SHUTTLE BOX LEARNED HELPLESSNESS

MED-STATE NOTATION^{TM} PROCEDURE

SOF-700RA-28 Manual
DOC-235
Rev. 1.0

MED Associates Inc.
P.O. Box 319
St. Albans, Vermont 05478
www.med-associates.com

TABLE OF CONTENTS

Chapter 1 1

Introduction 1

Overview of Shuttle Box Learned Helplessness Training.mpc 2

Overview of Shuttle Box Learned Helplessness Test.mpc......................................2

Inputs and Outputs 4

Misc. Information 5

Chapter 2 6

Getting Started 6

Software Installation 6

Backing Up the Software 6

Chapter 3 7

Beginning & Running an Experiment 7

Translating the MED-PC (.mpc) File 7

Using the MED-PC Load Wizard 8

Viewing/Changing Variable Values 13

Macros 15

Modifying the MedState Notation^{TM} Code 16

Chapter 4 17

Modifying the Existing Program 17

Introduction (Shuttle Box Learned Helplessness Test.mpc)................................ 17

Chapter 5 28

Understanding the Data Files 28

Chapter 6 29

Data Analysis – Using MED-PC to Excel 29

Using a Pre-Formatted Table Profile (.MTP file) 29

Editing the .MTP file 32

CHAPTER 1
Introduction

The purpose of this manual is to give an explanation of the MED State Notation™ Procedures that comprise the SOF-700RA-28 Shuttle Box Learned Helplessness Procedures. The files in this package can be found on the disk provided by MED Associates, Inc.

These procedures are intended to be run in MED Associates MED-PC® software. The latest version of MED-PC® gives researchers the ability to use pre-programmed procedures such as these to make hardware control and data collection easy. These pre-programmed procedures can also be modified to meet the evolving demands of a research protocol. Again, it is the intent of this manual to explain exactly what these procedures implement, and provide guidance into how to interpret what the program code achieves in order to let the user determine how to modify them to match their research protocol demands. The manual provides some examples of editing and modifying the procedure's programming code. The manual also defines the elements in the raw data file produced by these procedures.

In addition to this manual, refer to the **MED-PC® User's Manual** for the installation of the MED-Associates interface drivers, the MED-PC Software, and the Delphi® Compiler. Also refer to the User's Manual for instructions on developing a Hardware Configuration. Data file structure, file-saving format, and other related options are also determined by the Hardware Configuration. Running the Hardware Configuration software utility that accompanies MED-PC sets the Hardware Configuration. Its purpose is to assign the inputs and outputs on the interface cards in the interface cabinet for each task controlled by MED-PC. The particular type of interface card that is supplied in the interface cabinet may vary; please refer to the User's Manual provided for instructions on how to configure the type of card that is in the cabinet. A valid Hardware Configuration must exist in order for MED-PC to interface correctly with the MED Associates, Inc. hardware. This means that one should take the time to create a valid Hardware Configuration before attempting to run the procedures included in this package.

Should there be any problems, the staff at MED Associates, Inc. is available to answer any questions that may arise. Please e-mail us at support@med-associates.com with a detailed description of the problem or desired goals so that concise and detailed information may be provided.

The Shuttle Box Learned Helplessness procedures are designed to be as easy to use as possible. MED Associates, Inc. understands that researchers do not have the time to devote to programming and hardware design, and for that reason, we have undertaken that burden for you. We sincerely hope that you are satisfied with the products and services we provide, and look forward to meeting your future experimental needs as your research program evolves.

Overview of Shuttle Box Learned Helplessness Training.mpc

The subject should be loaded into either side of the chamber. When a START command is issued, the program detects whether or not the subject has been loaded into the chamber. If so, the experiment begins with the Left and Right lights turning on, and the door opening.

Then the ITI Period begins, which is randomly drawn from a list and will be between two and 15 seconds in duration.

When the ITI Period ends, the Tone turns on for the Tone Duration (default five seconds), then the Aversive Stimulus (duration is randomly selected from a list and varies between one and ten seconds) will be delivered. The program will alternate between the ITI and the Tone/Aversive Stimulus until the specified Number of Trials to Run (default 180) has been delivered.

Overview of Shuttle Box Learned Helplessness Test.mpc

The subject should be loaded into either side of the chamber. When a START command is issued, the program detects whether or not the subject has been loaded into the chamber. If so, the Left and Right lights turn on, the door opens and the Acclimation Period (default two minutes) begins.

Following the Acclimation Period, the door closes for the length of the ITI Period, which is randomly drawn from a list and will be between two and 15 seconds in duration.

When the ITI Period ends, the Door opens, the Tone turns on on the side of the chamber that the subject is on, and the Trial begins. The trial begins with the Avoid Interval (default five seconds). If the subject crosses over to the other side of the chamber during the Avoid Interval, an Avoid is recorded to the data file and the Trial ends. If the subject does not cross over to the other side of the chamber before the end of the Avoid Interval, the Escape Interval (default ten seconds) begins, and if the CS/UCS overlap is set to the default value of ‘1’, the Tone will stay on until the aversive stimulus turns off.

When the Escape Interval begins, the aversive stimulus turns on. If the subject crosses over to the other side of the chamber during the Escape Interval, and Escape is recorded to the data file, the aversive stimulus turns off, and the Trial ends. At the end of the Escape Interval, if the subject has not cross over to the other side of the chamber, the aversive stimulus is turned off, a Failure to Escape is recorded in the data file and the Trial ends.

When the Trial ends, the Door closes and the Tone turns off. Then, another ITI Period, followed by another Trial begins until the Number of Trials to Run (default 30 trials) has been reached. When the Number of Trials to Run is reached, the experiment is over, the Lights turn off and the Door closes.

Independent Variables

Variable Name	Description	Default Value
Acclimation Period (minutes)	Amount of time for the subject to acclimate to the chamber.	2
Number of Trials to Run	Variable defines the number of trials. A trial ends and the ITI period begins when either the subject crosses to the other side of the chamber OR the Escape Interval ends, whichever happens first.	30
Avoid Interval (seconds)	Maximum duration of the conditioned stimulus (CS), prior to the delivery of the unconditioned stimulus (UCS). In this case the CS is the tone.	5
Escape Interval (seconds)	Maximum duration of the UCS. In this case, the UCS is the aversive stimulus. The UCS will terminate after the subject has crossed over to the other side of the chamber or when this time has elapsed, whichever happens first.	10
CS/UCS Overlap (1 = Yes; 0 = No)	If this variable is set to 0, the CS (tone stimulus) is terminated at the onset of the UCS (aversive stimulus). If this variable is set to 1, then the CS and the UCS will terminate in unison.	1

Dependent Variables

Avoids

The number of times that the subject crosses over to the other side of the chamber before the end of the Avoid Interval.

Escapes

The number of times that the subject crosses over to the other side of the chamber before the end of the Escape Interval.

Crossings

A crossing is defined as the subject moving from one side of the chamber to the other.

Failures to Escape

The number of times that the subject does not cross over to the other side of the chamber before the end of the Escape Interval.

Inputs and Outputs

Shuttle Box Learned Helplessness is intended to be used with shuttle box systems that include eight I/R Photobeam sensors, separate Left and Right Lights, a Tone, and may include an optional Auto-Guillotine Door.

All Inputs must be set to **LEVEL MODE**.

The #1 Beam is the beam closest to the Door or Hurdle on each side, counting up to the #4 beam as beams reach the outside of the chamber. Viewing from the front of the chamber, the inputs are assigned as shown below:

Figure 1.1 – IRs Numbered, Front of the Chamber

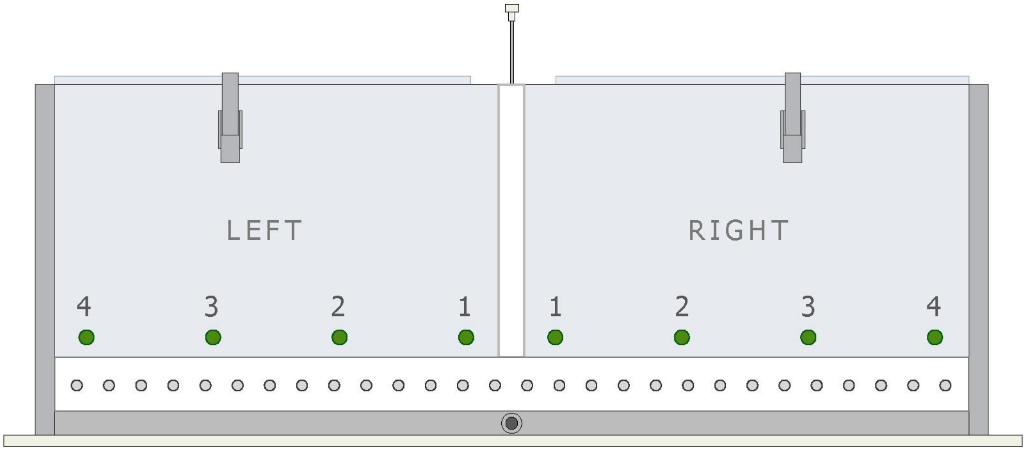
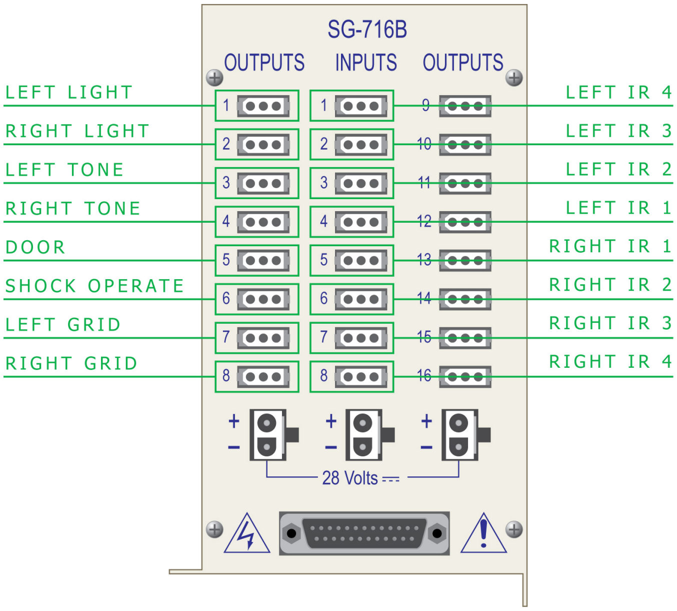


Figure 1.2 - Interface Panel Connections Labeled



Misc. Information

Data Array Dimensions

The dimensions of the data arrays are shown below. They are intended to allow the user to collect a maximum amount of data.

```
DIM A  =   4      \   Named Variables
DIM B  =   12     \   Summary Data Array
DIM D  =   6006   \   Trial Data Array
```

ITI Intervals

For both the Training and Test programs, the ITI Intervals are randomly chosen from a list in the program. The list must be defined as real numbers/times, and can be user-defined. The default values are shown below.

```
\ Available ITI Intervals
LIST Y = 2", 3", 4", 5", 6", 7", 8", 9", 10", 11", 12", 13", 14", 15"
```

For the Training program, the aversive stimulus durations are also randomly chosen from a list in the program. The list must be defined as real numbers/times, and can be user-defined. The default values are shown below.

```
\ Available Aversive Stimulus Lengths
LIST Z = 1", 2", 3", 4", 5", 6", 7", 8", 9", 10"
```

Array Columns and Number Format

Set the Array Columns and Number Format in ASCII Data File, where the columns are 6 and the number format is xxxxxx.xx.

```
DISKCOLUMNS = 6
DISKFORMAT = 6.2
```

CHAPTER 2

Getting Started

Software Installation

Please refer to the **MED-PC User's Manual** for a complete guide to installing the MED-PC software, building a valid Hardware configuration with the Hardware Configuration utility, and opening and compiling a MSN procedure in the Trans-IV utility.

