# K-LIMBIC

## DEVELOPER'S GUIDE

# K-Limbic
## Developers Guide

A Versatile System For The Control Of
Behavioural Experiments

Conclusive Marketing Ltd,
The Cottage,
High Wych Lane,
Sawbridgeworth,
CM21 0JR,
United Kingdom.

Tel. +44 (0) 1279 724226
Fax. +44 (0) 1279 726177

For product support contact Conclusive Marketing Ltd.

# Contents

Section

1

# Introduction

**'K-Limbic'** is a PC based system for the control of animal behavioural experiments.  Designed in collaboration with several international pharmaceutical companies K-Limbic has developed into a versatile, flexible and powerful tool for performing behavioural experiments.

The major component in the K-Limbic system is the integrated development environment, (IDE).  The IDE provides all the tools required to develop, edit and run your operant experiments.  K-Limbic provides a protocol manager to generate, edit and share experimental protocols, a database manager to create databases with subject details and a run manager to run experiments.

Behavioural experiments use a variety of techniques for selecting experimental parameters such as stimuli, time delays and reinforcement regimes, typical examples of such techniques are fixed ratios, variable intervals and matched, (and non-matched), to position delay values.  Historically, the application of these parameters to a specific experimental protocol has required a custom-written program for each particular experiment performed.  **K-Limbic provides a simple intuitive method of controlling behavioural experiments**, thus enabling the user to create individual experiments and please remember all this is done **without the requirement for a programmer**.

The essential themes of the K-Limbic design are ease of use, versatility and reliability.  The K-Limbic system is in daily use in many global pharmaceutical companies, testament to the quality of its development.  The flexible design means that K-Limbic is just as suited to a small independent laboratory or university department as it is to major corporations.

The K-Limbic design focuses on the major tasks that are presented to any successful behavioural or cognition laboratory.  Experimental task development is made simple by the use of the powerful protocol definition methodology, animal training and maintenance is easy with the use of the subject database, and most importantly K-Limbic is the first, (the only?) operant control system to automatically connect with data analysis macros to archive meaningful datasets on a per-animal or per group basis.

This document describes how to develop and edit a K-Limbic protocol.  Protocols determine the generic behaviour of an experiment and specify how the protocol will connect with the subject database to produce a specific behaviour tailored to the subject performing the test.  A tutorial section describes in detail the process for generating an example fixed ratio protocol.

Additionally, the tutorial section runs through the process of creating a test suite that uses the protocol so that we can test the protocol in a "live" situation.  The process for creating the test suite is the same as discussed in the K-Limbic Operators Manual.

The reference section lists the components available to build protocols, (stage types and auto control types).

The K-Limbic repository contains a number of protocols available for use or as starting points for developing new protocols.  Each protocol in the repository includes a description document that details all the settings required to generate the protocol from scratch.

<div align="right">

Section

2

# Software Overview

</div>

## 2.1 K-Limbic Terms
### Resources, Protocols & Subject Databases

As with all software applications, K-Limbic has its own vocabulary, (that is a collection of terms that are used in a specific manner). This section provides a definition of the commonly used terms where their meaning is specific within K-Limbic.

### Resources
A resource is any item of stimulus or response equipment placed within the animal chamber and connected to K-Limbic. Typical examples are lights, levers, position sensors etc. Experimental environments, (e.g. animal chambers, mazes etc), are defined within K-Limbic as a list of resources.

### Protocols
An individual experiment is described in K-Limbic as a Protocol. The protocol contains the information necessary to define an experiment from start to finish.

### Subject Databases
Subject databases are lists of subjects, (animals). The operator assigns a subject to a box during the process of starting an experiment. Each subject contains a unique set of parameters that have been predefined by the operator, the values of the various parameters are used by K-Limbic to customise a protocol for an individual animal. For example during training for any complex task novice animals will commence a task using a set of parameters suitable for a novice, as animals progress the task parameters increase in difficulty, using a subject database provides a simple method of customising any task to the current training level of a subject.

### Sessions
A K-Limbic experiment is run as a session. The session file defines how many boxes will run, which subject database will be used, where the data is stored and how the data is analysed.

### Test Suites
A test suite is a wrapper for sessions and databases. An individual test suite can contain up to 20 different session definitions and 20 different databases. Therefore a single test suite file can contain everything you will need to define a complete study, including training sessions, test sessions and maintenance sessions. K-Limbic defines two forms of test suite, "Development Test Suites" and "Installed Test Suites"; these are discussed in more detail later.

## 2.2 Starting K-Limbic

The K-Limbic installation program creates a K-Limbic item on the Start menu…

All Programs >> K-Limbic >> K-Limbic System

Select this item to start the K-Limbic application.  K-Limbic will present the log in window, select the appropriate user name from the list and enter the appropriate password.

## 2.3 The Main Screen

Enter K-Limbic by selecting the K-Limbic item from the Windows Start Menu and log in.
The software is loaded and presents the main screen to the user.  The main screen components are
listed below: -

1. **The main menu**
   The main menu appears below the title bar, the main menu gives the user access to the various
   functions of K-Limbic.
2. **The toolbar**
   The toolbar gives shortcut access to the main management windows of K-Limbic
3. **The user details section**
   The user details section gives an outline of the protocols and test suites that the current user
   has defined.
4. **The Test Suite section**
   The Test Suite section lists the primary database for the current test suite, (if one is loaded);
   this area also lists the recent activities performed with the test suite.
5. **The System Log**
   The system log is a list of recent activity that has occurred using K-Limbic.  The system log
   holds data for up to ten days.

## 2.4 Resources : Overview

A 'Resource' is a single item of operant equipment; a list of the resources which are currently supported by K-Limbic is given in table 2.1. K-Limbic defines an operant enclosure as of a collection of 'resources', i.e. a group of associated levers, lights, doors etc. K-Limbic controls the state of an operant enclosure by controlling the state of each resource in the resources list of the protocol.

**Table 2.1 Available K-Limbic Resources**

Retractable Levers
Cue Lights
Lights
Door Control
Tone Generator
Pellet Dispenser
Fluid Dispenser
Auxiliary Dispenser
Shock Trigger
Fan
Feeder Control (Retractable water bottle)
Lever Press Sensor
Sensor
Nose Poke Sensor
Position Sensor

The resources present in an operant enclosure will vary depending on the nature of the experiment it is used for. A K-Limbic developer configures a collection of resources as a part of the protocol description. The state of each configured resource is controlled at every stage of the protocol thus controlling the state of the operant enclosure.

## 2.5 Protocols : Overview

K-Limbic is a tool for performing user-defined experiments.  The method provided for the user to define the actions of an experiment is the protocol.

A K-Limbic protocol is a series of stages, at each stage the current state of an operant enclosure is fully defined, i.e. the state of any resources present in the enclosure is specified.  The developer defines the experiment as a set of distinct stages, (this is most easily represented as a graphical flowchart). Movement between stages is triggered by a trigger event, examples of trigger events are timeouts, lever presses and nose pokes.

The protocol description is defined using a number of "components".
These are as follows:

- The Header component, (the Reference, Title & Description of the protocol)
- The Resources component, (the items required in the operant chamber)
- The Stages component, (a list of stages for the experiment)
- The Endpoints component, (a list of endpoint conditions)
- The Process component, (the flow between stages)
- The Auto Parameters component, (how to customise the protocol for individual subjects)

A range of stage types is available to cater for various experimental situations; the range of stage types is listed in table 2.2.  The stage types have been created to permit the developer to create a wide range of experiments.  "Experiment specific" behaviour is included in some stage types, (such as in the case of "Serial Choice" type stages).  Experiment specific stage types make defining protocols easier as some "expert knowledge" is included in the stage itself removing the need for the developer to create the behaviour from scratch.

**Table 2.2: Predefined K-Limbic Stage Types**

| Stage Type | Function |
| --- | --- |
| Initial Stage | Sets the initial state of the operant box, (state prior to starting the protocol). |
| Exit Stage | Sets the exit state of the protocol, (state following completion of protocol). |
| Action | Waits for a trigger to occur |
| Reward | Dispenses a reward, (reinforcement). |
| Mark Trial/Set Outcome | Increments the trial number & defines the outcome of a trial, (e.g. correct/incorrect etc). |
| Fixed Ratio | Runs a fixed ratio. |
| Fixed Ratio (Meted) | Runs a fixed ratio. |
| Variable Interval | Runs a variable interval. |
| Shadow Stage | Responds to the activation of a shadow trigger. |
| Timer Stage | Used To Time the period of activation of a sensor. |
| Serial Choice - Inter-Trial Interval | The ITI stage of the 5CSRTT. |
| Serial Choice - Present Stimulus | The stimulus & limited hold of the 5CSRTT. |
| Serial Choice – Collection | The "Wait for collection" stage of the 5CSRTT. |
| Serial Choice – Timeout | The "Timeout" stage of the 5CSRTT. |
| Serial Choice – Training | A training stage for the 5CSRTT. |
| Interval Stage | A fixed or variable interval stage |
| Delay Stage | Waits for a specified delay to occur. |
| Progressive Ratio Stage | Implements a progressive ratio schedule |
| Cued Choice Stage | Used in DNMTP protocols |
| Sequential Choice Stage | Defines a series of responses |
| Cued Lever Stage | Defines a cued lever stage |
| Epoch Control Stage | Controls the epoch manager |
| Outcome Stage | Set the outcome of a trial |
| DRL Stage | Used in DRL tests |
| Timeout Stage | A general timeout stage |
| Location Stage | Defines the location of a subject. |

The important thing to note is that protocol definitions are generic, the protocol manager interacts with the details held in the database parameters of an individual subject to customise the protocol to a given subject.

For example, in the well-known "delayed match to position" task the database parameters determine the number of trials to be performed, the delay values to be used, the maximum time allowed for the experiment, whether or not the task is "matched to position" or "non-matched to position", whether or not incorrect trials are repeated and whether or not trials are organised in "banks".

As the details are subject specific you can run a session in which the details of the protocol running in box1 are completely different from the protocol running in box 2 etc.

This is achieved simply by assigning a subject to a chamber!

The list in table 2.2 details the various stages available to build a protocol.  If you have a protocol that is not catered for by the current stage types Conclusive want to hear from you, (we like nothing more than adding new stuff into K-Limbic).  Contact us with your requirements and we will add in the necessary controls to create your protocol, we can do this quickly and efficiently, (normally the longest process is communicating the details of the task), once your new facilities are added you will be able to adapt and run new protocols.

The need to have new stage types added into K-Limbic for new or novel protocols may appear disadvantageous, however once added the new stage types do not need to be added in again by other experimenters so in the global sense this saves an enormous amount of time.

Please remember Conclusive guarantee that we will not charge for making additions to K-Limbic for any currently supported K-Limbic system.

## 2.6 Subject Databases : Overview

K-Limbic provides a database utility for the storage of experiment parameters that are unique to individual subjects. Database parameters are central to the use of K-Limbic and are closely linked with protocol definitions. Database parameters interact with the protocol definition to create an experiment customised to the individual subject.

A subject database may store information on a very large number of subjects, (i.e. in excess of 1000), however it is recommended that databases are organised on a study basis, i.e. if a particular study consists of 64 subjects, then create a separate database for these subjects. K-Limbic can support any number of databases.

Databases are stored within installed test suite files. It is possible to maintain an arbitrary number of test suites on a K-Limbic system, the number and size being restricted by the available storage space, on most modern computers the space available is so large that it can be considered that an unlimited number of test suites can be supported.

K-Limbic provides utilities for creating and maintaining databases. It is advisable to become familiar with the operation and maintenance of databases before using the system in a "live" study.

All subjects in a database contain the following items:
- A Rewards Table
- A Dosing List
- A Parameters List

The Rewards Table stores a list of entries appropriate for delivering rewards and (ironically), shocks. The table entries define a series of "pulses" or "triggers". In the case of a protocol that only uses a pellet dispenser this table only requires a single entry defining a pulse long enough to trigger the pellet dispenser, the duration of the pulse required varies depending on the manufacturer of the pellet dispenser but 100mS is a typical pulse duration.
Protocols that deliver foot shocks as well as pellets will need two entries in the rewards table, one to define the pellet delivery pulse and one to specify the duration of the foot shock.

If alternative methods of reward delivery are used, such as syringe pumps or fluid dippers, the pulse duration will need to be determined by experimentation.

The Dosing List defines a list of doses that will be given to the subject during any given study. The operator selects the appropriate dose from the dosing list during the process of loading a session prior to running an experiment. The dose details appear in the output data file and can be included in the data analysis worksheets. Using suitable dosing lists is not essential to running K-Limbic but they can be used to save a lot of time when it comes to analysing data.

The Parameters List stores a collection of protocol specific parameters, such as FR schedules, five choice parameters, delay values etc. The parameter values are used to customise a given protocol for a specific subject. Refer to the documentation on your protocol to determine which database parameters are required for your protocol.

# 3

# Tutorial Guide

## Create A Simple Protocol

The following section is a step-by-step guide on how to create a K-Limbic protocol.

If K-Limbic has not been installed on the machine you are using refer to the installation and configuration instructions in the "K-Limbic Installation & Configuration Manual", (separate document).

This guide assumes that you have followed the instructions detailed in the "K-Limbic Installation & Configuration Manual". The instructions here detail the actions required to perform "Option 3" in the "What To Do Next" section of that manual.

K-Limbic supports a number of approaches to creating protocols, for instance:
- Importing an existing protocol.
- Copy and edit an existing protocol.
- Create a new protocol from scratch.

In many cases, (and not wishing to reinvent the wheel), importing existing protocols is the way to go, (perhaps combined with a bit of editing for individual preferences), Conclusive provide the K-Limbic repository, (a collection tried & tested protocols), for this purpose, or individual users can also export and share any of their own protocols.

This tutorial guide explains the process required to develop a simple fixed ratio type protocol from scratch. This is intended as an example of the development process, (the protocol details may not be applicable for your purposes). Further guides are available on the settings of a range of protocols. The process of creating and editing is common to all protocols.

These instructions will detail how to…
1. Check that the user profile is correctly assigned.
2. Create a protocol, (as an example we will create a Fixed Ratio training protocol).
3. Create a Development Test Suite, define a session and a database template
4. Build an Installed Test Suite from our Development Test Suite.
5. Prepare the Installed Test Suite for use
6. Load and run the session within the Test Suite.

**3.1 Check The User Profile**
The default user contains an operant profile and a five choice profile, each profile has it's own resource list, (this is different from the resource list that was configured during the installation). The resources in the user profile are used at certain points in the process of developing protocols and editing databases. The use of a separate resource list allows a user to develop a protocol, or test suite, on their K-Limbic system even if the system is not configured to run that protocol.

K-Limbic displays the current user name, group name and profile in the status bar at the bottom of the main screen. It is useful to check that the settings are correct before commencing the development of a protocol/test suite.

From the K-Limbic main menu select…
File / User / Configure User…
This will present the "Configure User Details" window.

The "Current Groups" list will contain a single entry, "Operant Group", if this entry does not have an arrow next to it with "Current" written on the right, then select the "Operant Group" entry in the list and click on the "Assign As Current" button, this will connect the current user to all the files available to the "Operant" group.

If the "Current Groups" list contains more than one group, then your system has been configured to use a number of user groups, contact your K-Limbic system administrator for advice on how they have assigned the groups, to determine which group is appropriate.

It is also possible to modify the password for the current subject using the "Modify Password" button.

Click on "OK" to close the "Configure User Details" window.

From the K-Limbic main menu select…
File / User / Profile Manager…
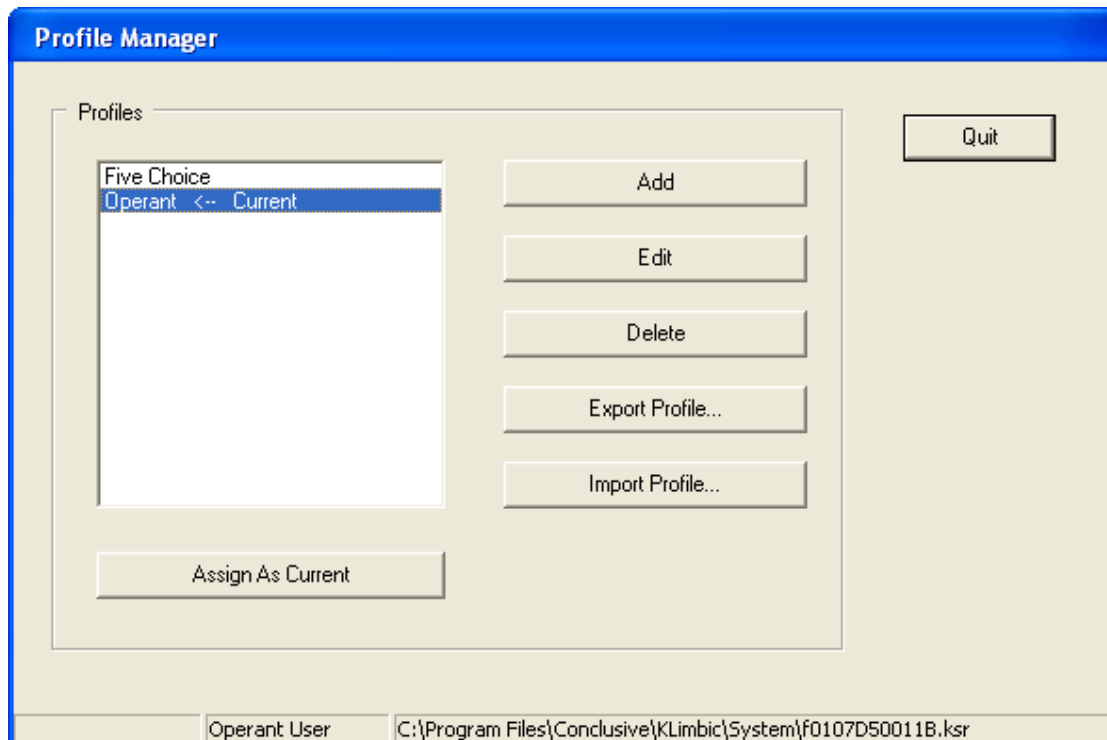This will present the "Profile Manager" window.



**Figure 3.1 Profile Manager Window**

The default user contains two profiles, "Operant" and "Five Choice", the appropriate profile for the fixed ratio protocol is the "Operant" profile, select the "Operant" entry in the "Profiles" list and click on the "Assign As Current" button, the "Operant" profile will be marked as "Current".
Click on the "Quit" button to close the "Profile Manager" window.

This completes the check of the user profile.

**3.2 Create Your New Protocol**

Protocols are created, edited and managed using the "Protocol Manager".
From the K-Limbic main menu select…
Protocols / Protocol Manager…          (or press function key F3)
This will present the "Protocol Manager" window.



**Figure 3.2 Protocol Manager Window**
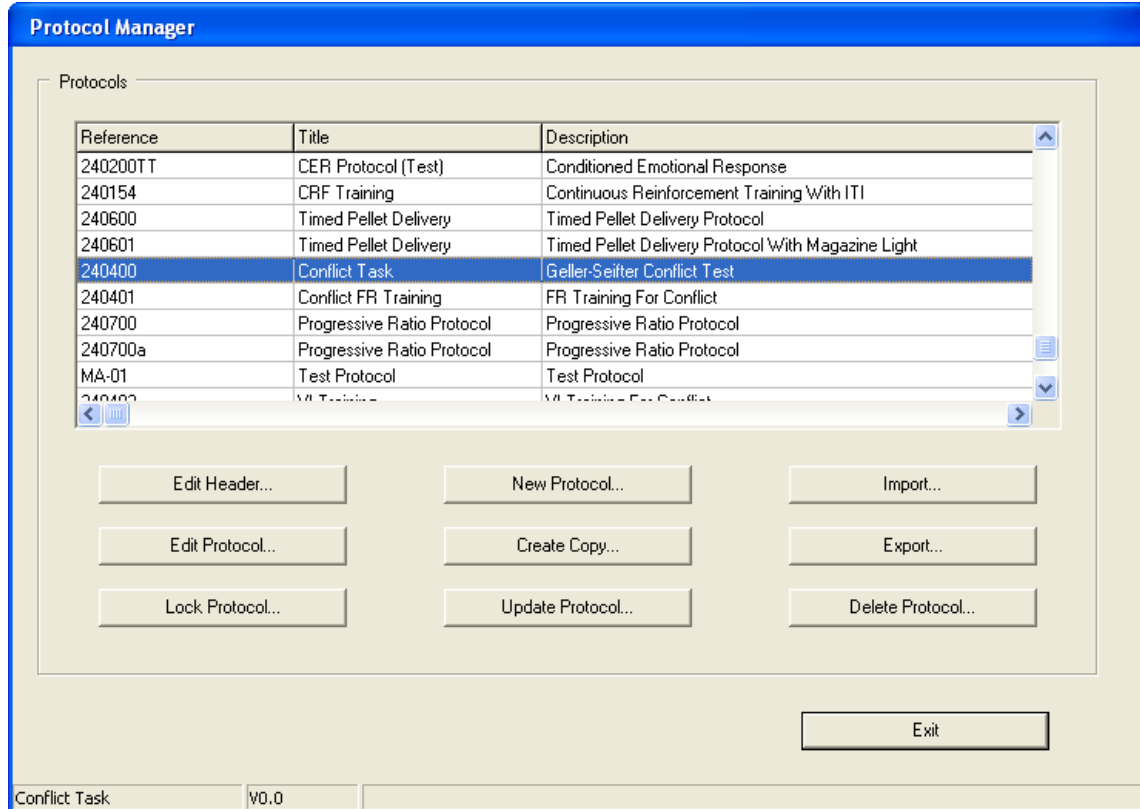
The window contains a list of the protocols installed for the current user, (figure 3.2 illustrates a typical system with a number of protocols already defined).  Below the list of protocols are a number of buttons used to perform a variety of actions.

We will use the protocol manager to create the new protocol and access the protocol editor to create and edit the components of the protocol.

### The Fixed Ratio Protocol

At this stage it would be instructive to discuss the fixed ratio protocol we are about to create. A protocol such as this is frequently used to train animals to respond on a lever to receive maximum rewards. The flowchart for our protocol is shown in figure 3.4.

The protocol performs a simple fixed ratio task. In the flowchart each yellow box represents a "stage" within the protocol. As can be seen the protocol has six stages:
- Start
- Fixed Ratio
- Dispense Pellet
- Correct Trial
- Pellet Collected
- Exit

The Start stage, (stage 1), determines the state of the operant chamber prior to running the protocol, (i.e. the state of the chamber when the animal is placed into the chamber. Once started the protocol enters the fixed ratio stage, (stage 2), this stage extends a single lever into the chamber and pauses until the current FR schedule is completed on the lever, when the FR is complete the stage ends and the protocol triggers a pellet delivery, (stage 3), (this stage simply triggers the pellet dispenser, it does not wait for the pellet to be delivered), in addition this stage also turns on the magazine light as a signal to the animal that the pellet is present in the receptacle.
The protocol then marks this trial as correct and increments the trial number, the protocol then loops back to the fixed ratio stage and starts again.
Stage 5, (Pellet Collected), is triggered when the subject collects the pellet from the receptacle, (sensed using a head entry detector), stage 5 turns the magazine light off and then returns to the fixed ratio stage.
The loop continues until the end of the protocol, this is triggered by a protocol timeout or a "count endpoint". The user specifies the protocol timeout, for sake of this guide we will assume a value of 20 minutes.

### Targets & Dummies
The assignment of "target" & "dummy" resources is important in most protocols. In general a "target" identifies the "correct response" sensor and the "dummy" identifies one or more "incorrect response" sensors. A number of stage types provide controls for setting the Target and Dummy whereas several stage types set the Target & Dummy sensors automatically.

For example: an "Action" stage type provides controls for setting the either the Target or the Dummy to the following options:
- None
- Selected Lever
- Selected Sensor
- All Levers
- All Sensors
- Standard (Auto)
- Reverse (Auto)
- Assigned Resource
- Use Previous

The effect of selecting an option from the above list is as follows:

| | |
|---|---|
| None | No target/dummy assigned |
| Selected Lever | Extract the setting from the "Selected Lever" database parameters |
| Selected Sensor | Extract the setting from the "Selected Sensor" database Parameters |
| All Levers | Set all levers as targets/dummies |
| All Sensors | Set all sensors as target/dummies |
| Standard (Auto) | Extract the setting from the auto control |
| Reverse (Auto) | Extract the setting from the auto control |
| Assigned Resource | Assign a specific resource as the target/dummy |
| Use Previous | Retain the setting from the previous stage(s) |

Target and dummy resources are assigned at each stage therefore in say a protocol that uses two retractable levers either lever can be assigned as the target lever at any stage in the protocol.

In many cases the auto control of the protocol will determine which lever is the current target, for instance in a DMTP protocol the sample lever is the "target". The auto control contains a setting for the target for each trial in the protocol. Using the auto control to assign the target is achieved by selecting the "Standard (Auto)" or "Reverse (Auto)" options.

The fixed ratio protocol requires the subject to make multiple responses on the target lever to obtain rewards however you will note that at no stage in the protocol do we make any assignment for the target lever, so how does the system know what which lever to use?

Fixed ratio stages "know" that a target is required, (and perhaps a dummy too), however they also know that they need to extract this information from the subject database, (from the FR Parameters), therefore as the session is loaded the protocol manager extracts the FR Parameters from the database and sets the appropriate sensor as the target.
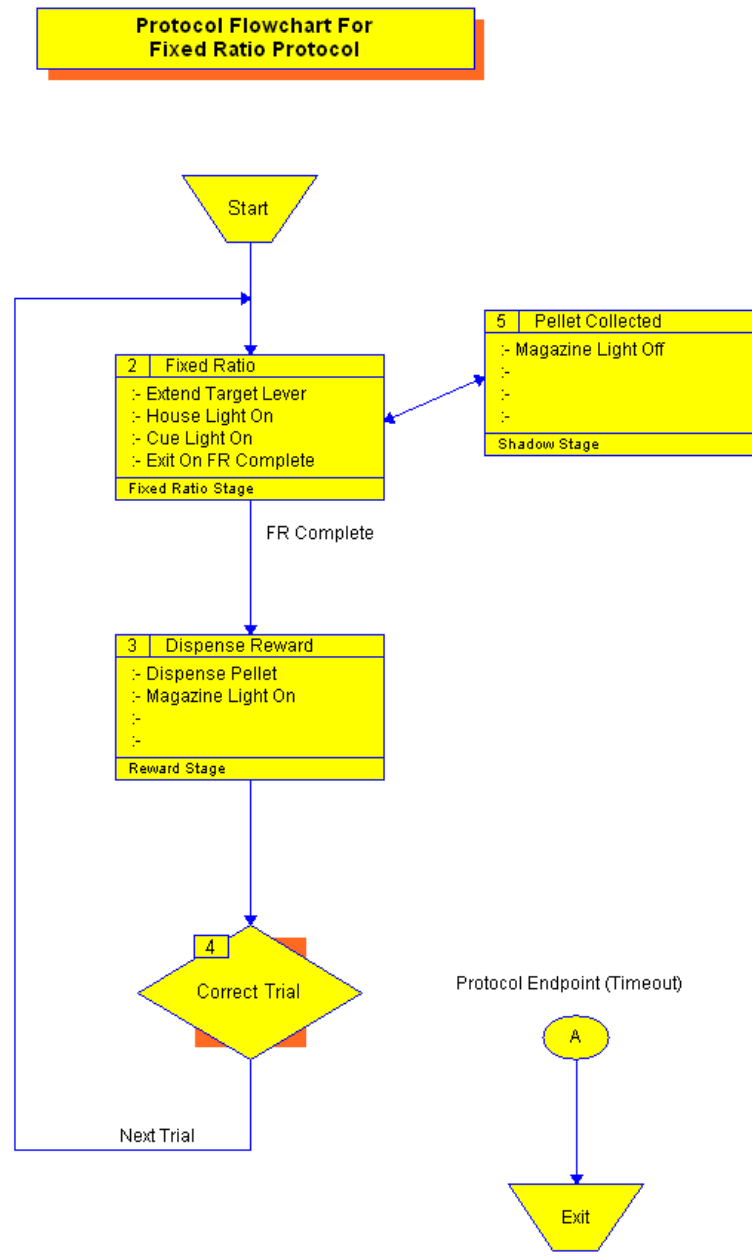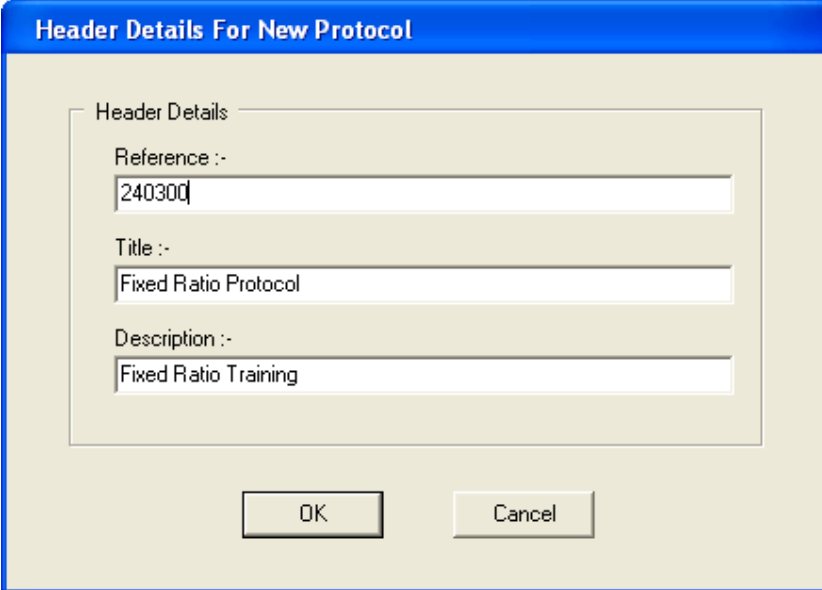
**Figure 3.4 Fixed Ratio Protocol Flowchart**

**Create The Protocol**

Click on the "New Protocol…" button, this will present the "Header Details For New Protocol" window. Enter details into the Reference, Title & Descriptions fields, suitable entries are illustrated in figure 3.3, the Reference entry should be unique, the Title & Description entries can be any values, (header values can be changed at any time so the precise entries are unimportant at this stage.



