Slice 1.1.1 Technical Manual

by Fanyee Anja Lee & Chris DiMattina

1 Purpose

This technical manual is provided to allow experienced IGOR programmers to modify the code for Slice to suit their needs. We advise you make a backup copy of Slice before you attempt to modify anything.

2 Slice Global Variable List

2.1 Portal Variables

Variable Name	Associated Control	Default	Range	Comments
Initialstring (string)	Initialsetvar	""		User initials. Required Field -- user cannot proceed if it is default
Animal_num	Animalsetvar	0	Low: 0 High:∞ Inc: 1	Animal #. Required Field
Cell_num	Cellsetvar	0	Low: 0 High:∞ Inc: 1	Animal #. Required Field -- user cannot proceed if it is default
Animal_age	Ageset	0	Low: 0 High:∞ Inc: 1	Days postnatal
Dissect_time	Dissectset	0	Low: 0 High:∞ Inc: 1	Dissection time (minutes)
Temp	Tempsetvar	0	Low: 0 High: 100 Inc: 1	Degrees Celcius
Elect_resist	Resistsetvar	0	Low: 0 High:∞ Inc: 1	Mega-ohms
Fifo_size	FIFOPop	1	{1,2}	1 - 1024 kilosample FIFO 2 - 256 kilosample FIFO
Anesthesia (string)	Anesthesiaset	"none"		Name of anesthetic used on animal
Site_string	Sitepop	LSO		Area recorded from
Prep_string	Preppop	Gerbil		Animal used
Treat_String	Treatpop	Control		Treatment
Dye_string	Dyepop	Biocytin		Dye
Solution_String	Solpop	K Gluc		Type of solution
Other_site_string	Othersiteset	""		Alternate recording site
Other_prep_string	Otherprepset	""		Alternate preparation
Other_treat_string	Othertreatset	""		Alternate treatment
Other_dye_string	Otherdyeset	""		Alternate dye

Other_solution_string	Othersolset	""		Alternate solution

2.2 Stimulation Variables

Variable Name	Associated Control	Default	Range	Comments
DAC01Gain	DAC01GainSet	10	Lo: 1 Hi:10 In: 9	Gain for DAC0 and DAC1 (stimulating electrodes)
Is_vc	StimVCPopup	1	0/1	1 - voltage clamp 0 - current clamp
Sensitivity	SensitivePopUp	1 (0.1)	1/2/3	DAC2 equipment sensitivity. 0.1/0.01/0.001
DAC0_state	DAC0TrainPop	1 (Train)	1/2/3	Train/Pulse/Off
DAC1_state	DAC1TrainPop	1 (Train)	1/2/3	Train/Pulse/Off
DAC2_state	DAC2PopUp	1 (Pulse)	1/2/3	Pulse/2 Pulse/Ramp/Off Ramp is not available in current clamp
Is_tonic	TonicAmp	0	0/1	Becomes 1 when tonic_amp is nonzero
Tonic_amp	TonicAmp	0	Lo: -150 mV -2000 pA Hi: 150 mV 2000 pA In: 1 mV 10 pA	Amplitude of tonically injected current or voltage
Reps	StimRepsSet	1	Lo: 1 Hi: 500 In: 1	# of repetitions of each member of the family specified
DAC0_amp_first	DAC0AmpFirstSet	5	Lo: -100 Hi: 100 In 0.1	DAC0 Start voltage
DAC0_amp_last	DAC0AmpLastSet	5	Lo: -100 Hi: 100 In: 0.1	DAC0 Stop voltage
DAC0_amp_step	DAC0AmpStepSet	0	Lo: -20 Hi: 20 In:0.1	DAC0 Step voltage
DAC0_lat_first	DAC0LatFirstSet	0	Lo:0 Hi:5000 In:0.05	DAC0 Start latency (msec)
DAC0_lat_last	DAC0LatLastSet	0	Lo:0 Hi:5000 In:0.05	DAC0 Stop latency (msec)
DAC0_lat_step	DAC0LatStepSet	0	Lo:0 Hi:5000 In:0.05	DAC0 Latency step (msec)
DAC0_int_first	DAC0IntFirstSet	1	Lo:0.1 Hi:5000	DAC0 Interval first (msec)