To install the Shuttle Box Learned Helplessness Procedures, insert the CD into the CD-ROM drive and click **Install the Shuttle Box Learned Helplessness Software**. The procedures are copied into the indicated folder.

Backing Up the Software

Med Associates strongly encourages creating backup copies of the programs in case of disk failure. Having copies of the original programs may be useful in the future should modifications be made to the existing programs.

CHAPTER 3

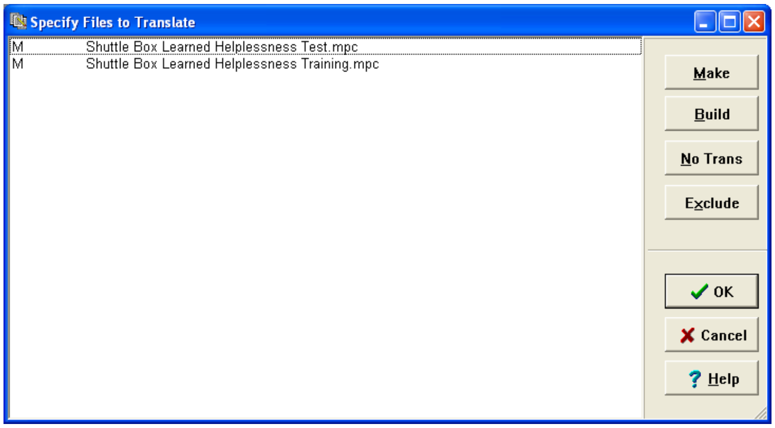
Beginning & Running an Experiment

Translating the MED-PC (.mpc) File

Programs written in MedState Notation must be translated using Trans IV before they can be executed in this application. Open Trans IV icon and select **Translation** | **Translate and Compile**.

Select the program(s) to use for the experiment and click **Make**. Click **OK** to start the translator, and it will automatically parse the MedState Notation and then open to a DOS screen to compile the Pascal code. Depending on the speed of the computer, each of these steps may not be seen. If any problems are encountered during this process, refer to the on-screen help menu or the **MED-PC User's Manual**, or contact MED Associates, Inc. for assistance.

Figure 3.1 - Trans IV Control Panel for Translating and Compiling MedState Notation Code



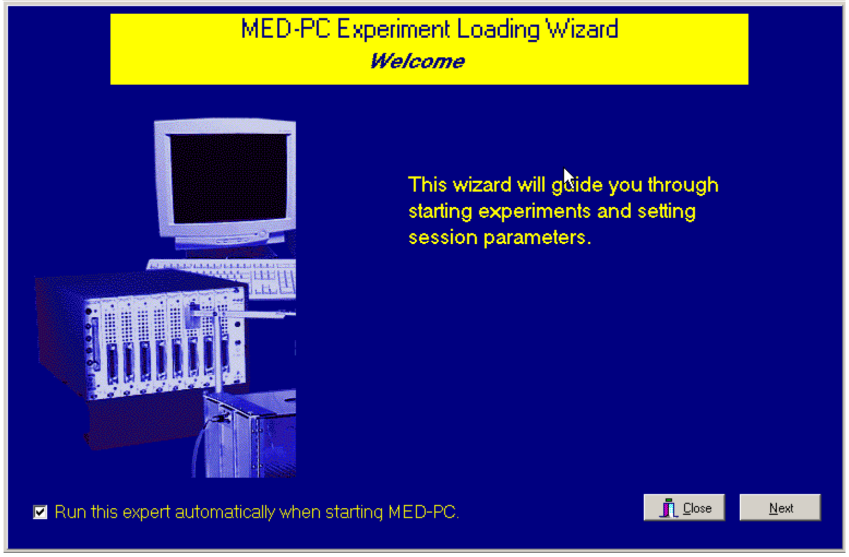
```
M          Shuttle Box Learned Helplessness Test.mpc
M          Shuttle Box Learned Helplessness Training.mpc
```

Using the MED-PC Load Wizard

MED-PC is designed to help the researcher run an experiment by guiding selection choices through its Experiment Loading Wizard. This section will describe how to initiate the Shuttle Box Learned Helplessness Test.mpc application, however the following steps apply to all other .mpc procedures.

Open MED-PC and the MED-PC Experiment Loading Wizard's Welcome screen, shown in Figure 3.2 will appear.

Figure 3.2 - The MED-PC Loading Wizard Welcome Screen



MED-PC Experiment Loading Wizard

Welcome

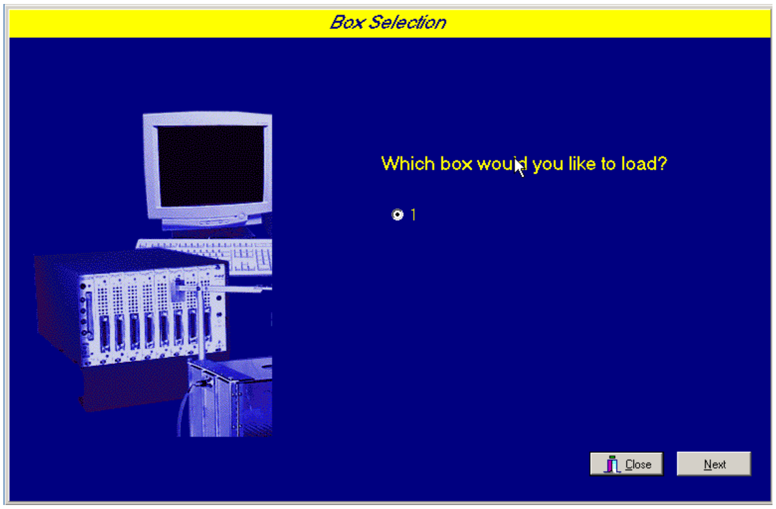
This wizard will guide you through starting experiments and setting session parameters.

☑ Run this expert automatically when starting MED-PC.

To avoid this load wizard, deselect the checkbox labeled **Run this experiment automatically when starting MED-PC**. Close this screen by clicking the **Close** button. Closing this screen immediately reveals the MED-PC Run-Time Screen shown in Figure 3.10. If the choice to continue with the Loading Wizard is made, then click the **Next** button.

The Box Selection screen will appear next, as shown in Figure 3.3. From this screen the researcher chooses which boxes will be used in the experiment. Select the boxes that will run the experiment by clicking in the radio button next to the box number. The figure shows that the Hardware Configuration included only 1 box, which was selected. Click **Next** to continue.

Figure 3.3 - The Box Selection Screen



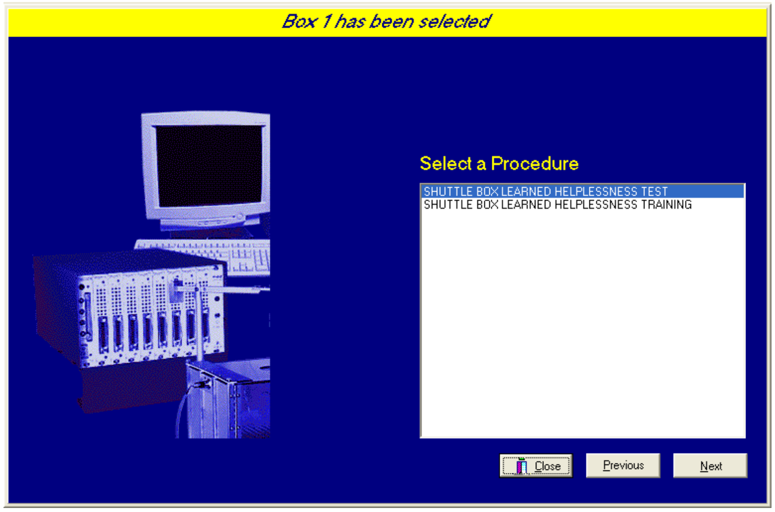
Box Selection

Which box would you like to load?

● 1

The Select a Procedure screen appears next, shown in Figure 3.4. This screen displays a list of all the currently compiled procedures. Highlight the desired procedure and click **Next**.

Figure 3.4 - The Select a Procedure Screen



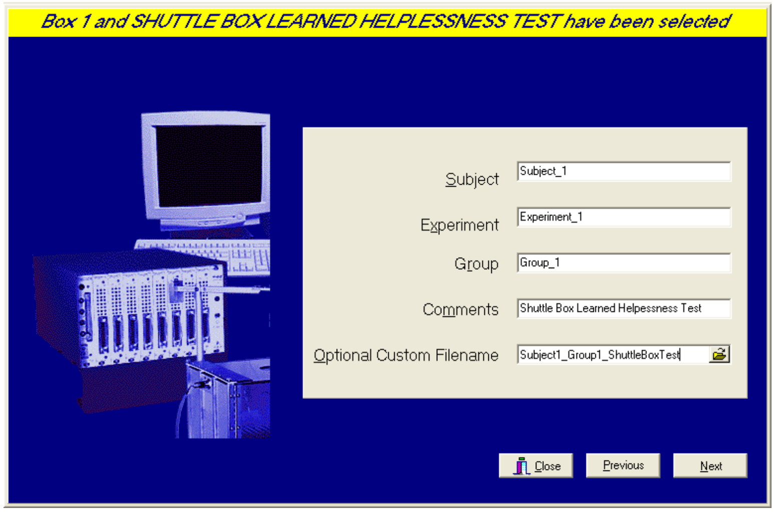
Box 1 has been selected

Select a Procedure

SHUTTLE BOX LEARNED HELPLESSNESS TEST
SHUTTLE BOX LEARNED HELPLESSNESS TRAINING

Next, the Enter Experiment Data Screen will appear, shown in Figure 3.5. This screen allows the user to add annotations to the data file that is produced by MED-PC. These annotations will help identify the Subject, Experiment, and Experiment Group from which data was collected. Comments can be added here as well, and the data file can be given a customized file name to help identify it from other data files. Enter the information desired, and click **Next**.

