FIXED RATIO APPLICATION

MED-STATE NOTATION™ PROCEDURE

SOF-700RA-1 User's Manual
DOC-021
Rev. 1.2

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

TABLE OF CONTENTS

Chapter 1 1

Introduction 1

Chapter 2 2

Overview of the Procedure 2

Chapter 3 3

Translating FR Training.mpc 3

Chapter 4 4

Running FR Training.mpc 4

Viewing/Changing Variable Values 9

Macros 10

Chapter 5 11

Modifying the MedState Notation™ Code 11

MED State Notation Code 12

Chapter 6 16

Understanding the Raw Data File 16

Sample Data File 16

Breakdown of Sample File 17

Chapter 7 20

Using SoftCR™ Pro 20

Time Components 20

Control Code Components 20

Data Element Format 20

Example of SoftCR Pro 22

CHAPTER 1
Introduction

The latest version of MED-PC® IV, and applications such as the Fixed Ratio Application, make it easier than ever to run recognized experiment protocols without first developing a MedState Notation program. Of course, this approach does not preclude modifying the supplied code to meet changes in the research requirements. This manual covers running the standard protocol, then provides some examples of editing and modifying of the code, and finally it identifies in detail the elements in the raw data file produced by this application.

In addition to this manual, please refer to the **MED-PC IV User's Manual** for the installation of the MED-Associates interface drivers, the MED-PC IV Software, and the Delphi® Compiler, and for instructions on hardware configuration. Hardware Configuration is a software utility that comes with MED-PC and is used to assign the inputs and outputs in the interface cabinet to each logical box that will be run. Data file structure, file saving format, and other related options are also determined by this utility. Refer to the **MED-PC IV Programmer's Manual** if unfamiliar with the process of translating and compiling an application. Trans IV must be run on the file FR Training.mpc or any .mpc files supplied for SOF-700RA-1 before MED-PC can be launched for the first time. Please contact MED Associates if any of these materials are needed.

Before proceeding with this manual, print a copy of the .mpc file (FR Training.mpc). Those lines beginning with a backslash "\" indicate comments that help identify key elements of the code and explain the function of each program step.

CHAPTER 2
Overview of the Procedure

The test animal is required to respond on the specified lever on a fixed ratio schedule. The ratio may be user specified. The subject is rewarded when the ratio is met. A time out following the reward may be implemented. The session ends based on time. Correct and incorrect lever presses are counted and the percent correct and incorrect are calculated at the end of the session.

The correct lever value should be set to '1' for left or '2' for right. The default is set to left lever.

The reward value should be set to '1' for a pellet, '2' for a dipper presentation, or '3' for a drug infusion. The default is set to pellet.

The reward time specifies how long, in seconds, to activate the reward. Pellet dispensers only need a 0.05 second pulse to activate. A dipper presentation or drug infusion may be set to longer times as required.

If desired, a time out (in seconds) following the reward may be implemented. If it is not wanted, set the time to zero seconds. Lever presses made during the time out are not counted.

The default session time is 60 minutes. This value is set in minutes.

By default, the lever presses and rewards are recorded in the C-array for later viewing in Med Associates' Soft Cumulative Recorder (SoftCR Pro). Correct lever presses are recorded as steps, rewards as pips, and incorrect lever presses as event pens.

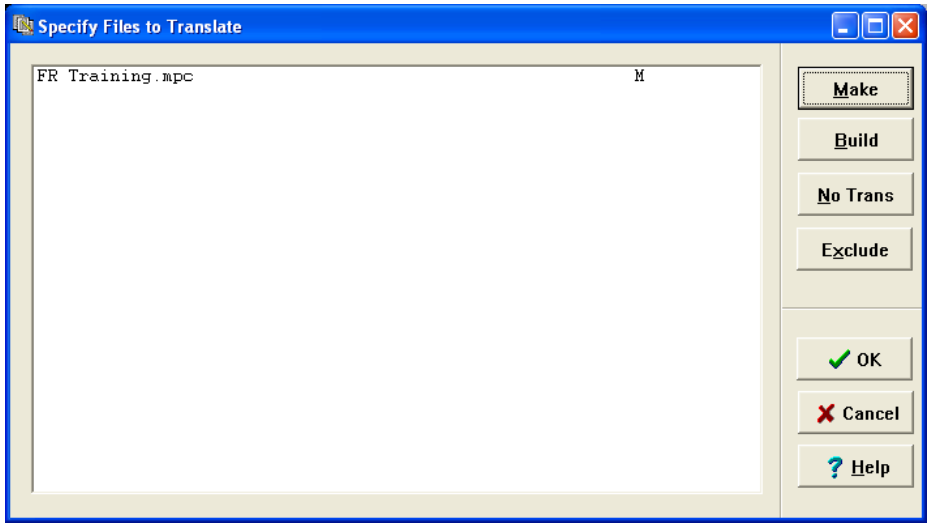
Modifications to the program will be discussed in greater detail later in this manual.

