StdpC 2007

(Spike timing dependent plasticity Clamp),

in parts based on **DYNCLAMP2** [Pinto et al., 2001]

MANUAL

Thomas Nowotny

Centre for Computational Neuroscience and Robotics,
Department of Informatics,
University of Sussex,
Brighton BN1 9QJ, UK
E-mail: t.nowotny@sussex.ac.uk September 2008,

Original DYNCLAMP2 developers:

Reynaldo Daniel Pinto,
Robert C. Elson,
Attila Szücs,
M. I. Rabinovich,
A. I. Selverston,
H. D. I. Abarbanel

Institute for Nonlinear Science,
University of California, San Diego,
9500 Gilman Dr.
Mail Code 0402
La Jolla, CA 92093-0402, USA

StdpC 2007 is software building on the ideas that led to the development of the earlier DYN-CLAMP2 software. The new software is based on QT and has in addition to the original native support for the old DigiData 1200/A interface now also support for all National Instruments devices that support NI's NIDAQmx API.

This is the first beta version of the software. Some functionality has not been tested thouroughly. Any feedback on problems with the software or potential bugs is greatly appreciated. An older version of the software (StdpC) is described in Journal of Neuroscience Methods [?]. If you use StdpC 2007 and publish results based on its use, please cite the website and this paper. Please contact us for any further questions (t.nowotny@sussex.ac.uk).

1 Introduction

The Dynamic Clamp protocol, developed by [Sharp et al., 1993] and independently by [Robinson and Kawai, 1993] allows inserting simulated membrane conductances into cells, and/or adding synapses between cells as well as a variety of other manipulations. Here, we describe StdpC 2007, a software for controlling biological neurons and an internal spike generator in (soft) real time.

The software allows to define up to 6 artificial synapses, 6 Hodgkin-Huxley conductances [Hodgkin et al., 1949] and one presynaptic artificial neuron (spike generator). The parameters of these simulated entities can be controlled on the graphical user interface (GUI) or through script files.

The software supports the DIGIDATA 1200A data acquisition card of Axon Instruments, Inc (now part of MDS Analytical Technologies), all NIDAQmx capable devices of National Instruments, and a pure simulation mode for testing. The program was written in C++ (using the Bloodshed Dev-C++ environment which is based on the popular gcc compiler (mingw)). The graphical user interface is based on Trolltech's QT libraries. The program has been tested on Windows XP but should also be compatible with earlier versions of Windows (Windows NT, Windows 2000). The actual performance of the dynamic clamp cycle depends on the speed of the PC hardware used. The DIGIDATA driver is based on a modified/extended version of PortTalk by Craig Peacock and is unlikely to work in Windows Vista. The NIDAQmx based driver may work with Windows Vista but has not been tested in this environment.

2 Hardware and software requirements

A standard PC (Pentium IV and up recommended), Windows XP (potentially Windows NT, 2000 ok). A DIGIDATA 1200A ADC/DAC board from Axon Instruments, or a National Instruments DAQ and installed NIDAQmx driver.

The program was tested on a PC with Windows XP professional and a DigiData 1200A board as well as with a NI PCI-MIO-16-4 DAQ using NIDAQmx for Windows, version 8.7.2.

3 Asynchronous Dynamic Clamp Cycle

The dynamic clamp protocol consists of a cycle of reading the membrane potential of the cells, calculating the current to be injected into the cells according to the synapses and conductances that are to be simulated, and generating the voltage commands that will generate the currents. StdpC aims at repeating this cycle at the maximal rate supported by the hardware in order to simulate continuous biological processes. Since Windows is not a real time operating system, the actual time from one cycle to the next can not be controlled in a strict way. Instead of aiming at such a control StdpC is based on an asynchronous dynamic clamp cycle. Every time the program updates the membrane potential of the cells it also reads the real time clock in the Digidata 1200A board (the system clock in case of using NI devices) which allows to calculate the duration of the previous Dynamic Clamp cycle. The measured time intervals between voltage updates are used in the computation of the currents. This limits the impact of unequal delays during repeated cycles unless these delays become large compared to the biological time scales in the system studied. As a side effect of the asynchronous Dynamic Clamp cycle, StdpC update cycles will always run at the maximume rate supported by the hardware. With the typical successive upgrade of computer hardware the quality of the Dynamic Clamp interaction will improve over time.

4 How to install the software

4.1 Binary package

Download the package and unzip it to a convenient location, e.g., C:\ Program Files\ StdpC. Then execute “StdpC.exe”.

4.2 Source package

Download the source package and unzip it to a convenient location. Open a shell and execute “myqmake.bat”. Then open the dev-c++ project “StdC.dev” and compile. Make sure that the Makefile generated by qmake is used. Note that you will need Dev-C++/mingw and QT 4 installed on your computer. If you are compiling for the NIDAQmx interface you need to have NIDAQmx installed and to enable the option “NIDAQmx” in the “.config” file.

5 Troublshooting

5.1 I/O conflicts

In the past, there have been problems with I/O conflicts between the DigiData board and other hardware devices. If you are experiencing problems, you may need to change the base address of the DigiData board. Hardware conflicts can occur even if the DIGIDATA board works fine with Axon programs like Axoscope because, unlike these, StdpC 2007 does read the Real Time Counter (RTC) in the Digidata board. Usually, the DigiData board is configured to use the I/O base address 0x320, and in this case the RTC reading ports are in the range of 0x330 - 0x332. Many computers use the address 0x330 for some devices like sound cards, etc.

To check for and resolve a hardware address conflict, *Windows NT, 2000, XP:*
Open: Start → Programs → Administrative Tools → Windows NT Diagnostics. Choose the folder ⟨Resources⟩, click in ⟨I/O Port⟩ and use the cursor to browse the list of used ports. You should look for 330. Note that 320 will not appear because the Digidata Board is not detected by Windows. If you find any reference in the range 330 - 33F and your Digidata board is configured to use the base address 320 you need to change I/O address of your Digidata board to run DynClamp and avoid other problems due to the I/O conflict.
Windows 95, 98:
Open: Start → Settings → Control Panel → System. Choose the folder ⟨Device Manager⟩, click ⟨I/O Port⟩ and use the cursor to browse the list of used ports. You should look for 330. Note that 320 will not appear because the Digidata Board is not detected by Windows. If you find any reference in the range 330 - 33F and your Digidata board is configured to use the base address 320 you need to change the I/O address of your Digidata board to run DynClamp and avoid other problems due to the I/O conflict.
Changing the base address of the Digidata Board and in the StdpC program:
Identify a free range of port addresses. If 0x330 is in use, addresses around 340-370 are usually available which is suifficient for the Digidata ports. Reconfigure the Digidata board for the base address 340 or 350 manually (the RTC will be 350 or 360 and no conflicts!) and run the StdpC program. The Digidata base address can be changed within StdpC on the Settings-¿DAQ dialog.

