`HPSearch2c_config`**.

*** Initialization - Opening GUI & Setup**

[Functions]	**[Descriptions]**
`HPSearch2c.HPSearch2c_OpeningFcn`	- This function is called when the GUI is opened.
`HPSearch2c_Opening`	- Setting up handles and parameters
`HPSearch2c_config`	- This function defines TDT-related parameters (e.g., circuit path and name, I/O functions)
`HPSearch2c_init`	- This function defines HPSearch2 parameters (e.g., search/curve parameters and limits, thresholds, etc.)

*** Finalization - Closing TDT & Cleaning-up**

[Functions]	**[Descriptions]**
`HPSearch2c_CloseRequestFcn`	- This function is called when the GUI is closed
`HPSearch2c_Closing`	- This function checks if TDT is running.
`HPSearch2c_TDTclose`	- If TDT is running, then the connection is closed.

*** Abort Button**

[Functions]	**[Descriptions]**
`buttonAbort_Callback`	- The on/off state of this button works as a flag for aborting.

+ The state of this button is monitored by Curve/Click functions.
+ If the button is detected to be ON, then the current recording is aborted.

*** Pause Button**

[Functions]	**[Descriptions]**
`buttonPause_Callback`	- The on/off state of this button works as a flag for stopping.

+ The state of this button is monitored by Curve/Click functions.
+ If the button is ON, then the current recording is stopped until the user presses the button again to make it OFF.

*** Plot Button**

[Functions]	**[Descriptions]**
`buttonCallPlot_Callback`	- This function calls an external plot function.
`TytoView_simpleView`	- Function for quickly seeing HPSearch data.

*** Show Variable Button**

[Functions]	**[Descriptions]**
buttonShowVal_Callback	- This function enables retrieving hidden variables.

*** Search Button**

[Functions]	**[Descriptions]**
buttonSearch_Callback	- This function is called when the **[Search]** button is pressed.
HPSearch2c_enableUIs	- Enabling/disabling GUI
HPSearch2c_Search	- Main function for playing search stimuli
***** Setting TDT parameters (duration, delay, etc.) *****	
HPSearch2c_RX6settings	- Defined as **handles.h2.config.TDTsetFunc**.
HPSearch2c_RX8settings	- Internally called by **HPSearch2c_RX6settings**.
RPsamplefreq	- This function gets the sampling frequency of the TDT hardware.
	(Location = C:\TytoLogy\toolbox\TDT\Functions\RP\RPsamplefreq.m)
RPsettag	- This function sets various parameters to TDT hardware.
	(Location = C:\TytoLogy\toolbox\TDT\Functions\RP\RPsettag.m)
***** Making stimulus waveforms and setting the attenuators *****	
HPSearch2c_searchParamFromUI	- Getting search parameter values from the GUI.
TytoLogy2_mergecal	- Merging L and R calibration data.
TytoLogy2_interpcal	- Interpolating calibration data to fit the stimulus frequency profile
syn_headphone_tone	- Making tonal stimuli
syn_headphone_noise	- Making noise stimuli
syn_headphone_amtone	- Making AM tone stimuli
syn_headphone_amnoise	- Making AM noise stimuli
computeLRspl	- Calculating sound intensities (SPL) from ABI and ILD
TytoLogy2_figureAtten	- Calculating attenuation levels from SPL and calibration data
PA5setatten	- Defined as **handles.h2.config.setattenFunc**. Setting attenuation to TDT PA5 hardware
***** Playing sounds and recording data *****	
HPSearch2c_spikeio	- Defined as **handles.h2.config.ioFunc**. Playing sounds and recording neural responses.
RPwriteV	- Setting output stimulus to TDT
RPtrig	- Sending trigger command to TDT
RPfastgettag	- Checking if stimulus sweep ends
RPgettag	- Getting index of recorded data from TDT
RPreadV	- Getting recorded data from TDT
*** Plotting data ***	
HPSearch2c_plotParamFromUI	- Getting plot/threshold settings from the GUI
HPSearch2c_spikedetect	- Spike detection using the threshold settings
HPSearch2c_plotResponse	- Plotting traces, rasters, PSTH, ISIH, etc.