Figure 3.5 - Enter Experiment Data Screen



Box 1 and SHUTTLE BOX LEARNED HELPLESSNESS TEST have been selected

Subject: Subject_1

Experiment: Experiment_1

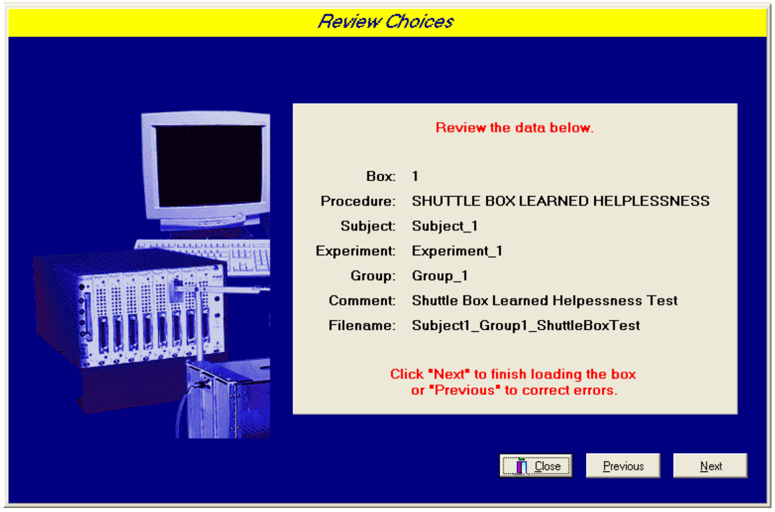
Group: Group_1

Comments: Shuttle Box Learned Helpessness Test

Optional Custom Filename: Subject1_Group1_ShuttleBoxTest

The next screen to appear is the Review Choices screen, shown in Figure 3.6. This screen allows the user to confirm that the information that was entered on previous screens. If it is not correct, click **Previous** to go back to the previous screen and edit the information. If it is correct, click **Next** to proceed.

Figure 3.6 - Review Choices Screen



Review Choices

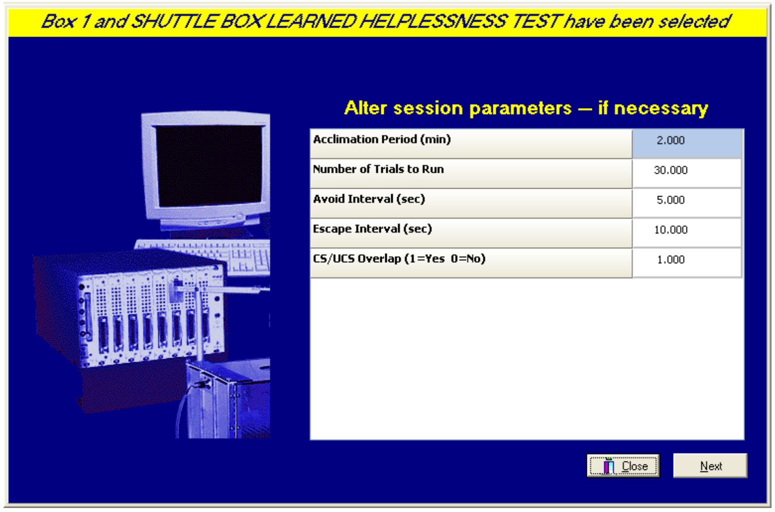
Review the data below.

Box: 1
Procedure: SHUTTLE BOX LEARNED HELPLESSNESS
Subject: Subject_1
Experiment: Experiment_1
Group: Group_1
Comment: Shuttle Box Learned Helpessness Test
Filename: Subject1_Group1_ShuttleBoxTest

Click "Next" to finish loading the box
or "Previous" to correct errors.

The next screen to appear is the Alter Session Parameters Screen, shown in Figure 3.7. This screen allows the user to alter the parameters by which a procedure executes. Make any desired changes to the session parameters and click **Next** to proceed.

Figure 3.7 - Alter Session Parameters Screen



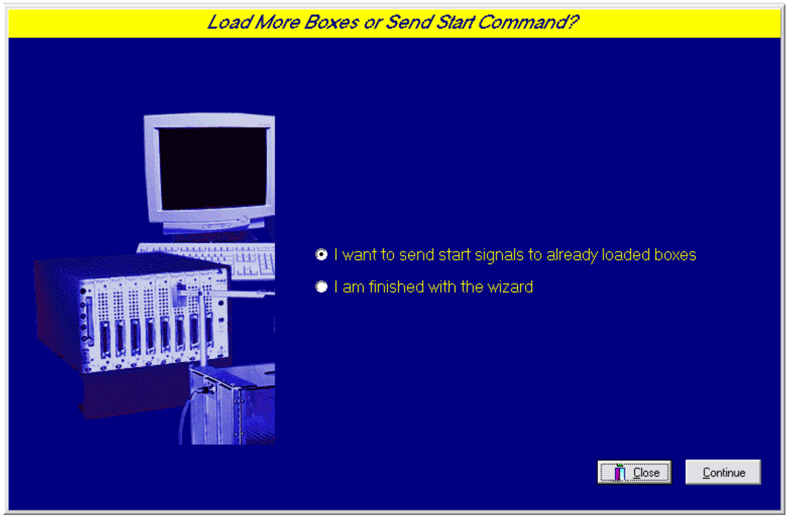
Box 1 and SHUTTLE BOX LEARNED HELPLESSNESS TEST have been selected

Alter session parameters — if necessary

Acclimation Period (min)	2.000
Number of Trials to Run	30.000
Avoid Interval (sec)	5.000
Escape Interval (sec)	10.000
CS/UCS Overlap (1=Yes 0=No)	1.000

The next screen to appear is the Send Start Command screen. The options available on this screen vary depending upon the number of boxes entered in the Hardware Configuration Utility (refer to the MED-PC IV User's Manual for more information regarding the Hardware Configuration Utility). In this example, there is only one box attached to the interface, so Figure 3.8 will appear next. If more than one box is attached to the interface, then Figure 3.9 will appear next.

Figure 3.8 - Send Start Command Screen for Single Box Configuration

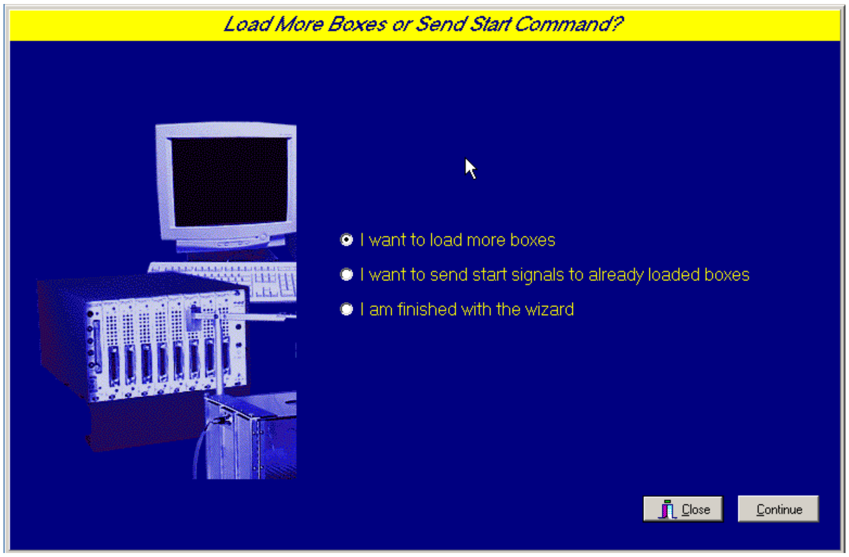


Load More Boxes or Send Start Command?

● I want to send start signals to already loaded boxes

○ I am finished with the wizard

Figure 3.9 - Send Start Command Screen for Multiple Box Configuration



Load More Boxes or Send Start Command?

◉ I want to load more boxes

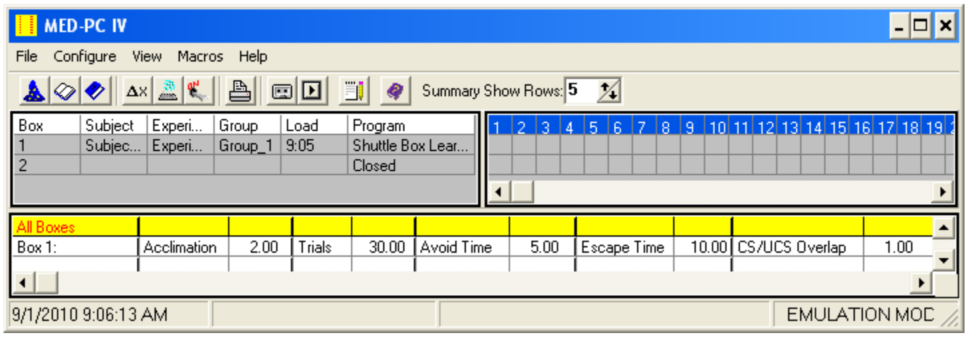
○ I want to send start signals to already loaded boxes

○ I am finished with the wizard

In both cases (Figure 3.8 and Figure 3.9), this screen is where the user decides to either load more boxes, send a start signal to boxes that are already loaded, or enter the MED-PC run-time environment without sending a start signal by selecting "I am finished with the wizard". Make a selection and click **Continue**.

For the purposes of this example, "I am finished with the wizard" was selected. This option results in the MED-PC IV run-time screen appearing, as shown in Figure 3.10.

Figure 3.10 - The MED-PC Run-Time Screen



Summary Show Rows: 5

Box	Subject	Experi...	Group	Load	Program
1	Subjec...	Experi...	Group_1	9:05	Shuttle Box Lear...
2					Closed

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

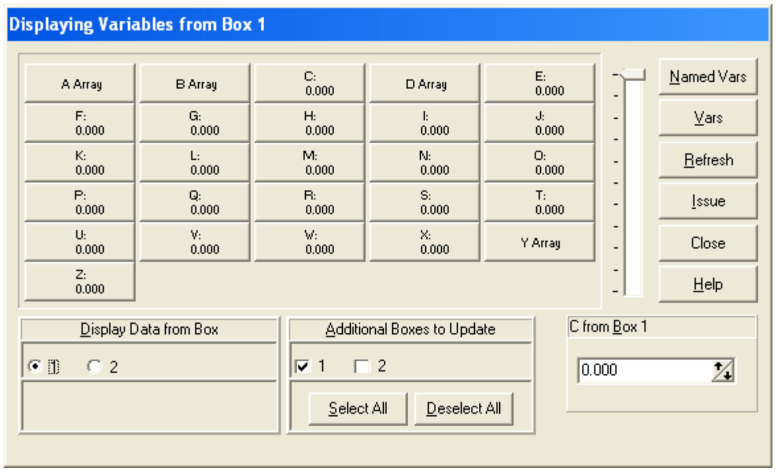
All Boxes										
Box 1:	Acclimation	2.00	Trials	30.00	Avoid Time	5.00	Escape Time	10.00	CS/UCS Overlap	1.00

9/1/2010 9:06:13 AM EMULATION MOC

Viewing/Changing Variable Values

Once a session is in progress, the session parameters can be changed by selecting **Configure | Change Variables**, or click the 4th tool bar item **∆X.** In the lower left hand corner of the Change Variables window, find the "Display Data from Box" display, and choose the chamber(s) to modify. By clicking additional boxes in the "Additional Boxes to Update" section, changes made to a single box are automatically loaded to all of the selected boxes.