6 Changes from previous versions/programs

6.1 Changes DYNCLAMP2 to StdpC version 3

The following is a list of features which were new in StdpC v3:

- *Spike generator*
  A spike generator can replace a biological presynaptic neuron. Spikes are generated either periodically in a fixed pattern or from a file containing predefined spike times. This feature is actually an original feature of R. Pinto but was not published yet.
- *Hidden parameter panels*
  To de-clutter the screen of the dynamic clamp computer, in StdpC the parameter dialogs for chemical synapses and Hodgkin-Huxley conductances have been moved to seperate popup windows.
- *Data displays for debugging*
  To allow for easy debugging of wiring and gain problems the software now has two data displays that can show input and/ or output channels or functions thereof (averages, spikes detetected).
- *Save and load parameter settings*
  You can now save and load the current parameter settings of the clamp. This way one does not have to redo parameter settings every time the StdpC program is restarted. The settings are stored in a simple ASCII format that allows editing by hand or snooping around for curiosity.
- *Spike timing dependent plasticity*
  The two chemical synapses can now be made plastic abeying a Spike Timing Dependent Plasticity protocol. There are two different protocols to choose from and a variety of paramters determining the details of the learning mechanism
- *Experimental automatization and scripting*
  The StdpC supports a simple form of experimental protocol automization (scripting). The user can specify a script file that contains events at given times. These events include switching on and off of synaptic connections or Hodgkin=Huxley type conductances as well as arbitrary parameter changes. The script is loaded on starting the clamping process and executes the commands at the given times after clamping started.

6.2 Changes from StdpC version 3 to version 4

- Since this version of StdpC, all synapses are freely reconfigurable as chemical or electrical synapses.
- the presynaptic and postsynaptic neuron of each synapse is freely reconfigurable. In particular it is now possible to combine synapses between two biological neurons (on channels IN0/OUT0 and IN1/OUT1) with synapses from the spike generator (channel SPG) to either of the neurons. This gives more flexibility in the experimental design
- Each chemical synapse can now been chosen individually to be plastic or not.
- The initial value for a plastic synapse is given by its individual G_s setting, not by a global initialization value in the plasticity block. The displays for the synaptic strength have been removed because they turned out to be disruptive for the clamping performance.
- Some smaller adjustments aiming at better user interaction and performance of the data displays. We, however, still recommend not to use the data displays during real experiments. They are for debugging purposes only and degrade clamping performance considerably when used.

6.3 Changes from StdpC v4 to StdpC 2007

- The user interface is now based on QT
- The driver for the DigiData 1200 board was re-created using (an extended version of) the free PortTalk package.
- StdpC 2007 supports all NIDAQmx capable National Instruments boards
- The source code has been re-organized to allow for rapid inclusion of additional hardware support.
- StdpC 2007 allows to use all input and output channels available on the acquisition hardware.
- All synaptic properties including the details of the synaptic plasticity are now individually adjustable for every synapse
- There are now two alternative descriptions for Hodgkin-Huxley type conductances, including several functional forms for the activation and inactivation curves.
- With the additional support for other hardware, there is now more fine-grained control over the channels that are used for voltage and current conversion. All channels are available in “input channels” and “output channels” dialogs. Only channels activated in these dialogs will appear in the drop-down menus within synapses and HH conductances.

7 How to use StdpC

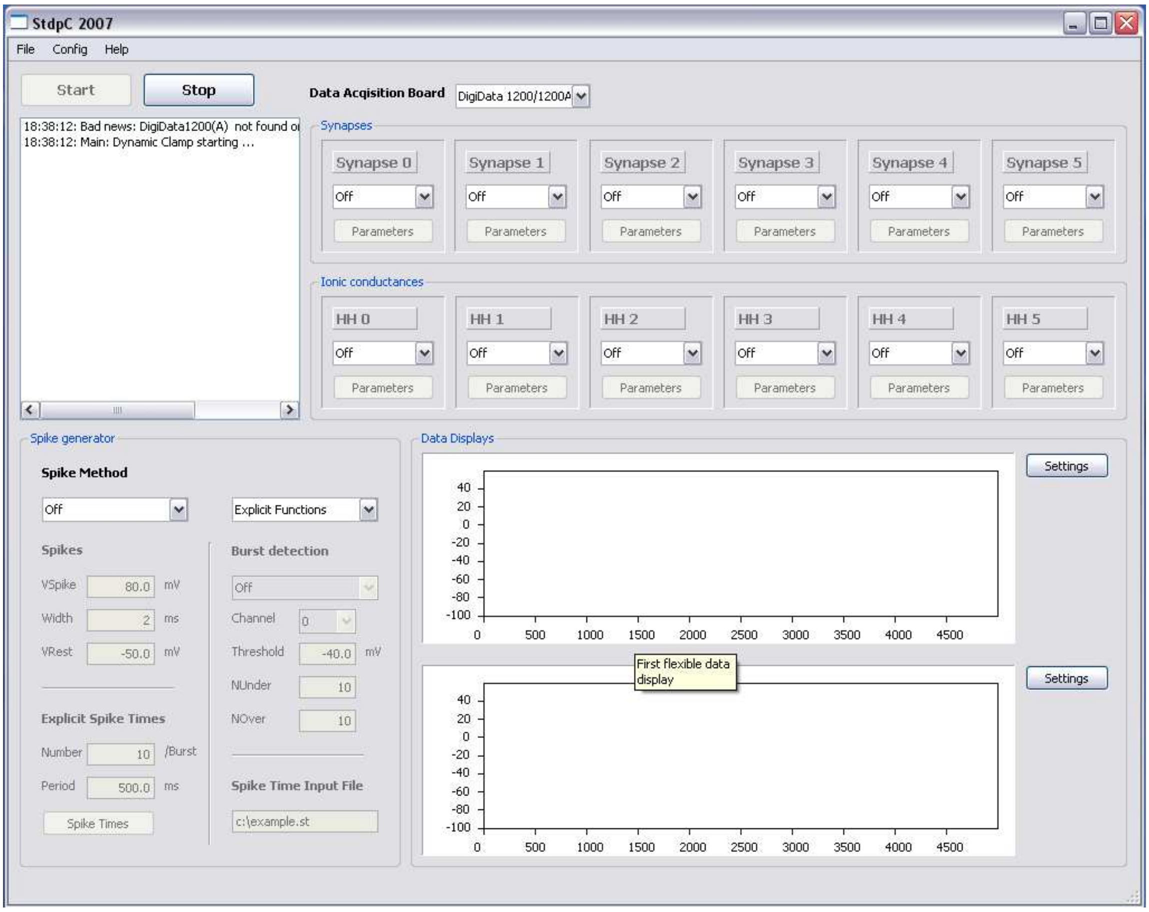
7.1 ”Connecting” the program to the cells

We encourage the insertion of two electrodes into each cell (or a careful impedance compensation of a single electrode) to avoid the generation of a wrong current commands due to artifacts introduced into the membrane potential measurement by the current injection.

The membrane potential output of the microelectrode amplifiers should be connected to the analog inputs channels chosen to be scanned in the “input channels” dialog. The command voltage for the injection of current into the cells will be in the analog output channels chosen in the corresponding drop-down boxes. For specific channels to appear in these boxes, they need to be activated in the “output channels” dialog. The built-in spike generator is internally mapped to be input channel SG.