			In:0.1	
DAC0_int_last	DAC0IntLastSet	1	Lo:0.1 Hi:5000 In:0.1	DAC0 Interval last (msec)
DAC0_int_step	DAC0IntStepSet	0	Lo:0.1 Hi:5000 In:0.1	DAC0 Interval step (msec)
DAC0_dur	DAC0DurSet	0.1	Lo:0.1 Hi:5000 In:0.1	DAC0 Pulse duration (msec)
DAC0_reps	DAC0RepSetVar	10	Lo: 1 Hi:500 In:1	DAC0 Number of pulses in train
DAC1_amp_first	DAC1AmpFirstSet	5	Lo: -100 Hi: 100 In: 0.1	DAC1 Start voltage
DAC1_amp_last	DAC1AmpLastSet	5	Lo: -100 Hi: 100 In: 0.1	DAC1 Stop voltage
DAC1_amp_step	DAC1AmpStepSet	0	Lo: -20 Hi: 20 In: 0.1	DAC1 Step voltage
DAC1_lat_first	DAC1LatFirstSet	0	Lo:0 Hi:100 In:0.05	DAC1 Start latency
DAC1_lat_last	DAC1LatLastSet	0	Lo:0 Hi:100 In:0.05	DAC1 Stop latency
DAC1_lat_step	DAC1LatStepSet	0	Lo:0 Hi:100 In:0.05	DAC1 Latency Step
DAC1_int_first	DAC1IntFirstSet	1	Lo: 0.1 Hi: 5000 In: 0.1	DAC1 Interval first (msec)
DAC1_int_last	DAC1IntLastSet	1	Lo: 0.1 Hi: 5000 In: 0.1	DAC1 Interval last (msec)
DAC1_int_step	DAC1IntStepSet	0	Lo:1 Hi:5000 In:1	DAC1 Interval step (msec)
DAC1_dur	DAC1DurSet	0.1	Lo:0.1 Hi:5000 In:0.1	DAC1 Pulse dur
DAC1_reps	DAC1RepSetVar	10	Lo:1 Hi:500 In:1	DAC1 number of pulses in train
DAC2_amp_first	DAC2AmpFirstSet	5	Lo: -150 mV -2000 pA Hi: 150 2000 pA	DAC2 start amplitude

			In: 1 mV 10 pA	
DAC2_amp_last	DAC2AmpLastSet	5	Lo: -150 mV -2000 pA Hi: 150 mV 2000 pA In: 1 mV 10 pA	DAC2 stop amplitude
DAC2_amp_step	DAC2AmpStepSet	0	Lo:-50 mV -2000 pA Hi:50 mV 2000 pA In:1 mV 10 pA	DAC2 step size
DAC2_dur_first	DAC2DurFirstSet	100	Lo:0 Hi:5000 In:1	DAC2 start duration
DAC2_dur_last	DAC2LastDurSet	100	Lo:0 Hi:5000 In:1	DAC2 stop duration
DAC2_dur_step	DAC2DurStepSet	0	Lo:-5000 Hi:5000 In:1	DAC2 step duration
DAC2_ramp_dur	DAC2RampDur	100	Lo:0 Hi:5000 In:1	DAC2 ramp duration
DAC2_ramp_start	DAC2RampStart	-50	Lo:-150 Hi:150 In:1	DAC2 ramp start voltage
DAC2_ramp_stop	DAC2RampStop	50	Lo:-150 Hi:150 In:1	DAC2 ramp stop voltage
DAC0_name_string	DAC0NameSet	"left ear"		Name of DAC0 channel
DAC1_name_string	DAC1NameSet	"right ear"		Name of DAC1 channel
DAC2_name_string	DAC2NameSet	"intra-cellular"		Name of intracellular channel