Figure 3.11 - Changing Variables Screen



Displaying Variables from Box 1

A Array	B Array	C: 0.000	D Array	E: 0.000
F: 0.000	G: 0.000	H: 0.000	I: 0.000	J: 0.000
K: 0.000	L: 0.000	M: 0.000	N: 0.000	O: 0.000
P: 0.000	Q: 0.000	R: 0.000	S: 0.000	T: 0.000
U: 0.000	V: 0.000	W: 0.000	X: 0.000	Y Array
Z: 0.000				

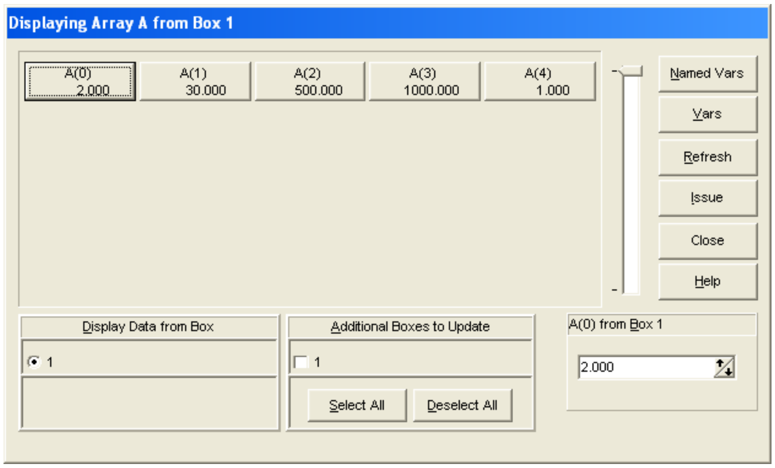
Display Data from Box: ● 1 ○ 2

Additional Boxes to Update: ☑ 1 □ 2

C from Box 1: 0.000

The value of any simple variable may be viewed from this screen by clicking an array on the table and each element in that array can be viewed, as shown in Figure 3.12. To change a value, simply highlight and replace the value in the lower right hand box or use the up/down arrows to increment by 1. Click the **Issue** button for the change to take effect. Click **Named Variables** to produce the display in Figure 3.13. Change variables here as needed.

Figure 3.12 - Displaying Array A from Box 1



Displaying Array A from Box 1

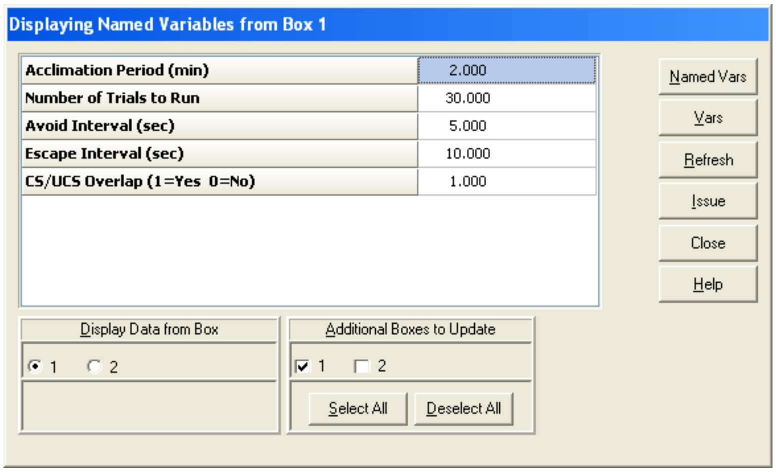
A(0)	A(1)	A(2)	A(3)	A(4)
2.000	30.000	500.000	1000.000	1.000

Display Data from Box: ● 1

Additional Boxes to Update: □ 1

A(0) from Box 1: 2.000

Figure 3.13 - Displaying Named Variables from Box 1



Displaying Named Variables from Box 1

Acclimation Period (min)	2.000
Number of Trials to Run	30.000
Avoid Interval (sec)	5.000
Escape Interval (sec)	10.000
CS/UCS Overlap (1=Yes 0=No)	1.000

Display Data from Box: ● 1 ○ 2

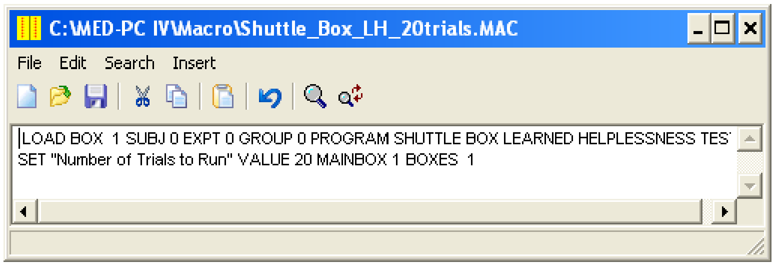
Additional Boxes to Update: ☑ 1 □ 2

Macros

The simplest way to initially create a macro is to record keyboard functions while performing the steps manually. Once the commands are in the macro, it is easy to create a number of macros with the macro editor. The following example illustrates the process of loading "Box 1" and changing the Number of Trials to Run to 20.

To begin, open MED-PC and going directly to the run time screen. Close the load wizard, if present. Before loading or opening the procedure, click **Macro** on the main menu and select **Turn On Macro Recorder** or click the 8th tool bar item with the cassette tape icon on top. A note on the bottom of the display indicates that the recorder is running. Open "Shuttle Box Learned Helplessness Test.mpc" by clicking **File | Open Session**. Change the variables using any of the methods described above. When all settings have been made, turn the recorder off again by using the main menu or tool bar. Save the macro with a distinctive name. The example in Figure 3.14 was named "Shuttle_Box_LH_20trials.mac" since the Number of Trials to Run was changed to 20.

Figure 3.14 – Shuttle_Box_LH_20trials.mac



```
LOAD BOX  1 SUBJ 0 EXPT 0 GROUP 0 PROGRAM SHUTTLE BOX LEARNED HELPLESSNESS TES
SET "Number of Trials to Run" VALUE 20 MAINBOX 1 BOXES  1
```

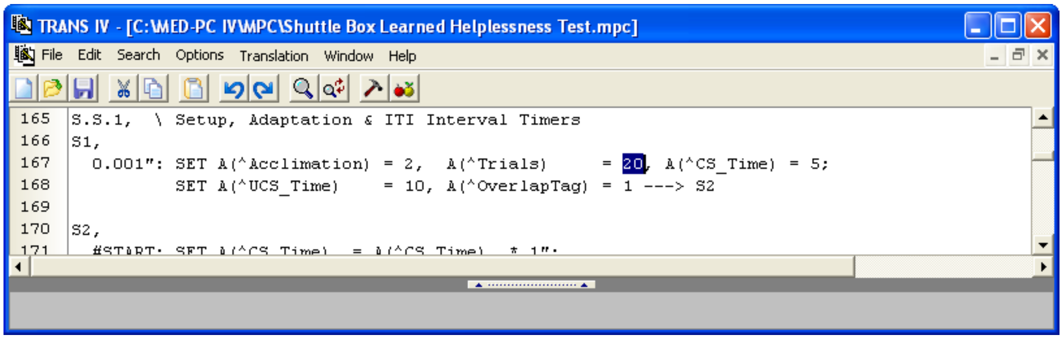
Once this macro is built, use the macro editor in MED-PC IV to make simple changes such as replacing reward or correct lever values. Open the macro editor by selecting **Macros | Editor**.

Review the Help file on screen or the **MED-PC User's Manual** for more information on macros and the features offered. A START command or message box followed by a START command could be added to the macro (it was left off here so changes could be verified before starting the procedure).

Modifying the MedState Notation^{TM} Code

Permanent changes to the Shuttle Box Learned Helplessness procedures can be made to the MedState Notation code. To make the same change to the Number of Trials to Run as shown above, do the following. Open Trans IV and select **File** | **Open** to place Shuttle Box Learned Helplessness Test.mpc into the text editor. Scroll down to approximately line 167 (note the line counter on the left side of the editor) to reveal the code shown in Figure 3.15.

Figure 3.15 – Shuttle Box Learned Helplessness Test.mpc Line 167


TRANS IV - [C:\MED-PC IV\MPC\Shuttle Box Learned Helplessness Test.mpc]
File
Edit
Search
Options
Translation
Window
Help

```
165  S.S.1,   \ Setup, Adaptation & ITI Interval Timers
166  S1,
167    0.001": SET A(^Acclimation) = 2,  A(^Trials)      = 20, A(^CS_Time) = 5;
168            SET A(^UCS_Time)     = 10, A(^OverlapTag) = 1 ---> S2
169
170  S2,
171    #START: SET A(^CS_Time)   = A(^CS_Time)   * 1";
```

Change A(^Trials) =30 to A(^Trials) =20 and save the changes with the same or a new file name such as Shuttle Box LH Test 20 Trials.mpc. Remember, if creating a new .mpc file name and are using a macro to load boxes, the file name in the macro also must be changed. Translate and compile the new or changed file as described previously and run MED-PC. Use the "Change Variables" screen to view/confirm the new values.

CHAPTER 4

Modifying the Existing Program

Introduction (Shuttle Box Learned Helplessness Test.mpc)

The Shuttle Box Learned Helplessness procedure was written to run for a specified number of trials (default = 30). Shuttle Box Learned Helplessness Test.mpc is listed below in its entirety followed by a brief explanation of each state set.

See the **MED-PC IV User's Manual** for details regarding the processing of commands in MED-PC. The **MED-PC IV Programmer's Manual** and online tutorial provides necessary information on Med-State Notation, and these resources will be very helpful when making adjustments to Shuttle Box Avoidance Level.mpc, or creating a custom .mpc program.