**Figure 3.3 Header Details For New Protocol Window**

When the three settings are complete, click on the "OK" button and the new, (blank), protocol is created and added to the "Protocols" list.

**Edit The Protocol**

Select the new protocol in the "Protocols" list and click on the "Edit Protocol" button; this opens the "Protocol Editor", shown in figure 3.5.
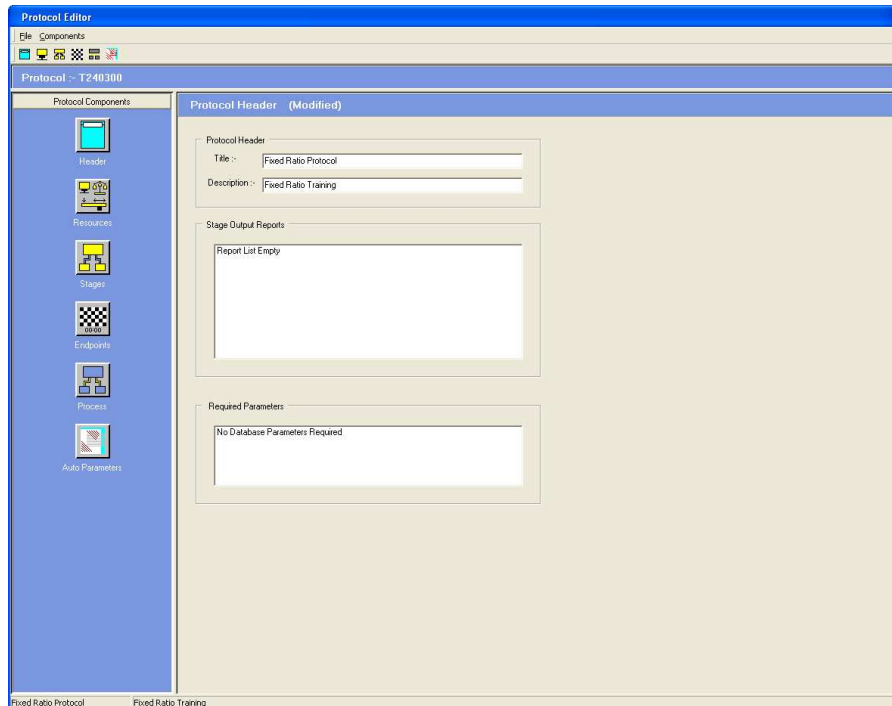


**Figure 3.5 Protocol Editor Window**

The Protocol Editor displays the new protocol, (as yet there is not much to display). The left hand section accesses editors for the various protocol components, the right hand section displays the current contents of each component of the protocol.

The procedure for creating/editing a protocol is to select each component in turn and make the appropriate settings.

The components are as follows:

- The Header Component, (the Reference, Title & Description of the protocol)
- The Resources component, (the items required in the operant chamber)
- The Stages component, (a list of stages for the experiment)
- The Endpoints component, (a list of endpoint conditions)
- The Process component, (the flow between stages)
- The Auto Parameters component, (how to customise the protocol for individual subjects)

We have already set the entries in the header but as can be seen the Title & Description can be modified here. Modify the Reference entry by using the "Edit Header" button from the Protocol Manager window.

**The Resources Component**

Click on the icon labelled "Resources"; the right hand side displays the current list of Resources. Perhaps surprisingly the list is populated already, this is because the list of resources defined for the current profile has automatically been copied into the protocol as it was created. The initial list of resources in the protocol will therefore depend on the resources list of the current profile, however if the default settings have been left in place the resource list should look like that shown in figure 3.6.
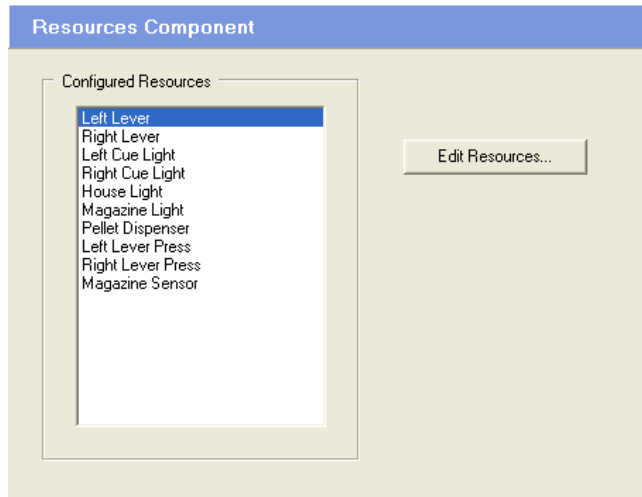


**Figure 3.6 The Configured Resources**

The resource list contains:
- Two retractable lever controls, (Left Lever & Right Lever)
- Two cue lights, (Left Cue Light & Right Cue Light)
- Two lights, (House Light & Magazine Light)
- One pellet dispenser
- Two lever press sensors, (Left Lever Press & Right Lever Press)
- One sensor, (Magazine Sensor, a head entry sensor in the pellet receptacle)

The list should contain an entry for each resource that will be used in the protocol, for the case of the fixed ratio protocol the default list is OK, however it is simple to add/delete/edit the entries in the Resources list as will now be shown.

Click on the "Edit Resources" button; this will display the "Configure Resources" window. The window contains the list of configured resources and a list of all available resources.

With the "Left Lever" entry highlighted in the "Configured Resources" list; click on the "Edit Resource" button. The "Edit Resource" window appears.

**Resource Type**
The window shows that this resource is "Lever Control <1>", i.e. it is the first resource in the list that is a "Lever Control" resource, lever control resources control the position of retractable levers. The Resource Type is not editable. If you do not want a resource of a particular type it should be deleted from the list.

**Resource Description**
The Resource Description is a user-defined field used as the screen identifier for this resource, at present this is set to "Left Lever", but it may be changed to any value that the user prefers.

**Class**
The class of the resource determines whether the resource is controlled by K-Limbic or whether it signals a response from the animal. In general items controlled by K-Limbic should be assigned as class "Driver", however the stimulus lights in a five choice chamber should be assigned as class "Varilux Driver". In general any sensors in the chamber should be assigned as class "Sensor", however

any sensors which are attached to K-Limbic via an NC-9 controller, (when using Diplomat hardware from Conclusive), should be assigned the class "NC-9 Sensor".

Leave the class of the lever control as "Driver". Click on the "OK" button to close the "Edit Resource" window.

To add a new resource to the list, select the required resource type in the "Available Resources" list and click on the "Add Resource" button, a new resource of the chosen type will be added to the list. For example to add a light to the list: select the "Light" entry in the Available Resources list and click on the "Add Resource" button. The "Configured Resources" list will update with the new entry "Light <3>", indicating that it is the third light in the list of configured resources. Click on the "Edit Resource" button to modify the description for this resource to something meaningful.

To delete an unwanted resource from the list, select the resource in the "Configured Resources" list and click on the "Delete Resource" button, the resource will be deleted.

For the purposes of this example, modify the resource list, (if required), so that it is the same as shown in figure 3.5 above and close the "Configure Resources" window.

**Configuring Resources**
Please note that the resources defined in the protocol do not specify the interface channel that the resource is attached to. When the protocol is loaded prior to running a session, each resource in the resource list is compared with the resource list configured for the particular system, the channel assignments from the configured list are then used for the resources in the protocol, i.e. the first resource in our list is "Lever Control <1>", the configured resource list is searched for a resource of type "Lever Control <1>", (the resource description field is ignored), when found the channel assignment for this resource is copied into the protocol. If no resource is found in the configured list then a warning message is issued and the session aborted. Using this technique the same protocol can be used on different systems regardless of the manner in which the resources are attached.

**The Stages Component**

We discussed the protocol flowchart a little earlier and determined that the fixed ratio protocol consists of six stages, we need to add these stages to the stages component and edit them so that they act as required.

Click on the icon labelled "Stages"; the right hand side displays the current list of stages. Unsurprisingly the stage list is initially empty. Click on the "Edit Stages" button, the "Configure Stages" window displays. The "Configure Stages" window is used to add, delete and edit the stages of the protocol.
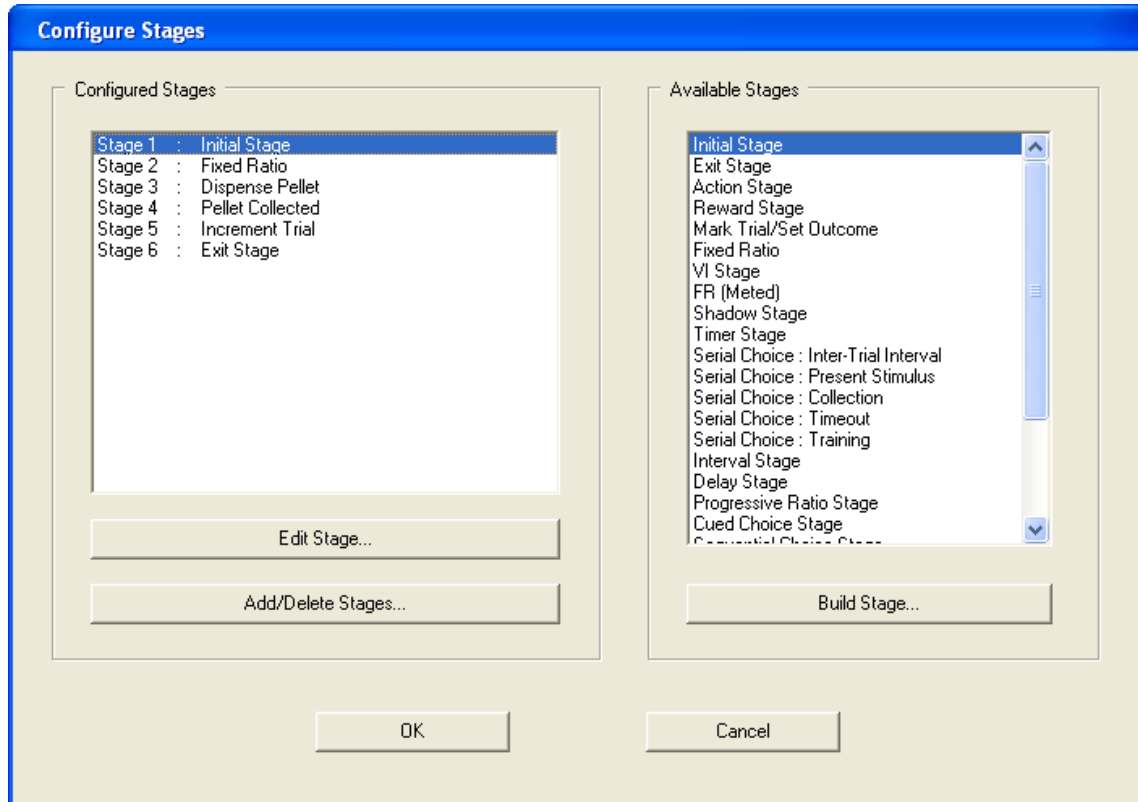


**Figure 3.7 The Configure Stages Window**

We have a couple of options as to how to add all the stages to our protocol, we can either add all the stages together and then edit them, or we can use the "Build Stage" button to add and edit stages individually, in either case the editing process is exactly the same so it is simply a matter of personal preference. Here we will use the "Build Stage" button.

**Add the "Start" stage**
The "Start" stage is of the type "Initial Stage" All protocols must have one stage of type Initial Stage. By conventional the Initial Stage is always called "Start" and is listed as stage 1. In this guide the terms "Initial Stage" and "Start" are considered synonymous. The Start stage determines the state of the operant chamber prior to the start of the protocol, (i.e. the state of the chamber as the animal is placed into the chamber).

Select the stage type "Initial Stage" in the "Available Stages" list and click on the "Build Stage…" button. The "Stage Wizard" appears: this is a series of windows that prompts the user to select the appropriate settings for each stage. In the case of the "Initial Stage" the title at the top of the wizard should display…

Step 1 of 6: Lever Positions
In our case we want the levers to be retracted and all lights off, the default values of the Start stage are therefore pretty much appropriate.
The "Stage" entry just below the title displays "Initial Stage", delete this entry and enter "Start" instead.

By default the levers are retracted so step 1 is OK.
Click on the "Next" button at the bottom of the window.

Step 2 of 6: Cue Light States
By default the cue lights are off, so step 2 is OK.
Click on the "Next" button at the bottom of the window.

Step 3 of 6: Enclosure Light States
By default the enclosure lights are off, so step 3 is OK.
Click on the "Next" button at the bottom of the window.

Step 4 of 6: Display Message
We do not wish to send a message to the screen display at this time so step 4 is OK.
Click on the "Next" button at the bottom of the window.

Step 5 of 6: Target/Dummy Settings
There are no Target/Dummy resources required at this time so step 5 is OK.
Click on the "Next" button at the bottom of the window.

Step 6 of 6: Message Filter
The "Message Filter" should be enabled for this protocol so ensure this option is checked in step 6.
Enabling the Message Filter instructs the system not to send messages sent when a sensor is
released.  In the fixed ratio protocol, we are not interested in when the lever is released, or when the
animal exits the pellet receptacle so we can ignore sensor releases.
(The Message Filter has no effect in the Med Associates version of K-Limbic).
Click on the "Finish" button at the bottom of the window.

The settings for the initial stage are complete so the wizard closes and the new stage is added to the
"Configured Stages" list, the list should now display "Stage 1: Start", indicating that the stage list
contains the first stage we have defined.

**Add the "Fixed Ratio" stage**
The "Fixed Ratio" stage is of the type "Fixed Ratio".

The fixed ratio stage:
- Extends the "target" lever
- Illuminates the cue light above the lever
- Illuminates the house light
- Leaves the magazine light unchanged.

The fixed ratio schedule, (the number of times the subject must respond before receiving a reward), is not fixed as part of the protocol, a more flexible approach is used. Prior to the start of the current trial the protocol manager interrogates the subject database to determine the appropriate fixed ratio to use during this stage. The fixed ratio is therefore subject dependent.

Select the stage type "Fixed Ratio" in the "Available Stages" list and click on the "Build Stage…" button. The "Stage Wizard" appears: the title at the top of the wizard should display…

Step 1 of 7: Lever Positions
We want the lever to extend during this stage, but only if it is the "target" lever, (more about target levers shortly). In the "Left Lever" tab click the "Extend If: Target" option, then select the "Right Lever" tab and click the "Extend If: Target" option.
Click on the "Next" button at the bottom of the window.

Step 2 of 7: Cue Light States
We want the light above the extended lever to come on during this stage. In the "Left Cue Light" tab click on the "On If: Target" option, then select the "Right Cue Light" tab and click the "On If: Target" option.
Click on the "Next" button at the bottom of the window.

Step 3 of 7: Enclosure Light States
We want the house light to be on throughout this stage. In the "House Light" tab click the "On" option. Select the "Magazine Light" tab. The magazine light is turned on when a pellet is dispensed and remains on until the pellet is collected, thus the fixed ratio stage does not directly control the state of the magazine light, click on the "No Change" option for the magazine light.
Click on the "Next" button at the bottom of the window.

Step 4 of 7: Display Message
We do not wish to send a message to the screen display at this time so step 4 is OK.
Click on the "Next" button at the bottom of the window.

Step 5 of 7: Stage Exit Triggers
We want to remain in the fixed ratio stage until the fixed ratio is complete, so ensure the Disable Timeout" option is selected and the "Exit When Ratio Complete: On Target" option is selected, (they will be by default).

When the subject collects the pellet we want to alter the state of the magazine light, without exiting the fixed ratio stage, this is achieved using a "Shadow" stage to change the state of the light. A response on the "Shadow Trigger" triggers the shadow stage, click on the "Assign Shadow Trigger" button on the right of the window; select the "Magazine Sensor" entry in the list of resources and click on the "OK" button. The Magazine Sensor is assigned as the shadow trigger for this stage.
Click on the "Next" button at the bottom of the window.

Step 6 of 7: FR Controls
A protocol can contain several fixed ratio stages and each can use a different setting of fixed ratio, a "Fixed Ratio Series" can be used for this, however in this simple protocol there is a single fixed ratio stage which requires a single fixed ratio value, therefore ensure the "Single Entry" option is selected, (as it will be by default).
Click on the "Next" button at the bottom of the window.

Step 7 of 7: Stage Reports
The Stage Reports section allows the developer control over whether or not the stage generates any reports to send to the output file.  In this example protocol we will enable some output reports, without being overly concerned with detailed analysis of results.
Check the "Enable Reports" option to enable the rest of the options.
Check the following options…

| | |
|---|---|
| Stage Entry | - The entry time of this stage |
| Stage Exit | - The exit time of this stage |
| Target ID | - The identifier of the target lever |
| Target Count | - The number of responses on the lever during the stage |
| Counts (Sensors) | - The number of nose pokes into the pellet receptacle |
| Target Ratio | - The fixed ratio schedule |

Click on the "Finish" button at the bottom of the window.
The fixed ratio stage will generate each of these reports at the end of each trial, i.e. once per iteration through the protocol.  They will appear in the output file as a row of data items.

The settings for the fixed ratio stage are complete so the wizard closes and the new stage is added to the "Configured Stages" list.

Note that we have not specified which lever is the "target" lever for this stage.  This is because the protocol manager knows that it has to interrogate the subject database for this information.  Using this approach the assignment of target lever and fixed ratio value become subject dependent.

**Add the "Dispense Reward" stage**
The "Dispense Reward" stage is of the type "Reward Stage".

The Dispense Reward stage:
- Triggers a pellet delivery, (it does not wait for the pellet to be delivered or check that delivery has occurred)
- Turns on the magazine light

Select the stage type "Reward Stage" in the "Available Stages" list and click on the "Build Stage…" button.  The "Stage Wizard" appears: the title at the top of the wizard should display…

Step 1 of 7: Rewards and Shocks
The "Stage" entry just below the title displays "Reward Stage", delete this entry and enter "Dispense Pellet" instead.
We want to trigger the pellet dispenser so click on the "Dispense" option.
Leave the "Reward Table" section unchanged.
Click on the "Next" button at the bottom of the window.

Step 2 of 7: Lever Positions
By default the lever controls are set to "No Change" this is OK.
Click on the "Next" button at the bottom of the window.

Step 3 of 7: Cue Light States
By default the cue light controls are set to "No Change" this is OK.
Click on the "Next" button at the bottom of the window.

Step 4 of 7: Enclosure Light States
By default the enclosure light controls are set to "No Change" this is OK for the house light but not for the magazine light.
Click on the "Magazine Light" tab and change the setting to "On"
Click on the "Next" button at the bottom of the window.

Step 5 of 7: Display Message
We do not wish to send a message to the screen display at this time so this step is OK.
Click on the "Next" button at the bottom of the window.

Step 6 of 7: Target/Dummy Settings

We do not require a target/dummy setting at this time so the default values of "None" are OK.
Click on the "Next" button at the bottom of the window.

Step 7 of 7: Stage Reports
No output reports are required in this stage so leave the "Enable Reports" option unchecked so that output reporting is disabled in this stage.
Click on the "Finish" button at the bottom of the window.

The settings for the "Dispense Pellet" stage are complete so the wizard closes and the new stage is added to the "Configured Stages" list.

**Add the "Correct Trial" stage**
The "Correct Trial" stage is of the type "Mark Trial / Set Outcome".

The Correct Trial stage:
- Increments the trial number
- Marks the trial as "Correct"

The "Correct Trial" stage marks the end of the current "trial".  The protocol manager performs housekeeping functions whenever a trial marker stage is encountered during a protocol, (in this case the counts within the fixed ratio stage are reset).  Trial marker stages are "invisible" to the subject.

Select the stage type "Mark Trial/Set Outcome" in the "Available Stages" list and click on the "Build Stage…" button.  The "Stage Wizard" appears: the title at the top of the wizard should display…

Step 1 of 2: Trial Marker Actions
The "Stage" entry just below the title displays "Mark Trial/Set Outcome", delete this entry and enter "Correct Trial" instead.
In the "Trial Control" section select the "Increment Trial" option
In the "Trial Outcome" list, select the "Correct Trial" entry.
Click on the "Next" button at the bottom of the window.

Step 2 of 2: Display Message
We do not wish to send a message to the screen display at this time so this step is OK.
Click on the "Finish" button at the bottom of the window.

**Add the "Pellet Collected" stage**
The "Pellet Collected" stage is of the type "Shadow Stage".

The Pellet Collected stage:
- Turns off the magazine light

In general any stage of the protocol defines a specific state for each of the resources in the chamber. In some cases it is useful to change the state of one or more of the resources during a stage, an example of this is turning off the magazine light when the subject collects a delivered pellet, a shadow stage is used for this purpose.  The shadow stage is activated by a response on the "Shadow Trigger" of the current stage, when the subject activates the shadow trigger the operant chamber is set to the state defined by the shadow stage.

Select the stage type "Shadow Stage" in the "Available Stages" list and click on the "Build Stage…" button.  The "Stage Wizard" appears: the title at the top of the wizard should display…

Step 1 of 5: Lever Positions
The "Stage" entry just below the title displays "Shadow Stage", delete this entry and enter "Pellet Collected" instead.
By default the lever controls are set to "No Change" this is OK.
Click on the "Next" button at the bottom of the window.

Step 2 of 5: Cue Light States
By default the cue light controls are set to "No Change" this is OK.
Click on the "Next" button at the bottom of the window.

Step 3 of 5: Enclosure Light States
By default the enclosure light controls are set to "No Change" this is OK for the house light but not for the magazine light.
Click on the "Magazine Light" tab and change the setting to "Off"
Click on the "Next" button at the bottom of the window.

Step 4 of 5: Display Message
We do not wish to send a message to the screen display at this time so this step is OK.
Click on the "Next" button at the bottom of the window.

Step 5 of 5: Stage Reports
No output reports are required in this stage so leave the "Enable Reports" option unchecked so that output reporting is disabled in this stage.
Click on the "Finish" button at the bottom of the window.

The settings for the "Pellet Collected" stage are complete so the wizard closes and the new stage is added to the "Configured Stages" list.


**Add the "Exit" stage**
The "Exit" stage is of the type "Exit Stage" All protocols must have one stage of type Exit Stage.  By conventional the Exit Stage is always called "Exit" and is listed as the final stage in the stage list.  In this guide the terms "Exit Stage" and "Exit" are considered synonymous.  The Exit stage determines the state of the operant chamber after the end of the protocol.
In our case we want the levers to be retracted and all lights off, these are the default values of the Exit stage.

Select the stage type "Exit Stage" in the "Available Stages" list and click on the "Build Stage…" button.
The "Stage Wizard" appears: the title at the top of the wizard should display…

Step 1 of 3: Lever Positions
The "Stage" entry just below the title displays "Exit Stage", delete this entry and enter "Exit" instead.
By default the levers are retracted so step 1 is OK.
Click on the "Next" button at the bottom of the window.

Step 2 of 3: Cue Light States
By default the cue lights are off, so step 2 is OK.
Click on the "Next" button at the bottom of the window.

Step 3 of 3: Enclosure Light States
By default the enclosure lights are off, so step 3 is OK.
Click on the "Finish" button at the bottom of the window.

The settings for the "Exit Stage" stage are complete so the wizard closes and the new stage is added to the "Configured Stages" list.

The "Configured Stages" list now contains the six stages required for the fixed ratio protocol, so the "Configure Stages" window is complete.  Click on the "OK" button to close the window and return to the protocol editor.

**The Endpoints Component**

A protocol ends by either:
- The operator issuing an "end of session" command.
- The "flow" of the protocol sending the protocol to an "Exit" stage.
- A predefined endpoint condition is met.

Although it is quite feasible to require the operator to manually end the protocol by issuing an "end of session" command, it is considered good practise to develop protocols that end automatically. In some cases, the "flow" of the protocol always takes the protocol to the "Exit" stage, in such a case there is no need to define an endpoint condition. However for any "trials" based test or a test that is designed to last for a specific period it is normal to define one or more endpoint conditions. The protocol will end when any endpoint condition is met.

We discussed the protocol flowchart a little earlier and determined that the fixed ratio protocol will end after a predetermined amount of time. In addition to this it is also useful to end the protocol after a maximum number of pellets have been delivered, this prevents the animal receiving too many pellets on any given session, which may have an effect on their motivation to perform the task on subsequent sessions.

Click on the icon labelled "Endpoints"; the right hand side displays the current list of endpoints. Unsurprisingly the endpoints list is initially empty. Click on the "Edit Endpoints" button, the "Configure Endpoints" window displays. The "Configure Endpoints" window is used to add, delete and edit the endpoints of the protocol.
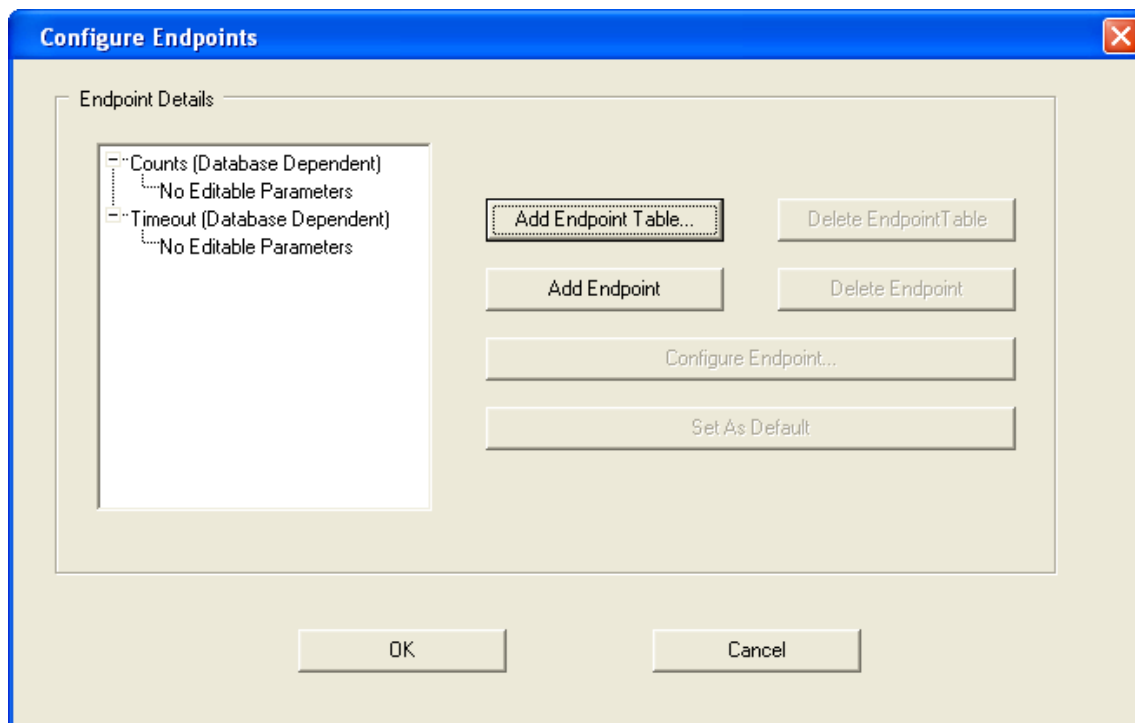


**Figure 3.8 The Configure Endpoints Window**

There are a number of endpoint triggers available, each is available as a table, if you define an endpoint table that has more than one value, for example a timeout table with three entries, the protocol manager will prompt the operator with the table during the load session procedure. The operator selects the endpoint entry appropriate for the current session.

To make the protocol as generic as possible we will add "database dependent" endpoint entries. This instructs the protocol manager to interrogate the subject database for suitable entries from which to set the endpoint values. We will add Timeout & Counts entries in the endpoints table.

**Add the endpoints**
Click on the "Add Endpoint Table" button.
In the "Select Endpoint Type" list select: "Timeout (Database Dependent)"
Click on the "OK" button.  The endpoint entry is added to the endpoints list.

Click on the "Add Endpoint Table" button.
In the "Select Endpoint Type" list select: "Counts (Database Dependent)"
Click on the "OK" button.  The endpoint entry is added to the endpoints list.

Click on the "OK" button, the "Configure Endpoints" window closes and the two endpoints are added to the protocol.

**The Process Component**

The process component determines the passage of the protocol from one stage to another, i.e. it represents the arrows in the flow diagram.
We now set the process component to define which stage is entered when another stage ends.
Refer to the flowchart of figure 3.4 to see the flow from one stage to the next.

The procedure is to select a "source" stage, identify each exit trigger in that stage and specify a "destination" stage for each trigger.

Click on the icon labelled "Process"; the right hand side displays the current list of stages and the destinations for the triggers of each stage. By default all destinations are undefined.
Click on the "Edit Process" button, the "Configure Process" window displays.

The "Source Stage" section contains a list of the stages of the protocol, and the destinations for the currently selected source stage.