2.3 Acquisition Variables

Variable Name	Associated Control	Default	Range	Comments
TrialPeriod	AcqTrialPeriodSet	1000	Lo: 50 Hi: 60000 In: 10	This is the amount of time between the beginning of trials
Prestim	AcqPreStimSet	20	Lo: 0 Hi: 50 In: 5	Time to take data before stimulating
SratekHz	AcqSrateSet	100	Lo:0.1 Hi:100 In:0.1	Sampling rate in kHz
Cc_gain	CCGainSet	10	Lo:1 Hi:10 In: 9	Gain of input signal on DAC2
Volts_div	AcqYScaleSet	1	Lo: 0.1 Hi: 2000 In: 0.1	Oscilloscope display
Acq_compress	AcqCompSet	10	Lo: 10 Hi: 100 In: 10	Factor by which acquired data is undersampled before saving
Acquire	AcqAcqSet	100	Lo:10 Hi:60000 In:10	Time to acquire data
Ext_trig	AcqExtTrigSet	0	{1/0}	External trigger mode toggle
N_ext_trials	AcqExtTrigTrialSet	1	Lo: 1 Hi: 1000 In: 1	Number of external trigger trials
Live_mode	AcqLiveCheck	0	{1/0}	Live mode toggle
Blind_mode	BlindCheck	0	{1/0}	Blind mode toggle
Family_name	AcqFamName	No default		Name of data file
Avg_mode	AvgCheck	0	{1/0}	Average mode toggle

2.4 Preview Variables

Variable Name	Associated Control	Default	Comments
Num_trials	PreviewNumTrialsDisplay	**Dependency--** Num_trials := reps*num_out_waves	Number of trials in stimulus regimen
Preview_trial	PreviewCurTrialSet	No Default	Current trial

2.5 Other Global Variables

Variable Name	Default	Comments
Srate	1/(SratekHz*1000)	Sampling rate in microseconds
Got_data	0	1 - got data 0 - don't have data
Baseline	0	Mean of baseline wave/ volts_div
Got_base	0	1 - got baseline 0 - don't have baseline
Old_trial	1	Previously displayed trial
Out_waves_made	0	1 - waves made 0 - not made
Family_num	1	Number of family
N_traces_col	0	Number of traces actually collected
Acquire_mem	Acquire	"memory" variable for Acquire time
Prestim_mem	Prestim	"memory" variable for Prestim time
SratekHz_mem	SratekHz	"memory" variable for SratekHz
Reps_mem	Reps	"memory" variable for Reps
Dac0_mem	Dac0_state	"memory" variable for DAC0
Dac1_mem	Dac1_state	"memory" variable for DAC1
Dac2_mem	Dac2_state	"memory" variable for DAC2
Pset_num	1	Number of stimulus parameter set

2.6 Modifying Variable Ranges

To modify the ranges of variables, select the window which has the control you wish to modify, and select the control whose associated variable you wish to modify. Double click and you will get a menu which will allow you to change the range. After changing the range, hit the "change" button. Then close the window and select "Replace". Then close all windows in the experiment and choose "Save Experiment" in the file menu. Then quit IGOR and re-start.

3 Slice Functions, Procedures and Waves

3.1 Function List

Function	Inputs	Outputs
Calcsize(msec,numwave)	msec - duration of wave numwave - number of channels being stimulated	This gives the number of points in a wave of msec duration , given that numwave channels are stimulated
AfterFileOpenHook()		Opens up Portal window when the experiment starts
AppendToLog(nb,str,stampDateTime)	nb - name of notebook str - string to add stampDateTime: 1 - stamp 0 - default	Used to append the portal information to the experimental notebook
CheckStimParms(f,l,s)	f - first l - last s - step	Makes sure first, last and step are consistent (1 - ok /0 - not ok)
GetDAC01Dur(r,i,d,lat,dac)	r - repetitions i - interval d - pulse duration lat - stimulus latency dac - channel (1 or 0)	Makes sure that the stimulus duration is not longer than the acquisition time. (1 - ok / 0 - not ok)
GetDAC2Dur(p_dur,r_dur)	p_dur - pulse duration r_dur - ramp duration	This gives the duration of the signal which is coming out of DAC2
MemTest(r,o,mode)	R - number of trials O - number of outwaves Mode - 0 = default 1= external trigger	1 - OK 0 - insufficient memory or FIFO space
CheckDAC2Parms(amp)	amp - amplitude of DAC2	Make sure that we are not asking the ITC18 to produce a voltage out of its operating range
Padnum(n)	n – number of spaces to pad	Not used in program but may be utilized to pad strings of numbers with “0”