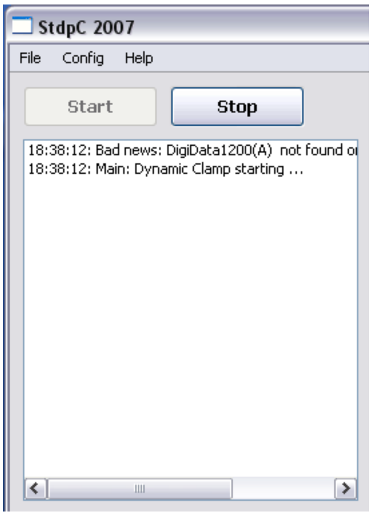
The currents are calculated individually for each conductance and synapse and then are summed for each cell to generate the corresponding voltage command for the microelectrode amplifiers. It is recommended to turn on the dynamic clamp cycle and to monitor the current command voltage in an oscilloscope before turning on the injection of the current in the microelectrode amplifier, to see if it looks like what is expected. If it is orders of magnitude wrong, check the conversion factors in the “input channels” and “output channels” dialogs.

7.2 Control Panel of the Program



The main control panel contains different blocks of control for the different functions of the software – simulating synapses (synapse control block; 1), simulating ionic conductances (Hodgkin-Huxley (HH) type conductances; HH control block; 2), simulating presynaptic activity (spike generator block; 3) and displaying data (data displays; for debugging only; 4). The control blocks are front-ends to more detailed control dialogs. There is a message window in the upper left part of the panel that shows system messages. These messages can be saved for future reference. They are also automatically written to a local file “StdpC_last.log”.

7.3 Configuration of the Hardware and Control of the Program



Once StdpC is started, the last known hardware configuration is loaded (from a local file named "StdpC.conf"). Other parameters are initialized with standard values that are pre-compiled into the software. The message window will show a message whether the chosen hardware was successfully initialized (which was not the case in the example shown on the left).
Failing hardware initialization can have several causes:

1. If using the DigiData 1200(A):
   StdpCs default address for the board is 0x320 (hexadecimal), which is the default address of the board, but the specific board can have been configured to use another range of addresses. Possible values are 0x340, 0x360, 0x380, 0x3A0, 0x3C0, etc. Make sure that the correct address is entered in the "DAQ" dialog.
2. If using a National Instruments board:
   Make sure that NIDAQmx is installed and the board is working properly (using the NI Automation explorer). Make sure your device is in the list of devices that support NIDAQmx (as opposed to the older devices supporting "old style" NIDAQ / NIDAQ legacy only.
3. I you are running with simulated data acquisition:
   Make sure that you have a correctly formatted input file for the membrane potential of cells and that the filename in the "DAQ" dialog points to the right location of this file. Also make sure that the location the output filename points to is writable to you.

The remaining controls are the "start" and the "stop" button. The "start" button initiates the start of dynamic clamp cycles. In particular, pressing this button will stop a running dynamic clamp thread, load the settings from (almost) all dialogs into memory, and start a the dynamic clamp thread again with the new settings. The dynamic clamp thread will then run until stopped by pressing the "stop" button.

The data acquisition hardware can be chosen in the combo box on the top. Note that the software that has been compiled with support for the NIDAQmx libraries will not run if these are not installed *even if a different device is chosen*. The specific hardware settings can be adjusted in the "DAQ" dialog int the "Config" menu of the main menu bar.

The choice of hardware, its settings and the settings for input and output channels are the only exception to the rule that changes only apply after a restart of the dynamic clamp thread (by pressing the "start" button).

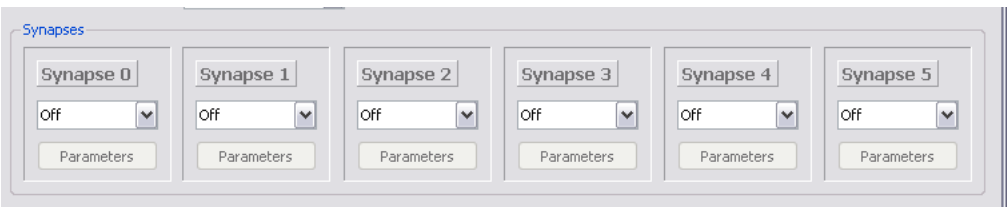
Other general control elements can be found in the menus of the main menu bar.

- "File" menu:
  - "Load Protocol": Load parameter settings from a previous session. The standard filename extension for these is "cpr", albeit the settings are saved in it in plain ASCII format.
  - "Save Protocol": Save the current parameter settings into a file for later use or documentation.
  - "Load Script": Load a script for experiment automation. Scripting is described in section ??.
  - "Unload Script": Remove a script from memory and use StdpC interactively.

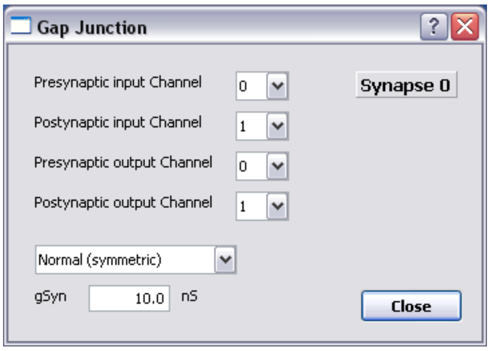
- "Export Log": Export the contents od the message box to a file.
- "Clear Log": Remove the message box contents.
- "Exit": Quit StdpC.

- "Config" menu:
  - "DAQ": Controls for the properties of the data acquisition hardware. This will show a dialog window that is specific to the chosen hardware. For an example for a simulated DAQ, see below.
  - "Input Channels": Brings up a dialog to change settings for the analog input channels, see below for more details.
  - "Output channels": Brings up a corresponding dialog for the output channels, see below.
- "Help": Offers a one-line description about StdpC.

7.4 Electrical synapses



Each subunit in the synapses block can be configured to be a chemical or an electrical synapse (gap junction). Clicking the "Parameters" button will open a control panel that allows to change propertis of the synapse.



If configured as an electrical synapse the control window on the left will appear.
The currents I_{pre} and I_{post} to be injected into the pre- and post-synaptic cells, respectively, are calculated according to:

$$I_{\text{post}}(t) = g_{\text{Syn}}[V_{\text{pre}}(t) - V_{\text{post}}(t)], \tag{1}$$

$$\text{and} \quad I_{\text{pre}}(t) = -I_{\text{post}}(t), \tag{2}$$

where $V_{\text{pre}}(t)$ and $V_{\text{post}}(t)$ are the membrane potential of the two cells. If the gap junction is chosen as rectifying, $I_x(t) = 0$ if $V_{\text{pre}}(t) < V_{\text{post}}(t)$.

The control elements are

- Presynaptic input channel: Combo to choose the input channel that contains the membrane potential of the presynaptic cell (SP= spike generator)
- Postynaptic input channel: Combo to choose the input channel that contains the membrane potential of the postynaptic cell
- Presynaptic output channel: Combo to choose the output channel that will contain the current command for the presynaptic cell.
- Postynaptic output channel: Combo to choose the output channel fot the current command intended for the postsynaptic cell.

- Type combo: Choose whether the gap junction is ordinary (the current can flow from the presynaptic cell to the postsynaptic cell an vice versa; both cells have perfectly symmetrical roles in this case) or rectifying (positive current can only from the presynaptic cell to the possynaptic cell but not in the other direction).
- gSyn: Conductance of the gap junction in nS.