The "Destinations" section contains a list of the stages of the protocol.

On entry to the "Configure Process" window the first stage in the stages list is the selected "Source" stage, in the fixed ratio protocol this is the "Start" stage. The "Start" stage has a single destination, labelled as "Continue", the protocol moves from the Start stage to the "Continue" destination when the operator issues the "start of session" command.

The "Fixed Ratio" stage is the destination of the "Start" stage, ensure the "Fixed Ratio" stage is selected in the "Destinations" list and click on the "Assign Destination" button. The "Continue" destination is updated. This completes the process for the "Start" stage.

In the "Source Stage" list select the "Fixed Ratio" stage, the destinations list updates to display three entries. Set the destinations as below:
- Ratio Complete (On Target)  Dispense Pellet
- On Shadow Trigger  Pellet Collected
- Protocol Timeout  Exit

This completes the process for the "Fixed Ratio" stage.

In the "Source Stage" list select the "Dispense Pellet" stage, the destinations list updates to display two entries. Set the destinations as below:
- Continue  Correct Trial
- Rewards Endpoint  Exit

This completes the process for the "Dispense Pellet" stage.
The "Rewards Endpoint" entry is displayed because of the "Counts" endpoint. The "Counts" endpoint may specify a maximum number of rewards. If the maximum number of rewards is triggered then the protocol will pass to the Exit stage, thus ending the protocol.

In the "Source Stage" list select the "Correct Trial" stage, the destinations list updates to display two entries. Set the destinations as below:
- Continue  Fixed Ratio
- Trials Complete  Exit

This completes the process for the "Dispense Pellet" stage.
Normally passing through the "Correct Trial" stage returns the protocol to the "Fixed Ratio" stage, however if a maximum number of trials is specified in the "Counts" endpoint then the protocol will pass to the Exit stage, thus ending the protocol.

The "Pellet Collected" stage and the "Exit" stage have no valid destinations therefore the process component is now complete. Click on the "OK" button to close the "Configure Process" window.

**The Auto Parameters Component**

The auto parameters component determines which type of "Auto Control" the protocol manager builds for use during the protocol. In general the "Auto Control" is a list of experiment specific values used by the protocol manager to customise a generic protocol to a specific subject.

A range of auto controls is available, some are specific to types of experiment, such as the CER & DMTP controls and some are more general such as the fixed ratio & VI controls.

The Auto Control type determines which parameters are required in the subject database, the required parameters appear in the list below the auto control type. The fixed ratio protocol uses a fixed ratio stage and therefore requires an auto control of type: "Fixed Ratio Control".

Click on the icon labelled "Auto Parameters"; the right hand side displays the currently selected "Auto Control Type".
Click on the "Edit Auto Parameters" button, the "Configure Auto Parameters" window appears.
Select the "Fixed Ratio Control" entry in the list. The required parameters list will update to indicate that FR Parameters are required in the subject database.
Click on the "OK" button to close the "Configure Auto Parameters" window.

**The Fixed Ratio Auto Control**

The fixed ratio protocol uses the fixed ratio auto control, but what does the fixed ratio auto control provide? The fixed ratio auto control produces one or more sets of fixed ratio values, the number of sets is determined by the database FR parameters. The fixed ratio values are referenced by one or more fixed ratio stages in the protocol.

In our simple fixed ratio protocol we have one fixed ratio stage, therefore an auto control with one set of fixed ratio values is required. It is possible for the database parameters to specify more than one set of values however only the first set of values is actually used by the protocol.

The set of fixed ratio values contains the following settings:
- Fixed Ratio – The number of responses required to complete 1 iteration
- Target Assignment – Controls which sensor is considered the target
- Dummy Assignment – Controls which sensor is considered the dummy
- Enable Reset – Reset the count if a response is made on the "Dummy"
- Enable Rewards – Enable/Disable rewards after the iteration is complete
- Disable Unused Lever – If a target & dummy are used, then following the first iteration the lever, (or sensor), that did not complete the iteration is disabled
- Dummy As Target – Consider the dummy sensor as a second target sensor

Using an FR auto control to determine these values, (as opposed to setting them directly in the protocol), means that all of these values become subject (and/or session) dependent.

**Return To The Header Component**

The protocol is now complete, but before we close the protocol editor it is useful to return to examine the Header component.

Click on the icon labelled "Header"; the right hand side displays the current protocol header, the list of stage reports and the list of required parameters. Examine the lists to check that the entries below are displayed.

**Stage Output Reports**
The settings of the stage component specified that the fixed ratio stage generate six stage reports, a list of the reports is displayed in the Stage Output Reports list.

**Required Parameters**
The Required Parameters list contains the FR Parameters entry used for the auto control component and two entries for the endpoints component, "Endpoints – Timeout Table" and "Endpoints – Counts Table".

The Required Parameters list specifies that all subjects in the subject database contain the three entries listed above.  In addition, if the FR Parameters specify that the target assignment is set to "Use Selected Lever", (as will be the case in our database), a "Selected Lever" parameter will also be required.

**Close The Protocol Editor**

The fixed ratio protocol is now complete and we can close the protocol editor.

Select File / Exit from the menu in the protocol editor window and click on "Yes" to save the changes to the protocol.

Click on the "Exit" button to close the Protocol Manager.

The protocol file is now complete.

**3.3 Create A Development Test Suite**
To use the fixed ratio protocol to run a session we need to create a Development Test Suite and then an installed test suite.
Test Suites are created, edited and managed using the "Test Suite Manager".
From the K-Limbic main menu select…
Test Suites / Test Suite Manager…           (or press function key F4)
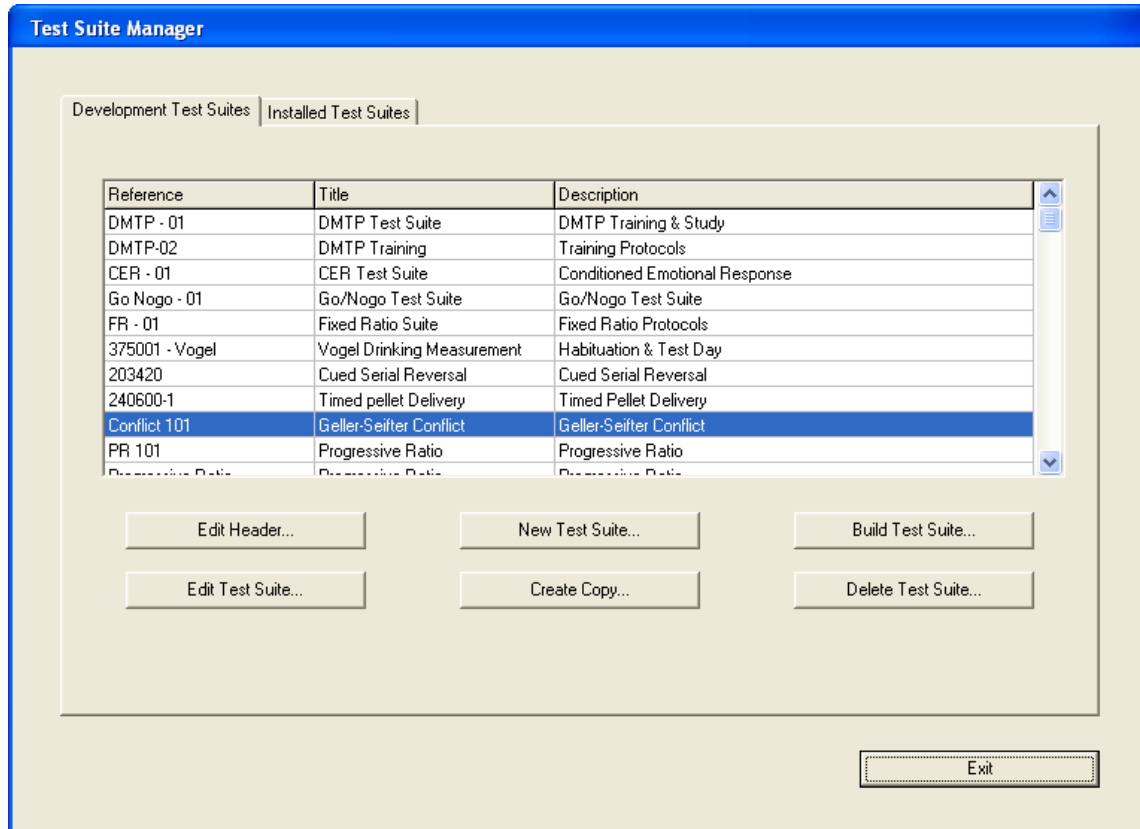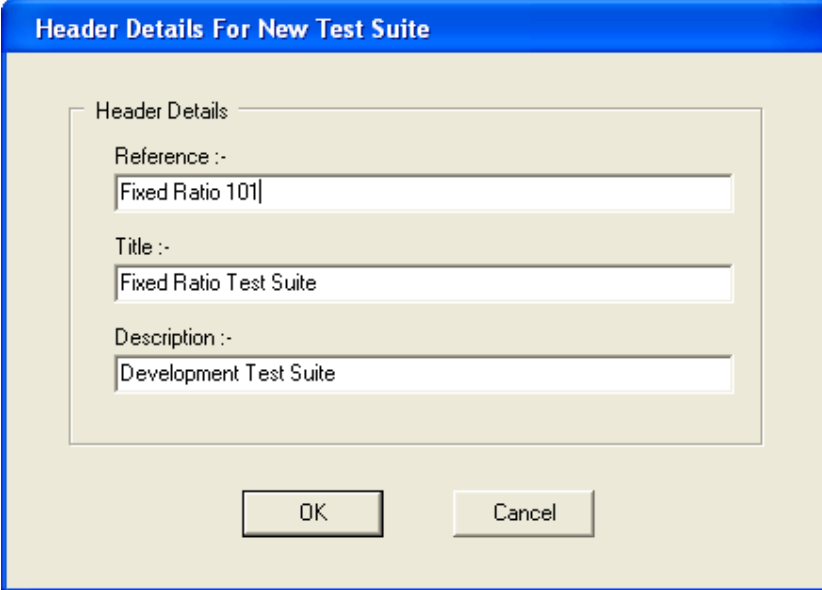This will present the "Test Suite Manager" window.



**Figure 3.9 Test Suite Manager Window**

The window contains two tabbed pages, one page for "Development Test Suites" the other for "Installed Test Suites", select the "Development Test Suite" page.

The "Development Test Suite" page contains a list of the Development Test Suites installed for the current user.  Below the list of Test Suites are a number of buttons used to perform a variety of actions.

Click on the "New Test Suite…" button, this presents a window asking you enter the Reference, Title and Description for the new test suite.  Each of these entries are user defined, so you can enter whatever you like for these entries, although the Reference entry must be unique.  For Development test suites I tend to use a descriptive value, so suggested entries are as follows:

**Figure 3.10 Test Suite Header Details**

Click on the "OK" button and the test suite list updates to display the new Test Suite.

**Launch The "Configure Test Suite" Window**
Select the new "FR 101" test suite and click on the "Edit Test Suite…" button, this presents the
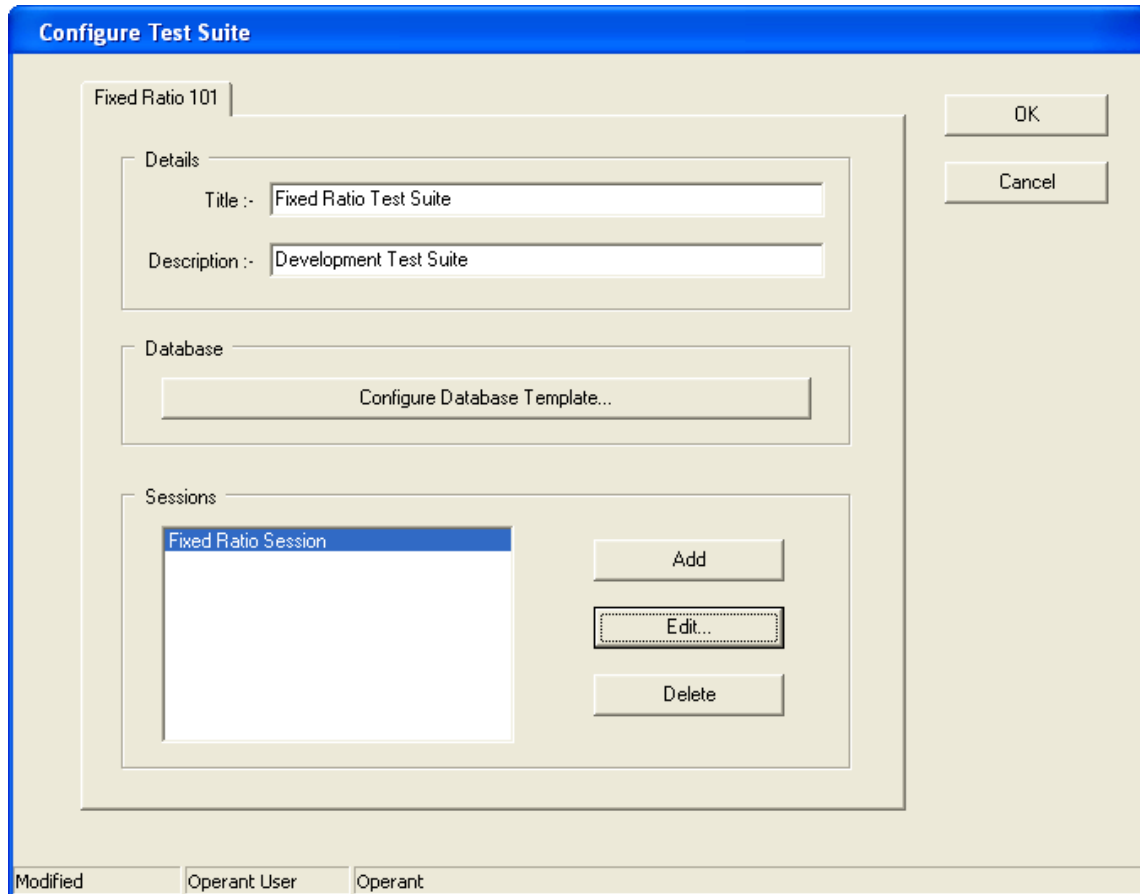"Configure Test Suite" window with the "FR 101" Test Suite loaded.



**<u>Figure 3.11 Configure Test Suite Header Window</u>**
 (Figure 3.11 illustrates the state of the test suite after the sessions have been configured)

The items we need to configure are:
 • The Database Template
 • The Sessions

**The Database Template Explained**
The Database Template will assist us when we need to populate the subject database, (the list of
animals we will use for the experiments).  The Database Template should contain all the database
parameters necessary to run the fixed ratio protocol.  The database template is used when we populate
the subject database later on.  Each time a new subject is added to the database the contents of the
database template are copied into the new subject.  The new subject contains its own independent set
of values, so any changes to the subjects' details will not affect the template and a change to the
template will not change the details already stored in the subjects.  It is possible to copy changes from
the template into one or more subjects as will be discussed later.

Pre-defining the Database Template is not essential, but it is recommended.  The more we pre-define
at this point the easier it becomes to maintain the database later on, especially for experiments where
there is a requirement to train animals to perform the task.  In many cases it is possible to pre-define all
the steps required to train animals from novice level right through to fully trained.

For new users of K-Limbic this sometimes sounds a little daunting, we haven't even started training
animals yet and we should pre-define what is going to happen in several weeks time?  Well, (a)
remember what has happened in the past, training regimes you, (or someone else), have followed

before you will probably use again, pre-defining things here simply helps this process and (b) if your pre-defined values are inappropriate it is very easy to change them.

**Database Parameters**
So what database parameters do we need to run the Fixed Ratio protocol?

All subjects in the database contain a Rewards Table and a Dosing List, plus as discussed during the protocol development section the fixed ratio protocol requires up to four more database parameters.

**Rewards Table**
> The rewards table for the FR protocol requires a single entry; the entry specifies the pulse duration for the pellet dispenser, set this value to 100mS.

**Dosing List**
> Set the dosing list to be appropriate for training the animals, i.e. a single "No Dose" entry. It is possible to pre-define the dosing list, but probably not worth doing so here. If the Dosing List contains a single entry, then that entry automatically becomes the value that will appear in the output data file, (default entry). If the Dosing List contains a default entry K-Limbic will not present the Dosing List to the operator during the session load process.

**FR Parameters**
> FR parameters specify a set of database parameters.
> FR parameters may be present in the database as either a single FR entry or as a table of FR entries. Using a table of entries is the preferred method. The FR stage of the protocol uses the FR settings.

**Selected Lever Parameters**
> Using "Selected Lever" parameters provides a simple method of assigning a "Target" lever to a subject. During the fixed ratio protocol a single lever is presented to the animal. The "Selected Lever" parameters determine whether this is the left or right lever. It is recommended that the database be arranged so that in any given chamber the left lever is used half the time and the right lever used for the other half, this is easy to organise by suitable manipulation of the Selected Lever settings.

**Endpoints – Timeout Table**
> Define an Endpoint – Timeout Table with a single entry, the time set in this entry will be the maximum duration of the FR protocol, (20 minutes).

**Endpoints – Counts Table**
> Define an Endpoint – Counts Table with a single entry, set the "Trials All" value in this entry, this will be the maximum number of trials in the FR protocol, since one pellet is delivered per trial this also controls the number of pellets delivered, (100).

**Configure The Database Template**
We will now configure the database template. The K-Limbic repository contains database files that you can import into the test suite, however here we will generate the template details manually.
In the **"Configure Test Suite"** window, click on the "Configure Database Template" button, this presents the "Edit Database Template" window.

The "Edit Database Template" window displays the current details contained in the database template, initially the Parameters section will display that the parameters list is empty.

Configure the Rewards: click on the "Edit Rewards" button. The default rewards list contains two entries, one for a pellet dispenser and one as a trigger for the foot shock. Click on the "Foot Shock" entry to select it and then click on the "Delete" button to delete it. Click on the "Dispense Pellet" entry and click on the "Edit" button to display the "Edit Reward Table". The table has a single entry, set by default to 100mS, this is a suitable duration pulse to trigger the pellet dispenser, click on "OK" to close the "Edit Reward Table" window and "OK" to close the "Edit Rewards" table.

Configure the Dosing List: click on the "Edit Dosing List" button. The default dosing list is empty, we want to add a single "No Dose" entry to the list. Click on the "Add" button to add an entry to the dosing list. Click on the "Edit" button to display the "Edit Dose" window. Change the entries to those shown below…

**Figure 3.12 Configure Test Suite Header Window**

The precise meaning of each entry is unimportant here, (the settings are discussed in the description documents for the automated analyses). Click on the "OK" button to close the window.

A dosing list with a single entry will automatically assign the entry as the "default" entry. During the process to load a session the protocol manager will interrogate the database to determine which dose has been given to a subject, if the list contains a default entry then the default dose will automatically be assigned to the subject. Obviously during training you will not administer a drug to the subjects so having a default "No Dose" entry in the dosing list is useful.
Click on the "OK" button to return to the "Edit Database Template" window.

Configure the FR Parameters
The fixed ratio protocol requires the database to define a value for the fixed ratio parameters. The database contains several options for providing FR parameters, any of which may be used, e.g. a single FR Parameters entry, a table of FR entries, a combined VI/FR entry or a table of VI/FR entries. The most useful of these for the fixed ratio protocol is the table of FR entries.

A table of FR entries normally defines a list of fixed ratios of gradually increasing difficulty, as the subject gains experience of the task the operator can assign more difficult ratios up to the maximum value required for the task.

Click on the "Add" button and in the "Select Parameter Set" list select the "FR Table" entry, then click on the "OK" button. An FR Table parameter is added to the parameter list.
Click on the "Edit" button to display the "Edit Parameter Table" window, the initial FR Table has a single entry, click on the "Edit" button to display the FR parameters.

We will set this entry to define a fixed ratio that creates an FR10 on a single lever, (the target lever). The target lever will be assigned using a "Selected Lever" entry in the database.
   1. Change the "Comment" entry to a meaningful value, e.g. FR10
   2. Change the "Ratio" value to 10 using the up/down buttons, (or simply type 10 in the box)
   3. Change the "Target" setting to "Use Selected Lever", (this disables the Dummy setting)
   4. Leave "Enable Rewards" as the only checked item in the "Control Options" section
   5. Click on "OK" to close the window.
Create another entry in the FR Table by clicking on the "Copy" button, then edit the values in this entry using the "Edit" button. Change this to say an FR5 entry and close the window. Repeat this process

until you have sufficient entries in the FR table to take a subject from novice level up to fully trained, for example using FR1, FR2, FR3, FR4, FR5 & FR10 entries, (you may prefer to use different values for the FR entries).  When complete click on the "OK" button to return to the "Edit Database Template" window.
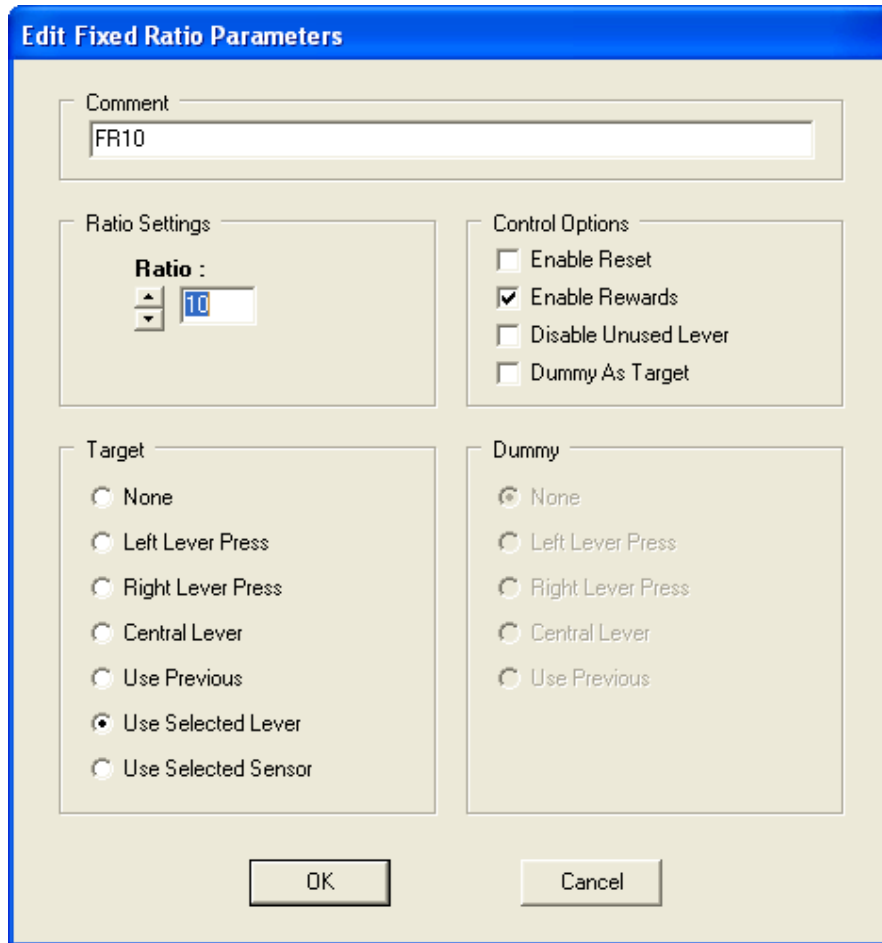


**Figure 3.13 Edit Fixed Ratio Parameters Window**

Configure the Selected Lever Parameters
The FR parameter entries opted to define the target lever using the selected lever parameter entry, this makes it easier to reassign the target lever to each subject.  It is good practise to define a database in which half the subjects use the left lever as the target and the half use the right lever, (assuming your chamber contains two levers).  To reassign the target lever we only need to edit the selected lever parameter entry rather than edit each of the entries in the FR table for each subject.  (The section on populating the subject database later on gives details on how this is done efficiently).

Click on the "Add" button and in the "Select Parameter Set" list select the "Selected Lever" entry, then click on the "OK" button.  A "Selected Lever" parameter is added to the parameter list.
Click on the "Edit" button to display the "Edit Selected Targets" window, the window contains two tabbed pages, one for the Selected Lever and one for the Selected Sensor.  Here we want to assign the selected lever.

In the Target Assignments section set the "Control Target" and "Active Target" settings to "Left Lever Press" and in the Dummy Assignments section set the "Control Dummy" and "Active Dummy" settings to "None", as shown in fig 3.9 below.

The "Control Target" setting is used as the target when a dose type of "Inactive" is assigned to the subject.  The "Active Target" setting is used when a dose type of "Active" is assigned to a subject.  Having dose dependent target/dummy levers is important when performing drug discrimination studies, (which is not important here).

We have set both of these to the same lever so the target lever is independent of the dose type of the assigned dose.  Click on the "OK" button to close the "Edit Selected Targets" window.
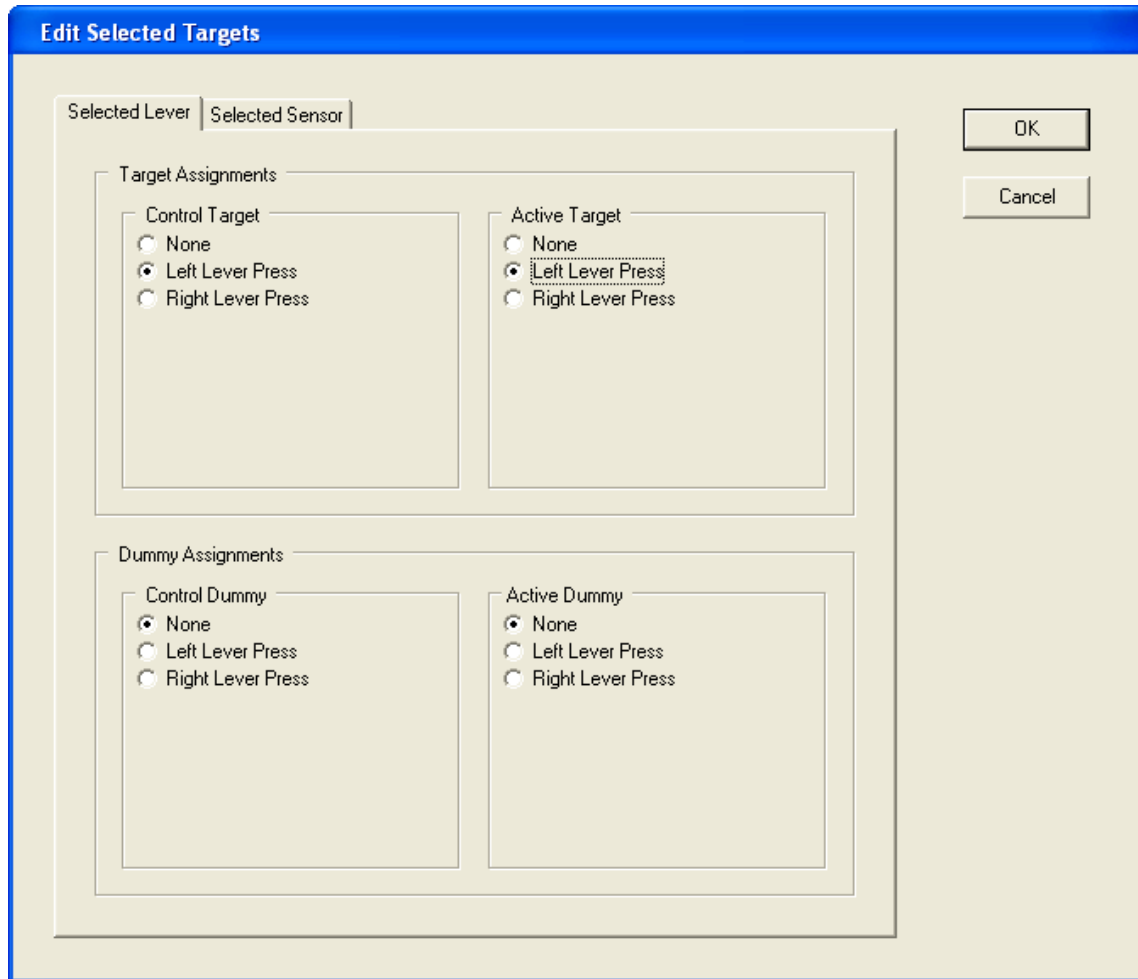


**Figure 3.14 Edit Selected Targets Window**

Configure the Endpoints – Timeout Table
The timeout table is required to set the protocol timeout.  In the "Edit Database Template" window click on the "Add" button, select the "Endpoints-Timeout Table" entry in the list and click on the "OK" button. Click on the "Edit" button to display the (empty), timeout table.
Add one entry to the table by clicking on the "Add" button; then adjust this entry to 20 minutes using the "Edit" button.