3.2 Procedure List

3.2.1 Acquisition

Procedure	Associated Control	Action
InitAcqVars		Initializes variables in the acquisition window
Acquisition [Window]		Creates the acquisition window
SaveButton	AcqSaveButton	Saves the compressed input waves to the current data file and then refreshes acquisition window and deletes the following input waves: **Multwave[n]** **Inwave[n]** **Inwave** **Base_dummy** **In_temp**
ClearButton	AcqClearButton	Refreshes acquisition window and kills the following waves: **Multwave[n]** **Inwave[n]** **Inwave** **Base_dummy**
PeriodProc	AcqTrialPeriodSet	Makes sure that the trial period is not less than prestim+ acquire
Wait(t)		Waits for t milliseconds
BaseButton	AcqBaseButton	Takes a trace of baseline activity
QuitButton	AcqQuitButton	Kills all waves and windows. Quits IGOR
VoltsDivProc	AcqYScaleSet	Updates the acquisition window when volts/div is changed
StimGoButton	AcquireGoButton	Outputs the stimulus regiment and takes input data
MakeInWaves		Creates the input waves
MakeMultiWaves		Multiplexes the output waves so they can be output to the DAC
PreStimSetVarProc	AcqPreStimSet	This sets the prestim time
AcquireSetVarProc	AcqAcqSet	This sets the acquire time
PeriodProc	AcqTrialPeriodSet	This sets the trial period
AcqNewButtonProc	AcqNewButton	This exits program and enters portal
AcqLiveCheckProc	AcqLiveCheck	This toggle live mode
BlindCheckProc	BlindCheck	This toggles blind mode
AverageOnCheckProc	AvgCheck	This toggles average mode
CompSetVarProc	AcqCompSet	This procedure gives the desired range of compression {1,10,20,..}
ExtTrigCheckProc	ExtTrigCheck	Turns on external trigger mode
SrateSetProc	AcqSrateButton	Beeps if the user changes the

		sampling rate if the stimulus regimen has already been made

3.2.2 Portal

Procedure	Associated Control	Action
Portal [Panel]		Creates the protal window
AboutButtonProc	AboutButton	Opens up the About Box
InitPortalPopups		Sets all of the portal popup menus to their default state
InitPortalVars(flag)		Initializes the portal variables to their default values. If flag = 1 it also initializes the portal pop-up menus
ResetButtonProc	ResetButton	Calls InitPortalVars(1)
PortalQuitButtonProc	QuitButton	Quits Slice
SitePopMenuProc	Sitepop	Sets the site string
PrepPopMenuProc	Preppop	Sets the prep string
TreatPopMenuProc	Treatpop	Sets the treat string
DyePopMenuProc	Dyepop	Sets the dye string
SolPopMenuProc	SolPop	Sets the solution string
FIFOPopMenuProc	FIFOPop	Sets the FIFO size
AboutBox [Panel]		Creates the "About Slice" box
AboutOkButtonProc	AboutOkButton	Kills the "About Slice" box
SetNotebookString		This creates the name of the notebook string using information from the portal
PortalStartButtonProc	StartButton	Enters the portal and starts the program
DataButtonProc	PortalDataButton	Allows user to select the data path
StartErrOkButtonProc	StartErrOkButton	Closes the "Start Error" message box
StartError [Panel]	StartButton	Creates the "Start Error" message box if portal parameters are not acceptable