7.5 Chemical Synapses

If the type of synapse is chemical, the control dialog of a chemical synapse is displayed on pressing "Parameters". The current to be injected into the postsynaptic cell, I_{post}, is calculated in each dynamic clamp cycle using a first order kinetics model of the release of neurotransmitter [Destexhe et al., 1994] and an additional inactivation term, $h(t)$, to simulate short term depression:

$$I_{\mathrm{post}} = g_{\mathrm{Syn}}\, S(t)\, h(t) [V_{\mathrm{Syn}} - V_{\mathrm{post}}(t)], \tag{3}$$

where the instantaneous activation, S(t), and inactivation, h(t), terms are given by the differential equations

$$(1 - S_\infty(V_{\mathrm{pre}}))\tau_{\mathrm{Syn}} \frac{dS(t)}{dt} = (S_\infty(V_{\mathrm{pre}}) - S(t)) \tag{4}$$

$$\tau_h \frac{dh(t)}{dt} = h_\infty(V_{\mathrm{pre}}) - h(t), \tag{5}$$

where

$$S_\infty(V_{\mathrm{pre}}) = \begin{cases} \tanh\left[\frac{V_x(t) - V_{\mathrm{Thresh}}}{V_{\mathrm{Slope}}}\right] & \text{if } V_{\mathrm{pre}} > V_{\mathrm{Thresh}} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

$$h_\infty(V_{\mathrm{pre}}) = \frac{A}{1 + \exp\left(\frac{V_{\mathrm{pre}} - V_{\mathrm{Thresh}}}{V_{\mathrm{Slope}}}\right)}, \tag{7}$$

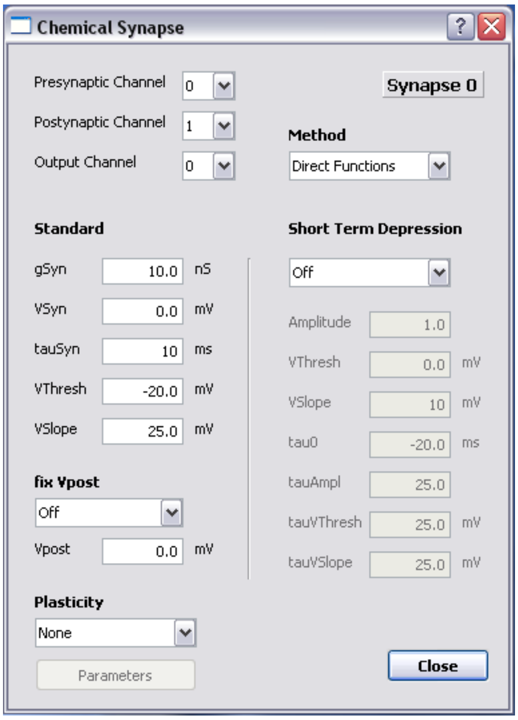
$$\tau_h(V_{\mathrm{pre}}) = \tau_0 - \frac{A_\tau}{1 + \exp\left(\frac{V_{\mathrm{pre}} - V_{\mathrm{Thresh},\tau}}{V_{\mathrm{Slope},\tau}}\right)} \tag{8}$$

The controls in the paramter dialog for the chemical synapse are

- Presynaptic Channel: Combo box to choose the input channel that contains the membrane potential of the presynaptic cell (SP = spike generator)
- Postynaptic Channel: Combo box that contains the membrane potential of the postynaptic cell
- Output Channel: The output channel to which the current command for the synapse will be added

"Standard":

- gSyn: The maximal conductance g_{syn} of the synapse in nS.
- VSyn: The reversal potential V_{Syn} in mV.
- tauSyn: The characteristic time constant τ_{Syn} of the synapse in ms.
- VThresh: The threshold potential V_{Thresh} for the release of neurotransmitter in mV.
- VSlope: The "slope" parameter V_{Slope} of the activation curve in mV.



"Short Term Depression":

- Combo box to switch this mechanism on or off
- Amplitude: The amplitude of the $h(t)$ depression variable. Typically set to 1 (so that h varies between 0 and 1). This was previously used to switch short term depression on or off and remains for historical reasons.
- VThresh: The threshold potential $V_{\mathrm{Thresh},\tau}$ for the activation of h in mV.
- VSlope: The "slope" parameter of the depression activation curve in mV.
- tau0, tauAmpl, tauVThresh, and tauVSlope: Parameters τ_0, A_τ, $V_{\mathrm{Thresh},\tau}$ and $V_{\mathrm{Slope},\tau}$ for the voltage-dependent characteristic time τ_h of short term depression.
- The plasticity combo box allows to choose whether the synapse is subject to long term plasticity and according to which model the plasticity is determined.

For "Spike STDP" a spike-timing based rule is applied according to the parameters defined in the corresponding control panel that appears upon clicking the "Parameters" button. If "ODE STDP" is chosen, the synaptic plasticity is implemented according to the ordinary differential equation (ODE) description in [Abarbanel et al., 2002]. Prameters again are adjusted in the separate panel that appears after pressing "Parameters".

7.6 Spike Timing Dependent Plasticity

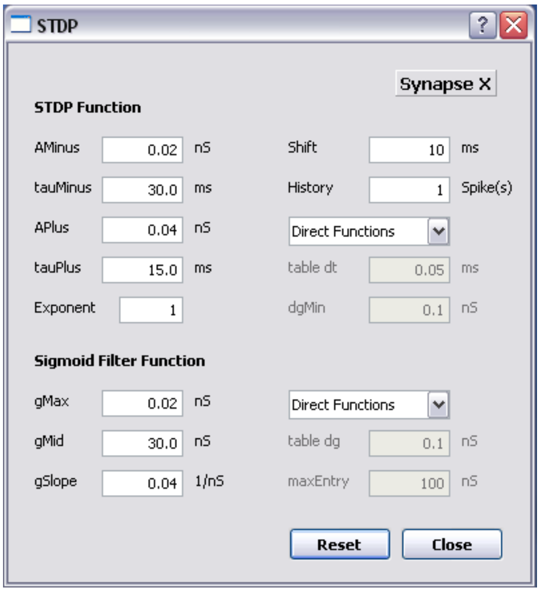
As described above, synapses can be equipped with a form of Spike Timing Dependent Plasticity (STDP). The typical way of implementation, denoted as "Spike STDP", is to detect spikes in the pre- and postsynaptic cells and define

$$\Delta g = \pm A_{\pm} \left(\frac{|\Delta t - \tau_{\mathrm{Shift}}|}{\tau_{\pm}} \right)^q \exp\left(-\frac{|\Delta t - \tau_{\mathrm{Shift}}|}{\tau_{\pm}} \right). \tag{9}$$

Δg is then added to the "raw" synaptic conductance g_{raw} whenever a pre- or postsynaptic spike occurs. Note that for correct function, spike detection needs to be switched on for the pre- and postsynaptic input channels (see input channel dialog). The synaptic conductance g_{Syn} is then determined from g_{raw} through a sigmoid filter

$$g = g_{\max} \tanh\left(\frac{g_{raw} - g_{\mathrm{Mid}}}{g_{\mathrm{Slope}}} \right) \tag{10}$$

to avoid problems of "run-away" potentiation.