CHAPTER 3
Translating FR Training.mpc

Translating MedState Notation procedures is covered in detail in the **MED-PC IV Programmers Manual**. Be sure that a copy of the file FR Training.mpc, or whichever file is going to be translated, is present in the directory C:\MED-PC IV\MPC\. Open Trans IV and then select **Translation | Translate and Compile** to produce the screen shown in Figure 1.

Click OK to start the translator. It should automatically parse the MedState Notation and then drop to a DOS screen to compile the Pascal code. Depending on the speed of the computer, each of these steps may not be visible. If any problems are encountered during this process, refer to the on screen Help menu, the MED-PC Programmer's Manual, or contact MED Associates, Inc. for assistance.

Figure 1 - Trans IV Control Panel for Translating and Compiling MedState Notation Code



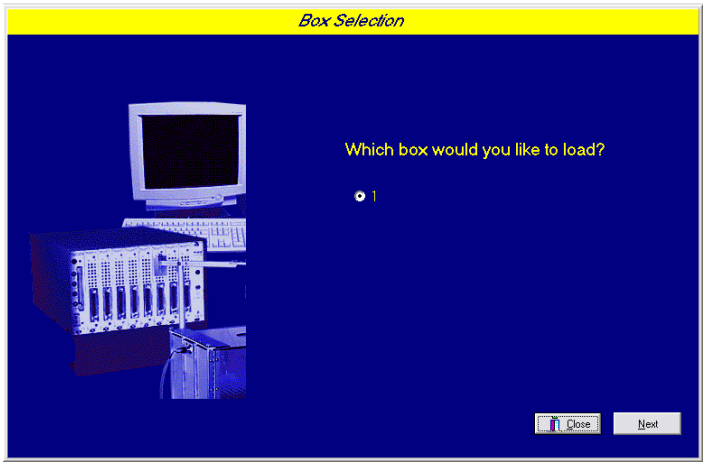
CHAPTER 4
Running FR Training.mpc

First time users may choose to use the load wizard, while experienced users may prefer to go directly to the run time screen to load the FR Training application. Clicking either the MED-PC IV icon from the **Start** | **Programs** list or the desktop shortcut brings up the MED-PC Experiment Loading Wizard Welcome screen shown in Figure 2. If it is not necessary to run Load Wizard, then deselect the box labeled "Run this expert automatically when starting MED-PC." Close this screen by clicking the **Close** button. Closing this screen immediately reveals the MED-PC Run Screen shown in Figure 10. Click **Next** to proceed with the wizard. The display shown in Figure 3 will appear.

Figure 2 - Loading Wizard Welcome Screen

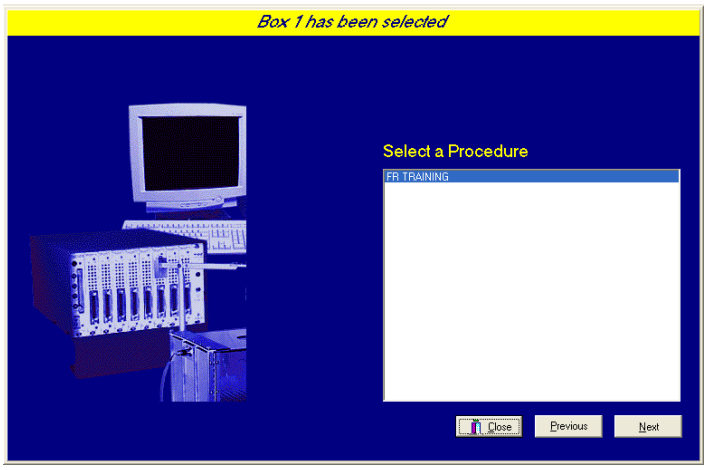
Select the boxes to load and click **Next**.

Figure 3 – Box Selection



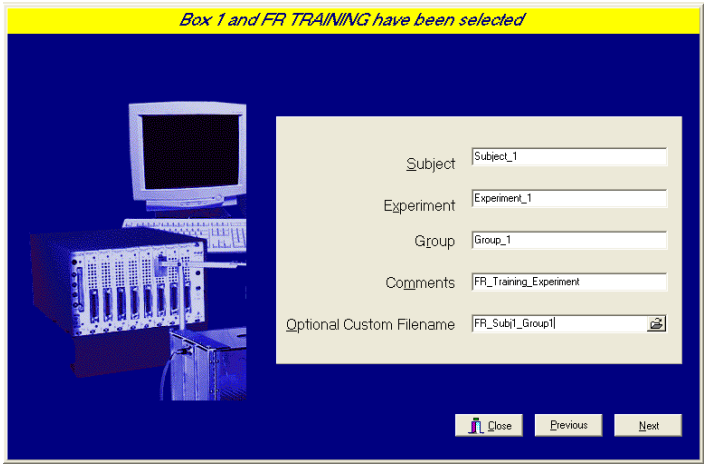
This is where the procedure to be run is selected. The screen displays a list of all the currently compiled procedures. Select the procedure to be run, and then click **Next**.