*** Curves Button**

[Functions]	**[Descriptions]**
buttonCurve_Callback	- This function is called when **[Run Curve]** button is pressed.
HPSearch2c_enableUIs	- Enabling/disabling GUI
HPSearch2c_Search	- Main function for recording
***** Setting TDT parameters (duration, delay, etc.) *****	
HPSearch2c_RX6settings	- Defined as **handles.h2.config.TDTsetFunc**.
HPSearch2c_RX8settings	- Internally called by **HPSearch2c_RX6settings**.
RPsamplefreq	- This function gets the sampling frequency of the TDT hardware.
RPsettag	- This function sets various parameters to TDT hardware.
***** Setting calibration data and checking frequency and intensity parameters *****	
TytoLogy2_mergecal	- Merging L and R calibration data.
TytoLogy2_interpcal	- Interpolating calibration data to fit the stimulus frequency profile
***** Setting Loop variables (freq, ITD, etc.) *****	
-- Data are stored in **loopvars**.	
***** Setting output file names *****	
TytoLogy2_buildFileName	
***** Storing stimulus parameters and randomizing order *****	
-- Data are stored in **stimcache**.	
***** Making stimulus waveforms *****	
syn_headphone_tone	- Making tonal stimuli
syn_headphone_noise	- Making noise stimuli
syn_headphone_amtone	- Making AM tone stimuli
syn_headphone_amnoise	- Making AM noise stimuli
***** Calculating attenuation levels *****	
computeLRspl	- Calculating sound intensities (SPL) from ABI and ILD
TytoLogy2_figureAtten	- Calculating attenuation levels from SPL and calibration data
***** Setting up binary data file *****	
TytoLogy2_writebinary	- Writing headers of binary data file
***** Getting spontaneous response data *****	
PA5setatten	- Defined as **handles.h2.config.setattenFunc**.
HPSearch2c_spikeio	- Defined as **handles.h2.config.ioFunc**.
***** Main loop: playing sounds and recording data *****	
PA5setatten	- Defined as **handles.h2.config.setattenFunc**.
HPSearch2c_spikeio	- Defined as **handles.h2.config.ioFunc**. Playing sounds and recording neural responses.
RPwriteV	- Setting output stimulus to TDT
RPtrig	- Sending trigger command to TDT
RPfastgettag	- Checking if stimulus sweep ends
RPgettag	- Getting index of recorded data from TDT
RPreadV	- Getting recorded data from TDT
TytoLogy2_writebinary	- Writing binary output data
*** Plotting data ***	
HPSearch2c_plotParamFromUI	- Getting plot/threshold settings from the GUI
HPSearch2c_spikedetect	- Spike detection using the threshold settings
HPSearch2c_plotResponse	- Plotting traces, rasters, PSTH, ISIH, etc.
HPSearch2c_plotCurve	- Plotting tuning curve
*** Check pause and abort flags, stop to fit ISI, and repeat ***	
-- If the abort flag is set, then recording is aborted.	
*** Saving and plotting ***	
TytoView_simpleplot	- Plotting recorded data (for quick checking)