The control parameters are:
"STDP Function"

- AMinus: The amplitude A_{-} of the negative (left) part of the STDP curve.
- tauMinus: The time scale τ_{-} (locus of the extremum) of the negative (left) part of the STDP curve.
- APlus: The amplitude A_{+} of the positive (right) part of the STDP curve.
- tauPlus: The time scale τ_{+} (locus of the extremum) of the positive (right) part of the STDP curve.
- Exponent: The exponent q of the polynomial factor in the STDP curve.
- Shift: The offset τ_{Shift} of the STDP curve on the Δt axis.
- History: The number of spikes to be considered in the other neuron if a spike occurs in a given neuron. For example, if set to 1 and a spike occurs in the postsynaptic neuron, the change Δg will only be calculated and applied for the last spike that occurred in the presynaptic neuron. If it was 2 it would be calculated for the last and the next to last spike in the presynaptic neuron.
- The method combo box allows to choose whether the STDP function is calculated directly with the approriate C functions when needed or whether it is tabled up front and this lookup-table is being used.
- table dt: The time step in the lookup table.
- dgMin: The minimum value for Δg that should be in the table. This determines how far to the left and right the table covers the STDP curve.

"Sigmoid Filter Function"

- gMax: The maximal g_{Syn} allowed.
- gMid: The midpoint g_{Mid} of the filter.
- gSlope: The "slope" parameter g_{slope} of the sigmoid filter.
- The method combo allowing the choice between direct calculation or lookup tables for the filter function.
- table dg: The stepping in terms of g_{raw} of the lookup table.
- maxEntry: The maximal g_{raw} entry in the table.

Alternatively the synaptic strength is governed by a set of differential equations according to the model of synaptic plasticity in [Abarbanel et al., 2002]. In this case, the maximal synaptic conductance g_{Syn} is subject to a differential equation system

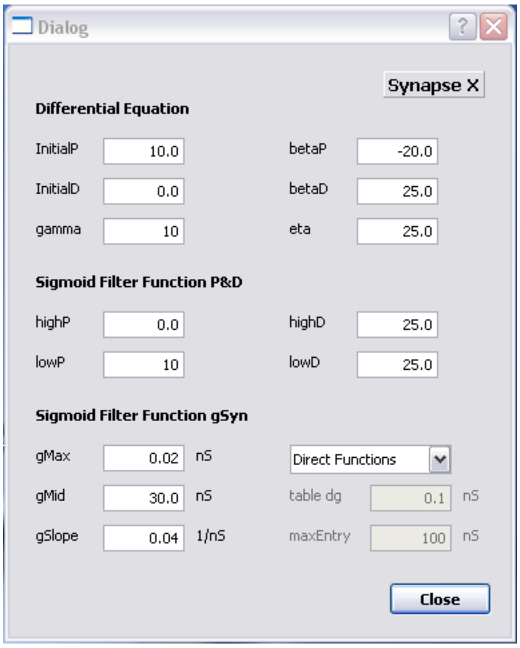
$$\frac{dP}{dt} = v_{\text{pre}} - \beta_P P \tag{11}$$
$$\frac{dD}{dt} = v_{\text{post}} - \beta_D D \tag{12}$$
$$\frac{dg_{\text{raw}}}{dt} = \gamma(PD^{\eta} - DP^{\eta}). \tag{13}$$

The normalized voltages v_x are derived from the measured potentials V_x through capped linear filters

$$v_{\text{pre}} = \begin{cases} 0 & V_{\text{pre}} < V_{P,\text{min}} \\ \frac{V_{\text{pre}} - V_{P,\text{min}}}{V_{P,\text{max}} - V_{P,\text{min}}} & V_{P,\text{min}} \leq V_{\text{pre}} \leq V_{P,\text{max}} \\ 1 & V_{P,\text{max}} < V_{\text{pre}} \end{cases} \tag{14}$$

and accordingly for v_{post},

$$v_{\text{post}} = \begin{cases} 0 & V_{\text{post}} < V_{D,\text{min}} \\ \frac{V_{\text{post}} - V_{D,\text{min}}}{V_{D,\text{max}} - V_{D,\text{min}}} & V_{D,\text{min}} \leq V_{\text{post}} \leq V_{D,\text{max}} \\ 1 & V_{D,\text{max}} < V_{\text{post}} \end{cases} \tag{15}$$



The control parameters for ODE based STDP are: "Differential Equation"

- InitialP: Initial value for the "potentiation variable" P
- InitialD: Initial value for the "depression variable" D
- gamma: Exponent γ
- betaP: Rate of decay β_P of P in 1/ms= kHz.
- betaD: Rate of decay β_D of D in 1/ms= kHz.
- eta: Rate of change of g_{raw} in nS/ms.

"Sigmoid Filter Function P & D"

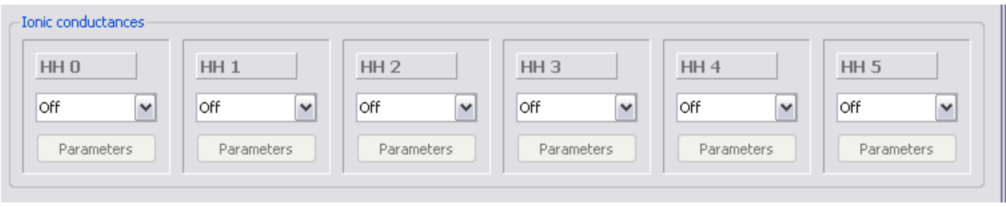
- highP: The upper limit $V_{P,\text{max}}$ of the P filter
- lowP: The lower limit $V_{P,\text{min}}$ of the P filter
- highD: The upper limit $V_{D,\text{max}}$ of the D filter
- lowD: The lower limit $V_{D,\text{min}}$ of the D filter

"Sigmoid Filter Function gSyn"

- gMax: The maximal allowed value for g_{Syn}.
- gMid: The mid point of the sigmoid filter gor g_{Syn}
- gSlope: The "slope" parameter of the sigmoid filter function for g_{Syn}.

- The method combo allows to switch between direct evaluation of the filter or the use of a lookup table
- table dg: The step size in the lookup table
- maxEntry: The maximum of g_{raw} for which table entries are generated.

7.7 Hodgkin-Huxley Type Conductances



StdpC can inject up to six Hodgkin-Huxley type ionic conductances into cells. The six conductances can be inserted into a single cell or into several cells in any desired combination. The control blocks of Hodgkin- Huxley type conductances allow to choose two different types for the description.

For the "m/h/tau" description, the current I_{HH} to be injected into a cell with membrane potential V(t) is calculated according to

$$I_{HH}(t) = g_{\text{Max}} m(t)^p h(t)^q (V_{\text{rev}} - V(t)), \tag{16}$$

where m and h are modeled as

$$\tau_m \frac{dm}{dt} = m_\infty(V) - m, \tag{17}$$

$$\tau_h \frac{dh}{dt} = h_\infty(V) - h, \tag{18}$$

with steady state values

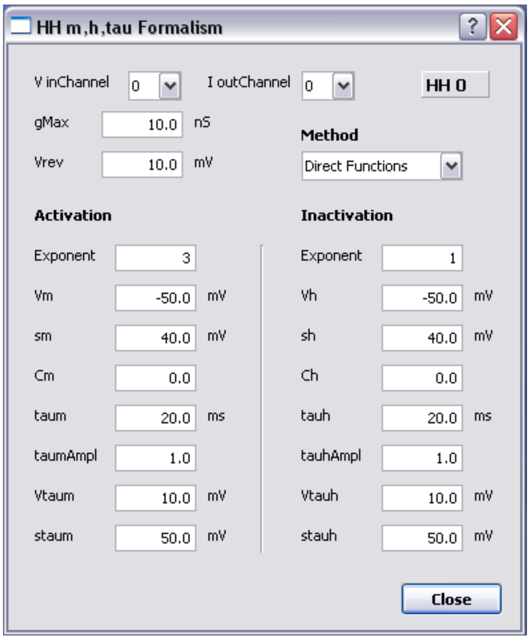
$$m_\infty = \frac{1}{1 + \exp\left(\frac{V - V_m}{s_m}\right)}, \tag{19}$$

$$h_\infty = \frac{1}{1 + \exp\left(\frac{V - V_h}{s_h}\right)}, \tag{20}$$

and time scales

$$\tau_m = \tau_{0,m} - \frac{A_{\tau,m}}{1 + \exp\left(\frac{V - V_{\tau,m}}{s_{\tau,m}}\right)}, \tag{21}$$

$$\tau_h = \tau_{0,h} - \frac{A_{\tau,h}}{1 + \exp\left(\frac{V - V_{\tau,h}}{s_{\tau,h}}\right)}, \tag{22}$$