Figure 4 – Select a Procedure to Load



The Box/Procedure Selected Screen will display next, as shown in Figure 5. The purpose of this screen is to allow annotations to be added to the data file that is produced by MED-PC IV. These annotations will help identify the Subject, Experiment, and Experiment Group upon 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 5 – Box/Procedure Selected

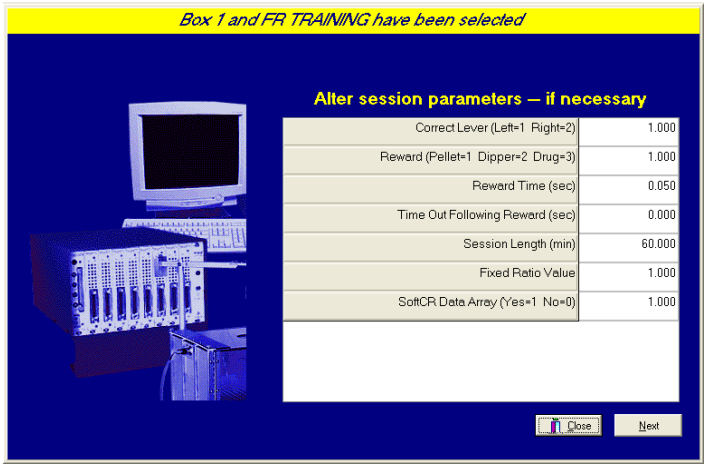


The next screen to appear is the Review Choices screen, as seen in Figure 6. This is a method of confirming that the information entered is correct. If it is not correct, select **Previous**, and edit the information. If it is correct, select **Next** to continue.

Figure 6 – Review Choices Screen

The Alter Session Parameters Screen, shown in Figure 7, is the next screen to appear, and is an important screen for the researcher. The Alter Session Parameters screen allows the researcher to alter the parameters by which a procedure executes. Any of the variables may be changed on this form. Simply highlight the value to change, and then enter the desired value. The Send Start Command Screen appears next.

Figure 7 – Alter Session Parameters Screen

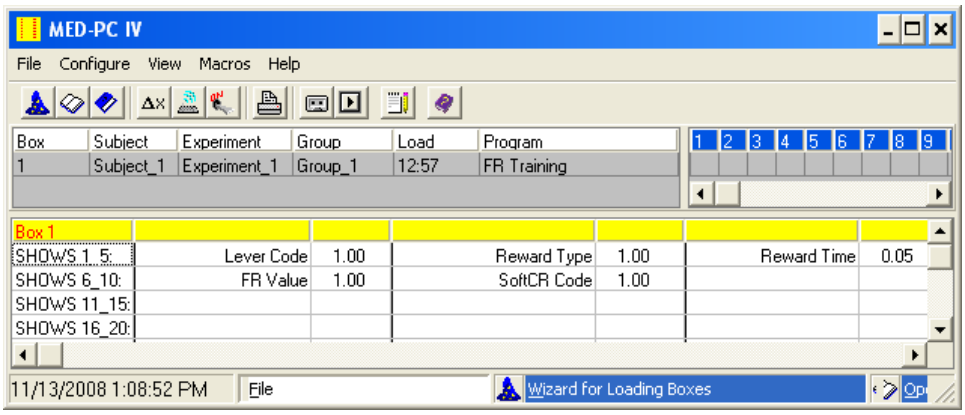


In this example only 1 box is described in the Hardware Configuration, so Figure 8 will appear next. If more than 1 box is in the Hardware Configuration, then Figure 9 will appear. In both cases, the screens are where the researcher decides to either load more boxes, send a start signal to boxes that are already loaded, or enter the MED-PC IV run-time environment without sending a start signal by selecting "I am finished with the wizard". This option results in the screen shown in Figure 10.

Figure 8 – Send Start Command Screen for Single Box Configuration

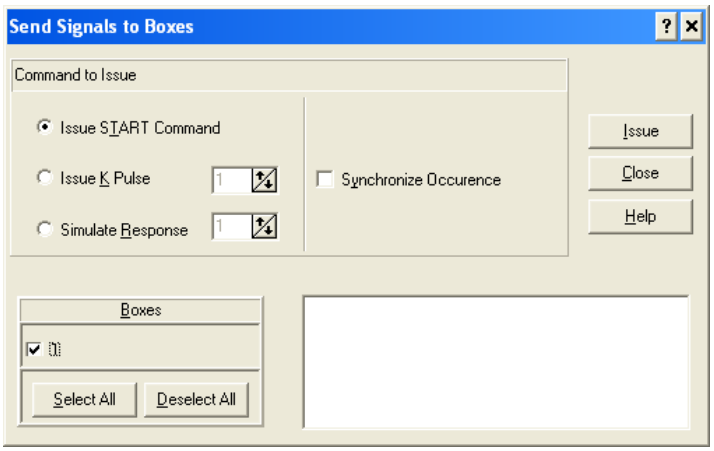
Figure 9 – Send Start Command Screen for Multiple Box Configuration

Figure 10 – MED-PC IV Runtime Screen



If all settings are correct, a **Start** command may be issued in one of two ways. One is to select C**onfigure | Signals**, and the other is to click the 5th tool-bar item that resembles a keyboard to reveal the window shown in Figure 11. To make additional changes to the variable values, proceed to the next section.