Using a timeout table with a single entry means that this entry will be used by the protocol manager by default, if we define a timeout table with more than one entry, then during the loading of a session the entries in the timeout table will be presented to the operator who has to select the appropriate value to use as the timeout for the session.
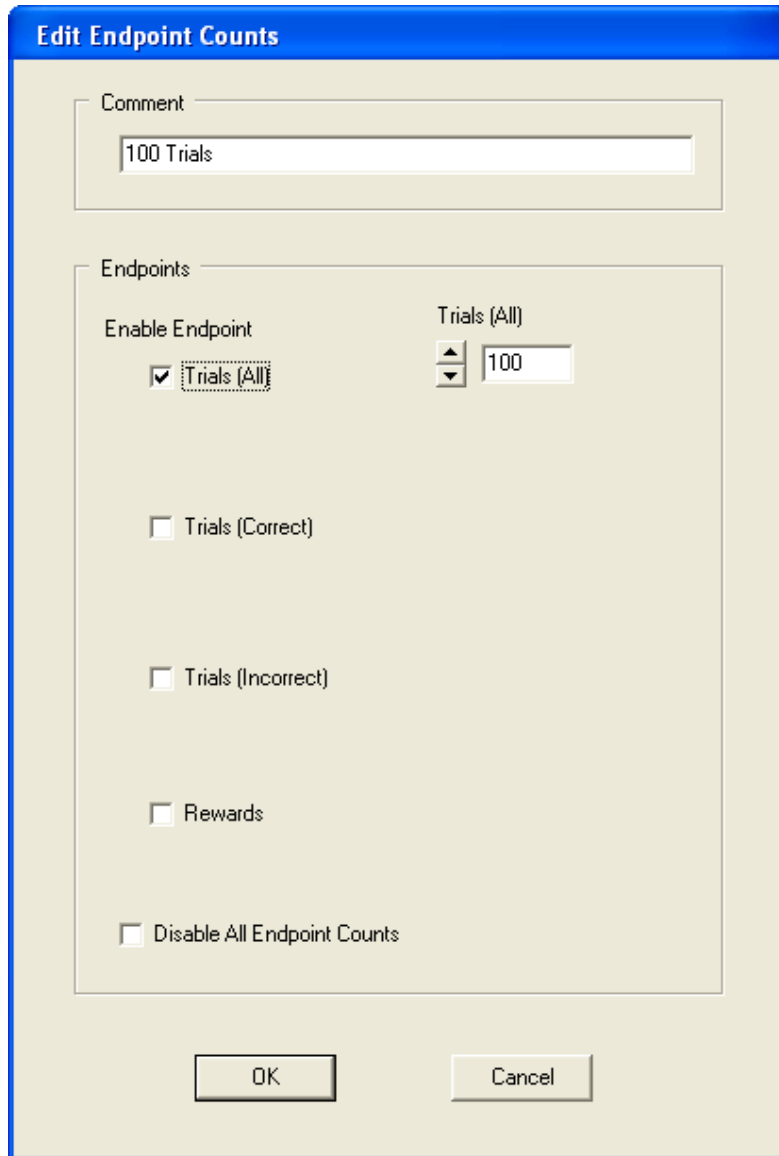
**Figure 3.15 Edit Endpoint Counts Window**

Configure the Endpoints – Counts Table
The counts table is required to set the protocol trials endpoint.  In the "Edit Database Template" window click on the "Add" button, select the "Endpoints-Counts Table" entry in the list and click on the "OK" button.
Click on the "Edit" button to display the (empty), timeout table.
Add one entry to the table by clicking on the "Add" button; then adjust this entry using the "Edit" button. Enable the "Trials (All)" endpoint and set the value to the maximum number of pellets you wish to deliver during one session, (1 pellet per trial), figure 3.10 illustrates this procedure using a maximum of 100 trials.  It is feasible to enable more than one count endpoint but for the fixed ratio protocol there is no point in doing so.

Using a table with a single entry means that this entry will be used by the protocol manager by default, if we define a table with more than one entry, then during the loading of a session the entries in the table will be presented to the operator who has to select the appropriate value to use as the endpoint for the session.

This completes the configuration of the database template.

Click on the "OK" button to close the "Edit Endpoint Counts" window and then on "OK" several more times to return to the "Configure Test Suite" window.

**Configure The Sessions**
The Fixed Ratio Test Suite requires a session to run the fixed ratio protocol
In the "Sessions" section of the "Configure Test Suite" window, click on the "Add" button, this will add
an empty session to the Test Suite, the session is titled "Session 1".

**Fixed Ratio Session**
We will set up all boxes in the session to run the fixed ratio protocol…

Select "Session 1" in the Sessions list and click on the "Edit…" button, this presents the "Configure
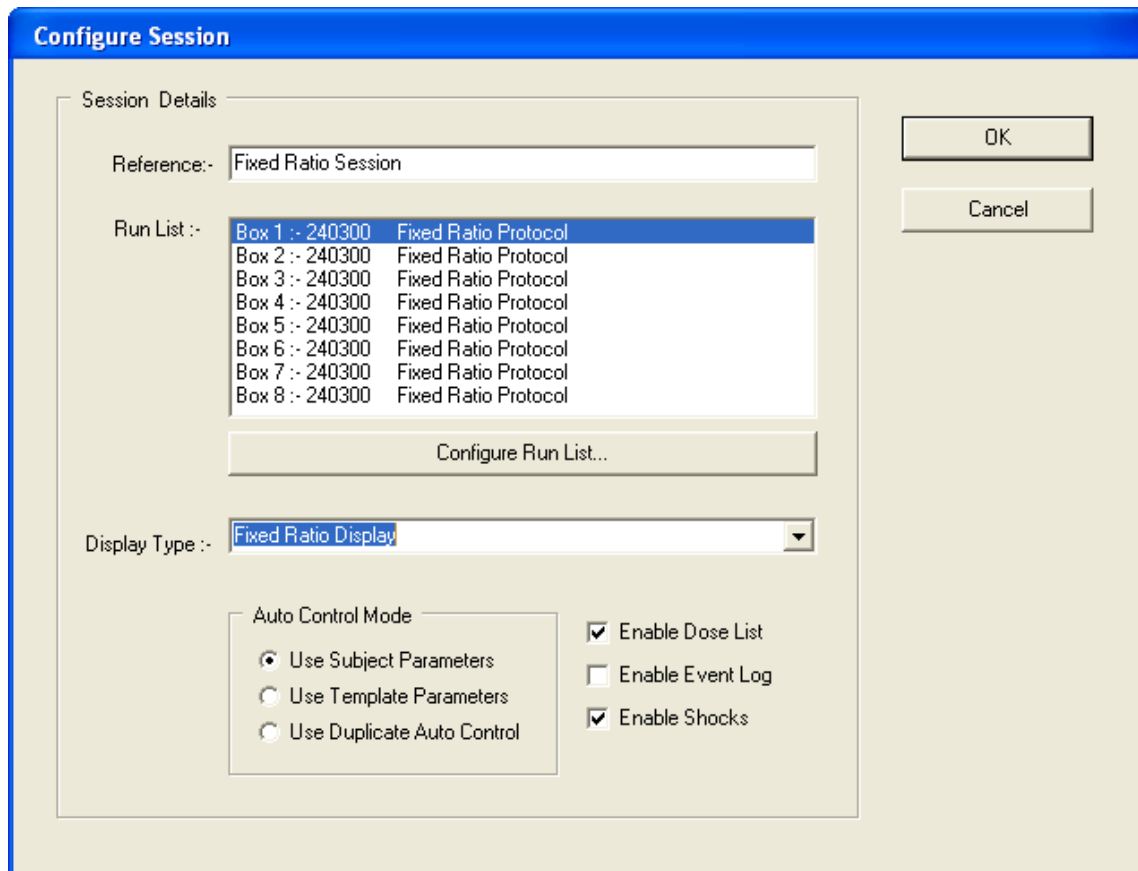Session" window. We will configure this Session as the fixed ratio session.



**Figure 3.16 Configure Session Window**

Modify the "Reference" entry to a meaningful value, e.g. "Fixed Ratio Session"

Click on the "Configure Run List" button to present the "Configure Run List" window, (this is where we
tell K-Limbic how many boxes to run and which protocol to run in which box).
Click the "Add Box" button once: this adds an empty box to the run list.
Select the Fixed Ratio protocol, reference 240300, in the "Protocols" list and click on the "Assign
Protocol To Box" button: this configures Box 1 to run the Fixed Ratio protocol.
Click on the "Add Box" button again, this will add another box to the Run List, by default this box will run
the same protocol as Box 1, which is what we want.
Click again on "Add Box" until the Run List contains the correct number of boxes, all running the Fixed
Ratio protocol, you need one entry in the Run List for each box in your system.
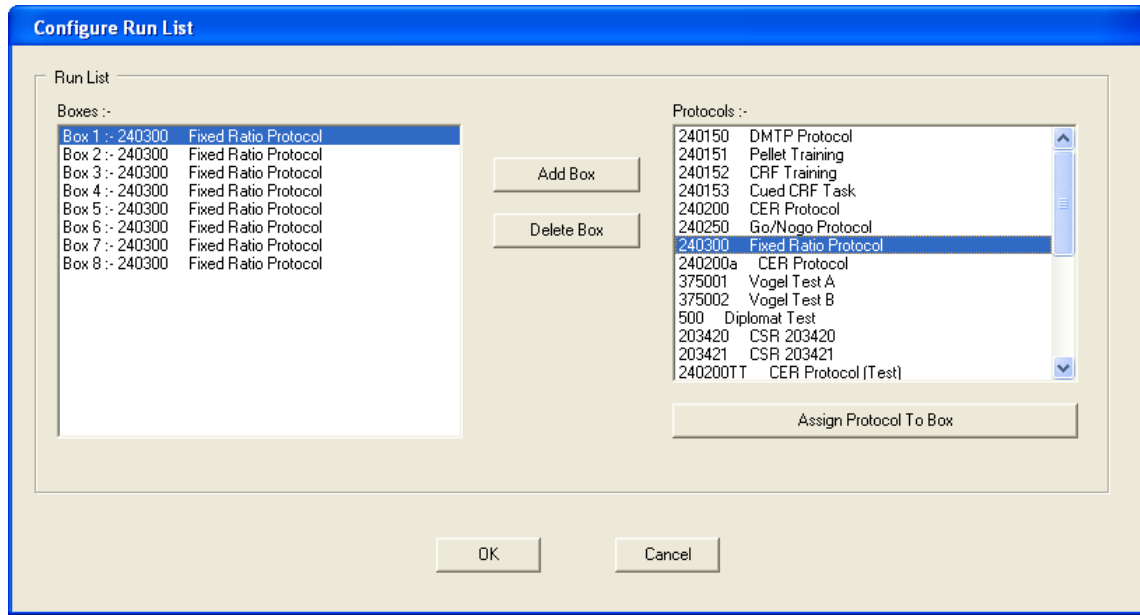
**Figure 3.17 Configure Run List Window**

When the Run List is complete, click on the "OK" button to return to the "Configure Session" window.

Back in the Configure Session window…
Set the "Display Type" to a suitable value, there is a specific display for fixed ratio protocols; select the "Fixed Ratio Display".
"Auto Control Mode" should remain as "Use Subject Parameters".
"Enable Dose List" should be checked.
"Enable Event Log" should be unchecked.
"Enable Shocks" should be checked, (although no shocks are delivered during the protocol).

This completes the configuration of the Fixed Ratio Session, click on the "OK" button to return to the "Configure Test Suite" window.

This completes the configuration of the Fixed Ratio 101 Development Test Suite.

Click on the "OK" button to return to the Test Suite Manager.

Click on the "Exit" button to return to the K-Limbic main screen. (This automatically saves the configuration of the Fixed Ratio 101 Test Suite.

**3.4 Build An Installed Test Suite**
Test Suites are created, edited and managed using the "Test Suite Manager".
From the K-Limbic main menu select…
Test Suites / Test Suite Manager…          (or press function key F4)
This will present the "Test Suite Manager" window.

Under the "Development Test Suite" tab: select the "Fixed Ratio 101" Development Test Suite.
We have previously configured this test suite, now we can use this to build as many "Installed Test Suites" as we need.

The organisation of Installed Test Suites is up to the individual user, however Conclusive recommend a one-to-one relationship between "Installed Test Suites" and groups of animals.  For instance, if your lab will run one squad of animals in the mornings and a different squad of animals in the afternoons then it is a good idea to have two "Installed Test Suites", one for each squad, but it is up to the user.  Of course, over time you will probably develop several protocols and hence have a collection of test suites.

With the "Fixed Ratio 101" development test suite selected: click on the "Build Test Suite" button, this presents the "Build Test Suite" window.  The header details of the Fixed Ratio 101 development test suite are shown in the top section of the window and any installed test suites are listed in the lower section.  If this is a new installation of K-Limbic this list will initially be empty.
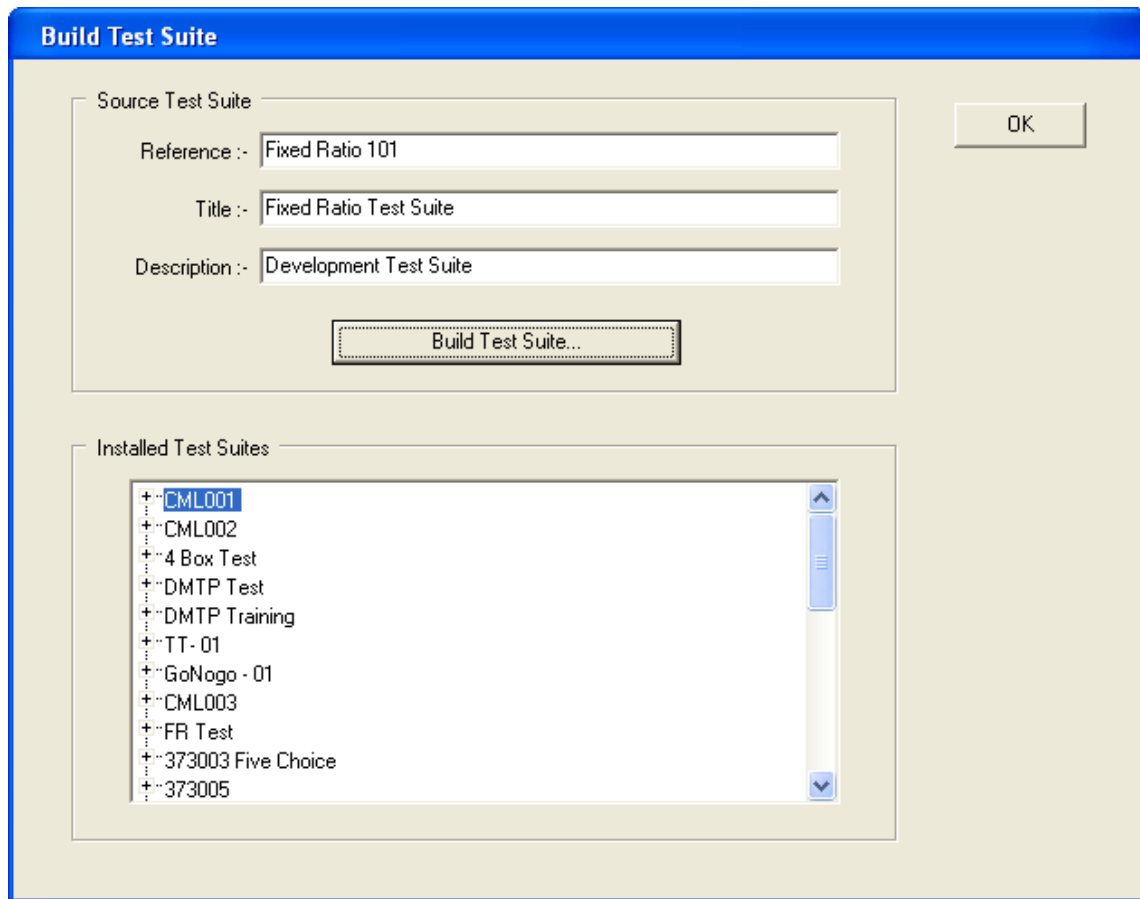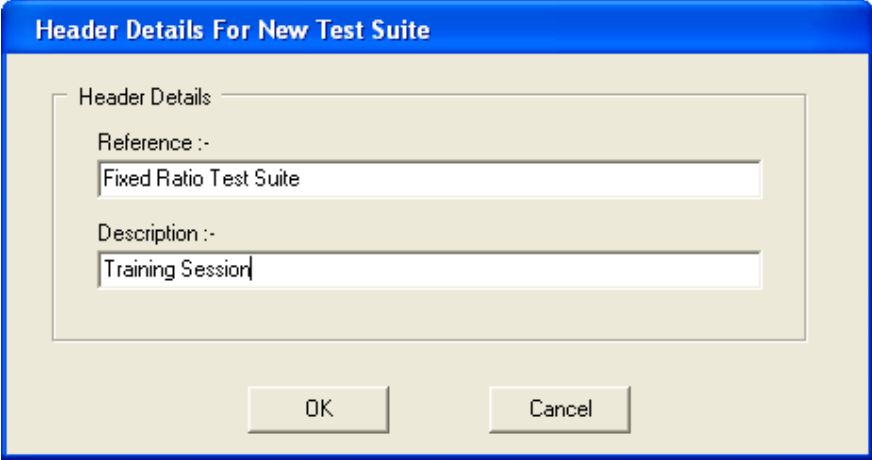


**Figure 3.18 Build Test Suite Window**

Click on the "Build Test Suite" button.

A window appears prompting for header details of the new "Installed Test Suite", the entries here are up to the user, but the following entries are suggested:



**Figure 3.19 New Test Suite Details Window**

These values are not cast in stone; you can change them later if desired.
Click on the "OK" button, this opens what is called the "Repository Manager", which builds the Installed Test Suite.  If everything builds successfully then no errors are reported: click on the "OK" button to close the Repository Manager, then click on "OK" to close the information window that appears.  The "Fixed Ratio Test Suite" appears in the list of "Installed Test Suites".
Click on the "OK" button to close the "Build Test Suite" window.
Click on the "Exit" button to close the Test Suite manager.
The User Details section of the main screen will update to show the "Fixed Ratio Test Suite", with its two sessions, in the "Installed Test Suites" list.

**3.5 Prepare The Test Suite For Use**
The final stage of configuration before we are ready to run an experiment is to prepare the newly installed test suite. Once this is complete the Sessions in the Installed Test Suite will be ready to use for experiments.
Preparing the Installed Test Suite for use consists of the following steps:
- Populate The Subject Database
- Configure The Sessions

To prepare an Installed Test Suite we must first load the it into K-Limbic:

From the K-Limbic main menu select…
File / Load Test Suite…          (Or press function key F2)
This will present the "Load Test Suite" window; the window contains a list of all the Installed Test Suites available to the current user.

Select the appropriate Test Suite in the list and click on the "OK" button.
The "Test Suite" section of the main screen will update with the details of the new test suite.

**Populate The Subject Database**
In this section I will describe how to populate the subject database, i.e. add in the animals that will be used with this test suite.
From the K-Limbic main menu select…
Test Suites / Database Maintenance…
This will present the "Database Maintenance" window. The "Database Maintenance" window presents all databases defined for the Test Suite, (at present the Fixed Ratio Test Suite contains just the default empty database).
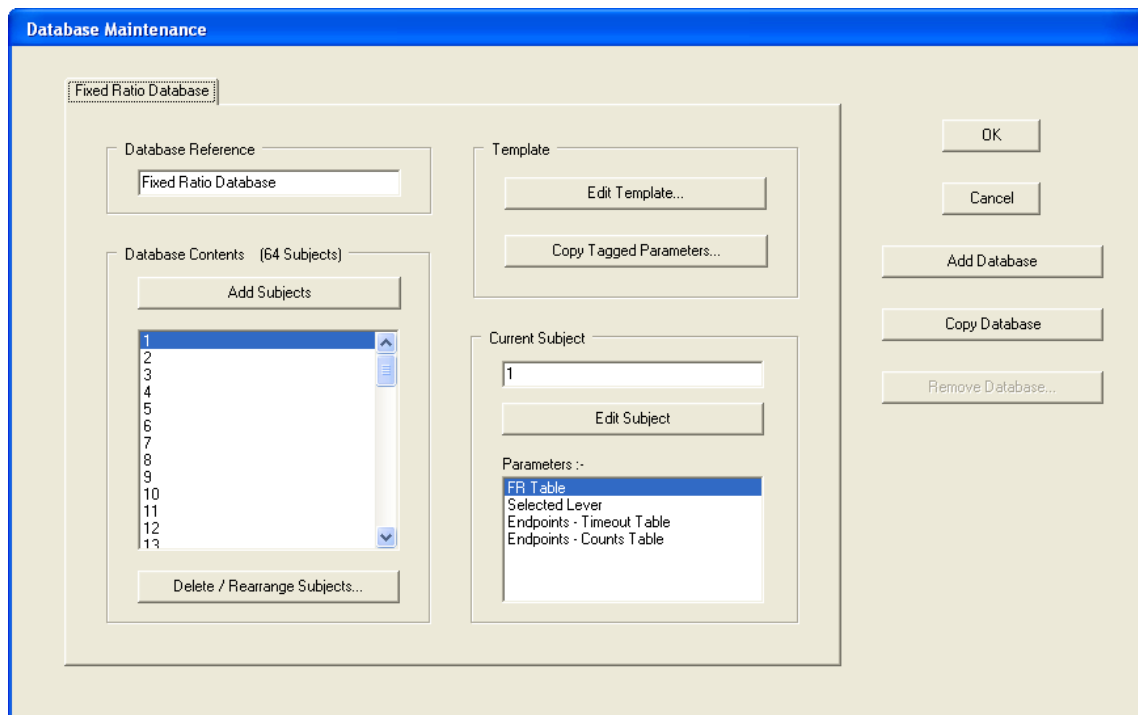


**Figure 3.20 Database Maintenance Window**
(Figure 3.20 illustrates the database after it has been populated).

Assuming that the instructions on how to "Configure The Database Template" were followed, the Database Template was set-up in the Development Test Suite.  The values in the Development Test Suite were copied into the Database Template in the installed Fixed Ratio Test Suite.  If you did not configure the Database Template in the Development Test Suite you should configure the Database Template in the Fixed Ratio Test Suite before proceeding, click on the "Edit Template" button and follow the instructions in the "Configure The Database Template" section above.

In the "Database Maintenance" window enter a meaningful value for the database reference, this may be an identifier for your squad of animals, if there is one, otherwise I suggest "Fixed Ratio Database" as the reference for this database.
Next we need to add in all the subjects in our squad of animals: click on the "Add Subjects" button, this presents the "Add Subjects" window.

We can either add and name subjects individually, using the "Add Subject…" button in the upper section, or we can add all our subjects in a single operation using the lower section, (it is possible to edit individual subjects later if required).  Adding subjects individually is recommended if you name your subjects with individual names, e.g. John, Paul, George & Ringo.  Adding subjects automatically is recommended if you name your subjects numerically, e.g. 1, 2, 3, 4 etc.

In the lower section the "prefix" and "suffix" entries allow you to customise the automatic names, e.g. A1, A2, A3, A4 etc or
1D, 2D, 3D, 4D etc, or even A1F, A2F, A3F, A4F etc, simply set-up the prefix and suffix values as required.

Set the "Subjects To Add" value to the number of animals in your squad, i.e. 64, check that the "Use Template" option in the lower section is checked then click on the "Add Subjects" button, this will create a database of 64 animals, each with their own set of parameters equal to those present in the Database Template[++].

[++]*The values in the Database Template are copied into the individual subjects as part of the creation process, these parameter values are now stored in each subject.  If you now make a change to the Database Template the changes will not automatically be copied into the subjects, more on this next.*
Click on the "OK" button to return to the "Database Maintenance" window.

**Adjusting The Database Parameters**
(This section is optional)
Those of you who have followed things so far might be thinking, 'Hey, what about the "Target" lever? If the "Selected Lever" parameters in all subjects are the same then all subjects will use the same Target lever, I want half my animals to use the Left lever as the target and half to use the right. What should I do?'

You have two options here, you can:
- Manually edit half the subjects to set the appropriate Target lever, (very boring!) or
- Edit the Database Template and copy the new values into half the animals, (much better!)

To make changes to several subjects at once, we use the "Copy Tagged Parameters" utility. The procedure here is to make a change to one or more parameters in the Template, "tag" the modified parameters, and then copy the "tagged" parameters into a list of selected subjects.

Click on the "Copy Tagged Parameters" button, to present the "Copy Tagged Parameters" window.
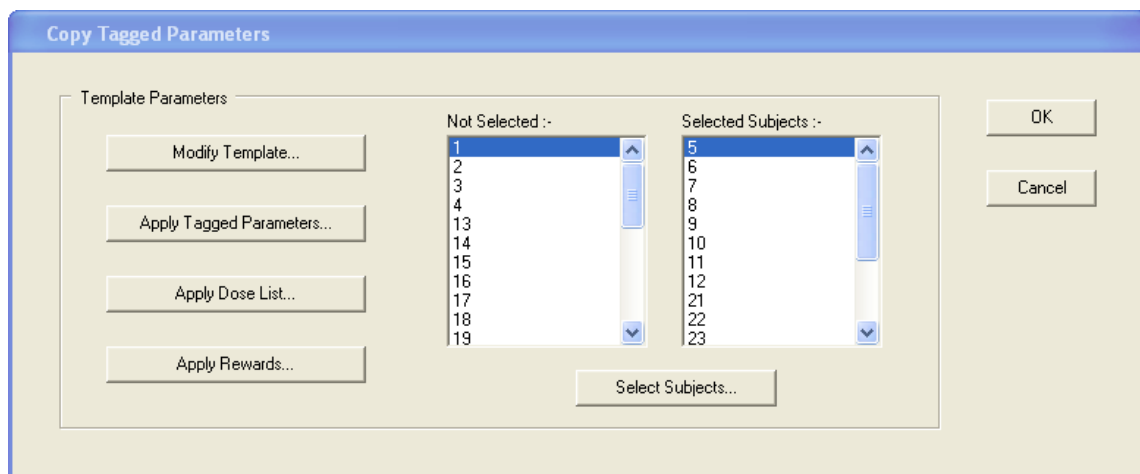


**Figure 3.21 Copy Tagged Parameters Window**
 (Shown with subjects selected)

Click on the "Modify Template" button.
Select the "Selected Lever" entry in the "Parameters" list and click on the "Edit" button.
This will present the "Edit Selected Targets" window.
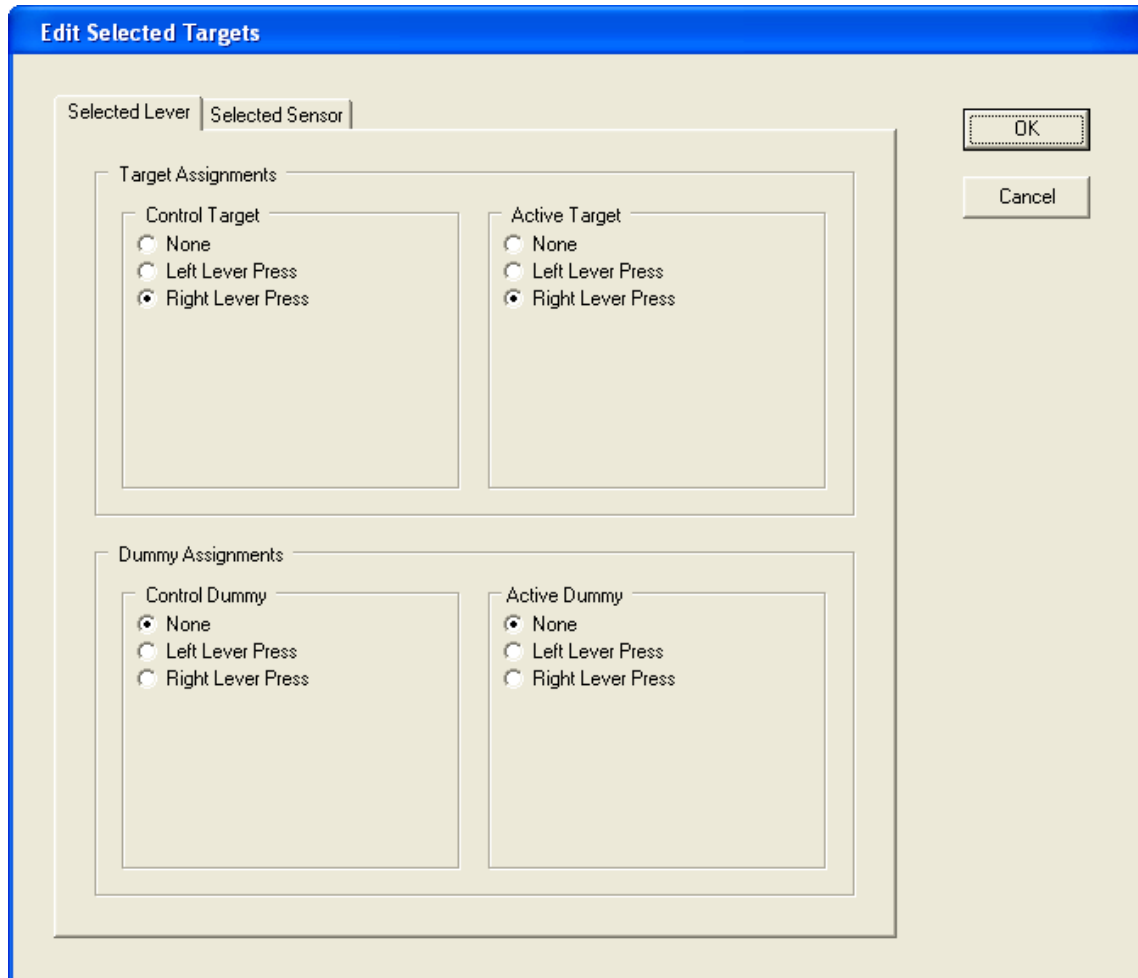
**Figure 3.22 Edit Selected Targets Window**

In the "Target Assignments" section select the "Right Lever Press" option for both the "Control Target" and "Active Target" settings[++].
Leave the "Dummy Assignments" as "None".
Click on the "OK" button to return to the "Edit Database Template" window.