Control parameters of the ionic conductances in the m/h/tau formalism are:

- V in Channel: The input channel that contains the membrane potential of the cell the conductance is injected into
- I out Channel: The output channel to which the current command for the inserted ionic conductance will be added
- gMax: The maximal conductance g_{Max} of the inserted channel
- Vrev: The reversal potential V_{rev} of the inserted channel
- Method: The combo box allows to choose between direct evaluation of C functions or a pre-calculated lookup table.

"Activation"

- Exponent: The exponent p in the current equation
- Vm: The activation potential V_m
- sm: The width s_m of the activation function
- Cm: The offset parameter C_m for persistent currents
- taum: The minimal time scale $\tau_{0,m}$ for τ_m.
- taumAmpl: The range (Amplitude) $A_{\tau\, m}$ for the time scale τ_m.
- Vtaum: The mid point $V_{\tau,m}$ for the time scale sigmoid function.
- staum: The "slope" parameter $s_{\tau,m}$ for the time scale sigmoid function

"Inactivation"

- Exponent: The exponent q in the current equation
- Vh: The inactivation potential V_h
- sh: The width s_h of the inactivation function
- Ch: The offset parameter C_h for persistently inactivated currents
- tauh: The minimal time scale $\tau_{0,h}$ for τ_h.
- tauhAmpl: The range (Amplitude) $A_{\tau,h}$ for the time scale τ_h.
- Vtauh: The mid point $V_{\tau,h}$ for the time scale sigmoid function.
- stauh: The "slope" parameter $s_{\tau,h}$ for the time scale sigmoid function.

Alternatively, the ionic conductances can be described in a α/β formalism, where

$$I_{\text{Syn}} = g_{\text{Max}} m^p h^q (V - V_{\text{rev}}) \tag{23}$$

$$\frac{dm}{dt} = \alpha_m(1-m) - \beta m \tag{24}$$

$$\alpha_m = k_{\alpha,m} F_{\alpha,m}\left(\frac{V - V_{\alpha,n}}{s_{\alpha,m}}\right) \tag{25}$$

$$\beta_m = k_{\beta,m} F_{\beta,m}\left(\frac{V - V_{\beta,n}}{s_{\beta,m}}\right) \tag{26}$$

$$\frac{dh}{dt} = \alpha_h(1-h) - \beta_h h \tag{27}$$

$$\alpha_h = k_{\alpha,h} F_{\alpha,h}\left(\frac{V - V_{\alpha,n}}{s_{\alpha,h}}\right) \tag{28}$$

$$\beta_h = k_{\beta,h} F_{\beta,h}\left(\frac{V - V_{\beta,n}}{s_{\beta,h}}\right). \tag{29}$$

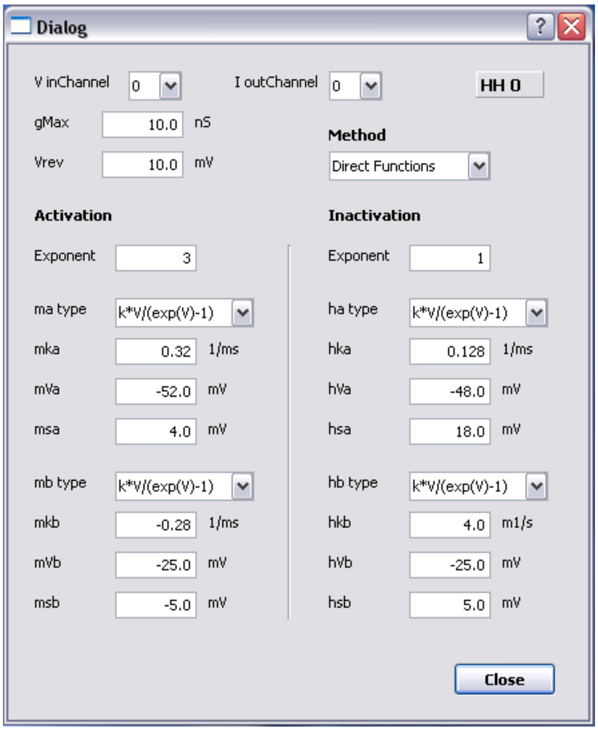
The "activation"/"inactivation" functions $F_{\bullet,\bullet}$ can be chosen from three choices,

$$F_1(x) = \frac{x}{\exp(x) - 1} \tag{30}$$

$$F_2(x) = \exp(x) \tag{31}$$

$$F_3(x) = \frac{1}{1 + \exp(x)}. \tag{32}$$

Note, that this formalism allows to implement, for example, the classic neuron model by Traub and Miles [Traub and Miles, 1991].



The control parameters are

- V in Channel: The input channel that contains the membrane potential of the cell the conductance is injected into
- I out Channel: The output channel to which the current command for the inserted ionic conductance will be added
- gMax: The maximal conductance g_{Max} of the inserted channel
- Vrev: The reversal potential V_{rev} of the inserted channel
- Method: The combo box allows to choose between direct evaluation of C functions or a pre-calculated lookup table.

"Activation"

- Exponent: The exponent p in the current equation
- ma type: The choice of the functional form for $F_{\alpha,m}$.
- mka: Rate parameter $k_{\alpha,m}$ for the rise of m
- mVa: The midpoint $V_{\alpha,m}$ for the sigmoid function for α_m
- msa: The "slope" parameter $s_{\alpha,m}$ for the sigmoid function for α_m
- mb type: The choice of the functional form for $F_{\beta,m}$.
- mkb: Rate parameter $k_{\beta,m}$ for the decay of m
- mVb: The midpoint $V_{\beta,m}$ for the sigmoid function for β_m
- msb: The "slope" parameter $s_{\beta,m}$ for the sigmoid function for β_m

"Inactivation"