3.2.3 Stimulus and Preview

Procedure	Associated Control	Action

Stimuli [Panel]		Draws the Stimuli Window
InitStimVars		Initializes the stimulus window variables
VCPopMenuProc	StimVCPopup	Sets labels for DAC2 and tonic stimulation
DAC0PopMenuProc	DAC0TrainPop	Sets the DAC0 state to one of three possible values
DAC1PopMenuProc	DAC1TrainPop	Sets the DAC1 state to one of three possible values
DAC2PopMenuProc	DAC2Pop	Sets the DAC2 state to one of three possible values
MakeWaves		Makes the stimulus regimen
MakeParmsWave		Makes parameter wave for the stimulus regiment
StimPreviewButtonProc	StimPreviewButton	Opens up the preview window
AppendToPreviewGraph(n)		Appends the n-th trial to the preview graph
RemoveFromPreviewGraph(n)		Removes the n-th trial from the preview graph
MakeDAC0Wave(amp,lat,n, chan)		Makes the wave from DAC0 for trial n with amplitude given by amp, latency by lat and chan different channels
MakeDAC1Wave(amp,lat,n, chan)		Makes the wave from DAC1 for trial n with amplitude given by amp, latency by lat and chan different channels
MakeDAC2Wave(amp,dur,n, chan)		Makes the wave from DAC2 for Trial n with amplidute amp and duration dur and chan different channels
MakeTTL0Wave(chan)		Makes the TTL0 wave. Takes as input the number of stimulus channels
BadParameters [Panel]		This panel comes up when one attempts to make waves with incorrect stimulus parameters This can happen either in the stimulus window when you try to preview or in the acquisition window
BadParmsOkButton	BadParmsOkButton	This procedure kills the bad parameters window
StimSaveParms	StimParmsSaveButton	This creates the name of the parameters file
SaveParmsSave		This writes the current stimulation and acquisition parameters into a file
StimSensPopProc	SensitivePopUp	This sets the senstivity
StimPamrsLoadProc	StimParmsLoad	Opens a file selection window so that user can open a parameters file
Preview [Graph]		This creates the preview window
PreviewCurTrialSet	PreviewCurTrialSet	Appends new traces to preview

		graph and removes old ones
PreviewCloseButton	PreviewCloseButton	Closes the preview graph without killing the output waves
PreviewKillButton	PreviewKillButton	Closes preview graph and also kills the output waves **D0wave[n]** **D1wave[n]** **D2wave[n]** **P_head** **Pwave0** **TTL0_outwave**
NotBookSaveButtonProc	SaveNoteButton	Saves the notebook file
RepsSetVarProc	StimRepsSet	Sets the number of repetitions of each output stimulus
TonicSetProc	TonicAmp	This changes the DAC3 output when the value of the variable tonic_amp is changed with the TonicAmp control

3.2.4 Other

Procedure	**Associated Control**	**Action**
OutofMemOkButtonProc	OutofMemOkButton	Closes the out of memory panel
MemoryError [Panel]		Panel warns of memory overload
ZeroITC		Sets DACS to 0 and clears FIFO
SaveOrClear [Panel]		Tells user to save or clear
SaveOrClearOkButtonProc	SaveOrClearOkButton	Closes the Save or Clear panel
PleaseKillOutput [Panel]		Tells user to please kill output
PleaseKillOutputButtonProc	PleaseKillOutputButton	Closes the Please Kill Output Panel
QuitAndSaveButtonProc	QuitAndSaveButton	Quits and saves notebook
JustQuitButtonProc	JustQuitButton	Quits and does not save notebook
QuitPanel [Panel]		Panel which is called when quitting Slice

3.3 Waves

Wave	**Function**
In_temp	This is a temporary wave which holds a given input wave's data, undersampled by a factor of acq_compress to allow for data compresion
Inwave[n]	Input wave of data collected from ADC0
Multwave[n]	Multiplexed output wave
Inwave	Input wave for the baseline
Base_dummy	Dummy output wave when you take a baseline
P_head	Parameters for the parameters file which do not change from trial to trial
Pwave[n]	Parameters for the parameters file which do change from trial to trial

D0wave[n]	Wave for trial n from DAC0
D1wave[n]	Wave for trial n from DAC1
D2wave[n]	Wave for trial n from DAC2
D0_p_temp	Temporary wave for preview of DAC0
D1_p_temp	Temporary wave for preview of DAC1
D2_p_temp	Temporary wave for preview of DAC2
TTL0_outwave	Wave of TTL0 output signals
parmwave	Parameters wave saved to stimulus parameters file

3.4 Modifying Procedures

To modify procedures, open Slice as usual and open up the procedure window. Modify the code you wish to modify, and then go to the file menu and choose "Save Procedure". Continue to use Slice as usual and when you quit and re-start your modifications will be there.