[++]*The settings in my version of the "Edit Selected Targets" window are not the same as shown above, what is going on?  The options available in this window, i.e. "None", "Left Lever Press" & "Right Lever Press" are taken from the resource list stored in the active profile of the current user, (this is different from the resource list that was configured during the installation).  The default user contains an operant profile and a five choice profile, each with their own resource list.  If the currently active profile is the "Operant" profile then the window should appear as shown, however if the Five Choice profile is active there will only be the option "None" in the "Edit Selected Targets" window.  You will need to change the active user profile, see section 3.1.  The current active user profile is displayed in the status bar at the bottom of the K-Limbic main screen; the status bar displays the "User", "Group" and "Profile" names, (in that order).*

With the "Selected Lever" entry selected in the parameters list, click on the "Tag" button, an arrow will appear next to the "Selected Lever" parameters showing that it is currently tagged, (successive clicking on the Tag button toggles the tag on and off).

Click on the "OK" button to return to the "Copy Tagged Parameters" window.

Click on the "Select Subjects" button, this presents the "Select Subjects" window.

We want to select half the subjects in the database so that the levers are used equally in each box, the way to do this depends on how many boxes are used in the system.  The table below assumes that the system consists of 8 boxes and that subjects are used sequentially, i.e. animals 1-8 go in boxes 1-8, animals 9-16 go in boxes 1-8 etc.

| Subjects | Target Lever |
|----------|--------------|
| 1-4      | Left         |
| 5-12     | Right        |
| 13-20    | Left         |
| 21-28    | Right        |
| 29-36    | Left         |
| 37-44    | Right        |
| 45-52    | Left         |
| 53-60    | Right        |
| 61-64    | Left         |

We want to change all subjects who should be assigned the right lever as the target, therefore select subjects 5-12, 21-28, 37-44 & 53-60.  To select a subject, highlight it in the "Available" list and click on the "Select >>" button, the subject moves over to the "Selected" list, repeat this for all the required subjects.  When all required subjects are selected, click on the "OK" button.

Click on the "Apply Tagged Parameters" button; confirm by clicking on "Yes" and the modified Selected Lever settings are copied into the selected subjects.
Click on the "OK" button to return to the "Database Maintenance" window.
The database is now complete.
Click on the "OK" button to close the "Database Maintenance" window.

**Configure The Fixed Ratio Session**
The majority of the session configuration was performed when we created the Development Test Suite.
All we need to do now is to configure the data output from the session.

From the K-Limbic main menu select…
Test Suites / Configure Sessions…
This will present the "Configure Run-Time Sessions" window.
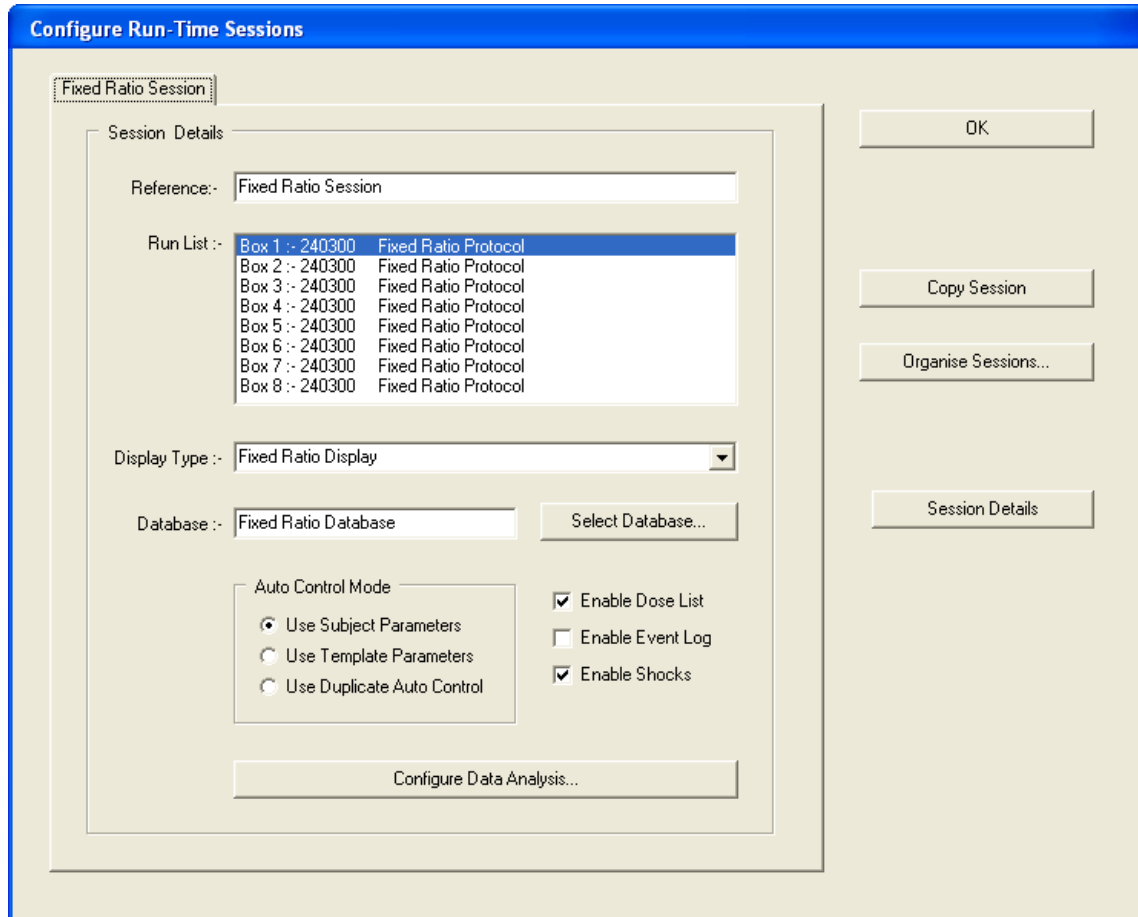


**Figure 3.23 Configure Run-Time Sessions Window**

The window contains a tab for each of the sessions in the Fixed Ratio Test Suite.  The test suite
includes a single tab for the "Fixed Ratio Session".
The options on the main page were set as part of the Development Test Suite, (although they can be
changed here); we do not want to change any of these settings.

Click on the "Configure Data Analysis…" this presents the "Configure Output" window.

Click on the "Enable Data Output" option so that it is checked.
Leave the "Enable Activity Log" option unchecked.
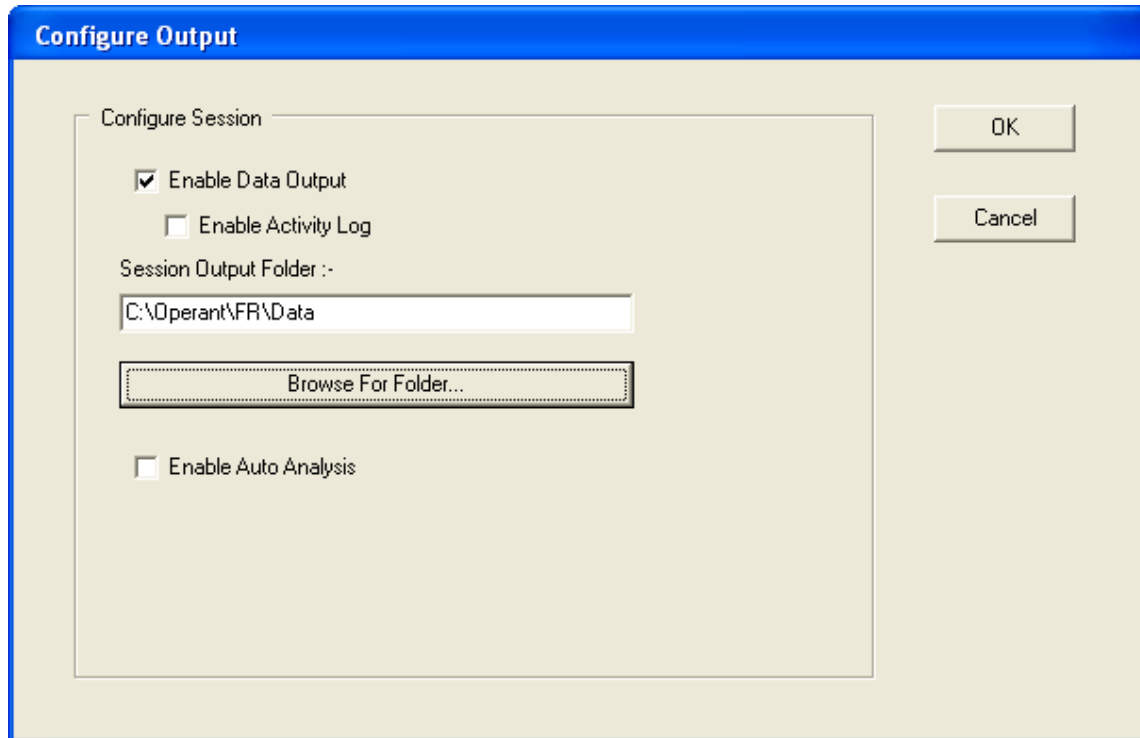
**Figure 3.24 Configure Run-Time Sessions Window**

Click on the "Browse For Folder…" button, this presents the "Select Folder – Session Data" window. Here we need to navigate to the appropriate folder where we want to store the output data files generated by the Fixed Ratio session. (A new folder can be created in this window if required). It is up to the user to determine where to store the data. Network drives are permitted but not recommended. A suggested location is illustrated in figure 3.15.

Navigate to, (or create), the desired folder location and click on the "OK" button.
The selected output data folder appears in the "Session Output Folder" box.
Click on the "OK" button to return to the "Configure Run-Time Sessions" window.
Click on the "OK" button to return to the K-Limbic main screen.

From the K-Limbic main menu select…
File / Close Test Suite
The settings in the Fixed Ratio Test Suite are saved.

**Congratulations!!!**
You have completed the full development and configuration of a K-Limbic experiment, and not just a simple, "Hello World!" style example experiment, the Fixed Ratio Test Suite is fully operational, and usable for live experiments.

**Automated Data Analysis**
Once again the keen-eyed observer will have noticed an omission. "What about the automated connection to the data analysis?" The configuration of the automated data analysis has been omitted from this section for brevity.

Conclusive recommend the use of automated data analysis packages as they significantly reduce the daily effort required to analyse your data, they are a leap ahead in the field of data analysis for operant experiments.

If you have purchased an analysis package for the Fixed Ratio task, you must configure the session to connect to the analysis package. Instructions on how to do this are given in the analysis package user guide.

**Important Note**
It is now necessary to exit and restart K-Limbic, (prior to running a first session with K-Limbic).

In some cases running a session from a test suite that has just been created causes a problem.  The operator is able to load a session and run it without any problems, however K-Limbic "freezes" when the operator selects the command to close down the session.

This error occurs on some machines and not others and only affects closing down the session the first time a session is run in a newly built test suite.  It does not happen at any other time.  The simple method of never encountering this problem is to exit K-Limbic and restart the application.

Conclusive Marketing Ltd apologise for this bug, we will remove the problem as soon as possible.

**3.6 Check The Chamber Definition**

Prior to loading the session for the first time it is a good idea to check the chamber resources…

The protocol defines a list of resources that it expects to control in the chamber, (see section 3.2), part of the process of loading a session includes checking that all resources in the protocol have corresponding entries in the "Chamber Definition".  The "Chamber Definition" informs the system of which channel each resource uses in the chamber.

To check that the chamber definition is correct:
From the K-Limbic main menu select…
Configure / Define Chamber…
This will present the "Display Chamber Resources" window as shown below.
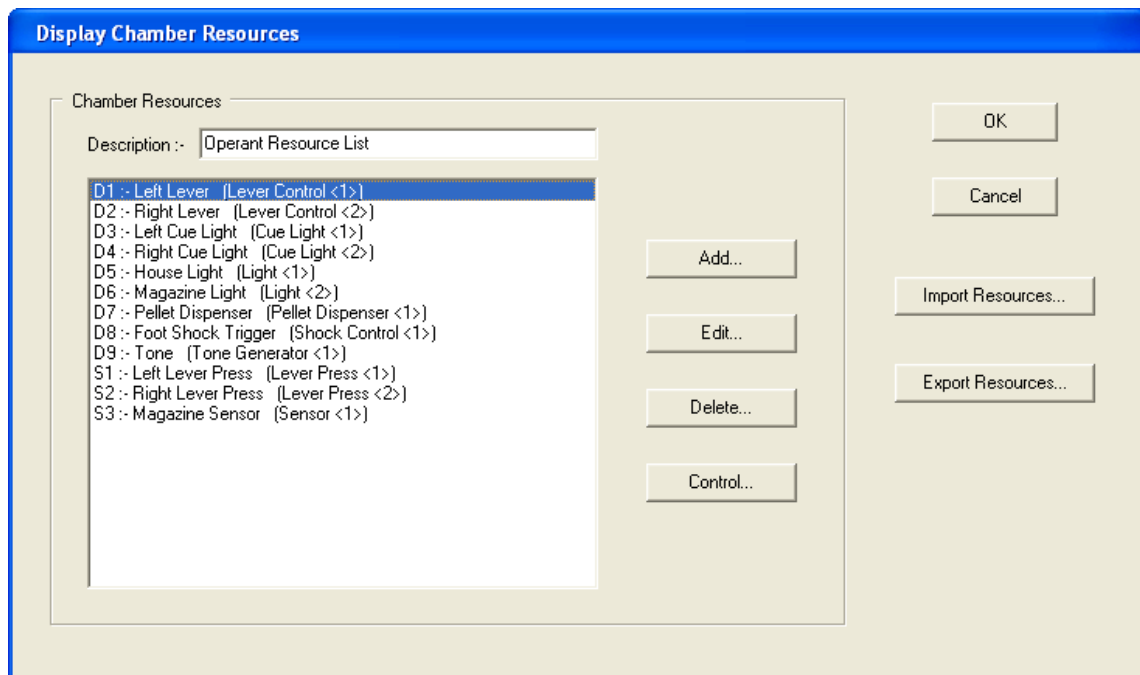


**Figure 3.25 Display Chamber Resources Window**

Check that each resource defined in the protocol has a corresponding entry in the "Display Chamber Resources" window.  An example is shown in figure 3.25.

If you a session defines a resource that is not present in the chamber definition the protocol manager will not be able to assign a channel to the missing resource, this will cause the load session operation to abort and report error 202 "Unconfigured Resource".  If you see this error when loading your session please check the chamber definition carefully to determine which resource is missing.

**3.7 How To Load & Run A K-Limbic Session**
Once the sessions in a test suite have been fully configured the system is ready to run experiments, unless your requirements change you will not need to reconfigure anything in the test suite.

The procedure to load and run sessions is quick and simple:
1. Load the Test Suite
2. Load the Session
3. Run the Session

**Load The Test Suite**
From the K-Limbic main menu select…
File / Load Test Suite…        (or press function key F2)
This will present the "Load Test Suite" window; the window contains a list of all the Installed Test Suites available to the current user.
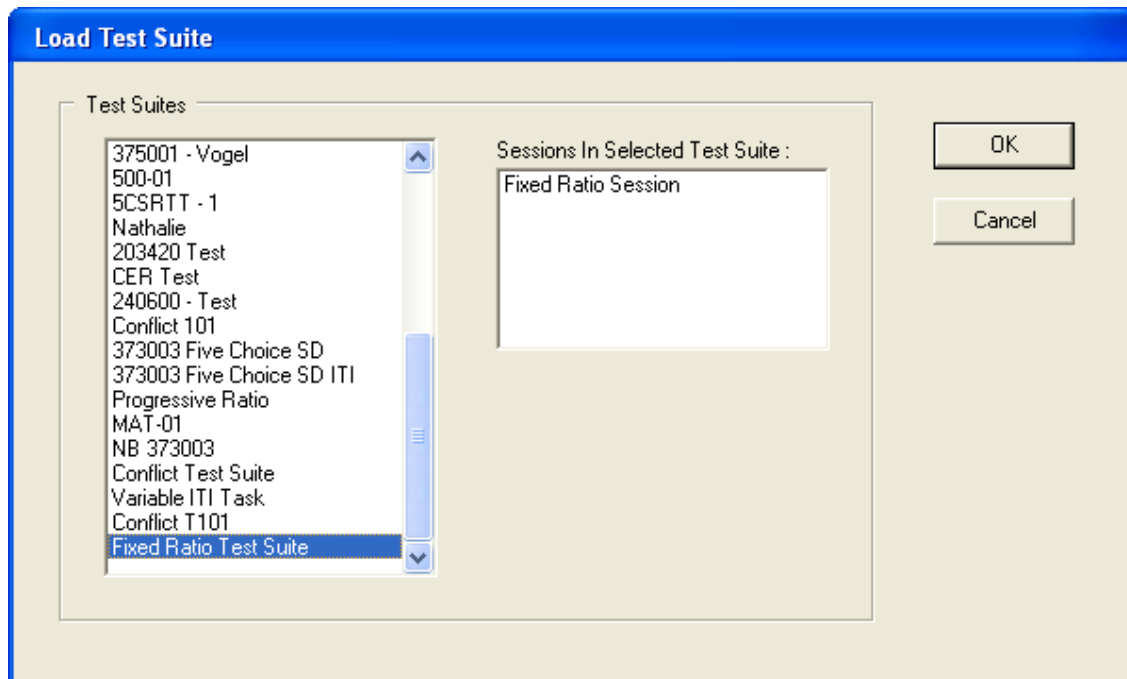


**Figure 3.26 Load Test Suite Window**

Select the "Fixed Ratio Test Suite" and click on "OK".
K-Limbic loads the Fixed Ratio Test Suite.

**Load The Fixed Ratio Session**
From the K-Limbic main menu select…
Session / Load Session…
This will present the "Select Session" window. The window contains a list of the sessions in the Fixed Ratio Test Suite, see figure 3.26.

Select the "Fixed Ratio Session" and click on "OK", see figure 3.27.

K-Limbic is now ready to assign subjects to chambers.

The "Database - Select Subject" window appears, it is prompting you to select the subject that will be placed in Box 1, by default the first subject in the list is selected.
K-Limbic remembers the assignment of subjects to chambers and will automatically select the most appropriate subject next time around.
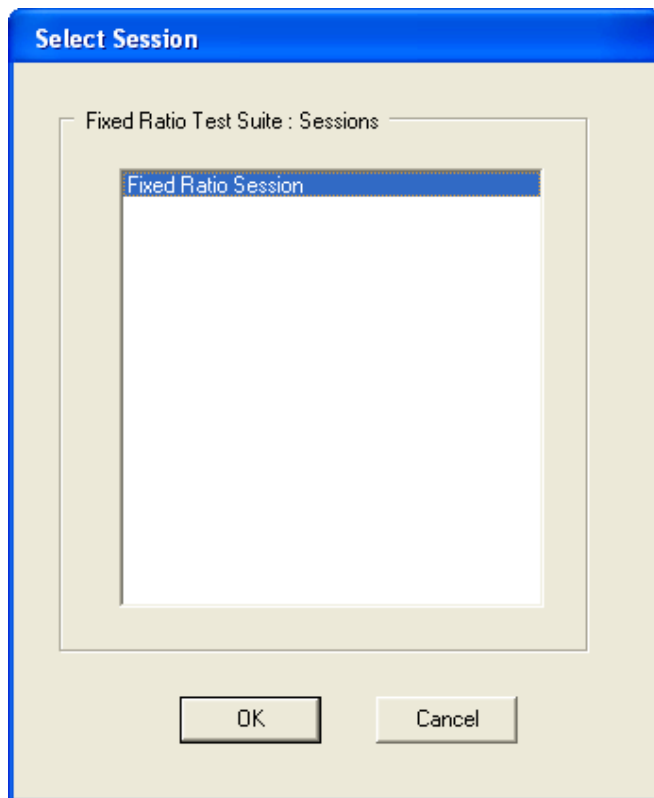
**Figure 3.27 Select Session Window**
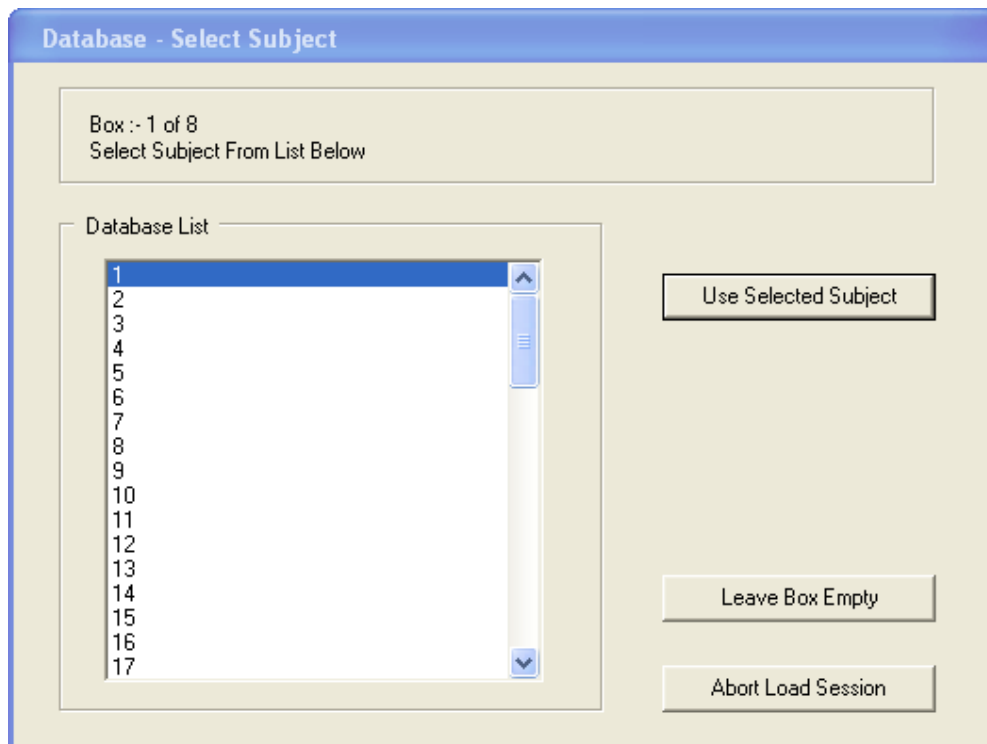



**Figure 3.28 Select Subject Window**

Select the appropriate subject in the list.

Click on the "Use Selected Subject" button.

Next: - the "Select Database Parameter" window appears.  It is prompting you to select the entry in the FR table that is required for the current experiment.  The protocol will use the FR value in whichever entry is selected in the table.  Select the required entry in the table and click on "OK".  If an entry with an FR setting of 1 is selected then the protocol becomes a continuous reinforcement type protocol.

The prompt repeats for the next box and so on until all boxes in the session have been assigned a subject and an entry from the subjects FR table, (the selection of FR entry may be different in each box).
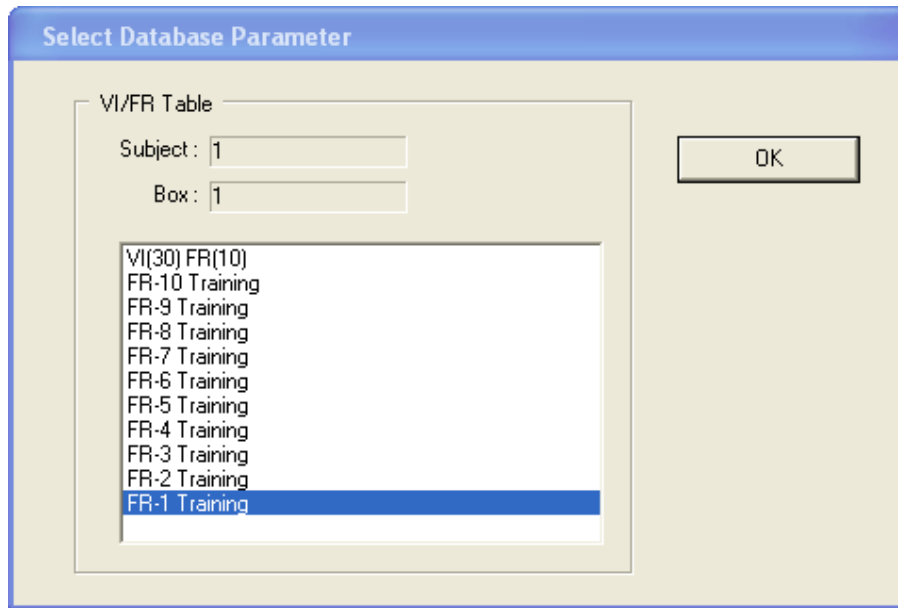


**Figure 3.29 Select Database Parameter**

The selected subject is assigned to Box 1 and the prompt moves on to the next box and so on until all boxes in the session have been assigned a subject.

It is possible to leave a box empty: simply click on the "Leave Box Empty" button.

K-Limbic now prepares the protocols to run in each box, this includes setting the state of the operant chambers to the state defined in the "Start" stage of the protocol, when this process is complete an information window appears telling you the session is ready to run, click on the "OK" button to close this window.

The subjects should now be placed in the chambers ready for the experiment to start.

The screen that is now visible is the run-time display.  The details are dependent on the "Display Type" selection, (part of the Session configuration).  The precise details will vary, but the general display layout is constant, see figure 3.30.
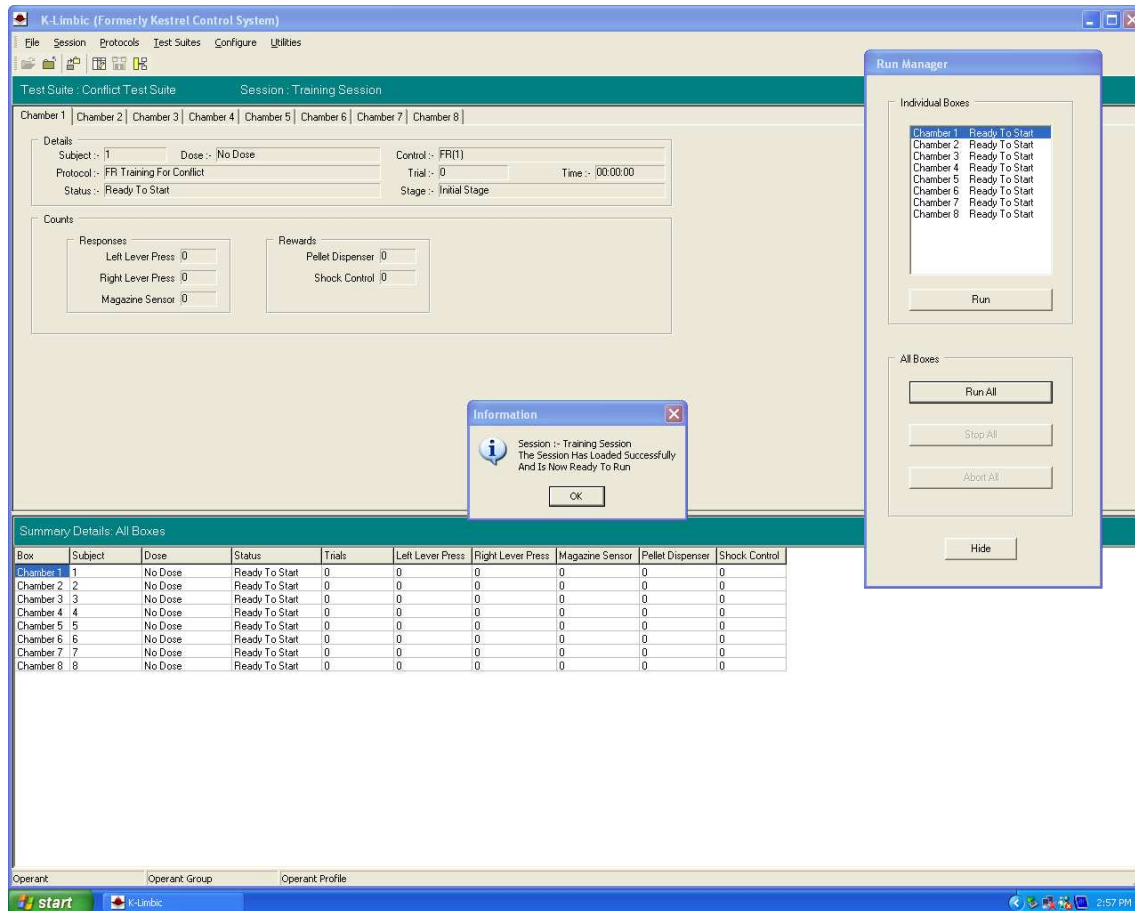
**Figure 3.30 Run Time Session Screen**

At the top there is a tabbed section, 1 tab per box, this section displays subject and protocol information.

At the bottom there is a summary section, 1 row per box so that all boxes are visible at once.

On the right hand side is the "Run Manager" window.  The "Run Manager" provides controls for starting and stopping the session.

**Start The Session**

The most common function is to start all boxes running in one operation, click on the "Run All" button in the Run Manager.  The display will update to show that all boxes are now running, (there is a slight delay between each box starting, this is deliberate).

Alternatively, it is possible to start boxes individually, select the required box in the list in the "Run Manager" and click on the "Run" button.

**3.8 What Happens During An Experiment**

In the majority of cases the operator will not need to take any actions during the course of an experiment, protocols are automatic and require no user interaction.  The protocols will progress until they reach their predefined endpoints.

It is recommended that the operator refrain from the use of other applications while K-Limbic is running a session, (apart from associated K-Limbic applications).  This is considered good practise when running real-time applications.

Please note, during a session K-Limbic disables the screen-saver.