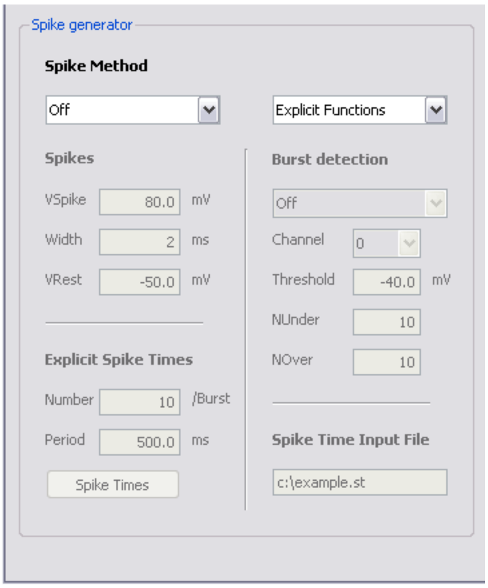
- Exponent: The exponent q in the current equation
- ha type: The choice of the functional form for $F_{\alpha,h}$.
- hka: Rate parameter $k_{\alpha,h}$ for the rise of h
- hVa: The midpoint $V_{\alpha,h}$ for the sigmoid function for α_h
- hsa: The "slope" parameter $s_{\alpha,h}$ for the sigmoid function for α_h
- hb type: The choice of the functional form for $F_{\beta,h}$.
- hkb: Rate parameter $k_{\beta,h}$ for the decay of h
- hVb: The midpoint $V_{\beta,h}$ for the sigmoid function for β_h
- hsb: The "slope" parameter $s_{\beta,h}$ for the sigmoid function for β_h

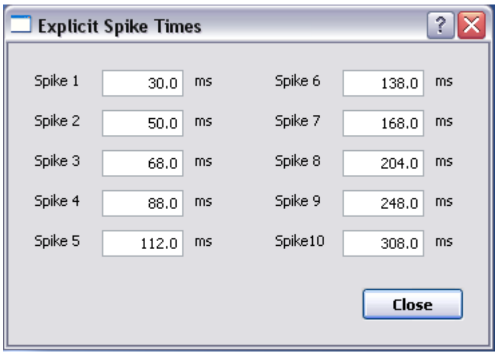
7.8 Spike generator

The spike generator unit can replace a presynaptic cell. It can operate in several very different modes. When its input are spike timings, the spikes have a pre-defined form. Alternatively it can replay a voltage waveform from a file directly. The spike time information can either be explicit spike times from a file, spike patterns (bursts) specified in a file, or a spike pattern defined explicitly through the graphical interface. The pattern options can be combined with burst detection, in which each spike pattern is triggered by a detected thershold event in another (measured) neuron.

The control elements for the spike generator are

- The "Spike Method" combo allows to choose the type of operating method – explicit spike times from the GUI, spike times from a file (that includes explicit times if burst detection is not enabled, or burst patterns if it is), and replay of a voltage waveform from a file.
- The method combo allows to choose for the use of functions or lookup tables.





"Spikes"

- VSpike: The amplitude of the spikes generated
- Width: The width of spikes in ms
- VRest: Th resting potential from which the spikes depart

"Explicit Spike Times"
If this method was chosen, the parameters are

- Number: The number of spikes in the pattern (burst)
- Period: The period with which to repeat the spike pattern periodically
- The "Spike Times" button will make the window visible, in which explicit spike times can be entered

"Burst detection"

- The transition combo allows to choose whether to trigger an event for low to high or high to low transitions through a threshold value
- Channel: The input channel on which to detect events
- Threshold: The trigger threshold for event detection. For low→high detection, an event is triggered whenever NUnder measurements were below threshold and afterwards NOver measurements were above. Note that for noisy signals it may be necessary to increase both numbers for reliable detection. The events below and above are not required to be contiguous in the current implementation.
- NUnder: see "Threshold" above.
- NOver: see "Threshold" above.

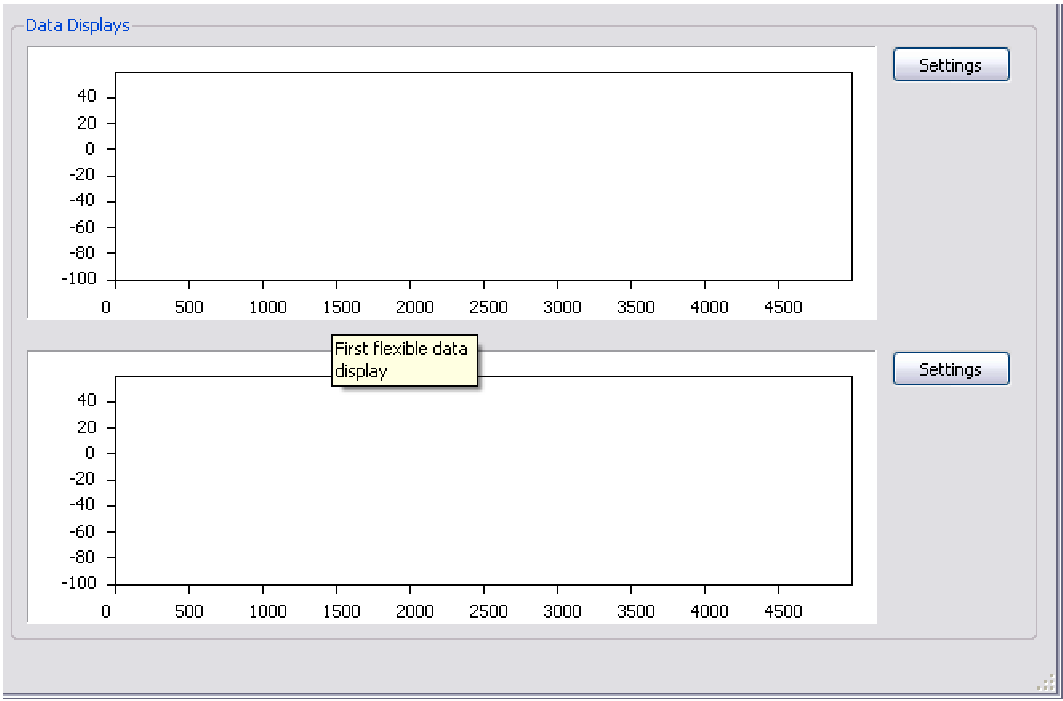
"Spike Time Input File"

- The input filename for the file that contains the spike time information. If burst detection is off, the file should contain one clear text column containing times in seconds when spikes shall occur. If burst detection is on, StdpC 2007 expects descriptions of spike patterns containing of
  - An integer denoting the number of spikes in the pattern
  - A matching number of spike times in seconds, measured from the detection event.

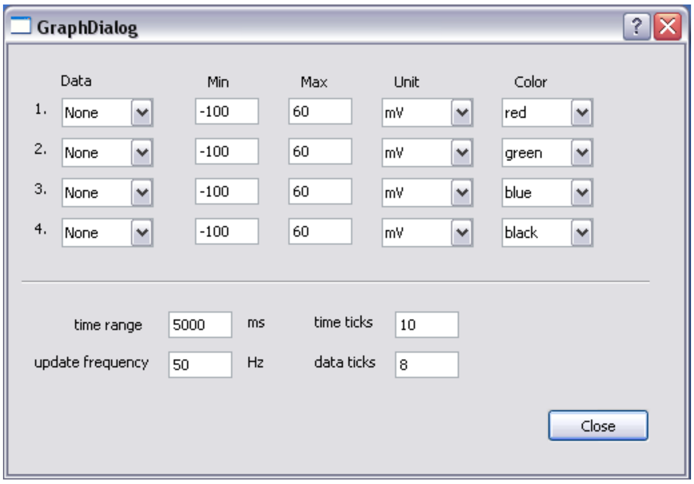
  These pattern descriptions are best separated by newlines.