Shuttle Box Learned Helplessness Test.mpc

```
\ Copyright (C) 2010 MED Associates, All Rights Reserved.

\ Shuttle Box Learned Helplessness Test.mpc
\
\ When the program is STARTed the Left and Right lights will be turned on.  The
\ Animal can be loaded into the chamber either before or after the START command
\ has been issued.  Once the program has detected that the Animal has been
\ loaded and the START command has been issued the Door will open and the
\ Acclimation Period will begin.
\
\ The program will start with the Acclimation Period.  After the Acclimation
\ Period ends the Door will be closed for the length of the ITI Period (randomly
\ drawn from a list).  When the ITI Period ends the Door will be reopened and
\ the Tone will be turned on, on the side that the Animal is located.  The
\ Animal will have the amount of time specified in the Avoid Interval to Avoid
\ the shock and the amount of time specified in the Escape Interval to Escape
\ from the Shock.  The Trial ends and the ITI Period begins when either the
\ Animal crosses to the other side or the Escape Interval ends.  When the Trial
\ ends the Shock will be turned off.  After the ITI Period (randomly drawn from
\ a list) ends the Door will open again and the next Trial will start.
\
\ Program runs only in Level Mode.


\ The #1 Beam is the Beam Closest to the Door or Hurdle on each Side.
\ Inputs are Assigned from Left to Right as viewed from front of Box.

\ Inputs
^LeftIR_4  = 1
^LeftIR_3  = 2
^LeftIR_2  = 3
^LeftIR_1  = 4
^RightIR_1 = 5
^RightIR_2 = 6
^RightIR_3 = 7
^RightIR_4 = 8

\ Outputs
^LeftLight    = 1
^RightLight   = 2
^LeftTone     = 3
^RightTone    = 4
^Door         = 5
^ShockOperate = 6
^LeftGrid     = 7
^RightGrid    = 8


\ Control Variables with Assigned Aliases as Defined
Var_Alias Acclimation Period (min)        = A(0)  \ Default = 2 minutes
Var_Alias Number of Trials to Run         = A(1)  \ Default = 30
Var_Alias Avoid Interval (sec)            = A(2)  \ Default = 5 seconds
Var_Alias Escape Interval (sec)           = A(3)  \ Default = 10 seconds
Var_Alias CS/UCS Overlap (1=Yes  0=No)    = A(4)  \ Default = Yes (Tone is On During Shock)
```

```
\ Subscript Constants for Control Variables
^Acclimation = 0
^Trials      = 1
^CS_Time     = 2
^UCS_Time    = 3
^OverlapTag  = 4


\ List Data Variables Here
\  B() = Summary Data Array
\  B(0)  = Total Trials Run
\  B(1)  = Total Movement in Left Chamber
\  B(2)  = Total Movement in Right Chamber
\  B(3)  = Total Number of Crossings
\  B(4)  = Total Failures to Escape            \ Defined as no crossing while the Shock is
turned on
\  B(5)  = Total Time spent in Left Chamber
\  B(6)  = Total Time spent in Right Chamber
\  B(7)  = Total Number of Avoids
\  B(8)  = Mean Avoid Latency                \ Divided by Total Avoids for Mean Latency
\  B(9)  = Total Number of Escapes
\  B(10) = Mean Escape Latency               \ Divided by Total Escapes for Mean Latency
\  B(11) = Total ITI Activity
\  B(12) = Adjusted Mean Escape Latency

\ Subscript Constants for Summary Data Array
^TrialCount     = 0
^LeftMovement   = 1
^RightMovement  = 2
^TotalCrossings = 3
^TotalFailures  = 4
^LeftTime       = 5
^RightTime      = 6
^TotalAvoids    = 7
^AvoidLatency   = 8
^TotalEscapes   = 9
^EscapeLatency  = 10
^ITIActivity    = 11
^EscapeAdjusted = 12

\  D() = Trial by Trial Data Array
\  D(I)    = Trial Number                   \ Trial 0 is the Acclimation Period
\  D(I+1)  = Left Movement Activity
\  D(I+2)  = Right Movement Activity
\  D(I+3)  = Crossings
\  D(I+4)  = Failure to Escape Tag
\  D(I+5)  = Left Time
\  D(I+6)  = Right Time
\  D(I+7)  = Avoid Tag
\  D(I+8)  = Avoid Latency
\  D(I+9)  = Escape Tag
\  D(I+10) = Escape Latency
\  D(I+11) = ITI Activity
\  D(I+12) = Adjusted Escape Latency
\ List Working Variables Here
\  E = Elapsed Time in Seconds for Acclimation Period and Session
\  I = Subscript for Data Array D
\  L = Location Flag (1 = Left, 2 = Right)
\  M = Trial ITI Drawn from List Y
\
\  Y() = List of ITI times 2 - 15 seconds.
\        This list may be edited as needed.  These numbers appear
\        in MED time units, i.e. 1 second = 100 MED Ticks given
\        a Resolution setting of 10 ms


\ Z-Pulses Used in This Procedure
^Start_CS    = 1   \ Z1  = Signal Start of CS
^Avoid       = 2   \ Z2  = Signal Avoid
^Escape      = 3   \ Z3  = Signal Escape
^Start_A     = 4   \ Z4  = Signal Start of Acclimation Period
^Start_ITI   = 5   \ Z5  = Signal Start of ITI
^EnterLeft   = 6   \ Z6  = Entrance to Left Compartment (must break beam 2 - 4)
^EnterRight  = 7   \ Z7  = Entrance to Right Compartment
^End_CS      = 8   \ Z8  = Signal no CS/UCS Overlap (Stim Off)
^EndSession  = 9   \ Z9  = End of Session
^One         = 21  \ Z21 = Beam 1 Break
^Two         = 22  \ Z22 = Beam 2 Break
^Three       = 23  \ Z23 = Beam 3 Break
^Four        = 24  \ Z24 = Beam 4 Break
^Five        = 25  \ Z25 = Beam 5 Break
^Six         = 26  \ Z26 = Beam 6 Break
^Seven       = 27  \ Z27 = Beam 7 Break
^Eight       = 28  \ Z28 = Beam 8 Break


\ Dimension Array Sizes
DIM A = 4     \ Named Variables
DIM B = 12    \ Summary Data Array
DIM D = 6006  \ Trial Data Array

\ Available ITI Intervals
LIST Y = 2", 3", 4", 5", 6", 7", 8", 9", 10", 11", 12", 13", 14", 15"


DISKCOLUMNS = 13
DISKFORMAT  = 6.2
```

DESCRIPTION OF STATE SET 1

State Set 1 is the core of program and sets the default values for all independent variables (S1), programs the load wizard setup application (S2), starts the procedure on animal entry (S3), times the acclimation period (S4), waits for the end of each trial (S5), times the ITI (S6), sets up a new trial (S7), and saves the data (S10-13).

```
\***********************************************
\ Shuttle Box Learned Helplessness Test Schedule
\  S1 - Set Default Values
\      Acclimation Period         (2 minutes)
\      Number of Trials to Run    (30)
\      Avoid Interval             (5 seconds)
\      Escape Interval            (10 seconds)
\      CS/UCS Overlap             (Yes)
\***********************************************
S.S.1,  \ Setup, Adaptation & ITI Interval Timers
S1,
  0.001": SET A(^Acclimation) = 2,  A(^Trials)     = 30, A(^CS_Time) = 5;
          SET A(^UCS_Time)    = 10, A(^OverlapTag) = 1 ---> S2

S2,
  #START: SET A(^CS_Time)  = A(^CS_Time)  * 1";
          SET A(^UCS_Time) = A(^UCS_Time) * 1";
          SET D(I+13)      = -987.987;
          ON ^LeftLight, ^RightLight ---> S3
  1": SHOW 1,Acclimation,A(^Acclimation), 2,Trials,A(^Trials), 3,Avoid Time,A(^CS_Time);
      SHOW 4,Escape Time,A(^UCS_Time),    5,CS/UCS Overlap,A(^OverlapTag) ---> SX
```

```
S3,     \ Verify Detection of Animal
  0.01": IF (L = 1) OR (L = 2) [@StartNow, @Wait]
            @Start: ON ^Door; Z^Start_A ---> S4
            @Wait: ---> SX

S4,     \ Time Acclimation Period
  1": ADD E; SHOW 5,Acclimation Time,E/60;
      IF E/60 >= A(^Acclimation) [@Done, @Cont]
         @Done: OFF ^Door; RANDD M = Y;
                SET E = 0; Z^Start_ITI ---> S6
         @Cont: ---> S4
  #Z^EnterLeft ! #Z^EnterRight: ADD D(I+3), B(^TotalCrossings) ---> SX

S5,     \ Wait for End of Trial
  #Z^Avoid ! #Z^Escape ! #Z^Start_ITI: OFF ^Door ---> S6

S6,     \ Time ITI Interval.  End Session following completion of
        \ ITI if Session Time or Number of Trials completed is
        \ equal to or greater than the set values.
  M#T: IF B(^TrialCount) >= A(^Trials) [@EndSession, @NewTrial]
          @End: Z^EndSession ---> S10
          @New: ON ^Door ---> S7

S7,     \ Set up new Trial
  0.01": ADD B(^TrialCount); SET I = I + 13;
         SET D(I) = B(^TrialCount), D(I+13) = -987.987;
         SHOW 1,Trial #,D(I); RANDD M = Y; Z^Start_CS ---> S5

S10,    \ End of Session Calculations - Check for Zero Avoids
  0.01": IF B(^TotalAvoids) = 0 [@True, @False]
            @True: ---> S11
            @False: SET B(^AvoidLatency) = B(^AvoidLatency) / B(^TotalAvoids) ---> S11

S11,    \ Check for Zero Escapes
  0.01": IF B(^TotalEscapes) = 0 [@True, @False]
            @True: ---> S12
            @False: SET B(^EscapeLatency) = B(^EscapeLatency) / B(^TotalEscapes) ---> S12

S12,    \ Calculate the Adjusted Mean Escape Latency
  0.01": SET B(^EscapeAdjusted) = B(^EscapeAdjusted) / B(^TrialCount) ---> S13

S13,    \ Delay for Screen Update
  2": OFF ^Door ---> STOPABORTFLUSH
```

DESCRIPTION OF STATE SET 2

State Set 2 tracks the location of the animal in the chamber.

```
\*********************************************
\            TRACK ANIMAL LOCATION
\*********************************************
S.S.2,  \ Set Location Flag L and issue Location Z-Pulse
S1,
  #R^LeftIR_1  ! #R^LeftIR_2  ! #R^LeftIR_3  ! #R^LeftIR_4:  SET L = 1 ---> S2
  #R^RightIR_1 ! #R^RightIR_2 ! #R^RightIR_3 ! #R^RightIR_4: SET L = 2 ---> S4

S2,     \ Animal on Left Grid, wait for Beam break on Right Side
  #R^RightIR_1 ! #R^RightIR_2 ! #R^RightIR_3 ! #R^RightIR_4: ---> S3
  #Z^EndSession:                                             ---> S6

S3,     \ Check to see if any Left Side Beams are being broken, if not then
        \ count it as a crossing to the Right Side.
  #R^LeftIR_4 ! #R^LeftIR_3 ! #R^LeftIR_2 ! #R^LeftIR_1: ---> S2
  0.02": SET L = 2; Z^EnterRight                         ---> S4
  #Z^EndSession:                                         ---> S6

S4,     \ Animal on Right Grid, wait for Beam break on Left Side
  #R^LeftIR_1  ! #R^LeftIR_2  ! #R^LeftIR_3  ! #R^LeftIR_4: ---> S5
  #Z^EndSession:                                            ---> S6

S5,     \ Check to see if any Right Side Beams are being broken, if not then
        \ count it as a crossing to the Left Side.
  #R^RightIR_1 ! #R^RightIR_2 ! #R^RightIR_3 ! #R^RightIR_4: ---> S4
  0.02": SET L = 1; Z^EnterLeft                              ---> S2
  #Z^EndSession:                                             ---> S6

S6,     \ Holding State at End of Session
  1': ---> SX
```