Figure 11 - Send Start Signals to Boxes

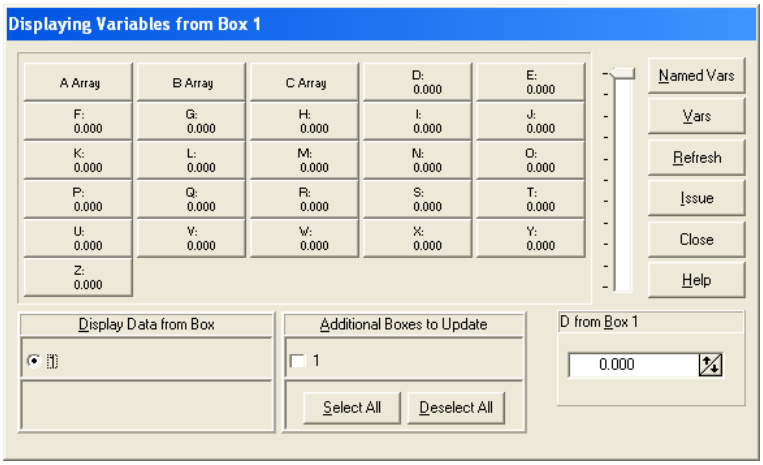


Click a single box, multiple boxes, or **Select All** followed by the **Issue** button to start the procedure. Click **Close** to remove the window from the screen. There is no reason to select **Synchronize Occurrence**.

Viewing/Changing Variable Values

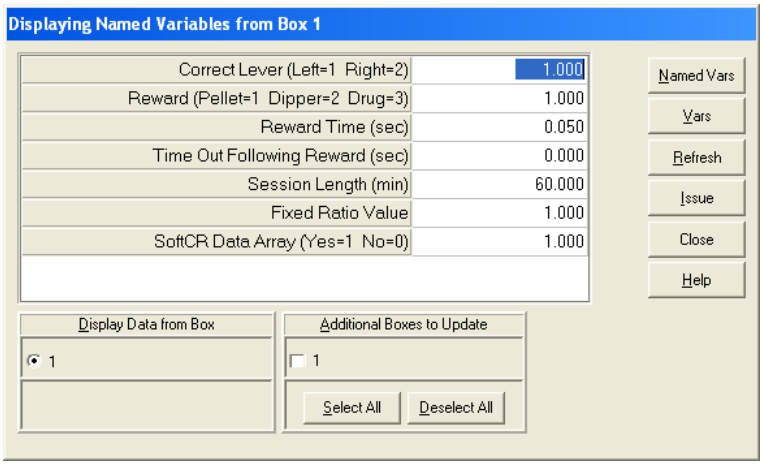
To change variables prior to issuing the **Start** command select **Configure** | **Change Variables**, or click the 4th tool bar item **Δ X**, to reveal the window shown in Figure 12. If a blank table is first presented, click the box number that is being used in the "Display Data from Box" area of the screen. By clicking additional boxes in the "Additional Boxes to Update" section, changes made to a single box are automatically loaded on all of the boxes selected.

Figure 12 - Change Variables Screen



The value of any simple variable or data element may be viewed from this screen. Click an array on the table and each element in that array can be viewed. 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. For the change to take effect, click **Issue**. Click **Named Variables** to produce the display shown in Figure 13. Edit these variables to best suit the experiment.

Figure 13 - Named Variables Display



Macros

To create a macro, commands will be entered manually and then the keyboard functions will be recorded. Once the commands are recorded as a macro, a number of macros can be created with the macro editor. The following example illustrates loading Box 1 and changing Session Time to 120 minutes.

Begin by opening MED-PC IV and then go 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 (cassette tape icon). A note on the bottom of the display indicates that the recorder is running. Proceed to load "FR Training.mpc." To do this, click **Files** | **Open Session**, or click the 2nd tool bar item (open book icon). 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, easily recognizable name. The example in Figure 14 is named "FR Training120min.mac" since it changes the session time to 120 minutes.

Figure 14 – FR Training120min.mac

```
 BOX  1 SUBJ 0 EXPT 0 GROUP 0 PROGRAM FR Training
"Session Time(Min)" VALUE  120.000 MAINBOX 1 BOXES 1
```

Once this macro is built, use the macro editor to make simple changes such as replacing reward or correct lever values. Review the Help file on screen or the **MED-PC IV Programmers 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.

CHAPTER 5
Modifying the MedState Notation™ Code

Permanent changes may also be made to the MedState-Notation code. To make the same change to the stimulus duration as shown above in the Macros section, complete the following steps:

1. Open Trans IV and select **File** | **Open** from the main menu to place "FR Training.mpc" into the text editor.
2. Scroll down to approximately line 93 (note the line counter in the lower right hand corner of the editor) to reveal the code shown in Figure 15.