4 Slice Configuration

4.1 Physical Setup

Slice was written specifically for the ITC18 Data Acquisition Interface by InstruTECH. The I/O voltage range of this device is -10.24 to 10.24 volts. Slice makes use of six channels on this instrument.

Channel	**Slice Function**
DAC Output 0 (DAC0)	Deliver pulses or trains to afferent fiber pathway
DAC Output 1 (DAC1)	Deliver pulses or trains to afferent fiber pathway
DAC Output 2 (DAC2)	Deliver voltage or current pulses or Voltage ramps to the cell
DAC Output 3 (DAC3)	Deliver tonic holding voltages or currents to the cell
ADC Input 0 (ADC0)	Records intracellular voltages or currents
TTL Output 0 (TTL0)	Outputs a TTL high (+5 V) during the Pre-stimulus and Acquisition periods

Outputs from DAC0 and DAC1 are sent into a Stimulus Isolator, which stimulates the afferent fiber pathways which converge upon the cell. The two isolator gains which we use most often are 1 and 10, with 10 being the default. One could in principle use the ITC18 DAC0/DAC1 output (with DAC01Gain = 1) to stimulate the fibers but this would limit the dynamic range to 10.24 V. The Grass stimultor gives us a range of 102.4 V.

Slice was developed for use with a PC-501A Patch Clamp from Warner Instruments. This device takes input voltages which it converts to either voltages or currents (depending on its mode of operation: VC or CC) to inject into the cell. The conversion factors were hard-coded into the program and so if one wants to use a different program one will need to make modifications to particular procedures. This is described in detail in the next section. Here is a table of the conversion factors:

DAC2 Sensitivity	**VC mode (V : mV)**	**CC mode (V : pA)**
0.1	1 : 100	1 : 1000
0.01	1 : 10	1 : 100
0.001	1 : 1	1 : 10

Inputs to the ADC0 channel on the ITC18 come from a Warner 505A. In current clamp mode, it outputs a voltage measured with a gain of 10. Slice takes this gain into account when figuring out the actual voltage of the cell by dividing its input by ten. In voltage clamp mode it also outputs a voltage which corresponds to the measured currents with 1 mV representing 1 pA of current.

4.2 Notes on the ITC18 Voltage Range

Voltages are represented using signed 16 bit integers. This means that the DAC converts the range (32768,32768) to (-10.24,10.24), and likewise the ADC maps the range (-10.24,10.24) onto (-32768,32768). This range is non-inclusive, meaning that the device cannot produce a voltage of precisely 10.24. What this means is that one cannot send out 32768 to the device. This will result in a sign bit error and you will get a negative output value. The highest number you can send out is 32767.

4.3 Modifying Slice for other Equipment

There are a number of variables and procedures which need to be modified if one wishes to use other equipment with different settings.

4.3.1 DAC0/1

The two values of *DAC01Gain* which are supported in this program are 1 and 10. If you are using equipment with a different gain, you will need to modify the range of this variable.

4.3.2 DAC2

If you are using equipment with different voltage to voltage or voltage to current conversion factors, then you will need to modify a number of procedures since these conversion factors were hard-coded into numerous procedures and are not set-variables. This should be very simple and does not require any in-depth knowledge of programming. The following procedures will need to be modified:

CheckDAC2Parms()
MakeDAC2Wave()
TonicSetProc()
AppendToPreviewGraph()

4.3.3 ADC0

The two values of current clamp gain (cc_gain) supported in this program are 1 and 10. If you are using equipment with different output gains in current clamp mode you will need to modify the range of this variable. If you are using equipment with different output gains in voltage clamp mode then you will need to modify the acquisition procedures *BaseButton()* and *StimGoButton()* as follows:

```
if(is_vc)
      $in = 1000*((10.24/32768))*$in
else
      $in = 1000*((10.24/32768)/cc_gain)*$in
endif
```

becomes:

```
if(is_vc)
      $in = 1000*((10.24/32768)/vc_gain)*$in
else
      $in = 1000*((10.24/32768)/cc_gain)*$in
endif
```

where vc_gain is either a variable or constant which represents the output gain of the device you are using to do your intracellular recordings.