**3.9 Ending A Session Prematurely**

It is possible to end a protocol prematurely, should the user wish to do so.  The details below outline the methods for ending a protocol early:

If you want to save any data generated so far:
Click on the "Stop All" button in the "Run Manager", (or select from the main menu the Session / Stop (Store Data) menu option.
K-Limbic presents an "Are you sure you want to do this?" window.  Click on "Yes".
This will send all running protocols to their exit stage.
All data generated to date is saved.

If you do not want to save any data generated so far:
Click on the "Abort All" button in the "Run Manager", (or select from the main menu the Session / Abort (Discard Data) menu option.
K-Limbic presents an "Are you sure you want to do this?" window.  Click on "Yes".
This will send all running protocols to their exit stage.
All data generated to date is discarded.


**3.10 What Happens At The End Of A Session**

A session ends when all the protocols in the session are complete.

At the end of the session the data generated for each box is formatted and stored in the data directory for the session.  A separate file is created each time a session is run.  A report will appear on screen as the files are being formatted, if no problems are encountered during formatting then this report is removed, if a problem is encountered then an error status message is presented.  Please note, the fixed ratio session as defined here does not generate any data.

The normal situation is that no errors occur and the "Session Complete" information window appears on the screen.  Click "OK" to remove this window.

To run another session:
From the K-Limbic main menu select…
Session / Close Session
This will close the run-time display and return to the main screen, then…
From the K-Limbic main menu select…
Session / Load Session…
Repeat the load session process outlined above.

**3.11 Where Now?**

If you have worked through the fixed ratio protocol example and tested that your final installed test suite and session runs a successful fixed ratio protocol, then you are ready to start writing your own protocols.  The procedure of creating any protocol is the same as for the fixed ratio protocol: configure each of the protocol components in turn until you have created the required experiment.

The details of this tutorial guide are intended not as a protocol guide, but as a guide to the process of creating and editing K-Limbic protocols.

It is always a good idea to illustrate your experiment with a flow chart such as the one in figure 3.4, using the available stage types as the building blocks of your protocol.  If you can display the protocol as a flowchart it is probable that you can create your protocol.  If however the available stage types do not cater for your experiment help is at hand.  Contact Conclusive Solutions and we will be happy to discuss your requirements and extend the capabilities of K-Limbic so that you are able to perform your experiment, (there's nothing we like better than adding new facilities to K-Limbic.

Sections 4 & 5 are a reference to the available stage types and auto controls in K-Limbic.  Section 4 lists each auto control type and discusses the uses and functions of each type or control.  Section 5 lists each stage type and discusses the uses and functions of each type of stage.

Using the procedure outlined in section 3 and the available facilities detailed in sections 4 & 5 we hope that you will be able to proceed with defining the protocol that you require.

Before you start writing your own protocol however please refer to the K-Limbic repository, (on your installation CD), it may well be that there is already a protocol that you can edit to suit your requirements.

<div align="right">

Section

4

# Auto Control Reference

</div>

K-Limbic protocols use "auto controls" to generate experiment specific variables.
The protocol manager checks which type of control is required prior to the start of a session, extracts the required parameters from the database of the selected subject and builds the control values automatically.
The facilities provided by each auto control type range from selecting a delay value to building tables of trials for tasks such as DMTP or the five choice serial reaction time task.  Auto controls support a wide range of behavioural experiments.  The values held in the auto control are available for use by the stages during a protocol, examples of this are given under the individual auto control types.


**Table 4.1 Available Auto Controls**
> Fixed Ratio (FR)
> Variable Interval (VI)
> VI/FR
> Delay Table
> Random Lever
> One Choice
> Five Choice
> T Maze (Constant)
> T Maze (Random)
> Progressive Ratio
> DMTP Control
> Sequential Choice
> Cued Lever
> DRL
> CER

**4.1 The Fixed Ratio Auto Control**

A fixed ratio auto control is built using FR parameters from the subject database.  The fixed ratio auto control consists of one or more fixed ratio controls.

The values held in a fixed ratio control are used by fixed ratio stages in the protocol.

FR Parameters are available in the database under the following parameter types:
- 
- FR Table
- FR Parameters
- VI/FR Table
- VI/FR Parameters

The FR Table may be specified in "Table" mode or in "Series" mode.  In "Table" mode a single set of FR parameters are used to build the Fixed Ratio control, in "Series" mode all entries in the FR Table are used to build the Fixed Ratio control.

If the database contains an FR Table in "Series" mode then multiple Fixed Ratio controls are generated, (one control per entry in the table), in all other cases the auto control consists of a single Fixed Ratio control.



**Figure 4.1 FR Parameters**

A fixed ratio control contains the following settings:

|                       |                                                       |
| --------------------: | ----------------------------------------------------- |
| Fixed Ratio           | The number of responses required, (FRn).              |
| Target Assignment     | The lever or sensor to use as the Target              |
| Dummy Assignment      | The lever or sensor to use as the Dummy               |
| Enable Reset          | Reset the count if a response is made on the Dummy     |
| Enable Rewards        | Enable/Disable rewards                                 |
| Disable Unused Lever  | Ignore the "unused" lever                             |
| Dummy As Target       | Treat the Dummy as if it were a Target                 |

Prior to the start of a protocol both the Target and Dummy are enabled, (if a dummy is used).
If the "Dummy As Target" option is active then the ratio is complete when either count reaches FRn.
If the "Disable Unused Lever" option is active then the when the first fixed ratio is complete the lever
that did not register FRn responses is disabled and will not trigger a ratio complete endpoint.

**4.2 The Variable Interval Auto Control**

A variable interval auto control is built using VI parameters from the subject database.  The variable interval auto control consists of a list of variable intervals.

The values held in a variable interval control are used by VI stages and/or Interval stages in the protocol.

VI Parameters are available in the database under the following parameter types:

- 
- VI Table
- VI Parameters
- VI/FR Table
- VI/FR Parameters

Variable intervals in two categories are available, they may be a table of random intervals, or a table of specified values.

In random mode the database specifies a "mean" value, a "delta" value and a "table length" value, (N). The auto control builds a list of "N" variable intervals where each interval falls randomly in the range

$$Mean - Delta \leq VI \leq Mean + Delta$$

In interval mode, the operator builds a table of delay values.  The table can contain any number of entries.

A new value from the list of variable intervals is used with each iteration through a VI stage or an Interval stage.  If all values in the table have been used the table wraps around to the beginning.

**Figure 4.2 VI Parameters (Random Variable Interval)**

**4.3 The VI/FR Auto Control**

A VI/FR auto control is built using VI/FR parameters from the subject database.  The VI/FR auto control is a combination or a variable interval auto control and a fixed ratio auto control.

VI/FR Parameters are available in the database under the following parameter types:
- VI/FR Table
- VI/FR Parameters

For a description of the variable interval auto control and the fixed ratio auto control refer to the sections on these types of auto control.



**Figure 4.3 VI/FR Parameters**

**4.4 The Delay Table Auto Control**

A Delay Table auto control is built using a Delay Table from the subject database.

The Delay Table parameters are the only applicable parameters in the subject database.

A Delay Table is a list of user specified delays.  The table may contain an arbitrary number of entries.

The operator selects an entry from the table for each box during the load session procedure.  The selected value of delay is used in any stages of type "Delay Stage" in the protocol, the delay value is used as the "Auto Delay" during the stage.
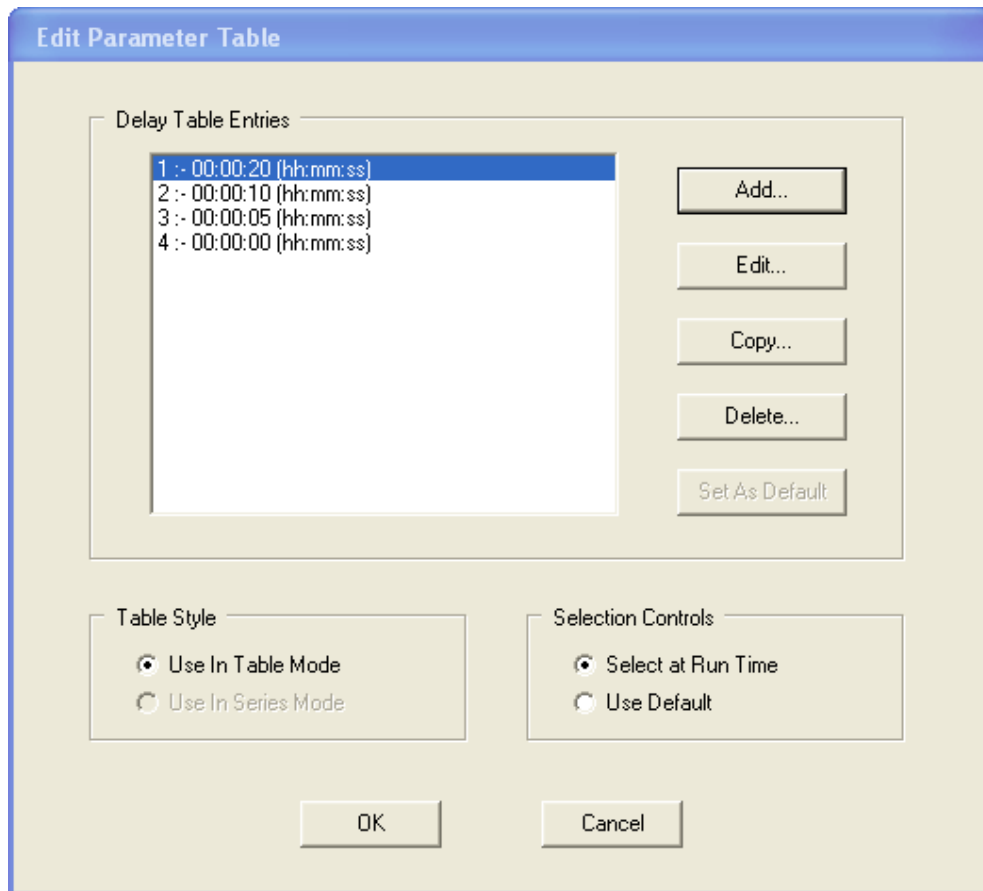


**Figure 4.4 Delay Table**

**4.5 The Random Lever Auto Control**

A Random Lever auto control is built using an Iterations parameter from the subject database.

The random lever auto control generates a list of "trials" each trial assigns the levers as either a "Standard" target, "Reverse" target, "Standard" dummy or "Reverse" dummy.

The number of trials generated is determined by the value of the Iterations parameter and the number of levers in the Resource list of the protocol.

For example if a protocol defines two levers in its resource list and the database parameter specifies that 50 iterations are required then a total of 100 trials will be generated.
In 50 of the trials the lever assignments will be as follows:

|  |  |
|---|---|
| Standard Target | Left Lever |
| Reverse Target | Right Lever |
| Standard Dummy | Right Lever |
| Reverse Dummy | Left Lever |

In the other 50 trials the lever assignments will be as follows:

|  |  |
|---|---|
| Standard Target | Right Lever |
| Reverse Target | Left Lever |
| Standard Dummy | Left Lever |
| Reverse Dummy | Right Lever |

The random lever auto control can cater for up to four levers.

The assignments of the random lever auto control are useable in any stage where it is possible to set the Target or Dummy to the "Standard (Auto)" or "Reverse (Auto)" settings.
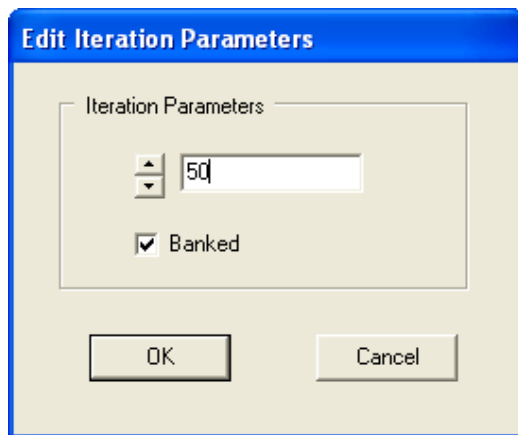


**Figure 4.5 Iteration Parameters**

**4.6 The One Choice Auto Control**

A One Choice auto control is built using a One Choice Table database parameter.  The one choice auto control is specifically intended for protocols implementing the one choice serial reaction-time task. The one choice serial reaction-time task is a simplified version of the five choice task; where only one stimulus light is used, (instead of all five).  The one choice task can be used as a training task for the full five choice task.

A one choice table parameter is a database table that can hold an arbitrary number of entries.  The table can be used in either "Table" mode or "Series" mode.

Each entry in the table specifies the following entries:
**Stimulus Parameters**

| | |
|---|---|
| Duration | Duration of the stimulus "flash" |
| Intensity[++] | Intensity of the stimulus |
| Limited Hold | Time allowed for response to stimulus |
| Active Channel | The index of the active stimulus light |

**Control Parameters**

| | |
|---|---|
| Iterations | Controls the number of trials |
| Banking | Controls the banking of trials |
| Inter-Trial Interval | Duration of the ITI stage |

**Tone Parameters**

| | |
|---|---|
| Tone Burst | Duration of the distraction tone burst |
| Tone Timing | Timing of the distraction tone burst |

[++]Has no effect when using Med Associates hardware.
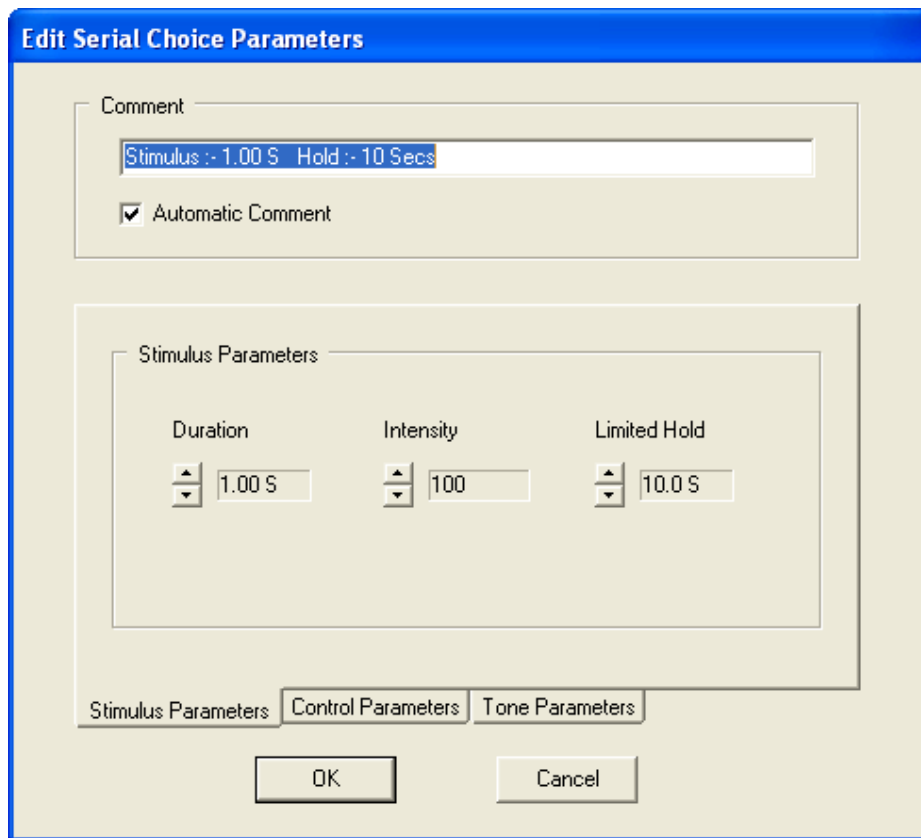


**Figure 4.6 One Choice (Stimulus) Parameters**

The number of trials generated depends on the Iterations setting and the mode of the table.

In "Table" mode, the operator selects one entry from the table, all trials in the session are built using the values held in this one entry. The number of trials is calculated as:
> Trials = 1 (Stimulus light) x Iterations

e.g. To generate a session with 100 trials set the Iterations parameter to 100.

In "Series" mode the trials in the session are built using all values in the table. The number of trials is calculated as:
> Trials = 1 (Stimulus Light) x Iterations x Entries

e.g. With a table with 5 entries, to generate a session with 100 trials set the Iterations parameter to 20.

The one choice table auto control is used by the "Serial Choice" stages held in a protocol in the same fashion as a five choice table auto control is used.

For a full discussion of the use of the five choice auto control please refer to the protocol description documents of any of the five choice serial reaction-time task protocols held in the K-Limbic repository.



**Figure 4.6.2 One Choice (Control) Parameters**

**4.7 The Five Choice Auto Control**

A Five Choice auto control is built using a Five Choice Table database parameter. The five choice auto control is specifically intended for protocols implementing the five choice serial reaction-time task.

A five choice table parameter is a database table that can hold an arbitrary number of entries. The table can be used in either "Table" mode or "Series" mode.

Each entry in the table specifies the following entries:
**Stimulus Parameters**

| | |
|---|---|
| Duration | Duration of the stimulus "flash" |
| Intensity[++] | Intensity of the stimulus |
| Limited Hold | Time allowed for response to stimulus |

**Control Parameters**

| | |
|---|---|
| Iterations | Controls the number of trials |
| Banking | Controls the banking of trials |
| Inter-Trial Interval | Duration of the ITI stage |

**Tone Parameters**

| | |
|---|---|
| Tone Burst | Duration of the distraction tone burst |
| Tone Timing | Timing of the distraction tone burst |

[++]Has no effect when using Med Associates hardware.

The five choice auto control assumes that five stimulus lights will be active in the protocol.



**Figure 4.7 Five Choice (Control) Parameters**

The number of trials generated depends on the Iterations setting and the mode of the table.

In "Table" mode, the operator selects one entry from the table, all trials in the session are built using the values held in this one entry. The number of trials is calculated as:

Trials = 5 (Stimulus lights) x Iterations

e.g. To generate a session with 100 trials set the Iterations parameter to 20.

In "Series" mode the trials in the session are built using all values in the table. The number of trials is calculated as:

Trials = 5 (Stimulus Lights) x Iterations x Entries

e.g. With a table with 4 entries, to generate a session with 100 trials set the Iterations parameter to 5.

The five choice table auto control is used by the "Serial Choice" stages held in a protocol.

For a full discussion of the use of the five choice auto control please refer to the protocol description documents of any of the five choice serial reaction-time task protocols held in the K-Limbic repository.
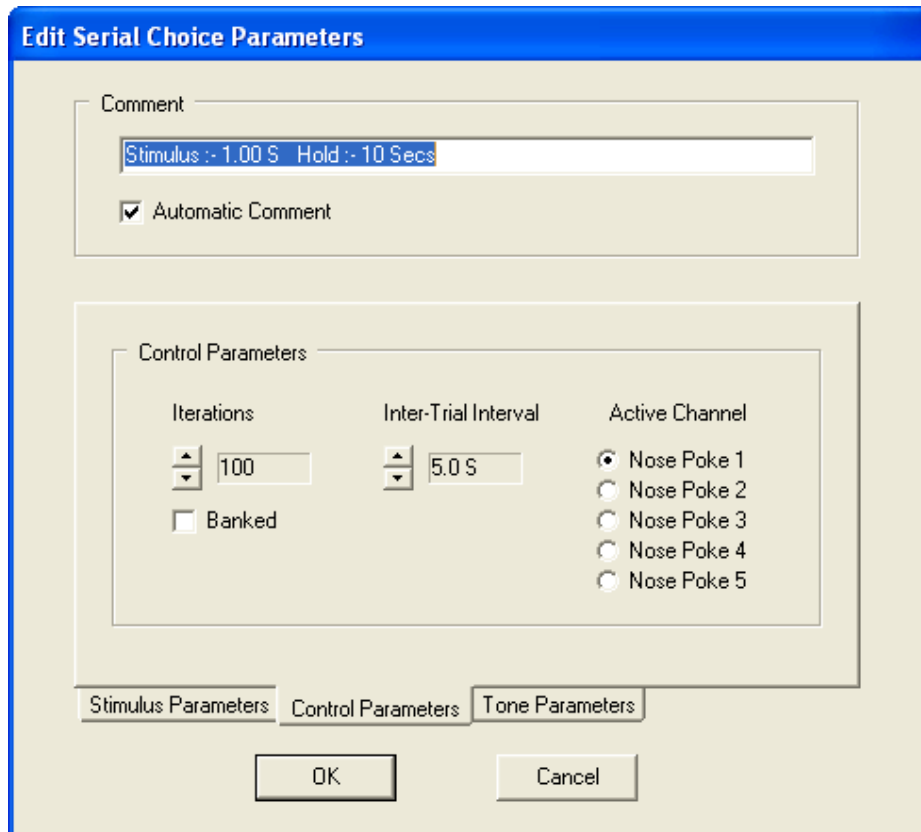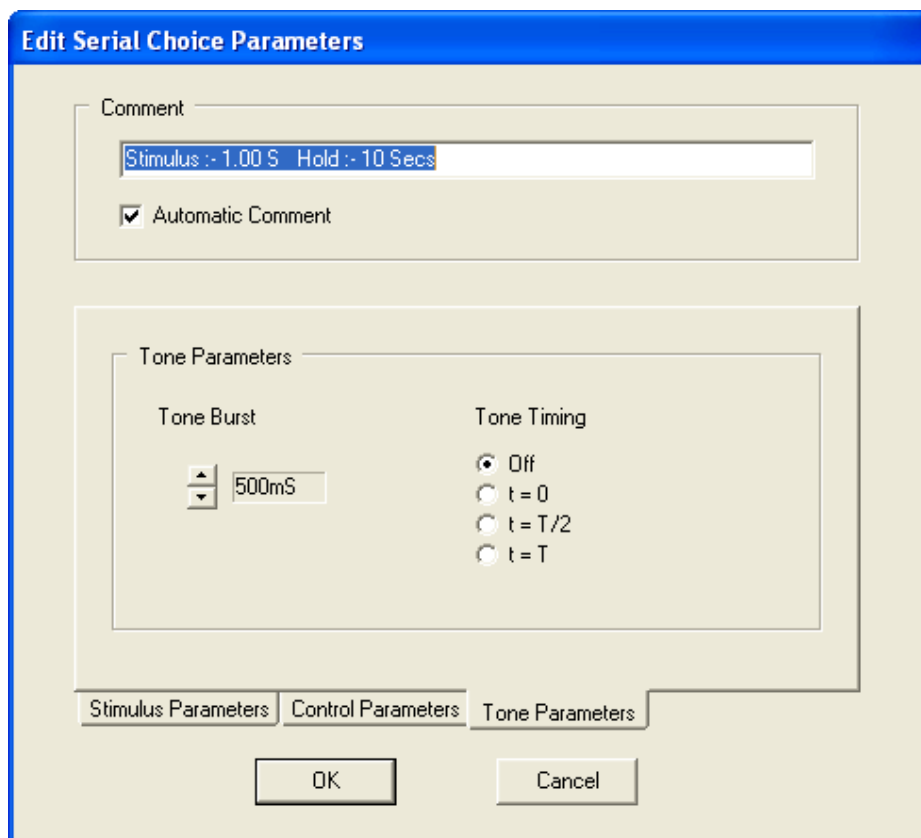


**Figure 4.7.2 Five Choice (Tone) Parameters**

**4.8 The T-Maze (Constant) Auto Control**

A T-Maze (Constant) auto control is built using an Iterations parameter and Delay Table from the subject database.

The T-Maze (Constant) auto control generates a list of "trials". Each trial in the list assigns a "Standard" target, "Reverse" target, "Standard" dummy, "Reverse" dummy and an auto timeout.

The number of trials generated is determined by the value of the Iterations parameter, one trial per iteration.

The operator selects one entry from the delay table for each subject; the delay value is used as the auto-timeout value for all trials.

The T-Maze (Constant) auto control is intended for use in T-Maze tests, where the choice of direction determines the difference between a correct and incorrect trial, in this case the "correct" choice of direction is always the same. The protocol must define resources of type Sensor<1> & Sensor<2>, we have assumed here the Sensor<1> detects a "Left" choice and Sensor<2> detects a "Right" choice.

| | | |
|---|---|---|
| Standard Target | Left | (Sensor<1>) |
| Reverse Target | Right | (Sensor<2>) |
| Standard Dummy | Right | (Sensor<2>) |
| Reverse Dummy | Left | (Sensor<1>) |

The assignments of the T-maze (Constant) auto control are useable in any stage where it is possible to set the Target or Dummy to the "Standard (Auto)" or "Reverse (Auto)" settings and where an "Auto-Timeout" exit trigger is used.

**4.9 The T-Maze (Random) Auto Control**

A T-Maze (Random) auto control is built using an Iterations parameter and Delay Table from the subject database.

The T-Maze (Random) auto control generates a list of "trials". Each trial in the list assigns a "Standard" target, "Reverse" target, "Standard" dummy, "Reverse" dummy and an auto timeout.

The number of trials generated is determined by the value of the Iterations parameter, one trial per iteration.

The operator selects one entry from the delay table for each subject; the delay value is used as the auto-timeout value for all trials.

The T-Maze (Random) auto control is intended for use in T-Maze tests, where the choice of direction determines the difference between a correct and incorrect trial, in 50% of trials the "correct" choice of direction is right and in 50% of trials the "correct" choice is left.

The protocol must define resources of type Sensor<1> & Sensor<2>, we have assumed here the Sensor<1> detects a "Left" choice and Sensor<2> detects a "Right" choice.

In 50% of trials the assignments are:

|  |  |  |
|---|---|---|
| Standard Target | Left | (Sensor<1>) |
| Reverse Target | Right | (Sensor<2>) |
| Standard Dummy | Right | (Sensor<2>) |
| Reverse Dummy | Left | (Sensor<1>) |

In the other 50% of trials the assignments are:

|  |  |  |
|---|---|---|
| Standard Target | Right | (Sensor<2>) |
| Reverse Target | Left | (Sensor<1>) |
| Standard Dummy | Left | (Sensor<1>) |
| Reverse Dummy | Right | (Sensor<2>) |

The assignments of the T-maze (Random) auto control are useable in any stage where it is possible to set the Target or Dummy to the "Standard (Auto)" or "Reverse (Auto)" settings and where an "Auto-Timeout" exit trigger is used.

**4.10 The Progressive Ratio Auto Control**

A Progressive Ratio auto control is built using a progressive ratio parameter from the subject database.

Progressive ratio parameters are available in the database under the following parameter types:

- Progressive Ratio Table
- Progressive Ratio Parameters

A Progressive Ratio table is always used in "Table" mode, i.e. the operator selects one entry from the table for use in the current session.

The progressive ratio parameters are intended for use in progressive ratio tests, the values held in a progressive ratio control are used by progressive ratio stages in the protocol.



**Edit PR Parameters**

Comment
Example Progressive Ratio

Progression
F(n) = 5 x EXP(0.2n) - 5

Control Options
☐ Enable Reset
☐ Reward On Dummy

Breakpoints
☑ Idle Timeout
Hours: 00  Mins: 00  Secs: 30

☑ No Reward Breakpoint
Hours: 00  Mins: 5  Secs: 00

OK          Cancel

**Figure 4.10 Progressive Ratio Parameters**

A progressive ratio control contains the following settings:

| | |
|---|---|
| Progression | The sequence of increasing ratios |
| Enable Reset | Reset the count if a response is made on the Dummy |
| Reward On Dummy | Treat The Dummy as if it were a Target |
| Idle Timeout Enable | Enable the idle timeout |
| Idle Timeout | The idle timeout value |
| No Reward Breakpoint Enable | Enable the "no reward" timeout |
| No Reward Breakpoint | The "no reward" timeout value |

The target ratio for the progressive ratio stage increases on each iteration through the stage. The ratio increases according to the selected progression type, examples of the available progressions are listed below.

**Available Progressions**
There are a number of options that can be selected for the progression:

1.  F(n) = 5 x EXP(0.2n) - 5
    Example sequence
    1, 2, 4, 6, 9, 12, 15, 20, 25, 32, 40, 50, 62, 77, 95, 118, 145, 178, 219, 268 etc

2.  Option 1
    Example sequence
    1, 2, 3, 4, 5, 6, 7, 8, 10, 12, 14, 16, 18, 20, 22, 24, 28, 32, 36, 40, 44, 48, 52,
    56, 64, 72, 80, 88, 96, 104 etc

3.  Option 2
    Example sequence
    1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 130, 150,
    170, 190, 210, 230, 250, 270, 290, 310 etc

4. Constant (10)
    This option uses a constant value for the ratio of 10.

5. Incremental (1 –100)
    This is a simple progression where the ratio increases by 1 each time.

**4.11 The DMTP Auto Control**
 (Delayed Match To Position/Delayed Non-Match To Position)

A DMTP auto control is built using a DMTP parameter from the subject database.

DMTP parameters are available in the database under the following parameter types:

- DMTP Table
- DMTP Parameter

A DMTP table is always used in "Table" mode, i.e. the operator selects one entry from the table for use in the current session.

The DMTP parameters are intended for use in DMTP tests.  (Refer to the K-Limbic repository for a full discussion of the DMTP test).



**Figure 4.11 DMTP Parameters**

A DMTP control contains the following settings:

|                        |                                                    |
|-----------------------:|:---------------------------------------------------|
| Standard Targets       | List of available targets, (levers or sensors)     |
| Reverse Targets        | List of available reverse targets, (levers or sensors) |
| Delays                 | List of delays to be used                          |
| Iterations             | Number of trial groups                             |
| Banking                | Enable/disable banking                             |
| Matching Mode          | Match to position / non-match to position          |
| Repeat Incorrect Trials | Enable/disable repeated trials                    |
| Repeat Missed Trials   | Enable/disable repeated trials                     |

The DMTP control produces a table of trial settings used to control a DMTP experiment.