Figure 15 – FR Training.mpc Lines 93-97

```
S.S.1,
S1,
  0.001": SET A(^CorrectLev) = 1, A(^Reward)      = 1,  A(^RewardTime) = 0.05;
          SET A(^TimeOut)    = 0, A(^SessionTime) = 60, A(^FRVal)      = 1;
             SET A(^SoftCR)     = 1 ---> S2
```

3. Change A(^SessionTime) = 60 to A(^SessionTime) = 120 and save the changes with the same or a new file name such as "FR Training_120min.mpc." Remember if a new .mpc file name is created and a macro is being used to load boxes, the file name in the macro also must be changed.
4. Translate and compile the new or changed file as described previously, and run MED-PC IV. Use the Change Variables screen to view/confirm the new values.

MED State Notation Code

The Med State Notation code for the FR Training program is shown below for reference purposes.

```
\ Copyright (C) 2008 MED Associates, All rights reserved.

\ FR_Training.mpc  Fixed Ratio Source Program for MED-PC Version IV


\ Constants Used in this Program
\ Compare the configurations below with the wiring chart supplied
\ with all system orders.  Edit Input and Output #'s if different
\ for your system or contact MED Associates for assistance.

\ Inputs
^LeftLever  = 1
^RightLever = 2

\ Outputs
^Pellet     = 3
^Dipper     = 3  \ If both Pellet and Dipper are ordered
                 \ It will be necessary to change one of these
^LeftLight  = 4
^RightLight = 5
^HouseLight = 7
^Pump       = 8


\ A() = Control Variables with Assigned Aliases as Defined
Var_Alias Correct Lever (Left=1  Right=2)     = A(0)  \ Default = Left Lever
Var_Alias Reward (Pellet=1  Dipper=2  Drug=3) = A(1)  \ Default = Pellet
Var_Alias Reward Time (sec)                   = A(2)  \ Default = 0.05 seconds
Var_Alias Time Out Following Reward (sec)     = A(3)  \ Default = 0 seconds
Var_Alias Session Length (min)                = A(4)  \ Default = 60 minutes
Var_Alias Fixed Ratio Value                   = A(5)  \ Default = 1
Var_Alias SoftCR Data Array (Yes=1  No=0)     = A(6)  \ Default = Yes

^CorrectLev   = 0
^Reward       = 1
^RewardTime   = 2
^TimeOut      = 3
^SessionTime  = 4
^FRVal        = 5
^SoftCR       = 6


\ List Data Variables Here
\  B() = Response Counts
\  B(0) = Total Responses
\  B(1) = Total Correct Response Count
\  B(2) = Total Incorrect Response Count

\  C() = IRT Array and Event Pen Code for Soft Cum. Recorder

\  D = % Correct
\  E = % Incorrect
\  F = Total Rewards


\ List Working Variables Here
\  I = Subscript for the IRT Array C
\  L = Ratio Count
\  S = Elapsed Time in Session
\  T = Elapsed Time in 0.01 sec Increments for SoftCR Data


DIM A = 6
DIM B = 2

^CArraySize = 10000  \ Change this Variable also when modifying the C array size.
DIM C = 10000  \ Dimension Array C for 10001 data points.
               \ Under MED-PC Version IV, this array may be
               \ enlarged up to 1 million elements; however,
               \ 10,000 was felt to be adequate for this application.
               \ An end of array seal -987.987 will limit the saved
               \ file to only those elements used during the running
               \ of the procedure.
```

```
\ Z-Pulses Used in this Program
^Z_Pellet = 1   \ Signal Pellet Reinforcement
^Z_Dipper = 2   \ Signal Dipper Reinforcement
^Z_Pump   = 3   \ Signal Pump Reinforcement
^Z_End    = 32  \ Signal End of Session


\************************************************
\                Fixed Ratio Schedule
\  S1 - Set Default Values
\       Correct Lever             (1-Left)
\       Reward Device             (1-Pellet)
\       Reward Time               (0.05 seconds)
\       Time Out Following Reward (0 seconds)
\       Session Length            (60 minutes)
\       Fixed Ratio Value         (1)
\       SoftCR Activation         (1-Yes)
\************************************************
S.S.1,
S1,
  0.001": SET A(^CorrectLev) = 1, A(^Reward)      = 1,  A(^RewardTime) = 0.05;
          SET A(^TimeOut)    = 0, A(^SessionTime) = 60, A(^FRVal)      = 1;
          SET A(^SoftCR)     = 1 ---> S2

S2,     \ First Statement: Wait for START signal, turn HouseLight and
        \ associated stimulus ON.
        \
        \ Second Statement: Update screen display with default values
        \ for Control Variables.  This will show any changes made via
        \ the "Configure | Change Variables" Window prior to START.
  #START: CLEAR 1,60;
          SET A(^RewardTime) = A(^RewardTime) * 1";
          SET A(^TimeOut)    = A(^TimeOut)    * 1";
          ON ^HouseLight, ^LeftLight, ^RightLight ---> S3
  1": SHOW 1,Lever Code,A(^CorrectLev), 2,Reward Type,A(^Reward),         3,Reward
Time,A(^RewardTime);
      SHOW 4,Time Out,A(^TimeOut),      5,Session Length,A(^SessionTime), 6,FR
Value,A(^FRVal);
      SHOW 7,SoftCR Code,A(^SoftCR) ---> SX

S3,     \ Test for Correct Lever
  #RA(^CorrectLev): ADD L, B(0), B(1);
                    IF L >= A(^FRVal) [@FR_Value_Met, @False]