DESCRIPTION OF STATE SET 3

State Set 3 controls the trial sequence for when the subject starts the trial on the left (S1, S5-S8) and right (S10-S13) side of the chamber.

```
\*********************************************
\     TRIAL SEQUENCE AND DATA COLLECTION
\*********************************************
S.S.3,
S1,     \ Initiate Trials
  #Z^Start_CS: IF L = 1 [@LeftTrial, @Next]
                  @LeftTrial: ---> S5
                  @Next: IF L = 2 [@RightTrial, @Error]
                            @RightTrial: ---> S10
                            @Error:      ---> SX

S5,     \ Trial Sequence Starting from Left Chamber - CS Period
        \ Activate Shock - Left Side
  A(^CS_Time)#T: ON ^LeftGrid;
                 IF A(^OverLapTag) >= 1 [@OverLap, @NoOverLap]
                    @OverLap:   ON ^ShockOperate           ---> S7
                    @NoOverLap: ON ^ShockOperate; Z^End_CS ---> S7
  #Z^EnterRight: ADD D(I+7), B(^TotalAvoids);
                 ADD D(I+3), B(^TotalCrossings);
                 Z^Avoid; Z^End_CS ---> S1

S7,     \ UCS Period - Left Chamber
  A(^UCS_Time)#T: OFF ^ShockOperate, ^LeftGrid;
                  ADD D(I+4), B(^TotalFailures);
                  Z^Start_ITI; Z^End_CS ---> S1
  #Z^EnterRight: ADD D(I+9), B(^TotalEscapes);
                 ADD D(I+3), B(^TotalCrossings);
                 Z^Escape; Z^End_CS ---> S8

S8,     \ Shock OFF Delay to Force a Complete Crossing
  1.5": OFF ^ShockOperate, ^LeftGrid ---> S1

\------------------------------------------------------
\ S10 - S13 are identical in function to S5 - S8 Above
\------------------------------------------------------
S10,    \ Trial Sequence Starting from Right Chamber - CS Period
        \ Activate Shock - Right Side
  A(^CS_Time)#T: ON ^RightGrid;
                 IF A(^OverLapTag) >= 1 [@OverLap, @NoOverLap]
                    @OverLap:   ON ^ShockOperate           ---> S12
                    @NoOverLap: ON ^ShockOperate; Z^End_CS ---> S12
  #Z^EnterLeft: ADD D(I+7), B(^TotalAvoids);
                ADD D(I+3), B(^TotalCrossings);
                Z^Avoid; Z^End_CS ---> S1

S12,    \ UCS Period - Right Chamber
  A(^UCS_Time)#T: OFF ^ShockOperate, ^RightGrid;
                  ADD D(I+4), B(^TotalFailures);
                  Z^Start_ITI; Z^End_CS ---> S1
  #Z^EnterLeft: ADD D(I+9), B(^TotalEscapes);
                ADD D(I+3), B(^TotalCrossings);
                Z^Escape; Z^End_CS ---> S13

S13,    \ Shock OFF Delay to Force a Complete Crossing
  1.5": OFF ^ShockOperate, ^RightGrid ---> S1
```

DESCRIPTION OF STATE SET 4

State Set 4 counts the ITI activity of the subject

```
\*********************************************
\             COUNT ITI ACTIVITY
\*********************************************
S.S.4,
S1,
  #START: ---> S2

S2,     \ Wait for Start of ITI
  #Z^Avoid ! #Z^Escape ! #Z^Start_ITI: IF L = 1 [@Left, @Right]
                                          @Left:  ---> S3
                                          @Right: ---> S8

\------------------- Animal on Left -------------------
S3,     \ Wait for first Beam Break
  #Z^EndSession: ---> S1
  #Z^Start_CS:   ---> S2  \ End of ITI
  #Z^Four:       ---> S4
  #Z^Three:      ---> S5
  #Z^Two:        ---> S6
  #Z^One:        ---> S7

S4,     \ Movement following a Beam 1 Break
  #Z^EndSession: ---> S1
  #Z^Start_CS:   ---> S2
  #Z^Three: ADD B(^ITIActivity), D(I+11) ---> S5
  #Z^Two:   ADD B(^ITIActivity), D(I+11) ---> S6
  #Z^One:   ADD B(^ITIActivity), D(I+11) ---> S7

S5,     \ Movement following a Beam 2 Break
  #Z^EndSession: ---> S1
  #Z^Start_CS:   ---> S2
  #Z^Four:  ADD B(^ITIActivity), D(I+11) ---> S4
  #Z^Two:   ADD B(^ITIActivity), D(I+11) ---> S6
  #Z^One:   ADD B(^ITIActivity), D(I+11) ---> S7

S6,     \ Movement following a Beam 3 Break
  #Z^EndSession: ---> S1
  #Z^Start_CS:   ---> S2
  #Z^Four:  ADD B(^ITIActivity), D(I+11) ---> S4
  #Z^Three: ADD B(^ITIActivity), D(I+11) ---> S5
  #Z^One:   ADD B(^ITIActivity), D(I+11) ---> S7

S7,     \ Movement following a Beam 4 Break
  #Z^EndSession: ---> S1
  #Z^Start_CS:   ---> S2
  #Z^Four:  ADD B(^ITIActivity), D(I+11) ---> S4
  #Z^Three: ADD B(^ITIActivity), D(I+11) ---> S5
  #Z^Two:   ADD B(^ITIActivity), D(I+11) ---> S6

\------------------ Animal on Right -------------------
S8,     \ Wait for first Beam Break
  #Z^EndSession: ---> S1
  #Z^Start_CS:   ---> S2  \ End of ITI
  #Z^Four:       ---> S9
  #Z^Three:      ---> S10
  #Z^Two:        ---> S11
  #Z^One:        ---> S12

S9,     \ Movement following a Beam 5 Break
  #Z^EndSession: ---> S1
  #Z^Start_CS:   ---> S2
  #Z^Six:   ADD B(^ITIActivity), D(I+11) ---> S10
  #Z^Seven: ADD B(^ITIActivity), D(I+11) ---> S11
  #Z^Eight: ADD B(^ITIActivity), D(I+11) ---> S12

S10,    \ Movement following a Beam 6 Break
  #Z^EndSession: ---> S1
  #Z^Start_CS:   ---> S2
  #Z^Five:  ADD B(^ITIActivity), D(I+11) ---> S9
  #Z^Seven: ADD B(^ITIActivity), D(I+11) ---> S11
  #Z^Eight: ADD B(^ITIActivity), D(I+11) ---> S12
```

```
S11,    \ Movement following a Beam 7 Break
  #Z^EndSession: ---> S1
  #Z^Start_CS:   ---> S2
  #Z^Five:  ADD B(^ITIActivity), D(I+11) ---> S9
  #Z^Six:   ADD B(^ITIActivity), D(I+11) ---> S10
  #Z^Eight: ADD B(^ITIActivity), D(I+11) ---> S12

S12,    \ Movement following a Beam 8 Break
  #Z^EndSession: ---> S1
  #Z^Start_CS:   ---> S2
  #Z^Five:  ADD B(^ITIActivity), D(I+11) ---> S9
  #Z^Six:   ADD B(^ITIActivity), D(I+11) ---> S10
  #Z^Seven: ADD B(^ITIActivity), D(I+11) ---> S11
```

DESCRIPTION OF STATE SETS 5 & 6

State Set 5 calculates the response latency variables and prints the information in the data file. State Set 6 prints the summary information to the MED-PC Run-Time Screen.

```
\*********************************************
\       RESPONSE LATENCY DETERMINATION
\*********************************************
S.S.5,
S1,
  #Z^Start_CS: ---> S2

S2,
  #Z^Avoid:     SET B(^AvoidLatency)   = B(^AvoidLatency)   + D(I+8), D(I+10) = 0;
                SET B(^EscapeAdjusted) = B(^EscapeAdjusted) + D(I+12) ---> S1
  #Z^Escape:    SET B(^EscapeLatency)  = B(^EscapeLatency)  + D(I+10), D(I+8) = 0;
                SET B(^EscapeAdjusted) = B(^EscapeAdjusted) + D(I+12) ---> S1
  #Z^Start_ITI: SET B(^EscapeAdjusted) = B(^EscapeAdjusted) + D(I+12);
                SET D(I+8) = 0, D(I+10) = 0 ---> S1  \ No Response
  0.01": SET D(I+8)  = D(I+8)  + 0.01;               \ Avoid Latency
         SET D(I+10) = D(I+10) + 0.01;               \ Escape Latency
         SET D(I+12) = D(I+12) + 0.01 ---> SX        \ Adjusted Escape Latency


\*********************************************
\        SHOW TRIAL # AND SUMMARY DATA
\*********************************************
S.S.6,
S1,
  #START: CLEAR 1,60 ---> S2

S2,
  1": SHOW 1,Trial #,B(^TrialCount), 2,Avoids,B(^TotalAvoids),  3,Escapes,B(^TotalEscapes);
      SHOW  4,Failures,B(^TotalFailures),6,ITI Activity,B(^ITI Activity), 9,Crossings,B(^TotalCrossings);
      SHOW 10,Adj Escape Lat,B(^EscapeAdjusted);
      SHOW 11,Left Movement,B(^LeftMovement),   12,Left Time,B(^LeftTime);
      SHOW 16,Right Movement,B(^RightMovement), 17,Right Time,B(^RightTime) ---> SX
  #Z^EndSession: ---> S1
```

DESCRIPTION OF STATE SETS 7 & 8

State Sets 7 and 8 print avoid and escape latency data to the Run-Time Screen.

```
\*********************************************
\           SHOW MEAN AVOID LATENCY
\*********************************************
S.S.7,
S1,
  #START: ---> S2

S2,
  1": IF B(^TotalAvoids) >= 1 [] ---> S3

S3,
  1": SHOW 7,Avoid Latency,B(^AvoidLatency)/B(^TotalAvoids) ---> SX
  #Z^EndSession: ---> S1
```

```
\*********************************************
\          SHOW MEAN ESCAPE LATENCY
\*********************************************
S.S.8,
S1,
  #START: ---> S2

S2,
  1": IF B(^TotalEscapes) >= 1 [] ---> S3

S3,
  1": SHOW 8,Escape Latency,B(^EscapeLatency)/B(^TotalEscapes) ---> SX
  #Z^EndSession: ---> S1
```