5 Output File Format

5.1 Acquisition Parameters File

When the user acquires data and saves a set of traces two files are created: a parameters file , which is a text file and describes the set of stimulus parameters which were used to generate the family of curves as well as the parameters used in the larger experiment, plus the trial-by-trial values of the six variables which change from trial to trial in a family of outputs.

The structure of the parameters file is as follows: at the beginning of the file are the values of 72 parameters which do not change from trial to trial. After these parameters, there are 10n more parameters , which are the six following values during the i-th trial as i ranges from 1 to n.

DAC0Amplitude
DAC0Latency
DAC0Interval
DAC1Amplitude
DAC1Latency

DAC1Interval
DAC2 Pulse 1 Amplitude
DAC2 Pulse 2 Amplitude
DAC2 Pulse 1 Duration
DAC2 Pulse 2 Duration

Thus the number of rows in this text file will be 72 + 10n , where n is the number of trials in the family. In external trigger mode, only the header part of the parameters file gets saved since there is no output in this mode.

In this table is the structure of the header part of the parameters wave. This is essential to know if you want to write analysis software and to be able to study the effects of different parameter values on properties of the traces.

Header Wave Index	**Parameter Value**	**Comments**
0	Num_trials	Number of trials
1	Ph_len	Length of the header file (72)
2	Dat_len	Length of a wave
3	Trial_Period	
4	Prestim	
5	Srate	
6	Acquire	
7	Volts_div	
8	Sensitivity	
9	Is_vc	
10	Is_tonic	
11	Tonic_amp	
12	Reps	
13	DAC01Gain	
14	DAC0_state	
15	DAC1_state	
16	DAC2_state	
17	DAC0_amp_first	
18	DAC0_amp_last	
19	DAC0_amp_step	
20	DAC0_lat_first	
21	DAC0_lat_last	
22	DAC0_lat_step	
23	Ext_trig	
24	DAC0_dur	
25	DAC0_reps	
26	DAC1_amp_first	
27	DAC1_amp_last	
28	DAC1_amp_step	
29	DAC1_lat_first	
30	DAC1_lat_last	
31	DAC1_lat_step	
32	N_ext_trials	
33	DAC1_dur	
34	DAC1_reps	
35	DAC2_amp_first	
36	DAC2_amp_last	

37	DAC2_amp_step	
38	DAC2_dur_first	
39	DAC2_dur_last	
40	DAC2_dur_step	
41	DAC2_ramp_dur	
42	DAC2_ramp_start	
43	DAC2_ramp_stop	
44	Cc_gain	
45	Acq_compress	
46	Animal_num	
47	Animal_age	
48	Elect_resist	
49	Temp	
50	Site_string	
51	Prep_string	
52	Treat_string	
53	Dye_string	
54	Sol_string	
55	DAC0_name_string	
56	DAC1_name_string	
57	DAC2_name_string	
58	DAC0_int_first	
59	DAC0_int_last	
60	DAC0_int_step	
61	DAC1_int_first	
62	DAC1_int_last	
63	DAC1_int_step	
64	DAC2_amp2_first	
65	DAC2_amp2_last	
66	DAC2_amp2_step	
67	DAC2_dur2_first	
68	DAC2_dur2_last	
69	DAC2_dur2_step	
70	N_traces_col	
71	10	(length of the parms wave for each trial)

5.2 Data file

The data file is an IGOR binary file. All of the traces in the file are appended into one big wave of length (num_trials)*(data_length) which is written into the file. The trace for trial i for 1 < i < n is simply the set of points bigwave[(i-1)*data_length,(i*data_length)-1].

6 Stimulus Parameters File Structure

In the Stimuli window the user has the option of saving sets of stimulation parameters and loading saved parameter sets. The parameters are saved to a text file of the form p_xxx_i_j_k , where xxx is the initial string, i the animal number, j the cell number and k is the number of the parameter set. The "p" at the beginning is to quickly identify what type of file it is. The structure of the output wave is shown in the procedure *SaveParmsSave()*.