*** Clicks Button**

[Functions]	**[Descriptions]**
buttonClick_Callback	- This function is called when **[Run Clicks]** button is pressed.
HPSearch2c_enableUIs	- Enabling/disabling GUI
HPSearch2c_Click	- Main function for playing clicks and recording
***** Setting TDT parameters (duration, delay, etc.) *****	
HPSearch2c_RX6settings	- Defined as **handles.h2.config.TDTsetFunc**.
HPSearch2c_RX8settings	- Internally called by **HPSearch2c_RX6settings**.
RPsamplefreq	- This function gets the sampling frequency of the TDT hardware.
RPsettag	- This function sets various parameters to TDT hardware.
***** Setting output file names *****	
TytoLogy2_buildFileName	
***** Setting loop variables (reps, ITD, etc.) *****	
-- Data are stored in **loopvars**.	
***** Setting up binary data file *****	
TytoLogy2_writebinary	- Writing headers of binary data file
***** Getting spontaneous response data *****	
PA5setatten	- Defined as **handles.h2.config.setattenFunc**.
HPSearch2c_spikeio	- Defined as **handles.h2.config.ioFunc**.
***** Main loop: playing sounds and recording data *****	
syn_headphone_click	- Making tonal stimuli
PA5setatten	- Defined as **handles.h2.config.setattenFunc**.
HPSearch2c_spikeio	- Defined as **handles.h2.config.ioFunc**. Playing sounds and recording neural responses.
RPwriteV	- Setting output stimulus to TDT
RPtrig	- Sending trigger command to TDT
RPfastgettag	- Checking if stimulus sweep ends
RPgettag	- Getting index of recorded data from TDT
RPreadV	- Getting recorded data from TDT
TytoLogy2_writebinary	- Writing binary output data
*** Plotting data ***	
HPSearch2c_plotParamFromUI	- Getting plot/threshold settings from the GUI
HPSearch2c_spikedetect	- Spike detection using the threshold settings
HPSearch2c_plotResponse	- Plotting traces, rasters, PSTH, ISIH, etc.
HPSearch2c_plotCurve	- Plotting tuning curve
*** Check pause and abort flags, stop to fit ISI, and repeat ***	
-- If the abort flag is set, then recording is aborted.	
*** Saving and plotting ***	
TytoView_clickplot	- Plotting recorded click response data (for quick checking)

*** Run Stim Button (External Stimulus module)**

[Functions]	**[Descriptions]**
buttonExtStimRun_Callback	- This function is called when **[Run Stim]** button is pressed.
HPSearch2c_enableUIs	- Enabling/disabling GUI
HPSearch2c_RunStim	- Main function for playing external stimulus and recording
***** Setting output file names *****	
TytoLogy2_buildFileName	
***** Setting TDT parameters (duration, delay, etc.) *****	
HPSearch2c_RX6settings	- Defined as **handles.h2.config.TDTsetFunc**.
HPSearch2c_RX8settings	- Internally called by **HPSearch2c_RX6settings**.
RPsamplefreq	- This function gets the sampling frequency of the TDT hardware.
RPsettag	- This function sets various parameters to TDT hardware.
***** Setting calibration data and checking frequency and intensity parameters *****	
TytoLogy2_mergecal	- Merging L and R calibration data.
***** Setting up binary data file *****	
TytoLogy2_writebinary	- Writing headers of binary data file
***** Setting attenuator *****	
PA5setatten	- Defined as **handles.h2.config.setattenFunc**.
***** Getting spontaneous response data *****	
HPSearch2c_spikeio	- Defined as **handles.h2.config.ioFunc**.
***** Main loop: playing sounds and recording data *****	
PA5setatten	- Defined as **handles.h2.config.setattenFunc**.
HPSearch2c_spikeio	- Defined as **handles.h2.config.ioFunc**. Playing sounds and recording neural responses.
RPwriteV	- Setting output stimulus to TDT
RPtrig	- Sending trigger command to TDT
RPfastgettag	- Checking if stimulus sweep ends
RPgettag	- Getting index of recorded data from TDT
RPreadV	- Getting recorded data from TDT
TytoLogy2_writebinary	- Writing binary output data
*** Plotting data ***	
HPSearch2c_plotParamFromUI	- Getting plot/threshold settings from the GUI
HPSearch2c_spikedetect	- Spike detection using the threshold settings
HPSearch2c_plotResponse	- Plotting traces, rasters, PSTH, ISIH, etc.