                       @FR_Met: ADD F; SET L = 0; ZA(^Reward) ---> S4
                       @False: ---> SX
  #R^LeftLever:  ADD B(0), B(2) ---> SX
  #R^RightLever: ADD B(0), B(2) ---> SX
  #Z^Z_End: ---> S5

S4,     \ Time Out Interval Following Reward
  A(^TimeOut)#T: ---> S3
  #Z^Z_End: ---> S5

S5,     \ End of Session - Turn Lights Off
        \ Calculate % Correct and % Incorrect
  0.01": OFF ^HouseLight, ^LeftLight, ^RightLight;
         IF B(0) = 0 [@NoCalc, @Calculate]
            @NoCalc: ---> S6
            @Calculate: SET D = B(1) / B(0) * 100;
                        SET E = B(2) / B(0) * 100;
                        SHOW 7,% Correct,D, 8,% Incorrect,E ---> S6
S6,     \ Holding State at End of Session
  1': ---> SX


\*****************************************
\           REWARD CONTROL TIMER
\*****************************************
S.S.2,
S1,
  #Z^Z_Pellet: ON ^Pellet ---> S2
  #Z^Z_Dipper: ON ^Dipper ---> S2
  #Z^Z_Pump:   ON ^Pump   ---> S2

S2,     \ Time Reward Device for A(^RewardTime) seconds
        \ Default value set in S.S.1, S1 & S2
  A(^RewardTime)#T: OFF ^Pellet, ^Dipper, ^Pump ---> S1
```

```
\*********************************************
\               UPDATE DISPLAY
\*********************************************
S.S.3,
S1,
  #START: SHOW 2,Correct Rsp,B(1), 3,Incor Rsp,B(2), 4,Reward,F ---> S2

S2,
  1": SHOW 2,Correct Rsp,B(1), 3,Incor Rsp,B(2), 4,Reward,F ---> S2


\*********************************************
\   INCREMENT TIME T FOR IRT AND EVENT ARRAY
\*********************************************
S.S.4,
S1,
  #START: IF A(^SoftCR) > 0 [] ---> S2

S2,
  0.01": ADD T ---> SX
  #Z^Z_End: ---> S1


\*****************************************************
\ ADD A STEP ELEMENT TO ARRAY C FOR EACH CORRECT RESPONSE
\*****************************************************
S.S.5,
S1,
  #START: IF A(^SoftCR) > 0 [@ActivateSoftCR, @NoSoftCR]
             @ActSoftCR: IF A(^CorrectLev) = 1 [@LeftLeverCorrect, @RightLeverCorrect]
                            @LeftCorrect:  ---> S2
                            @RightCorrect: ---> S3
             @NoSoftCR: ---> SX

S2,     \ Left Lever Correct
  #R^LeftLever: SET C(I) = T + 0.10, T = 0; ADD I;
                IF I >= ^CArraySize [@True, @False]
                   @True: ---> S1
                   @False: SET C(I) = -987.987 ---> SX
  #Z^Z_End: ---> S1

S3,     \ Right Lever Correct
  #R^RightLever: SET C(I) = T + 0.10, T = 0; ADD I;
                 IF I >= ^CArraySize [@True, @False]
                    @True: ---> S1
                    @False: SET C(I) = -987.987 ---> SX
  #Z^Z_End: ---> S1


\*************************************************
\    ADD A PIP ELEMENT TO ARRAY C FOR EACH REWARD
\*************************************************
S.S.6,
S1,
  #START: IF A(^SoftCR) > 0 [] ---> S2

S2,
  #Z^Z_Pellet ! #Z^Z_Dipper ! #Z^Z_Pump: SET C(I) = T + 0.20, T = 0; ADD I;
                                         IF I >= ^CArraySize [@True, @False]
                                            @True: ---> S1
                                            @False: SET C(I) = -987.987 ---> SX


\***********************************************************
\ ADD AN EVENT ELEMENT TO ARRAY C FOR EACH INCORRECT RESPONSE
\      ADDITIONAL EVENT ELEMENTS MAY BE ADDED AS NEEDED
\          FOR EXAMPLE TO MARK TIME OUT INTERVALS.
\***********************************************************
S.S.7,
S1,
  #START: IF A(^SoftCR) > 0 [@ActivateSoftCR, @NoSoftCR]
             @ActSoftCR: SET C(I) = 0.50, I = I + 1, C(I) = -987.987;
                         IF      A(^CorrectLev)      =      2      [@LeftLeverIncorrect,
@RightLeverIncorrect]
                            @LeftIncorrect:  ---> S2
                            @RightIncorrect: ---> S3
             @NoSoftCR: SET C(0) = -987.987 ---> SX  \ Seal Array
```

```
S2,     \ Left Lever Incorrect
  #R^LeftLever: SET C(I) = T + 0.60, C(I+1) = 0.50, T = 0, I = I + 2;
                IF I >= ^CArraySize [@True, @False]
                   @True: ---> S1
                   @False: SET C(I) = -987.987 ---> SX
  #Z^Z_End: ---> S1

S3,     \ Right Lever Incorrect
  #R^RightLever: SET C(I) = T + 0.60, C(I+1) = 0.50, T = 0, I = I + 2;
                 IF I >= ^CArraySize [@True, @False]
                    @True: ---> S1
                    @False: SET C(I) = -987.987 ---> SX
  #Z^Z_End: ---> S1


\*****************************************
\                SESSION CLOCK
\*****************************************
S.S.8,
S1,
  #START: SHOW 1,Session,S/60 ---> S2

S2,
  1": ADD S; SHOW 1,Session,S/60;
      IF S/60 >= A(^SessionTime) [@EndSession, @ContinueTiming]
         @End: IF I < ^CArraySize [@True, @False]
                  @True: SET C(I) = T + 0.30, I = I + 1, C(I) = -987.987; Z^Z_End ---> S3
                  @False: Z^Z_End ---> S3
         @Cont: ---> SX

S3,     \ Wait for Screen Update and end with
        \ STOPABORTFLUSH for Automatic Data Saving
  2": ---> STOPABORTFLUSH
```