**Standard Targets**
The Standard Targets list should contain one entry for each lever/sensor that will be used as a "Sample" lever during the test.  For example, in a lever based DMTP test the two levers will appear in the Standard Targets list.

**Reverse Targets**
The Reverse Targets list should contain one entry for each lever/sensor that will be used as a "Reverse" lever during the test.  For example, in a lever based DMTP test if Lever <1> is the "Standard" lever then Lever <2> will be the "Reverse".  Both levers are presented in the "Choice" stage of the test, a response on the "correct" lever results in a reward.

The lists of standard and reverse targets can also contain lists of "sensors" or "nose pokes" for alternative DMTP tests, for example where you need to perform a DMTP test in a five choice chamber.

**Delays**
A list of delay entries.  In the DMTP task a variable delay is introduced between the "Sample Lever" response and the "Choice" stage.  The delay list determines the delay values that will be used.

Refer to fig 4.11 for an example DMTP control using two levers and six values of delay.

**Iterations**
In the DMTP control, an "iteration" consists of a group of trials.  In each group a trial is defined in which each sample lever is associated with each delay value, for example the DMTP control in fig 4.11 uses 2 levers and 6 delay values therefore each iteration contains 12 trials.  Using an Iterations value of 8 therefore produces a total of 96 trials in the control.

**Banking**
If "banking" is enabled then the trials are randomised within each group but presented as a group of trials, i.e. the first 12 trials will contain one trial where each sample lever is associated with each delay, the same is true of the next group of 12 trials etc.  If "banking" is disabled then the organisation of trials is fully random.  Using "banking" prevents the situation where consecutive trials present the same combination of sample lever & delay.

**Matching Mode**
The "matching mode" has two options:
|                      |                                                      |
|----------------------|------------------------------------------------------|
| Match To Position    | The "Standard" lever is the "correct" lever in the "Choice" stage. |
| Non-Match To Position | The "Reverse" lever is the "correct" lever in the "Choice" stage. |

**Repeat Incorrect Trials**
If enabled any "incorrect" trials detected during the test are repeated.

**Repeat Missed Trials**
If enabled any "missed" trials detected during the test are repeated.

The controls to repeat trials are useful during training to correct subjects displaying a "bias" towards one lever, i.e. animals that always press the same lever at the "choice" stage, (which would result in 50% of trials being correct).

The settings in the DMTP auto control are accessed in any stage where it is possible to set the Target/Dummy options to "Standard (Auto)" or "Reverse (Auto)".  Access the delay value from any stage with the "Use Auto Timeout" exit trigger.  The standard DMTP protocol uses "Action" stages, refer to protocol 240150 in the K-Limbic repository.

**Target/Dummy Assignments with the DMTP control**
An important aspect of using the DMTP control is the understanding of how the Standard & Reverse targets are assigned to the Target & Dummy resources during the protocol.

**Using the "Standard (Auto)" option**
If the "Target" assignment in any stage is set to "Standard (Auto)" then the "Standard Target" from the DMTP control for the current trial will be assigned as the "Target" resource.

If the "Dummy" assignment in any stage is set to "Standard (Auto)" then "Reverse Target" from the DMTP control for the current trial will be assigned as the "Dummy" resource.

**Using the "Reverse (Auto)" option**
The result of using the "Reverse (Auto)" option is determined by the "Matching Mode" value.

In "Match To Position" mode…
   If the "Target" assignment in any stage is set to "Reverse (Auto)" then the "Standard Target" from the DMTP control for the current trial will be assigned as the "Target" resource.
   If the "Dummy" assignment in any stage is set to "Reverse (Auto)" then the "Reverse Target" from the DMTP control for the current trial will be assigned as the "Dummy" resource.

In "Non-Match To Position" mode…
   If the "Target" assignment in any stage is set to "Reverse (Auto)" then the "Reverse Target" from the DMTP control for the current trial will be assigned as the "Target" resource.
   If the "Dummy" assignment in any stage is set to "Reverse (Auto)" then the "Standard Target" from the DMTP control for the current trial will be assigned as the "Dummy" resource.

To create the "Sample Lever" & "Lever Choice" stages in a DMTP protocol use the following Target/Dummy assignments:

Sample Lever Stage
        Target Assignment:     Standard (Auto)
        Dummy Assignment:      None

Lever Choice Stage
        Target Assignment:     Reverse (Auto)
        Dummy Assignment:      Reverse (Auto)

Using the above Target/Dummy assignments it is possible to create a DMTP protocol that can be used for both DMTP & DNMTP tasks, the style of task being dependent on the settings in the subject database.

**4.12 The Sequential Choice Auto Control**

A sequential choice auto control is built using sequential choice parameters from the subject database.

The sequential choice auto control is intended for use in sequential choice tests using a five choice chamber.
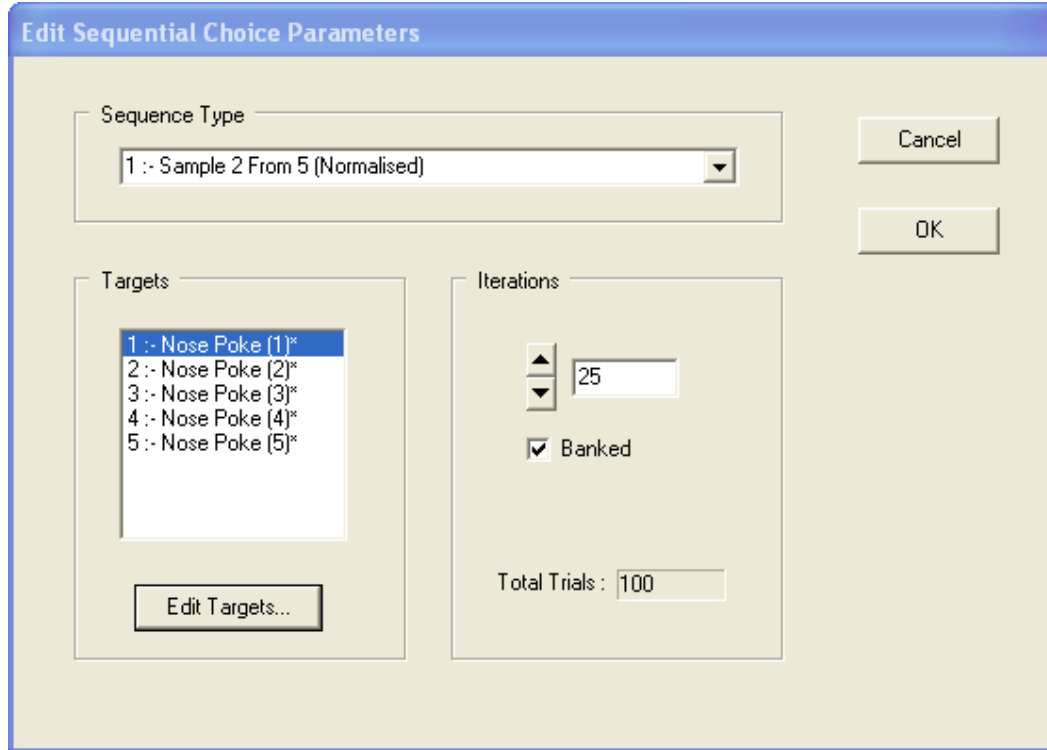


**Figure 4.12 Sequential Choice Parameters**

A sequential choice control contains the following settings:

| | |
|---|---|
| Sequence Type | List of sequence options |
| Targets | List of available targets, (levers, sensors or nose pokes) |
| Iterations | Number of trial groups |
| Banking | Enable/disable banking |

The control produces a table of trial settings used to control a sequential choice experiment.

There are two available options for the sequence:
    (1)  Sample 2 from 5 (Normalised)
    (2)  Sample 3 from 5 (Normalised)

Sequence 1 is used to present a sequence of 2 cue lights followed by presenting both lights together, to gain a reward the subject must respond on the light presented first.

Sequence 2 is used to present a sequence of 3 cue lights followed by presenting two of the three lights together, to gain a reward the subject must respond on the light presented first.

The sequences are split into groups.
In sequence 1 each group has the same "distance" between the lights, i.e. 1, 2, 3 or 4, resulting in 4 groups and a bank size or 4.
In sequence 2 each group has the same combination of "lag" and "distance" between the lights, resulting in 8 groups and a bank size of 8.

The settings of the sequential choice auto control are used by stages of type "Sequential Choice", (refer to the K-Limbic repository for a full discussion of the sequential choice test).

**4.13 The Cued Lever Auto Control**

A Cued Lever auto control is built using cued lever reversal parameters from the subject database.

The Cued Lever auto control is intended for use in cued lever, (serial reversal) tests using a skinner box with two levers and two cue lights.  The settings of the cued lever auto control are accessed from "Cued Lever" stages in the protocol.



**Figure 4.13 Cued Lever Reversal Parameters**

The Cued Lever test, (Serial Reversal), protocol is available in the K-Limbic repository, refer to the protocol description document for a full discussion of the protocol.  In brief, the task consists of presenting both levers to the subject and illuminating the cue light above one of the levers, the subject must learn to respond on one lever, (either cued or un-cued), to receive a reward, after completing a given number of trials the task changes and the subject must learn to switch to the opposite lever, (either un-cued or cued).

The cued lever parameters support tasks where the target lever changes during the session on a trial-by-trial basis, or where the target lever remains constant within a session but varies from one session to the next.

The sequence of target levers may be preset or random as specified by the "Sequence Control" parameter, eight normalised sequences are provided and four training sequences.

The cue lights are controlled by the "Cue Control" parameter of the auto control and the "Task Control" parameter of the "Cued Lever" stage:

| Cue Control | Task Control | Effect on Cue Light |
|-------------|--------------|---------------------|
| Standard | Standard | Illuminate above target lever |
| Standard | Reverse | Illuminate above dummy lever |
| Reverse | Standard | Illuminate above dummy lever |
| Reverse | Reverse | Illuminate above target lever |
| None | na | Both cue lights off |
| Both | na | Both cue lights on |

For protocols that vary the cue from session to session, it is possible to instruct the protocol manager to store the current settings in the cued lever parameters so that on the next session the cue control follows the sequence.

A full discussion of the cued lever, (serial reversal), task is included in the protocol description document for the cued lever task in the K-Limbic repository.

**4.14 The DRL Auto Control**

A DRL auto control is built using DRL parameters from the subject database.

The DRL parameters are available as a "DRL Table", the table can contain an arbitrary number of DRL parameter entries.  The operator selects one entry from the table during the load session procedure.

The DRL auto control is intended for use in DRL, (delayed-reinforcement to lever) tests.
The settings of the CRL auto control are accessed from "DRL" stages in the protocol.



**Figure 4.14 DRL Parameters**

The DRL auto control generates a list of trials, for each trial the control generates a "Delay" and a "Trial Timeout", the values of these parameters are specified by the user.

The number of trials generated is determined by the Iterations & Bank Size parameters.  A "peak" trial is one in which the "Delay" value is indefinite, if "Peak" trials are enabled then one trial per bank will be a "peak" trial, the peak trial is randomly presented.  If "banking" is enabled then "peak" trials are evenly distributed throughout the test, if "banking" is disabled then the distribution of "peak" trials is fully random.

Refer to the K-Limbic repository for a discussion of the DRL test.

**4.15 The CER Auto Control**
(Conditioned Emotional Response)

A CER auto control is built using a CER parameter from the subject database.

CER parameters are available in the database under the following parameter types:

- CER Table
- CER Parameter

A CER table is always used in "Table" mode, i.e. the operator selects one entry from the table for use in the current session.

The CER parameters are intended for use in CER tests. (Refer to the K-Limbic repository for a full discussion of the CER test).
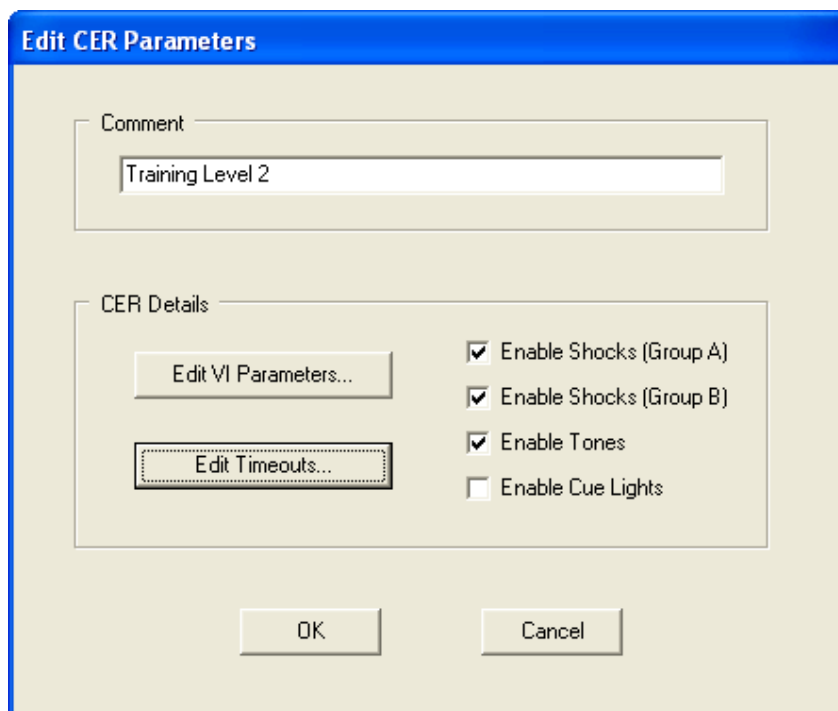


**Figure 4.14 CER Parameters**

As illustrated in fig 4.14, the CER parameters are a combination of standard VI parameters, a variable auto timeout and four enabling controls.

The VI parameters generate a standard list of variable intervals; see section 4.2 for a discussion. He variable interval values are accessed as the interval value during VI stages.

The Timeout value can be either a set period or as with a random variable interval a value picked randomly from a specified range.

The enabling controls act as global controls throughout a protocol.
If shocks in the protocol are assigned as being in "Shock Group A" or "Shock Group B" then shock delivery is controlled by the auto control.
If the Tone setting in the protocol is set to "On If Enabled" then the auto control controls delivery of the tone.
If Cue Lights are disabled in the auto control, then they are disabled in the protocol, whatever setting in the protocol is used.

Careful setting of the parameters of a CER table will result in on CER table that caters for all possible variants on the CER test, from training through to test, using a single protocol.



**Figure 4.14.1 CER Auto Timeout Parameters**

A CER protocol uses a sequence of VI stages, for full details refer to the CER protocol description in the K-Limbic repository.

Section

# 5
# Stage Type Reference

K-Limbic provides a range of stage types to cater for various behavioural paradigms; the range of stage types is listed in table 5.1.
 "Experiment specific" behaviour is included in some stage types, (such as in the case of "Serial Choice" type stages).  Experiment specific stage types make defining protocols easier as some "expert knowledge" is included in the stage itself removing the need for the developer to create the behaviour from scratch.

The "Serial Choice – Present Stimulus" stage provides an example of experiment specific behaviour.  The stage has the built in behaviour of presenting a stimulus in one of the niche lights and then waiting for the limited hold period for a response.  The stage automatically retrieves the stimulus parameters and the limited hold period from the auto control, the developer simply defines the protocol with the stage in the correct place and the stimulus will appear.

**Table 5.1: K-Limbic Stage Types**

| Stage Type | Function |
| --- | --- |
| Initial Stage | Sets the initial state of the operant box, (prior to starting the protocol). |
| Exit Stage | Sets the exit state of the protocol, (following completion of protocol). |
| Action | Waits for a trigger to occur |
| Reward | Dispenses a reward, (reinforcement). |
| Mark Trial/Set Outcome | Increments the trial number & defines the outcome of a trial, (e.g. correct/incorrect etc). |
| Fixed Ratio | Runs a fixed ratio. |
| Fixed Ratio (Meted) | Runs a fixed ratio. |
| Variable Interval | Runs a variable interval. |
| Shadow Stage | Responds to the activation of a shadow trigger. |
| Timer Stage | Used To Time the period of activation of a sensor. |
| Serial Choice - Inter-Trial Interval | The ITI stage of the 5CSRTT. |
| Serial Choice - Present Stimulus | The stimulus & limited hold of the 5CSRTT. |
| Serial Choice – Collection | The "Wait for collection" stage of the 5CSRTT. |
| Serial Choice – Timeout | The "Timeout" stage of the 5CSRTT. |
| Serial Choice – Training | A training stage for the 5CSRTT. |
| Interval Stage | A fixed or variable interval stage |
| Delay Stage | Waits for a specified delay to occur. |
| Progressive Ratio Stage | Implements a progressive ratio schedule |
| Cued Choice Stage | Used in DNMTP protocols |
| Sequential Choice Stage | Defines a series of responses |
| Cued Lever Stage | Defines a cued lever stage |
| Epoch Control Stage | Controls the epoch manager |
| Outcome Stage | Set the outcome of a trial |
| DRL Stage | Used in DRL tests |
| Timeout Stage | A general timeout stage |
| Location Stage | Defines the location of a subject. |

**5.1 The Components Of A Stage**

All stages are built from a number of components.  To develop a protocol the developer needs to edit the settings in each component of each stage.  The "Stage Wizard" is provided to assist the developer in this process, (see the tutorial guide).  The stage wizard steps through each component of a stage and provides controls for the developer to select the required setting for each component.

In general each stage has the components listed below, some stages have additional components to provide a specific behaviour, the components for each stage type are listed under each stage type in this reference.

In most cases the components for each stage type are tailored to suit the specific behaviour of that stage type, for example the available stage reports for a fixed ratio stage are different from the stage reports for a reward stage etc.

The common components of a stage:

- Driver Controls – The driver controls specify the settings for each resource in the protocol, i.e. retractable levers, cue lights, enclosure lights, tones, feeders, fans etc.  The stage wizard only presents controls for the particular resources defined in the protocol.
- Display Message – A stage can send a "message" to the on-screen display during a session, a display message may inform the display for instance that the protocol has moved from a VI phase to an FR phase, or that counting on a resource has been disabled.
- Target/Dummy Settings – Targets & Dummies are important in many protocols, in general a target identifies the "correct response" sensor and the dummy identifies one or more "incorrect response" sensors.
- Stage Exit Triggers – specify how the stage will end, i.e. on a timeout, a response on the target sensor, or a completed fixed ratio etc.
- Stage Reports – Allows the developer to tailor the contents of the output file.

The components above are common to most stage types, (although not all stage types have all the above components).  In addition, stage types exhibit specialised behaviour, the specialised behaviours are controlled by stage specific components, these are discussed for each stage type.

The following sections list each stage type available in K-Limbic and give an indication of the available functions.

**5.2.1 Initial Stage**

**Components:**
- Driver Controls
- Display Message
- Target/Dummy Settings
- Message Filter

**Function**
An "Initial Stage" sets the state of the operant chamber prior to the start of the session.  In most cases the state specified in the "Initial Stage" is the state required when the animal is placed in the operant chamber.

The "Initial Stage" is active once the session is loaded, when the operator starts the session the protocol manager exits the "Initial Stage" and enters the stage specified as the "Continue" destination of the "Initial Stage".

All protocols require one stage of type "Initial Stage", by convention the "Initial Stage" appears as stage 1 in the stage list and is called "Start".

The "Initial Stage" does not monitor the state of the sensors therefore it produces no output.

The "Message Filter": Enables/disables sensor release messages from the hardware throughout the protocol.  If the "Message Filter" is enabled then the hardware reports when a sensor is activated but not when a sensor is released, in many cases the experimenter is not interested in when a sensor is released.  (The "Message Filter" setting has no effect in the Med Associates version of K-Limbic).

**5.2.2 Exit Stage**

**Components:**
- Driver Controls

**Function**
An "Exit Stage" sets the state of the operant chamber subsequent to the end of the session.

The "Exit Stage" is active when the protocol is complete.  If the flow of an experiment sends the protocol to the exit stage then the protocol is considered to have ended, even if none of the specified endpoints have been triggered.

All protocols require one stage of type "Exit Stage", by convention the "Exit Stage" appears as the final stage in the stage list and is called "Exit".

The "Initial Stage" does not monitor the state of the sensors therefore it produces no output.

### 5.2.3 Action Stage

**Components:**
- Driver Controls
- Display Message
- Target/Dummy Settings
- Stage Exit Triggers
- Stage Reports

**Function**
An "Action Stage" sets the state of the operant chamber, then waits for a response or a timeout.

The Target/Dummy setting for an "action" stage may be a specified sensor, or the "Standard" or "Reverse" sensor as defined for the current trial in the auto control.

The timeout value may be a pre-defined, (fixed) timeout, or the "Delay" value for the current trial in the auto control.

"Action" stages always reset the stage timer on entry therefore they can be re-entered many times in the same trial and will always last for the full timeout period.

**5.2.4 Reward Stage**

**Components:**
- Rewards & Shocks
- Driver Controls
- Display Message
- Target/Dummy Settings
- Stage Reports

**Function**
A "Reward Stage" triggers the delivery of a reinforcer.  A reinforcer may be reward device, such as a pellet dispenser or syringe pump, or it may be a shock generator.  A "Reward Stage" also sets the state of the operant chamber.

The trigger generated by a reward stage is a pulse of a user-specified pulse length.  The pulse length is specified using the "Rewards Table" of the subject database.

A "Reward Stage" is instantaneous, although there is a finite time taken to process a "reward stage" this time is very short and unperceivable by the subject, the protocol triggers the reinforcer, sets the state of the operant chamber and then moves directly to the next stage.  The end of the reward stage is immediate; it does not wait until the end of the trigger pulse.

A "reward stage" does not monitor the pellet dispenser status line or the pellet delivery sensor.

"Rewards & Shocks": A reward reinforcer may be disabled, unconditionally enabled or conditionally enabled, ("Dispense If Target").  A dispenser is a "target" if it has the same index as the current target resource, for instance if the current target is of type "Sensor <1>" then "Pellet Dispenser <1>" will be considered the "Target" dispenser.  An example of this is in maze tests, where you wish to deliver a pellet into the target receptacle prior to the start of a trial.  The "Dispense If Target" option allows a single reward stage to deliver a pellet into the current target receptacle, regardless of the "target" setting for the current trial.

A shock control may be disabled, unconditionally enabled or conditionally enabled.  Conditional enables separate the shocks into one of two groups, (A & B), the auto control and/or session controls can be used to enable/disable shocks in these groups.

The duration of the reward and shock pulses is governed by entries in the Rewards Table of the database.  The reward stage specifies which list of rewards from the rewards table is applicable to each reinforcer.

### 5.2.5 Mark Trial/Set Outcome

**Components:**
- Trial Marker Actions
- Display Message

**Function**
A "Mark Trial/Set Outcome" stage is a protocol management stage.  The stage is used to mark the outcome of a trial, (e.g. "Correct", "Incorrect", "Missed" etc) and/or instruct the protocol manager to increment the trial pointer, repeat the last trial or mark a trial as incomplete.

A "Mark Trial/Set Outcome" stage is instantaneous; although there is a finite time taken to process a "Mark Trial" stage this time is very short and unperceivable by the subject.

A "Mark Trial/Set Outcome" stage is essential in any "trials" based protocol, (i.e. a protocol in which the same basic task is repeated many times during a test), the stage marks the end of the current trial and instructs the protocol manager to move on to the next trial.

If all trials in the current protocol have been completed the stage exits via the "Trials Complete" exit point, otherwise the stage exits via the "Continue" exit point.

**5.2.6 Fixed Ratio Stage**

**Components:**
- Driver Controls
- Display Message
- Stage Exit Triggers
- FR Controls
- Stage Reports

**Function**
A "Fixed Ratio" stage sets the state of the operant chamber, then waits until a specified number of responses, (the fixed ratio), are made on the target/dummy sensor or a timeout occurs.

The Target/Dummy setting and the fixed ratio value are specified by the fixed ratio parameters of the subject database.

A fixed ratio stage may be iterated, (looped), many times during the same trial.  The stage timeout begins on the first entry to the fixed ratio stage in any one trial; it is not reset on subsequent entries to the stage during the same trial.  This allows the protocol to define an "FR phase" of a specified length.

Consider an FR stage with a fixed timeout of 5 minutes, each time the subject completes a fixed ratio the protocol exits the FR stage, passes through a reward stage, (to trigger the pellet dispenser) and then loops back and re-enters the FR stage.  The sequence of fixed ratio complete/reward delivery may be repeated many times during the five minute period, at the end of the 5 minutes the stage exits via the stage timeout exit trigger and moves on to another part of the protocol

### 5.2.7 Fixed Ratio (Meted) Stage

**Components:**
- Driver Controls
- Display Message
- Stage Exit Triggers
- FR Controls
- Stage Reports

**Function**

A "Fixed Ratio (Meted)" stage is an adapted "fixed ratio" stage intended for situations where the rate of reinforcement must be limited. Protocols in which the rate of drug delivery is dependent on the rate of responding sometimes require the protocol to limit the rate of drug delivery.

A "fixed ratio (meted)" stage exhibits the same behaviour as a standard "fixed ratio" stage, but adds in a "Metering Period", the metering period is the minimum period between successive triggers of the end of the stage, (this is usually the same as the period between reinforcements).

Consider a protocol that counts "licks" and delivers a reward on every $20^{th}$ lick. The reward mechanism takes 1.5 seconds to deliver the reinforcement and does not respond to a trigger that occurs before the end of the current delivery. In practise it is found that the subjects are able to produce 20 licks in less than 1.5 seconds, so using a standard fixed ratio stage results in some triggers of the reinforcer being "ignored".

If a "fixed ratio (meted)" stage is used instead and a metering period of 2 seconds is specified, then there is minimum period of 2 seconds between entry to the stage and exit from the stage, so no reinforcement triggers will be ignored. Responses are accumulated so any excess licks count towards the next reinforcement, in the case above with a fixed ratio of 20 and a metering period of 2 seconds, if the subject records 25 licks during the 2 second period then 5 licks will be carried over to the next ratio count.

**5.2.8 Variable Interval Stage**

**Components:**
- Driver Controls
- Display Message
- Target/Dummy Settings
- Stage Exit Triggers
- Interval Controls
- Stage Reports

**Function**
A "Variable Interval" stage provides a period sub-divided into smaller (random) intervals.  Variable intervals are used commonly in operant tasks where it is desirable to encourage a subject to respond at a high rate.

The variable interval settings are specified by the variable interval parameters of the subject database. A VI stage contains stage timeout exit triggers and an "end of interval" trigger.  The developer specifies the "end of interval" trigger to occur either immediately the interval timeout occurs, (Trigger On Interval) or on the first response that occurs after the interval timeout, (Wait For Target).

A VI stage may be iterated, (looped), many times during the same trial.  The stage timeout begins on the first entry to the VI stage in any one trial; it is not reset on subsequent entries to the stage during the same trial.  This allows the protocol to define a "VI phase" of a specified length.

Consider a VI stage with a fixed timeout of 5 minutes, each time the end of interval exit is triggered the protocol exits the VI stage, passes through a reward stage, (to trigger the pellet dispenser) and then loops back and re-enters the VI stage.  The sequence of interval complete/reward delivery may be repeated many times during the five minute period, at the end of the 5 minutes the stage exits via the stage timeout exit trigger and moves on to another part of the protocol

**5.2.9 Shadow Stage**

**Components:**
- Driver Controls
- Display Message
- Stage Reports

**Function**
A "Shadow" stage provides a secondary state of the operant chamber.

The "Stage Exit Triggers" of many stage types include a "Shadow Trigger"; any sensor in the resource list may be assigned as the shadow trigger.  If the subject activates the shadow trigger in any stage, the protocol resets the state of the operant chamber to the state defined in the "Shadow" stage, but continues with the current stage.

A common example of the use of a shadow stage: consider an operant chamber that includes a light in the pellet receptacle.  A reward stage in the protocol turns on the light in the pellet receptacle to signal that the pellet is there, the protocol then enters the next stage, an FR stage for example.  In the FR stage the head entry detector on the pellet receptacle is assigned as the shadow trigger, when the subject collects the pellet this triggers the shadow stage.  The shadow stage defines the state of the receptacle light as "Off", so collecting the pellet triggers the protocol to turn off the receptacle light, the protocol continues with the FR stage.

**5.2.10 Timer Stage**

**Components:**
- Driver Controls
- Display Message
- Target/Dummy Settings
- Stage Exit Triggers
- Stage Reports

**Function**

A "Timer" stage is used in situations where the protocol needs to record the time a sensor is active.

The stage assigns a sensor as the "target" sensor, (and an optional "dummy" sensor). The stage records the time the sensor is active. An optional "Timeout On Target" exit trigger provides the ability to exit the stage after the sensor has been active for a specified period.

A Timer stage is useful for monitoring the duration a subject is present in a particular location of the chamber, at the drinking spout for example.

A protocol that uses any "Timer" stages should not enable the "Message Filter", (see the "Initial" stage description).

### 5.2.11 Serial Choice – Inter-Trial Interval Stage

**Components:**
- Driver Controls
- Display Message
- Target/Dummy Settings
- Stage Exit Triggers
- Stage Reports

**Function**

A "Serial Choice – Inter-Trial Interval" stage is specifically designed to perform the inter-trial interval, (ITI), function in the five-choice serial reaction-time task.