DESCRIPTION OF STATE SET 9

State Set 9 operates the tone stimulus. S3 waits for either the escape/avoid response or the end of the escape interval to turn off the stimuli.

```
\*********************************************
\                TONE CONTROL
\*********************************************
S.S.9,
S1,
  #START: ---> S2

S2,     \ Wait for Signal to turn Stimuli ON
  #Z^Start_CS: IF L = 1 [@Left, @Right]
                  @Left:  ON ^LeftTone  ---> S3
                  @Right: ON ^RightTone ---> S3

S3,     \ Wait for Signal to turn Stimuli OFF
  #Z^End_CS: OFF ^LeftTone, ^RightTone ---> S2
```

DESCRIPTION OF STATE SETS 10 & 11

State Sets 10 and 11 record left and right movement within the chamber, respectively.

```
\*********************************************
\           LEFT MOVEMENT ACTIVITY
\*********************************************
S.S.10,
S1,
  #START: ---> S2

S2,     \ Wait for Start of First Trial
  #Z^Start_A: ---> S3

S3,     \ Wait for first Beam Break
  #Z^Four:  ---> S4
  #Z^Three: ---> S5
  #Z^Two:   ---> S6
  #Z^One:   ---> S7

S4,     \ Movement following a Beam 1 Break
  #Z^EndSession:                         ---> S1
  #Z^Three: ADD B(^LeftMovement), D(I+1) ---> S5
  #Z^Two:   ADD B(^LeftMovement), D(I+1) ---> S6
  #Z^One:   ADD B(^LeftMovement), D(I+1) ---> S7

S5,     \ Movement following a Beam 2 Break
  #Z^EndSession:                         ---> S1
  #Z^Four:  ADD B(^LeftMovement), D(I+1) ---> S4
  #Z^Two:   ADD B(^LeftMovement), D(I+1) ---> S6
  #Z^One:   ADD B(^LeftMovement), D(I+1) ---> S7

S6,     \ Movement following a Beam 3 Break
  #Z^EndSession:                         ---> S1
  #Z^Four:  ADD B(^LeftMovement), D(I+1) ---> S4
  #Z^Three: ADD B(^LeftMovement), D(I+1) ---> S5
  #Z^One:   ADD B(^LeftMovement), D(I+1) ---> S7
```

```
S7,     \ Movement following a Beam 4 Break
  #Z^EndSession:                         ---> S1
  #Z^Four:  ADD B(^LeftMovement), D(I+1) ---> S4
  #Z^Three: ADD B(^LeftMovement), D(I+1) ---> S5
  #Z^Two:   ADD B(^LeftMovement), D(I+1) ---> S6


\********************************************
\           RIGHT MOVEMENT ACTIVITY
\********************************************
S.S.11,
S1,
  #START: ---> S2

S2,     \ Wait for Start of First Trial
  #Z^Start_A: ---> S3

S3,     \ Wait for first Beam Break
  #Z^Five:  ---> S4
  #Z^Six:   ---> S5
  #Z^Seven: ---> S6
  #Z^Eight: ---> S7

S4,     \ Movement following a Beam 5 Break
  #Z^EndSession:                          ---> S1
  #Z^Six:   ADD B(^RightMovement), D(I+2) ---> S5
  #Z^Seven: ADD B(^RightMovement), D(I+2) ---> S6
  #Z^Eight: ADD B(^RightMovement), D(I+2) ---> S7

S5,     \ Movement following a Beam 6 Break
  #Z^EndSession:                          ---> S1
  #Z^Five:  ADD B(^RightMovement), D(I+2) ---> S4
  #Z^Seven: ADD B(^RightMovement), D(I+2) ---> S6
  #Z^Eight: ADD B(^RightMovement), D(I+2) ---> S7

S6,     \ Movement following a Beam 7 Break
  #Z^EndSession:                          ---> S1
  #Z^Five:  ADD B(^RightMovement), D(I+2) ---> S4
  #Z^Six:   ADD B(^RightMovement), D(I+2) ---> S5
  #Z^Eight: ADD B(^RightMovement), D(I+2) ---> S7

S7,     \ Movement following a Beam 8 Break
  #Z^EndSession:                          ---> S1
  #Z^Five:  ADD B(^RightMovement), D(I+2) ---> S4
  #Z^Six:   ADD B(^RightMovement), D(I+2) ---> S5
  #Z^Seven: ADD B(^RightMovement), D(I+2) ---> S6
```

DESCRIPTION OF STATE SET 12

State Set 12 records the amount of time spent on each side of the chamber.

```
\********************************************
\           COUNT TIME ON EACH SIDE
\********************************************
S.S.12,
S1,     \ Verify Detection of Animal
  #START: ---> S2

S2,
  0.01": IF L = 1 [@Left, @Next]
            @Left: ---> S3
            @Next: IF L = 2 [@Right, @Wait]
                      @Right: ---> S4
                      @Wait:  ---> SX

S3,     \ Animal is on the Left
  0.01": SET B(^LeftTime) = B(^LeftTime) + 0.01;
         SET D(I+5)       = D(I+5)       + 0.01 ---> S3
  #Z^EndSession: ---> S1
  #Z^EnterRight: ---> S4

S4,     \ Animal is on the Right
  0.01": SET B(^RightTime) = B(^RightTime) + 0.01;
         SET D(I+6)        = D(I+6)        + 0.01 ---> S4
  #Z^EndSession: ---> S1
  #Z^EnterLeft:  ---> S3
```

DESCRIPTION OF STATE SETS 21-28

State Sets 21-28 record the IR inputs for each Beam exactly like Toggle Mode records inputs.

```
\*********************************************
\                BEAM 1 BREAK
\*********************************************
S.S.21,
S1,
  #START: ---> S2

S2,
  #R^LeftIR_4: Z^One ---> S3

S3,
  #R^LeftIR_4: ---> SX
  0.01":       ---> S2


\*********************************************
\                BEAM 2 BREAK
\*********************************************
S.S.22,
S1,
  #START: ---> S2

S2,
  #R^LeftIR_3: Z^Two ---> S3

S3,
  #R^LeftIR_3: ---> SX
  0.01":       ---> S2


\*********************************************
\                BEAM 3 BREAK
\*********************************************
S.S.23,
S1,
  #START: ---> S2

S2,
  #R^LeftIR_2: Z^Three ---> S3

S3,
  #R^LeftIR_2: ---> SX
  0.01":       ---> S2


\*********************************************
\                BEAM 4 BREAK
\*********************************************
S.S.24,
S1,
  #START: ---> S2

S2,
  #R^LeftIR_1: Z^Four ---> S3

S3,
  #R^LeftIR_1: ---> SX
  0.01":       ---> S2


\*********************************************
\                BEAM 5 BREAK
\*********************************************
S.S.25,
S1,
  #START: ---> S2

S2,
  #R^RightIR_1: Z^Five ---> S3

S3,
  #R^RightIR_1: ---> SX
  0.01":        ---> S2
```

```
\*********************************************
\                 BEAM 6 BREAK
\*********************************************
S.S.26,
S1,
  #START: ---> S2

S2,
  #R^RightIR_2: Z^Six ---> S3

S3,
  #R^RightIR_2: ---> SX
  0.01":        ---> S2


\*********************************************
\                 BEAM 7 BREAK
\*********************************************
S.S.27,

S1,
  #START: ---> S2

S2,
  #R^RightIR_3: Z^Seven ---> S3

S3,
  #R^RightIR_3: ---> SX
  0.01":        ---> S2


\*********************************************
\                 BEAM 8 BREAK
\*********************************************
S.S.28,
S1,
  #START: ---> S2

S2,
  #R^RightIR_4: Z^Eight ---> S3

S3,
  #R^RightIR_4: ---> SX
  0.01":        ---> S2
```

DESCRIPTION OF STATE SET 32

State Set 32 counts the time for the session.

```
\*********************************************
\              SESSION TEST TIMER
\*********************************************
S.S.32,
S1,
  #START: ---> S2

S2,      \ Wait for Start of First Trial
  #Z^Start_CS: ---> S3

S3,      \ Increment and Display time only.
         \ Session End is not triggered in S.S.1, S5
         \ until Current Trial is completed.
  1": ADD E; SHOW 5,Elapsed Time,E/60 ---> SX
  #Z^EndSession: ---> S1
```

CHAPTER 5

Understanding the Data Files

Data can be saved manually by selecting **FILE | SAVE DATA MANUALLY** or **FILE | SAVE DATA (FLUSH)**. The file name that is used to save the data in depends on the option that was chosen in the Hardware Configuration Utility and may also be dependent on the Subject, Experiment, and Group name provided in the MED-PC load wizard. Within each data file, the headings are created for each Subject, Experiment, Group, Box, etc., (see below). Data files may be opened with note pad, word pad, or any word processor or spreadsheet; however, be sure they are always saved “unformatted” in case a data extraction utility such as MED-PC to Excel might ever be used. Data file formats are explained in detail in the **MED-PC User’s Manual**.

CHAPTER 6

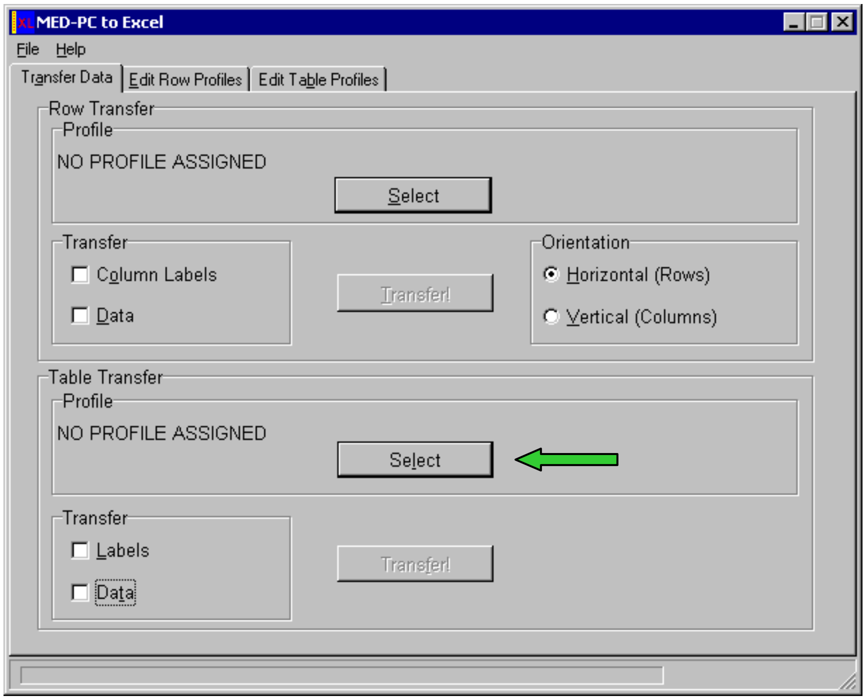
Data Analysis – Using MED-PC to Excel

Using a Pre-Formatted Table Profile (.MTP file)

MED-PC to Excel (MPC2XL) is a program that helps to import data from MED-PC (the raw-data file format, previous section) to a spreadsheet program such as Microsoft Excel. MPC2XL needs to be installed separately from MED-PC. Please refer to the "User's Manual for MPC2XL" for installation instructions**.** Once MPC2XL is installed, open the folder that the program was saved to, and the .MTP files have been saved in the C:\MED-PC IV\Data folder. Follow the step-by-step instructions below for importing data.