CHAPTER 6
Understanding the Raw Data File

Selecting “Annotated” on the file options page during hardware installation produces a raw data file similar to the following. Data files are located in C:\MED-PC IV\Data\ unless an alternate path was defined during hardware installation. They may be opened with note pad, word pad, or any word processor, however make sure they are always saved unformatted in case a data extraction utility, such as MPC2XL, is ever used. The header information should be self-explanatory. Data File Formats are explained in detail in the **MED-PC IV User’s Manual**.

Sample Data File

```
File: C:\MED-PC IV\DATA\FRTraining_Subject1_Group1


Start Date: 11/19/08
End Date: 11/19/08
Subject: Subject_1
Experiment: Experiment_1
Group: Group_1
Box: 1
Start Time: 11:54:17
End Time: 11:55:17
MSN: FR Training
D:       71.429
E:       28.571
F:       25.000
G:        0.000
H:        0.000
I:       72.000
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:      300.000
T:     1203.000
U:        0.000
V:        0.000
W:        0.000
X:        0.000
Y:        0.000
Z:        0.000
A:
     0:        1.000        1.000        5.000        0.000       60.000
     5:        1.000        1.000
B:
     0:       35.000       25.000       10.000
C:
     0:        0.500      759.100        0.200      656.100        0.200
     5:     1581.600        0.500      735.600        0.500      753.100
    10:        0.200     1021.100        0.200      669.600        0.500
```

Breakdown of Sample File

Working Variables

The following simple variables are shown immediately following the header information:

```
D:       71.429    - Percent Correct Lever Presses
E:       28.571    - Percent Incorrect Lever Presses
F:       25.000    - Total Rewards
G:        0.000    - Not Used
H:        0.000    - Not Used
I:       72.000    - Subscript for the IRT Array C (indicates number of elements in C array)
J:        0.000    - Not Used
K:        0.000    - Not Used
L:        0.000    - Ratio Count
M:        0.000    - Not Used
N:        0.000    - Not Used
O:        0.000    - Not Used
P:        0.000    - Not Used
Q:        0.000    - Not Used
R:        0.000    - Not Used
S:      300.000    - Elapsed Time in Seconds
T:     1203.000    - Elapsed Time in 0.01 sec Increments for SoftCR Data
U:        0.000    - Not Used
V:        0.000    - Not Used
W:        0.000    - Not Used
X:        0.000    - Not Used
Y:        0.000    - Not Used
Z:        0.000    - Not Used
```

Arrays

The arrays are presented in rows of 5 elements each. The first row begins with element 0 and ends with element 4; the second row begins with element 5 and ends with element 9, etc. Each row begins with an element marker followed by a colon, 0:, 5:,10:, 15:, 20:, 25:, 30:, etc. Size of Array C was set to 10000 with the DIM command, which means that Array C was defined for 10001 data points. Under MED-PC Version IV, this array may be enlarged up to 1 million elements; however, 10,000 was felt to be adequate for this application. An end of array seal -987.987 will limit the saved file to only those elements used during the running of the procedure.

Each array has been defined in the comments section at the beginning of the .mpc file. That information was used to add the following labels to each array. The lines were expanded to make room for the labels. See MPC2XL for producing labeled data files with all or just some of the information below.

Array A - Raw Data

Array A contains the control variable values. The values are preset to default values in State Set 1, State 1. They are displayed as named variables and can be changed by any of the methods described previously. Once a session is started these values should not be changed to insure the integrity of the experiment.

```
A:
     0:        1.000        1.000        5.000        0.000       60.000
     5:        1.000        1.000
```

Array A with Labels

	A(0)	A(1)	A(2)	A(3)	A(4)
Row Marker	Correct Lever	Reward	Reward Time in MED Ticks	Time Out	Session Time
0:	1	1	5	0	60

	A(5)	A(6)
Row Marker	Fixed Ratio Value	SoftCR Data Array
5:	1	1

Array B - Raw Data

Array B is used for the working variables that may change as the session runs. Elements for Correct Responses, Incorrect Response and Total Responses will be incremented according to animal's responses as session runs.

```
B:
     0:       35.000       25.000       10.000
```

Array B with Labels

Row Marker	B(0)	B(1)	B(2)
	Total Responses	Correct Responses	Incorrect Reponses
0:	35	25	10

Array C – SoftCR Pro Data

Array C contains summary information for SoftCR Pro. The value of every variable conforms to the Data Element Format described. Please refer to the **SoftCR Pro User's Manual** for more information.