The stage sets the state of the operant chamber, (normally just the house light), then pauses for a specified period, the "ITI". The ITI is specified as a fixed stage timeout or uses the ITI setting of the five choice parameters of the subject database. Using the database is the preferred method as it allows for multiple ITI values during the same session.

Time-variable tone presentation is a special function of the "serial choice – inter-trial interval" stage. The five choice parameters include optional settings for the "distraction" tone. In any trial it is possible to trigger the tone at either the beginning, the end or half-way through the ITI period. The developer decides when they want the tone to appear and sets the appropriate value in the subject database. For example, a session may include two types of trial, in one type the ITI is 5 seconds duration, in the other type of trial the ITI is 10 seconds duration, but at T/2, (5 seconds), the ITI stage presents distraction tone. The stimulus presentation immediately follows the ITI stage. In other words, in 50% of trials just as the subject is expecting to see the stimulus the tone sounds to distract the subject, the actual stimulus appears 5 seconds later. Alternatively, the distraction tone could be set to present at the end of the ITI, i.e. coincident with the stimulus.

The "Serial Choice – Inter-Trial Interval" stage always resets the stage timer on entry therefore it can be re-entered many times in the same trial and will always last for the full ITI period.

The "Serial Choice – Inter-Trial Interval" stage automatically assigns all "nosepoke" sensors in the chamber as "Dummy" resources, i.e. all the niche sensors. A response in any niche during the ITI is normally considered a "premature response", i.e. a response prior to the stimulus. A five choice paradigm normally sends the protocol into a timeout state as a result of a premature response.

The "Target" resource may be any sensor in the chamber that is not a "nosepoke" sensor.

### 5.2.12 Serial Choice – Present Stimulus Stage

**Components:**
- Driver Controls
- Display Message
- Stage Exit Triggers
- Stage Reports

**Function**

A "Serial Choice – Present Stimulus" stage is specifically designed to perform the "Present Stimulus" function in the five-choice serial reaction-time task.

The stage sets the state of the operant chamber, (normally just the house light), then presents the stimulus, the stimulus is a "flash" of a cue light in one of the niches. The stimulus parameters are controlled by the auto control, which in turn is determined by settings in the five choice parameters of the subject database. The stage pauses for a response in one of the niches, (correct or incorrect), or for the stage timeout to occur. The stage timeout is called the "Limited Hold" period.

The "Limited Hold" period is specified using the Limited Hold setting of the five choice parameters of the subject database.

Stimulus presentation is a special function of the "serial choice – present stimulus" stage. Presentation of the stimulus is automatic. The developer is not concerned with defining the stimulus when developing the protocol. The subject database and auto control deal with the stimulus presentation.

The "Serial Choice – Present Stimulus" stage always resets the stage timer on entry therefore it can be re-entered many times in the same trial and will always last for the full Limited Hold period, however the standard five choice paradigm assumes that one stimulus occurs per trial.

The "Serial Choice – Present Stimulus" stage automatically assigns the niche in which the stimulus occurred as the "target" resource, all remaining "nosepoke" sensors are automatically assigned as "dummy" resources. A response in the correct niche triggers the "Exit On Target" endpoint, a response in any incorrect niches triggers the "Exit On Dummy" endpoint.

**5.2.13 Serial Choice – Collection Stage**

**Components:**
- Driver Controls
- Display Message
- Target/Dummy Settings
- Stage Exit Triggers
- Stage Reports


**Function**
A "Serial Choice – Collection" stage is specifically designed to perform the "Wait For Collection" function in the five-choice serial reaction-time task.

The stage sets the state of the operant chamber, (normally just the house light), then waits for a response in any sensor. Normally there is no timeout associated with this stage, the protocol pauses indefinitely until the subject makes some form of response.

The "Serial Choice – Collection" stage automatically assigns all "nosepoke" sensors in the chamber as "Dummy" resources, i.e. all the niche sensors. A response in any niche during the "Wait For Collection" stage is normally considered a "perseverative" response", i.e. a repeated response to the stimulus. A five choice paradigm normally sends the protocol into a timeout state as a result of a perseverative response.

The developer normally assigns the sensor in the pellet receptacle as the "target" resource. Activation of the target resource triggers the "Exit On Target" endpoint. Activation of any dummy resource triggers the "Exit On Dummy" endpoint.

**5.2.14 Serial Choice – Timeout Stage**

**Components:**
- Driver Controls
- Display Message
- Stage Exit Triggers

**Function**
A "Serial Choice – Timeout" stage is specifically designed to perform the "Timeout" function in the five-choice serial reaction-time task.

The stage sets the state of the operant chamber, (normally just the house light), then waits for a response in any sensor or for the stage timeout to occur.

The "Serial Choice – Timeout" stage inherits the target/dummy assignments of the serial choice stage that precedes it in the stage list, (either a "Serial Choice – Inter-Trial Interval" stage or a "Serial Choice – Collection" stage), the "Reset Timeout" option causes the stage to reset its timeout if a response on any "dummy" resource is made, i.e. repeated premature or perseverative responses.  The "Exit On Target" & "Exit On Dummy" endpoints are usually disabled.

The developer normally assigns the timeout stage as a resetting 5 second timeout, where the subject is required to "do nothing" for a minimum of 5 seconds in order for the protocol to progress to the next stage.

### 5.2.15 Serial Choice – Training Stage

**Components:**
- Driver Controls
- Display Message
- Target Position
- Stage Exit Triggers
- Stage Reports

**Function**
A "Serial Choice – Training" stage is specifically designed to assist the developer in defining five choice training protocols.

The stage sets the state of the operant chamber, (normally just the house light), then illuminates any niche that has been defined as a "target", (in the "Target Position" options). The stage then waits for a response in any sensor or for the optional stage timeout to occur.

The general intention of the "Serial Choice – Training" stage is to create simple continuous reinforcement, (CRF), protocols, where all stimulus lights are illuminated and a response in any niche results in a pellet delivery.

**5.2.16 Interval Stage**

**Components:**
- Driver Controls
- Display Message
- Target/Dummy Settings
- Stage Exit Triggers
- Interval Controls
- Stage Reports

**Function**

An "Interval" stage provides a period sub-divided into smaller intervals. An "Interval" stage is similar to a Variable Interval stage, however it includes additional control options.

Variable intervals are used commonly in operant tasks where it is desirable to encourage a subject to respond at a high rate.

An "Interval" stage lasts for a period specified in the "Stage Exit Trigger" options. The stage timeout is either a fixed timeout, or an "auto timeout", auto timeouts are specified using the "Auto Timeout" parameters of the subject database.

The "intervals" of an "Interval" stage may be either variable intervals extracted from a Variable Interval auto control or fixed intervals, specified in the "Interval Controls" options.

The timing of the intervals is subject to several control options:
- Wait For Target (Initial) – If enabled then the timing of the first interval does not start until the subject activates the target resource
- Reset On Target – If enabled then the timer of the current interval is reset when the subject activates the target resource.
- Reset On Dummy – If enabled then the timer of the current interval is reset when the subject activates the dummy resource.
- Exit On Target – Trigger an endpoint when the subject activates the target resource.
- Exit On Dummy – Trigger an endpoint when the subject activates the target resource.

An interval is marked as "active" if the subject activates the target resource during the interval, otherwise the interval is considered "inactive". "Active" and "inactive" intervals have the additional controls:
- Trigger On Interval
- Wait For Target

If "Trigger On Interval" is selected then the interval trigger endpoint occurs immediately the interval timeout occurs, if "Wait For Target" is selected then the interval trigger endpoint occurs on the first activation of the target resource after the interval timeout.

"Interval" stage also include the following controls:
- On Stage Exit – Reset Timer Control
- On New Trial – Reset Auto Control

If "Reset Timer Control" is enabled then whenever the protocol leaves the stage the interval timer is reset, so on the next entry to the stage a full interval is required, otherwise if the stage is entered and then re-entered the interval timer continues as if a stage exit did not occur.

If "Reset Auto Control" is active then on a new trial the sequence of interval values repeats, (useful where the variable interval control is specified as a table of delay values rather than as random variable intervals).

The variable interval settings are specified by the variable interval parameters of the subject database. A VI stage contains stage timeout exit triggers and an "end of interval" trigger. The developer specifies the "end of interval" trigger to occur either immediately the interval timeout occurs, (Trigger On Interval) or on the first response that occurs after the interval timeout, (Wait For Target).

A VI stage may be iterated, (looped), many times during the same trial.  The stage timeout begins on the first entry to the VI stage in any one trial; it is not reset on subsequent entries to the stage during the same trial.  This allows the protocol to define a "VI phase" of a specified length.

Consider a VI stage with a fixed timeout of 5 minutes, each time the end of interval exit is triggered the protocol exits the VI stage, passes through a reward stage, (to trigger the pellet dispenser) and then loops back and re-enters the VI stage.  The sequence of interval complete/reward delivery may be repeated many times during the five minute period, at the end of the 5 minutes the stage exits via the stage timeout exit trigger and moves on to another part of the protocol

For an example of the use of "Interval" stages refer to the Go/Nogo protocol in the K-Limbic repository.

**5.2.17 Delay Stage**

**Components:**
- Driver Controls
- Display Message
- Target/Dummy Settings
- Stage Exit Triggers
- Delay Controls
- Interval Controls
- Stage Reports

**Function**

A "Delay" stage provides a period sub-divided into one or more "Delays". It is similar to an "Interval" stage with a fixed interval, but the "Delay" value is specified in the subject database, (from a "Delay Table").

A "Delay" stage lasts for a fixed period or for a specified number of iterations, (i.e. the number of "loops" through the stage.

The "Delay" timeout is either as specified using the Delay Table of the subject database, or is a specified fixed delay value.

The Delay endpoint trigger either occurs immediately the delay timeout occurs or on the first activation of the target resource after the delay timeout.

The following delay endpoint options are available:
- Trigger On Delay – Trigger the "End Of Delay" endpoint on the delay timeout
- Trigger On Target – Trigger the "End Of Delay" endpoint on the first activation of the target resource after the end of the delay.

The timing of the delays is subject to control options:
- Reset On Target – If enabled then the timer of the current interval is reset when the subject activates the target resource.
- Reset On Dummy – If enabled then the timer of the current interval is reset when the subject activates the dummy resource.

On exit of the stage the delay timer can be reset so that a new entry to the stage starts with the delay timer reset.

### 5.2.18 Progressive Ratio Stage

**Components:**
- Driver Controls
- Display Message
- Target/Dummy Settings
- Stage Exit Triggers
- Stage Reports

**Function**
A "Progressive Ratio" stage sets the state of the operant chamber, then waits until a specified number of responses, (the fixed ratio), are made on the target/dummy sensor or a timeout occurs.

As the name implies each time a ratio is completed the target count for the next ratio increases, the sequence of increasing ratio values is determined by the progressive ratio parameters of the subject database.

The stage timeout is reset on each entry to the stage, even if the stage is "iterated" repeatedly in the same trial. It is good practise to define progressive ratio protocols so that the trial is incremented after each successful completion of a ratio, (this will result in an output file with details of performance on each ratio on each row of the output file).

**5.2.19 Cued Choice Stage**

**Components:**
- Driver Controls
- Display Message
- Cue Light States
- Target/Dummy Settings
- Stage Exit Triggers
- Stage Reports

**Function**
"Cued Choice" stages are designed for protocols running in chambers with "nose-poke" niches, e.g. a five choice chamber.

"Cued Choice" stages were originally designed for use in protocols that run the DMTP paradigm in a five choice chamber.  The DMTP protocol is essentially the same as the lever based DMTP protocol however "Cued Choice" stages should be used for the "Present Sample" and "Choice" stages.

A "Cued Choice Stage" sets the state of the operant chamber, then waits for a response or a timeout.

The Target/Dummy settings are extracted from the DMTP auto control, define two or more "Nosepoke" resources, (instead of the left & right levers) in the Standard and Reverse target lists.

 "Cued Choice" stages always reset the stage timer on entry therefore they can be re-entered many times in the same trial and will always last for the full timeout period.

For an example of the use of "Cued Choice" stages refer to the K-Limbic repository.

### 5.2.20 Sequential Choice Stage

**Components:**
- Driver Controls
- Display Message
- Target/Dummy Settings
- Stage Exit Triggers
- Stage Reports

**Function**

"Sequential Choice" stages are designed for protocols running in chambers with "nose-poke" niches, e.g. a five choice chamber.

"Sequential Choice" stages are designed to implement protocols where a sequence of cues are presented to the subject, followed by the presentation of 2 or more of the cues. The sequence and choice parameters are determined by the "Sequential Choice" database parameters.

A "Cued Choice Stage" sets the state of the operant chamber, then waits for a response or a timeout.

"Cued Choice" stages always reset the stage timer on entry therefore they can be re-entered many times in the same trial and will always last for the full timeout period.

For an example of the use of "Cued Choice" stages refer to the K-Limbic repository.

### 5.2.21 Cued Lever Stage

**Components:**
- Driver Controls
- Display Message
- Task Setting
- Stage Exit Triggers
- Stage Reports

**Function**
A "Cued Lever" stage is specifically designed to cater for the "Cued Lever – Serial Reversal" protocol.

The stage interacts with the "Cued Lever" auto control to determine the current setting for the target/dummy levers and cue lights.

The stage sets the state of the operant chamber, then waits for a response or a timeout.

 "Cued Lever" stages always reset the stage timer on entry therefore they can be re-entered many times in the same trial and will always last for the full timeout period.  The standard cued lever – serial reversal protocols define the flow so that each cued lever stage is used only once per trial.

For an example of the use of "Cued Choice" stages refer to the K-Limbic repository.

### 5.2.22 Epoch Control Stage

**Components:**
- Epoch Controls
- Display Message

**Function**
An "Epoch Control" stage is a protocol management stage.  The stage is used to mark the beginning of an "Epoch" within a protocol.  The duration of the "Epoch" is specified in the "Epoch Control" stage.

If an "Epoch" is running then any stage can use the "Exit On Epoch" stage endpoint as a means of ending the stage.  The protocol manager keeps a track of the active epoch, when it times out if the current stage has an enabled "Exit On Epoch" endpoint trigger then the stage will exit.

A protocol can use as many epoch control stages as required, although only one epoch can be active at any time.  If an epoch is active, (i.e. started but not yet timed out), when a second epoch control stage starts a new epoch, then the first epoch is abandoned and the epoch timer is reset to the values in the second epoch control stage.

An "Epoch Control" stage is instantaneous; although there is a finite time taken to process an "Epoch Control" stage this time is very short and unperceivable by the subject.

**5.2.23 Outcome stage**

**Components:**
- Trial Outcome
- Display Message

**Function**
An "Outcome" stage is a protocol management stage.  The stage is used to mark the outcome of a trial, (e.g. "Correct", "Incorrect", "Missed" etc).

An "Outcome" stage is instantaneous; although there is a finite time taken to process an "Outcome" stage this time is very short and unperceivable by the subject.

A protocol can use an outcome stage to mark the performance of a trial.  The flow of a protocol will vary depending on whether a subject performs the trial correctly, incorrectly, or not at all etc, the protocol can contain multiple outcome stages at different points in the protocol to mark the performance of the subject.

A "Mark Trial/Set Outcome" stage also performs this function, but it also affects the trial count.

An example of the use of "Outcome" stages is the DMTP protocol, where the trial performance is either "correct", "incorrect" or "missed".  The DMTP protocol uses three outcome stages at various points to mark the trial as one of the tree options.  The "Increment Trial" stage, (a "Mark Trial/Set Outcome" stage), then simply increments the trial count without making any changes to the stage outcome.

A protocol should be arranged so that each trial passes through one stage that sets the outcome per trial, i.e. the protocol flow should not pass through an "Outcome" stage that say sets the trial outcome then later in the same trial passes through a second "Outcome" stage.

**5.2.24 DRL Stage**

**Components:**
- Driver Controls
- Display Message
- Target/Dummy Settings
- Stage Exit Triggers
- Stage Reports

**Function**
A "DRL" stage is specifically designed to cater for the "DRL", (delayed reinforcement), protocol.

The stage sets the state of the operant chamber, then waits for a response or a timeout.

On entry the DRL stage enters a "delay" phase, any response on the target resource during the delay phase is considered an "incorrect" response. The stage provides an incorrect response endpoint so that the stage may exit when an incorrect response is made. The duration of the "delay" is determined by the DRL auto control.

At the end of the delay phase the stage enters a "waiting", (for response), phase. The waiting phase lasts from the end of the delay until the stage timeout. The stage timeout is set either as a fixed timeout or via the DRL parameters of the subject database.

The subject must learn to delay any response to the lever for the duration of the "Delay" period in order to obtain a reward. A fully trained subject would operate at a typical delay of 72 seconds.

The stage interacts with the "DRL" auto control to determine the current settings for the delay period and stage timeout and number of trials.

The DRL auto control includes an "Enable Peak Trials" option. A "Peak" trial is a trial in which the "delay" phase lasts longer than the normal delay period, (a peak trial has a delay period of 6 minutes in K-Limbic), the delay phase of a peak trial normally lasts for the entire duration of the DRL stage. If peak trials are enabled then each bank of trials will include one peak trial.

For an example of the use of "DRL" stages refer to the K-Limbic repository.

**5.2.25 Timeout Stage**

**Components:**
- Driver Controls
- Display Message
- Stage Exit Triggers
- Stage Reports

**Function**
A "Timeout" stage is provides a flexible method of providing a database dependent timeout period.

In many behavioural tests it is necessary to react to an incorrect or unwanted response by sending the protocol into a timeout period; i.e. a period in which the protocol pauses before continuing.

Any stage type can be used to provide a timeout period, simply by setting the driver controls to the appropriate values and enabling one of the stage timeout endpoints, e.g. "Action" stages can easily be used for this purpose using a fixed timeout endpoint.

"Timeout" stages have the advantage that the timeout value can be specified by the "Auto Timeout" parameters of the subject database.  Using the subject database provides flexible setting of the timeout period without the requirement to change the protocol definition.

The stage sets the state of the operant chamber, then waits for the timeout period.

### 5.2.26 Location Stage

**Components:**
- Driver Controls
- Display Message
- Stage Exit Triggers
- Target Position
- Stage Reports

**Function**

A "Location" stage is provided to monitor the current location of a subject. "Location" stages are useful in protocols that are required to monitor and/or react to the position of a subject within an environment.

An example of monitoring the position of a subject would be a "Passive Avoidance" test.

A "Passive Avoidance" test environment is split into "light" and "dark" chambers separated by a door. The subject is placed into the "light" chamber with the door closed. The door opens at the start of the test and system monitors the time it takes the animal to cross into the "dark" chamber.

Position sensors monitor the current location of the subject; light beams are often used as position sensors.

In a location stage the developer defines a "target" position by specifying a desired state for each position sensor. When the subject moves to the location that provides the desired combination of states, the stage triggers the "Exit On Target Position" endpoint.

A position sensor may be:
- Active
- Inactive
- Don't Care

A position sensor is "Active" when the sensor detects the presence of the subject.
A position sensor is "Inactive" when the sensor does not detect the presence of the subject.

A location stage can also trigger an endpoint when any of the position sensors changes state, ("Exit On Any Sensor"). This is useful if you wish to start the timing from when you place the subject into the environment. In this case you start the session before placing the subject into the environment, the protocol enters a Location stage with the "Exit On Any Sensor" endpoint enabled, ("Wait for Subject"). When you place the subject into the environment it is normal for one or more position sensors to become active, this will trigger the end of the "Wait For Subject" stage and the protocol can begin it's timing.

A protocol that uses any "Location" stages should not enable the "Message Filter", (see the "Initial" stage description).

<div align="right">

Section

6

**Output File Format**

</div>

The output file generated by K-Limbic is a comma-separated variables, (CSV), file, suitable for import to common spreadsheet packages such as MS Excel, or databases, such as MS Access. This section details the components within the output file for those users who wish to analyse their data files using their own data analysis utilities.

The automated analysis facility offered by Conclusive Solutions uses MS Visual Basic hosted within MS Excel. Those opting to use one of the automated analysis packages obtain output as an Excel workbook containing customised results and performance histories; the CSV files are used only for raw data storage.

**Generated Files: Session Files**
K-Limbic creates one output file for each session it runs, the session output file contains the raw data for each subject run during the session.
The session file is stored in the data folder specified in the session configuration.
The filename is automatic and of the form:
> 01-Jan-2006_nnn.csv

the nnn value is a number between 001 & 999, allowing K-Limbic to store a total of 999 sessions of data per day.

**Generated Files: Combined Files**
In addition to the session files, K-Limbic also maintains a "combined" file. The combined file contains all the session files for any given day. The combined file reduces the data handling considerably as only one data file per day is required for raw data storage, (or analysis).
The combined file is stored in the data folder specified in the session configuration.
The filename is automatic and of the form:
> 01-Jan-2006_Combined.csv

An Excel file is included with the developers guide, (filename "Output File.xls"), please refer to the example file for examples of each portion of the output file.

**File Content**
The contents of an output file consists of a number of sections, these sections are:

- Header Data
- Auto Control Data
- Trials Data
- Activity Log Data

**Header Data**
The header data consists of a number of items that are output from all protocols.
All output files contain a header section.

The "STARTDATA" cell marks the beginning of the data for one subject.  All data entries commence with this cell.
The "KLIMBIC OUTPUT FILE" and "MS EXCEL FORMAT" cells also appear in all data entries.
The "System" entry records the name of the K-Limbic system that created the file.
The "date" entry records the date of the session.
The "Box Index" entry records the number of the box.
The "ExptType" entry = 0 for a "Control" experiment & = 1 for an "Active" experiment, (this information is extracted from the dosing details.
The "Session Title" records the title of the session.
The "Protocol Title" records the title of the protocol.
The "Protocol Description" records the description field of the protocol.
The "Analysis Ref" records the reference of the automated analysis package used to analyse this data.
The "Dose Details" entry is followed underneath by the fields from the does details, e.g. dose description, pre-dose period & dose category.  The index value is the index of the selected dose in the dose table.
The "Subject Id" is the identifier of the subject.
The "DB Ref" is the reference of the database of the subject.
The "Duration" field is the total duration of the experiment, (in centi-seconds).
The "Pellet Count" is the total of all pellets delivered during the experiment
The "Fluid Count" is the total of all fluid rewards delivered during the experiment
The "Air Puff Count" is the total of all air puffs delivered during the experiment
The "Air Puff Count" is the total of all air puffs delivered during the experiment
The "Shock Count" is the total number of shocks delivered during the experiment
The "F(10) Sensor" identifies the sensor that was first activated 10 times.
The "SL Target" identifies the index of the "target" lever from the selected lever parameters.
The "SL Dummy" identifies the index of the "dummy" lever from the selected lever parameters.
The "SS Target" identifies the index of the "target" sensor from the selected sensor parameters.
The "SS Dummy" identifies the index of the "dummy" sensor from the selected sensor parameters.
The "SensorTC" lists the activation counts of each of the sensors, e.g. S(1), S(2) etc

It can be seen that in most cases the majority of the header items are not required for data analysis and can be ignored.
To extract any particular value from the header, search for the identifier in column A and read the value.

**Auto Control Data**

The auto control data lists details of the auto control generated for use with the experiment.
All output files contain an auto control section, even those that do not use an auto control.

The auto control data starts with the "AutoControl" identifier in column A.

The contents of the auto control vary depending on which form of auto control is specified in the
protocol.  The fields of the auto control section should be self-explanatory.

The "Index" field is the index from the database of the chosen parameters, (0 if the entry was not
selected from a database table).  The "AC Comment" is the comment field from the database
parameters.

The auto control data on the "Individual Subject" page of the example file is for a fixed ratio type auto
control, this is a relatively simple report that records the FR schedule, the target index, the dummy
index, the "Reset on Dummy" value, the "Disable Unused" value & the "Enable Rewards" value.

The auto control data on the "Auto Control Data" illustrates the auto control data section for a 100 trial
five choice experiment.

The "Choice Details" fields, record the index of the selected parameters from the five choice table,
(same as the "Index" field above), the "Sensors" field records the number of active sensors, e.g. 5.  The
"Iterations" field records the value of the "Iterations" field from the database.  The "banked" field = 1 if
banking is enabled, =0 if banking is disabled.  The "Trials" field records the number of trials in the auto
control, (normally the maximum number of trials in the session).

Following the choice details are the trial details.  The trial details contain 100 columns, (1 per trial), that
list the bank, group, stimulus duration (centi-seconds), channel number, intensity, inter-trial interval
(centi-seconds), tone-timing index, tone burst duration (x2mS), and the hold-time (centiseconds), for
each trial.

(The "Group" entry = 0 in all "Table Mode" auto controls, otherwise the "Group" entry is set to the table
index of the current parameter set).

Auto Control Data sections vary for each type of auto control, if in any doubt as to the content of this
section please contact Conclusive Solutions.

**Trials Data**
The trials data section lists the output reports from the stages.
All output files contain a trials data section.

The Trials data section commences with the "Trials" identifier, then comes the output reports header that details the contents of each column.  Below the output reports header are the output reports themselves, split into one row per "trial".

The size of the trials data section depends on the number of output reports identified in the protocol definition and the number of trials performed during the session.

The first three columns of the data are always:
Trial Ref: The index number of the trial
Outcome: A numerical value identifying the trial "Outcome", (see below).
Item Index: The index into the auto control used for the trial.

The remaining columns are determined by the stage reports settings in the protocol.

The "Trial Ref" is a zero based index of the trial number.  The "Item Index" is generally the same as the "Trial Ref", it is the index into the trials table used by the current trial, it increments at the end of each trial, however if a trial is repeated the "Item Index" does not increment.

The following are the numerical codes for each trial outcome.

| | |
|---:|---|
| Correct Trial | 0 |
| Incorrect Trial | 1 |
| Missed Trial | 2 |
| Repeat Trial | 3 |
| Incomplete Trial | 4 |
| Rewards Endpoint | 5 |
| Idle Timeout | 6 |
| No Reward Timeout | 7 |
| Protocol Timeout | 128 |
| Trials Complete | 150 |
| User Termination | -1 |

All timings are reported in centiseconds and are relative to the start of the protocol.

The "Trials Data" sheet of the example file illustrates the Trials Data section for a 100 trial five choice test.

The "Individual Subject" sheet of the example file illustrates the Trials Data section for a fixed ratio task that uses a single trial, the trials data therefore becomes a summary over the entire session.

**Activity Log Data**
The Activity Log section is a list of the activity reports created during the test.
Not all output files contain an Activity Log Data section.
Enable the activity log section by enabling the "Activity Log" in the session configuration.

Refer to the "Activity Log Data" page of the Output File.xls file for an example activity log.

All rows in the activity log start with one of the following identifiers…

- 'ACTIVITYLOG'
- 'TRIAL'
- 'DATA'
- 'END'

The activity log section commences with the "ACTIVITYLOG" identifier.
The activity log is split into trials as per the protocol.
The activity log can contain an arbitrary number of reports, each row of data can contain up to 250 reports; each report appears in a single Excel "cell".

Trial zero is the "Start" stage, i.e. before the protocol begins, this log will not contain any useful reports, it just confirms that the Start stage, (stage type "Initial Stage"), was used prior to the protocol start.

Trial 1 is the first trial in the session, it will contain all activity reports generated in the trial.
The "DATA" identifier appears in column A, followed by a row of up to 250 activity reports, if more than 250 activity reports were generated during the trial the log wraps onto the next row.
The final activity report in the trial is followed by the 'EOD' identifier.

The end of the activity log section is marked by the 'END' identifier.

**Activity log reports**
The activity log can contain the following reports:

**Start of session report**
Report appears once per session and will always be as below
**2000;PS;0;**

**End of session report**
Report to mark the end of the session.
**2001;PE;180000;**

The '180000' value is the time at which the session ended (in centiseconds), (30 minutes).

**Response Activated**
Report generated each time a sensor is activated
**2002;ON;1;S(0);7685;**

The value 1 is the index of the current stage.  The index is zero based, thus stage 1 is reported as 0, stage 2 is reported as 1 etc.
S(n) identifies that sensor S(n) was activated, (the index "n" is zero based, thus sensor <1> is reported as S(0), sensor <2> is reported as S(1) etc).
The activation time was 76.85 seconds after the start of the session

**Response Released**
Report generated each time a sensor is released
**2003;OFF;1;S(0);8055;**

The value 1 is the index of the current stage.  The index is zero based, thus stage 1 is reported as 0, stage 2 is reported as 1 etc.
S(n) identifies that sensor S(n) was released, (the index "n" is zero based, thus sensor <1> is reported as S(0), sensor <2> is reported as S(1) etc).
The release time was 80.55 seconds after the start of the session

Please note that if the message filter is enabled "Response Released" activity reports will not be generated.

**Stage Entry**
Report generated whenever a stage is entered
**2004;STIN;4;8877;**

The value "4" is the index of the new stage.  The index is zero based, thus stage 1 is reported as 0, stage 2 is reported as 1 etc.
Here stage 5 was entered 88.77 seconds after the start of the session.

**Stage Exit**
Report generated by the end of a stage.
**2005;STOUT;3;9866;2;**

The value "3" is the index of the current stage.  The index is zero based, thus stage 1 is reported as 0, stage 2 is reported as 1 etc.
Here stage 4 was exited 98.66 seconds after the start of the session.
The end of the stage was triggered by exit trigger number 2.

**Reward Dispensed**
Report generated whenever a reward is dispensed
2006;DS;2;F(0);2345;

The value "2" is the index of the current stage.  The index is zero based, thus stage 1 is reported as 0, stage 2 is reported as 1 etc.
F(0) identifies that fluid dispenser <1> dispensed the reward, 23.45 seconds after the start of the session.