1. Open Microsoft Excel, and then minimize the window. Open **MED-PC to Excel** and the display shown in Figure 6.1 will appear. The uppermost file display should be titled **Transfer Data**. Under the **Table Transfer** window at the bottom of the screen, click on **Select**.

Figure 6.1 – Table Transfer



Row Transfer – Profile: NO PROFILE ASSIGNED

Transfer: □ Column Labels □ Data

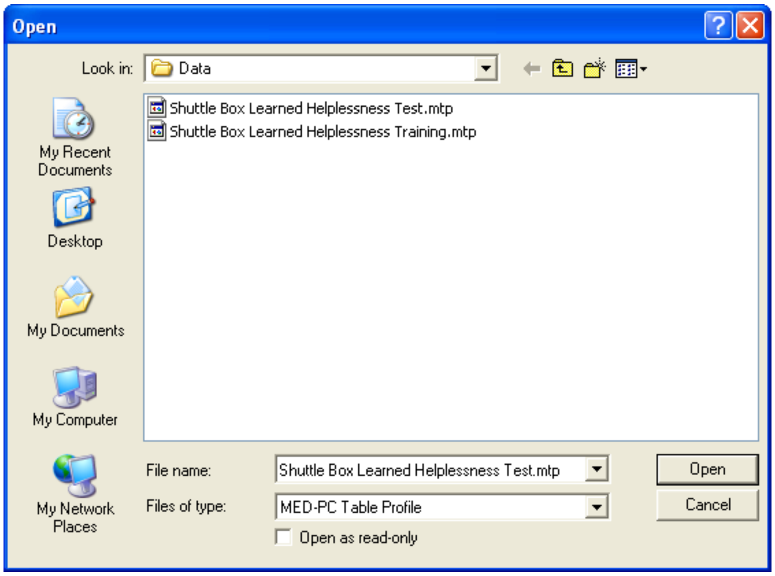
Orientation: ● Horizontal (Rows) ○ Vertical (Columns)

Table Transfer – Profile: NO PROFILE ASSIGNED

Transfer: □ Labels □ Data

2. Choose the .MTP file in the C:\MED-PC IV\Data folder that corresponds to the MED State Notation Procedure that was run and click **Open**. For this example Shuttle Box Learned Helplessness Test.mtp will be used.

Figure 6.2 - Select File to Open

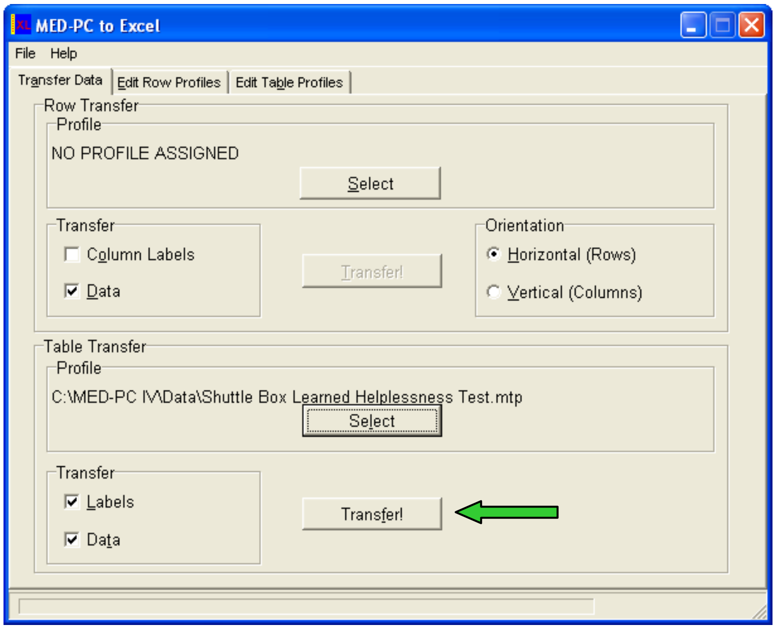


Shuttle Box Learned Helplessness Test.mtp
Shuttle Box Learned Helplessness Training.mtp

File name: Shuttle Box Learned Helplessness Test.mtp
Files of type: MED-PC Table Profile
□ Open as read-only

3. Note that .MTP file is listed under the Table Transfer "Profile." Select **Labels** and **Data**, because selecting these options will print data labels as well as import data. Click **Transfer!**

Figure 6.3 - Transfer Data



Row Transfer
Profile
NO PROFILE ASSIGNED

Transfer
□ Column Labels
☑ Data

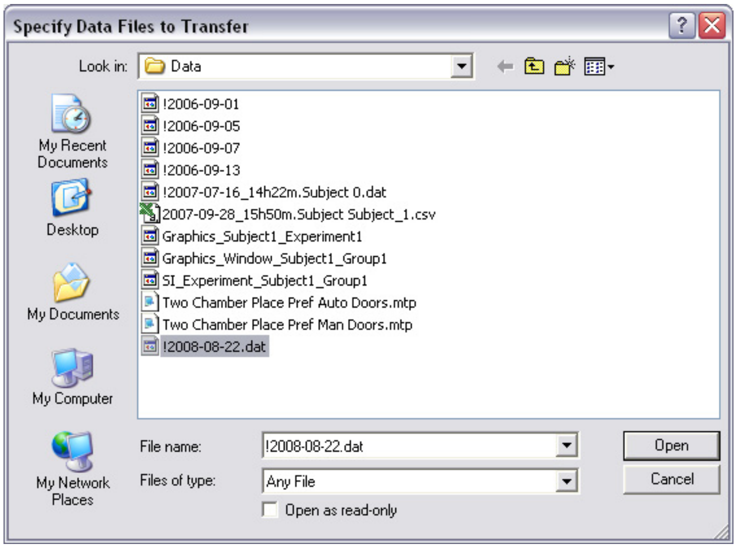
Orientation
● Horizontal (Rows)
○ Vertical (Columns)

Table Transfer
Profile
C:\MED-PC IV\Data\Shuttle Box Learned Helplessness Test.mtp

Transfer
☑ Labels
☑ Data

4. Specify the raw data file to transfer, and then click **Open**. This step performs the transfer, and now the data has been sent to Microsoft Excel.

Figure 6.4 - Specify Data Files to Transfer



Look in: Data

!2006-09-01
!2006-09-05
!2006-09-07
!2006-09-13
!2007-07-16_14h22m.Subject 0.dat
2007-09-28_15h50m.Subject Subject_1.csv
Graphics_Subject1_Experiment1
Graphics_Window_Subject1_Group1
SI_Experiment_Subject1_Group1
Two Chamber Place Pref Auto Doors.mtp
Two Chamber Place Pref Man Doors.mtp
!2008-08-22.dat

File name: !2008-08-22.dat

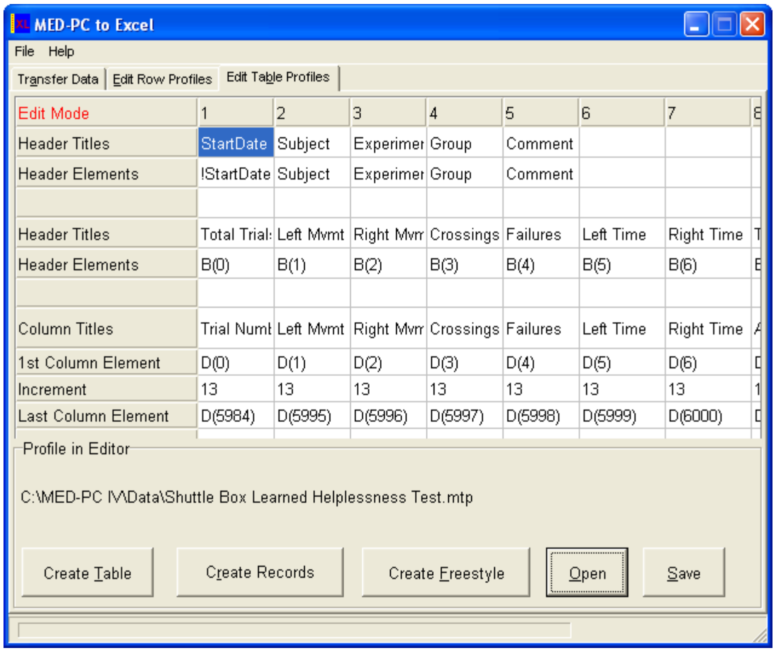
Files of type: Any File

☐ Open as read-only

Editing the .MTP file

The .MTP file can be edited to customize the transfer process and display the data of most interest. See the "User's Manual for MPC2XL" for explicit instructions about how to modify the MTP file using the Edit Table Profiles tab, shown in Figure 6.5. "Header Titles" are user defined, and can include any information that will help label the data listed below the title. "Header Elements" are the data points that will get transferred from the raw data file into Excel. The raw data file will list the elements that can be included in the .MTP file (e.g. A-Z).

Figure 6.5 - Edit Table Profiles



Edit Mode	1	2	3	4	5	6	7	8
Header Titles	StartDate	Subject	Experimer	Group	Comment			
Header Elements	!StartDate	Subject	Experimer	Group	Comment			
Header Titles	Total Trial:	Left Mvmt	Right Mvr	Crossings	Failures	Left Time	Right Time	T
Header Elements	B(0)	B(1)	B(2)	B(3)	B(4)	B(5)	B(6)	B
Column Titles	Trial Numt	Left Mvmt	Right Mvr	Crossings	Failures	Left Time	Right Time	A
1st Column Element	D(0)	D(1)	D(2)	D(3)	D(4)	D(5)	D(6)	D
Increment	13	13	13	13	13	13	13	1
Last Column Element	D(5984)	D(5995)	D(5996)	D(5997)	D(5998)	D(5999)	D(6000)	D

Profile in Editor

C:\MED-PC IV\Data\Shuttle Box Learned Helplessness Test.mtp

To edit either the Header Titles or Header Elements, click on the appropriate cell in the Edit Table Profiles window. Rows and columns can be added to the file. First, select the desired location, then right-click to add either the desired row or column. Use the right-click option titled **Paste an Identifier** to include subject or session identifying information. Note that when using the Paste an Identifier function, Header Titles and Header Elements are edited and pasted automatically.

To save the edited .MTP file, select **Save** and create a new filename in the Data subfolder. To use this newly edited and saved .MTP file, verify that the file is selected in the **Table Transfer Profile** display (Figure 6.3), and then click **Transfer.**