7.9 Data displays

The two data displays in StdpC are very useful to check the parameter settings chosen. Typical errors in dynamic clamp setups are wrong conversion factors on input- and output channels. By displaying the data acquired on input channels in the graph displays, some of these errors can easily be detected and corrected.



The graph panels are controlled by the dialog that appears when pressing "Settings".



The control elements are:

- Data column: Combo boxes to choose which data to display
- Min: Minimum on the y axis for this data
- Max: Maximum ion the y axis for this data
- Unit: Units in which Min and Max are expressed
- Color: Choice of color of the data trace.

- time range: The length of the x axis in ms
- time ticks: Tumber of ticks on the x axis.
- update frequency: The rate with which points on the display are updated. Note that a too high rate will likely lead to an overload and subsequent freeze of the program. A more efficient and

safe implementation of the displays is on the todo list.

8 Experiment automation

So far we have described how to cotrol StdpC 2007 using the graphical user interface. Instead, or in addition, most parameters can be controlled through a scripting mechanism. Through the File$rightarrow$Load Script one can specify a script file. The script file should contain three columns:

1. A time in seconds when a certain event shall occur
2. A parameter name in clear text which shall be changed
3. A new value for the parameter in question

The value column can contain double precision numbers or integers depending on the parameter being changed.

The script file will be read into a list of events in memory such that changes to the file have no effect after the script was loaded.

The parameter names are the same as used when the state of the gui is saved ("save protocol"). However, not all parameters can reasonably changed while the clamp is running. A list of the most common parameters on may want to change through scripting are given in the table.

Please note that internally, all parameters and variables are in SI units V, A, s, S, etc. The same applies to values in the script files. The variable i in the variable names in the table has tobe replaced by the appropriate number of the synapse or conductance, $i \in \{0, \dots, 5\}$.

9 User feedback

Please do give feedback if you found the software useful, buggy, or have general comments or questions. Please send correspondence through the sourceforge email or directly to T.Nowotny@sussex.ac.uk. If you are interested in contributing to the future development please do not hesitate to contact me as well.

10 Acknowledgments

The StdpC software would not exist without the early dynamic clamp versions developed by Reynaldo D. Pinto and I (TN) am grateful to him for supplying the source code for further development back then. I am also grateful to Attila Szücs for his excellent suggestions for improving and extending the software and his tireless testing of new and buggy versions.

References

Abarbanel, H. D. I., Huerta, R., and Rabinovich, M. I. (2002). Dynamical model of longt-term synaptic plasticity. *P Natl Acad Sci USA*, 99:10132–10136.

Destexhe, A., Mainen, Z. F., and Sejnowski, T. J. (1994). An efficient method for computing synaptic conductances based on a kinetic model of receptor binding. *Neural Comput*, 6:14–18.

Hodgkin, A. L., Huxley, A. F., and Katz, B. (1949). Ionic current underlying activity in the giant axon of the squid. *Arch Sci Physiol*, 3:129–150.

Name	value range
CSynp[i].ST.AMinus	double
CSynp[i].ST.tauMinus	double
CSynp[i].ST.APlus	double
CSynp[i].ST.tauPlus	double
CSynp[i].ST.Exponent	int
CSynp[i].ST.Shift	double
CSynp[i].ST.History	int
CSynp[i].ST.tableDt	double
CSynp[i].ST.tableDgMin	double
CSynp[i].ST.gMax	double
CSynp[i].ST.gMid	double
CSynp[i].ST.gSlope	double
CSynp[i].ODE.InitialP	double
CSynp[i].ODE.InitialD	double
CSynp[i].ODE.betaP	double
CSynp[i].ODE.betaD	double
CSynp[i].ODE.gamma	double
CSynp[i].ODE.eta	int
CSynp[i].ODE.highP	double
CSynp[i].ODE.lowP	double
CSynp[i].ODE.highD	double
CSynp[i].ODE.lowD	double
CSynp[i].ODE.gMax	double
CSynp[i].ODE.gMid	double
CSynp[i].ODE.gSlope	double
CSynp[i].gSyn	double
CSynp[i].VSyn	double
CSynp[i].tauSyn	double
CSynp[i].VThresh	double
CSynp[i].VSlope	double
CSynp[i].STD	0-1
CSynp[i].STDAmpl	double
CSynp[i].STDVThresh	double
CSynp[i].STDVSlope	double
CSynp[i].STDtauAmpl	double
CSynp[i].STDtau0	double
CSynp[i].STDtauVThresh	double
CSynp[i].STDtauVSlope	double
CSynp[i].fixVpost	0-1
CSynp[i].Vpost	double
CSynp[i].Plasticity	0-2
ESynp[i].type	0-1
ESynp[i].gSyn	double
mhHHp[i].gMax	double
mhHHp[i].Vrev	double
mhHHp[i].mExpo	int
mhHHp[i].hExpo	int
mhHHp[i].Vm	double
mhHHp[i].sm	double

Name	value range
mhHHp[i].Cm	double
mhHHp[i].taum	double
mhHHp[i].taumAmpl	double
mhHHp[i].Vtaum	double
mhHHp[i].staum	double
mhHHp[i].Vh	double
mhHHp[i].sh	double
mhHHp[i].Ch	double
mhHHp[i].tauh	double
mhHHp[i].tauhAmpl	double
mhHHp[i].Vtauh	double
mhHHp[i].stauh	double
abHHp[i].gMax	double
abHHp[i].Vrev	double
abHHp[i].mExpo	int
abHHp[i].hExpo	int
abHHp[i].maFunc	0-2
abHHp[i].mka	double
abHHp[i].mVa	double
abHHp[i].msa	double
abHHp[i].mbFunc	double
abHHp[i].mkb	double
abHHp[i].mVb	double
abHHp[i].msb	double
abHHp[i].haFunc	0-2
abHHp[i].hka	double
abHHp[i].hVa	double
abHHp[i].hsa	double
abHHp[i].hbFunc	0-2
abHHp[i].hkb	double
abHHp[i].hVb	double
abHHp[i].hsb	double
SGp.VSpike	double
SGp.spkTimeScaling	double
SGp.VRest	double
SGp.bdThresh	double
SGp.bdNUnder	double
SGp.bdNOver	double
SGp.period	double
SGp.SpikeNo	int
SGp.SpikeT[0]	double
SGp.SpikeT[1]	double
SGp.SpikeT[2]	double
SGp.SpikeT[3]	double
SGp.SpikeT[4]	double
SGp.SpikeT[5]	double
SGp.SpikeT[6]	double
SGp.SpikeT[7]	double
SGp.SpikeT[8]	double
SGp.SpikeT[9]	double

Pinto, R. D., Elson, R. C., Szücs, A., Rabinovich, M. I., Selverston, A. I., and Abarbanel, H. D. I. (2001). Extended dynamic clamp: Controlling up to four neurons using a single desktop computer and interface. *J Neurosci Meth*, 108:39–48.

Robinson, H. P. and Kawai, N. (1993). Injection of digitally synthesized synaptic conductance transients to measure the integrative properties of neurons. *J Neurosci Meth*, 49:157–165.

Sharp, A. A., O'Neil, M. B., Abbott, L. F., and Marder, E. (1993). Dynamic clamp: computer-generated conductances in real neurons. *J Neurophysiol*, 69:992–995.

Traub, R. D. and Miles, R. (1991). *Neural Networks of the Hippocampus*. Cambridge University Press, New York.