If the fixed ratio is left at 1 (default value), then reinforcement takes place every time a response on the correct lever occurs. Response time values are represented in SoftCR ticks. To calculate when a response has occurred in real time, multiply the number by 0.01 sec since SoftCR's resolution is 10ms (0.01 sec). For example, value in C (1) is 759 ticks. 759 multiplied by 0.01 sec yields 7.59 seconds. This is when the first response has occurred in real time after program has started.

```
C:
     0:        0.500      759.100        0.200      656.100        0.200
     5:     1581.600        0.500      735.600        0.500      753.100
    10:        0.200     1021.100        0.200      669.600        0.500
```

Array C with Labels

	C(0)	C(1)	C(2)	C(3)	C(4)
Row Marker	SoftCR is Activated	Correct Response	Reinforcement	Correct Response	Reinforcement
0:	0.500	759.100	0.200	656.100	0.200
	C(5)	**C(6)**	**C(7)**	**C(8)**	**C(9)**
Row Marker	Incorrect Response	Set Event Pen to Level 0	Incorrect Response	Set Event Pen to Level 0	Correct Response
5:	1581.600	0.500	735.600	0.500	753.100
	C(10)	**C(11)**	**C(12)**	**C(13)**	**C(14)**
Row Marker	Reinforcement	Correct Response	Reinforcement	Incorrect Response	Set Event Pen to Level 0
10:	0.200	1021.100	0.200	669.600	0.500

CHAPTER 7
Using SoftCR™ Pro

The Fixed Ratio procedure stores IRT data in array C that may be used by the SoftCR™ Pro Cumulative Recorder program to generate a graphical record on screen or to print a Cumulative Record after the data has been collected. Each element in this "Cumulative Record" data array consists of two components. The integer value or value to the left of the decimal point is the time component. The decimal value or value to the right of the decimal point is a SoftCR Pro code component. Refer to the **SoftCR Pro User's Manual** for further information.

Time Components

Time components may be Relative or Absolute. In a Relative file, the time component of a given data element is equal to the elapsed time since the previous element. In an Absolute file, it is equal to the elapsed time since the beginning of the Record. The elements used in Fixed Ratio are all Relative with a resolution of 0.01 seconds (10 milliseconds).

Control Code Components

The control code component indicates whether the element is a Response Step, Reinforcement Pip, Pen Reset, Event Pen Up, or Event Pen Down element. Up to 10 Event Pens (0 - 9) and 10 Trace Pens (0 - 9) may be specified in MED-PC. In addition, Resets may be made with the Pen "Up" (no vertical tracing) or "Down." Additional coded information may be possible in the future.

Data Element Format

Each data element must conform to the following format:

nnnnn.xyz

Where:

nnnnn = Time Component

xyz = Control Code Component

x = Datum Type

y = Datum Index

z = Don't Care Digit

NOTE: If there are more than three numbers past the decimal point, then the array will be considered invalid and will be ignored by SoftCR Pro.

Allowed datum types and indices are:

nnnnn.1yz = Response or Step
y = Index (0 - 9) for Multi-Trace Files
"NULL" for Single Trace Files

nnnnn.2yz = Reinforcement or Pip
y = Index (0 - 9) for Multi-Trace Files
"NULL" for Single Trace Files

nnnnn.3yz = Rest Pen
y=Pen Mode, where
0=Pen Up During Reset
1=Pen Down During Reset

nnnnn.4yz = Unused

nnnnn.5yz = Set Event Pen To Level 0 ("Down" or at baseline), where
y = Event Pen Number (0 - 9)

nnnnn.6yz = Set Event Pen To Level 1 ("Up"), where
y = Event Pen Number (0 - 9)

nnnnn.7yz = Unused

nnnnn.8yz = Unused

nnnnn.9yz = Unused

Data in the array being plotted that does not conform to the above rules will be ignored.

Example of SoftCR Pro

Figure 16 is an example of what Fixed Ratio will look like in SoftCR Pro after an experiment is run. The red line records a Step for every correct response that is made. Since this is a Fixed Ratio of 1 schedule the subject was reinforced after one response on the Left Lever. A reward is recorded as a Pip, or a hash mark on the red line.

The colored traces at the bottom of the screen are called Event Pens and can be used in a variety of ways for a number of purposes. The Event Pens can go up and down, creating quick spikes like the ones seen here, or long plateaus that represent certain periods of time during an experiment. Here, the Event Pen 0 records a Spike for every incorrect response that is made.

Figure 16 - Example of a Drug Self Infusion Schedule Shown in SoftCR Pro